

# Mobile-C

## 2.0.7

Generated by Doxygen 1.6.3

Tue Dec 28 15:15:25 2010



# Contents

<b>1</b>	<b>MobileC</b>	<b>1</b>
1.1	Introduction . . . . .	1
<b>2</b>	<b>Data Structure Index</b>	<b>3</b>
2.1	Data Structures . . . . .	3
<b>3</b>	<b>File Index</b>	<b>5</b>
3.1	File List . . . . .	5
<b>4</b>	<b>Data Structure Documentation</b>	<b>9</b>
4.1	_hr_time Struct Reference . . . . .	9
4.1.1	Detailed Description . . . . .	9
4.1.2	Field Documentation . . . . .	9
4.1.2.1	start . . . . .	9
4.2	_ssl_context Struct Reference . . . . .	10
4.2.1	Detailed Description . . . . .	11
4.2.2	Field Documentation . . . . .	11
4.2.2.1	authmode . . . . .	11
4.2.2.2	ca_chain . . . . .	11
4.2.2.3	ciphers . . . . .	11
4.2.2.4	client_auth . . . . .	12
4.2.2.5	ctx_dec . . . . .	12
4.2.2.6	ctx_enc . . . . .	12
4.2.2.7	dhm_ctx . . . . .	12
4.2.2.8	do_crypt . . . . .	12
4.2.2.9	endpoint . . . . .	12
4.2.2.10	f_dbg . . . . .	12
4.2.2.11	f_recv . . . . .	13
4.2.2.12	f_rng . . . . .	13

4.2.2.13	f_send	13
4.2.2.14	fin_md5	13
4.2.2.15	fin_sha1	13
4.2.2.16	hostname	13
4.2.2.17	hostname_len	13
4.2.2.18	in_ctr	14
4.2.2.19	in_hdr	14
4.2.2.20	in_hslen	14
4.2.2.21	in_left	14
4.2.2.22	in_msg	14
4.2.2.23	in_msglen	14
4.2.2.24	in_msgtype	15
4.2.2.25	in_offt	15
4.2.2.26	iv_dec	15
4.2.2.27	iv_enc	15
4.2.2.28	ivlen	15
4.2.2.29	keylen	15
4.2.2.30	mac_dec	15
4.2.2.31	mac_enc	16
4.2.2.32	maclen	16
4.2.2.33	major_ver	16
4.2.2.34	max_major_ver	16
4.2.2.35	max_minor_ver	16
4.2.2.36	minlen	16
4.2.2.37	minor_ver	17
4.2.2.38	nb_zero	17
4.2.2.39	out_ctr	17
4.2.2.40	out_hdr	17
4.2.2.41	out_left	17
4.2.2.42	out_msg	17
4.2.2.43	out_msglen	18
4.2.2.44	out_msgtype	18
4.2.2.45	own_cert	18
4.2.2.46	p_dbg	18
4.2.2.47	p_rcv	18
4.2.2.48	p_rng	18

4.2.2.49	p_send	19
4.2.2.50	peer_cert	19
4.2.2.51	peer_cn	19
4.2.2.52	pmslen	19
4.2.2.53	premaster	19
4.2.2.54	randbytes	19
4.2.2.55	resume	19
4.2.2.56	rsa_key	20
4.2.2.57	s_get	20
4.2.2.58	s_set	20
4.2.2.59	session	20
4.2.2.60	state	20
4.2.2.61	timeout	21
4.2.2.62	verify_result	21
4.3	_ssl_session Struct Reference	22
4.3.1	Detailed Description	22
4.3.2	Field Documentation	22
4.3.2.1	cipher	22
4.3.2.2	id	22
4.3.2.3	length	22
4.3.2.4	master	22
4.3.2.5	next	23
4.3.2.6	start	23
4.4	_x509_buf Struct Reference	24
4.4.1	Detailed Description	24
4.4.2	Field Documentation	24
4.4.2.1	len	24
4.4.2.2	p	24
4.4.2.3	tag	24
4.5	_x509_cert Struct Reference	25
4.5.1	Detailed Description	25
4.5.2	Field Documentation	25
4.5.2.1	ca_istrue	25
4.5.2.2	issuer	25
4.5.2.3	issuer_id	25
4.5.2.4	issuer_raw	26

4.5.2.5	max_pathlen	26
4.5.2.6	next	26
4.5.2.7	pk_oid	26
4.5.2.8	raw	26
4.5.2.9	rsa	26
4.5.2.10	serial	26
4.5.2.11	sig	26
4.5.2.12	sig_oid1	27
4.5.2.13	sig_oid2	27
4.5.2.14	subject	27
4.5.2.15	subject_id	27
4.5.2.16	subject_raw	27
4.5.2.17	tbs	27
4.5.2.18	v3_ext	27
4.5.2.19	valid_from	27
4.5.2.20	valid_to	27
4.5.2.21	version	28
4.6	_x509_name Struct Reference	29
4.6.1	Detailed Description	29
4.6.2	Field Documentation	29
4.6.2.1	next	29
4.6.2.2	oid	29
4.6.2.3	val	29
4.7	_x509_node Struct Reference	30
4.7.1	Detailed Description	30
4.7.2	Field Documentation	30
4.7.2.1	data	30
4.7.2.2	end	30
4.7.2.3	len	30
4.7.2.4	p	30
4.8	_x509_raw Struct Reference	31
4.8.1	Detailed Description	31
4.8.2	Field Documentation	31
4.8.2.1	issuer	31
4.8.2.2	raw	31
4.8.2.3	serial	31

4.8.2.4	sign	31
4.8.2.5	signalg	31
4.8.2.6	subject	31
4.8.2.7	subpubkey	32
4.8.2.8	tbs	32
4.8.2.9	tbs_signalg	32
4.8.2.10	validity	32
4.8.2.11	version	32
4.9	_x509_time Struct Reference	33
4.9.1	Detailed Description	33
4.9.2	Field Documentation	33
4.9.2.1	day	33
4.9.2.2	hour	33
4.9.2.3	min	33
4.9.2.4	mon	33
4.9.2.5	sec	33
4.9.2.6	year	34
4.10	aes_context Struct Reference	35
4.10.1	Detailed Description	35
4.10.2	Field Documentation	35
4.10.2.1	buf	35
4.10.2.2	nr	35
4.10.2.3	rk	35
4.11	agency_s Struct Reference	36
4.11.1	Detailed Description	36
4.11.2	Field Documentation	36
4.11.2.1	agentInitCallback	36
4.11.2.2	agentInitUserData	36
4.11.2.3	bluetooth	36
4.11.2.4	client	37
4.11.2.5	default_agentstatus	37
4.11.2.6	enable_security	37
4.11.2.7	filename	37
4.11.2.8	hostName	37
4.11.2.9	initInterps	37
4.11.2.10	known_host_filename	37

4.11.2.11 last_error . . . . .	37
4.11.2.12 mc_platform . . . . .	38
4.11.2.13 portno . . . . .	38
4.11.2.14 portnoc . . . . .	38
4.11.2.15 priv_key_filename . . . . .	38
4.11.2.16 server . . . . .	38
4.11.2.17 stack_size . . . . .	38
4.11.2.18 threads . . . . .	39
4.12 agent_datastate_s Struct Reference . . . . .	40
4.12.1 Detailed Description . . . . .	40
4.12.2 Field Documentation . . . . .	40
4.12.2.1 agent_code . . . . .	40
4.12.2.2 agent_code_ids . . . . .	40
4.12.2.3 agent_codes . . . . .	40
4.12.2.4 init_agent_status . . . . .	41
4.12.2.5 number_of_tasks . . . . .	41
4.12.2.6 persistent . . . . .	41
4.12.2.7 progress_modifier . . . . .	41
4.12.2.8 return_data . . . . .	41
4.12.2.9 task_progress . . . . .	41
4.12.2.10 tasks . . . . .	41
4.12.2.11 xml_agent_root . . . . .	42
4.12.2.12 xml_root . . . . .	42
4.13 agent_mailbox_s Struct Reference . . . . .	43
4.13.1 Detailed Description . . . . .	43
4.13.2 Field Documentation . . . . .	43
4.13.2.1 mail_queue . . . . .	43
4.14 agent_s Struct Reference . . . . .	44
4.14.1 Detailed Description . . . . .	44
4.14.2 Field Documentation . . . . .	44
4.14.2.1 agent_address . . . . .	44
4.14.2.2 agent_interp . . . . .	45
4.14.2.3 agent_persistent . . . . .	45
4.14.2.4 agent_pipe_active . . . . .	45
4.14.2.5 agent_pipe_ready_to_read . . . . .	45
4.14.2.6 agent_ready_to_send . . . . .	45



4.14.2.7	agent_script_ready	45
4.14.2.8	agent_status	45
4.14.2.9	agent_thread	46
4.14.2.10	agent_thread_id	46
4.14.2.11	agent_type	46
4.14.2.12	arrival_time	46
4.14.2.13	binary	46
4.14.2.14	connect_id	46
4.14.2.15	datastate	46
4.14.2.16	home	47
4.14.2.17	home_port	47
4.14.2.18	id	47
4.14.2.19	lock	47
4.14.2.20	mailbox	47
4.14.2.21	mc_platform	47
4.14.2.22	name	47
4.14.2.23	orphan	48
4.14.2.24	owner	48
4.14.2.25	return_data	48
4.14.2.26	run_lock	48
4.14.2.27	sender	48
4.14.2.28	wg_code	48
4.15	agent_task_s Struct Reference	49
4.15.1	Detailed Description	49
4.15.2	Field Documentation	49
4.15.2.1	agent_return_data	49
4.15.2.2	agent_variable_list	49
4.15.2.3	code_id	49
4.15.2.4	init_agent_status	50
4.15.2.5	num_saved_variables	50
4.15.2.6	number_of_elements	50
4.15.2.7	persistent	50
4.15.2.8	saved_variables	50
4.15.2.9	server_name	50
4.15.2.10	size_of_element_array	50
4.15.2.11	var_name	50

4.16	AP_GENERIC_s Struct Reference	52
4.16.1	Detailed Description	52
4.16.2	Field Documentation	52
4.16.2.1	none	52
4.17	arc4_context Struct Reference	53
4.17.1	Detailed Description	53
4.17.2	Field Documentation	53
4.17.2.1	m	53
4.17.2.2	x	53
4.17.2.3	y	53
4.18	barrier_node_s Struct Reference	54
4.18.1	Detailed Description	54
4.18.2	Field Documentation	54
4.18.2.1	cond	54
4.18.2.2	id	54
4.18.2.3	lock	54
4.18.2.4	num_registered	54
4.18.2.5	num_waiting	54
4.19	barrier_queue_s Struct Reference	55
4.19.1	Detailed Description	55
4.19.2	Field Documentation	55
4.19.2.1	list	55
4.19.2.2	lock	55
4.19.2.3	size	55
4.20	cmd_prompt_s Struct Reference	56
4.20.1	Detailed Description	56
4.20.2	Field Documentation	56
4.20.2.1	thread	56
4.21	command_s Struct Reference	57
4.21.1	Detailed Description	57
4.21.2	Field Documentation	57
4.21.2.1	args	57
4.21.2.2	index	57
4.21.2.3	num_args	57
4.22	connection_s Struct Reference	58
4.22.1	Detailed Description	58

4.22.2	Field Documentation	58
4.22.2.1	addr	58
4.22.2.2	AES_key	58
4.22.2.3	clientfd	58
4.22.2.4	connect_id	58
4.22.2.5	nonce	58
4.22.2.6	remote_hostname	59
4.22.2.7	serverfd	59
4.23	fipa_expression_s::content_u Union Reference	60
4.23.1	Detailed Description	60
4.23.2	Field Documentation	60
4.23.2.1	datetime	60
4.23.2.2	expression	60
4.23.2.3	number	60
4.23.2.4	string	60
4.23.2.5	word	60
4.24	des3_context Struct Reference	62
4.24.1	Detailed Description	62
4.24.2	Field Documentation	62
4.24.2.1	mode	62
4.24.2.2	sk	62
4.25	des_context Struct Reference	63
4.25.1	Detailed Description	63
4.25.2	Field Documentation	63
4.25.2.1	mode	63
4.25.2.2	sk	63
4.26	dhm_context Struct Reference	64
4.26.1	Detailed Description	64
4.26.2	Field Documentation	64
4.26.2.1	G	64
4.26.2.2	GX	64
4.26.2.3	GY	64
4.26.2.4	K	65
4.26.2.5	len	65
4.26.2.6	P	65
4.26.2.7	RP	65

4.26.2.8	X	65
4.27	dynstring_s Struct Reference	66
4.27.1	Detailed Description	66
4.27.2	Field Documentation	66
4.27.2.1	len	66
4.27.2.2	message	66
4.27.2.3	size	66
4.28	fipa_acl_envelope_Received_s Struct Reference	67
4.28.1	Detailed Description	67
4.28.2	Field Documentation	67
4.28.2.1	received_by	67
4.28.2.2	received_date	67
4.28.2.3	received_from	67
4.28.2.4	received_id	67
4.28.2.5	received_via	67
4.29	fipa_acl_envelope_s Struct Reference	69
4.29.1	Detailed Description	69
4.29.2	Field Documentation	69
4.29.2.1	num_params	69
4.29.2.2	params	69
4.30	fipa_acl_message_s Struct Reference	70
4.30.1	Detailed Description	70
4.30.2	Field Documentation	70
4.30.2.1	content	70
4.30.2.2	conversation_id	70
4.30.2.3	encoding	70
4.30.2.4	in_reply_to	71
4.30.2.5	language	71
4.30.2.6	ontology	71
4.30.2.7	performative	71
4.30.2.8	protocol	71
4.30.2.9	receiver	71
4.30.2.10	receiver_num	71
4.30.2.11	reply_by	72
4.30.2.12	reply_to	72
4.30.2.13	reply_with	72

4.30.2.14 sender . . . . .	72
4.31 fipa_acl_Param_s Struct Reference . . . . .	73
4.31.1 Detailed Description . . . . .	73
4.31.2 Field Documentation . . . . .	73
4.31.2.1 acl_representation . . . . .	73
4.31.2.2 comments . . . . .	73
4.31.2.3 date . . . . .	73
4.31.2.4 from . . . . .	73
4.31.2.5 intended_receiver . . . . .	74
4.31.2.6 payload_encoding . . . . .	74
4.31.2.7 payload_length . . . . .	74
4.31.2.8 received . . . . .	74
4.31.2.9 to . . . . .	74
4.32 fipa_agent_identifier_s Struct Reference . . . . .	75
4.32.1 Detailed Description . . . . .	75
4.32.2 Field Documentation . . . . .	75
4.32.2.1 addresses . . . . .	75
4.32.2.2 name . . . . .	75
4.32.2.3 resolvers . . . . .	75
4.33 fipa_agent_identifier_set_s Struct Reference . . . . .	76
4.33.1 Detailed Description . . . . .	76
4.33.2 Field Documentation . . . . .	76
4.33.2.1 fipa_agent_identifiers . . . . .	76
4.33.2.2 num . . . . .	76
4.33.2.3 retain_order . . . . .	76
4.34 fipa_comm_action_t Struct Reference . . . . .	77
4.34.1 Detailed Description . . . . .	77
4.34.2 Field Documentation . . . . .	77
4.34.2.1 callback . . . . .	77
4.34.2.2 check_list . . . . .	77
4.34.2.3 next . . . . .	77
4.34.2.4 reply . . . . .	77
4.34.2.5 type . . . . .	77
4.35 fipa_comm_message_check_t Struct Reference . . . . .	78
4.35.1 Detailed Description . . . . .	78
4.35.2 Field Documentation . . . . .	78

4.35.2.1	macro	78
4.35.2.2	next	78
4.35.2.3	type	78
4.36	fipa_comm_performative_t Struct Reference	79
4.36.1	Detailed Description	79
4.36.2	Field Documentation	79
4.36.2.1	action_list	79
4.36.2.2	default_reply	79
4.37	fipa_comm_protocol_cn_t Struct Reference	80
4.37.1	Detailed Description	80
4.37.2	Field Documentation	80
4.37.2.1	cfp_request	80
4.37.2.2	contract_func	80
4.37.2.3	proposal_func	80
4.37.2.4	queue	80
4.37.2.5	replies	80
4.37.2.6	require_inform	80
4.37.2.7	time_out	81
4.38	fipa_comm_protocol_t Struct Reference	82
4.38.1	Detailed Description	82
4.38.2	Field Documentation	82
4.38.2.1	conversation_id	82
4.38.2.2	error	82
4.38.2.3	info	82
4.38.2.4	network	82
4.38.2.5	next	82
4.38.2.6	state	82
4.38.2.7	type	83
4.39	fipa_comm_reply_t Struct Reference	84
4.39.1	Detailed Description	84
4.39.2	Field Documentation	84
4.39.2.1	content	84
4.39.2.2	type	84
4.40	fipa_comm_t Struct Reference	85
4.40.1	Detailed Description	85
4.40.2	Field Documentation	85

4.40.2.1	agent	85
4.40.2.2	agent_address	85
4.40.2.3	agent_name	85
4.40.2.4	debug	85
4.40.2.5	default_reply	85
4.40.2.6	error	85
4.40.2.7	pCallbacks	86
4.40.2.8	performative	86
4.40.2.9	protocol	86
4.41	fipa_DateTime_s Struct Reference	87
4.41.1	Detailed Description	87
4.41.2	Field Documentation	87
4.41.2.1	day	87
4.41.2.2	hour	87
4.41.2.3	is_utc	87
4.41.2.4	millisecond	87
4.41.2.5	minute	87
4.41.2.6	month	88
4.41.2.7	second	88
4.41.2.8	sign	88
4.41.2.9	year	88
4.42	fipa_expression_s Struct Reference	89
4.42.1	Detailed Description	89
4.42.2	Field Documentation	89
4.42.2.1	content	89
4.42.2.2	type	89
4.43	fipa_list_t Struct Reference	90
4.43.1	Detailed Description	90
4.43.2	Field Documentation	90
4.43.2.1	data	90
4.43.2.2	next	90
4.44	fipa_message_string_s Struct Reference	91
4.44.1	Detailed Description	91
4.44.2	Field Documentation	91
4.44.2.1	message	91
4.44.2.2	parse	91

4.45	<a href="#">fipa_number_s Struct Reference</a>	92
4.45.1	<a href="#">Detailed Description</a>	92
4.45.2	<a href="#">Field Documentation</a>	92
4.45.2.1	<a href="#">str</a>	92
4.46	<a href="#">fipa_string_s Struct Reference</a>	93
4.46.1	<a href="#">Detailed Description</a>	93
4.46.2	<a href="#">Field Documentation</a>	93
4.46.2.1	<a href="#">content</a>	93
4.47	<a href="#">fipa_url_s Struct Reference</a>	94
4.47.1	<a href="#">Detailed Description</a>	94
4.47.2	<a href="#">Field Documentation</a>	94
4.47.2.1	<a href="#">str</a>	94
4.48	<a href="#">fipa_url_sequence_s Struct Reference</a>	95
4.48.1	<a href="#">Detailed Description</a>	95
4.48.2	<a href="#">Field Documentation</a>	95
4.48.2.1	<a href="#">num</a>	95
4.48.2.2	<a href="#">urls</a>	95
4.49	<a href="#">fipa_word_s Struct Reference</a>	96
4.49.1	<a href="#">Detailed Description</a>	96
4.49.2	<a href="#">Field Documentation</a>	96
4.49.2.1	<a href="#">content</a>	96
4.50	<a href="#">foo_c Class Reference</a>	97
4.50.1	<a href="#">Detailed Description</a>	97
4.50.2	<a href="#">Constructor &amp; Destructor Documentation</a>	97
4.50.2.1	<a href="#">foo_c</a>	97
4.50.2.2	<a href="#">~foo_c</a>	97
4.50.3	<a href="#">Member Function Documentation</a>	97
4.50.3.1	<a href="#">get_bar</a>	97
4.50.3.2	<a href="#">get_barfoo</a>	98
4.50.3.3	<a href="#">get_foo</a>	98
4.50.3.4	<a href="#">get_global</a>	98
4.50.3.5	<a href="#">set_bar</a>	98
4.50.3.6	<a href="#">set_foo</a>	98
4.50.3.7	<a href="#">setfoobar</a>	98
4.50.4	<a href="#">Field Documentation</a>	98
4.50.4.1	<a href="#">bar</a>	98



4.50.4.2	barfoo	98
4.50.4.3	foo	99
4.50.4.4	global	99
4.51	foo_s Struct Reference	100
4.51.1	Detailed Description	100
4.51.2	Constructor & Destructor Documentation	100
4.51.2.1	foo_s	100
4.51.2.2	~foo_s	100
4.51.3	Member Function Documentation	100
4.51.3.1	get_bar	100
4.51.3.2	get_foo	100
4.51.3.3	set_bar	101
4.51.3.4	set_foo	101
4.51.4	Field Documentation	101
4.51.4.1	bar	101
4.51.4.2	foo	101
4.52	havege_state Struct Reference	102
4.52.1	Detailed Description	102
4.52.2	Field Documentation	102
4.52.2.1	offset	102
4.52.2.2	pool	102
4.52.2.3	PT1	102
4.52.2.4	PT2	102
4.52.2.5	WALK	102
4.53	host_id_s Struct Reference	104
4.53.1	Detailed Description	104
4.53.2	Field Documentation	104
4.53.2.1	hostname	104
4.53.2.2	port	104
4.54	hr_time Struct Reference	105
4.54.1	Detailed Description	105
4.54.2	Field Documentation	105
4.54.2.1	opaque	105
4.55	interpreter_variable_data_s Struct Reference	106
4.55.1	Detailed Description	106
4.55.2	Field Documentation	106

4.55.2.1	array_dim	106
4.55.2.2	array_extent	106
4.55.2.3	data	106
4.55.2.4	data_type	107
4.55.2.5	name	107
4.55.2.6	size	107
4.56	list_s Struct Reference	108
4.56.1	Detailed Description	108
4.56.2	Field Documentation	108
4.56.2.1	listhead	108
4.56.2.2	size	108
4.57	listNode_s Struct Reference	109
4.57.1	Detailed Description	109
4.57.2	Field Documentation	109
4.57.2.1	next	109
4.57.2.2	node_data	109
4.58	mc_platform_s Struct Reference	110
4.58.1	Detailed Description	110
4.58.2	Field Documentation	111
4.58.2.1	acc	111
4.58.2.2	agency	111
4.58.2.3	agent_queue	111
4.58.2.4	ams	111
4.58.2.5	asm_message_queue	111
4.58.2.6	barrier_queue	111
4.58.2.7	bluetooth	111
4.58.2.8	client_thread_arg	112
4.58.2.9	cmd_prompt	112
4.58.2.10	connection_queue	112
4.58.2.11	default_agentstatus	112
4.58.2.12	df	112
4.58.2.13	err	112
4.58.2.14	giant	112
4.58.2.15	giant_cond	112
4.58.2.16	giant_lock	113
4.58.2.17	hostname	113

4.58.2.18	initInterps	113
4.58.2.19	interp_options	113
4.58.2.20	interpreter_queue	113
4.58.2.21	listen_thread_arg	113
4.58.2.22	MC_signal	113
4.58.2.23	MC_signal_cond	113
4.58.2.24	MC_signal_lock	114
4.58.2.25	MC_steer_command	114
4.58.2.26	MC_steer_cond	114
4.58.2.27	MC_steer_lock	114
4.58.2.28	MC_sync_cond	114
4.58.2.29	MC_sync_lock	114
4.58.2.30	message_queue	114
4.58.2.31	port	114
4.58.2.32	private_key	115
4.58.2.33	quit	115
4.58.2.34	quit_cond	115
4.58.2.35	quit_lock	115
4.58.2.36	sockfd	115
4.58.2.37	stack_size	115
4.58.2.38	syncList	115
4.59	mc_rwlock_s Struct Reference	117
4.59.1	Detailed Description	117
4.59.2	Field Documentation	117
4.59.2.1	cond	117
4.59.2.2	lock	117
4.59.2.3	num_readers	117
4.59.2.4	write_flag	117
4.59.2.5	write_request	117
4.60	MCAgencyOptions_s Struct Reference	119
4.60.1	Detailed Description	119
4.60.2	Field Documentation	119
4.60.2.1	bluetooth	119
4.60.2.2	ch_options	119
4.60.2.3	default_agent_status	119
4.60.2.4	enable_security	119

4.60.2.5	initInterps	120
4.60.2.6	known_host_filename	120
4.60.2.7	modified	120
4.60.2.8	passphrase	120
4.60.2.9	priv_key_filename	120
4.60.2.10	stack_size	120
4.60.2.11	threads	120
4.61	md2_context Struct Reference	121
4.61.1	Detailed Description	121
4.61.2	Field Documentation	121
4.61.2.1	buffer	121
4.61.2.2	cksum	121
4.61.2.3	ipad	121
4.61.2.4	left	121
4.61.2.5	opad	121
4.61.2.6	state	122
4.62	md4_context Struct Reference	123
4.62.1	Detailed Description	123
4.62.2	Field Documentation	123
4.62.2.1	buffer	123
4.62.2.2	ipad	123
4.62.2.3	opad	123
4.62.2.4	state	123
4.62.2.5	total	123
4.63	md5_context Struct Reference	125
4.63.1	Detailed Description	125
4.63.2	Field Documentation	125
4.63.2.1	buffer	125
4.63.2.2	ipad	125
4.63.2.3	opad	125
4.63.2.4	state	125
4.63.2.5	total	126
4.64	message_s Struct Reference	127
4.64.1	Detailed Description	127
4.64.2	Field Documentation	127
4.64.2.1	addr	127

4.64.2.2	agent_xml_flag	127
4.64.2.3	connect_id	127
4.64.2.4	from_address	128
4.64.2.5	http_type	128
4.64.2.6	isHTTP	128
4.64.2.7	message_body	128
4.64.2.8	message_id	128
4.64.2.9	message_type	128
4.64.2.10	sending_agent_name	128
4.64.2.11	target	128
4.64.2.12	to_address	129
4.64.2.13	update_name	129
4.64.2.14	update_num	129
4.64.2.15	xml_payload	129
4.64.2.16	xml_root	129
4.65	message_send_arg_s Struct Reference	130
4.65.1	Detailed Description	130
4.65.2	Field Documentation	130
4.65.2.1	mc_platform	130
4.65.2.2	message	130
4.65.2.3	privatekey	130
4.66	mpi Struct Reference	131
4.66.1	Detailed Description	131
4.66.2	Field Documentation	131
4.66.2.1	n	131
4.66.2.2	p	131
4.66.2.3	s	131
4.67	mtp_http_content_s Struct Reference	133
4.67.1	Detailed Description	133
4.67.2	Field Documentation	133
4.67.2.1	content_type	133
4.67.2.2	data	133
4.68	mtp_http_s Struct Reference	134
4.68.1	Detailed Description	134
4.68.2	Field Documentation	134
4.68.2.1	accept_ranges	134

4.68.2.2	boundary	134
4.68.2.3	cache_control	134
4.68.2.4	connection	135
4.68.2.5	content	135
4.68.2.6	content_length	135
4.68.2.7	content_type	135
4.68.2.8	date	135
4.68.2.9	header_length	135
4.68.2.10	host	135
4.68.2.11	http_performative	135
4.68.2.12	http_status_code	136
4.68.2.13	http_version	136
4.68.2.14	message_parts	136
4.68.2.15	mime_version	136
4.68.2.16	response_code	136
4.68.2.17	response_string	136
4.68.2.18	return_code	136
4.68.2.19	server	136
4.68.2.20	target	136
4.68.2.21	user_agent	137
4.69	mxml_attr_s Struct Reference	138
4.69.1	Detailed Description	138
4.69.2	Field Documentation	138
4.69.2.1	name	138
4.69.2.2	value	138
4.70	mxml_custom_s Struct Reference	139
4.70.1	Detailed Description	139
4.70.2	Field Documentation	139
4.70.2.1	data	139
4.70.2.2	destroy	139
4.71	mxml_fdbuf_s Struct Reference	140
4.71.1	Detailed Description	140
4.71.2	Field Documentation	140
4.71.2.1	buffer	140
4.71.2.2	current	140
4.71.2.3	end	140

4.71.2.4	fd	140
4.72	mxml_index_s Struct Reference	141
4.72.1	Detailed Description	141
4.72.2	Field Documentation	141
4.72.2.1	alloc_nodes	141
4.72.2.2	attr	141
4.72.2.3	cur_node	141
4.72.2.4	nodes	141
4.72.2.5	num_nodes	141
4.73	mxml_node_s Struct Reference	143
4.73.1	Detailed Description	143
4.73.2	Field Documentation	143
4.73.2.1	child	143
4.73.2.2	last_child	143
4.73.2.3	next	143
4.73.2.4	parent	143
4.73.2.5	prev	144
4.73.2.6	type	144
4.73.2.7	value	144
4.74	mxml_text_s Struct Reference	145
4.74.1	Detailed Description	145
4.74.2	Field Documentation	145
4.74.2.1	string	145
4.74.2.2	whitespace	145
4.75	mxml_value_s Struct Reference	146
4.75.1	Detailed Description	146
4.75.2	Field Documentation	146
4.75.2.1	attrs	146
4.75.2.2	name	146
4.75.2.3	num_attrs	146
4.76	mxml_value_u Union Reference	147
4.76.1	Detailed Description	147
4.76.2	Field Documentation	147
4.76.2.1	custom	147
4.76.2.2	element	147
4.76.2.3	integer	147

4.76.2.4	opaque	147
4.76.2.5	real	147
4.76.2.6	text	148
4.77	options Struct Reference	149
4.77.1	Detailed Description	149
4.77.2	Field Documentation	149
4.77.2.1	buffer_size	149
4.77.2.2	command	149
4.77.2.3	conn_timeout	149
4.77.2.4	debug_level	149
4.77.2.5	force_cipher	150
4.77.2.6	iomode	150
4.77.2.7	max_bytes	150
4.77.2.8	max_connections	150
4.77.2.9	opmode	150
4.77.2.10	server_name	150
4.77.2.11	server_port	150
4.77.2.12	session_lifetime	150
4.77.2.13	session_reuse	150
4.78	rsa_context Struct Reference	152
4.78.1	Detailed Description	152
4.78.2	Field Documentation	152
4.78.2.1	D	152
4.78.2.2	DP	152
4.78.2.3	DQ	153
4.78.2.4	E	153
4.78.2.5	f_rng	153
4.78.2.6	hash_id	153
4.78.2.7	len	153
4.78.2.8	N	153
4.78.2.9	P	154
4.78.2.10	p_rng	154
4.78.2.11	padding	154
4.78.2.12	Q	154
4.78.2.13	QP	154
4.78.2.14	RN	154



4.78.2.15 RP	155
4.78.2.16 RQ	155
4.78.2.17 ver	155
4.79 sha1_context Struct Reference	156
4.79.1 Detailed Description	156
4.79.2 Field Documentation	156
4.79.2.1 buffer	156
4.79.2.2 ipad	156
4.79.2.3 opad	156
4.79.2.4 state	156
4.79.2.5 total	157
4.80 sha2_context Struct Reference	158
4.80.1 Detailed Description	158
4.80.2 Field Documentation	158
4.80.2.1 buffer	158
4.80.2.2 ipad	158
4.80.2.3 is224	158
4.80.2.4 opad	158
4.80.2.5 state	159
4.80.2.6 total	159
4.81 sha4_context Struct Reference	160
4.81.1 Detailed Description	160
4.81.2 Field Documentation	160
4.81.2.1 buffer	160
4.81.2.2 ipad	160
4.81.2.3 is384	160
4.81.2.4 opad	160
4.81.2.5 state	161
4.81.2.6 total	161
4.82 stationary_agent_info_s Struct Reference	162
4.82.1 Detailed Description	162
4.82.2 Field Documentation	162
4.82.2.1 agency	162
4.82.2.2 agent	162
4.82.2.3 args	162
4.82.2.4 attr	162

4.82.2.5	thread	162
4.83	syncList_s Struct Reference	163
4.83.1	Detailed Description	163
4.83.2	Field Documentation	163
4.83.2.1	giant_lock	163
4.83.2.2	list	163
4.83.2.3	lock	163
4.83.2.4	size	163
4.84	syncListNode_s Struct Reference	164
4.84.1	Detailed Description	164
4.84.2	Field Documentation	164
4.84.2.1	cond	164
4.84.2.2	id	164
4.84.2.3	lock	164
4.84.2.4	sem	164
4.84.2.5	signalled	164
<b>5</b>	<b>File Documentation</b>	<b>167</b>
5.1	/home/dko/Projects/mobilec/src/acc.c File Reference	167
5.1.1	Define Documentation	169
5.1.1.1	BACKLOG	169
5.1.1.2	BUFLen	169
5.1.1.3	CONN_THREADS	169
5.1.1.4	CONNECT_THREAD_EXIT	169
5.1.1.5	UDPPORT	169
5.1.2	Function Documentation	169
5.1.2.1	acc_connection_Thread	169
5.1.2.2	acc_Destroy	170
5.1.2.3	acc_Initialize	170
5.1.2.4	acc_MessageHandlerThread	170
5.1.2.5	acc_Start	170
5.1.2.6	acc_Thread	171
5.1.2.7	auth_conn_rece_key	171
5.1.2.8	listen_Thread	171
5.1.2.9	udplisten_Thread	171
5.2	/home/dko/Projects/mobilec/src/agent.c File Reference	172
5.2.1	Function Documentation	172

5.2.1.1	<a href="#">agent_AddPersistentVariable</a>	172
5.2.1.2	<a href="#">agent_ChScriptInitVar</a>	172
5.2.1.3	<a href="#">agent_Copy</a>	173
5.2.1.4	<a href="#">agent_Destroy</a>	173
5.2.1.5	<a href="#">agent_Initialize</a>	173
5.2.1.6	<a href="#">agent_New</a>	174
5.2.1.7	<a href="#">agent_NewBinary</a>	174
5.2.1.8	<a href="#">agent_RunChScript</a>	174
5.2.1.9	<a href="#">agent_RunChScriptThread</a>	174
5.3	<a href="#">/home/dko/Projects/mobilec/src/agent_datastate.c File Reference</a>	176
5.3.1	Function Documentation	176
5.3.1.1	<a href="#">agent_datastate_Copy</a>	176
5.3.1.2	<a href="#">agent_datastate_Destroy</a>	176
5.3.1.3	<a href="#">agent_datastate_New</a>	176
5.4	<a href="#">/home/dko/Projects/mobilec/src/agent_mailbox.c File Reference</a>	177
5.4.1	Function Documentation	177
5.4.1.1	<a href="#">agent_mailbox_Copy</a>	177
5.4.1.2	<a href="#">agent_mailbox_Destroy</a>	177
5.4.1.3	<a href="#">agent_mailbox_New</a>	177
5.4.1.4	<a href="#">agent_mailbox_Post</a>	177
5.4.1.5	<a href="#">agent_mailbox_Retrieve</a>	177
5.4.1.6	<a href="#">agent_mailbox_WaitRetrieve</a>	178
5.5	<a href="#">/home/dko/Projects/mobilec/src/agent_return_data.c File Reference</a>	179
5.5.1	Function Documentation	179
5.5.1.1	<a href="#">interpreter_variable_data_Copy</a>	179
5.5.1.2	<a href="#">interpreter_variable_data_Destroy</a>	179
5.5.1.3	<a href="#">interpreter_variable_data_Initialize</a>	179
5.5.1.4	<a href="#">interpreter_variable_data_InitializeFromAgent</a>	180
5.5.1.5	<a href="#">interpreter_variable_data_New</a>	180
5.6	<a href="#">/home/dko/Projects/mobilec/src/agent_task.c File Reference</a>	181
5.6.1	Function Documentation	181
5.6.1.1	<a href="#">agent_task_Copy</a>	181
5.6.1.2	<a href="#">agent_task_Destroy</a>	181
5.6.1.3	<a href="#">agent_task_New</a>	181
5.7	<a href="#">/home/dko/Projects/mobilec/src/ams.c File Reference</a>	182
5.7.1	Function Documentation	182

5.7.1.1	ams_Destroy	182
5.7.1.2	ams_Initialize	182
5.7.1.3	ams_ManageAgentList	182
5.7.1.4	ams_Print	183
5.7.1.5	ams_Start	183
5.7.1.6	ams_Thread	183
5.8	/home/dko/Projects/mobilec/src/barrier.c File Reference	184
5.8.1	Function Documentation	184
5.8.1.1	barrier_node_Destroy	184
5.8.1.2	barrier_node_Initialize	184
5.8.1.3	barrier_queue_Add	184
5.8.1.4	barrier_queue_Delete	185
5.8.1.5	barrier_queue_Destroy	185
5.8.1.6	barrier_queue_Get	185
5.8.1.7	barrier_queue_New	185
5.8.1.8	barrier_queue_Pop	185
5.9	/home/dko/Projects/mobilec/src/cmd_prompt.c File Reference	186
5.9.1	Function Documentation	186
5.9.1.1	cmd_prompt_Destroy	186
5.9.1.2	cmd_prompt_Initialize	186
5.9.1.3	cmd_prompt_Start	187
5.9.1.4	cmd_prompt_Thread	187
5.9.1.5	dealloc_command	187
5.9.1.6	exec_command	187
5.9.1.7	handler_COMPOSE_SEND	187
5.9.1.8	handler_FLUSH_AGENTS	187
5.9.1.9	handler_HELP	187
5.9.1.10	handler_PRINT_CONNECTLIST	188
5.9.1.11	handler_PRINTLIST_AGENTS	188
5.9.1.12	handler_PRINTLIST_MESSAGE	188
5.9.1.13	handler_QUIT	188
5.9.1.14	handler_SEND	188
5.9.1.15	process_command	188
5.9.1.16	split_string	188
5.10	/home/dko/Projects/mobilec/src/connection.c File Reference	189
5.10.1	Function Documentation	189

5.10.1.1	connection_Copy	189
5.10.1.2	connection_Destroy	189
5.10.1.3	connection_New	189
5.11	/home/dko/Projects/mobilec/src/data_structures.c File Reference	190
5.11.1	Function Documentation	190
5.11.1.1	agent_queue_Flush	190
5.11.1.2	AP_GENERIC_Copy	190
5.11.1.3	AP_GENERIC_Destroy	190
5.11.1.4	AP_QUEUE_SEARCH_TEMPLATE	191
5.11.1.5	AP_QUEUE_SEARCH_TEMPLATE	191
5.11.1.6	AP_QUEUE_STD_DEFN_TEMPLATE	191
5.11.1.7	AP_QUEUE_STD_DEFN_TEMPLATE	191
5.11.1.8	AP_QUEUE_STD_DEFN_TEMPLATE	191
5.12	/home/dko/Projects/mobilec/src/df.c File Reference	192
5.12.1	Define Documentation	192
5.12.1.1	REQUEST	192
5.12.2	Function Documentation	193
5.12.2.1	df_Add	193
5.12.2.2	df_AddRequest	193
5.12.2.3	df_Destroy	193
5.12.2.4	df_Initialize	193
5.12.2.5	df_node_Destroy	193
5.12.2.6	df_ProcessRequest	194
5.12.2.7	df_request_list_Destroy	194
5.12.2.8	df_request_list_New	194
5.12.2.9	df_request_list_node_Destroy	194
5.12.2.10	df_request_list_node_New	194
5.12.2.11	df_request_list_Pop	194
5.12.2.12	df_request_search_Destroy	194
5.12.2.13	df_request_search_New	195
5.12.2.14	df_SearchForService	195
5.12.2.15	df_Start	195
5.12.2.16	df_Thread	195
5.12.2.17	request_handler_DEREGISTER	195
5.12.2.18	request_handler_REGISTER	195
5.12.2.19	request_handler_SEARCH	195

5.12.2.20 request_handler_SUBSCRIBE . . . . .	196
5.13 /home/dko/Projects/mobilec/src/dynstring.c File Reference . . . . .	197
5.13.1 Function Documentation . . . . .	197
5.13.1.1 dynstring_Append . . . . .	197
5.13.1.2 dynstring_Destroy . . . . .	197
5.13.1.3 dynstring_New . . . . .	197
5.14 /home/dko/Projects/mobilec/src/fipa_acl.c File Reference . . . . .	198
5.14.1 Define Documentation . . . . .	199
5.14.1.1 FREEMEM . . . . .	199
5.14.2 Function Documentation . . . . .	200
5.14.2.1 fipa_acl_Compose . . . . .	200
5.14.2.2 fipa_acl_message_Copy . . . . .	200
5.14.2.3 fipa_acl_message_Destroy . . . . .	200
5.14.2.4 fipa_acl_message_New . . . . .	200
5.14.2.5 fipa_acl_Parse . . . . .	200
5.14.2.6 fipa_agent_identifer_Compose . . . . .	201
5.14.2.7 fipa_agent_identifer_Copy . . . . .	201
5.14.2.8 fipa_agent_identifer_Destroy . . . . .	201
5.14.2.9 fipa_agent_identifer_New . . . . .	201
5.14.2.10 fipa_agent_identifer_Parse . . . . .	201
5.14.2.11 fipa_agent_identifer_set_Compose . . . . .	202
5.14.2.12 fipa_agent_identifer_set_Copy . . . . .	202
5.14.2.13 fipa_agent_identifer_set_Destroy . . . . .	202
5.14.2.14 fipa_agent_identifer_set_New . . . . .	202
5.14.2.15 fipa_agent_identifer_set_Parse . . . . .	202
5.14.2.16 fipa_CheckNextToken . . . . .	202
5.14.2.17 fipa_DateTime_Compose . . . . .	203
5.14.2.18 fipa_DateTime_Copy . . . . .	203
5.14.2.19 fipa_DateTime_Destroy . . . . .	203
5.14.2.20 fipa_DateTime_New . . . . .	203
5.14.2.21 fipa_datetime_Parse . . . . .	203
5.14.2.22 fipa_expression_Compose . . . . .	203
5.14.2.23 fipa_expression_Copy . . . . .	204
5.14.2.24 fipa_expression_Destroy . . . . .	204
5.14.2.25 fipa_expression_New . . . . .	204
5.14.2.26 fipa_expression_Parse . . . . .	204

5.14.2.27	fipa_GetAtom	204
5.14.2.28	fipa_GetNextWord	204
5.14.2.29	fipa_GetWholeToken	205
5.14.2.30	fipa_message_parameter_Parse	205
5.14.2.31	fipa_message_string_Copy	205
5.14.2.32	fipa_message_string_Destroy	205
5.14.2.33	fipa_message_string_New	205
5.14.2.34	fipa_message_type_Parse	205
5.14.2.35	fipa_number_Compose	206
5.14.2.36	fipa_number_Copy	206
5.14.2.37	fipa_number_Destroy	206
5.14.2.38	fipa_number_New	206
5.14.2.39	fipa_performative_Compose	206
5.14.2.40	fipa_protocol_Compose	206
5.14.2.41	fipa_protocol_type_Parse	207
5.14.2.42	fipa_Reply	207
5.14.2.43	fipa_string_Compose	207
5.14.2.44	fipa_string_Copy	207
5.14.2.45	fipa_string_Destroy	207
5.14.2.46	fipa_string_New	208
5.14.2.47	fipa_string_Parse	208
5.14.2.48	fipa_url_Compose	208
5.14.2.49	fipa_url_Copy	208
5.14.2.50	fipa_url_Destroy	208
5.14.2.51	fipa_url_New	208
5.14.2.52	fipa_url_Parse	208
5.14.2.53	fipa_url_sequence_Compose	209
5.14.2.54	fipa_url_sequence_Copy	209
5.14.2.55	fipa_url_sequence_Destroy	209
5.14.2.56	fipa_url_sequence_New	209
5.14.2.57	fipa_url_sequence_Parse	209
5.14.2.58	fipa_word_Compose	209
5.14.2.59	fipa_word_Copy	209
5.14.2.60	fipa_word_Destroy	210
5.14.2.61	fipa_word_New	210
5.14.2.62	fipa_word_Parse	210

5.15	/home/dko/Projects/mobilec/src/fipa_envelope.c File Reference	211
5.15.1	Function Documentation	212
5.15.1.1	fipa_acl_envelope_Copy	212
5.15.1.2	fipa_acl_envelope_Destroy	212
5.15.1.3	fipa_acl_envelope_New	212
5.15.1.4	fipa_acl_envelope_Received_Copy	212
5.15.1.5	fipa_acl_envelope_Received_Destroy	212
5.15.1.6	fipa_acl_envelope_Received_New	213
5.15.1.7	fipa_acl_Param_Copy	213
5.15.1.8	fipa_acl_Param_Destroy	213
5.15.1.9	fipa_acl_Param_New	213
5.15.1.10	fipa_envelope_Compose	213
5.15.1.11	fipa_envelope_Compose__acl_representation	214
5.15.1.12	fipa_envelope_Compose__date	214
5.15.1.13	fipa_envelope_Compose__envelope	214
5.15.1.14	fipa_envelope_Compose__from	214
5.15.1.15	fipa_envelope_Compose__intended_receiver	214
5.15.1.16	fipa_envelope_Compose__params	214
5.15.1.17	fipa_envelope_Compose__payload_encoding	215
5.15.1.18	fipa_envelope_Compose__to	215
5.15.1.19	fipa_envelope_HandleAclRepresentation	215
5.15.1.20	fipa_envelope_HandleComments	215
5.15.1.21	fipa_envelope_HandleDate	215
5.15.1.22	fipa_envelope_HandleEnvelope	216
5.15.1.23	fipa_envelope_HandleFrom	216
5.15.1.24	fipa_envelope_HandleIntendedReceiver	216
5.15.1.25	fipa_envelope_HandleParams	216
5.15.1.26	fipa_envelope_HandlePayloadEncoding	216
5.15.1.27	fipa_envelope_HandlePayloadLength	217
5.15.1.28	fipa_envelope_HandleReceived	217
5.15.1.29	fipa_envelope_HandleTo	217
5.15.1.30	fipa_envelope_Parse	217
5.15.1.31	fipa_envelope_ParseAddresses	217
5.15.1.32	fipa_envelope_ParseAgentIdentifier	218
5.15.1.33	fipa_envelope_ParseResolvers	218
5.16	/home/dko/Projects/mobilec/src/include/acc.h File Reference	219



5.16.1	Function Documentation	219
5.16.1.1	acc_connection_Thread	219
5.16.1.2	acc_Destroy	220
5.16.1.3	acc_MessageHandlerThread	220
5.16.1.4	acc_Start	220
5.16.1.5	acc_Thread	220
5.16.1.6	listen_Thread	220
5.16.1.7	STRUCT	221
5.16.1.8	STRUCT	221
5.16.1.9	udplisten_Thread	221
5.16.2	Variable Documentation	221
5.16.2.1	addr	221
5.16.2.2	client_fd	221
5.16.2.3	port	221
5.16.2.4	server_fd	221
5.17	/home/dko/Projects/mobilec/src/include/agent.h File Reference	222
5.17.1	Define Documentation	222
5.17.1.1	AGENT_T	222
5.17.2	Typedef Documentation	223
5.17.2.1	agent_p	223
5.17.2.2	agent_t	223
5.17.2.3	MCAgent_t	223
5.17.3	Function Documentation	223
5.17.3.1	agent_AddPersistentVariable	223
5.17.3.2	agent_ChScriptInitVar	223
5.17.3.3	agent_Copy	224
5.17.3.4	agent_Destroy	224
5.17.3.5	agent_Initialize	224
5.17.3.6	agent_New	224
5.17.3.7	agent_NewBinary	225
5.17.3.8	agent_RunChScript	225
5.17.3.9	agent_RunChScriptThread	225
5.18	/home/dko/Projects/mobilec/src/include/agent_datastate.h File Reference	226
5.18.1	Typedef Documentation	226
5.18.1.1	agent_datastate_p	226
5.18.1.2	agent_datastate_t	226

5.18.2	Function Documentation . . . . .	226
5.18.2.1	agent_datastate_Copy . . . . .	226
5.18.2.2	agent_datastate_Destroy . . . . .	226
5.18.2.3	agent_datastate_New . . . . .	227
5.19	/home/dko/Projects/mobilec/src/include/agent_lib.h File Reference . . . . .	228
5.19.1	Function Documentation . . . . .	229
5.19.1.1	MC_AclAddReceiver_chdl . . . . .	229
5.19.1.2	MC_AclAddReplyTo_chdl . . . . .	229
5.19.1.3	MC_AclDestroy_chdl . . . . .	229
5.19.1.4	MC_AclGetContent_chdl . . . . .	230
5.19.1.5	MC_AclGetConversationID_chdl . . . . .	230
5.19.1.6	MC_AclGetPerformative_chdl . . . . .	230
5.19.1.7	MC_AclGetProtocol_chdl . . . . .	230
5.19.1.8	MC_AclGetSender_chdl . . . . .	230
5.19.1.9	MC_AclNew_chdl . . . . .	230
5.19.1.10	MC_AclPost_chdl . . . . .	230
5.19.1.11	MC_AclReply_chdl . . . . .	231
5.19.1.12	MC_AclRetrieve_chdl . . . . .	231
5.19.1.13	MC_AclSend_chdl . . . . .	231
5.19.1.14	MC_AclSetContent_chdl . . . . .	231
5.19.1.15	MC_AclSetConversationID_chdl . . . . .	231
5.19.1.16	MC_AclSetPerformative_chdl . . . . .	231
5.19.1.17	MC_AclSetProtocol_chdl . . . . .	231
5.19.1.18	MC_AclSetSender_chdl . . . . .	232
5.19.1.19	MC_AclWaitRetrieve_chdl . . . . .	232
5.19.1.20	MC_AddAgent_chdl . . . . .	232
5.19.1.21	MC_AgentAddTask_chdl . . . . .	232
5.19.1.22	MC_AgentVariableRetrieve_chdl . . . . .	232
5.19.1.23	MC_AgentVariableSave_chdl . . . . .	232
5.19.1.24	MC_Barrier_chdl . . . . .	232
5.19.1.25	MC_BarrierDelete_chdl . . . . .	233
5.19.1.26	MC_BarrierInit_chdl . . . . .	233
5.19.1.27	MC_CallAgentFunc_chdl . . . . .	233
5.19.1.28	MC_ComposeAgent_chdl . . . . .	233
5.19.1.29	MC_ComposeAgentFromFile_chdl . . . . .	233
5.19.1.30	MC_ComposeAgentFromFileWithWorkgroup_chdl . . . . .	233

5.19.1.31 MC_ComposeAgentWithWorkgroup_chdl . . . . .	233
5.19.1.32 MC_CondBroadcast_chdl . . . . .	234
5.19.1.33 MC_CondReset_chdl . . . . .	234
5.19.1.34 MC_CondSignal_chdl . . . . .	234
5.19.1.35 MC_CondWait_chdl . . . . .	234
5.19.1.36 MC_DeleteAgent_chdl . . . . .	234
5.19.1.37 MC_DeleteAgentWG_chdl . . . . .	234
5.19.1.38 MC_DeregisterService_chdl . . . . .	234
5.19.1.39 MC_DestroyServiceSearchResult_chdl . . . . .	235
5.19.1.40 MC_End_chdl . . . . .	235
5.19.1.41 MC_FindAgentByID_chdl . . . . .	235
5.19.1.42 MC_FindAgentByName_chdl . . . . .	235
5.19.1.43 MC_GetAgentID_chdl . . . . .	235
5.19.1.44 MC_GetAgentName_chdl . . . . .	235
5.19.1.45 MC_GetAgentNumTasks_chdl . . . . .	235
5.19.1.46 MC_GetAgentStatus_chdl . . . . .	236
5.19.1.47 MC_GetAgentXMLString_chdl . . . . .	236
5.19.1.48 MC_GetTimeOfDay_chdl . . . . .	236
5.19.1.49 MC_HaltAgency_chdl . . . . .	236
5.19.1.50 MC_MigrateAgent_chdl . . . . .	236
5.19.1.51 MC_MutexLock_chdl . . . . .	236
5.19.1.52 MC_MutexUnlock_chdl . . . . .	236
5.19.1.53 MC_PrintAgentCode_chdl . . . . .	237
5.19.1.54 MC_RegisterService_chdl . . . . .	237
5.19.1.55 MC_ResumeAgency_chdl . . . . .	237
5.19.1.56 MC_RetrieveAgent_chdl . . . . .	237
5.19.1.57 MC_RetrieveAgentCode_chdl . . . . .	237
5.19.1.58 MC_SaveData_chdl . . . . .	237
5.19.1.59 MC_SearchForService_chdl . . . . .	237
5.19.1.60 MC_SemaphorePost_chdl . . . . .	238
5.19.1.61 MC_SemaphoreWait_chdl . . . . .	238
5.19.1.62 MC_SendAgentMigrationMessage_chdl . . . . .	238
5.19.1.63 MC_SendAgentMigrationMessageFile_chdl . . . . .	238
5.19.1.64 MC_SendSteerCommand_chdl . . . . .	238
5.19.1.65 MC_SetAgentStatus_chdl . . . . .	238
5.19.1.66 MC_SetDefaultAgentStatus_chdl . . . . .	238

5.19.1.67	MC_SyncDelete_chdl . . . . .	239
5.19.1.68	MC_SyncInit_chdl . . . . .	239
5.19.1.69	MC_TerminateAgent_chdl . . . . .	239
5.19.1.70	MC_TerminateAgentWG_chdl . . . . .	239
5.20	/home/dko/Projects/mobilec/src/include/agent_mailbox.h File Reference . . . . .	240
5.20.1	Typedef Documentation . . . . .	240
5.20.1.1	agent_mailbox_p . . . . .	240
5.20.1.2	agent_mailbox_t . . . . .	240
5.20.2	Function Documentation . . . . .	240
5.20.2.1	agent_mailbox_Copy . . . . .	240
5.20.2.2	agent_mailbox_Destroy . . . . .	240
5.20.2.3	agent_mailbox_New . . . . .	240
5.20.2.4	agent_mailbox_Post . . . . .	241
5.20.2.5	agent_mailbox_Retrieve . . . . .	241
5.20.2.6	agent_mailbox_WaitRetrieve . . . . .	241
5.21	/home/dko/Projects/mobilec/src/include/agent_task.h File Reference . . . . .	242
5.21.1	Typedef Documentation . . . . .	242
5.21.1.1	agent_task_p . . . . .	242
5.21.1.2	agent_task_t . . . . .	242
5.21.2	Function Documentation . . . . .	242
5.21.2.1	agent_task_Copy . . . . .	242
5.21.2.2	agent_task_Destroy . . . . .	242
5.21.2.3	agent_task_New . . . . .	243
5.22	/home/dko/Projects/mobilec/src/include/ams.h File Reference . . . . .	244
5.22.1	Function Documentation . . . . .	244
5.22.1.1	ams_Destroy . . . . .	244
5.22.1.2	ams_ManageAgentList . . . . .	244
5.22.1.3	ams_Print . . . . .	244
5.22.1.4	ams_RemoveFinishedAgents . . . . .	244
5.22.1.5	ams_Start . . . . .	244
5.22.1.6	ams_Thread . . . . .	244
5.22.1.7	STRUCT . . . . .	245
5.23	/home/dko/Projects/mobilec/src/include/ap_queue_template.h File Reference . . . . .	246
5.23.1	Define Documentation . . . . .	246
5.23.1.1	AP_QUEUE_DECL_TEMPLATE . . . . .	246
5.23.1.2	AP_QUEUE_GENERIC_DECL_TEMPLATE . . . . .	247

5.23.1.3	AP_QUEUE_REMOVE_TEMPLATE . . . . .	247
5.23.1.4	AP_QUEUE_SEARCH_TEMPLATE . . . . .	247
5.23.1.5	AP_QUEUE_STD_DEFN_TEMPLATE . . . . .	248
5.23.2	Typedef Documentation . . . . .	248
5.23.2.1	AP_GENERIC_p . . . . .	248
5.23.2.2	AP_GENERIC_t . . . . .	248
5.23.3	Function Documentation . . . . .	248
5.23.3.1	interpreter_queue_CreateRetrieve . . . . .	248
5.24	/home/dko/Projects/mobilec/src/include/barrier.h File Reference . . . . .	249
5.24.1	Typedef Documentation . . . . .	249
5.24.1.1	barrier_node_p . . . . .	249
5.24.1.2	barrier_node_t . . . . .	249
5.24.1.3	barrier_queue_p . . . . .	249
5.24.1.4	barrier_queue_t . . . . .	250
5.24.2	Function Documentation . . . . .	250
5.24.2.1	barrier_node_Destroy . . . . .	250
5.24.2.2	barrier_node_Initialize . . . . .	250
5.24.2.3	barrier_queue_Add . . . . .	250
5.24.2.4	barrier_queue_Delete . . . . .	250
5.24.2.5	barrier_queue_Destroy . . . . .	250
5.24.2.6	barrier_queue_Get . . . . .	250
5.24.2.7	barrier_queue_New . . . . .	251
5.24.2.8	barrier_queue_Pop . . . . .	251
5.25	/home/dko/Projects/mobilec/src/include/cmd_prompt.h File Reference . . . . .	252
5.25.1	Define Documentation . . . . .	252
5.25.1.1	CMDLINE_SIZE . . . . .	252
5.25.2	Typedef Documentation . . . . .	252
5.25.2.1	cmd_prompt_p . . . . .	252
5.25.2.2	cmd_prompt_t . . . . .	253
5.25.2.3	command_p . . . . .	253
5.25.2.4	command_t . . . . .	253
5.25.3	Function Documentation . . . . .	253
5.25.3.1	cmd_prompt_Destroy . . . . .	253
5.25.3.2	cmd_prompt_Initialize . . . . .	253
5.25.3.3	cmd_prompt_Start . . . . .	253
5.25.3.4	cmd_prompt_Thread . . . . .	253

5.25.3.5	<a href="#">dealloc_command</a>	253
5.25.3.6	<a href="#">exec_command</a>	253
5.25.3.7	<a href="#">process_command</a>	253
5.25.3.8	<a href="#">split_string</a>	253
5.26	<a href="#">/home/dko/Projects/mobilec/src/include/commands.h File Reference</a>	255
5.26.1	<a href="#">Define Documentation</a>	255
5.26.1.1	<a href="#">COMMAND</a>	255
5.26.1.2	<a href="#">COMMAND</a>	256
5.26.1.3	<a href="#">COMMAND</a>	256
5.26.1.4	<a href="#">COMMAND</a>	256
5.26.1.5	<a href="#">COMMAND</a>	256
5.26.2	<a href="#">Typedef Documentation</a>	256
5.26.2.1	<a href="#">cmd_handler_t</a>	256
5.26.3	<a href="#">Enumeration Type Documentation</a>	256
5.26.3.1	<a href="#">command_indices_e</a>	256
5.26.4	<a href="#">Function Documentation</a>	256
5.26.4.1	<a href="#">COMMAND</a>	256
5.26.4.2	<a href="#">COMMAND</a>	256
5.26.4.3	<a href="#">COMMAND</a>	257
5.26.5	<a href="#">Variable Documentation</a>	257
5.26.5.1	<a href="#">cmd_handlers</a>	257
5.26.5.2	<a href="#">command_cmds</a>	257
5.26.5.3	<a href="#">command_descriptions</a>	257
5.27	<a href="#">/home/dko/Projects/mobilec/src/include/commands.x.h File Reference</a>	258
5.27.1	<a href="#">Function Documentation</a>	258
5.27.1.1	<a href="#">COMMAND</a>	258
5.27.1.2	<a href="#">COMMAND</a>	258
5.27.1.3	<a href="#">COMMAND</a>	259
5.27.2	<a href="#">Variable Documentation</a>	259
5.27.2.1	<a href="#">compose_send</a>	259
5.27.2.2	<a href="#">help</a>	259
5.27.2.3	<a href="#">is</a>	259
5.27.2.4	<a href="#">print_connectlist</a>	259
5.27.2.5	<a href="#">printlist_agents</a>	259
5.27.2.6	<a href="#">send</a>	259
5.28	<a href="#">/home/dko/Projects/mobilec/src/include/connection.h File Reference</a>	260

5.28.1	Typedef Documentation	260
5.28.1.1	connection_p	260
5.28.1.2	connection_t	260
5.28.2	Function Documentation	260
5.28.2.1	connection_Close	260
5.28.2.2	connection_Copy	260
5.28.2.3	connection_CreateMessage	261
5.28.2.4	connection_Destroy	261
5.28.2.5	connection_New	261
5.28.2.6	connection_RecvFromConnection	261
5.28.2.7	connection_SendtoConnection	261
5.28.2.8	connection_SetHosts	261
5.29	/home/dko/Projects/mobilec/src/include/data_structures.h File Reference	262
5.29.1	Function Documentation	263
5.29.1.1	agent_queue_Flush	263
5.29.1.2	AP_GENERIC_Destroy	263
5.29.1.3	AP_QUEUE_DECL_TEMPLATE	263
5.29.1.4	AP_QUEUE_DECL_TEMPLATE	263
5.29.1.5	AP_QUEUE_DECL_TEMPLATE	263
5.29.1.6	AP_QUEUE_DECL_TEMPLATE	263
5.29.1.7	AP_QUEUE_DECL_TEMPLATE	263
5.29.1.8	AP_QUEUE_DECL_TEMPLATE	263
5.29.1.9	AP_QUEUE_GENERIC_DECL_TEMPLATE	263
5.29.1.10	AP_QUEUE_GENERIC_DECL_TEMPLATE	264
5.29.1.11	AP_QUEUE_GENERIC_DECL_TEMPLATE	264
5.29.1.12	AP_QUEUE_GENERIC_DECL_TEMPLATE	264
5.29.1.13	AP_QUEUE_GENERIC_DECL_TEMPLATE	264
5.29.1.14	AP_QUEUE_GENERIC_DECL_TEMPLATE	264
5.29.2	Variable Documentation	264
5.29.2.1	agent_mailbox_p	264
5.29.2.2	int	264
5.29.2.3	Remove	264
5.29.2.4	RemoveID	265
5.29.2.5	RemoveName	265
5.29.2.6	Search	265
5.29.2.7	SearchName	265

5.30	/home/dko/Projects/mobilec/src/include/df.h File Reference . . . . .	266
5.30.1	Define Documentation . . . . .	267
5.30.1.1	REQUEST . . . . .	267
5.30.1.2	REQUEST . . . . .	267
5.30.2	Enumeration Type Documentation . . . . .	267
5.30.2.1	df_request_list_index_e . . . . .	267
5.30.2.2	service_types_e . . . . .	267
5.30.3	Function Documentation . . . . .	268
5.30.3.1	df_AddRequest . . . . .	268
5.30.3.2	df_Destroy . . . . .	268
5.30.3.3	df_Initialize . . . . .	268
5.30.3.4	df_node_Destroy . . . . .	268
5.30.3.5	df_ProcessRequest . . . . .	268
5.30.3.6	df_request_list_Destroy . . . . .	268
5.30.3.7	df_request_list_New . . . . .	268
5.30.3.8	df_request_list_node_Destroy . . . . .	269
5.30.3.9	df_request_list_node_New . . . . .	269
5.30.3.10	df_request_list_Pop . . . . .	269
5.30.3.11	df_request_search_Destroy . . . . .	269
5.30.3.12	df_request_search_New . . . . .	269
5.30.3.13	df_SearchForService . . . . .	269
5.30.3.14	df_Start . . . . .	269
5.30.3.15	df_Thread . . . . .	269
5.30.3.16	REQUEST . . . . .	270
5.30.3.17	REQUEST . . . . .	270
5.30.3.18	STRUCT . . . . .	270
5.30.3.19	STRUCT . . . . .	270
5.30.3.20	STRUCT . . . . .	270
5.30.3.21	STRUCT . . . . .	270
5.30.4	Variable Documentation . . . . .	270
5.30.4.1	cond . . . . .	270
5.30.4.2	lock . . . . .	270
5.30.4.3	mc_platform . . . . .	270
5.30.4.4	node . . . . .	270
5.30.4.5	num_entries . . . . .	271
5.30.4.6	request_list . . . . .	271



5.30.4.7	search_results	271
5.30.4.8	search_string	271
5.30.4.9	service_list	271
5.30.4.10	size	271
5.30.4.11	thread	271
5.30.4.12	waiting	272
5.30.4.13	waiting_cond	272
5.30.4.14	waiting_lock	272
5.31	/home/dko/Projects/mobilec/src/include/df_request.x.h File Reference	273
5.31.1	Function Documentation	273
5.31.1.1	REQUEST	273
5.31.1.2	REQUEST	273
5.31.2	Variable Documentation	273
5.31.2.1	register	273
5.31.2.2	search	273
5.32	/home/dko/Projects/mobilec/src/include/dynstring.h File Reference	274
5.32.1	Define Documentation	274
5.32.1.1	COMPOSE_BLOCKSIZE	274
5.32.2	Typedef Documentation	274
5.32.2.1	dynstring_p	274
5.32.2.2	dynstring_t	274
5.32.3	Function Documentation	274
5.32.3.1	dynstring_Append	274
5.32.3.2	dynstring_Destroy	275
5.32.3.3	dynstring_New	275
5.33	/home/dko/Projects/mobilec/src/include/fipa_acl.h File Reference	276
5.33.1	Typedef Documentation	279
5.33.1.1	fipa_acl_message_p	279
5.33.1.2	fipa_acl_message_t	279
5.33.1.3	fipa_agent_identifier_p	279
5.33.1.4	fipa_agent_identifier_set_p	279
5.33.1.5	fipa_agent_identifier_set_t	279
5.33.1.6	fipa_agent_identifier_t	279
5.33.1.7	fipa_DateTime_p	279
5.33.1.8	fipa_DateTime_t	279
5.33.1.9	fipa_expression_p	279

5.33.1.10	<a href="#">fipa_expression_t</a>	279
5.33.1.11	<a href="#">fipa_expression_type_t</a>	279
5.33.1.12	<a href="#">fipa_message_string_p</a>	279
5.33.1.13	<a href="#">fipa_message_string_t</a>	279
5.33.1.14	<a href="#">fipa_number_p</a>	279
5.33.1.15	<a href="#">fipa_number_t</a>	279
5.33.1.16	<a href="#">fipa_string_p</a>	279
5.33.1.17	<a href="#">fipa_string_t</a>	280
5.33.1.18	<a href="#">fipa_url_p</a>	280
5.33.1.19	<a href="#">fipa_url_sequence_p</a>	280
5.33.1.20	<a href="#">fipa_url_sequence_t</a>	280
5.33.1.21	<a href="#">fipa_url_t</a>	280
5.33.1.22	<a href="#">fipa_word_p</a>	280
5.33.1.23	<a href="#">fipa_word_t</a>	280
5.33.2	Enumeration Type Documentation	280
5.33.2.1	<a href="#">fipa_expression_type_e</a>	280
5.33.2.2	<a href="#">fipa_performative_e</a>	280
5.33.2.3	<a href="#">fipa_protocol_e</a>	282
5.33.3	Function Documentation	282
5.33.3.1	<a href="#">fipa_acl_Compose</a>	282
5.33.3.2	<a href="#">fipa_acl_message_Copy</a>	282
5.33.3.3	<a href="#">fipa_acl_message_Destroy</a>	283
5.33.3.4	<a href="#">fipa_acl_message_New</a>	283
5.33.3.5	<a href="#">fipa_acl_Parse</a>	283
5.33.3.6	<a href="#">fipa_agent_identifier_Compose</a>	283
5.33.3.7	<a href="#">fipa_agent_identifier_Copy</a>	283
5.33.3.8	<a href="#">fipa_agent_identifier_Destroy</a>	283
5.33.3.9	<a href="#">fipa_agent_identifier_New</a>	283
5.33.3.10	<a href="#">fipa_agent_identifier_Parse</a>	284
5.33.3.11	<a href="#">fipa_agent_identifier_set_Compose</a>	284
5.33.3.12	<a href="#">fipa_agent_identifier_set_Copy</a>	284
5.33.3.13	<a href="#">fipa_agent_identifier_set_Destroy</a>	284
5.33.3.14	<a href="#">fipa_agent_identifier_set_New</a>	284
5.33.3.15	<a href="#">fipa_agent_identifier_set_Parse</a>	285
5.33.3.16	<a href="#">fipa_CheckNextToken</a>	285
5.33.3.17	<a href="#">fipa_DateTime_Compose</a>	285

5.33.3.18	<a href="#">fipa_DateTime_Copy</a>	285
5.33.3.19	<a href="#">fipa_DateTime_Destroy</a>	285
5.33.3.20	<a href="#">fipa_DateTime_New</a>	285
5.33.3.21	<a href="#">fipa_datetime_Parse</a>	285
5.33.3.22	<a href="#">fipa_expression_Compose</a>	286
5.33.3.23	<a href="#">fipa_expression_Copy</a>	286
5.33.3.24	<a href="#">fipa_expression_Destroy</a>	286
5.33.3.25	<a href="#">fipa_expression_New</a>	286
5.33.3.26	<a href="#">fipa_expression_Parse</a>	287
5.33.3.27	<a href="#">fipa_GetAtom</a>	287
5.33.3.28	<a href="#">fipa_GetNextWord</a>	287
5.33.3.29	<a href="#">fipa_GetWholeToken</a>	287
5.33.3.30	<a href="#">fipa_message_parameter_Parse</a>	287
5.33.3.31	<a href="#">fipa_message_string_Copy</a>	287
5.33.3.32	<a href="#">fipa_message_string_Destroy</a>	287
5.33.3.33	<a href="#">fipa_message_string_New</a>	287
5.33.3.34	<a href="#">fipa_message_type_Parse</a>	288
5.33.3.35	<a href="#">fipa_number_Compose</a>	288
5.33.3.36	<a href="#">fipa_number_Copy</a>	288
5.33.3.37	<a href="#">fipa_number_Destroy</a>	288
5.33.3.38	<a href="#">fipa_number_New</a>	288
5.33.3.39	<a href="#">fipa_performative_Compose</a>	288
5.33.3.40	<a href="#">fipa_protocol_Compose</a>	289
5.33.3.41	<a href="#">fipa_protocol_type_Parse</a>	289
5.33.3.42	<a href="#">fipa_Reply</a>	289
5.33.3.43	<a href="#">fipa_string_Compose</a>	289
5.33.3.44	<a href="#">fipa_string_Copy</a>	290
5.33.3.45	<a href="#">fipa_string_Destroy</a>	290
5.33.3.46	<a href="#">fipa_string_New</a>	290
5.33.3.47	<a href="#">fipa_string_Parse</a>	290
5.33.3.48	<a href="#">fipa_url_Compose</a>	290
5.33.3.49	<a href="#">fipa_url_Copy</a>	290
5.33.3.50	<a href="#">fipa_url_Destroy</a>	290
5.33.3.51	<a href="#">fipa_url_New</a>	291
5.33.3.52	<a href="#">fipa_url_Parse</a>	291
5.33.3.53	<a href="#">fipa_url_sequence_Compose</a>	291

5.33.3.54	<a href="#">fipa_url_sequence_Copy</a>	291
5.33.3.55	<a href="#">fipa_url_sequence_Destroy</a>	291
5.33.3.56	<a href="#">fipa_url_sequence_New</a>	291
5.33.3.57	<a href="#">fipa_url_sequence_Parse</a>	291
5.33.3.58	<a href="#">fipa_word_Compose</a>	292
5.33.3.59	<a href="#">fipa_word_Copy</a>	292
5.33.3.60	<a href="#">fipa_word_Destroy</a>	292
5.33.3.61	<a href="#">fipa_word_New</a>	292
5.33.3.62	<a href="#">fipa_word_Parse</a>	292
5.34	<a href="#">/home/dko/Projects/mobilec/src/include/fipa_acl_envelope.h File Reference</a>	293
5.34.1	<a href="#">Typedef Documentation</a>	294
5.34.1.1	<a href="#">fipa_acl_envelope_p</a>	294
5.34.1.2	<a href="#">fipa_acl_envelope_Received_t</a>	294
5.34.1.3	<a href="#">fipa_acl_envelope_t</a>	294
5.34.1.4	<a href="#">fipa_acl_Param_t</a>	294
5.34.2	<a href="#">Function Documentation</a>	294
5.34.2.1	<a href="#">fipa_acl_envelope_Copy</a>	294
5.34.2.2	<a href="#">fipa_acl_envelope_Destroy</a>	294
5.34.2.3	<a href="#">fipa_acl_envelope_New</a>	295
5.34.2.4	<a href="#">fipa_acl_envelope_Received_Copy</a>	295
5.34.2.5	<a href="#">fipa_acl_envelope_Received_Destroy</a>	295
5.34.2.6	<a href="#">fipa_acl_envelope_Received_New</a>	295
5.34.2.7	<a href="#">fipa_acl_Param_Copy</a>	295
5.34.2.8	<a href="#">fipa_acl_Param_Destroy</a>	295
5.34.2.9	<a href="#">fipa_acl_Param_New</a>	296
5.34.2.10	<a href="#">fipa_envelope_Compose</a>	296
5.34.2.11	<a href="#">fipa_envelope_Compose__acl_representation</a>	296
5.34.2.12	<a href="#">fipa_envelope_Compose__date</a>	296
5.34.2.13	<a href="#">fipa_envelope_Compose__envelope</a>	296
5.34.2.14	<a href="#">fipa_envelope_Compose__from</a>	296
5.34.2.15	<a href="#">fipa_envelope_Compose__intended_receiver</a>	297
5.34.2.16	<a href="#">fipa_envelope_Compose__params</a>	297
5.34.2.17	<a href="#">fipa_envelope_Compose__payload_encoding</a>	297
5.34.2.18	<a href="#">fipa_envelope_Compose__to</a>	297
5.34.2.19	<a href="#">fipa_envelope_HandleAclRepresentation</a>	297
5.34.2.20	<a href="#">fipa_envelope_HandleComments</a>	298

5.34.2.21	<a href="#">fipa_envelope_HandleDate</a>	298
5.34.2.22	<a href="#">fipa_envelope_HandleEnvelope</a>	298
5.34.2.23	<a href="#">fipa_envelope_HandleFrom</a>	298
5.34.2.24	<a href="#">fipa_envelope_HandleIntendedReceiver</a>	298
5.34.2.25	<a href="#">fipa_envelope_HandleParams</a>	299
5.34.2.26	<a href="#">fipa_envelope_HandlePayloadEncoding</a>	299
5.34.2.27	<a href="#">fipa_envelope_HandlePayloadLength</a>	299
5.34.2.28	<a href="#">fipa_envelope_HandleReceived</a>	299
5.34.2.29	<a href="#">fipa_envelope_HandleTo</a>	299
5.34.2.30	<a href="#">fipa_envelope_Parse</a>	300
5.34.2.31	<a href="#">fipa_envelope_ParseAddresses</a>	300
5.34.2.32	<a href="#">fipa_envelope_ParseAgentIdentifier</a>	300
5.34.2.33	<a href="#">fipa_envelope_ParseResolvers</a>	300
5.35	<a href="#">/home/dko/Projects/mobilec/src/include/fipa_comm.h File Reference</a>	301
5.35.1	<a href="#">Define Documentation</a>	303
5.35.1.1	<a href="#">false</a>	303
5.35.1.2	<a href="#">MC_NUM_PERFORMATIVES</a>	303
5.35.2	<a href="#">Typedef Documentation</a>	303
5.35.2.1	<a href="#">fipa_comm_action_p</a>	303
5.35.2.2	<a href="#">fipa_comm_action_s</a>	303
5.35.2.3	<a href="#">fipa_comm_callback</a>	303
5.35.2.4	<a href="#">fipa_comm_message_check_p</a>	303
5.35.2.5	<a href="#">fipa_comm_message_check_s</a>	303
5.35.2.6	<a href="#">fipa_comm_p</a>	303
5.35.2.7	<a href="#">fipa_comm_performative_p</a>	303
5.35.2.8	<a href="#">fipa_comm_performative_s</a>	304
5.35.2.9	<a href="#">fipa_comm_protocol_cn_p</a>	304
5.35.2.10	<a href="#">fipa_comm_protocol_cn_s</a>	304
5.35.2.11	<a href="#">fipa_comm_protocol_p</a>	304
5.35.2.12	<a href="#">fipa_comm_protocol_s</a>	304
5.35.2.13	<a href="#">fipa_comm_reply_p</a>	304
5.35.2.14	<a href="#">fipa_comm_reply_s</a>	304
5.35.2.15	<a href="#">fipa_comm_s</a>	304
5.35.2.16	<a href="#">fipa_list_p</a>	304
5.35.2.17	<a href="#">fipa_list_s</a>	304
5.35.2.18	<a href="#">MCFIPAComm_p</a>	304

5.35.2.19	MCFIPACComm_s	305
5.35.2.20	protocol_contract_net_callback	305
5.35.3	Enumeration Type Documentation	305
5.35.3.1	fipa_comm_action_type_e	305
5.35.3.2	fipa_comm_check_type_e	305
5.35.3.3	fipa_comm_logic_type_e	305
5.35.3.4	fipa_comm_protocol_cn_state_e	306
5.35.3.5	fipa_comm_type_e	306
5.35.4	Function Documentation	308
5.35.4.1	fipa_comm_protocol_info_COPY	308
5.35.4.2	fipa_list_Add	308
5.35.4.3	mc_FIPACComm_AddNewProtocol	308
5.35.4.4	mc_FIPACComm_DeregisterPerformative	308
5.35.4.5	mc_FIPACComm_Destroy	308
5.35.4.6	mc_FIPACComm_FSM_CN	308
5.35.4.7	mc_FIPACComm_HandleMessage	308
5.35.4.8	mc_FIPACComm_HandleMessageOnce	308
5.35.4.9	mc_FIPACComm_HandleMessageWait	308
5.35.4.10	mc_FIPACComm_HandleProtocol	308
5.35.4.11	mc_FIPACComm_New	308
5.35.4.12	mc_FIPACComm_NewProtocol	308
5.35.4.13	mc_FIPACComm_RegisterAction	308
5.35.4.14	mc_FIPACComm_RegisterActionS	308
5.35.4.15	mc_FIPACComm_RegisterPerformative	308
5.35.4.16	mc_FIPACComm_RegisterProtocol	308
5.35.4.17	mc_FIPACComm_RegisterProtocolCallback	308
5.35.4.18	mc_FIPACComm_SetAgent	308
5.35.4.19	mc_FIPACComm_SetDefaultReply	308
5.35.4.20	mc_FIPACComm_SetPerformativeDefaultReply	308
5.35.4.21	mc_FIPACComm_SetProtocolState	308
5.35.4.22	mc_FIPACComm_StartProtocol	308
5.35.4.23	mc_FIPAProtocol_New	308
5.36	/home/dko/Projects/mobilec/src/include/host_id.h File Reference	309
5.36.1	Typedef Documentation	309
5.36.1.1	host_id_p	309
5.36.1.2	host_id_t	309

5.36.2	Function Documentation	309
5.36.2.1	host_id_Compare	309
5.36.2.2	host_id_Destroy	309
5.36.2.3	host_id_Initialize	309
5.37	/home/dko/Projects/mobilec/src/include/interpreter_variable_data.h File Reference	310
5.37.1	Typedef Documentation	310
5.37.1.1	interpreter_variable_data_p	310
5.37.1.2	interpreter_variable_data_t	310
5.37.2	Function Documentation	310
5.37.2.1	interpreter_variable_data_Copy	310
5.37.2.2	interpreter_variable_data_Destroy	311
5.37.2.3	interpreter_variable_data_Initialize	311
5.37.2.4	interpreter_variable_data_InitializeFromAgent	311
5.37.2.5	interpreter_variable_data_New	311
5.38	/home/dko/Projects/mobilec/src/include/libmc.h File Reference	312
5.38.1	Detailed Description	320
5.38.2	Define Documentation	320
5.38.2.1	_ERROR_CODE_E_	320
5.38.2.2	_FIPA_PERFORMATIVE_E_	320
5.38.2.3	AGENT_T	320
5.38.2.4	EXPORTMC	320
5.38.2.5	MC_Wait	320
5.38.3	Typedef Documentation	321
5.38.3.1	agency_p	321
5.38.3.2	agency_t	321
5.38.3.3	agent_p	321
5.38.3.4	agent_t	321
5.38.3.5	error_code_t	321
5.38.3.6	MC_AgentInitCallbackFunc_t	321
5.38.3.7	MC_SteerCommand_t	321
5.38.3.8	MCAgency_t	321
5.38.3.9	MCAgencyOptions_t	321
5.38.3.10	MCAgent_t	321
5.38.3.11	stationary_agent_info_t	322
5.38.4	Enumeration Type Documentation	322
5.38.4.1	error_code_e	322

5.38.4.2	fipa_performative_e . . . . .	322
5.38.4.3	MC_AgentStatus_e . . . . .	324
5.38.4.4	MC_AgentType_e . . . . .	324
5.38.4.5	MC_Signal_e . . . . .	324
5.38.4.6	MC_SteerCommand_e . . . . .	325
5.38.4.7	MC_ThreadIndex_e . . . . .	325
5.38.5	Function Documentation . . . . .	325
5.38.5.1	MC_AclAddReceiver . . . . .	325
5.38.5.2	MC_AclAddReplyTo . . . . .	325
5.38.5.3	MC_AclDestroy . . . . .	326
5.38.5.4	MC_AclGetContent . . . . .	326
5.38.5.5	MC_AclGetConversationID . . . . .	326
5.38.5.6	MC_AclGetPerformative . . . . .	326
5.38.5.7	MC_AclGetProtocol . . . . .	326
5.38.5.8	MC_AclGetSender . . . . .	326
5.38.5.9	MC_AclNew . . . . .	327
5.38.5.10	MC_AclPost . . . . .	327
5.38.5.11	MC_AclReply . . . . .	327
5.38.5.12	MC_AclRetrieve . . . . .	328
5.38.5.13	MC_AclSend . . . . .	328
5.38.5.14	MC_AclSetContent . . . . .	328
5.38.5.15	MC_AclSetConversationID . . . . .	329
5.38.5.16	MC_AclSetPerformative . . . . .	329
5.38.5.17	MC_AclSetProtocol . . . . .	329
5.38.5.18	MC_AclSetSender . . . . .	329
5.38.5.19	MC_AclWaitRetrieve . . . . .	329
5.38.5.20	MC_AddAgent . . . . .	330
5.38.5.21	MC_AddAgentInitCallback . . . . .	330
5.38.5.22	MC_AddStationaryAgent . . . . .	330
5.38.5.23	MC_AgentAddTask . . . . .	331
5.38.5.24	MC_AgentAddTaskFromFile . . . . .	331
5.38.5.25	MC_AgentInfo_GetAgency . . . . .	332
5.38.5.26	MC_AgentInfo_GetAgent . . . . .	332
5.38.6	Examples . . . . .	332
5.38.6.1	MC_AgentInfo_GetAgentArgs . . . . .	332
5.38.6.2	MC_AgentReturnArrayDim . . . . .	333



5.38.6.3	MC_AgentReturnArrayExtent	333
5.38.6.4	MC_AgentReturnArrayNum	333
5.38.6.5	MC_AgentReturnDataGetSymbolAddr	334
5.38.6.6	MC_AgentReturnDataSize	334
5.38.6.7	MC_AgentReturnDataType	335
5.38.6.8	MC_AgentReturnIsArray	335
5.38.6.9	MC_AgentVariableRetrieve	335
5.38.7	Examples	336
5.38.7.1	MC_AgentVariableRetrieveInfo	336
5.38.7.2	MC_AgentVariableSave	336
5.38.8	Examples	337
5.38.9	Examples	337
5.38.9.1	MC_Barrier	337
5.38.9.2	MC_BarrierDelete	337
5.38.9.3	MC_BarrierInit	337
5.38.10	Examples	338
5.38.10.1	MC_CallAgentFunc	338
5.38.11	Example	338
5.38.11.1	MC_CallAgentFuncArg	339
5.38.12	Example	339
5.38.12.1	MC_CallAgentFuncV	339
5.38.12.2	MC_CallAgentFuncVar	339
5.38.12.3	MC_ComposeAgent	340
5.38.12.4	MC_ComposeAgentFromFile	340
5.38.12.5	MC_ComposeAgentFromFileS	341
5.38.12.6	MC_ComposeAgentFromFileWithWorkgroup	341
5.38.12.7	MC_ComposeAgentS	342
5.38.12.8	MC_ComposeAgentWithWorkgroup	343
5.38.12.9	MC_CondBroadcast	343
5.38.12.10	MC_CondReset	344
5.38.12.11	MC_CondSignal	344
5.38.13	Example	345
5.38.13.1	MC_CondWait	345
5.38.14	Example	345
5.38.14.1	MC_CopyAgent	345
5.38.14.2	MC_DeleteAgent	346

5.38.14.3 MC_DeleteAgentWG . . . . .	346
5.38.14.4 MC_DestroyServiceSearchResult . . . . .	346
5.38.14.5 MC_End . . . . .	347
5.38.15 Example . . . . .	347
5.38.15.1 MC_FindAgentByID . . . . .	347
5.38.15.2 MC_FindAgentByName . . . . .	348
5.38.16 Example . . . . .	348
5.38.16.1 MC_GetAgentExecEngine . . . . .	348
5.38.16.2 MC_GetAgentID . . . . .	348
5.38.16.3 MC_GetAgentName . . . . .	349
5.38.16.4 MC_GetAgentNumTasks . . . . .	349
5.38.17 Example . . . . .	349
5.38.17.1 MC_GetAgentReturnData . . . . .	349
5.38.18 Example . . . . .	349
5.38.18.1 MC_GetAgentStatus . . . . .	350
5.38.18.2 MC_GetAgentType . . . . .	350
5.38.18.3 MC_GetAgentXMLString . . . . .	350
5.38.18.4 MC_HaltAgency . . . . .	350
5.38.18.5 MC_Initialize . . . . .	351
5.38.19 Example . . . . .	351
5.38.19.1 MC_InitializeAgencyOptions . . . . .	351
5.38.20 Example . . . . .	351
5.38.20.1 MC_LoadAgentFromFile . . . . .	352
5.38.20.2 MC_MainLoop . . . . .	352
5.38.20.3 MC_MigrateAgent . . . . .	352
5.38.20.4 MC_MutexLock . . . . .	353
5.38.21 Example . . . . .	353
5.38.21.1 MC_MutexUnlock . . . . .	353
5.38.21.2 MC_PrintAgentCode . . . . .	353
5.38.21.3 MC_RegisterService . . . . .	354
5.38.22 Example . . . . .	354
5.38.22.1 MC_ResetSignal . . . . .	354
5.38.22.2 MC_ResumeAgency . . . . .	355
5.38.22.3 MC_RetrieveAgent . . . . .	355
5.38.22.4 MC_RetrieveAgentCode . . . . .	355
5.38.22.5 MC_SearchForService . . . . .	355

5.38.23 Example	356
5.38.23.1 MC_SemaphorePost	356
5.38.24 Example	356
5.38.24.1 MC_SemaphoreWait	356
5.38.25 Example	357
5.38.25.1 MC_SendAgent	357
5.38.25.2 MC_SendAgentFile	357
5.38.25.3 MC_SendAgentMigrationMessage	358
5.38.25.4 MC_SendAgentMigrationMessageFile	358
5.38.25.5 MC_SetAgentStatus	358
5.38.25.6 MC_SetDefaultAgentStatus	359
5.38.25.7 MC_SetThreadOff	359
5.38.25.8 MC_SetThreadOn	360
5.38.25.9 MC_SetThreadsAllOff	360
5.38.25.10 MC_SetThreadsAllOn	360
5.38.25.11 MC_Steer	361
5.38.26 Example	361
5.38.26.1 MC_SteerControl	361
5.38.26.2 MC_SyncDelete	361
5.38.26.3 MC_SyncInit	362
5.38.26.4 MC_TerminateAgent	362
5.38.26.5 MC_TerminateAgentWG	363
5.38.26.6 MC_WaitAgent	363
5.38.26.7 MC_WaitRetrieveAgent	363
5.38.26.8 MC_WaitSignal	363
5.38.27 Example	364
5.39 /home/dko/Projects/mobilec/src/include/macros.h File Reference	365
5.39.1 Define Documentation	366
5.39.1.1 CH_DATATYPE_SIZE	366
5.39.1.2 CH_DATATYPE_STR_TO_VAL	366
5.39.1.3 CH_DATATYPE_STRING	367
5.39.1.4 CH_DATATYPE_VALUE_STRING	368
5.39.1.5 CH_STRING_DATATYPE	368
5.39.1.6 CHECK_NULL	369
5.39.1.7 COND_BROADCAST	369
5.39.1.8 COND_DESTROY	369

5.39.1.9 COND_INIT . . . . .	369
5.39.1.10 COND_RESET . . . . .	370
5.39.1.11 COND_SIGNAL . . . . .	370
5.39.1.12 COND_SLEEP . . . . .	370
5.39.1.13 COND_SLEEP_ACTION . . . . .	370
5.39.1.14 COND_T . . . . .	370
5.39.1.15 COND_WAIT . . . . .	370
5.39.1.16 GET_THREAD_MODE . . . . .	371
5.39.1.17 MUTEX_DESTROY . . . . .	371
5.39.1.18 MUTEX_INIT . . . . .	371
5.39.1.19 MUTEX_LOCK . . . . .	371
5.39.1.20 MUTEX_NEW . . . . .	371
5.39.1.21 MUTEX_T . . . . .	372
5.39.1.22 MUTEX_UNLOCK . . . . .	372
5.39.1.23 PTHREAD_STACK_SIZE . . . . .	372
5.39.1.24 RWLOCK_DESTROY . . . . .	372
5.39.1.25 RWLOCK_INIT . . . . .	372
5.39.1.26 RWLOCK_RDLOCK . . . . .	372
5.39.1.27 RWLOCK_RDUNLOCK . . . . .	373
5.39.1.28 RWLOCK_T . . . . .	373
5.39.1.29 RWLOCK_WRLOCK . . . . .	373
5.39.1.30 RWLOCK_WRUNLOCK . . . . .	373
5.39.1.31 SEMAPHORE_DESTROY . . . . .	373
5.39.1.32 SEMAPHORE_INIT . . . . .	373
5.39.1.33 SEMAPHORE_POST . . . . .	373
5.39.1.34 SEMAPHORE_T . . . . .	373
5.39.1.35 SEMAPHORE_WAIT . . . . .	374
5.39.1.36 SET_THREAD_OFF . . . . .	374
5.39.1.37 SET_THREAD_ON . . . . .	374
5.39.1.38 SIGNAL . . . . .	374
5.39.1.39 SLEEP_QUEUE . . . . .	374
5.39.1.40 SLEEP_RESET . . . . .	374
5.39.1.41 SOCKET_ERROR . . . . .	374
5.39.1.42 STRUCT . . . . .	375
5.39.1.43 THREAD_CANCEL . . . . .	375
5.39.1.44 THREAD_CREATE . . . . .	375

5.39.1.45	THREAD_DETACH	375
5.39.1.46	THREAD_EXIT	375
5.39.1.47	THREAD_JOIN	376
5.39.1.48	WAKE_QUEUE	376
5.39.1.49	WARN	376
5.40	/home/dko/Projects/mobilec/src/include/mc_error.h File Reference	377
5.40.1	Typedef Documentation	377
5.40.1.1	error_code_t	377
5.40.2	Enumeration Type Documentation	377
5.40.2.1	error_code_e	377
5.41	/home/dko/Projects/mobilec/src/include/mc_platform.h File Reference	379
5.41.1	Typedef Documentation	379
5.41.1.1	mc_platform_p	379
5.41.1.2	mc_platform_t	380
5.41.2	Function Documentation	380
5.41.2.1	mc_platform_Destroy	380
5.41.2.2	mc_platform_Initialize	380
5.42	/home/dko/Projects/mobilec/src/include/mc_rwlock.h File Reference	381
5.42.1	Typedef Documentation	381
5.42.1.1	mc_rwlock_p	381
5.42.1.2	mc_rwlock_t	381
5.42.2	Function Documentation	381
5.42.2.1	mc_rwlock_destroy	381
5.42.2.2	mc_rwlock_init	381
5.42.2.3	mc_rwlock_rdlock	382
5.42.2.4	mc_rwlock_rdunlock	382
5.42.2.5	mc_rwlock_wrlock	382
5.42.2.6	mc_rwlock_wrunlock	382
5.43	/home/dko/Projects/mobilec/src/include/message.h File Reference	383
5.43.1	Typedef Documentation	384
5.43.1.1	message_p	384
5.43.1.2	message_send_arg_t	384
5.43.1.3	message_t	384
5.43.1.4	message_type_t	384
5.43.2	Enumeration Type Documentation	384
5.43.2.1	message_type_e	384

5.43.3	Function Documentation . . . . .	384
5.43.3.1	auth_rece_send_msg . . . . .	384
5.43.3.2	http_to_hostport . . . . .	385
5.43.3.3	message_Copy . . . . .	385
5.43.3.4	message_Destroy . . . . .	385
5.43.3.5	message_InitializeFromAgent . . . . .	385
5.43.3.6	message_InitializeFromConnection . . . . .	385
5.43.3.7	message_InitializeFromString . . . . .	385
5.43.3.8	message_New . . . . .	385
5.43.3.9	message_Send . . . . .	386
5.43.3.10	message_send_Thread . . . . .	386
5.44	/home/dko/Projects/mobilec/src/include/mobilec.h File Reference . . . . .	387
5.45	/home/dko/Projects/mobilec/src/include/mtp_http.h File Reference . . . . .	388
5.45.1	Define Documentation . . . . .	389
5.45.1.1	SOCKET_INPUT_SIZE . . . . .	389
5.45.2	Typedef Documentation . . . . .	389
5.45.2.1	mtp_http_content_t . . . . .	389
5.45.2.2	mtp_http_p . . . . .	389
5.45.2.3	mtp_http_t . . . . .	389
5.45.3	Enumeration Type Documentation . . . . .	389
5.45.3.1	http_performative_e . . . . .	389
5.45.3.2	http_status_code_e . . . . .	390
5.45.4	Function Documentation . . . . .	391
5.45.4.1	http_GetExpression . . . . .	391
5.45.4.2	http_GetToken . . . . .	391
5.45.4.3	http_ParseExpression . . . . .	391
5.45.4.4	http_ParseRequest . . . . .	392
5.45.4.5	mtp_http_ComposeMessage . . . . .	392
5.45.4.6	mtp_http_CreateMessage . . . . .	392
5.45.4.7	mtp_http_Destroy . . . . .	392
5.45.4.8	mtp_http_InitializeFromConnection . . . . .	392
5.45.4.9	mtp_http_New . . . . .	393
5.45.4.10	mtp_http_Parse . . . . .	393
5.45.4.11	mtp_http_ParseHeader . . . . .	393
5.46	/home/dko/Projects/mobilec/src/include/xml_compose.h File Reference . . . . .	394
5.46.1	Function Documentation . . . . .	394

5.46.1.1	<a href="#">agent_xml_compose</a>	394
5.46.1.2	<a href="#">agent_xml_compose__agent_code</a>	394
5.46.1.3	<a href="#">agent_xml_compose__agent_data</a>	394
5.46.1.4	<a href="#">agent_xml_compose__create_row_nodes</a>	395
5.46.1.5	<a href="#">agent_xml_compose__data</a>	395
5.46.1.6	<a href="#">agent_xml_compose__gaf_message</a>	395
5.46.1.7	<a href="#">agent_xml_compose__home</a>	395
5.46.1.8	<a href="#">agent_xml_compose__message</a>	395
5.46.1.9	<a href="#">agent_xml_compose__mobile_agent</a>	395
5.46.1.10	<a href="#">agent_xml_compose__name</a>	396
5.46.1.11	<a href="#">agent_xml_compose__owner</a>	396
5.46.1.12	<a href="#">agent_xml_compose__row</a>	396
5.46.1.13	<a href="#">agent_xml_compose__task</a>	396
5.46.1.14	<a href="#">agent_xml_compose__tasks</a>	396
5.46.1.15	<a href="#">agent_xml_compose__wg_code</a>	397
5.47	<a href="#">/home/dko/Projects/mobilec/src/include/xml_helper.h File Reference</a>	398
5.47.1	<a href="#">Function Documentation</a>	398
5.47.1.1	<a href="#">whitespace_cb</a>	398
5.47.1.2	<a href="#">xml_find_sibling</a>	398
5.47.1.3	<a href="#">xml_get_cdata</a>	398
5.47.1.4	<a href="#">xml_get_child</a>	398
5.47.1.5	<a href="#">xml_get_deep_child</a>	399
5.47.1.6	<a href="#">xml_get_element_name</a>	399
5.47.1.7	<a href="#">xml_get_next_element</a>	399
5.47.1.8	<a href="#">xml_get_text</a>	399
5.47.1.9	<a href="#">xml_new_cdata</a>	399
5.48	<a href="#">/home/dko/Projects/mobilec/src/include/xml_parser.h File Reference</a>	400
5.48.1	<a href="#">Function Documentation</a>	400
5.48.1.1	<a href="#">agent_xml_parse</a>	400
5.48.1.2	<a href="#">agent_xml_parse__agent_code</a>	400
5.48.1.3	<a href="#">agent_xml_parse__agent_data</a>	401
5.48.1.4	<a href="#">agent_xml_parse__data</a>	401
5.48.1.5	<a href="#">agent_xml_parse__fill_row_data</a>	401
5.48.1.6	<a href="#">agent_xml_parse__home</a>	401
5.48.1.7	<a href="#">agent_xml_parse__mobile_agent</a>	401
5.48.1.8	<a href="#">agent_xml_parse__name</a>	401

5.48.1.9	agent_xml_parse__owner	402
5.48.1.10	agent_xml_parse__row	402
5.48.1.11	agent_xml_parse__sender	402
5.48.1.12	agent_xml_parse__task	402
5.48.1.13	agent_xml_parse__tasks	402
5.48.1.14	agent_xml_parse__wg_code	403
5.48.1.15	message_xml_parse	403
5.48.1.16	message_xml_parse__message	403
5.48.1.17	STRUCT	403
5.49	/home/dko/Projects/mobilec/src/libmc.c File Reference	404
5.49.1	Function Documentation	412
5.49.1.1	MC_AclAddReceiver	412
5.49.1.2	MC_AclAddReceiver_chdl	412
5.49.1.3	MC_AclAddReplyTo	413
5.49.1.4	MC_AclAddReplyTo_chdl	413
5.49.1.5	MC_AclDestroy	413
5.49.1.6	MC_AclDestroy_chdl	413
5.49.1.7	MC_AclGetContent	413
5.49.1.8	MC_AclGetContent_chdl	414
5.49.1.9	MC_AclGetConversationID	414
5.49.1.10	MC_AclGetConversationID_chdl	414
5.49.1.11	MC_AclGetPerformative	414
5.49.1.12	MC_AclGetPerformative_chdl	414
5.49.1.13	MC_AclGetProtocol	414
5.49.1.14	MC_AclGetProtocol_chdl	414
5.49.1.15	MC_AclGetSender	415
5.49.1.16	MC_AclGetSender_chdl	415
5.49.1.17	MC_AclNew	415
5.49.1.18	MC_AclNew_chdl	415
5.49.1.19	MC_AclPost	415
5.49.1.20	MC_AclPost_chdl	416
5.49.1.21	MC_AclReply	416
5.49.1.22	MC_AclReply_chdl	416
5.49.1.23	MC_AclRetrieve	416
5.49.1.24	MC_AclRetrieve_chdl	417
5.49.1.25	MC_AclSend	417



5.49.1.26	MC_AclSend_chdl	417
5.49.1.27	MC_AclSetContent	417
5.49.1.28	MC_AclSetContent_chdl	418
5.49.1.29	MC_AclSetConversationID	418
5.49.1.30	MC_AclSetConversationID_chdl	418
5.49.1.31	MC_AclSetPerformative	418
5.49.1.32	MC_AclSetPerformative_chdl	418
5.49.1.33	MC_AclSetProtocol	418
5.49.1.34	MC_AclSetProtocol_chdl	418
5.49.1.35	MC_AclSetSender	419
5.49.1.36	MC_AclSetSender_chdl	419
5.49.1.37	MC_AclWaitRetrieve	419
5.49.1.38	MC_AclWaitRetrieve_chdl	419
5.49.1.39	MC_AddAgent	419
5.49.1.40	MC_AddAgent_chdl	420
5.49.1.41	MC_AddAgentInitCallback	420
5.49.1.42	MC_AddStationaryAgent	420
5.49.1.43	MC_AgentAddTask	421
5.49.1.44	MC_AgentAddTask_chdl	421
5.49.1.45	MC_AgentAddTaskFromFile	421
5.49.1.46	MC_AgentInfo_GetAgency	422
5.49.1.47	MC_AgentInfo_GetAgent	422
5.49.2	Examples	422
5.49.2.1	MC_AgentInfo_GetAgentArgs	423
5.49.2.2	MC_AgentReturnArrayDim	423
5.49.2.3	MC_AgentReturnArrayExtent	423
5.49.2.4	MC_AgentReturnArrayNum	424
5.49.2.5	MC_AgentReturnDataGetSymbolAddr	424
5.49.2.6	MC_AgentReturnDataSize	424
5.49.2.7	MC_AgentReturnDataType	425
5.49.2.8	MC_AgentReturnIsArray	425
5.49.2.9	MC_AgentVariableRetrieve	425
5.49.3	Examples	426
5.49.3.1	MC_AgentVariableRetrieve_chdl	426
5.49.3.2	MC_AgentVariableRetrieveInfo	426
5.49.3.3	MC_AgentVariableSave	426

5.49.4	Examples	427
5.49.5	Examples	427
5.49.5.1	MC_AgentVariableSave_chdl	427
5.49.5.2	MC_Barrier	427
5.49.5.3	MC_Barrier_chdl	427
5.49.5.4	MC_BarrierDelete	427
5.49.5.5	MC_BarrierDelete_chdl	428
5.49.5.6	MC_BarrierInit	428
5.49.6	Examples	428
5.49.6.1	MC_BarrierInit_chdl	428
5.49.6.2	MC_CallAgentFunc	429
5.49.7	Example	429
5.49.7.1	MC_CallAgentFunc_chdl	429
5.49.7.2	MC_CallAgentFuncArg	429
5.49.8	Example	430
5.49.8.1	MC_CallAgentFuncV	430
5.49.8.2	MC_CallAgentFuncVar	430
5.49.8.3	MC_ComposeAgent	430
5.49.8.4	MC_ComposeAgent_chdl	431
5.49.8.5	MC_ComposeAgentFromFile	431
5.49.8.6	MC_ComposeAgentFromFile_chdl	432
5.49.8.7	MC_ComposeAgentFromFileS	432
5.49.8.8	MC_ComposeAgentFromFileWithWorkgroup	432
5.49.8.9	MC_ComposeAgentFromFileWithWorkgroup_chdl	433
5.49.8.10	MC_ComposeAgentS	433
5.49.8.11	MC_ComposeAgentWithWorkgroup	434
5.49.8.12	MC_ComposeAgentWithWorkgroup_chdl	435
5.49.8.13	MC_CondBroadcast	435
5.49.8.14	MC_CondBroadcast_chdl	435
5.49.8.15	MC_CondReset	435
5.49.8.16	MC_CondReset_chdl	436
5.49.8.17	MC_CondSignal	436
5.49.9	Example	436
5.49.9.1	MC_CondSignal_chdl	436
5.49.9.2	MC_CondWait	437
5.49.10	Example	437

5.49.10.1 MC_CondWait_chdl . . . . .	437
5.49.10.2 MC_CopyAgent . . . . .	437
5.49.10.3 MC_DeleteAgent . . . . .	438
5.49.10.4 MC_DeleteAgent_chdl . . . . .	438
5.49.10.5 MC_DeleteAgentWG . . . . .	438
5.49.10.6 MC_DeleteAgentWG_chdl . . . . .	438
5.49.10.7 MC_DeregisterService . . . . .	439
5.49.10.8 MC_DeregisterService_chdl . . . . .	439
5.49.10.9 MC_DestroyServiceSearchResult . . . . .	439
5.49.10.10 MC_DestroyServiceSearchResult_chdl . . . . .	439
5.49.10.11 MC_End . . . . .	439
5.49.11 Example . . . . .	440
5.49.11.1 MC_End_chdl . . . . .	440
5.49.11.2 MC_FindAgentByID . . . . .	440
5.49.11.3 MC_FindAgentByID_chdl . . . . .	440
5.49.11.4 MC_FindAgentByName . . . . .	440
5.49.12 Example . . . . .	441
5.49.12.1 MC_FindAgentByName_chdl . . . . .	441
5.49.12.2 MC_GetAgentArrivalTime . . . . .	441
5.49.12.3 MC_GetAgentArrivalTime_chdl . . . . .	441
5.49.12.4 MC_GetAgentExecEngine . . . . .	441
5.49.12.5 MC_GetAgentID . . . . .	442
5.49.12.6 MC_GetAgentID_chdl . . . . .	442
5.49.12.7 MC_GetAgentName . . . . .	442
5.49.12.8 MC_GetAgentName_chdl . . . . .	442
5.49.12.9 MC_GetAgentNumTasks . . . . .	442
5.49.13 Example . . . . .	442
5.49.13.1 MC_GetAgentNumTasks_chdl . . . . .	442
5.49.13.2 MC_GetAgentReturnData . . . . .	443
5.49.14 Example . . . . .	443
5.49.14.1 MC_GetAgentStatus . . . . .	443
5.49.14.2 MC_GetAgentStatus_chdl . . . . .	443
5.49.14.3 MC_GetAgentType . . . . .	444
5.49.14.4 MC_GetAgentXMLString . . . . .	444
5.49.14.5 MC_GetAgentXMLString_chdl . . . . .	444
5.49.14.6 MC_GetAllAgents . . . . .	444

5.49.14.7 MC_GetTimeOfDay_chdl . . . . .	444
5.49.14.8 MC_HaltAgency . . . . .	444
5.49.14.9 MC_HaltAgency_chdl . . . . .	445
5.49.14.10 MC_Initialize . . . . .	445
5.49.15 Example . . . . .	445
5.49.15.1 MC_InitializeAgencyOptions . . . . .	445
5.49.16 Example . . . . .	446
5.49.16.1 MC_LoadAgentFromFile . . . . .	446
5.49.16.2 MC_MainLoop . . . . .	446
5.49.16.3 MC_MigrateAgent . . . . .	447
5.49.16.4 MC_MigrateAgent_chdl . . . . .	447
5.49.16.5 MC_MutexLock . . . . .	447
5.49.17 Example . . . . .	447
5.49.17.1 MC_MutexLock_chdl . . . . .	448
5.49.17.2 MC_MutexUnlock . . . . .	448
5.49.17.3 MC_MutexUnlock_chdl . . . . .	448
5.49.17.4 MC_PrintAgentCode . . . . .	448
5.49.17.5 MC_PrintAgentCode_chdl . . . . .	448
5.49.17.6 MC_RegisterService . . . . .	448
5.49.18 Example . . . . .	449
5.49.18.1 MC_RegisterService_chdl . . . . .	449
5.49.18.2 MC_ResetSignal . . . . .	449
5.49.18.3 MC_ResumeAgency . . . . .	450
5.49.18.4 MC_ResumeAgency_chdl . . . . .	450
5.49.18.5 MC_RetrieveAgent . . . . .	450
5.49.18.6 MC_RetrieveAgent_chdl . . . . .	450
5.49.18.7 MC_RetrieveAgentCode . . . . .	450
5.49.18.8 MC_RetrieveAgentCode_chdl . . . . .	451
5.49.18.9 MC_SaveData_chdl . . . . .	451
5.49.18.10 MC_SearchForService . . . . .	451
5.49.19 Example . . . . .	451
5.49.19.1 MC_SearchForService_chdl . . . . .	452
5.49.19.2 MC_SemaphorePost . . . . .	452
5.49.20 Example . . . . .	452
5.49.20.1 MC_SemaphorePost_chdl . . . . .	452
5.49.20.2 MC_SemaphoreWait . . . . .	452

5.49.21 Example	453
5.49.21.1 MC_SemaphoreWait_chdl	453
5.49.21.2 MC_SendAgent	453
5.49.21.3 MC_SendAgentFile	453
5.49.21.4 MC_SendAgentMigrationMessage	453
5.49.21.5 MC_SendAgentMigrationMessage_chdl	454
5.49.21.6 MC_SendAgentMigrationMessageFile	454
5.49.21.7 MC_SendAgentMigrationMessageFile_chdl	454
5.49.21.8 MC_SendCh	454
5.49.21.9 MC_SendSteerCommand	455
5.49.21.10 MC_SendSteerCommand_chdl	455
5.49.21.11 MC_SetAgentStatus	455
5.49.21.12 MC_SetAgentStatus_chdl	455
5.49.21.13 MC_SetDefaultAgentStatus	455
5.49.21.14 MC_SetDefaultAgentStatus_chdl	456
5.49.21.15 MC_SetThreadOff	456
5.49.21.16 MC_SetThreadOn	456
5.49.21.17 MC_SetThreadsAllOff	457
5.49.21.18 MC_SetThreadsAllOn	457
5.49.21.19 MC_Steer	457
5.49.22 Example	458
5.49.22.1 MC_SteerControl	458
5.49.22.2 MC_SyncDelete	458
5.49.22.3 MC_SyncDelete_chdl	459
5.49.22.4 MC_SyncInit	459
5.49.22.5 MC_SyncInit_chdl	459
5.49.22.6 MC_TerminateAgent	459
5.49.22.7 MC_TerminateAgent_chdl	460
5.49.22.8 MC_TerminateAgentWG	460
5.49.22.9 MC_TerminateAgentWG_chdl	460
5.49.22.10 MC_WaitAgent	460
5.49.22.11 MC_WaitRetrieveAgent	460
5.49.22.12 MC_WaitSignal	461
5.49.23 Example	461
5.49.24 Variable Documentation	461
5.49.24.1 g_mc_platform	461

5.50	/home/dko/Projects/mobilec/src/mc_list/list.c File Reference	462
5.50.1	Function Documentation	462
5.50.1.1	list_pGetSize	462
5.50.1.2	ListAdd	462
5.50.1.3	ListDelete	462
5.50.1.4	ListGetHead	462
5.50.1.5	ListInitialize	463
5.50.1.6	ListInsert	463
5.50.1.7	ListPop	463
5.50.1.8	ListSearch	463
5.50.1.9	ListTerminate	463
5.51	/home/dko/Projects/mobilec/src/mc_list/list.h File Reference	464
5.51.1	Define Documentation	464
5.51.1.1	DATA	464
5.51.1.2	QUEUE_TEMPLATE	464
5.51.2	Typedef Documentation	465
5.51.2.1	list_p	465
5.51.2.2	list_t	465
5.51.2.3	listNode_p	465
5.51.2.4	listNode_t	465
5.51.3	Function Documentation	465
5.51.3.1	ListAdd	465
5.51.3.2	ListDelete	465
5.51.3.3	ListGetHead	465
5.51.3.4	ListGetSize	465
5.51.3.5	ListInitialize	465
5.51.3.6	ListInsert	465
5.51.3.7	ListPop	466
5.51.3.8	ListSearch	466
5.51.3.9	ListTerminate	466
5.52	/home/dko/Projects/mobilec/src/mc_platform.c File Reference	467
5.52.1	Define Documentation	467
5.52.1.1	DEFAULT_HOSTNAME_LENGTH	467
5.52.2	Function Documentation	467
5.52.2.1	mc_platform_Destroy	467
5.52.2.2	mc_platform_Initialize	468

5.53	/home/dko/Projects/mobilec/src/mc_rwlock.c File Reference	469
5.53.1	Function Documentation	469
5.53.1.1	mc_rwlock_destroy	469
5.53.1.2	mc_rwlock_init	469
5.53.1.3	mc_rwlock_rdlock	469
5.53.1.4	mc_rwlock_rdunlock	469
5.53.1.5	mc_rwlock_wrlock	470
5.53.1.6	mc_rwlock_wrunlock	470
5.54	/home/dko/Projects/mobilec/src/mc_sync/sync_list.c File Reference	471
5.54.1	Function Documentation	471
5.54.1.1	syncListAddNode	471
5.54.1.2	syncListDelete	471
5.54.1.3	syncListDestroy	471
5.54.1.4	syncListFind	472
5.54.1.5	syncListInit	472
5.54.1.6	syncListNew	472
5.54.1.7	syncListNodeDestroy	472
5.54.1.8	syncListNodeInit	472
5.54.1.9	syncListNodeNew	472
5.54.1.10	syncListRemove	473
5.55	/home/dko/Projects/mobilec/src/mc_sync/sync_list.h File Reference	474
5.55.1	Typedef Documentation	474
5.55.1.1	syncList_p	474
5.55.1.2	syncList_t	474
5.55.1.3	syncListNode_p	474
5.55.1.4	syncListNode_t	475
5.55.2	Function Documentation	475
5.55.2.1	syncListAddNode	475
5.55.2.2	syncListDelete	475
5.55.2.3	syncListDestroy	475
5.55.2.4	syncListFind	475
5.55.2.5	syncListGet	475
5.55.2.6	syncListInit	475
5.55.2.7	syncListNew	476
5.55.2.8	syncListNodeDestroy	476
5.55.2.9	syncListNodeInit	476

5.55.2.10 syncListNodeNew . . . . .	476
5.55.2.11 syncListRemove . . . . .	476
5.56 /home/dko/Projects/mobilec/src/message.c File Reference . . . . .	477
5.56.1 Define Documentation . . . . .	478
5.56.1.1 MSG_THREAD_EXIT . . . . .	478
5.56.1.2 MSG_THREADS . . . . .	478
5.56.1.3 SOCKET_INPUT_SIZE . . . . .	478
5.56.2 Function Documentation . . . . .	478
5.56.2.1 auth_rece_send_msg . . . . .	478
5.56.2.2 http_to_hostport . . . . .	478
5.56.2.3 message_Copy . . . . .	479
5.56.2.4 message_Destroy . . . . .	479
5.56.2.5 message_InitializeFromAgent . . . . .	479
5.56.2.6 message_InitializeFromConnection . . . . .	479
5.56.2.7 message_InitializeFromString . . . . .	479
5.56.2.8 message_New . . . . .	480
5.56.2.9 message_Send . . . . .	480
5.56.2.10 message_send_Thread . . . . .	480
5.57 /home/dko/Projects/mobilec/src/message_queue.c File Reference . . . . .	481
5.57.1 Function Documentation . . . . .	481
5.57.1.1 message_queue_SendOutgoing . . . . .	481
5.58 /home/dko/Projects/mobilec/src/mtp_http.c File Reference . . . . .	482
5.58.1 Define Documentation . . . . .	483
5.58.1.1 HTTP_PARSE_EXPR . . . . .	483
5.58.1.2 SAFE_FREE . . . . .	483
5.58.1.3 SAFE_FREE . . . . .	483
5.58.2 Function Documentation . . . . .	483
5.58.2.1 http_GetExpression . . . . .	483
5.58.2.2 http_GetToken . . . . .	484
5.58.2.3 http_ParseExpression . . . . .	484
5.58.2.4 http_ParseRequest . . . . .	484
5.58.2.5 mtp_http_ComposeMessage . . . . .	484
5.58.2.6 mtp_http_CreateMessage . . . . .	484
5.58.2.7 mtp_http_Destroy . . . . .	485
5.58.2.8 mtp_http_InitializeFromConnection . . . . .	485
5.58.2.9 mtp_http_New . . . . .	485



5.58.2.10	mtp_http_Parse	485
5.58.2.11	mtp_http_ParseHeader	485
5.58.2.12	mtp_http_ParseResponse	486
5.58.2.13	rece_de_msg	486
5.59	/home/dko/Projects/mobilec/src/mxml-2.2.2/config.h File Reference	487
5.59.1	Define Documentation	487
5.59.1.1	HAVE_STRDUP	487
5.59.1.2	HAVE_VSNPRINTF	487
5.59.1.3	MXML_VERSION	487
5.59.2	Function Documentation	487
5.59.2.1	mxml_strdupf	487
5.60	/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/config.h File Reference	488
5.60.1	Define Documentation	488
5.60.1.1	XYSSL_AES_C	488
5.60.1.2	XYSSL_ARC4_C	488
5.60.1.3	XYSSL_BASE64_C	488
5.60.1.4	XYSSL_BIGNUM_C	488
5.60.1.5	XYSSL_CERTS_C	489
5.60.1.6	XYSSL_DEBUG_C	489
5.60.1.7	XYSSL_DEBUG_MSG	489
5.60.1.8	XYSSL_DES_C	489
5.60.1.9	XYSSL_DHM_C	489
5.60.1.10	XYSSL_GENPRIME	489
5.60.1.11	XYSSL_HAVE_ASM	489
5.60.1.12	XYSSL_HAVEGE_C	489
5.60.1.13	XYSSL_MD5_C	489
5.60.1.14	XYSSL_NET_C	489
5.60.1.15	XYSSL_PADLOCK_C	489
5.60.1.16	XYSSL_RSA_C	490
5.60.1.17	XYSSL_SELF_TEST	490
5.60.1.18	XYSSL_SHA1_C	490
5.60.1.19	XYSSL_SHA2_C	490
5.60.1.20	XYSSL_SHA4_C	490
5.60.1.21	XYSSL_SSL_CLI_C	490
5.60.1.22	XYSSL_SSL_SRV_C	490
5.60.1.23	XYSSL_SSL_TLS_C	490

5.60.1.24	XYSSL_TIMING_C	490
5.60.1.25	XYSSL_X509_PARSE_C	490
5.60.1.26	XYSSL_X509_WRITE_C	490
5.61	/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-attr.c File Reference	491
5.61.1	Function Documentation	491
5.61.1.1	mxmlElementGetAttr	491
5.61.1.2	mxmlElementSetAttr	491
5.62	/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-entity.c File Reference	492
5.62.1	Function Documentation	492
5.62.1.1	default_callback	492
5.62.1.2	mxmlEntityAddCallback	492
5.62.1.3	mxmlEntityGetName	492
5.62.1.4	mxmlEntityGetValue	492
5.62.1.5	mxmlEntityRemoveCallback	493
5.62.2	Variable Documentation	493
5.62.2.1	callbacks	493
5.62.2.2	num_callbacks	493
5.63	/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-file.c File Reference	494
5.63.1	Define Documentation	495
5.63.1.1	ENCODE_UTF16BE	495
5.63.1.2	ENCODE_UTF16LE	495
5.63.1.3	ENCODE_UTF8	495
5.63.1.4	mxml_bad_char	495
5.63.2	Typedef Documentation	495
5.63.2.1	mxml_fdbuf_t	495
5.63.3	Function Documentation	495
5.63.3.1	mxml_add_char	495
5.63.3.2	mxml_fd_getc	496
5.63.3.3	mxml_fd_putc	496
5.63.3.4	mxml_fd_read	496
5.63.3.5	mxml_fd_write	496
5.63.3.6	mxml_file_getc	496
5.63.3.7	mxml_file_putc	496
5.63.3.8	mxml_get_entity	497
5.63.3.9	mxml_load_data	497
5.63.3.10	mxml_parse_element	497

5.63.3.11	<a href="#">mxml_string_getc</a>	497
5.63.3.12	<a href="#">mxml_string_putc</a>	497
5.63.3.13	<a href="#">mxml_write_name</a>	497
5.63.3.14	<a href="#">mxml_write_node</a>	498
5.63.3.15	<a href="#">mxml_write_string</a>	498
5.63.3.16	<a href="#">mxml_write_ws</a>	498
5.63.3.17	<a href="#">mxmlLoadFd</a>	498
5.63.3.18	<a href="#">mxmlLoadFile</a>	498
5.63.3.19	<a href="#">mxmlLoadString</a>	499
5.63.3.20	<a href="#">mxmlSaveAllocString</a>	499
5.63.3.21	<a href="#">mxmlSaveFd</a>	499
5.63.3.22	<a href="#">mxmlSaveFile</a>	499
5.63.3.23	<a href="#">mxmlSaveString</a>	499
5.63.3.24	<a href="#">mxmlSetCustomHandlers</a>	499
5.63.3.25	<a href="#">mxmlSetErrorCallback</a>	500
5.63.4	<a href="#">Variable Documentation</a>	500
5.63.4.1	<a href="#">mxml_custom_load_cb</a>	500
5.63.4.2	<a href="#">mxml_custom_save_cb</a>	500
5.63.4.3	<a href="#">mxml_error_cb</a>	500
5.64	<a href="#">/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-index.c File Reference</a>	501
5.64.1	<a href="#">Function Documentation</a>	501
5.64.1.1	<a href="#">index_compare</a>	501
5.64.1.2	<a href="#">index_find</a>	501
5.64.1.3	<a href="#">index_sort</a>	501
5.64.1.4	<a href="#">mxmlIndexDelete</a>	501
5.64.1.5	<a href="#">mxmlIndexEnum</a>	502
5.64.1.6	<a href="#">mxmlIndexFind</a>	502
5.64.1.7	<a href="#">mxmlIndexNew</a>	502
5.64.1.8	<a href="#">mxmlIndexReset</a>	502
5.65	<a href="#">/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-node.c File Reference</a>	503
5.65.1	<a href="#">Function Documentation</a>	503
5.65.1.1	<a href="#">mxml_new</a>	503
5.65.1.2	<a href="#">mxmlAdd</a>	503
5.65.1.3	<a href="#">mxmlDelete</a>	503
5.65.1.4	<a href="#">mxmlNewCustom</a>	504
5.65.1.5	<a href="#">mxmlNewElement</a>	504

5.65.1.6	<a href="#">mxmlNewInteger</a>	504
5.65.1.7	<a href="#">mxmlNewOpaque</a>	504
5.65.1.8	<a href="#">mxmlNewReal</a>	505
5.65.1.9	<a href="#">mxmlNewText</a>	505
5.65.1.10	<a href="#">mxmlNewTextf</a>	505
5.65.1.11	<a href="#">mxmlRemove</a>	505
5.66	<a href="#">/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-private.c File Reference</a>	506
5.66.1	Function Documentation	506
5.66.1.1	<a href="#">mxml_error</a>	506
5.66.1.2	<a href="#">mxml_integer_cb</a>	506
5.66.1.3	<a href="#">mxml_opaque_cb</a>	506
5.66.1.4	<a href="#">mxml_real_cb</a>	506
5.66.2	Variable Documentation	507
5.66.2.1	<a href="#">mxml_error_cb</a>	507
5.67	<a href="#">/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-search.c File Reference</a>	508
5.67.1	Function Documentation	508
5.67.1.1	<a href="#">mxmlFindElement</a>	508
5.67.1.2	<a href="#">mxmlWalkNext</a>	508
5.67.1.3	<a href="#">mxmlWalkPrev</a>	508
5.68	<a href="#">/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-set.c File Reference</a>	509
5.68.1	Function Documentation	509
5.68.1.1	<a href="#">mxmlSetCustom</a>	509
5.68.1.2	<a href="#">mxmlSetElement</a>	509
5.68.1.3	<a href="#">mxmlSetInteger</a>	509
5.68.1.4	<a href="#">mxmlSetOpaque</a>	509
5.68.1.5	<a href="#">mxmlSetReal</a>	509
5.68.1.6	<a href="#">mxmlSetText</a>	510
5.68.1.7	<a href="#">mxmlSetTextf</a>	510
5.69	<a href="#">/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-string.c File Reference</a>	511
5.69.1	Function Documentation	511
5.69.1.1	<a href="#">mxml_strdup</a>	511
5.69.1.2	<a href="#">mxml_strdupf</a>	511
5.69.1.3	<a href="#">mxml_vsnprintf</a>	511
5.70	<a href="#">/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml.h File Reference</a>	512
5.70.1	Define Documentation	514
5.70.1.1	<a href="#">MXML_ADD_AFTER</a>	514

5.70.1.2	<a href="#">MXML_ADD_BEFORE</a>	514
5.70.1.3	<a href="#">MXML_ADD_TO_PARENT</a>	514
5.70.1.4	<a href="#">MXML_DESCEND</a>	514
5.70.1.5	<a href="#">MXML_DESCEND_FIRST</a>	514
5.70.1.6	<a href="#">MXML_INTEGER_CALLBACK</a>	515
5.70.1.7	<a href="#">MXML_NO_CALLBACK</a>	515
5.70.1.8	<a href="#">MXML_NO_DESCEND</a>	515
5.70.1.9	<a href="#">MXML_NO_PARENT</a>	515
5.70.1.10	<a href="#">MXML_OPAQUE_CALLBACK</a>	515
5.70.1.11	<a href="#">MXML_REAL_CALLBACK</a>	515
5.70.1.12	<a href="#">MXML_TAB</a>	516
5.70.1.13	<a href="#">MXML_TEXT_CALLBACK</a>	516
5.70.1.14	<a href="#">MXML_WRAP</a>	516
5.70.1.15	<a href="#">MXML_WS_AFTER_CLOSE</a>	516
5.70.1.16	<a href="#">MXML_WS_AFTER_OPEN</a>	516
5.70.1.17	<a href="#">MXML_WS_BEFORE_CLOSE</a>	516
5.70.1.18	<a href="#">MXML_WS_BEFORE_OPEN</a>	516
5.70.2	<a href="#">Typedef Documentation</a>	516
5.70.2.1	<a href="#">mxml_attr_t</a>	516
5.70.2.2	<a href="#">mxml_custom_load_cb_t</a>	516
5.70.2.3	<a href="#">mxml_custom_save_cb_t</a>	516
5.70.2.4	<a href="#">mxml_custom_t</a>	517
5.70.2.5	<a href="#">mxml_element_t</a>	517
5.70.2.6	<a href="#">mxml_index_t</a>	517
5.70.2.7	<a href="#">mxml_node_t</a>	517
5.70.2.8	<a href="#">mxml_text_t</a>	517
5.70.2.9	<a href="#">mxml_type_t</a>	517
5.70.2.10	<a href="#">mxml_value_t</a>	517
5.70.3	<a href="#">Enumeration Type Documentation</a>	517
5.70.3.1	<a href="#">mxml_type_e</a>	517
5.70.4	<a href="#">Function Documentation</a>	517
5.70.4.1	<a href="#">mxml_error</a>	517
5.70.4.2	<a href="#">mxml_integer_cb</a>	517
5.70.4.3	<a href="#">mxml_opaque_cb</a>	517
5.70.4.4	<a href="#">mxml_real_cb</a>	518
5.70.4.5	<a href="#">mxmlAdd</a>	518

5.70.4.6	<code>mxmDelete</code>	518
5.70.4.7	<code>mxmElementGetAttr</code>	518
5.70.4.8	<code>mxmElementSetAttr</code>	518
5.70.4.9	<code>mxmEntityAddCallback</code>	519
5.70.4.10	<code>mxmEntityGetName</code>	519
5.70.4.11	<code>mxmEntityGetValue</code>	519
5.70.4.12	<code>mxmEntityRemoveCallback</code>	519
5.70.4.13	<code>mxmFindElement</code>	519
5.70.4.14	<code>mxmIndexDelete</code>	519
5.70.4.15	<code>mxmIndexEnum</code>	520
5.70.4.16	<code>mxmIndexFind</code>	520
5.70.4.17	<code>mxmIndexNew</code>	520
5.70.4.18	<code>mxmIndexReset</code>	520
5.70.4.19	<code>mxmLoadFd</code>	520
5.70.4.20	<code>mxmLoadFile</code>	520
5.70.4.21	<code>mxmLoadString</code>	520
5.70.4.22	<code>mxmNewCustom</code>	520
5.70.4.23	<code>mxmNewElement</code>	521
5.70.4.24	<code>mxmNewInteger</code>	521
5.70.4.25	<code>mxmNewOpaque</code>	521
5.70.4.26	<code>mxmNewReal</code>	521
5.70.4.27	<code>mxmNewText</code>	521
5.70.4.28	<code>mxmNewTextf</code>	522
5.70.4.29	<code>mxmRemove</code>	522
5.70.4.30	<code>mxmSaveAllocString</code>	522
5.70.4.31	<code>mxmSaveFd</code>	522
5.70.4.32	<code>mxmSaveFile</code>	522
5.70.4.33	<code>mxmSaveString</code>	522
5.70.4.34	<code>mxmSetCustom</code>	522
5.70.4.35	<code>mxmSetCustomHandlers</code>	522
5.70.4.36	<code>mxmSetElement</code>	522
5.70.4.37	<code>mxmSetErrorCallback</code>	522
5.70.4.38	<code>mxmSetInteger</code>	523
5.70.4.39	<code>mxmSetOpaque</code>	523
5.70.4.40	<code>mxmSetReal</code>	523
5.70.4.41	<code>mxmSetText</code>	523

5.70.4.42	mxmSetTextf	523
5.70.4.43	mxmWalkNext	523
5.70.4.44	mxmWalkPrev	523
5.71	/home/dko/Projects/mobilec/src/mxml-2.2.2/mxmldoc.c File Reference	524
5.71.1	Define Documentation	524
5.71.1.1	STATE_C_COMMENT	524
5.71.1.2	STATE_CHARACTER	524
5.71.1.3	STATE_CXX_COMMENT	524
5.71.1.4	STATE_IDENTIFIER	525
5.71.1.5	STATE_NONE	525
5.71.1.6	STATE_PREPROCESSOR	525
5.71.1.7	STATE_STRING	525
5.71.2	Function Documentation	525
5.71.2.1	add_variable	525
5.71.2.2	main	525
5.71.2.3	safe_strcpy	525
5.71.2.4	scan_file	526
5.71.2.5	sort_node	526
5.71.2.6	update_comment	526
5.71.2.7	write_documentation	526
5.71.2.8	write_element	526
5.71.2.9	write_string	527
5.71.2.10	ws_cb	527
5.72	/home/dko/Projects/mobilec/src/mxml-2.2.2/test/class.cxx File Reference	528
5.72.1	Variable Documentation	528
5.72.1.1	bar	528
5.72.1.2	f	528
5.72.1.3	foo	528
5.73	/home/dko/Projects/mobilec/src/mxml-2.2.2/test/enum.cxx File Reference	529
5.73.1	Typedef Documentation	529
5.73.1.1	foo_enum_t	529
5.73.2	Enumeration Type Documentation	529
5.73.2.1	foo_enum_e	529
5.74	/home/dko/Projects/mobilec/src/mxml-2.2.2/test/function.cxx File Reference	530
5.74.1	Function Documentation	530
5.74.1.1	foo_default_int	530

5.74.1.2	<a href="#">foo_default_string</a>	530
5.74.1.3	<a href="#">foo_float_function</a>	530
5.74.1.4	<a href="#">foo_void_function</a>	530
5.75	<a href="#">/home/dko/Projects/mobilec/src/mxml-2.2.2/test/struct.cxx File Reference</a>	531
5.75.1	<a href="#">Typedef Documentation</a>	531
5.75.1.1	<a href="#">foo_t</a>	531
5.76	<a href="#">/home/dko/Projects/mobilec/src/mxml-2.2.2/testmxml.c File Reference</a>	532
5.76.1	<a href="#">Function Documentation</a>	532
5.76.1.1	<a href="#">main</a>	532
5.76.1.2	<a href="#">type_cb</a>	532
5.76.1.3	<a href="#">whitespace_cb</a>	532
5.77	<a href="#">/home/dko/Projects/mobilec/src/mxml-2.2.2/winconfig.h File Reference</a>	533
5.77.1	<a href="#">Define Documentation</a>	533
5.77.1.1	<a href="#">HAVE_STRDUP</a>	533
5.77.1.2	<a href="#">HAVE_VSNPRINTF</a>	533
5.77.1.3	<a href="#">MXML_VERSION</a>	533
5.77.1.4	<a href="#">strdup</a>	533
5.77.1.5	<a href="#">vsnprintf</a>	534
5.77.2	<a href="#">Function Documentation</a>	534
5.77.2.1	<a href="#">mxml_strdup</a>	534
5.77.2.2	<a href="#">mxml_strdupf</a>	534
5.77.2.3	<a href="#">mxml_vsnprintf</a>	534
5.78	<a href="#">/home/dko/Projects/mobilec/src/winconfig.h File Reference</a>	535
5.78.1	<a href="#">Define Documentation</a>	535
5.78.1.1	<a href="#">PACKAGE_STRING</a>	535
5.78.1.2	<a href="#">PACKAGE_VERSION</a>	535
5.78.1.3	<a href="#">strdup</a>	535
5.78.1.4	<a href="#">strtok_r</a>	535
5.79	<a href="#">/home/dko/Projects/mobilec/src/security/asm.c File Reference</a>	536
5.80	<a href="#">/home/dko/Projects/mobilec/src/security/asm.h File Reference</a>	537
5.81	<a href="#">/home/dko/Projects/mobilec/src/security/asm_message_composer.c File Reference</a>	538
5.82	<a href="#">/home/dko/Projects/mobilec/src/security/asm_message_composer.h File Reference</a>	539
5.83	<a href="#">/home/dko/Projects/mobilec/src/security/asm_message_parser.c File Reference</a>	540
5.84	<a href="#">/home/dko/Projects/mobilec/src/security/asm_message_parser.h File Reference</a>	541
5.85	<a href="#">/home/dko/Projects/mobilec/src/security/asm_node.c File Reference</a>	542
5.86	<a href="#">/home/dko/Projects/mobilec/src/security/asm_node.h File Reference</a>	543



5.87	/home/dko/Projects/mobilec/src/security/interface.c File Reference	544
5.87.1	Function Documentation	544
5.87.1.1	aes_en_de	544
5.87.1.2	append_nonce_to_MA	544
5.87.1.3	extract_nonce_from_MA	545
5.87.1.4	generate_AES_key	545
5.87.1.5	generate_RSA_keys_ciphertext	545
5.87.1.6	generate_RSA_keys_plaintext	545
5.87.1.7	initiate_migration_process	545
5.87.1.8	read_encrypted_file	545
5.87.1.9	read_known_host_file	546
5.87.1.10	receive_AES_en_MA	546
5.87.1.11	remove_nonce_from_MA	546
5.87.1.12	reply_migration_process	546
5.87.1.13	rsa_decryption	546
5.87.1.14	rsa_encryption	546
5.87.1.15	send_AES_en_MA	546
5.87.1.16	separate_key_parts	547
5.88	/home/dko/Projects/mobilec/src/security/interface.h File Reference	548
5.88.1	Define Documentation	549
5.88.1.1	EXPONENT	549
5.88.1.2	KEY_SIZE	549
5.88.1.3	MAXDATASIZE	549
5.88.1.4	MODE_DECRYPT	549
5.88.1.5	MODE_ENCRYPT	549
5.88.1.6	PATH	549
5.88.1.7	RSA_DE	549
5.88.1.8	RSA_EN	549
5.88.2	Function Documentation	550
5.88.2.1	aes_en_de	550
5.88.2.2	generate_AES_key	550
5.88.2.3	generate_encrypt_send_AES_key	550
5.88.2.4	generate_RSA_keys_ciphertext	550
5.88.2.5	generate_RSA_keys_plaintext	550
5.88.2.6	initiate_migration_process	550
5.88.2.7	read_encrypted_file	551

5.88.2.8	<a href="#">read_known_host_file</a>	551
5.88.2.9	<a href="#">receive_AES_en_MA</a>	551
5.88.2.10	<a href="#">receive_decrypt_AES_key</a>	551
5.88.2.11	<a href="#">receiving_decrypting_MA_RSA</a>	551
5.88.2.12	<a href="#">receiving_verifying_MA_RSA</a>	551
5.88.2.13	<a href="#">reply_migration_process</a>	551
5.88.2.14	<a href="#">rsa_decryption</a>	551
5.88.2.15	<a href="#">rsa_encryption</a>	551
5.88.2.16	<a href="#">send_AES_en_MA</a>	552
5.89	<a href="#">/home/dko/Projects/mobilec/src/security/mc_dh.c File Reference</a>	553
5.90	<a href="#">/home/dko/Projects/mobilec/src/security/mc_dh.h File Reference</a>	554
5.91	<a href="#">/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/aes.h File Reference</a>	555
5.91.1	Detailed Description	555
5.91.2	Define Documentation	555
5.91.2.1	<a href="#">AES_DECRYPT</a>	555
5.91.2.2	<a href="#">AES_ENCRYPT</a>	556
5.91.3	Function Documentation	556
5.91.3.1	<a href="#">aes_crypt_cbc</a>	556
5.91.3.2	<a href="#">aes_crypt_cfb</a>	556
5.91.3.3	<a href="#">aes_crypt_ecb</a>	557
5.91.3.4	<a href="#">aes_self_test</a>	557
5.91.3.5	<a href="#">aes_setkey_dec</a>	557
5.91.3.6	<a href="#">aes_setkey_enc</a>	558
5.92	<a href="#">/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/arc4.h File Reference</a>	559
5.92.1	Detailed Description	559
5.92.2	Function Documentation	559
5.92.2.1	<a href="#">arc4_crypt</a>	559
5.92.2.2	<a href="#">arc4_self_test</a>	559
5.92.2.3	<a href="#">arc4_setup</a>	560
5.93	<a href="#">/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/base64.h File Reference</a>	561
5.93.1	Detailed Description	561
5.93.2	Define Documentation	561
5.93.2.1	<a href="#">XYSSL_ERR_BASE64_BUFFER_TOO_SMALL</a>	561
5.93.2.2	<a href="#">XYSSL_ERR_BASE64_INVALID_CHARACTER</a>	561
5.93.3	Function Documentation	561
5.93.3.1	<a href="#">base64_decode</a>	561

5.93.3.2	base64_encode . . . . .	562
5.93.3.3	base64_self_test . . . . .	562
5.94	/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/bignum.h File Reference . . . . .	563
5.94.1	Detailed Description . . . . .	565
5.94.2	Define Documentation . . . . .	566
5.94.2.1	MPI_CHK . . . . .	566
5.94.2.2	XYSSL_ERR_MPI_BAD_INPUT_DATA . . . . .	566
5.94.2.3	XYSSL_ERR_MPI_BUFFER_TOO_SMALL . . . . .	566
5.94.2.4	XYSSL_ERR_MPI_DIVISION_BY_ZERO . . . . .	566
5.94.2.5	XYSSL_ERR_MPI_FILE_IO_ERROR . . . . .	566
5.94.2.6	XYSSL_ERR_MPI_INVALID_CHARACTER . . . . .	566
5.94.2.7	XYSSL_ERR_MPI_NEGATIVE_VALUE . . . . .	566
5.94.2.8	XYSSL_ERR_MPI_NOT_ACCEPTABLE . . . . .	566
5.94.3	Typedef Documentation . . . . .	567
5.94.3.1	t_dbl . . . . .	567
5.94.3.2	t_int . . . . .	567
5.94.4	Function Documentation . . . . .	567
5.94.4.1	mpi_add_abs . . . . .	567
5.94.4.2	mpi_add_int . . . . .	567
5.94.4.3	mpi_add_mpi . . . . .	567
5.94.4.4	mpi_cmp_abs . . . . .	568
5.94.4.5	mpi_cmp_int . . . . .	568
5.94.4.6	mpi_cmp_mpi . . . . .	568
5.94.4.7	mpi_copy . . . . .	568
5.94.4.8	mpi_div_int . . . . .	569
5.94.4.9	mpi_div_mpi . . . . .	569
5.94.4.10	mpi_exp_mod . . . . .	569
5.94.4.11	mpi_free . . . . .	570
5.94.4.12	mpi_gcd . . . . .	570
5.94.4.13	mpi_gen_prime . . . . .	570
5.94.4.14	mpi_grow . . . . .	570
5.94.4.15	mpi_init . . . . .	571
5.94.4.16	mpi_inv_mod . . . . .	571
5.94.4.17	mpi_is_prime . . . . .	571
5.94.4.18	mpi_lsb . . . . .	572
5.94.4.19	mpi_lset . . . . .	572

5.94.4.20	<a href="#">mpi_mod_int</a>	572
5.94.4.21	<a href="#">mpi_mod_mpi</a>	572
5.94.4.22	<a href="#">mpi_msb</a>	573
5.94.4.23	<a href="#">mpi_mul_int</a>	573
5.94.4.24	<a href="#">mpi_mul_mpi</a>	573
5.94.4.25	<a href="#">mpi_read_binary</a>	573
5.94.4.26	<a href="#">mpi_read_file</a>	574
5.94.4.27	<a href="#">mpi_read_mystring</a>	574
5.94.4.28	<a href="#">mpi_read_string</a>	574
5.94.4.29	<a href="#">mpi_self_test</a>	575
5.94.4.30	<a href="#">mpi_shift_l</a>	575
5.94.4.31	<a href="#">mpi_shift_r</a>	575
5.94.4.32	<a href="#">mpi_size</a>	575
5.94.4.33	<a href="#">mpi_sub_abs</a>	576
5.94.4.34	<a href="#">mpi_sub_int</a>	576
5.94.4.35	<a href="#">mpi_sub_mpi</a>	576
5.94.4.36	<a href="#">mpi_swap</a>	576
5.94.4.37	<a href="#">mpi_write_binary</a>	576
5.94.4.38	<a href="#">mpi_write_file</a>	577
5.94.4.39	<a href="#">mpi_write_string</a>	577
5.95	<a href="#">/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/bn_mul.h File Reference</a>	579
5.95.1	<a href="#">Detailed Description</a>	579
5.95.2	<a href="#">Define Documentation</a>	579
5.95.2.1	<a href="#">MULADDC_CORE</a>	579
5.95.2.2	<a href="#">MULADDC_INIT</a>	579
5.95.2.3	<a href="#">MULADDC_STOP</a>	580
5.96	<a href="#">/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/certs.h File Reference</a>	581
5.96.1	<a href="#">Detailed Description</a>	581
5.96.2	<a href="#">Variable Documentation</a>	581
5.96.2.1	<a href="#">test_ca_cert</a>	581
5.96.2.2	<a href="#">test_ca_key</a>	581
5.96.2.3	<a href="#">test_ca_pwd</a>	581
5.96.2.4	<a href="#">test_cli_cert</a>	581
5.96.2.5	<a href="#">test_cli_key</a>	581
5.96.2.6	<a href="#">test_srv_cert</a>	582
5.96.2.7	<a href="#">test_srv_key</a>	582

5.96.2.8	xyssl_ca_cert	582
5.97	/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/debug.h File Reference	583
5.97.1	Detailed Description	583
5.97.2	Define Documentation	583
5.97.2.1	SSL_DEBUG_BUF	583
5.97.2.2	SSL_DEBUG_CRT	584
5.97.2.3	SSL_DEBUG_MPI	584
5.97.2.4	SSL_DEBUG_MSG	584
5.97.2.5	SSL_DEBUG_RET	584
5.97.3	Function Documentation	584
5.97.3.1	debug_fmt	584
5.97.3.2	debug_print_buf	585
5.97.3.3	debug_print_crt	585
5.97.3.4	debug_print_mpi	585
5.97.3.5	debug_print_msg	585
5.97.3.6	debug_print_ret	585
5.98	/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/des.h File Reference	586
5.98.1	Detailed Description	587
5.98.2	Define Documentation	587
5.98.2.1	DES_DECRYPT	587
5.98.2.2	DES_ENCRYPT	587
5.98.3	Function Documentation	587
5.98.3.1	des3_crypt_cbc	587
5.98.3.2	des3_crypt_ecb	587
5.98.3.3	des3_set2key_dec	588
5.98.3.4	des3_set2key_enc	588
5.98.3.5	des3_set3key_dec	588
5.98.3.6	des3_set3key_enc	589
5.98.3.7	des_crypt_cbc	589
5.98.3.8	des_crypt_ecb	589
5.98.3.9	des_self_test	590
5.98.3.10	des_setkey_dec	590
5.98.3.11	des_setkey_enc	590
5.99	/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/dhm.h File Reference	591
5.99.1	Detailed Description	591
5.99.2	Define Documentation	592

5.99.2.1	XYSSL_ERR_DHM_BAD_INPUT_DATA . . . . .	592
5.99.2.2	XYSSL_ERR_DHM_CALC_SECRET_FAILED . . . . .	592
5.99.2.3	XYSSL_ERR_DHM_MAKE_PARAMS_FAILED . . . . .	592
5.99.2.4	XYSSL_ERR_DHM_MAKE_PUBLIC_FAILED . . . . .	592
5.99.2.5	XYSSL_ERR_DHM_READ_PARAMS_FAILED . . . . .	592
5.99.2.6	XYSSL_ERR_DHM_READ_PUBLIC_FAILED . . . . .	592
5.99.3	Function Documentation . . . . .	592
5.99.3.1	dhm_calc_secret . . . . .	592
5.99.3.2	dhm_free . . . . .	593
5.99.3.3	dhm_make_params . . . . .	593
5.99.3.4	dhm_make_public . . . . .	593
5.99.3.5	dhm_read_params . . . . .	594
5.99.3.6	dhm_read_public . . . . .	594
5.99.3.7	dhm_self_test . . . . .	595
5.100	/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/havege.h File Reference . . . . .	596
5.100.1	Detailed Description . . . . .	596
5.100.2	Define Documentation . . . . .	596
5.100.2.1	COLLECT_SIZE . . . . .	596
5.100.3	Function Documentation . . . . .	596
5.100.3.1	havege_init . . . . .	596
5.100.3.2	havege_rand . . . . .	597
5.101	/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md2.h File Reference . . . . .	598
5.101.1	Detailed Description . . . . .	598
5.101.2	Function Documentation . . . . .	599
5.101.2.1	md2 . . . . .	599
5.101.2.2	md2_file . . . . .	599
5.101.2.3	md2_finish . . . . .	599
5.101.2.4	md2_hmac . . . . .	599
5.101.2.5	md2_hmac_finish . . . . .	600
5.101.2.6	md2_hmac_starts . . . . .	600
5.101.2.7	md2_hmac_update . . . . .	600
5.101.2.8	md2_self_test . . . . .	600
5.101.2.9	md2_starts . . . . .	600
5.101.2.10	md2_update . . . . .	601
5.102	/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md4.h File Reference . . . . .	602
5.102.1	Detailed Description . . . . .	602

5.102.2 Function Documentation . . . . .	603
5.102.2.1 md4 . . . . .	603
5.102.2.2 md4_file . . . . .	603
5.102.2.3 md4_finish . . . . .	603
5.102.2.4 md4_hmac . . . . .	603
5.102.2.5 md4_hmac_finish . . . . .	604
5.102.2.6 md4_hmac_starts . . . . .	604
5.102.2.7 md4_hmac_update . . . . .	604
5.102.2.8 md4_self_test . . . . .	604
5.102.2.9 md4_starts . . . . .	604
5.102.2.10md4_update . . . . .	605
5.103/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md5.h File Reference . .	606
5.103.1 Detailed Description . . . . .	606
5.103.2 Function Documentation . . . . .	607
5.103.2.1 md5 . . . . .	607
5.103.2.2 md5_file . . . . .	607
5.103.2.3 md5_finish . . . . .	607
5.103.2.4 md5_hmac . . . . .	608
5.103.2.5 md5_hmac_finish . . . . .	608
5.103.2.6 md5_hmac_starts . . . . .	608
5.103.2.7 md5_hmac_update . . . . .	609
5.103.2.8 md5_self_test . . . . .	609
5.103.2.9 md5_starts . . . . .	609
5.103.2.10md5_update . . . . .	609
5.104/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/net.h File Reference . . .	611
5.104.1 Detailed Description . . . . .	611
5.104.2 Define Documentation . . . . .	612
5.104.2.1 XYSSL_ERR_NET_ACCEPT_FAILED . . . . .	612
5.104.2.2 XYSSL_ERR_NET_BIND_FAILED . . . . .	612
5.104.2.3 XYSSL_ERR_NET_CONN_RESET . . . . .	612
5.104.2.4 XYSSL_ERR_NET_CONNECT_FAILED . . . . .	612
5.104.2.5 XYSSL_ERR_NET_LISTEN_FAILED . . . . .	612
5.104.2.6 XYSSL_ERR_NET_RECV_FAILED . . . . .	612
5.104.2.7 XYSSL_ERR_NET_SEND_FAILED . . . . .	612
5.104.2.8 XYSSL_ERR_NET_SOCKET_FAILED . . . . .	612
5.104.2.9 XYSSL_ERR_NET_TRY_AGAIN . . . . .	613

5.104.2.10XYSSL_ERR_NET_UNKNOWN_HOST . . . . .	613
5.104.3 Function Documentation . . . . .	613
5.104.3.1 net_accept . . . . .	613
5.104.3.2 net_bind . . . . .	613
5.104.3.3 net_close . . . . .	613
5.104.3.4 net_connect . . . . .	614
5.104.3.5 net_recv . . . . .	614
5.104.3.6 net_send . . . . .	614
5.104.3.7 net_set_block . . . . .	614
5.104.3.8 net_set_nonblock . . . . .	615
5.104.3.9 net_usleep . . . . .	615
5.105/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/openssl.h File Reference . . . . .	616
5.105.1 Detailed Description . . . . .	617
5.105.2 Define Documentation . . . . .	617
5.105.2.1 AES_BLOCK_SIZE . . . . .	617
5.105.2.2 AES_cbc_encrypt . . . . .	617
5.105.2.3 AES_KEY . . . . .	617
5.105.2.4 AES_set_decrypt_key . . . . .	617
5.105.2.5 AES_set_encrypt_key . . . . .	617
5.105.2.6 AES_SIZE . . . . .	617
5.105.2.7 d2i_RSAPrivateKey . . . . .	617
5.105.2.8 ERR_get_error . . . . .	617
5.105.2.9 MD5_CTX . . . . .	617
5.105.2.10MD5_Final . . . . .	618
5.105.2.11MD5_Init . . . . .	618
5.105.2.12MD5_Update . . . . .	618
5.105.2.13RSA . . . . .	618
5.105.2.14RSA_blinding_off . . . . .	618
5.105.2.15RSA_free . . . . .	618
5.105.2.16RSA_PKCS1_PADDING . . . . .	618
5.105.2.17RSA_size . . . . .	618
5.105.2.18SHA1_Final . . . . .	618
5.105.2.19SHA1_Init . . . . .	618
5.105.2.20SHA1_Update . . . . .	618
5.105.2.21SHA_CTX . . . . .	619
5.105.3 Function Documentation . . . . .	619



5.105.3.1	<a href="#">__RSA_Passthrough</a>	619
5.105.3.2	<a href="#">d2i_RSA_PUBKEY</a>	619
5.105.3.3	<a href="#">RSA_private_decrypt</a>	619
5.105.3.4	<a href="#">RSA_private_encrypt</a>	619
5.105.3.5	<a href="#">RSA_public_decrypt</a>	619
5.105.3.6	<a href="#">RSA_public_encrypt</a>	619
5.106/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/padlock.h	File Reference	620
5.106.1	Detailed Description	620
5.107/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/rsa.h	File Reference	621
5.107.1	Detailed Description	622
5.107.2	Define Documentation	622
5.107.2.1	<a href="#">ASN1_HASH_MDX</a>	622
5.107.2.2	<a href="#">ASN1_HASH_SHA1</a>	623
5.107.2.3	<a href="#">RSA_CRYPT</a>	623
5.107.2.4	<a href="#">RSA_MD2</a>	623
5.107.2.5	<a href="#">RSA_MD4</a>	623
5.107.2.6	<a href="#">RSA_MD5</a>	623
5.107.2.7	<a href="#">RSA_PKCS_V15</a>	623
5.107.2.8	<a href="#">RSA_PKCS_V21</a>	623
5.107.2.9	<a href="#">RSA_PRIVATE</a>	623
5.107.2.10	<a href="#">RSA_PUBLIC</a>	624
5.107.2.11	<a href="#">RSA_RAW</a>	624
5.107.2.12	<a href="#">RSA_SHA1</a>	624
5.107.2.13	<a href="#">RSA_SHA256</a>	624
5.107.2.14	<a href="#">RSA_SIGN</a>	624
5.107.2.15	<a href="#">XYSSL_ERR_RSA_BAD_INPUT_DATA</a>	624
5.107.2.16	<a href="#">XYSSL_ERR_RSA_INVALID_PADDING</a>	624
5.107.2.17	<a href="#">XYSSL_ERR_RSA_KEY_CHECK_FAILED</a>	624
5.107.2.18	<a href="#">XYSSL_ERR_RSA_KEY_GEN_FAILED</a>	625
5.107.2.19	<a href="#">XYSSL_ERR_RSA_PRIVATE_FAILED</a>	625
5.107.2.20	<a href="#">XYSSL_ERR_RSA_PUBLIC_FAILED</a>	625
5.107.2.21	<a href="#">XYSSL_ERR_RSA_VERIFY_FAILED</a>	625
5.107.3	Function Documentation	625
5.107.3.1	<a href="#">rsa_check_privkey</a>	625
5.107.3.2	<a href="#">rsa_check_pubkey</a>	625
5.107.3.3	<a href="#">rsa_free</a>	626

5.107.3.4	<a href="#">rsa_gen_key</a>	626
5.107.3.5	<a href="#">rsa_init</a>	626
5.107.3.6	<a href="#">rsa_pkcs1_decrypt</a>	627
5.107.3.7	<a href="#">rsa_pkcs1_encrypt</a>	627
5.107.3.8	<a href="#">rsa_pkcs1_sign</a>	628
5.107.3.9	<a href="#">rsa_pkcs1_verify</a>	629
5.107.3.10	<a href="#">rsa_private</a>	629
5.107.3.11	<a href="#">rsa_public</a>	630
5.107.3.12	<a href="#">rsa_self_test</a>	630
5.108	<a href="#">/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha1.h File Reference</a>	631
5.108.1	<a href="#">Detailed Description</a>	631
5.108.2	<a href="#">Function Documentation</a>	632
5.108.2.1	<a href="#">sha1</a>	632
5.108.2.2	<a href="#">sha1_file</a>	632
5.108.2.3	<a href="#">sha1_finish</a>	632
5.108.2.4	<a href="#">sha1_hmac</a>	633
5.108.2.5	<a href="#">sha1_hmac_finish</a>	633
5.108.2.6	<a href="#">sha1_hmac_starts</a>	633
5.108.2.7	<a href="#">sha1_hmac_update</a>	634
5.108.2.8	<a href="#">sha1_self_test</a>	634
5.108.2.9	<a href="#">sha1_starts</a>	634
5.108.2.10	<a href="#">sha1_update</a>	634
5.109	<a href="#">/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha2.h File Reference</a>	636
5.109.1	<a href="#">Detailed Description</a>	636
5.109.2	<a href="#">Function Documentation</a>	637
5.109.2.1	<a href="#">sha2</a>	637
5.109.2.2	<a href="#">sha2_file</a>	637
5.109.2.3	<a href="#">sha2_finish</a>	637
5.109.2.4	<a href="#">sha2_hmac</a>	638
5.109.2.5	<a href="#">sha2_hmac_finish</a>	638
5.109.2.6	<a href="#">sha2_hmac_starts</a>	638
5.109.2.7	<a href="#">sha2_hmac_update</a>	639
5.109.2.8	<a href="#">sha2_self_test</a>	639
5.109.2.9	<a href="#">sha2_starts</a>	639
5.109.2.10	<a href="#">sha2_update</a>	639
5.110	<a href="#">/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha4.h File Reference</a>	641

5.110.1 Detailed Description . . . . .	642
5.110.2 Define Documentation . . . . .	642
5.110.2.1 int64 . . . . .	642
5.110.2.2 UL64 . . . . .	642
5.110.3 Function Documentation . . . . .	642
5.110.3.1 sha4 . . . . .	642
5.110.3.2 sha4_file . . . . .	642
5.110.3.3 sha4_finish . . . . .	643
5.110.3.4 sha4_hmac . . . . .	643
5.110.3.5 sha4_hmac_finish . . . . .	643
5.110.3.6 sha4_hmac_starts . . . . .	644
5.110.3.7 sha4_hmac_update . . . . .	644
5.110.3.8 sha4_self_test . . . . .	644
5.110.3.9 sha4_starts . . . . .	644
5.110.3.10sha4_update . . . . .	645
5.111/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/ssl.h File Reference . . .	646
5.111.1 Detailed Description . . . . .	650
5.111.2 Define Documentation . . . . .	650
5.111.2.1 SSL_ALERT_CLOSE_NOTIFY . . . . .	650
5.111.2.2 SSL_ALERT_FATAL . . . . .	650
5.111.2.3 SSL_ALERT_NO_CERTIFICATE . . . . .	650
5.111.2.4 SSL_ALERT_WARNING . . . . .	650
5.111.2.5 SSL_BUFFER_LEN . . . . .	650
5.111.2.6 SSL_COMPRESS_NULL . . . . .	650
5.111.2.7 SSL_EDH_RSA_AES_256_SHA . . . . .	650
5.111.2.8 SSL_EDH_RSA_DES_168_SHA . . . . .	651
5.111.2.9 SSL_HS_CERTIFICATE . . . . .	651
5.111.2.10SSL_HS_CERTIFICATE_REQUEST . . . . .	651
5.111.2.11SSL_HS_CERTIFICATE_VERIFY . . . . .	651
5.111.2.12SSL_HS_CLIENT_HELLO . . . . .	651
5.111.2.13SSL_HS_CLIENT_KEY_EXCHANGE . . . . .	651
5.111.2.14SSL_HS_FINISHED . . . . .	651
5.111.2.15SSL_HS_HELLO_REQUEST . . . . .	651
5.111.2.16SSL_HS_SERVER_HELLO . . . . .	651
5.111.2.17SSL_HS_SERVER_HELLO_DONE . . . . .	652
5.111.2.18SSL_HS_SERVER_KEY_EXCHANGE . . . . .	652

5.111.2.19SSL_IS_CLIENT . . . . .	652
5.111.2.20SSL_IS_SERVER . . . . .	652
5.111.2.21SSL_MAJOR_VERSION_3 . . . . .	652
5.111.2.22SSL_MAX_CONTENT_LEN . . . . .	652
5.111.2.23SSL_MINOR_VERSION_0 . . . . .	652
5.111.2.24SSL_MINOR_VERSION_1 . . . . .	652
5.111.2.25SSL_MINOR_VERSION_2 . . . . .	653
5.111.2.26SSL_MSG_ALERT . . . . .	653
5.111.2.27SSL_MSG_APPLICATION_DATA . . . . .	653
5.111.2.28SSL_MSG_CHANGE_CIPHER_SPEC . . . . .	653
5.111.2.29SSL_MSG_HANDSHAKE . . . . .	653
5.111.2.30SSL_RSA_AES_128_SHA . . . . .	653
5.111.2.31SSL_RSA_AES_256_SHA . . . . .	653
5.111.2.32SSL_RSA_DES_168_SHA . . . . .	654
5.111.2.33SSL_RSA_RC4_128_MD5 . . . . .	654
5.111.2.34SSL_RSA_RC4_128_SHA . . . . .	654
5.111.2.35SSL_VERIFY_NONE . . . . .	654
5.111.2.36SSL_VERIFY_OPTIONAL . . . . .	654
5.111.2.37SSL_VERIFY_REQUIRED . . . . .	654
5.111.2.38TLS_EXT_SERVERNAME . . . . .	654
5.111.2.39TLS_EXT_SERVERNAME_HOSTNAME . . . . .	654
5.111.2.40XYSSL_ERR_SSL_BAD_HS_CERTIFICATE . . . . .	654
5.111.2.41XYSSL_ERR_SSL_BAD_HS_CERTIFICATE_REQUEST . . . . .	655
5.111.2.42XYSSL_ERR_SSL_BAD_HS_CERTIFICATE_VERIFY . . . . .	655
5.111.2.43XYSSL_ERR_SSL_BAD_HS_CHANGE_CIPHER_SPEC . . . . .	655
5.111.2.44XYSSL_ERR_SSL_BAD_HS_CLIENT_HELLO . . . . .	655
5.111.2.45XYSSL_ERR_SSL_BAD_HS_CLIENT_KEY_EXCHANGE . . . . .	655
5.111.2.46XYSSL_ERR_SSL_BAD_HS_FINISHED . . . . .	655
5.111.2.47XYSSL_ERR_SSL_BAD_HS_SERVER_HELLO . . . . .	655
5.111.2.48XYSSL_ERR_SSL_BAD_HS_SERVER_HELLO_DONE . . . . .	655
5.111.2.49XYSSL_ERR_SSL_BAD_HS_SERVER_KEY_EXCHANGE . . . . .	655
5.111.2.50XYSSL_ERR_SSL_BAD_INPUT_DATA . . . . .	656
5.111.2.51XYSSL_ERR_SSL_CA_CHAIN_REQUIRED . . . . .	656
5.111.2.52XYSSL_ERR_SSL_CERTIFICATE_REQUIRED . . . . .	656
5.111.2.53XYSSL_ERR_SSL_CERTIFICATE_TOO_LARGE . . . . .	656
5.111.2.54XYSSL_ERR_SSL_FATAL_ALERT_MESSAGE . . . . .	656

5.111.2.55	XYSSL_ERR_SSL_FEATURE_UNAVAILABLE . . . . .	656
5.111.2.56	XYSSL_ERR_SSL_INVALID_MAC . . . . .	656
5.111.2.57	XYSSL_ERR_SSL_INVALID_MODULUS_SIZE . . . . .	656
5.111.2.58	XYSSL_ERR_SSL_INVALID_RECORD . . . . .	656
5.111.2.59	XYSSL_ERR_SSL_NO_CIPHER_CHOSEN . . . . .	657
5.111.2.60	XYSSL_ERR_SSL_NO_CLIENT_CERTIFICATE . . . . .	657
5.111.2.61	XYSSL_ERR_SSL_NO_SESSION_FOUND . . . . .	657
5.111.2.62	XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY . . . . .	657
5.111.2.63	XYSSL_ERR_SSL_PEER_VERIFY_FAILED . . . . .	657
5.111.2.64	XYSSL_ERR_SSL_PRIVATE_KEY_REQUIRED . . . . .	657
5.111.2.65	XYSSL_ERR_SSL_UNEXPECTED_MESSAGE . . . . .	657
5.111.2.66	XYSSL_ERR_SSL_UNKNOWN_CIPHER . . . . .	657
5.111.3	Typedef Documentation . . . . .	658
5.111.3.1	ssl_context . . . . .	658
5.111.3.2	ssl_session . . . . .	658
5.111.4	Enumeration Type Documentation . . . . .	658
5.111.4.1	ssl_states . . . . .	658
5.111.5	Function Documentation . . . . .	658
5.111.5.1	ssl_calc_verify . . . . .	658
5.111.5.2	ssl_close_notify . . . . .	659
5.111.5.3	ssl_derive_keys . . . . .	659
5.111.5.4	ssl_fetch_input . . . . .	659
5.111.5.5	ssl_flush_output . . . . .	659
5.111.5.6	ssl_free . . . . .	659
5.111.5.7	ssl_get_bytes_avail . . . . .	660
5.111.5.8	ssl_get_cipher . . . . .	660
5.111.5.9	ssl_get_verify_result . . . . .	660
5.111.5.10	ssl_handshake . . . . .	661
5.111.5.11	ssl_handshake_client . . . . .	661
5.111.5.12	ssl_handshake_server . . . . .	661
5.111.5.13	ssl_init . . . . .	661
5.111.5.14	ssl_parse_certificate . . . . .	662
5.111.5.15	ssl_parse_change_cipher_spec . . . . .	662
5.111.5.16	ssl_parse_finished . . . . .	662
5.111.5.17	ssl_read . . . . .	663
5.111.5.18	ssl_read_record . . . . .	663

5.111.5.19	ssl_set_authmode . . . . .	663
5.111.5.20	ssl_set_bio . . . . .	664
5.111.5.21	ssl_set_ca_chain . . . . .	664
5.111.5.22	ssl_set_ciphers . . . . .	664
5.111.5.23	ssl_set_dbg . . . . .	665
5.111.5.24	ssl_set_dh_param . . . . .	665
5.111.5.25	ssl_set_endpoint . . . . .	665
5.111.5.26	ssl_set_hostname . . . . .	666
5.111.5.27	ssl_set_own_cert . . . . .	666
5.111.5.28	ssl_set_rng . . . . .	666
5.111.5.29	ssl_set_scb . . . . .	667
5.111.5.30	ssl_set_session . . . . .	667
5.111.5.31	ssl_write . . . . .	667
5.111.5.32	ssl_write_certificate . . . . .	668
5.111.5.33	ssl_write_change_cipher_spec . . . . .	668
5.111.5.34	ssl_write_finished . . . . .	668
5.111.5.35	ssl_write_record . . . . .	668
5.111.6	Variable Documentation . . . . .	669
5.111.6.1	ssl_default_ciphers . . . . .	669
5.112	/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/timing.h File Reference . . . . .	670
5.112.1	Detailed Description . . . . .	670
5.112.2	Function Documentation . . . . .	670
5.112.2.1	get_timer . . . . .	670
5.112.2.2	hardlock . . . . .	671
5.112.2.3	m_sleep . . . . .	671
5.112.2.4	set_alarm . . . . .	671
5.112.3	Variable Documentation . . . . .	671
5.112.3.1	alarmed . . . . .	671
5.113	/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/x509.h File Reference . . . . .	672
5.113.1	Detailed Description . . . . .	674
5.113.2	Define Documentation . . . . .	674
5.113.2.1	ASN1_BIT_STRING . . . . .	674
5.113.2.2	ASN1_BMP_STRING . . . . .	674
5.113.2.3	ASN1_BOOLEAN . . . . .	675
5.113.2.4	ASN1_CONSTRUCTED . . . . .	675
5.113.2.5	ASN1_CONTEXT_SPECIFIC . . . . .	675

5.113.2.6 ASN1_IA5_STRING . . . . .	675
5.113.2.7 ASN1_INTEGER . . . . .	675
5.113.2.8 ASN1_NULL . . . . .	675
5.113.2.9 ASN1_OCTET_STRING . . . . .	675
5.113.2.10ASN1_OID . . . . .	675
5.113.2.11ASN1_PRIMITIVE . . . . .	675
5.113.2.12ASN1_PRINTABLE_STRING . . . . .	676
5.113.2.13ASN1_SEQUENCE . . . . .	676
5.113.2.14ASN1_SET . . . . .	676
5.113.2.15ASN1_T61_STRING . . . . .	676
5.113.2.16ASN1_UNIVERSAL_STRING . . . . .	676
5.113.2.17ASN1_UTC_TIME . . . . .	676
5.113.2.18ASN1_UTF8_STRING . . . . .	676
5.113.2.19BADCERT_CN_MISMATCH . . . . .	676
5.113.2.20BADCERT_EXPIRED . . . . .	677
5.113.2.21BADCERT_NOT_TRUSTED . . . . .	677
5.113.2.22BADCERT_REVOKED . . . . .	677
5.113.2.23OID_CN . . . . .	677
5.113.2.24OID_PKCS1 . . . . .	677
5.113.2.25OID_PKCS1_RSA . . . . .	677
5.113.2.26OID_PKCS1_RSA_SHA . . . . .	677
5.113.2.27OID_PKCS9 . . . . .	677
5.113.2.28OID_PKCS9_EMAIL . . . . .	677
5.113.2.29OID_X520 . . . . .	678
5.113.2.30PEM_LINE_LENGTH . . . . .	678
5.113.2.31PKCS9_EMAIL . . . . .	678
5.113.2.32X509_ISSUER . . . . .	678
5.113.2.33X509_OUTPUT_DER . . . . .	678
5.113.2.34X509_OUTPUT_PEM . . . . .	678
5.113.2.35X509_SUBJECT . . . . .	678
5.113.2.36X520_COMMON_NAME . . . . .	678
5.113.2.37X520_COUNTRY . . . . .	678
5.113.2.38X520_LOCALITY . . . . .	678
5.113.2.39X520_ORG_UNIT . . . . .	679
5.113.2.40X520_ORGANIZATION . . . . .	679
5.113.2.41X520_STATE . . . . .	679

5.113.2.42XYSSL_ERR_ASN1_INVALID_DATA . . . . .	679
5.113.2.43XYSSL_ERR_ASN1_INVALID_LENGTH . . . . .	679
5.113.2.44XYSSL_ERR_ASN1_LENGTH_MISMATCH . . . . .	679
5.113.2.45XYSSL_ERR_ASN1_OUT_OF_DATA . . . . .	679
5.113.2.46XYSSL_ERR_ASN1_UNEXPECTED_TAG . . . . .	679
5.113.2.47XYSSL_ERR_X509_CERT_INVALID_ALG . . . . .	680
5.113.2.48XYSSL_ERR_X509_CERT_INVALID_DATE . . . . .	680
5.113.2.49XYSSL_ERR_X509_CERT_INVALID_EXTENSIONS . . . . .	680
5.113.2.50XYSSL_ERR_X509_CERT_INVALID_FORMAT . . . . .	680
5.113.2.51XYSSL_ERR_X509_CERT_INVALID_NAME . . . . .	680
5.113.2.52XYSSL_ERR_X509_CERT_INVALID_PEM . . . . .	680
5.113.2.53XYSSL_ERR_X509_CERT_INVALID_PUBKEY . . . . .	680
5.113.2.54XYSSL_ERR_X509_CERT_INVALID_SERIAL . . . . .	680
5.113.2.55XYSSL_ERR_X509_CERT_INVALID_SIGNATURE . . . . .	680
5.113.2.56XYSSL_ERR_X509_CERT_INVALID_VERSION . . . . .	681
5.113.2.57XYSSL_ERR_X509_CERT_SIG_MISMATCH . . . . .	681
5.113.2.58XYSSL_ERR_X509_CERT_UNKNOWN_PK_ALG . . . . .	681
5.113.2.59XYSSL_ERR_X509_CERT_UNKNOWN_SIG_ALG . . . . .	681
5.113.2.60XYSSL_ERR_X509_CERT_UNKNOWN_VERSION . . . . .	681
5.113.2.61XYSSL_ERR_X509_CERT_VERIFY_FAILED . . . . .	681
5.113.2.62XYSSL_ERR_X509_FEATURE_UNAVAILABLE . . . . .	681
5.113.2.63XYSSL_ERR_X509_KEY_INVALID_ENC_IV . . . . .	681
5.113.2.64XYSSL_ERR_X509_KEY_INVALID_FORMAT . . . . .	681
5.113.2.65XYSSL_ERR_X509_KEY_INVALID_PEM . . . . .	682
5.113.2.66XYSSL_ERR_X509_KEY_INVALID_VERSION . . . . .	682
5.113.2.67XYSSL_ERR_X509_KEY_PASSWORD_MISMATCH . . . . .	682
5.113.2.68XYSSL_ERR_X509_KEY_PASSWORD_REQUIRED . . . . .	682
5.113.2.69XYSSL_ERR_X509_KEY_UNKNOWN_ENC_ALG . . . . .	682
5.113.2.70XYSSL_ERR_X509_POINT_ERROR . . . . .	682
5.113.2.71XYSSL_ERR_X509_VALUE_TO_LENGTH . . . . .	682
5.113.3 Typedef Documentation . . . . .	683
5.113.3.1 x509_buf . . . . .	683
5.113.3.2 x509_cert . . . . .	683
5.113.3.3 x509_name . . . . .	683
5.113.3.4 x509_node . . . . .	683
5.113.3.5 x509_raw . . . . .	683



5.113.3.6 x509_time . . . . .	683
5.113.4 Function Documentation . . . . .	683
5.113.4.1 x509_free . . . . .	683
5.113.4.2 x509_self_test . . . . .	683
5.113.4.3 x509parse_cert_info . . . . .	683
5.113.4.4 x509parse_cert . . . . .	684
5.113.4.5 x509parse_certfile . . . . .	684
5.113.4.6 x509parse_dn_gets . . . . .	684
5.113.4.7 x509parse_expired . . . . .	685
5.113.4.8 x509parse_key . . . . .	685
5.113.4.9 x509parse_keyfile . . . . .	685
5.113.4.10 x509parse_verify . . . . .	686
5.114/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/aes.c File Reference . . . . .	687
5.114.1 Define Documentation . . . . .	688
5.114.1.1 AES_FROUND . . . . .	688
5.114.1.2 AES_RROUND . . . . .	689
5.114.1.3 GET_ULONG_LE . . . . .	689
5.114.1.4 MUL . . . . .	689
5.114.1.5 PUT_ULONG_LE . . . . .	689
5.114.1.6 ROTL8 . . . . .	690
5.114.1.7 XTIME . . . . .	690
5.114.2 Function Documentation . . . . .	690
5.114.2.1 aes_crypt_cbc . . . . .	690
5.114.2.2 aes_crypt_cfb . . . . .	690
5.114.2.3 aes_crypt_ecb . . . . .	691
5.114.2.4 aes_gen_tables . . . . .	691
5.114.2.5 aes_self_test . . . . .	691
5.114.2.6 aes_setkey_dec . . . . .	691
5.114.2.7 aes_setkey_enc . . . . .	692
5.114.3 Variable Documentation . . . . .	692
5.114.3.1 aes_init_done . . . . .	692
5.114.3.2 aes_test_cbc_dec . . . . .	692
5.114.3.3 aes_test_cbc_enc . . . . .	693
5.114.3.4 aes_test_cfb_dec . . . . .	693
5.114.3.5 aes_test_cfb_enc . . . . .	693
5.114.3.6 aes_test_ecb_dec . . . . .	693

5.114.3.7 aes_test_ecb_enc . . . . .	694
5.114.3.8 FSb . . . . .	694
5.114.3.9 FT0 . . . . .	694
5.114.3.10FT1 . . . . .	694
5.114.3.11FT2 . . . . .	694
5.114.3.12FT3 . . . . .	695
5.114.3.13RCON . . . . .	695
5.114.3.14RSb . . . . .	695
5.114.3.15RT0 . . . . .	695
5.114.3.16RT1 . . . . .	695
5.114.3.17RT2 . . . . .	695
5.114.3.18RT3 . . . . .	695
5.115/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/arc4.c File Reference . . . . .	696
5.115.1 Function Documentation . . . . .	696
5.115.1.1 arc4_crypt . . . . .	696
5.115.1.2 arc4_self_test . . . . .	696
5.115.1.3 arc4_setup . . . . .	697
5.115.2 Variable Documentation . . . . .	697
5.115.2.1 arc4_test_ct . . . . .	697
5.115.2.2 arc4_test_key . . . . .	697
5.115.2.3 arc4_test_pt . . . . .	697
5.116/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/base64.c File Reference . . . . .	699
5.116.1 Function Documentation . . . . .	699
5.116.1.1 base64_decode . . . . .	699
5.116.1.2 base64_encode . . . . .	700
5.116.1.3 base64_self_test . . . . .	700
5.116.2 Variable Documentation . . . . .	700
5.116.2.1 base64_dec_map . . . . .	700
5.116.2.2 base64_enc_map . . . . .	701
5.116.2.3 base64_test_dec . . . . .	701
5.116.2.4 base64_test_enc . . . . .	701
5.117/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/bignum.c File Reference . . . . .	703
5.117.1 Define Documentation . . . . .	706
5.117.1.1 biH . . . . .	706
5.117.1.2 biL . . . . .	706
5.117.1.3 BITS_TO_LIMBS . . . . .	706

5.117.1.4 CHARS_TO_LIMBS . . . . .	706
5.117.1.5 ciL . . . . .	706
5.117.2 Function Documentation . . . . .	706
5.117.2.1 mpi_add_abs . . . . .	706
5.117.2.2 mpi_add_int . . . . .	706
5.117.2.3 mpi_add_mpi . . . . .	707
5.117.2.4 mpi_cmp_abs . . . . .	707
5.117.2.5 mpi_cmp_int . . . . .	707
5.117.2.6 mpi_cmp_mpi . . . . .	707
5.117.2.7 mpi_copy . . . . .	708
5.117.2.8 mpi_div_int . . . . .	708
5.117.2.9 mpi_div_mpi . . . . .	708
5.117.2.10 mpi_exp_mod . . . . .	709
5.117.2.11 mpi_free . . . . .	709
5.117.2.12 mpi_gcd . . . . .	709
5.117.2.13 mpi_gen_prime . . . . .	710
5.117.2.14 mpi_get_digit . . . . .	710
5.117.2.15 mpi_grow . . . . .	710
5.117.2.16 mpi_init . . . . .	710
5.117.2.17 mpi_inv_mod . . . . .	711
5.117.2.18 mpi_is_prime . . . . .	711
5.117.2.19 mpi_lsb . . . . .	711
5.117.2.20 mpi_lset . . . . .	711
5.117.2.21 mpi_mod_int . . . . .	712
5.117.2.22 mpi_mod_mpi . . . . .	712
5.117.2.23 mpi_montg_init . . . . .	712
5.117.2.24 mpi_montmul . . . . .	712
5.117.2.25 mpi_montred . . . . .	713
5.117.2.26 mpi_msb . . . . .	713
5.117.2.27 mpi_mul_hlp . . . . .	713
5.117.2.28 mpi_mul_int . . . . .	713
5.117.2.29 mpi_mul_mpi . . . . .	713
5.117.2.30 mpi_read_binary . . . . .	714
5.117.2.31 mpi_read_file . . . . .	714
5.117.2.32 mpi_read_mystring . . . . .	714
5.117.2.33 mpi_read_string . . . . .	714

5.117.2.34	<a href="#">mpi_self_test</a>	715
5.117.2.35	<a href="#">mpi_shift_l</a>	715
5.117.2.36	<a href="#">mpi_shift_r</a>	715
5.117.2.37	<a href="#">mpi_size</a>	716
5.117.2.38	<a href="#">mpi_sub_abs</a>	716
5.117.2.39	<a href="#">mpi_sub_hlp</a>	716
5.117.2.40	<a href="#">mpi_sub_int</a>	716
5.117.2.41	<a href="#">mpi_sub_mpi</a>	716
5.117.2.42	<a href="#">mpi_swap</a>	717
5.117.2.43	<a href="#">mpi_write_binary</a>	717
5.117.2.44	<a href="#">mpi_write_file</a>	717
5.117.2.45	<a href="#">mpi_write_hlp</a>	718
5.117.2.46	<a href="#">mpi_write_string</a>	718
5.117.3	<a href="#">Variable Documentation</a>	718
5.117.3.1	<a href="#">small_prime</a>	718
5.118/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/certs.c	<a href="#">File Reference</a>	720
5.118.1	<a href="#">Variable Documentation</a>	720
5.118.1.1	<a href="#">test_ca_cert</a>	720
5.118.1.2	<a href="#">test_ca_key</a>	720
5.118.1.3	<a href="#">test_ca_pwd</a>	721
5.118.1.4	<a href="#">test_cli_cert</a>	721
5.118.1.5	<a href="#">test_cli_key</a>	721
5.118.1.6	<a href="#">test_srv_cert</a>	722
5.118.1.7	<a href="#">test_srv_key</a>	722
5.118.1.8	<a href="#">xyssl_ca_cert</a>	723
5.119/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/debug.c	<a href="#">File Reference</a>	724
5.119.1	<a href="#">Function Documentation</a>	724
5.119.1.1	<a href="#">debug_fmt</a>	724
5.119.1.2	<a href="#">debug_print_buf</a>	724
5.119.1.3	<a href="#">debug_print_cert</a>	724
5.119.1.4	<a href="#">debug_print_mpi</a>	724
5.119.1.5	<a href="#">debug_print_msg</a>	725
5.119.1.6	<a href="#">debug_print_ret</a>	725
5.120/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/des.c	<a href="#">File Reference</a>	726
5.120.1	<a href="#">Define Documentation</a>	727
5.120.1.1	<a href="#">DES_FP</a>	727

5.120.1.2 DES_IP	728
5.120.1.3 DES_ROUND	728
5.120.1.4 GET_ULONG_BE	728
5.120.1.5 PUT_ULONG_BE	728
5.120.1.6 SWAP	729
5.120.2 Function Documentation	729
5.120.2.1 des3_crypt_cbc	729
5.120.2.2 des3_crypt_ecb	729
5.120.2.3 des3_set2key	730
5.120.2.4 des3_set2key_dec	730
5.120.2.5 des3_set2key_enc	730
5.120.2.6 des3_set3key	730
5.120.2.7 des3_set3key_dec	730
5.120.2.8 des3_set3key_enc	731
5.120.2.9 des_crypt_cbc	731
5.120.2.10 des_crypt_ecb	731
5.120.2.11 des_self_test	732
5.120.2.12 des_setkey	732
5.120.2.13 des_setkey_dec	732
5.120.2.14 des_setkey_enc	732
5.120.3 Variable Documentation	733
5.120.3.1 des3_test_buf	733
5.120.3.2 des3_test_cbc_dec	733
5.120.3.3 des3_test_cbc_enc	733
5.120.3.4 des3_test_ecb_dec	733
5.120.3.5 des3_test_ecb_enc	734
5.120.3.6 des3_test_iv	734
5.120.3.7 des3_test_keys	734
5.120.3.8 LHs	734
5.120.3.9 RHs	735
5.120.3.10 SB1	735
5.120.3.11 SB2	735
5.120.3.12 SB3	736
5.120.3.13 SB4	736
5.120.3.14 SB5	736
5.120.3.15 SB6	737

5.120.3.16SB7 . . . . .	737
5.120.3.17SB8 . . . . .	738
5.121/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/dhm.c File Reference . . . . .	739
5.121.1 Define Documentation . . . . .	739
5.121.1.1 DHM_MPI_EXPORT . . . . .	739
5.121.2 Function Documentation . . . . .	740
5.121.2.1 dhm_calc_secret . . . . .	740
5.121.2.2 dhm_free . . . . .	740
5.121.2.3 dhm_make_params . . . . .	740
5.121.2.4 dhm_make_public . . . . .	741
5.121.2.5 dhm_read_bignum . . . . .	741
5.121.2.6 dhm_read_params . . . . .	741
5.121.2.7 dhm_read_public . . . . .	742
5.121.2.8 dhm_self_test . . . . .	742
5.122/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/havege.c File Reference . . . . .	743
5.122.1 Define Documentation . . . . .	743
5.122.1.1 ONE_ITERATION . . . . .	743
5.122.1.2 SWAP . . . . .	743
5.122.1.3 TST1_ENTER . . . . .	743
5.122.1.4 TST1_LEAVE . . . . .	743
5.122.1.5 TST2_ENTER . . . . .	744
5.122.1.6 TST2_LEAVE . . . . .	744
5.122.2 Function Documentation . . . . .	744
5.122.2.1 havege_fill . . . . .	744
5.122.2.2 havege_init . . . . .	744
5.122.2.3 havege_rand . . . . .	744
5.123/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md2.c File Reference . . . . .	745
5.124/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md4.c File Reference . . . . .	746
5.125/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md5.c File Reference . . . . .	747
5.125.1 Define Documentation . . . . .	748
5.125.1.1 F . . . . .	748
5.125.1.2 F . . . . .	748
5.125.1.3 F . . . . .	748
5.125.1.4 F . . . . .	748
5.125.1.5 GET_ULONG_LE . . . . .	748
5.125.1.6 P . . . . .	748

5.125.1.7 PUT_ULONG_LE . . . . .	749
5.125.1.8 S . . . . .	749
5.125.2 Function Documentation . . . . .	749
5.125.2.1 md5 . . . . .	749
5.125.2.2 md5_file . . . . .	749
5.125.2.3 md5_finish . . . . .	750
5.125.2.4 md5_hmac . . . . .	750
5.125.2.5 md5_hmac_finish . . . . .	750
5.125.2.6 md5_hmac_starts . . . . .	751
5.125.2.7 md5_hmac_update . . . . .	751
5.125.2.8 md5_process . . . . .	751
5.125.2.9 md5_self_test . . . . .	751
5.125.2.10md5_starts . . . . .	752
5.125.2.11md5_update . . . . .	752
5.125.3 Variable Documentation . . . . .	752
5.125.3.1 md5_hmac_test_buf . . . . .	752
5.125.3.2 md5_hmac_test_buflen . . . . .	753
5.125.3.3 md5_hmac_test_key . . . . .	753
5.125.3.4 md5_hmac_test_keylen . . . . .	753
5.125.3.5 md5_hmac_test_sum . . . . .	753
5.125.3.6 md5_padding . . . . .	754
5.125.3.7 md5_test_buf . . . . .	754
5.125.3.8 md5_test_buflen . . . . .	754
5.125.3.9 md5_test_sum . . . . .	755
5.126/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/net.c File Reference . . . . .	756
5.126.1 Function Documentation . . . . .	757
5.126.1.1 net_accept . . . . .	757
5.126.1.2 net_bind . . . . .	757
5.126.1.3 net_close . . . . .	757
5.126.1.4 net_connect . . . . .	757
5.126.1.5 net_h tons . . . . .	758
5.126.1.6 net_is_blocking . . . . .	758
5.126.1.7 net_recv . . . . .	758
5.126.1.8 net_send . . . . .	758
5.126.1.9 net_set_block . . . . .	759
5.126.1.10net_set_nonblock . . . . .	759

5.126.1.1 <code>lnet_usleep</code> . . . . .	759
5.127/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/padlock.c File Reference . . . . .	760
5.128/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/rsa.c File Reference . . . . .	761
5.128.1 Define Documentation . . . . .	762
5.128.1.1 <code>KEY_LEN</code> . . . . .	762
5.128.1.2 <code>PT_LEN</code> . . . . .	762
5.128.1.3 <code>RSA_D</code> . . . . .	762
5.128.1.4 <code>RSA_DP</code> . . . . .	763
5.128.1.5 <code>RSA_DQ</code> . . . . .	763
5.128.1.6 <code>RSA_E</code> . . . . .	763
5.128.1.7 <code>RSA_N</code> . . . . .	763
5.128.1.8 <code>RSA_P</code> . . . . .	763
5.128.1.9 <code>RSA_PT</code> . . . . .	764
5.128.1.10 <code>RSA_Q</code> . . . . .	764
5.128.1.11 <code>RSA_QP</code> . . . . .	764
5.128.2 Function Documentation . . . . .	764
5.128.2.1 <code>rsa_check_privkey</code> . . . . .	764
5.128.2.2 <code>rsa_check_pubkey</code> . . . . .	765
5.128.2.3 <code>rsa_free</code> . . . . .	765
5.128.2.4 <code>rsa_gen_key</code> . . . . .	765
5.128.2.5 <code>rsa_init</code> . . . . .	766
5.128.2.6 <code>rsa_pkcs1_decrypt</code> . . . . .	766
5.128.2.7 <code>rsa_pkcs1_encrypt</code> . . . . .	767
5.128.2.8 <code>rsa_pkcs1_sign</code> . . . . .	767
5.128.2.9 <code>rsa_pkcs1_verify</code> . . . . .	768
5.128.2.10 <code>rsa_private</code> . . . . .	768
5.128.2.11 <code>rsa_public</code> . . . . .	769
5.128.2.12 <code>rsa_self_test</code> . . . . .	769
5.129/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha1.c File Reference . . . . .	771
5.129.1 Define Documentation . . . . .	772
5.129.1.1 <code>F</code> . . . . .	772
5.129.1.2 <code>F</code> . . . . .	772
5.129.1.3 <code>F</code> . . . . .	772
5.129.1.4 <code>F</code> . . . . .	772
5.129.1.5 <code>GET_ULONG_BE</code> . . . . .	772
5.129.1.6 <code>K</code> . . . . .	773



5.129.1.7 K . . . . .	773
5.129.1.8 K . . . . .	773
5.129.1.9 K . . . . .	773
5.129.1.10P . . . . .	773
5.129.1.11PUT_ULONG_BE . . . . .	773
5.129.1.12R . . . . .	773
5.129.1.13S . . . . .	773
5.129.2 Function Documentation . . . . .	773
5.129.2.1 sha1 . . . . .	773
5.129.2.2 sha1_file . . . . .	774
5.129.2.3 sha1_finish . . . . .	774
5.129.2.4 sha1_hmac . . . . .	774
5.129.2.5 sha1_hmac_finish . . . . .	775
5.129.2.6 sha1_hmac_starts . . . . .	775
5.129.2.7 sha1_hmac_update . . . . .	775
5.129.2.8 sha1_process . . . . .	776
5.129.2.9 sha1_self_test . . . . .	776
5.129.2.10sha1_starts . . . . .	776
5.129.2.11sha1_update . . . . .	776
5.129.3 Variable Documentation . . . . .	777
5.129.3.1 sha1_hmac_test_buf . . . . .	777
5.129.3.2 sha1_hmac_test_buflen . . . . .	777
5.129.3.3 sha1_hmac_test_key . . . . .	777
5.129.3.4 sha1_hmac_test_keylen . . . . .	778
5.129.3.5 sha1_hmac_test_sum . . . . .	778
5.129.3.6 sha1_padding . . . . .	778
5.129.3.7 sha1_test_buf . . . . .	779
5.129.3.8 sha1_test_buflen . . . . .	779
5.129.3.9 sha1_test_sum . . . . .	779
5.130/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha2.c File Reference . . . . .	780
5.130.1 Define Documentation . . . . .	781
5.130.1.1 F0 . . . . .	781
5.130.1.2 F1 . . . . .	781
5.130.1.3 GET_ULONG_BE . . . . .	781
5.130.1.4 P . . . . .	781
5.130.1.5 PUT_ULONG_BE . . . . .	782

5.130.1.6 R . . . . .	782
5.130.1.7 ROTR . . . . .	782
5.130.1.8 S0 . . . . .	782
5.130.1.9 S1 . . . . .	782
5.130.1.10 S2 . . . . .	782
5.130.1.11 S3 . . . . .	782
5.130.1.12 SHR . . . . .	782
5.130.2 Function Documentation . . . . .	782
5.130.2.1 sha2 . . . . .	782
5.130.2.2 sha2_file . . . . .	783
5.130.2.3 sha2_finish . . . . .	783
5.130.2.4 sha2_hmac . . . . .	783
5.130.2.5 sha2_hmac_finish . . . . .	784
5.130.2.6 sha2_hmac_starts . . . . .	784
5.130.2.7 sha2_hmac_update . . . . .	784
5.130.2.8 sha2_process . . . . .	785
5.130.2.9 sha2_self_test . . . . .	785
5.130.2.10 sha2_starts . . . . .	785
5.130.2.11 sha2_update . . . . .	785
5.130.3 Variable Documentation . . . . .	786
5.130.3.1 sha2_hmac_test_buf . . . . .	786
5.130.3.2 sha2_hmac_test_buflen . . . . .	786
5.130.3.3 sha2_hmac_test_key . . . . .	786
5.130.3.4 sha2_hmac_test_keylen . . . . .	787
5.130.3.5 sha2_hmac_test_sum . . . . .	787
5.130.3.6 sha2_padding . . . . .	787
5.130.3.7 sha2_test_buf . . . . .	787
5.130.3.8 sha2_test_buflen . . . . .	787
5.130.3.9 sha2_test_sum . . . . .	788
5.131/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha4.c File Reference . . . . .	789
5.131.1 Define Documentation . . . . .	790
5.131.1.1 F0 . . . . .	790
5.131.1.2 F1 . . . . .	790
5.131.1.3 GET_UINT64_BE . . . . .	790
5.131.1.4 P . . . . .	790
5.131.1.5 PUT_UINT64_BE . . . . .	791

5.131.1.6 ROTR	791
5.131.1.7 S0	791
5.131.1.8 S1	791
5.131.1.9 S2	791
5.131.1.10 S3	791
5.131.1.11 SHR	791
5.131.2 Function Documentation	791
5.131.2.1 sha4	791
5.131.2.2 sha4_file	792
5.131.2.3 sha4_finish	792
5.131.2.4 sha4_hmac	792
5.131.2.5 sha4_hmac_finish	793
5.131.2.6 sha4_hmac_starts	793
5.131.2.7 sha4_hmac_update	793
5.131.2.8 sha4_process	793
5.131.2.9 sha4_self_test	794
5.131.2.10 sha4_starts	794
5.131.2.11 sha4_update	794
5.131.3 Variable Documentation	794
5.131.3.1 K	794
5.131.3.2 sha4_hmac_test_buf	795
5.131.3.3 sha4_hmac_test_buflen	795
5.131.3.4 sha4_hmac_test_key	795
5.131.3.5 sha4_hmac_test_keylen	796
5.131.3.6 sha4_hmac_test_sum	796
5.131.3.7 sha4_padding	796
5.131.3.8 sha4_test_buf	796
5.131.3.9 sha4_test_buflen	797
5.131.3.10 sha4_test_sum	797
5.132/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl_cli.c File Reference	798
5.132.1 Function Documentation	798
5.132.1.1 ssl_handshake_client	798
5.132.1.2 ssl_parse_certificate_request	798
5.132.1.3 ssl_parse_server_hello	799
5.132.1.4 ssl_parse_server_hello_done	799
5.132.1.5 ssl_parse_server_key_exchange	799

5.132.1.6	ssl_write_certificate_verify	799
5.132.1.7	ssl_write_client_hello	800
5.132.1.8	ssl_write_client_key_exchange	800
5.133/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl_srv.c	File Reference	801
5.133.1	Function Documentation	801
5.133.1.1	ssl_handshake_server	801
5.133.1.2	ssl_parse_certificate_verify	801
5.133.1.3	ssl_parse_client_hello	802
5.133.1.4	ssl_parse_client_key_exchange	802
5.133.1.5	ssl_write_certificate_request	802
5.133.1.6	ssl_write_server_hello	802
5.133.1.7	ssl_write_server_hello_done	803
5.133.1.8	ssl_write_server_key_exchange	803
5.134/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl_tls.c	File Reference	804
5.134.1	Function Documentation	806
5.134.1.1	ssl_calc_finished	806
5.134.1.2	ssl_calc_verify	806
5.134.1.3	ssl_close_notify	806
5.134.1.4	ssl_decrypt_buf	806
5.134.1.5	ssl_derive_keys	807
5.134.1.6	ssl_encrypt_buf	807
5.134.1.7	ssl_fetch_input	807
5.134.1.8	ssl_flush_output	807
5.134.1.9	ssl_free	808
5.134.1.10	ssl_get_bytes_avail	808
5.134.1.11	ssl_get_cipher	808
5.134.1.12	ssl_get_verify_result	808
5.134.1.13	ssl_handshake	809
5.134.1.14	ssl_init	809
5.134.1.15	ssl_mac_md5	809
5.134.1.16	ssl_mac_sha1	810
5.134.1.17	ssl_parse_certificate	810
5.134.1.18	ssl_parse_change_cipher_spec	810
5.134.1.19	ssl_parse_finished	810
5.134.1.20	ssl_read	810
5.134.1.21	ssl_read_record	811

5.134.1.22	ssl_set_authmode	811
5.134.1.23	ssl_set_bio	812
5.134.1.24	ssl_set_ca_chain	812
5.134.1.25	ssl_set_ciphers	812
5.134.1.26	ssl_set_dbg	813
5.134.1.27	ssl_set_dh_param	813
5.134.1.28	ssl_set_endpoint	813
5.134.1.29	ssl_set_hostname	814
5.134.1.30	ssl_set_own_cert	814
5.134.1.31	ssl_set_rng	814
5.134.1.32	ssl_set_scb	815
5.134.1.33	ssl_set_session	815
5.134.1.34	ssl_write	815
5.134.1.35	ssl_write_certificate	816
5.134.1.36	ssl_write_change_cipher_spec	816
5.134.1.37	ssl_write_finished	816
5.134.1.38	ssl_write_record	816
5.134.1.39	ssl_prf	817
5.134.2	Variable Documentation	817
5.134.2.1	ssl_default_ciphers	817
5.135/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/timing.c	File Reference	818
5.135.1	Function Documentation	818
5.135.1.1	get_timer	818
5.135.1.2	hardclock	819
5.135.1.3	m_sleep	819
5.135.1.4	set_alarm	819
5.135.1.5	sighandler	819
5.135.2	Variable Documentation	819
5.135.2.1	alarmed	819
5.135.2.2	hardclock_init	819
5.135.2.3	tv_init	820
5.136/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/x509parse.c	File Reference	821
5.136.1	Function Documentation	822
5.136.1.1	asn1_get_bool	822
5.136.1.2	asn1_get_int	822
5.136.1.3	asn1_get_len	822

5.136.1.4	<a href="#">asn1_get_mpi</a>	823
5.136.1.5	<a href="#">asn1_get_tag</a>	823
5.136.1.6	<a href="#">x509_des3_decrypt</a>	823
5.136.1.7	<a href="#">x509_free</a>	823
5.136.1.8	<a href="#">x509_get_alg</a>	823
5.136.1.9	<a href="#">x509_get_dates</a>	823
5.136.1.10	<a href="#">x509_get_ext</a>	824
5.136.1.11	<a href="#">x509_get_iv</a>	824
5.136.1.12	<a href="#">x509_get_name</a>	824
5.136.1.13	<a href="#">x509_get_pubkey</a>	824
5.136.1.14	<a href="#">x509_get_serial</a>	825
5.136.1.15	<a href="#">x509_get_sig</a>	825
5.136.1.16	<a href="#">x509_get_uid</a>	825
5.136.1.17	<a href="#">x509_get_version</a>	825
5.136.1.18	<a href="#">x509_hash</a>	825
5.136.1.19	<a href="#">x509_self_test</a>	825
5.136.1.20	<a href="#">x509parse_cert_info</a>	826
5.136.1.21	<a href="#">x509parse_cert</a>	826
5.136.1.22	<a href="#">x509parse_certfile</a>	827
5.136.1.23	<a href="#">x509parse_dn_gets</a>	827
5.136.1.24	<a href="#">x509parse_expired</a>	827
5.136.1.25	<a href="#">x509parse_key</a>	827
5.136.1.26	<a href="#">x509parse_keyfile</a>	828
5.136.1.27	<a href="#">x509parse_verify</a>	828
5.137	<a href="#">/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/aes/aesencrypt2.c File Reference</a>	830
5.137.1	<a href="#">Define Documentation</a>	830
5.137.1.1	<a href="#">MODE_DECRYPT</a>	830
5.137.1.2	<a href="#">MODE_ENCRYPT</a>	830
5.137.1.3	<a href="#">USAGE</a>	830
5.137.2	<a href="#">Function Documentation</a>	831
5.137.2.1	<a href="#">main</a>	831
5.138	<a href="#">/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/hello.c File Reference</a>	832
5.138.1	<a href="#">Function Documentation</a>	832
5.138.1.1	<a href="#">main</a>	832
5.139	<a href="#">/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/md5sum.c File Reference</a>	833
5.139.1	<a href="#">Function Documentation</a>	833

5.139.1.1 main . . . . .	833
5.139.1.2 md5_check . . . . .	833
5.139.1.3 md5_print . . . . .	833
5.139.1.4 md5_wrapper . . . . .	833
5.140/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/sha1sum.c File Reference	834
5.140.1 Function Documentation . . . . .	834
5.140.1.1 main . . . . .	834
5.140.1.2 sha1_check . . . . .	834
5.140.1.3 sha1_print . . . . .	834
5.140.1.4 sha1_wrapper . . . . .	834
5.141/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/sha2sum.c File Reference	835
5.141.1 Function Documentation . . . . .	835
5.141.1.1 main . . . . .	835
5.141.1.2 sha2_check . . . . .	835
5.141.1.3 sha2_print . . . . .	835
5.141.1.4 sha2_wrapper . . . . .	835
5.142/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh_client.c File Reference	836
5.142.1 Define Documentation . . . . .	836
5.142.1.1 SERVER_NAME . . . . .	836
5.142.1.2 SERVER_PORT . . . . .	836
5.142.2 Function Documentation . . . . .	836
5.142.2.1 main . . . . .	836
5.143/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh_genprime.c File Reference	837
5.143.1 Define Documentation . . . . .	837
5.143.1.1 DH_P_SIZE . . . . .	837
5.143.1.2 GENERATOR . . . . .	837
5.143.2 Function Documentation . . . . .	837
5.143.2.1 main . . . . .	837
5.144/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh_server.c File Reference	838
5.144.1 Define Documentation . . . . .	838
5.144.1.1 PLAINTEXT . . . . .	838
5.144.1.2 SERVER_PORT . . . . .	838
5.144.2 Function Documentation . . . . .	838
5.144.2.1 main . . . . .	838

5.145/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/mpi_demo.c File Reference . . . . .	839
5.145.1 Function Documentation . . . . .	839
5.145.1.1 main . . . . .	839
5.146/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa_genkey.c File Reference . . . . .	840
5.146.1 Define Documentation . . . . .	840
5.146.1.1 EXPONENT . . . . .	840
5.146.1.2 KEY_SIZE . . . . .	840
5.146.2 Function Documentation . . . . .	840
5.146.2.1 main . . . . .	840
5.147/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa_sign.c File Reference . . . . .	841
5.147.1 Function Documentation . . . . .	841
5.147.1.1 main . . . . .	841
5.148/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa_verify.c File Reference . . . . .	842
5.148.1 Function Documentation . . . . .	842
5.148.1.1 main . . . . .	842
5.149/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl_client1.c File Reference . . . . .	843
5.149.1 Define Documentation . . . . .	843
5.149.1.1 DEBUG_LEVEL . . . . .	843
5.149.1.2 GET_REQUEST . . . . .	843
5.149.1.3 SERVER_NAME . . . . .	843
5.149.1.4 SERVER_PORT . . . . .	843
5.149.2 Function Documentation . . . . .	844
5.149.2.1 main . . . . .	844
5.149.2.2 my_debug . . . . .	844
5.150/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl_client2.c File Reference . . . . .	845
5.150.1 Define Documentation . . . . .	845
5.150.1.1 DEBUG_LEVEL . . . . .	845
5.150.1.2 GET_REQUEST . . . . .	845
5.150.1.3 SERVER_NAME . . . . .	845
5.150.1.4 SERVER_PORT . . . . .	845
5.150.2 Function Documentation . . . . .	846
5.150.2.1 main . . . . .	846
5.150.2.2 my_debug . . . . .	846
5.151/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl_server.c File Reference . . . . .	847



5.151.1 Define Documentation . . . . .	847
5.151.1.1 DEBUG_LEVEL . . . . .	847
5.151.1.2 HTTP_RESPONSE . . . . .	847
5.151.2 Function Documentation . . . . .	848
5.151.2.1 main . . . . .	848
5.151.2.2 my_debug . . . . .	848
5.151.2.3 my_get_session . . . . .	848
5.151.2.4 my_set_session . . . . .	848
5.151.3 Variable Documentation . . . . .	848
5.151.3.1 cur . . . . .	848
5.151.3.2 my_ciphers . . . . .	848
5.151.3.3 my_dhm_G . . . . .	849
5.151.3.4 my_dhm_P . . . . .	849
5.151.3.5 prv . . . . .	849
5.151.3.6 s_list_1st . . . . .	849
5.152/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/benchmark.c File Reference . . . . .	850
5.152.1 Define Documentation . . . . .	850
5.152.1.1 BUFSIZE . . . . .	850
5.152.2 Function Documentation . . . . .	850
5.152.2.1 main . . . . .	850
5.152.2.2 myrand . . . . .	851
5.152.3 Variable Documentation . . . . .	851
5.152.3.1 buf . . . . .	851
5.153/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/selftest.c File Reference . . . . .	852
5.153.1 Function Documentation . . . . .	852
5.153.1.1 main . . . . .	852
5.154/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/ssl_test.c File Reference . . . . .	853
5.154.1 Define Documentation . . . . .	854
5.154.1.1 COMMAND_BOTH . . . . .	854
5.154.1.2 COMMAND_READ . . . . .	854
5.154.1.3 COMMAND_WRITE . . . . .	854
5.154.1.4 DFL_BUFFER_SIZE . . . . .	854
5.154.1.5 DFL_COMMAND . . . . .	854
5.154.1.6 DFL_CONN_TIMEOUT . . . . .	854
5.154.1.7 DFL_DEBUG_LEVEL . . . . .	854
5.154.1.8 DFL_FORCE_CIPHER . . . . .	855

5.154.1.9 DFL_IOMODE . . . . .	855
5.154.1.10 DFL_MAX_BYTES . . . . .	855
5.154.1.11 DFL_MAX_CONNECTIONS . . . . .	855
5.154.1.12 DFL_OPMODE . . . . .	855
5.154.1.13 DFL_SERVER_NAME . . . . .	855
5.154.1.14 DFL_SERVER_PORT . . . . .	855
5.154.1.15 DFL_SESSION_LIFETIME . . . . .	855
5.154.1.16 DFL_SESSION_REUSE . . . . .	855
5.154.1.17 IOMODE_BLOCK . . . . .	856
5.154.1.18 IOMODE_NONBLOCK . . . . .	856
5.154.1.19 OPMODE_CLIENT . . . . .	856
5.154.1.20 OPMODE_NONE . . . . .	856
5.154.1.21 OPMODE_SERVER . . . . .	856
5.154.1.22 USAGE . . . . .	856
5.154.2 Function Documentation . . . . .	857
5.154.2.1 lcppm5 . . . . .	857
5.154.2.2 main . . . . .	857
5.154.2.3 my_debug . . . . .	857
5.154.2.4 ssl_test . . . . .	857
5.154.3 Variable Documentation . . . . .	857
5.154.3.1 dhm_G . . . . .	857
5.154.3.2 dhm_P . . . . .	858
5.154.3.3 server_fd . . . . .	858
5.155/home/dko/Projects/mobilec/src/util/mc_genkey.c File Reference . . . . .	859
5.155.1 Function Documentation . . . . .	859
5.155.1.1 main . . . . .	859
5.155.1.2 print_usage . . . . .	859
5.156/home/dko/Projects/mobilec/src/xml_compose.c File Reference . . . . .	860
5.156.1 Function Documentation . . . . .	860
5.156.1.1 agent_xml_compose . . . . .	860
5.156.1.2 agent_xml_compose__agent_code . . . . .	860
5.156.1.3 agent_xml_compose__agent_data . . . . .	861
5.156.1.4 agent_xml_compose__create_row_nodes . . . . .	861
5.156.1.5 agent_xml_compose__data . . . . .	861
5.156.1.6 agent_xml_compose__gaf_message . . . . .	861
5.156.1.7 agent_xml_compose__home . . . . .	861

5.156.1.8 agent_xml_compose__message . . . . .	861
5.156.1.9 agent_xml_compose__mobile_agent . . . . .	862
5.156.1.10 agent_xml_compose__name . . . . .	862
5.156.1.11 agent_xml_compose__owner . . . . .	862
5.156.1.12 agent_xml_compose__row . . . . .	862
5.156.1.13 agent_xml_compose__task . . . . .	862
5.156.1.14 agent_xml_compose__tasks . . . . .	863
5.156.1.15 agent_xml_compose__wg_code . . . . .	863
5.157/home/dko/Projects/mobilec/src/xml_helper.c File Reference . . . . .	864
5.157.1 Function Documentation . . . . .	864
5.157.1.1 whitespace_cb . . . . .	864
5.157.1.2 xml_find_sibling . . . . .	864
5.157.1.3 xml_get_cdata . . . . .	864
5.157.1.4 xml_get_child . . . . .	865
5.157.1.5 xml_get_deep_child . . . . .	865
5.157.1.6 xml_get_element_name . . . . .	865
5.157.1.7 xml_get_next_element . . . . .	865
5.157.1.8 xml_get_text . . . . .	865
5.157.1.9 xml_new_cdata . . . . .	865
5.158/home/dko/Projects/mobilec/src/xml_parser.c File Reference . . . . .	866
5.158.1 Define Documentation . . . . .	866
5.158.1.1 _XOPEN_SOURCE . . . . .	866
5.158.2 Function Documentation . . . . .	867
5.158.2.1 agent_return_xml_parse . . . . .	867
5.158.2.2 agent_xml_parse . . . . .	867
5.158.2.3 agent_xml_parse__agent_code . . . . .	867
5.158.2.4 agent_xml_parse__agent_data . . . . .	867
5.158.2.5 agent_xml_parse__data . . . . .	867
5.158.2.6 agent_xml_parse__fill_row_data . . . . .	868
5.158.2.7 agent_xml_parse__home . . . . .	868
5.158.2.8 agent_xml_parse__mobile_agent . . . . .	868
5.158.2.9 agent_xml_parse__name . . . . .	868
5.158.2.10 agent_xml_parse__owner . . . . .	868
5.158.2.11 agent_xml_parse__row . . . . .	868
5.158.2.12 agent_xml_parse__sender . . . . .	869
5.158.2.13 agent_xml_parse__task . . . . .	869

5.158.2.14	<a href="#">agent_xml_parse__tasks</a> . . . . .	869
5.158.2.15	<a href="#">agent_xml_parse__wg_code</a> . . . . .	869
5.158.2.16	<a href="#">message_xml_parse</a> . . . . .	869
5.158.2.17	<a href="#">message_xml_parse__message</a> . . . . .	869

# Chapter 1

# MobileC

## 1.1 Introduction

Welcome to the MobileC documentation. For a quick user-api reference, please refer to the file [libmc.h](#).

This documentation is provided supplementary to the main Mobile-C User Guide. The official user guide may be obtained at <http://www.mobilec.org> , and should also be included with your Mobile-C Distribution package in the file docs/mobilec.pdf .



## Chapter 2

# Data Structure Index

### 2.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">_hr_time</a>	9
<a href="#">_ssl_context</a>	10
<a href="#">_ssl_session</a>	22
<a href="#">_x509_buf</a>	24
<a href="#">_x509_cert</a>	25
<a href="#">_x509_name</a>	29
<a href="#">_x509_node</a>	30
<a href="#">_x509_raw</a>	31
<a href="#">_x509_time</a>	33
<a href="#">aes_context</a> (AES context structure )	35
<a href="#">agency_s</a> (The agency handle )	36
<a href="#">agent_datastate_s</a>	40
<a href="#">agent_mailbox_s</a>	43
<a href="#">agent_s</a>	44
<a href="#">agent_task_s</a>	49
<a href="#">AP_GENERIC_s</a>	52
<a href="#">arc4_context</a> (ARC4 context structure )	53
<a href="#">barrier_node_s</a>	54
<a href="#">barrier_queue_s</a>	55
<a href="#">cmd_prompt_s</a>	56
<a href="#">command_s</a>	57
<a href="#">connection_s</a>	58
<a href="#">fipa_expression_s::content_u</a>	60
<a href="#">des3_context</a> (Triple-DES context structure )	62
<a href="#">des_context</a> (DES context structure )	63
<a href="#">dhm_context</a>	64
<a href="#">dynstring_s</a>	66
<a href="#">fipa_acl_envelope_Received_s</a>	67
<a href="#">fipa_acl_envelope_s</a>	69
<a href="#">fipa_acl_message_s</a>	70
<a href="#">fipa_acl_Param_s</a>	73
<a href="#">fipa_agent_identifier_s</a>	75
<a href="#">fipa_agent_identifier_set_s</a>	76

fipa_comm_action_t	77
fipa_comm_message_check_t	78
fipa_comm_performative_t	79
fipa_comm_protocol_cn_t	80
fipa_comm_protocol_t	82
fipa_comm_reply_t	84
fipa_comm_t	85
fipa_DateTime_s	87
fipa_expression_s	89
fipa_list_t	90
fipa_message_string_s	91
fipa_number_s	92
fipa_string_s	93
fipa_url_s	94
fipa_url_sequence_s	95
fipa_word_s	96
foo_c	97
foo_s	100
havege_state (HAVEGE state structure)	102
host_id_s	104
hr_time (Timer structure)	105
interpreter_variable_data_s	106
list_s	108
listNode_s	109
mc_platform_s	110
mc_rwlock_s	117
MCAgencyOptions_s (User modifiable agency options)	119
md2_context (MD2 context structure)	121
md4_context (MD4 context structure)	123
md5_context (MD5 context structure)	125
message_s	127
message_send_arg_s	130
mpi (MPI structure)	131
mtp_http_content_s	133
mtp_http_s	134
mxml_attr_s	138
mxml_custom_s	139
mxml_fdbuf_s	140
mxml_index_s	141
mxml_node_s	143
mxml_text_s	145
mxml_value_s	146
mxml_value_u	147
options	149
rsa_context (RSA context structure)	152
sha1_context (SHA-1 context structure)	156
sha2_context (SHA-256 context structure)	158
sha4_context (SHA-512 context structure)	160
stationary_agent_info_s	162
syncList_s	163
syncListNode_s	164



## Chapter 3

# File Index

### 3.1 File List

Here is a list of all files with brief descriptions:

/home/dko/Projects/mobilec/src/acc.c	167
/home/dko/Projects/mobilec/src/agent.c	172
/home/dko/Projects/mobilec/src/agent_datastate.c	176
/home/dko/Projects/mobilec/src/agent_mailbox.c	177
/home/dko/Projects/mobilec/src/agent_return_data.c	179
/home/dko/Projects/mobilec/src/agent_task.c	181
/home/dko/Projects/mobilec/src/ams.c	182
/home/dko/Projects/mobilec/src/barrier.c	184
/home/dko/Projects/mobilec/src/cmd_prompt.c	186
/home/dko/Projects/mobilec/src/connection.c	189
/home/dko/Projects/mobilec/src/data_structures.c	190
/home/dko/Projects/mobilec/src/df.c	192
/home/dko/Projects/mobilec/src/dynstring.c	197
/home/dko/Projects/mobilec/src/fipa_acl.c	198
/home/dko/Projects/mobilec/src/fipa_envelope.c	211
/home/dko/Projects/mobilec/src/libmc.c	404
/home/dko/Projects/mobilec/src/mc_platform.c	467
/home/dko/Projects/mobilec/src/mc_rwlock.c	469
/home/dko/Projects/mobilec/src/message.c	477
/home/dko/Projects/mobilec/src/message_queue.c	481
/home/dko/Projects/mobilec/src/mtp_http.c	482
/home/dko/Projects/mobilec/src/winconfig.h	535
/home/dko/Projects/mobilec/src/xml_compose.c	860
/home/dko/Projects/mobilec/src/xml_helper.c	864
/home/dko/Projects/mobilec/src/xml_parser.c	866
/home/dko/Projects/mobilec/src/include/acc.h	219
/home/dko/Projects/mobilec/src/include/agent.h	222
/home/dko/Projects/mobilec/src/include/agent_datastate.h	226
/home/dko/Projects/mobilec/src/include/agent_lib.h	228
/home/dko/Projects/mobilec/src/include/agent_mailbox.h	240
/home/dko/Projects/mobilec/src/include/agent_task.h	242
/home/dko/Projects/mobilec/src/include/ams.h	244
/home/dko/Projects/mobilec/src/include/ap_queue_template.h	246

/home/dko/Projects/mobilec/src/include/barrier.h	249
/home/dko/Projects/mobilec/src/include/cmd_prompt.h	252
/home/dko/Projects/mobilec/src/include/commands.h	255
/home/dko/Projects/mobilec/src/include/commands.x.h	258
/home/dko/Projects/mobilec/src/include/connection.h	260
/home/dko/Projects/mobilec/src/include/data_structures.h	262
/home/dko/Projects/mobilec/src/include/df.h	266
/home/dko/Projects/mobilec/src/include/df_request.x.h	273
/home/dko/Projects/mobilec/src/include/dynstring.h	274
/home/dko/Projects/mobilec/src/include/fipa_acl.h	276
/home/dko/Projects/mobilec/src/include/fipa_acl_envelope.h	293
/home/dko/Projects/mobilec/src/include/fipa_comm.h	301
/home/dko/Projects/mobilec/src/include/host_id.h	309
/home/dko/Projects/mobilec/src/include/interpreter_variable_data.h	310
/home/dko/Projects/mobilec/src/include/libmc.h (MobileC api header file )	312
/home/dko/Projects/mobilec/src/include/macros.h	365
/home/dko/Projects/mobilec/src/include/mc_error.h	377
/home/dko/Projects/mobilec/src/include/mc_platform.h	379
/home/dko/Projects/mobilec/src/include/mc_rwlock.h	381
/home/dko/Projects/mobilec/src/include/message.h	383
/home/dko/Projects/mobilec/src/include/mobilec.h	387
/home/dko/Projects/mobilec/src/include/mtp_http.h	388
/home/dko/Projects/mobilec/src/include/xml_compose.h	394
/home/dko/Projects/mobilec/src/include/xml_helper.h	398
/home/dko/Projects/mobilec/src/include/xml_parser.h	400
/home/dko/Projects/mobilec/src/mc_list/list.c	462
/home/dko/Projects/mobilec/src/mc_list/list.h	464
/home/dko/Projects/mobilec/src/mc_sync/sync_list.c	471
/home/dko/Projects/mobilec/src/mc_sync/sync_list.h	474
/home/dko/Projects/mobilec/src/mxml-2.2.2/config.h	487
/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-attr.c	491
/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-entity.c	492
/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-file.c	494
/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-index.c	501
/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-node.c	503
/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-private.c	506
/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-search.c	508
/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-set.c	509
/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-string.c	511
/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml.h	512
/home/dko/Projects/mobilec/src/mxml-2.2.2/mxmldoc.c	524
/home/dko/Projects/mobilec/src/mxml-2.2.2/testmxml.c	532
/home/dko/Projects/mobilec/src/mxml-2.2.2/winconfig.h	533
/home/dko/Projects/mobilec/src/mxml-2.2.2/test/class.cxx	528
/home/dko/Projects/mobilec/src/mxml-2.2.2/test/enum.cxx	529
/home/dko/Projects/mobilec/src/mxml-2.2.2/test/function.cxx	530
/home/dko/Projects/mobilec/src/mxml-2.2.2/test/struct.cxx	531
/home/dko/Projects/mobilec/src/security/asm.c	536
/home/dko/Projects/mobilec/src/security/asm.h	537
/home/dko/Projects/mobilec/src/security/asm_message_composer.c	538
/home/dko/Projects/mobilec/src/security/asm_message_composer.h	539
/home/dko/Projects/mobilec/src/security/asm_message_parser.c	540
/home/dko/Projects/mobilec/src/security/asm_message_parser.h	541
/home/dko/Projects/mobilec/src/security/asm_node.c	542

/home/dko/Projects/mobilec/src/security/asm_node.h . . . . .	543
/home/dko/Projects/mobilec/src/security/interface.c . . . . .	544
/home/dko/Projects/mobilec/src/security/interface.h . . . . .	548
/home/dko/Projects/mobilec/src/security/mc_dh.c . . . . .	553
/home/dko/Projects/mobilec/src/security/mc_dh.h . . . . .	554
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/aes.h . . . . .	555
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/arc4.h . . . . .	559
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/base64.h . . . . .	561
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/bignum.h . . . . .	563
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/bn_mul.h . . . . .	579
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/certs.h . . . . .	581
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/config.h . . . . .	488
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/debug.h . . . . .	583
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/des.h . . . . .	586
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/dhm.h . . . . .	591
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/havege.h . . . . .	596
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md2.h . . . . .	598
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md4.h . . . . .	602
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md5.h . . . . .	606
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/net.h . . . . .	611
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/openssl.h . . . . .	616
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/padlock.h . . . . .	620
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/rsa.h . . . . .	621
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha1.h . . . . .	631
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha2.h . . . . .	636
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha4.h . . . . .	641
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/ssl.h . . . . .	646
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/timing.h . . . . .	670
/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/x509.h . . . . .	672
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/aes.c . . . . .	687
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/arc4.c . . . . .	696
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/base64.c . . . . .	699
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/bignum.c . . . . .	703
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/certs.c . . . . .	720
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/debug.c . . . . .	724
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/des.c . . . . .	726
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/dhm.c . . . . .	739
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/havege.c . . . . .	743
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md2.c . . . . .	745
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md4.c . . . . .	746
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md5.c . . . . .	747
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/net.c . . . . .	756
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/padlock.c . . . . .	760
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/rsa.c . . . . .	761
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha1.c . . . . .	771
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha2.c . . . . .	780
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha4.c . . . . .	789
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl_cli.c . . . . .	798
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl_srv.c . . . . .	801
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl_tls.c . . . . .	804
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/timing.c . . . . .	818
/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/x509parse.c . . . . .	821
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/aes/aesencrypt2.c . . . . .	830
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/hello.c . . . . .	832

/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/md5sum.c . . . . .	833
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/sha1sum.c . . . . .	834
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/sha2sum.c . . . . .	835
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh_client.c . . . . .	836
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh_genprime.c . . . . .	837
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh_server.c . . . . .	838
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/mpi_demo.c . . . . .	839
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa_genkey.c . . . . .	840
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa_sign.c . . . . .	841
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa_verify.c . . . . .	842
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl_client1.c . . . . .	843
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl_client2.c . . . . .	845
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl_server.c . . . . .	847
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/benchmark.c . . . . .	850
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/selftest.c . . . . .	852
/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/ssl_test.c . . . . .	853
/home/dko/Projects/mobilec/src/util/mc_genkey.c . . . . .	859

## Chapter 4

# Data Structure Documentation

### 4.1 `_hr_time` Struct Reference

#### Data Fields

- struct timeval [start](#)

#### 4.1.1 Detailed Description

Definition at line 45 of file `timing.c`.

#### 4.1.2 Field Documentation

##### 4.1.2.1 `struct timeval _hr_time::start`

Definition at line 47 of file `timing.c`.

Referenced by `get_timer()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/security/xyssl-0.9/library/timing.c`

## 4.2 `_ssl_context` Struct Reference

```
#include <ssl.h>
```

### Data Fields

- `int state`
- `int major_ver`
- `int minor_ver`
- `int max_major_ver`
- `int max_minor_ver`
- `int(* f_rng)(void *)`
- `void(* f_dbg)(void *, int, char *)`
- `int(* f_recv)(void *, unsigned char *, int)`
- `int(* f_send)(void *, unsigned char *, int)`
- `void * p_rng`
- `void * p_dbg`
- `void * p_recv`
- `void * p_send`
- `int resume`
- `int timeout`
- `ssl_session * session`
- `int(* s_get)(ssl_context *)`
- `int(* s_set)(ssl_context *)`
- `unsigned char * in_ctr`
- `unsigned char * in_hdr`
- `unsigned char * in_msg`
- `unsigned char * in_offt`
- `int in_msgtype`
- `int in_msglen`
- `int in_left`
- `int in_hrlen`
- `int nb_zero`
- `unsigned char * out_ctr`
- `unsigned char * out_hdr`
- `unsigned char * out_msg`
- `int out_msgtype`
- `int out_msglen`
- `int out_left`
- `rsa_context * rsa_key`
- `x509_cert * own_cert`
- `x509_cert * ca_chain`
- `x509_cert * peer_cert`
- `char * peer_cn`
- `int endpoint`
- `int authmode`
- `int client_auth`
- `int verify_result`
- `dhm_context dhm_ctx`

- `md5_context` `fin_md5`
- `sha1_context` `fin_sha1`
- `int` `do_crypt`
- `int *` `ciphers`
- `int` `pmslen`
- `int` `keylen`
- `int` `minlen`
- `int` `ivlen`
- `int` `maclen`
- `unsigned char` `randbytes` [64]
- `unsigned char` `premaster` [256]
- `unsigned char` `iv_enc` [16]
- `unsigned char` `iv_dec` [16]
- `unsigned char` `mac_enc` [32]
- `unsigned char` `mac_dec` [32]
- `unsigned long` `ctx_enc` [128]
- `unsigned long` `ctx_dec` [128]
- `unsigned char *` `hostname`
- `unsigned long` `hostname_len`

### 4.2.1 Detailed Description

Definition at line 149 of file `ssl.h`.

### 4.2.2 Field Documentation

#### 4.2.2.1 `int _ssl_context::authmode`

verification mode

Definition at line 220 of file `ssl.h`.

Referenced by `ssl_parse_certificate()`, `ssl_set_authmode()`, and `ssl_write_certificate_request()`.

#### 4.2.2.2 `x509_cert* _ssl_context::ca_chain`

own trusted CA chain

Definition at line 215 of file `ssl.h`.

Referenced by `ssl_parse_certificate()`, `ssl_set_ca_chain()`, and `ssl_write_certificate_request()`.

#### 4.2.2.3 `int* _ssl_context::ciphers`

allowed ciphersuites

Definition at line 232 of file `ssl.h`.

Referenced by `ssl_parse_client_hello()`, `ssl_parse_server_hello()`, `ssl_set_ciphers()`, and `ssl_write_client_hello()`.

#### 4.2.2.4 `int _ssl_context::client_auth`

flag for client auth.

Definition at line 221 of file `ssl.h`.

Referenced by `ssl_parse_certificate_request()`, `ssl_parse_server_hello_done()`, `ssl_write_certificate()`, and `ssl_write_certificate_verify()`.

#### 4.2.2.5 `unsigned long _ssl_context::ctx_dec[128]`

decryption context

Definition at line 249 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`, and `ssl_derive_keys()`.

#### 4.2.2.6 `unsigned long _ssl_context::ctx_enc[128]`

encryption context

Definition at line 248 of file `ssl.h`.

Referenced by `ssl_derive_keys()`, and `ssl_encrypt_buf()`.

#### 4.2.2.7 `dhm_context _ssl_context::dhm_ctx`

DHM key exchange

Definition at line 227 of file `ssl.h`.

Referenced by `ssl_free()`, `ssl_parse_client_key_exchange()`, `ssl_parse_server_key_exchange()`, `ssl_set_dh_param()`, `ssl_write_client_key_exchange()`, and `ssl_write_server_key_exchange()`.

#### 4.2.2.8 `int _ssl_context::do_crypt`

en(de)cryption flag

Definition at line 231 of file `ssl.h`.

Referenced by `ssl_parse_change_cipher_spec()`, `ssl_parse_finished()`, `ssl_read_record()`, `ssl_write_change_cipher_spec()`, `ssl_write_finished()`, and `ssl_write_record()`.

#### 4.2.2.9 `int _ssl_context::endpoint`

0: client, 1: server

Definition at line 219 of file `ssl.h`.

Referenced by `ssl_derive_keys()`, `ssl_handshake()`, `ssl_parse_certificate()`, `ssl_parse_finished()`, `ssl_set_endpoint()`, `ssl_write_certificate()`, and `ssl_write_finished()`.

#### 4.2.2.10 `void(* _ssl_context::f_dbg)(void *, int, char *)`

Definition at line 166 of file `ssl.h`.



Referenced by debug\_print\_buf(), debug\_print\_crt(), debug\_print\_mpi(), debug\_print\_msg(), debug\_print\_ret(), and ssl\_set\_dbg().

#### 4.2.2.11 int(\* \_ssl\_context::f\_recv)(void \*, unsigned char \*, int)

Definition at line 167 of file ssl.h.

Referenced by ssl\_fetch\_input(), and ssl\_set\_bio().

#### 4.2.2.12 int(\* \_ssl\_context::f\_rng)(void \*)

Definition at line 165 of file ssl.h.

Referenced by ssl\_parse\_client\_key\_exchange(), ssl\_set\_rng(), ssl\_write\_client\_hello(), ssl\_write\_client\_key\_exchange(), ssl\_write\_server\_hello(), and ssl\_write\_server\_key\_exchange().

#### 4.2.2.13 int(\* \_ssl\_context::f\_send)(void \*, unsigned char \*, int)

Definition at line 168 of file ssl.h.

Referenced by ssl\_flush\_output(), and ssl\_set\_bio().

#### 4.2.2.14 md5\_context \_ssl\_context::fin\_md5

Finished MD5 checksum

Definition at line 228 of file ssl.h.

Referenced by ssl\_calc\_verify(), ssl\_init(), ssl\_parse\_client\_hello(), ssl\_parse\_finished(), ssl\_read\_record(), ssl\_write\_finished(), and ssl\_write\_record().

#### 4.2.2.15 sha1\_context \_ssl\_context::fin\_sha1

Finished SHA-1 checksum

Definition at line 229 of file ssl.h.

Referenced by ssl\_calc\_verify(), ssl\_init(), ssl\_parse\_client\_hello(), ssl\_parse\_finished(), ssl\_read\_record(), ssl\_write\_finished(), and ssl\_write\_record().

#### 4.2.2.16 unsigned char\* \_ssl\_context::hostname

Definition at line 254 of file ssl.h.

Referenced by ssl\_free(), ssl\_init(), ssl\_set\_hostname(), and ssl\_write\_client\_hello().

#### 4.2.2.17 unsigned long \_ssl\_context::hostname\_len

Definition at line 255 of file ssl.h.

Referenced by ssl\_free(), ssl\_init(), ssl\_set\_hostname(), and ssl\_write\_client\_hello().

**4.2.2.18 unsigned char\* \_ssl\_context::in\_ctr**

64-bit incoming message counter

Definition at line 187 of file ssl.h.

Referenced by ssl\_decrypt\_buf(), ssl\_free(), and ssl\_init().

**4.2.2.19 unsigned char\* \_ssl\_context::in\_hdr**

5-byte record header (in\_ctr+8)

Definition at line 188 of file ssl.h.

Referenced by ssl\_decrypt\_buf(), ssl\_fetch\_input(), ssl\_init(), ssl\_parse\_client\_hello(), and ssl\_read\_record().

**4.2.2.20 int \_ssl\_context::in\_hhlen**

current handshake message length

Definition at line 196 of file ssl.h.

Referenced by ssl\_parse\_certificate(), ssl\_parse\_certificate\_verify(), ssl\_parse\_client\_key\_exchange(), ssl\_parse\_finished(), ssl\_parse\_server\_hello(), ssl\_parse\_server\_hello\_done(), ssl\_parse\_server\_key\_exchange(), and ssl\_read\_record().

**4.2.2.21 int \_ssl\_context::in\_left**

amount of data read so far

Definition at line 194 of file ssl.h.

Referenced by ssl\_fetch\_input(), ssl\_parse\_client\_hello(), and ssl\_read\_record().

**4.2.2.22 unsigned char\* \_ssl\_context::in\_msg**

the message contents (in\_hdr+5)

Definition at line 189 of file ssl.h.

Referenced by ssl\_decrypt\_buf(), ssl\_init(), ssl\_parse\_certificate(), ssl\_parse\_certificate\_request(), ssl\_parse\_certificate\_verify(), ssl\_parse\_change\_cipher\_spec(), ssl\_parse\_client\_hello(), ssl\_parse\_client\_key\_exchange(), ssl\_parse\_finished(), ssl\_parse\_server\_hello(), ssl\_parse\_server\_hello\_done(), ssl\_parse\_server\_key\_exchange(), ssl\_read(), and ssl\_read\_record().

**4.2.2.23 int \_ssl\_context::in\_msglen**

record header: message length

Definition at line 193 of file ssl.h.

Referenced by ssl\_decrypt\_buf(), ssl\_get\_bytes\_avail(), ssl\_parse\_certificate(), ssl\_parse\_change\_cipher\_spec(), ssl\_read(), and ssl\_read\_record().

**4.2.2.24 `int _ssl_context::in_msgtype`**

record header: message type

Definition at line 192 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`, `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `ssl_read()`, and `ssl_read_record()`.

**4.2.2.25 `unsigned char* _ssl_context::in_offt`**

read offset in application data

Definition at line 190 of file `ssl.h`.

Referenced by `ssl_get_bytes_avail()`, and `ssl_read()`.

**4.2.2.26 `unsigned char _ssl_context::iv_dec[16]`**

IV (decryption)

Definition at line 243 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`, and `ssl_derive_keys()`.

**4.2.2.27 `unsigned char _ssl_context::iv_enc[16]`**

IV (encryption)

Definition at line 242 of file `ssl.h`.

Referenced by `ssl_derive_keys()`, and `ssl_encrypt_buf()`.

**4.2.2.28 `int _ssl_context::ivlen`**

IV length

Definition at line 236 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`, `ssl_derive_keys()`, and `ssl_encrypt_buf()`.

**4.2.2.29 `int _ssl_context::keylen`**

symmetric key length

Definition at line 234 of file `ssl.h`.

Referenced by `ssl_derive_keys()`.

**4.2.2.30 `unsigned char _ssl_context::mac_dec[32]`**

MAC (decryption)

Definition at line 246 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`, and `ssl_derive_keys()`.

#### **4.2.2.31 unsigned char \_ssl\_context::mac\_enc[32]**

MAC (encryption)

Definition at line 245 of file `ssl.h`.

Referenced by `ssl_derive_keys()`, and `ssl_encrypt_buf()`.

#### **4.2.2.32 int \_ssl\_context::maclen**

MAC length

Definition at line 237 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`, `ssl_derive_keys()`, and `ssl_encrypt_buf()`.

#### **4.2.2.33 int \_ssl\_context::major\_ver**

equal to `SSL_MAJOR_VERSION_3`

Definition at line 156 of file `ssl.h`.

Referenced by `ssl_parse_client_hello()`, `ssl_read_record()`, `ssl_write_client_hello()`, `ssl_write_record()`, and `ssl_write_server_hello()`.

#### **4.2.2.34 int \_ssl\_context::max\_major\_ver**

max. major version from client

Definition at line 159 of file `ssl.h`.

Referenced by `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_write_client_hello()`, and `ssl_write_client_key_exchange()`.

#### **4.2.2.35 int \_ssl\_context::max\_minor\_ver**

max. minor version from client

Definition at line 160 of file `ssl.h`.

Referenced by `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_write_client_hello()`, and `ssl_write_client_key_exchange()`.

#### **4.2.2.36 int \_ssl\_context::minlen**

min. ciphertext length

Definition at line 235 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`, `ssl_derive_keys()`, and `ssl_read_record()`.

**4.2.2.37 int `_ssl_context::minor_ver`**

either 0 (SSL3) or 1 (TLS1.0)

Definition at line 157 of file `ssl.h`.

Referenced by `ssl_calc_finished()`, `ssl_calc_verify()`, `ssl_decrypt_buf()`, `ssl_derive_keys()`, `ssl_encrypt_buf()`, `ssl_parse_certificate()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_read_record()`, `ssl_write_certificate()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, and `ssl_write_server_hello()`.

**4.2.2.38 int `_ssl_context::nb_zero`**

# of 0-length encrypted messages

Definition at line 197 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`.

**4.2.2.39 unsigned char\* `_ssl_context::out_ctr`**

64-bit outgoing message counter

Definition at line 202 of file `ssl.h`.

Referenced by `ssl_encrypt_buf()`, `ssl_free()`, and `ssl_init()`.

**4.2.2.40 unsigned char\* `_ssl_context::out_hdr`**

5-byte record header (`out_ctr+8`)

Definition at line 203 of file `ssl.h`.

Referenced by `ssl_flush_output()`, `ssl_init()`, and `ssl_write_record()`.

**4.2.2.41 int `_ssl_context::out_left`**

amount of data not yet written

Definition at line 208 of file `ssl.h`.

Referenced by `ssl_flush_output()`, `ssl_write()`, and `ssl_write_record()`.

**4.2.2.42 unsigned char\* `_ssl_context::out_msg`**

the message contents (`out_hdr+5`)

Definition at line 204 of file `ssl.h`.

Referenced by `ssl_close_notify()`, `ssl_encrypt_buf()`, `ssl_init()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

#### 4.2.2.43 `int _ssl_context::out_msglen`

record header: message length

Definition at line 207 of file `ssl.h`.

Referenced by `ssl_close_notify()`, `ssl_encrypt_buf()`, `ssl_flush_output()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

#### 4.2.2.44 `int _ssl_context::out_msgtype`

record header: message type

Definition at line 206 of file `ssl.h`.

Referenced by `ssl_close_notify()`, `ssl_encrypt_buf()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

#### 4.2.2.45 `x509_cert* _ssl_context::own_cert`

own X.509 certificate

Definition at line 214 of file `ssl.h`.

Referenced by `ssl_set_own_cert()`, and `ssl_write_certificate()`.

#### 4.2.2.46 `void* _ssl_context::p_dbg`

context for the debug function

Definition at line 171 of file `ssl.h`.

Referenced by `debug_print_buf()`, `debug_print_crt()`, `debug_print_mpi()`, `debug_print_msg()`, `debug_print_ret()`, and `ssl_set_dbg()`.

#### 4.2.2.47 `void* _ssl_context::p_recv`

context for reading operations

Definition at line 172 of file `ssl.h`.

Referenced by `ssl_fetch_input()`, and `ssl_set_bio()`.

#### 4.2.2.48 `void* _ssl_context::p_rng`

context for the RNG function

Definition at line 170 of file `ssl.h`.

Referenced by `ssl_parse_client_key_exchange()`, `ssl_set_rng()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_server_hello()`, and `ssl_write_server_key_exchange()`.

**4.2.2.49 void\* `_ssl_context::p_send`**

context for writing operations

Definition at line 173 of file `ssl.h`.

Referenced by `ssl_flush_output()`, and `ssl_set_bio()`.

**4.2.2.50 x509\_cert\* `_ssl_context::peer_cert`**

peer X.509 cert chain

Definition at line 216 of file `ssl.h`.

Referenced by `main()`, `ssl_free()`, `ssl_parse_certificate()`, `ssl_parse_certificate_verify()`, `ssl_parse_server_key_exchange()`, and `ssl_write_client_key_exchange()`.

**4.2.2.51 char\* `_ssl_context::peer_cn`**

expected peer CN

Definition at line 217 of file `ssl.h`.

Referenced by `ssl_parse_certificate()`, and `ssl_set_ca_chain()`.

**4.2.2.52 int `_ssl_context::pmslen`**

premaster length

Definition at line 233 of file `ssl.h`.

Referenced by `ssl_derive_keys()`, `ssl_parse_client_key_exchange()`, and `ssl_write_client_key_exchange()`.

**4.2.2.53 unsigned char `_ssl_context::premaster[256]`**

premaster secret

Definition at line 240 of file `ssl.h`.

Referenced by `ssl_derive_keys()`, `ssl_parse_client_key_exchange()`, and `ssl_write_client_key_exchange()`.

**4.2.2.54 unsigned char `_ssl_context::randbytes[64]`**

random bytes

Definition at line 239 of file `ssl.h`.

Referenced by `ssl_derive_keys()`, `ssl_parse_client_hello()`, `ssl_parse_server_hello()`, `ssl_parse_server_key_exchange()`, `ssl_write_client_hello()`, `ssl_write_server_hello()`, and `ssl_write_server_key_exchange()`.

**4.2.2.55 int `_ssl_context::resume`**

session resuming flag

Definition at line 178 of file `ssl.h`.

Referenced by `my_get_session()`, `ssl_derive_keys()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_set_session()`, `ssl_write_client_hello()`, `ssl_write_finished()`, and `ssl_write_server_hello()`.

#### 4.2.2.56 `rsa_context* _ssl_context::rsa_key`

own RSA private key

Definition at line 213 of file `ssl.h`.

Referenced by `ssl_parse_client_key_exchange()`, `ssl_set_own_cert()`, `ssl_write_certificate_verify()`, and `ssl_write_server_key_exchange()`.

#### 4.2.2.57 `int(* _ssl_context::s_get)(ssl_context *)`

(server) get callback

Definition at line 181 of file `ssl.h`.

Referenced by `ssl_set_scb()`, and `ssl_write_server_hello()`.

#### 4.2.2.58 `int(* _ssl_context::s_set)(ssl_context *)`

(server) set callback

Definition at line 182 of file `ssl.h`.

Referenced by `ssl_parse_client_key_exchange()`, and `ssl_set_scb()`.

#### 4.2.2.59 `ssl_session* _ssl_context::session`

current session data

Definition at line 180 of file `ssl.h`.

Referenced by `my_get_session()`, `my_set_session()`, `ssl_calc_finished()`, `ssl_calc_verify()`, `ssl_derive_keys()`, `ssl_get_cipher()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_server_hello()`, `ssl_parse_server_key_exchange()`, `ssl_set_session()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_server_hello()`, and `ssl_write_server_key_exchange()`.

#### 4.2.2.60 `int _ssl_context::state`

SSL handshake: current state

Definition at line 154 of file `ssl.h`.

Referenced by `ssl_close_notify()`, `ssl_handshake_client()`, `ssl_handshake_server()`, `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `ssl_read()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.



**4.2.2.61 `int _ssl_context::timeout`**

sess. expiration time

Definition at line 179 of file `ssl.h`.

Referenced by `my_get_session()`, `my_set_session()`, `ssl_set_session()`, and `ssl_write_client_hello()`.

**4.2.2.62 `int _ssl_context::verify_result`**

verification result

Definition at line 222 of file `ssl.h`.

Referenced by `ssl_get_verify_result()`, and `ssl_parse_certificate()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/ssl.h`

## 4.3 `_ssl_session` Struct Reference

```
#include <ssl.h>
```

### Data Fields

- `time_t start`
- `int cipher`
- `int length`
- unsigned char `id` [32]
- unsigned char `master` [48]
- `ssl_session * next`

### 4.3.1 Detailed Description

Definition at line 139 of file `ssl.h`.

### 4.3.2 Field Documentation

#### 4.3.2.1 `int _ssl_session::cipher`

chosen cipher

Definition at line 142 of file `ssl.h`.

Referenced by `my_get_session()`, `ssl_derive_keys()`, `ssl_get_cipher()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_server_hello()`, `ssl_parse_server_key_exchange()`, `ssl_write_client_key_exchange()`, `ssl_write_server_hello()`, and `ssl_write_server_key_exchange()`.

#### 4.3.2.2 `unsigned char _ssl_session::id[32]`

session identifier

Definition at line 144 of file `ssl.h`.

Referenced by `my_get_session()`, `my_set_session()`, `ssl_parse_client_hello()`, `ssl_parse_server_hello()`, `ssl_write_client_hello()`, and `ssl_write_server_hello()`.

#### 4.3.2.3 `int _ssl_session::length`

session id length

Definition at line 143 of file `ssl.h`.

Referenced by `my_get_session()`, `my_set_session()`, `ssl_parse_client_hello()`, `ssl_parse_server_hello()`, `ssl_write_client_hello()`, and `ssl_write_server_hello()`.

#### 4.3.2.4 `unsigned char _ssl_session::master[48]`

the master secret

Definition at line 145 of file `ssl.h`.

Referenced by `my_get_session()`, `ssl_calc_finished()`, `ssl_calc_verify()`, and `ssl_derive_keys()`.

#### 4.3.2.5 `ssl_session* _ssl_session::next`

next session entry

Definition at line 146 of file `ssl.h`.

Referenced by `main()`, `my_get_session()`, and `my_set_session()`.

#### 4.3.2.6 `time_t _ssl_session::start`

starting time

Definition at line 141 of file `ssl.h`.

Referenced by `my_get_session()`, `my_set_session()`, `ssl_parse_server_hello()`, and `ssl_write_client_hello()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/ssl.h`

## 4.4 `_x509_buf` Struct Reference

```
#include <x509.h>
```

### Data Fields

- `int tag`
- `int len`
- `unsigned char * p`

### 4.4.1 Detailed Description

Definition at line 96 of file `x509.h`.

### 4.4.2 Field Documentation

#### 4.4.2.1 `int _x509_buf::len`

Definition at line 99 of file `x509.h`.

Referenced by `ssl_write_certificate()`, `ssl_write_certificate_request()`, `x509_free()`, `x509_get_alg()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_serial()`, `x509_get_sig()`, `x509_get_uid()`, `x509parse_cert_info()`, `x509parse_crt()`, `x509parse_dn_gets()`, and `x509parse_verify()`.

#### 4.4.2.2 `unsigned char* _x509_buf::p`

Definition at line 100 of file `x509.h`.

Referenced by `ssl_write_certificate()`, `ssl_write_certificate_request()`, `x509_free()`, `x509_get_alg()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_serial()`, `x509_get_sig()`, `x509_get_uid()`, `x509parse_cert_info()`, `x509parse_crt()`, `x509parse_dn_gets()`, and `x509parse_verify()`.

#### 4.4.2.3 `int _x509_buf::tag`

Definition at line 98 of file `x509.h`.

Referenced by `x509_get_alg()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_serial()`, `x509_get_sig()`, and `x509_get_uid()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/x509.h`

## 4.5 `_x509_cert` Struct Reference

```
#include <x509.h>
```

### Data Fields

- `x509_buf` raw
- `x509_buf` tbs
- `int` version
- `x509_buf` serial
- `x509_buf` sig\_oid1
- `x509_buf` issuer\_raw
- `x509_buf` subject\_raw
- `x509_name` issuer
- `x509_name` subject
- `x509_time` valid\_from
- `x509_time` valid\_to
- `x509_buf` pk\_oid
- `rsa_context` rsa
- `x509_buf` issuer\_id
- `x509_buf` subject\_id
- `x509_buf` v3\_ext
- `int` ca\_istrue
- `int` max\_pathlen
- `x509_buf` sig\_oid2
- `x509_buf` sig
- `struct _x509_cert` \* next

### 4.5.1 Detailed Description

Definition at line 119 of file `x509.h`.

### 4.5.2 Field Documentation

#### 4.5.2.1 `int _x509_cert::ca_istrue`

Definition at line 144 of file `x509.h`.

Referenced by `x509parse_cert()`, and `x509parse_verify()`.

#### 4.5.2.2 `x509_name _x509_cert::issuer`

Definition at line 131 of file `x509.h`.

Referenced by `x509_free()`, `x509parse_cert_info()`, and `x509parse_cert()`.

#### 4.5.2.3 `x509_buf _x509_cert::issuer_id`

Definition at line 140 of file `x509.h`.

Referenced by `x509parse_cert()`.

**4.5.2.4 x509\_buf\_x509\_cert::issuer\_raw**

Definition at line 128 of file x509.h.

Referenced by x509parse\_cert(), and x509parse\_verify().

**4.5.2.5 int\_x509\_cert::max\_pathlen**

Definition at line 145 of file x509.h.

Referenced by x509parse\_cert(), and x509parse\_verify().

**4.5.2.6 struct\_x509\_cert\*\_x509\_cert::next**

Definition at line 150 of file x509.h.

Referenced by debug\_print\_cert(), main(), ssl\_test(), ssl\_write\_certificate(), ssl\_write\_certificate\_request(), x509\_free(), x509parse\_cert(), and x509parse\_verify().

**4.5.2.7 x509\_buf\_x509\_cert::pk\_oid**

Definition at line 137 of file x509.h.

Referenced by x509parse\_cert().

**4.5.2.8 x509\_buf\_x509\_cert::raw**

Definition at line 121 of file x509.h.

Referenced by ssl\_write\_certificate(), x509\_free(), and x509parse\_cert().

**4.5.2.9 rsa\_context\_x509\_cert::rsa**

Definition at line 138 of file x509.h.

Referenced by debug\_print\_cert(), ssl\_parse\_certificate\_verify(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_client\_key\_exchange(), x509\_free(), x509parse\_cert\_info(), x509parse\_cert(), and x509parse\_verify().

**4.5.2.10 x509\_buf\_x509\_cert::serial**

Definition at line 125 of file x509.h.

Referenced by x509parse\_cert\_info(), and x509parse\_cert().

**4.5.2.11 x509\_buf\_x509\_cert::sig**

Definition at line 148 of file x509.h.

Referenced by x509parse\_cert(), and x509parse\_verify().

**4.5.2.12 x509\_buf\_x509\_cert::sig\_oid1**

Definition at line 126 of file x509.h.

Referenced by x509parse\_cert\_info(), x509parse\_crt(), and x509parse\_verify().

**4.5.2.13 x509\_buf\_x509\_cert::sig\_oid2**

Definition at line 147 of file x509.h.

Referenced by x509parse\_crt().

**4.5.2.14 x509\_name\_x509\_cert::subject**

Definition at line 132 of file x509.h.

Referenced by x509\_free(), x509parse\_cert\_info(), x509parse\_crt(), and x509parse\_verify().

**4.5.2.15 x509\_buf\_x509\_cert::subject\_id**

Definition at line 141 of file x509.h.

Referenced by x509parse\_crt().

**4.5.2.16 x509\_buf\_x509\_cert::subject\_raw**

Definition at line 129 of file x509.h.

Referenced by ssl\_write\_certificate\_request(), x509parse\_crt(), and x509parse\_verify().

**4.5.2.17 x509\_buf\_x509\_cert::tbs**

Definition at line 122 of file x509.h.

Referenced by x509parse\_crt(), and x509parse\_verify().

**4.5.2.18 x509\_buf\_x509\_cert::v3\_ext**

Definition at line 142 of file x509.h.

Referenced by x509parse\_crt().

**4.5.2.19 x509\_time\_x509\_cert::valid\_from**

Definition at line 134 of file x509.h.

Referenced by x509parse\_cert\_info(), and x509parse\_crt().

**4.5.2.20 x509\_time\_x509\_cert::valid\_to**

Definition at line 135 of file x509.h.

Referenced by x509parse\_cert\_info(), x509parse\_crt(), and x509parse\_expired().

#### 4.5.2.21 int \_x509\_cert::version

Definition at line 124 of file x509.h.

Referenced by x509parse\_cert\_info(), x509parse\_crt(), and x509parse\_verify().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/x509.h](#)



## 4.6 `_x509_name` Struct Reference

```
#include <x509.h>
```

### Data Fields

- [x509\\_buf oid](#)
- [x509\\_buf val](#)
- [struct `\_x509\_name` \\* next](#)

### 4.6.1 Detailed Description

Definition at line 104 of file `x509.h`.

### 4.6.2 Field Documentation

#### 4.6.2.1 `struct _x509_name* _x509_name::next`

Definition at line 108 of file `x509.h`.

Referenced by `x509_free()`, `x509_get_name()`, `x509parse_dn_gets()`, and `x509parse_verify()`.

#### 4.6.2.2 `x509_buf _x509_name::oid`

Definition at line 106 of file `x509.h`.

Referenced by `x509_get_name()`, `x509parse_dn_gets()`, and `x509parse_verify()`.

#### 4.6.2.3 `x509_buf _x509_name::val`

Definition at line 107 of file `x509.h`.

Referenced by `x509_get_name()`, `x509parse_dn_gets()`, and `x509parse_verify()`.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/x509.h](#)

## 4.7 `_x509_node` Struct Reference

```
#include <x509.h>
```

### Data Fields

- unsigned char \* [data](#)
- unsigned char \* [p](#)
- unsigned char \* [end](#)
- size\_t [len](#)

#### 4.7.1 Detailed Description

Definition at line 157 of file x509.h.

#### 4.7.2 Field Documentation

##### 4.7.2.1 unsigned char\* `_x509_node::data`

Definition at line 159 of file x509.h.

##### 4.7.2.2 unsigned char\* `_x509_node::end`

Definition at line 161 of file x509.h.

##### 4.7.2.3 size\_t `_x509_node::len`

Definition at line 163 of file x509.h.

##### 4.7.2.4 unsigned char\* `_x509_node::p`

Definition at line 160 of file x509.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/x509.h](#)

## 4.8 `_x509_raw` Struct Reference

```
#include <x509.h>
```

### Data Fields

- [x509\\_node raw](#)
- [x509\\_node tbs](#)
- [x509\\_node version](#)
- [x509\\_node serial](#)
- [x509\\_node tbs\\_signalg](#)
- [x509\\_node issuer](#)
- [x509\\_node validity](#)
- [x509\\_node subject](#)
- [x509\\_node subpubkey](#)
- [x509\\_node signalg](#)
- [x509\\_node sign](#)

### 4.8.1 Detailed Description

Definition at line 167 of file `x509.h`.

### 4.8.2 Field Documentation

#### 4.8.2.1 `x509_node _x509_raw::issuer`

Definition at line 175 of file `x509.h`.

#### 4.8.2.2 `x509_node _x509_raw::raw`

Definition at line 169 of file `x509.h`.

#### 4.8.2.3 `x509_node _x509_raw::serial`

Definition at line 173 of file `x509.h`.

#### 4.8.2.4 `x509_node _x509_raw::sign`

Definition at line 181 of file `x509.h`.

#### 4.8.2.5 `x509_node _x509_raw::signalg`

Definition at line 180 of file `x509.h`.

#### 4.8.2.6 `x509_node _x509_raw::subject`

Definition at line 177 of file `x509.h`.

**4.8.2.7 x509\_node\_x509\_raw::subpubkey**

Definition at line 178 of file x509.h.

**4.8.2.8 x509\_node\_x509\_raw::tbs**

Definition at line 170 of file x509.h.

**4.8.2.9 x509\_node\_x509\_raw::tbs\_signalg**

Definition at line 174 of file x509.h.

**4.8.2.10 x509\_node\_x509\_raw::validity**

Definition at line 176 of file x509.h.

**4.8.2.11 x509\_node\_x509\_raw::version**

Definition at line 172 of file x509.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/x509.h](#)

## 4.9 \_x509\_time Struct Reference

```
#include <x509.h>
```

### Data Fields

- [int year](#)
- [int mon](#)
- [int day](#)
- [int hour](#)
- [int min](#)
- [int sec](#)

### 4.9.1 Detailed Description

Definition at line 112 of file x509.h.

### 4.9.2 Field Documentation

#### 4.9.2.1 int \_x509\_time::day

Definition at line 114 of file x509.h.

Referenced by [x509\\_get\\_dates\(\)](#), [x509parse\\_cert\\_info\(\)](#), and [x509parse\\_expired\(\)](#).

#### 4.9.2.2 int \_x509\_time::hour

Definition at line 115 of file x509.h.

Referenced by [x509\\_get\\_dates\(\)](#), and [x509parse\\_cert\\_info\(\)](#).

#### 4.9.2.3 int \_x509\_time::min

Definition at line 115 of file x509.h.

Referenced by [x509\\_get\\_dates\(\)](#), and [x509parse\\_cert\\_info\(\)](#).

#### 4.9.2.4 int \_x509\_time::mon

Definition at line 114 of file x509.h.

Referenced by [x509\\_get\\_dates\(\)](#), [x509parse\\_cert\\_info\(\)](#), and [x509parse\\_expired\(\)](#).

#### 4.9.2.5 int \_x509\_time::sec

Definition at line 115 of file x509.h.

Referenced by [x509\\_get\\_dates\(\)](#), and [x509parse\\_cert\\_info\(\)](#).

#### 4.9.2.6 `int _x509_time::year`

Definition at line 114 of file x509.h.

Referenced by `x509_get_dates()`, `x509parse_cert_info()`, and `x509parse_expired()`.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/x509.h](#)

## 4.10 aes\_context Struct Reference

AES context structure.

```
#include <aes.h>
```

### Data Fields

- `int nr`
- `unsigned long * rk`
- `unsigned long buf[68]`

### 4.10.1 Detailed Description

AES context structure.

Definition at line 13 of file `aes.h`.

### 4.10.2 Field Documentation

#### 4.10.2.1 `unsigned long aes_context::buf[68]`

unaligned data

Definition at line 17 of file `aes.h`.

Referenced by `aes_setkey_dec()`, and `aes_setkey_enc()`.

#### 4.10.2.2 `int aes_context::nr`

number of rounds

Definition at line 15 of file `aes.h`.

Referenced by `aes_crypt_ecb()`, `aes_setkey_dec()`, and `aes_setkey_enc()`.

#### 4.10.2.3 `unsigned long* aes_context::rk`

AES round keys

Definition at line 16 of file `aes.h`.

Referenced by `aes_crypt_ecb()`, `aes_setkey_dec()`, and `aes_setkey_enc()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/aes.h`

## 4.11 agency\_s Struct Reference

The agency handle.

```
#include <libmc.h>
```

### Data Fields

- [int client](#)
- [int server](#)
- [char \\* hostName](#)
- [char \\* filename](#)
- [int portno](#)
- [int portnoc](#)
- [int initInterps](#)
- [struct mc\\_platform\\_s \\* mc\\_platform](#)
- [int default\\_agentstatus](#)
- [int threads](#)
- [int enable\\_security](#)
- [int stack\\_size \[MC\\_THREAD\\_ALL\]](#)
- [int bluetooth](#)
- [char \\* priv\\_key\\_filename](#)
- [char \\* known\\_host\\_filename](#)
- [error\\_code\\_t last\\_error](#)
- [MC\\_AgentInitCallbackFunc\\_t agentInitCallback](#)
- [void \\* agentInitUserData](#)

### 4.11.1 Detailed Description

The agency handle.

Definition at line 225 of file libmc.h.

### 4.11.2 Field Documentation

#### 4.11.2.1 MC\_AgentInitCallbackFunc\_t agency\_s::agentInitCallback

Definition at line 245 of file libmc.h.

Referenced by [agent\\_RunChScriptThread\(\)](#), [MC\\_AddAgentInitCallback\(\)](#), and [MC\\_Initialize\(\)](#).

#### 4.11.2.2 void\* agency\_s::agentInitUserData

Definition at line 246 of file libmc.h.

Referenced by [agent\\_RunChScriptThread\(\)](#), [MC\\_AddAgentInitCallback\(\)](#), and [MC\\_Initialize\(\)](#).

#### 4.11.2.3 int agency\_s::bluetooth

Definition at line 238 of file libmc.h.

Referenced by [MC\\_Initialize\(\)](#), and [mc\\_platform\\_Initialize\(\)](#).



**4.11.2.4 int agency\_s::client**

Definition at line 226 of file libmc.h.

Referenced by MC\_Initialize().

**4.11.2.5 int agency\_s::default\_agentstatus**

Agency default agent status

Definition at line 234 of file libmc.h.

Referenced by MC\_Initialize(), and mc\_platform\_Initialize().

**4.11.2.6 int agency\_s::enable\_security**

Security flag

Definition at line 236 of file libmc.h.

**4.11.2.7 char\* agency\_s::filename**

Definition at line 229 of file libmc.h.

**4.11.2.8 char\* agency\_s::hostName**

Local Hostname

Definition at line 228 of file libmc.h.

Referenced by MC\_End(), and MC\_Initialize().

**4.11.2.9 int agency\_s::initInterps**

Definition at line 232 of file libmc.h.

Referenced by MC\_Initialize(), and mc\_platform\_Initialize().

**4.11.2.10 char\* agency\_s::known\_host\_filename**

Definition at line 241 of file libmc.h.

Referenced by listen\_Thread(), MC\_Initialize(), and message\_send\_Thread().

**4.11.2.11 error\_code\_t agency\_s::last\_error**

Definition at line 242 of file libmc.h.

Referenced by mc\_platform\_Initialize().

#### 4.11.2.12 struct mc\_platform\_s\* agency\_s::mc\_platform

Local MobileC Platform

Definition at line 233 of file libmc.h.

Referenced by MC\_AclSend(), MC\_AclSend\_chdl(), MC\_AddAgent(), MC\_AddAgent\_chdl(), MC\_AddStationaryAgent(), MC\_Barrier(), MC\_Barrier\_chdl(), MC\_BarrierDelete(), MC\_BarrierDelete\_chdl(), MC\_BarrierInit(), MC\_BarrierInit\_chdl(), MC\_CondBroadcast(), MC\_CondBroadcast\_chdl(), MC\_CondReset(), MC\_CondReset\_chdl(), MC\_CondSignal(), MC\_CondSignal\_chdl(), MC\_CondWait(), MC\_CondWait\_chdl(), MC\_DeregisterService(), MC\_DeregisterService\_chdl(), MC\_End(), MC\_End\_chdl(), MC\_FindAgentByID(), MC\_FindAgentByID\_chdl(), MC\_FindAgentByName(), MC\_FindAgentByName\_chdl(), MC\_GetAllAgents(), MC\_HaltAgency(), MC\_HaltAgency\_chdl(), MC\_Initialize(), MC\_LoadAgentFromFile(), MC\_MainLoop(), MC\_MutexLock(), MC\_MutexLock\_chdl(), MC\_MutexUnlock(), MC\_MutexUnlock\_chdl(), MC\_RegisterService(), MC\_RegisterService\_chdl(), MC\_ResetSignal(), MC\_ResumeAgency(), MC\_ResumeAgency\_chdl(), MC\_RetrieveAgent(), MC\_RetrieveAgent\_chdl(), MC\_SearchForService(), MC\_SearchForService\_chdl(), MC\_SemaphorePost(), MC\_SemaphorePost\_chdl(), MC\_SemaphoreWait(), MC\_SemaphoreWait\_chdl(), MC\_SendAgent(), MC\_SendAgentFile(), MC\_SendAgentMigrationMessage(), MC\_SendAgentMigrationMessage\_chdl(), MC\_SendAgentMigrationMessageFile(), MC\_SendSteerCommand(), MC\_SendSteerCommand\_chdl(), MC\_SetDefaultAgentStatus(), MC\_SetDefaultAgentStatus\_chdl(), MC\_Steer(), MC\_SyncDelete(), MC\_SyncDelete\_chdl(), MC\_SyncInit(), MC\_SyncInit\_chdl(), MC\_WaitAgent(), MC\_WaitRetrieveAgent(), and MC\_WaitSignal().

#### 4.11.2.13 int agency\_s::portno

Local port number

Definition at line 230 of file libmc.h.

Referenced by listen\_Thread(), MC\_End(), MC\_Initialize(), and mc\_platform\_Initialize().

#### 4.11.2.14 int agency\_s::portnoc

Definition at line 231 of file libmc.h.

#### 4.11.2.15 char\* agency\_s::priv\_key\_filename

Definition at line 240 of file libmc.h.

Referenced by MC\_Initialize().

#### 4.11.2.16 int agency\_s::server

Definition at line 227 of file libmc.h.

Referenced by MC\_Initialize().

#### 4.11.2.17 int agency\_s::stack\_size[MC\_THREAD\_ALL]

Definition at line 237 of file libmc.h.

Referenced by MC\_Initialize(), and mc\_platform\_Initialize().

**4.11.2.18 int agency\_s::threads**

flag which determines which threads to start

Definition at line 235 of file libmc.h.

Referenced by MC\_End(), MC\_Initialize(), and mc\_platform\_Initialize().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/libmc.h](#)

## 4.12 agent\_datastate\_s Struct Reference

```
#include <agent_datastate.h>
```

### Data Fields

- char \*\* agent\_code\_ids
- char \*\* agent\_codes
- char \* agent\_code
- agent\_task\_p \* tasks
- mxml\_node\_t \* xml\_agent\_root
- mxml\_node\_t \* xml\_root
- int task\_progress
- int return\_data
- int number\_of\_tasks
- int persistent
- int init\_agent\_status
- int progress\_modifier

### 4.12.1 Detailed Description

Definition at line 41 of file agent\_datastate.h.

### 4.12.2 Field Documentation

#### 4.12.2.1 char\* agent\_datastate\_s::agent\_code

Definition at line 45 of file agent\_datastate.h.

Referenced by agent\_datastate\_Copy(), agent\_datastate\_New(), agent\_xml\_parse\_\_agent\_code(), agent\_xml\_parse\_\_tasks(), MC\_ComposeAgentWithWorkgroup(), MC\_PrintAgentCode(), and MC\_RetrieveAgentCode().

#### 4.12.2.2 char\*\* agent\_datastate\_s::agent\_code\_ids

Definition at line 43 of file agent\_datastate.h.

Referenced by agent\_datastate\_Copy(), agent\_datastate\_Destroy(), agent\_xml\_compose\_\_agent\_code(), agent\_xml\_parse\_\_agent\_code(), agent\_xml\_parse\_\_tasks(), MC\_AgentAddTask(), and MC\_ComposeAgentWithWorkgroup().

#### 4.12.2.3 char\*\* agent\_datastate\_s::agent\_codes

Definition at line 44 of file agent\_datastate.h.

Referenced by agent\_datastate\_Copy(), agent\_datastate\_Destroy(), agent\_xml\_compose\_\_agent\_code(), agent\_xml\_parse\_\_agent\_code(), agent\_xml\_parse\_\_tasks(), MC\_AgentAddTask(), and MC\_ComposeAgentWithWorkgroup().

#### 4.12.2.4 int agent\_datastate\_s::init\_agent\_status

Definition at line 63 of file agent\_datastate.h.

Referenced by agent\_datastate\_Copy(), and agent\_datastate\_New().

#### 4.12.2.5 int agent\_datastate\_s::number\_of\_tasks

Definition at line 59 of file agent\_datastate.h.

Referenced by agent\_datastate\_Copy(), agent\_datastate\_Destroy(), agent\_datastate\_New(), agent\_xml\_compose\_\_tasks(), agent\_xml\_parse\_\_agent\_code(), agent\_xml\_parse\_\_tasks(), MC\_AgentAddTask(), MC\_AgentReturnArrayDim(), MC\_AgentReturnArrayExtent(), MC\_AgentReturnArrayNum(), MC\_AgentReturnDataGetSymbolAddr(), MC\_AgentReturnDataSize(), MC\_AgentReturnDataType(), MC\_AgentReturnIsArray(), MC\_ComposeAgentWithWorkgroup(), MC\_GetAgentNumTasks(), MC\_GetAgentReturnData(), MC\_PrintAgentCode(), and message\_InitializeFromAgent().

#### 4.12.2.6 int agent\_datastate\_s::persistent

Definition at line 62 of file agent\_datastate.h.

Referenced by acc\_MessageHandlerThread(), agent\_datastate\_Copy(), agent\_datastate\_New(), and agent\_xml\_compose\_\_task().

#### 4.12.2.7 int agent\_datastate\_s::progress\_modifier

Definition at line 71 of file agent\_datastate.h.

Referenced by agent\_datastate\_New(), and MC\_MigrateAgent().

#### 4.12.2.8 int agent\_datastate\_s::return\_data

Definition at line 56 of file agent\_datastate.h.

Referenced by agent\_datastate\_Copy(), and agent\_datastate\_New().

#### 4.12.2.9 int agent\_datastate\_s::task\_progress

Definition at line 55 of file agent\_datastate.h.

Referenced by agent\_AddPersistentVariable(), agent\_datastate\_Copy(), agent\_datastate\_New(), agent\_RunChScriptThread(), agent\_xml\_compose\_\_tasks(), agent\_xml\_parse\_\_agent\_code(), agent\_xml\_parse\_\_tasks(), interpreter\_variable\_data\_InitializeFromAgent(), MC\_AgentVariableRetrieve(), MC\_AgentVariableRetrieveInfo(), MC\_AgentVariableSave(), MC\_MigrateAgent(), MC\_PrintAgentCode(), MC\_RetrieveAgentCode(), MC\_SaveData\_chdl(), and message\_InitializeFromAgent().

#### 4.12.2.10 agent\_task\_p\* agent\_datastate\_s::tasks

Definition at line 48 of file agent\_datastate.h.

Referenced by agent\_AddPersistentVariable(), agent\_datastate\_Copy(), agent\_datastate\_Destroy(), agent\_datastate\_New(), agent\_xml\_compose\_\_task(), agent\_xml\_parse\_\_agent\_code(),

agent\_xml\_parse\_\_data(), agent\_xml\_parse\_\_task(), agent\_xml\_parse\_\_tasks(), interpreter\_variable\_data\_InitializeFromAgent(), MC\_AgentAddTask(), MC\_AgentReturnArrayDim(), MC\_AgentReturnArrayExtent(), MC\_AgentReturnArrayNum(), MC\_AgentReturnDataGetSymbolAddr(), MC\_AgentReturnDataSize(), MC\_AgentReturnDataType(), MC\_AgentReturnIsArray(), MC\_AgentVariableRetrieve(), MC\_AgentVariableRetrieveInfo(), MC\_AgentVariableSave(), MC\_ComposeAgentWithWorkgroup(), MC\_GetAgentReturnData(), MC\_MigrateAgent(), MC\_SaveData\_chdl(), and message\_InitializeFromAgent().

#### 4.12.2.11 `mxml_node_t* agent_datastate_s::xml_agent_root`

Definition at line 51 of file agent\_datastate.h.

Referenced by agent\_datastate\_New(), agent\_Initialize(), agent\_xml\_parse(), and MC\_GetAgentXMLString().

#### 4.12.2.12 `mxml_node_t* agent_datastate_s::xml_root`

Definition at line 52 of file agent\_datastate.h.

Referenced by agent\_datastate\_Destroy(), agent\_datastate\_New(), agent\_Initialize(), and agent\_return\_xml\_parse().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/agent\\_datastate.h](/home/dko/Projects/mobilec/src/include/agent_datastate.h)

## 4.13 agent\_mailbox\_s Struct Reference

```
#include <agent_mailbox.h>
```

### Data Fields

- struct mail\_queue\_s \* [mail\\_queue](#)

### 4.13.1 Detailed Description

Definition at line 8 of file agent\_mailbox.h.

### 4.13.2 Field Documentation

#### 4.13.2.1 struct mail\_queue\_s\* agent\_mailbox\_s::mail\_queue

Definition at line 10 of file agent\_mailbox.h.

Referenced by agent\_mailbox\_Copy(), agent\_mailbox\_Destroy(), agent\_mailbox\_New(), agent\_mailbox\_Post(), agent\_mailbox\_Retrieve(), and agent\_mailbox\_WaitRetrieve().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/include/[agent\\_mailbox.h](#)

## 4.14 agent\_s Struct Reference

```
#include <agent.h>
```

### Data Fields

- u\_long [id](#)
- char \* [name](#)
- u\_long [connect\\_id](#)
- time\_t [arrival\\_time](#)
- char \* [owner](#)
- char \* [home](#)
- char \* [sender](#)
- int [home\\_port](#)
- char \* [wg\\_code](#)
- char \* [agent\\_address](#)
- int [orphan](#)
- agent\_datastate\_p [datastate](#)
- enum [MC\\_AgentType\\_e](#) [agent\\_type](#)
- enum [MC\\_AgentStatus\\_e](#) [agent\\_status](#)
- int [return\\_data](#)
- ChInterp\_t \* [agent\\_interp](#)
- MUTEX\_T \* [run\\_lock](#)
- int [agent\\_thread\\_id](#)
- THREAD\_T [agent\\_thread](#)
- agent\_mailbox\_p [mailbox](#)
- int [agent\\_pipe\\_active](#)
- int [agent\\_ready\\_to\\_send](#)
- int [agent\\_pipe\\_ready\\_to\\_read](#)
- int [agent\\_script\\_ready](#)
- int [agent\\_persistent](#)
- struct [mc\\_platform\\_s](#) \* [mc\\_platform](#)
- MUTEX\_T \* [lock](#)
- int [binary](#)

### 4.14.1 Detailed Description

Definition at line 48 of file [agent.h](#).

### 4.14.2 Field Documentation

#### 4.14.2.1 char\* agent\_s::agent\_address

Definition at line 66 of file [agent.h](#).

Referenced by [agent\\_Initialize\(\)](#), and [agent\\_NewBinary\(\)](#).



#### 4.14.2.2 ChInterp\_t\* agent\_s::agent\_interp

Definition at line 79 of file agent.h.

Referenced by agent\_AddPersistentVariable(), agent\_Copy(), agent\_Destroy(), agent\_RunChScriptThread(), interpreter\_variable\_data\_Initialize(), interpreter\_variable\_data\_InitializeFromAgent(), MC\_CallAgentFunc(), MC\_CallAgentFuncArg(), MC\_CallAgentFuncV(), MC\_CallAgentFuncVar(), MC\_GetAgentExecEngine(), MC\_TerminateAgent(), and MC\_TerminateAgentWG().

#### 4.14.2.3 int agent\_s::agent\_persistent

Definition at line 99 of file agent.h.

Referenced by agent\_Copy().

#### 4.14.2.4 int agent\_s::agent\_pipe\_active

Definition at line 95 of file agent.h.

Referenced by agent\_Initialize(), and agent\_NewBinary().

#### 4.14.2.5 int agent\_s::agent\_pipe\_ready\_to\_read

Definition at line 97 of file agent.h.

Referenced by agent\_Initialize(), and agent\_NewBinary().

#### 4.14.2.6 int agent\_s::agent\_ready\_to\_send

Definition at line 96 of file agent.h.

Referenced by agent\_Initialize(), and agent\_NewBinary().

#### 4.14.2.7 int agent\_s::agent\_script\_ready

Definition at line 98 of file agent.h.

Referenced by agent\_Initialize(), and agent\_NewBinary().

#### 4.14.2.8 enum MC\_AgentStatus\_e agent\_s::agent\_status

Definition at line 74 of file agent.h.

Referenced by acc\_MessageHandlerThread(), agent\_Copy(), agent\_Destroy(), agent\_Initialize(), agent\_NewBinary(), agent\_queue\_Flush(), agent\_RunChScript(), ams\_ManageAgentList(), ams\_Print(), AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), MC\_ComposeAgentWithWorkgroup(), MC\_GetAgentStatus(), MC\_RetrieveAgent(), MC\_SendAgent(), MC\_SendAgentFile(), MC\_SendAgentMigrationMessageFile(), and MC\_SetAgentStatus().

#### 4.14.2.9 **THREAD\_T agent\_s::agent\_thread**

Definition at line 89 of file agent.h.

Referenced by agent\_RunChScript().

#### 4.14.2.10 **int agent\_s::agent\_thread\_id**

Definition at line 85 of file agent.h.

Referenced by agent\_Initialize(), and agent\_NewBinary().

#### 4.14.2.11 **enum MC\_AgentType\_e agent\_s::agent\_type**

Definition at line 73 of file agent.h.

Referenced by agent\_Copy(), agent\_Initialize(), agent\_xml\_compose\_\_message(), MC\_ComposeAgentWithWorkgroup(), and MC\_GetAgentType().

#### 4.14.2.12 **time\_t agent\_s::arrival\_time**

Definition at line 55 of file agent.h.

Referenced by agent\_Copy(), agent\_Initialize(), agent\_NewBinary(), and MC\_GetAgentArrivalTime().

#### 4.14.2.13 **int agent\_s::binary**

Definition at line 105 of file agent.h.

Referenced by agent\_NewBinary(), and ams\_ManageAgentList().

#### 4.14.2.14 **u\_long agent\_s::connect\_id**

Definition at line 53 of file agent.h.

Referenced by ams\_Print().

#### 4.14.2.15 **agent\_datastate\_p agent\_s::datastate**

Definition at line 72 of file agent.h.

Referenced by acc\_MessageHandlerThread(), agent\_AddPersistentVariable(), agent\_Copy(), agent\_Destroy(), agent\_Initialize(), agent\_return\_xml\_parse(), agent\_RunChScriptThread(), agent\_xml\_compose\_\_agent\_code(), agent\_xml\_compose\_\_task(), agent\_xml\_compose\_\_tasks(), agent\_xml\_parse(), agent\_xml\_parse\_\_agent\_code(), agent\_xml\_parse\_\_data(), agent\_xml\_parse\_\_task(), agent\_xml\_parse\_\_tasks(), interpreter\_variable\_data\_InitializeFromAgent(), MC\_AgentAddTask(), MC\_AgentReturnArrayDim(), MC\_AgentReturnArrayExtent(), MC\_AgentReturnArrayNum(), MC\_AgentReturnDataGetSymbolAddr(), MC\_AgentReturnDataSize(), MC\_AgentReturnDataType(), MC\_AgentReturnIsArray(), MC\_AgentVariableRetrieve(), MC\_AgentVariableRetrieveInfo(), MC\_AgentVariableSave(), MC\_ComposeAgentWithWorkgroup(), MC\_GetAgentNumTasks(), MC\_GetAgentReturnData(), MC\_GetAgentXMLString(), MC\_MigrateAgent(), MC\_PrintAgentCode(), MC\_RetrieveAgentCode(), MC\_SaveData\_chdl(), and message\_InitializeFromAgent().

**4.14.2.16 char\* agent\_s::home**

Definition at line 61 of file agent.h.

Referenced by agent\_Copy(), agent\_Destroy(), agent\_Initialize(), agent\_NewBinary(), agent\_xml\_compose\_\_home(), agent\_xml\_parse\_\_home(), MC\_ComposeAgentWithWorkgroup(), and message\_InitializeFromAgent().

**4.14.2.17 int agent\_s::home\_port**

Definition at line 63 of file agent.h.

Referenced by agent\_Copy().

**4.14.2.18 u\_long agent\_s::id**

Definition at line 51 of file agent.h.

Referenced by agent\_Copy(), agent\_Initialize(), agent\_NewBinary(), agent\_queue\_Flush(), agent\_RunChScriptThread(), ams\_Print(), AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), MC\_GetAgentID(), and MC\_RegisterService().

**4.14.2.19 MUTEX\_T\* agent\_s::lock**

Definition at line 103 of file agent.h.

Referenced by acc\_MessageHandlerThread(), agent\_Copy(), agent\_Destroy(), agent\_Initialize(), agent\_New(), agent\_NewBinary(), ams\_ManageAgentList(), MC\_GetAgentName(), MC\_GetAgentStatus(), MC\_PrintAgentCode(), MC\_RetrieveAgentCode(), and MC\_SetAgentStatus().

**4.14.2.20 agent\_mailbox\_p agent\_s::mailbox**

Definition at line 92 of file agent.h.

Referenced by acc\_connection\_Thread(), agent\_Copy(), agent\_Destroy(), agent\_Initialize(), agent\_NewBinary(), MC\_AclPost(), MC\_AclRetrieve(), and MC\_AclWaitRetrieve().

**4.14.2.21 struct mc\_platform\_s\* agent\_s::mc\_platform**

Definition at line 101 of file agent.h.

Referenced by acc\_connection\_Thread(), agent\_Destroy(), agent\_Initialize(), agent\_NewBinary(), agent\_RunChScript(), agent\_RunChScriptThread(), MC\_AddAgent(), and MC\_SetAgentStatus().

**4.14.2.22 char\* agent\_s::name**

Definition at line 52 of file agent.h.

Referenced by acc\_MessageHandlerThread(), agent\_AddPersistentVariable(), agent\_Copy(), agent\_Destroy(), agent\_queue\_Flush(), agent\_RunChScriptThread(), agent\_xml\_compose\_\_name(), agent\_xml\_parse\_\_name(), ams\_ManageAgentList(), AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), MC\_AddStationaryAgent(), MC\_ComposeAgentWithWorkgroup(), MC\_GetAgentName(), MC\_RegisterService(), and message\_InitializeFromAgent().

**4.14.2.23 int agent\_s::orphan**

Definition at line 68 of file agent.h.

Referenced by agent\_Copy(), agent\_Initialize(), agent\_NewBinary(), ams\_ManageAgentList(), MC\_ComposeAgentWithWorkgroup(), and MC\_SetAgentStatus().

**4.14.2.24 char\* agent\_s::owner**

Definition at line 60 of file agent.h.

Referenced by agent\_Copy(), agent\_Destroy(), agent\_xml\_compose\_\_owner(), agent\_xml\_parse\_\_owner(), and MC\_ComposeAgentWithWorkgroup().

**4.14.2.25 int agent\_s::return\_data**

Definition at line 76 of file agent.h.

Referenced by agent\_Copy().

**4.14.2.26 MUTEX\_T\* agent\_s::run\_lock**

Definition at line 82 of file agent.h.

Referenced by agent\_Copy(), agent\_Destroy(), agent\_Initialize(), agent\_New(), agent\_NewBinary(), ams\_ManageAgentList(), interpreter\_variable\_data\_Initialize(), MC\_CallAgentFunc(), MC\_CallAgentFuncArg(), MC\_CallAgentFuncV(), and MC\_CallAgentFuncVar().

**4.14.2.27 char\* agent\_s::sender**

Definition at line 62 of file agent.h.

Referenced by agent\_Destroy(), agent\_Initialize(), agent\_NewBinary(), and agent\_xml\_parse\_\_sender().

**4.14.2.28 char\* agent\_s::wg\_code**

Definition at line 64 of file agent.h.

Referenced by agent\_Destroy(), agent\_xml\_compose\_\_wg\_code(), agent\_xml\_parse\_\_wg\_code(), MC\_ComposeAgentWithWorkgroup(), MC\_DeleteAgent(), MC\_DeleteAgentWG(), and MC\_TerminateAgentWG().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/agent.h](#)

## 4.15 agent\_task\_s Struct Reference

```
#include <agent_task.h>
```

### Data Fields

- [int number\\_of\\_elements](#)
- [int size\\_of\\_element\\_array](#)
- [int persistent](#)
- [int init\\_agent\\_status](#)
- [char \\* var\\_name](#)
- [char \\* server\\_name](#)
- [char \\* code\\_id](#)
- [interpreter\\_variable\\_data\\_t \\* agent\\_return\\_data](#)
- [struct agent\\_variable\\_list\\_s \\* agent\\_variable\\_list](#)
- [char \\*\\* saved\\_variables](#)
- [int num\\_saved\\_variables](#)

### 4.15.1 Detailed Description

Definition at line 40 of file `agent_task.h`.

### 4.15.2 Field Documentation

#### 4.15.2.1 `interpreter_variable_data_t* agent_task_s::agent_return_data`

Definition at line 52 of file `agent_task.h`.

Referenced by `agent_task_Copy()`, `agent_task_Destroy()`, `agent_xml_compose__task()`, `agent_xml_parse__data()`, `MC_AgentReturnArrayDim()`, `MC_AgentReturnArrayExtent()`, `MC_AgentReturnArrayNum()`, `MC_AgentReturnDataGetSymbolAddr()`, `MC_AgentReturnDataSize()`, `MC_AgentReturnDataType()`, `MC_AgentReturnIsArray()`, and `MC_GetAgentReturnData()`.

#### 4.15.2.2 `struct agent_variable_list_s* agent_task_s::agent_variable_list`

Definition at line 53 of file `agent_task.h`.

Referenced by `agent_AddPersistentVariable()`, `agent_task_Copy()`, `agent_task_Destroy()`, `agent_task_New()`, `agent_xml_compose__task()`, `agent_xml_parse__data()`, `MC_AgentVariableRetrieve()`, `MC_AgentVariableRetrieveInfo()`, and `MC_SaveData_chdl()`.

#### 4.15.2.3 `char* agent_task_s::code_id`

Definition at line 49 of file `agent_task.h`.

Referenced by `agent_task_Copy()`, `agent_task_Destroy()`, `agent_xml_compose__task()`, `agent_xml_parse__agent_code()`, `agent_xml_parse__task()`, and `MC_AgentAddTask()`.

**4.15.2.4 int agent\_task\_s::init\_agent\_status**

Definition at line 45 of file agent\_task.h.

Referenced by agent\_task\_Copy().

**4.15.2.5 int agent\_task\_s::num\_saved\_variables**

Definition at line 55 of file agent\_task.h.

Referenced by agent\_task\_Copy(), agent\_task\_New(), and MC\_AgentVariableSave().

**4.15.2.6 int agent\_task\_s::number\_of\_elements**

Definition at line 42 of file agent\_task.h.

Referenced by agent\_task\_Copy().

**4.15.2.7 int agent\_task\_s::persistent**

Definition at line 44 of file agent\_task.h.

Referenced by agent\_task\_Copy(), agent\_xml\_compose\_\_task(), agent\_xml\_parse\_\_data(), agent\_xml\_parse\_\_task(), MC\_AgentAddTask(), and MC\_ComposeAgentWithWorkgroup().

**4.15.2.8 char\*\* agent\_task\_s::saved\_variables**

Definition at line 54 of file agent\_task.h.

Referenced by agent\_task\_Copy(), agent\_task\_Destroy(), agent\_task\_New(), and MC\_AgentVariableSave().

**4.15.2.9 char\* agent\_task\_s::server\_name**

Definition at line 48 of file agent\_task.h.

Referenced by agent\_task\_Copy(), agent\_task\_Destroy(), agent\_xml\_compose\_\_task(), agent\_xml\_parse\_\_task(), MC\_AgentAddTask(), MC\_ComposeAgentWithWorkgroup(), MC\_MigrateAgent(), and message\_InitializeFromAgent().

**4.15.2.10 int agent\_task\_s::size\_of\_element\_array**

Definition at line 43 of file agent\_task.h.

Referenced by agent\_task\_Copy().

**4.15.2.11 char\* agent\_task\_s::var\_name**

Definition at line 47 of file agent\_task.h.

Referenced by agent\_task\_Copy(), agent\_task\_Destroy(), agent\_xml\_compose\_\_task(), agent\_xml\_parse\_\_task(), interpreter\_variable\_data\_InitializeFromAgent(), MC\_AgentAddTask(), and MC\_ComposeAgentWithWorkgroup().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/agent\\_task.h](#)

## 4.16 AP\_GENERIC\_s Struct Reference

```
#include <ap_queue_template.h>
```

### Data Fields

- void \* [none](#)

### 4.16.1 Detailed Description

Definition at line 41 of file `ap_queue_template.h`.

### 4.16.2 Field Documentation

#### 4.16.2.1 void\* AP\_GENERIC\_s::none

Definition at line 41 of file `ap_queue_template.h`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/include/ap_queue_template.h`



## 4.17 arc4\_context Struct Reference

ARC4 context structure.

```
#include <arc4.h>
```

### Data Fields

- [int](#) `x`
- [int](#) `y`
- unsigned char `m` [256]

#### 4.17.1 Detailed Description

ARC4 context structure.

Definition at line 10 of file `arc4.h`.

#### 4.17.2 Field Documentation

##### 4.17.2.1 unsigned char arc4\_context::m[256]

permutation table

Definition at line 14 of file `arc4.h`.

Referenced by `arc4_crypt()`, and `arc4_setup()`.

##### 4.17.2.2 int arc4\_context::x

permutation index

Definition at line 12 of file `arc4.h`.

Referenced by `arc4_crypt()`, and `arc4_setup()`.

##### 4.17.2.3 int arc4\_context::y

permutation index

Definition at line 13 of file `arc4.h`.

Referenced by `arc4_crypt()`, and `arc4_setup()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/arc4.h`

## 4.18 barrier\_node\_s Struct Reference

```
#include <barrier.h>
```

### Data Fields

- `MUTEX_T * lock`
- `COND_T * cond`
- `int id`
- `int num_registered`
- `int num_waiting`

### 4.18.1 Detailed Description

Definition at line 42 of file barrier.h.

### 4.18.2 Field Documentation

#### 4.18.2.1 `COND_T* barrier_node_s::cond`

Definition at line 43 of file barrier.h.

Referenced by `barrier_node_Destroy()`, `barrier_node_Initialize()`, and `MC_Barrier()`.

#### 4.18.2.2 `int barrier_node_s::id`

Definition at line 44 of file barrier.h.

Referenced by `barrier_node_Initialize()`, `barrier_queue_Add()`, and `barrier_queue_Delete()`.

#### 4.18.2.3 `MUTEX_T* barrier_node_s::lock`

Definition at line 42 of file barrier.h.

Referenced by `barrier_node_Destroy()`, `barrier_node_Initialize()`, and `MC_Barrier()`.

#### 4.18.2.4 `int barrier_node_s::num_registered`

Definition at line 45 of file barrier.h.

Referenced by `barrier_node_Initialize()`, and `MC_Barrier()`.

#### 4.18.2.5 `int barrier_node_s::num_waiting`

Definition at line 46 of file barrier.h.

Referenced by `barrier_node_Initialize()`, and `MC_Barrier()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/include/barrier.h`

## 4.19 barrier\_queue\_s Struct Reference

```
#include <barrier.h>
```

### Data Fields

- `RWLOCK_T * lock`
- `list_p list`
- `int size`

### 4.19.1 Detailed Description

Definition at line 51 of file barrier.h.

### 4.19.2 Field Documentation

#### 4.19.2.1 list\_p barrier\_queue\_s::list

Definition at line 54 of file barrier.h.

Referenced by barrier\_queue\_Add(), barrier\_queue\_Delete(), barrier\_queue\_Destroy(), barrier\_queue\_Get(), barrier\_queue\_New(), and barrier\_queue\_Pop().

#### 4.19.2.2 RWLOCK\_T\* barrier\_queue\_s::lock

Definition at line 52 of file barrier.h.

Referenced by barrier\_queue\_Add(), barrier\_queue\_Delete(), barrier\_queue\_Destroy(), barrier\_queue\_Get(), and barrier\_queue\_New().

#### 4.19.2.3 int barrier\_queue\_s::size

Definition at line 55 of file barrier.h.

Referenced by barrier\_queue\_Add(), and barrier\_queue\_Delete().

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/include/barrier.h`

## 4.20 cmd\_prompt\_s Struct Reference

```
#include <cmd_prompt.h>
```

### Data Fields

- THREAD\_T [thread](#)

### 4.20.1 Detailed Description

Definition at line 41 of file cmd\_prompt.h.

### 4.20.2 Field Documentation

#### 4.20.2.1 THREAD\_T cmd\_prompt\_s::thread

Definition at line 42 of file cmd\_prompt.h.

Referenced by cmd\_prompt\_Start(), and MC\_End().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/include/[cmd\\_prompt.h](#)

## 4.21 `command_s` Struct Reference

```
#include <cmd_prompt.h>
```

### Data Fields

- [int](#) `index`
- [int](#) `num_args`
- [char \\*\\*](#) `args`

### 4.21.1 Detailed Description

Definition at line 46 of file `cmd_prompt.h`.

### 4.21.2 Field Documentation

#### 4.21.2.1 `char** command_s::args`

Definition at line 49 of file `cmd_prompt.h`.

Referenced by `cmd_prompt_Thread()`, `dealloc_command()`, `exec_command()`, `handler_COMPOSE_SEND()`, `handler_HELP()`, `handler_SEND()`, and `process_command()`.

#### 4.21.2.2 `int command_s::index`

Definition at line 47 of file `cmd_prompt.h`.

Referenced by `cmd_prompt_Thread()`, `exec_command()`, and `process_command()`.

#### 4.21.2.3 `int command_s::num_args`

Definition at line 48 of file `cmd_prompt.h`.

Referenced by `cmd_prompt_Thread()`, `dealloc_command()`, `exec_command()`, `handler_COMPOSE_SEND()`, `handler_HELP()`, `handler_SEND()`, and `process_command()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/include/cmd_prompt.h`

## 4.22 connection\_s Struct Reference

```
#include <connection.h>
```

### Data Fields

- [int connect\\_id](#)
- [int nonce](#)
- [char \\* remote\\_hostname](#)
- [struct sockaddr\\_in addr](#)
- [u\\_long clientfd](#)
- [u\\_long serverfd](#)
- [unsigned char \\* AES\\_key](#)

### 4.22.1 Detailed Description

Definition at line 46 of file connection.h.

### 4.22.2 Field Documentation

#### 4.22.2.1 struct sockaddr\_in connection\_s::addr

Definition at line 56 of file connection.h.

Referenced by [connection\\_Copy\(\)](#), [listen\\_Thread\(\)](#), and [message\\_InitializeFromConnection\(\)](#).

#### 4.22.2.2 unsigned char\* connection\_s::AES\_key

Definition at line 63 of file connection.h.

Referenced by [listen\\_Thread\(\)](#), and [rece\\_de\\_msg\(\)](#).

#### 4.22.2.3 u\_long connection\_s::clientfd

Definition at line 59 of file connection.h.

Referenced by [connection\\_Copy\(\)](#), [connection\\_Destroy\(\)](#), [listen\\_Thread\(\)](#), [message\\_InitializeFromConnection\(\)](#), [mtp\\_http\\_InitializeFromConnection\(\)](#), and [rece\\_de\\_msg\(\)](#).

#### 4.22.2.4 int connection\_s::connect\_id

Definition at line 49 of file connection.h.

Referenced by [AP\\_QUEUE\\_SEARCH\\_TEMPLATE\(\)](#), [connection\\_Copy\(\)](#), [listen\\_Thread\(\)](#), and [message\\_InitializeFromConnection\(\)](#).

#### 4.22.2.5 int connection\_s::nonce

Definition at line 51 of file connection.h.

Referenced by [listen\\_Thread\(\)](#), and [rece\\_de\\_msg\(\)](#).

**4.22.2.6 char\* connection\_s::remote\_hostname**

Definition at line 54 of file connection.h.

Referenced by AP\_QUEUE\_SEARCH\_TEMPLATE(), connection\_Copy(), connection\_Destroy(), and listen\_Thread().

**4.22.2.7 u\_long connection\_s::serverfd**

Definition at line 60 of file connection.h.

Referenced by connection\_Copy(), and listen\_Thread().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/include/[connection.h](#)

## 4.23 fipa\_expression\_s::content\_u Union Reference

```
#include <fipa_acl.h>
```

### Data Fields

- struct [fipa\\_word\\_s](#) \* word
- struct [fipa\\_string\\_s](#) \* string
- struct [fipa\\_number\\_s](#) \* number
- struct [fipa\\_DateTime\\_s](#) \* datetime
- struct [fipa\\_expression\\_s](#) \*\* expression

### 4.23.1 Detailed Description

Definition at line 196 of file [fipa\\_acl.h](#).

### 4.23.2 Field Documentation

#### 4.23.2.1 struct fipa\_DateTime\_s\* fipa\_expression\_s::content\_u::datetime

Definition at line 201 of file [fipa\\_acl.h](#).

Referenced by [fipa\\_expression\\_Compose\(\)](#), [fipa\\_expression\\_Copy\(\)](#), and [fipa\\_expression\\_Destroy\(\)](#).

#### 4.23.2.2 struct fipa\_expression\_s\*\* fipa\_expression\_s::content\_u::expression

Definition at line 202 of file [fipa\\_acl.h](#).

Referenced by [fipa\\_expression\\_Compose\(\)](#), [fipa\\_expression\\_Copy\(\)](#), and [fipa\\_expression\\_Destroy\(\)](#).

#### 4.23.2.3 struct fipa\_number\_s\* fipa\_expression\_s::content\_u::number

Definition at line 200 of file [fipa\\_acl.h](#).

Referenced by [fipa\\_expression\\_Compose\(\)](#), [fipa\\_expression\\_Copy\(\)](#), and [fipa\\_expression\\_Destroy\(\)](#).

#### 4.23.2.4 struct fipa\_string\_s\* fipa\_expression\_s::content\_u::string

Definition at line 199 of file [fipa\\_acl.h](#).

Referenced by [fipa\\_expression\\_Compose\(\)](#), [fipa\\_expression\\_Copy\(\)](#), [fipa\\_expression\\_Destroy\(\)](#), [MC\\_AclGetConversationID\(\)](#), and [MC\\_AclSetConversationID\(\)](#).

#### 4.23.2.5 struct fipa\_word\_s\* fipa\_expression\_s::content\_u::word

Definition at line 198 of file [fipa\\_acl.h](#).

Referenced by [fipa\\_expression\\_Compose\(\)](#), [fipa\\_expression\\_Copy\(\)](#), and [fipa\\_expression\\_Destroy\(\)](#).

The documentation for this union was generated from the following file:



- [/home/dko/Projects/mobilec/src/include/fipa\\_acl.h](#)

## 4.24 des3\_context Struct Reference

Triple-DES context structure.

```
#include <des.h>
```

### Data Fields

- [int mode](#)
- unsigned long [sk](#) [96]

### 4.24.1 Detailed Description

Triple-DES context structure.

Definition at line 23 of file des.h.

### 4.24.2 Field Documentation

#### 4.24.2.1 int des3\_context::mode

encrypt/decrypt

Definition at line 25 of file des.h.

#### 4.24.2.2 unsigned long des3\_context::sk[96]

3DES subkeys

Definition at line 26 of file des.h.

Referenced by `des3_crypt_ecb()`, `des3_set2key_dec()`, `des3_set2key_enc()`, `des3_set3key_dec()`, and `des3_set3key_enc()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/des.h`

## 4.25 `des_context` Struct Reference

DES context structure.

```
#include <des.h>
```

### Data Fields

- `int mode`
- unsigned long `sk` [32]

#### 4.25.1 Detailed Description

DES context structure.

Definition at line 13 of file `des.h`.

#### 4.25.2 Field Documentation

##### 4.25.2.1 `int des_context::mode`

encrypt/decrypt

Definition at line 15 of file `des.h`.

##### 4.25.2.2 `unsigned long des_context::sk[32]`

DES subkeys

Definition at line 16 of file `des.h`.

Referenced by `des_crypt_ecb()`, `des_setkey_dec()`, and `des_setkey_enc()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/des.h`

## 4.26 dhm\_context Struct Reference

```
#include <dhm.h>
```

### Data Fields

- [int len](#)
- [mpi P](#)
- [mpi G](#)
- [mpi X](#)
- [mpi GX](#)
- [mpi GY](#)
- [mpi K](#)
- [mpi RP](#)

### 4.26.1 Detailed Description

Definition at line 16 of file dhm.h.

### 4.26.2 Field Documentation

#### 4.26.2.1 mpi dhm\_context::G

generator

Definition at line 20 of file dhm.h.

Referenced by dhm\_free(), dhm\_make\_params(), dhm\_make\_public(), dhm\_read\_params(), main(), ssl\_parse\_server\_key\_exchange(), ssl\_set\_dh\_param(), and ssl\_write\_server\_key\_exchange().

#### 4.26.2.2 mpi dhm\_context::GX

$\text{self} = G^X \bmod P$

Definition at line 22 of file dhm.h.

Referenced by dhm\_free(), dhm\_make\_params(), dhm\_make\_public(), ssl\_write\_client\_key\_exchange(), and ssl\_write\_server\_key\_exchange().

#### 4.26.2.3 mpi dhm\_context::GY

$\text{peer} = G^Y \bmod P$

Definition at line 23 of file dhm.h.

Referenced by dhm\_calc\_secret(), dhm\_free(), dhm\_read\_params(), dhm\_read\_public(), ssl\_parse\_client\_key\_exchange(), and ssl\_parse\_server\_key\_exchange().

#### 4.26.2.4 mpi dhm\_context::K

$\text{key} = \text{GY}^X \bmod P$

Definition at line 24 of file dhm.h.

Referenced by dhm\_calc\_secret(), dhm\_free(), ssl\_parse\_client\_key\_exchange(), and ssl\_write\_client\_key\_exchange().

#### 4.26.2.5 int dhm\_context::len

size(P) in chars

Definition at line 18 of file dhm.h.

Referenced by dhm\_make\_params(), dhm\_make\_public(), dhm\_read\_params(), dhm\_read\_public(), main(), ssl\_parse\_client\_key\_exchange(), ssl\_parse\_server\_key\_exchange(), and ssl\_write\_client\_key\_exchange().

#### 4.26.2.6 mpi dhm\_context::P

prime modulus

Definition at line 19 of file dhm.h.

Referenced by dhm\_calc\_secret(), dhm\_free(), dhm\_make\_params(), dhm\_make\_public(), dhm\_read\_params(), main(), ssl\_parse\_server\_key\_exchange(), ssl\_set\_dh\_param(), and ssl\_write\_server\_key\_exchange().

#### 4.26.2.7 mpi dhm\_context::RP

cached  $R^2 \bmod P$

Definition at line 25 of file dhm.h.

Referenced by dhm\_calc\_secret(), dhm\_free(), dhm\_make\_params(), and dhm\_make\_public().

#### 4.26.2.8 mpi dhm\_context::X

secret value

Definition at line 21 of file dhm.h.

Referenced by dhm\_calc\_secret(), dhm\_free(), dhm\_make\_params(), dhm\_make\_public(), ssl\_write\_client\_key\_exchange(), and ssl\_write\_server\_key\_exchange().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/[dhm.h](#)

## 4.27 dynstring\_s Struct Reference

```
#include <dynstring.h>
```

### Data Fields

- [int len](#)
- [int size](#)
- [char \\* message](#)

### 4.27.1 Detailed Description

Definition at line 9 of file dynstring.h.

### 4.27.2 Field Documentation

#### 4.27.2.1 int dynstring\_s::len

Definition at line 10 of file dynstring.h.

Referenced by [dynstring\\_Append\(\)](#), [dynstring\\_New\(\)](#), and [mtp\\_http\\_CreateMessage\(\)](#).

#### 4.27.2.2 char\* dynstring\_s::message

Definition at line 12 of file dynstring.h.

Referenced by [dynstring\\_Append\(\)](#), [dynstring\\_Destroy\(\)](#), [dynstring\\_New\(\)](#), [fipa\\_envelope\\_Compose\\_\\_\\_from\(\)](#), [MC\\_AclSend\(\)](#), [message\\_send\\_Thread\(\)](#), and [mtp\\_http\\_CreateMessage\(\)](#).

#### 4.27.2.3 int dynstring\_s::size

Definition at line 11 of file dynstring.h.

Referenced by [dynstring\\_Append\(\)](#), and [dynstring\\_New\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/dynstring.h](#)

## 4.28 fipa\_acl\_envelope\_Received\_s Struct Reference

```
#include <fipa_acl_envelope.h>
```

### Data Fields

- struct [fipa\\_url\\_s](#) \* [received\\_by](#)
- struct [fipa\\_url\\_s](#) \* [received\\_from](#)
- struct [fipa\\_DateTime\\_s](#) \* [received\\_date](#)
- char \* [received\\_id](#)
- struct [fipa\\_url\\_s](#) \* [received\\_via](#)

### 4.28.1 Detailed Description

Definition at line 7 of file `fipa_acl_envelope.h`.

### 4.28.2 Field Documentation

#### 4.28.2.1 struct [fipa\\_url\\_s](#)\* [fipa\\_acl\\_envelope\\_Received\\_s::received\\_by](#)

Definition at line 9 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_envelope_Received_Copy()`, `fipa_acl_envelope_Received_Destroy()`, and `fipa_envelope_HandleReceived()`.

#### 4.28.2.2 struct [fipa\\_DateTime\\_s](#)\* [fipa\\_acl\\_envelope\\_Received\\_s::received\\_date](#)

Definition at line 11 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_envelope_Received_Copy()`, `fipa_acl_envelope_Received_Destroy()`, and `fipa_envelope_HandleReceived()`.

#### 4.28.2.3 struct [fipa\\_url\\_s](#)\* [fipa\\_acl\\_envelope\\_Received\\_s::received\\_from](#)

Definition at line 10 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_envelope_Received_Copy()`, `fipa_acl_envelope_Received_Destroy()`, and `fipa_envelope_HandleReceived()`.

#### 4.28.2.4 char\* [fipa\\_acl\\_envelope\\_Received\\_s::received\\_id](#)

Definition at line 12 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_envelope_Received_Copy()`, `fipa_acl_envelope_Received_Destroy()`, and `fipa_envelope_HandleReceived()`.

#### 4.28.2.5 struct [fipa\\_url\\_s](#)\* [fipa\\_acl\\_envelope\\_Received\\_s::received\\_via](#)

Definition at line 13 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_envelope_Received_Copy()`, `fipa_acl_envelope_Received_Destroy()`, and `fipa_acl_envelope_HandleReceived()`.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl\\_envelope.h](#)



## 4.29 fipa\_acl\_envelope\_s Struct Reference

```
#include <fipa_acl_envelope.h>
```

### Data Fields

- [int num\\_params](#)
- [struct fipa\\_acl\\_Param\\_s \\*\\* params](#)

### 4.29.1 Detailed Description

Definition at line 40 of file `fipa_acl_envelope.h`.

### 4.29.2 Field Documentation

#### 4.29.2.1 `int fipa_acl_envelope_s::num_params`

Definition at line 42 of file `fipa_acl_envelope.h`.

Referenced by `acc_connection_Thread()`, `fipa_acl_envelope_Copy()`, `fipa_acl_envelope_Destroy()`, and `fipa_envelope_HandleParams()`.

#### 4.29.2.2 `struct fipa_acl_Param_s** fipa_acl_envelope_s::params`

Definition at line 43 of file `fipa_acl_envelope.h`.

Referenced by `acc_connection_Thread()`, `fipa_acl_envelope_Copy()`, `fipa_acl_envelope_Destroy()`, `fipa_envelope_HandleAclRepresentation()`, `fipa_envelope_HandleComments()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandleFrom()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleParams()`, `fipa_envelope_HandlePayloadEncoding()`, `fipa_envelope_HandlePayloadLength()`, `fipa_envelope_HandleReceived()`, and `fipa_envelope_HandleTo()`.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl\\_envelope.h](#)

## 4.30 fipa\_acl\_message\_s Struct Reference

```
#include <fipa_acl.h>
```

### Data Fields

- enum [fipa\\_performative\\_e](#) performative
- struct [fipa\\_agent\\_identifier\\_s](#) \* sender
- struct [fipa\\_agent\\_identifier\\_set\\_s](#) \* receiver
- int receiver\_num
- struct [fipa\\_agent\\_identifier\\_set\\_s](#) \* reply\_to
- struct [fipa\\_string\\_s](#) \* content
- struct [fipa\\_expression\\_s](#) \* language
- struct [fipa\\_expression\\_s](#) \* encoding
- struct [fipa\\_expression\\_s](#) \* ontology
- enum [fipa\\_protocol\\_e](#) protocol
- struct [fipa\\_expression\\_s](#) \* conversation\_id
- struct [fipa\\_expression\\_s](#) \* reply\_with
- struct [fipa\\_expression\\_s](#) \* in\_reply\_to
- struct [fipa\\_DateTime\\_s](#) \* reply\_by

### 4.30.1 Detailed Description

Definition at line 105 of file [fipa\\_acl.h](#).

### 4.30.2 Field Documentation

#### 4.30.2.1 struct [fipa\\_string\\_s](#)\* [fipa\\_acl\\_message\\_s::content](#)

Definition at line 117 of file [fipa\\_acl.h](#).

Referenced by [fipa\\_acl\\_Compose\(\)](#), [fipa\\_acl\\_message\\_Copy\(\)](#), [fipa\\_acl\\_message\\_Destroy\(\)](#), [fipa\\_message\\_parameter\\_Parse\(\)](#), [MC\\_AclGetContent\(\)](#), [MC\\_AclSetContent\(\)](#), and [MC\\_AclSetContent\\_chdl\(\)](#).

#### 4.30.2.2 struct [fipa\\_expression\\_s](#)\* [fipa\\_acl\\_message\\_s::conversation\\_id](#)

Definition at line 128 of file [fipa\\_acl.h](#).

Referenced by [fipa\\_acl\\_Compose\(\)](#), [fipa\\_acl\\_message\\_Copy\(\)](#), [fipa\\_acl\\_message\\_Destroy\(\)](#), [fipa\\_message\\_parameter\\_Parse\(\)](#), [fipa\\_Reply\(\)](#), [MC\\_AclGetConversationID\(\)](#), and [MC\\_AclSetConversationID\(\)](#).

#### 4.30.2.3 struct [fipa\\_expression\\_s](#)\* [fipa\\_acl\\_message\\_s::encoding](#)

Definition at line 121 of file [fipa\\_acl.h](#).

Referenced by [fipa\\_acl\\_Compose\(\)](#), [fipa\\_acl\\_message\\_Copy\(\)](#), [fipa\\_acl\\_message\\_Destroy\(\)](#), and [fipa\\_message\\_parameter\\_Parse\(\)](#).

**4.30.2.4 struct fipa\_expression\_s\* fipa\_acl\_message\_s::in\_reply\_to**

Definition at line 132 of file fipa\_acl.h.

Referenced by fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_acl\_message\_Destroy(), and fipa\_message\_parameter\_Parse().

**4.30.2.5 struct fipa\_expression\_s\* fipa\_acl\_message\_s::language**

Definition at line 119 of file fipa\_acl.h.

Referenced by fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_acl\_message\_Destroy(), and fipa\_message\_parameter\_Parse().

**4.30.2.6 struct fipa\_expression\_s\* fipa\_acl\_message\_s::ontology**

Definition at line 123 of file fipa\_acl.h.

Referenced by fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_acl\_message\_Destroy(), and fipa\_message\_parameter\_Parse().

**4.30.2.7 enum fipa\_performative\_e fipa\_acl\_message\_s::performative**

Definition at line 107 of file fipa\_acl.h.

Referenced by fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_acl\_Parse(), MC\_AclGetPerformative(), MC\_AclGetProtocol(), MC\_AclSetPerformative(), and MC\_AclSetPerformative\_chdl().

**4.30.2.8 enum fipa\_protocol\_e fipa\_acl\_message\_s::protocol**

Definition at line 126 of file fipa\_acl.h.

Referenced by fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_message\_parameter\_Parse(), fipa\_Reply(), MC\_AclSetProtocol(), and MC\_AclSetProtocol\_chdl().

**4.30.2.9 struct fipa\_agent\_identifier\_set\_s\* fipa\_acl\_message\_s::receiver**

Definition at line 111 of file fipa\_acl.h.

Referenced by AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_acl\_message\_Destroy(), fipa\_envelope\_Compose\_intended\_receiver(), fipa\_envelope\_Compose\_to(), fipa\_message\_parameter\_Parse(), fipa\_Reply(), MC\_AclAddReceiver(), and MC\_AclSend().

**4.30.2.10 int fipa\_acl\_message\_s::receiver\_num**

Definition at line 112 of file fipa\_acl.h.

Referenced by MC\_AclAddReceiver().

**4.30.2.11 struct fipa\_DateTime\_s\* fipa\_acl\_message\_s::reply\_by**

Definition at line 134 of file fipa\_acl.h.

Referenced by fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_acl\_message\_Destroy(), and fipa\_message\_parameter\_Parse().

**4.30.2.12 struct fipa\_agent\_identifier\_set\_s\* fipa\_acl\_message\_s::reply\_to**

Definition at line 114 of file fipa\_acl.h.

Referenced by fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_acl\_message\_Destroy(), fipa\_message\_parameter\_Parse(), fipa\_Reply(), and MC\_AclAddReplyTo().

**4.30.2.13 struct fipa\_expression\_s\* fipa\_acl\_message\_s::reply\_with**

Definition at line 130 of file fipa\_acl.h.

Referenced by fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_acl\_message\_Destroy(), and fipa\_message\_parameter\_Parse().

**4.30.2.14 struct fipa\_agent\_identifier\_s\* fipa\_acl\_message\_s::sender**

Definition at line 109 of file fipa\_acl.h.

Referenced by fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_acl\_message\_Destroy(), fipa\_envelope\_Compose\_\_from(), fipa\_message\_parameter\_Parse(), fipa\_Reply(), MC\_AclGetSender(), and MC\_AclSetSender().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl.h](#)

## 4.31 fipa\_acl\_Param\_s Struct Reference

```
#include <fipa_acl_envelope.h>
```

### Data Fields

- struct [fipa\\_agent\\_identifier\\_set\\_s](#) \* to
- struct [fipa\\_agent\\_identifier\\_s](#) \* from
- char \* [comments](#)
- char \* [acl\\_representation](#)
- char \* [payload\\_length](#)
- char \* [payload\\_encoding](#)
- struct [fipa\\_DateTime\\_s](#) \* date
- struct [fipa\\_agent\\_identifier\\_set\\_s](#) \* intended\_receiver
- struct [fipa\\_acl\\_envelope\\_Received\\_s](#) \* received

### 4.31.1 Detailed Description

Definition at line 22 of file [fipa\\_acl\\_envelope.h](#).

### 4.31.2 Field Documentation

#### 4.31.2.1 char\* fipa\_acl\_Param\_s::acl\_representation

Definition at line 27 of file [fipa\\_acl\\_envelope.h](#).

Referenced by [fipa\\_acl\\_Param\\_Copy\(\)](#), [fipa\\_acl\\_Param\\_Destroy\(\)](#), and [fipa\\_envelope\\_HandleAclRepresentation\(\)](#).

#### 4.31.2.2 char\* fipa\_acl\_Param\_s::comments

Definition at line 26 of file [fipa\\_acl\\_envelope.h](#).

Referenced by [fipa\\_acl\\_Param\\_Copy\(\)](#), [fipa\\_acl\\_Param\\_Destroy\(\)](#), and [fipa\\_envelope\\_HandleComments\(\)](#).

#### 4.31.2.3 struct fipa\_DateTime\_s\* fipa\_acl\_Param\_s::date

Definition at line 30 of file [fipa\\_acl\\_envelope.h](#).

Referenced by [fipa\\_acl\\_Param\\_Copy\(\)](#), [fipa\\_acl\\_Param\\_Destroy\(\)](#), and [fipa\\_envelope\\_HandleDate\(\)](#).

#### 4.31.2.4 struct fipa\_agent\_identifier\_s\* fipa\_acl\_Param\_s::from

Definition at line 25 of file [fipa\\_acl\\_envelope.h](#).

Referenced by [fipa\\_acl\\_Param\\_Copy\(\)](#), [fipa\\_acl\\_Param\\_Destroy\(\)](#), and [fipa\\_envelope\\_HandleFrom\(\)](#).

#### 4.31.2.5 struct fipa\_agent\_identifier\_set\_s\* fipa\_acl\_Param\_s::intended\_receiver

Definition at line 31 of file fipa\_acl\_envelope.h.

Referenced by fipa\_acl\_Param\_Copy(), fipa\_acl\_Param\_Destroy(), and fipa\_envelope\_HandleIntendedReceiver().

#### 4.31.2.6 char\* fipa\_acl\_Param\_s::payload\_encoding

Definition at line 29 of file fipa\_acl\_envelope.h.

Referenced by fipa\_acl\_Param\_Copy(), fipa\_acl\_Param\_Destroy(), and fipa\_envelope\_HandlePayloadEncoding().

#### 4.31.2.7 char\* fipa\_acl\_Param\_s::payload\_length

Definition at line 28 of file fipa\_acl\_envelope.h.

Referenced by fipa\_acl\_Param\_Copy(), fipa\_acl\_Param\_Destroy(), and fipa\_envelope\_HandlePayloadLength().

#### 4.31.2.8 struct fipa\_acl\_envelope\_Received\_s\* fipa\_acl\_Param\_s::received

Definition at line 32 of file fipa\_acl\_envelope.h.

Referenced by fipa\_acl\_Param\_Copy(), fipa\_acl\_Param\_Destroy(), and fipa\_envelope\_HandleReceived().

#### 4.31.2.9 struct fipa\_agent\_identifier\_set\_s\* fipa\_acl\_Param\_s::to

Definition at line 24 of file fipa\_acl\_envelope.h.

Referenced by acc\_connection\_Thread(), fipa\_acl\_Param\_Copy(), fipa\_acl\_Param\_Destroy(), and fipa\_envelope\_HandleTo().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl\\_envelope.h](#)

## 4.32 fipa\_agent\_identifier\_s Struct Reference

```
#include <fipa_acl.h>
```

### Data Fields

- char \* [name](#)
- struct [fipa\\_url\\_sequence\\_s](#) \* [addresses](#)
- struct [fipa\\_agent\\_identifier\\_set\\_s](#) \* [resolvers](#)

### 4.32.1 Detailed Description

Definition at line 180 of file [fipa\\_acl.h](#).

### 4.32.2 Field Documentation

#### 4.32.2.1 struct fipa\_url\_sequence\_s\* fipa\_agent\_identifier\_s::addresses

Definition at line 183 of file [fipa\\_acl.h](#).

Referenced by [acc\\_connection\\_Thread\(\)](#), [fipa\\_agent\\_identifier\\_Compose\(\)](#), [fipa\\_agent\\_identifier\\_Copy\(\)](#), [fipa\\_agent\\_identifier\\_Destroy\(\)](#), [fipa\\_envelope\\_Compose\\_\\_from\(\)](#), [fipa\\_envelope\\_Compose\\_\\_intended\\_receiver\(\)](#), [fipa\\_envelope\\_Compose\\_\\_to\(\)](#), [fipa\\_envelope\\_ParseAddresses\(\)](#), [MC\\_AclAddReceiver\(\)](#), [MC\\_AclAddReplyTo\(\)](#), [MC\\_AclGetSender\(\)](#), [MC\\_AclSend\(\)](#), and [MC\\_AclSetSender\(\)](#).

#### 4.32.2.2 char\* fipa\_agent\_identifier\_s::name

Definition at line 182 of file [fipa\\_acl.h](#).

Referenced by [acc\\_connection\\_Thread\(\)](#), [AP\\_QUEUE\\_STD\\_DEFN\\_TEMPLATE\(\)](#), [fipa\\_agent\\_identifier\\_Compose\(\)](#), [fipa\\_agent\\_identifier\\_Copy\(\)](#), [fipa\\_agent\\_identifier\\_Destroy\(\)](#), [fipa\\_envelope\\_Compose\\_\\_from\(\)](#), [fipa\\_envelope\\_Compose\\_\\_intended\\_receiver\(\)](#), [fipa\\_envelope\\_Compose\\_\\_to\(\)](#), [MC\\_AclAddReceiver\(\)](#), [MC\\_AclAddReplyTo\(\)](#), [MC\\_AclGetSender\(\)](#), [MC\\_AclSend\(\)](#), and [MC\\_AclSetSender\(\)](#).

#### 4.32.2.3 struct fipa\_agent\_identifier\_set\_s\* fipa\_agent\_identifier\_s::resolvers

Definition at line 184 of file [fipa\\_acl.h](#).

Referenced by [fipa\\_agent\\_identifier\\_Compose\(\)](#), [fipa\\_agent\\_identifier\\_Copy\(\)](#), [fipa\\_agent\\_identifier\\_Destroy\(\)](#), and [fipa\\_envelope\\_ParseResolvers\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl.h](#)

## 4.33 fipa\_agent\_identifier\_set\_s Struct Reference

```
#include <fipa_acl.h>
```

### Data Fields

- [int num](#)
- [int retain\\_order](#)
- struct [fipa\\_agent\\_identifier\\_s](#) \*\* [fipa\\_agent\\_identifiers](#)

### 4.33.1 Detailed Description

Definition at line 166 of file [fipa\\_acl.h](#).

### 4.33.2 Field Documentation

#### 4.33.2.1 struct fipa\_agent\_identifier\_s\*\* fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers

Definition at line 170 of file [fipa\\_acl.h](#).

Referenced by [acc\\_connection\\_Thread\(\)](#), [AP\\_QUEUE\\_STD\\_DEFN\\_TEMPLATE\(\)](#), [fipa\\_agent\\_identifier\\_set\\_Compose\(\)](#), [fipa\\_agent\\_identifier\\_set\\_Copy\(\)](#), [fipa\\_agent\\_identifier\\_set\\_Destroy\(\)](#), [fipa\\_envelope\\_Compose\\_\\_intended\\_receiver\(\)](#), [fipa\\_envelope\\_Compose\\_\\_to\(\)](#), [fipa\\_envelope\\_HandleIntendedReceiver\(\)](#), [fipa\\_envelope\\_HandleTo\(\)](#), [fipa\\_envelope\\_ParseResolvers\(\)](#), [fipa\\_Reply\(\)](#), [MC\\_AclAddReceiver\(\)](#), [MC\\_AclAddReplyTo\(\)](#), and [MC\\_AclSend\(\)](#).

#### 4.33.2.2 int fipa\_agent\_identifier\_set\_s::num

Definition at line 168 of file [fipa\\_acl.h](#).

Referenced by [acc\\_connection\\_Thread\(\)](#), [fipa\\_agent\\_identifier\\_Compose\(\)](#), [fipa\\_agent\\_identifier\\_set\\_Compose\(\)](#), [fipa\\_agent\\_identifier\\_set\\_Copy\(\)](#), [fipa\\_agent\\_identifier\\_set\\_Destroy\(\)](#), [fipa\\_envelope\\_Compose\\_\\_to\(\)](#), [fipa\\_envelope\\_HandleIntendedReceiver\(\)](#), [fipa\\_envelope\\_HandleTo\(\)](#), [fipa\\_envelope\\_ParseResolvers\(\)](#), [fipa\\_Reply\(\)](#), [MC\\_AclAddReceiver\(\)](#), [MC\\_AclAddReplyTo\(\)](#), and [MC\\_AclSend\(\)](#).

#### 4.33.2.3 int fipa\_agent\_identifier\_set\_s::retain\_order

Definition at line 169 of file [fipa\\_acl.h](#).

Referenced by [fipa\\_agent\\_identifier\\_set\\_Copy\(\)](#), [fipa\\_agent\\_identifier\\_set\\_Parse\(\)](#), [fipa\\_envelope\\_HandleIntendedReceiver\(\)](#), [fipa\\_envelope\\_HandleTo\(\)](#), [fipa\\_envelope\\_ParseResolvers\(\)](#), and [fipa\\_Reply\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl.h](#)



## 4.34 fipa\_comm\_action\_t Struct Reference

```
#include <fipa_comm.h>
```

### Data Fields

- enum [fipa\\_comm\\_action\\_type\\_e](#) type
- [fipa\\_comm\\_callback](#) callback
- [fipa\\_comm\\_message\\_check\\_p](#) check\_list
- [fipa\\_comm\\_reply\\_p](#) reply
- struct [fipa\\_comm\\_action\\_t](#) \* next

#### 4.34.1 Detailed Description

Definition at line 99 of file [fipa\\_comm.h](#).

#### 4.34.2 Field Documentation

##### 4.34.2.1 [fipa\\_comm\\_callback](#) [fipa\\_comm\\_action\\_t::callback](#)

Definition at line 102 of file [fipa\\_comm.h](#).

##### 4.34.2.2 [fipa\\_comm\\_message\\_check\\_p](#) [fipa\\_comm\\_action\\_t::check\\_list](#)

Definition at line 103 of file [fipa\\_comm.h](#).

##### 4.34.2.3 [struct fipa\\_comm\\_action\\_t\\*](#) [fipa\\_comm\\_action\\_t::next](#)

Definition at line 105 of file [fipa\\_comm.h](#).

##### 4.34.2.4 [fipa\\_comm\\_reply\\_p](#) [fipa\\_comm\\_action\\_t::reply](#)

Definition at line 104 of file [fipa\\_comm.h](#).

##### 4.34.2.5 [enum fipa\\_comm\\_action\\_type\\_e](#) [fipa\\_comm\\_action\\_t::type](#)

Definition at line 101 of file [fipa\\_comm.h](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_comm.h](#)

## 4.35 fipa\_comm\_message\_check\_t Struct Reference

```
#include <fipa_comm.h>
```

### Data Fields

- char \* [macro](#)
- enum [fipa\\_comm\\_check\\_type\\_e](#) type
- struct [fipa\\_comm\\_message\\_check\\_t](#) \* next

### 4.35.1 Detailed Description

Definition at line 72 of file [fipa\\_comm.h](#).

### 4.35.2 Field Documentation

#### 4.35.2.1 char\* fipa\_comm\_message\_check\_t::macro

Definition at line 74 of file [fipa\\_comm.h](#).

#### 4.35.2.2 struct fipa\_comm\_message\_check\_t\* fipa\_comm\_message\_check\_t::next

Definition at line 76 of file [fipa\\_comm.h](#).

#### 4.35.2.3 enum fipa\_comm\_check\_type\_e fipa\_comm\_message\_check\_t::type

Definition at line 75 of file [fipa\\_comm.h](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_comm.h](#)

## 4.36 fipa\_comm\_performative\_t Struct Reference

```
#include <fipa_comm.h>
```

### Data Fields

- [fipa\\_comm\\_action\\_p action\\_list](#)
- [fipa\\_comm\\_reply\\_p default\\_reply](#)

### 4.36.1 Detailed Description

Definition at line 115 of file fipa\_comm.h.

### 4.36.2 Field Documentation

#### 4.36.2.1 fipa\_comm\_action\_p fipa\_comm\_performative\_t::action\_list

Definition at line 117 of file fipa\_comm.h.

#### 4.36.2.2 fipa\_comm\_reply\_p fipa\_comm\_performative\_t::default\_reply

Definition at line 118 of file fipa\_comm.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_comm.h](#)

## 4.37 fipa\_comm\_protocol\_cn\_t Struct Reference

```
#include <fipa_comm.h>
```

### Data Fields

- [protocol\\_contract\\_net\\_callback contract\\_func](#)
- [fipa\\_comm\\_callback proposal\\_func](#)
- char [require\\_inform](#)
- int [time\\_out](#)
- char \* [cfp\\_request](#)
- [fipa\\_list\\_p](#) [replies](#)
- [fipa\\_list\\_p](#) [queue](#)

### 4.37.1 Detailed Description

Definition at line 143 of file `fipa_comm.h`.

### 4.37.2 Field Documentation

#### 4.37.2.1 char\* fipa\_comm\_protocol\_cn\_t::cfp\_request

Definition at line 149 of file `fipa_comm.h`.

#### 4.37.2.2 protocol\_contract\_net\_callback fipa\_comm\_protocol\_cn\_t::contract\_func

Definition at line 145 of file `fipa_comm.h`.

#### 4.37.2.3 fipa\_comm\_callback fipa\_comm\_protocol\_cn\_t::proposal\_func

Definition at line 146 of file `fipa_comm.h`.

#### 4.37.2.4 fipa\_list\_p fipa\_comm\_protocol\_cn\_t::queue

Definition at line 153 of file `fipa_comm.h`.

#### 4.37.2.5 fipa\_list\_p fipa\_comm\_protocol\_cn\_t::replies

Definition at line 152 of file `fipa_comm.h`.

#### 4.37.2.6 char fipa\_comm\_protocol\_cn\_t::require\_inform

Definition at line 147 of file `fipa_comm.h`.

**4.37.2.7 int fipa\_comm\_protocol\_cn\_t::time\_out**

Definition at line 148 of file fipa\_comm.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_comm.h](#)

## 4.38 fipa\_comm\_protocol\_t Struct Reference

```
#include <fipa_comm.h>
```

### Data Fields

- char [error](#)
- char [state](#)
- char \* [conversation\\_id](#)
- char \* [network](#)
- enum [fipa\\_protocol\\_e](#) type
- void \* [info](#)
- struct [fipa\\_comm\\_protocol\\_t](#) \* [next](#)

### 4.38.1 Detailed Description

Definition at line 163 of file `fipa_comm.h`.

### 4.38.2 Field Documentation

#### 4.38.2.1 char\* fipa\_comm\_protocol\_t::conversation\_id

Definition at line 166 of file `fipa_comm.h`.

#### 4.38.2.2 char fipa\_comm\_protocol\_t::error

Definition at line 164 of file `fipa_comm.h`.

#### 4.38.2.3 void\* fipa\_comm\_protocol\_t::info

Definition at line 169 of file `fipa_comm.h`.

#### 4.38.2.4 char\* fipa\_comm\_protocol\_t::network

Definition at line 167 of file `fipa_comm.h`.

#### 4.38.2.5 struct fipa\_comm\_protocol\_t\* fipa\_comm\_protocol\_t::next

Definition at line 170 of file `fipa_comm.h`.

#### 4.38.2.6 char fipa\_comm\_protocol\_t::state

Definition at line 165 of file `fipa_comm.h`.

#### 4.38.2.7 enum fipa\_protocol\_e fipa\_comm\_protocol\_t::type

Definition at line 168 of file fipa\_comm.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_comm.h](#)

## 4.39 fipa\_comm\_reply\_t Struct Reference

```
#include <fipa_comm.h>
```

### Data Fields

- enum [fipa\\_performative\\_e](#) type
- char \* [content](#)

### 4.39.1 Detailed Description

Definition at line 86 of file [fipa\\_comm.h](#).

### 4.39.2 Field Documentation

#### 4.39.2.1 char\* fipa\_comm\_reply\_t::content

Definition at line 89 of file [fipa\\_comm.h](#).

#### 4.39.2.2 enum fipa\_performative\_e fipa\_comm\_reply\_t::type

Definition at line 88 of file [fipa\\_comm.h](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_comm.h](#)



## 4.40 fipa\_comm\_t Struct Reference

```
#include <fipa_comm.h>
```

### Data Fields

- [int error](#)
- [char \\* agent\\_name](#)
- [char \\* agent\\_address](#)
- [void \\* agent](#)
- [char debug](#)
- [fipa\\_comm\\_performative\\_p performative](#) [MC\_NUM\_PERFORMATIVES]
- [fipa\\_comm\\_reply\\_p default\\_reply](#)
- [fipa\\_comm\\_protocol\\_p protocol](#) [FIPA\_PROTOCOL\_END]
- [fipa\\_comm\\_callback pCallbacks](#) [FIPA\_PROTOCOL\_END]

### 4.40.1 Detailed Description

Definition at line 179 of file fipa\_comm.h.

### 4.40.2 Field Documentation

#### 4.40.2.1 void\* fipa\_comm\_t::agent

Definition at line 184 of file fipa\_comm.h.

#### 4.40.2.2 char\* fipa\_comm\_t::agent\_address

Definition at line 183 of file fipa\_comm.h.

#### 4.40.2.3 char\* fipa\_comm\_t::agent\_name

Definition at line 182 of file fipa\_comm.h.

#### 4.40.2.4 char fipa\_comm\_t::debug

Definition at line 185 of file fipa\_comm.h.

#### 4.40.2.5 fipa\_comm\_reply\_p fipa\_comm\_t::default\_reply

Definition at line 187 of file fipa\_comm.h.

#### 4.40.2.6 int fipa\_comm\_t::error

Definition at line 181 of file fipa\_comm.h.

**4.40.2.7 fipa\_comm\_callback fipa\_comm\_t::pCallbacks[FIPA\_PROTOCOL\_END]**

Definition at line 189 of file fipa\_comm.h.

**4.40.2.8 fipa\_comm\_performative\_p fipa\_comm\_t::performative[MC\_NUM\_PERFORMATIVES]**

Definition at line 186 of file fipa\_comm.h.

**4.40.2.9 fipa\_comm\_protocol\_p fipa\_comm\_t::protocol[FIPA\_PROTOCOL\_END]**

Definition at line 188 of file fipa\_comm.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_comm.h](#)

## 4.41 fipa\_DateTime\_s Struct Reference

```
#include <fipa_acl.h>
```

### Data Fields

- [int year](#)
- [int month](#)
- [int day](#)
- [int hour](#)
- [int minute](#)
- [int second](#)
- [int millisecond](#)
- [char sign](#)
- [int is\\_utc](#)

#### 4.41.1 Detailed Description

Definition at line 234 of file fipa\_acl.h.

#### 4.41.2 Field Documentation

##### 4.41.2.1 int fipa\_DateTime\_s::day

Definition at line 238 of file fipa\_acl.h.

Referenced by fipa\_DateTime\_Compose().

##### 4.41.2.2 int fipa\_DateTime\_s::hour

Definition at line 239 of file fipa\_acl.h.

Referenced by fipa\_DateTime\_Compose().

##### 4.41.2.3 int fipa\_DateTime\_s::is\_utc

Definition at line 244 of file fipa\_acl.h.

##### 4.41.2.4 int fipa\_DateTime\_s::millisecond

Definition at line 242 of file fipa\_acl.h.

Referenced by fipa\_DateTime\_Compose().

##### 4.41.2.5 int fipa\_DateTime\_s::minute

Definition at line 240 of file fipa\_acl.h.

Referenced by fipa\_DateTime\_Compose().

**4.41.2.6 int fipa\_DateTime\_s::month**

Definition at line 237 of file fipa\_acl.h.

Referenced by fipa\_DateTime\_Compose().

**4.41.2.7 int fipa\_DateTime\_s::second**

Definition at line 241 of file fipa\_acl.h.

Referenced by fipa\_DateTime\_Compose().

**4.41.2.8 char fipa\_DateTime\_s::sign**

Definition at line 243 of file fipa\_acl.h.

Referenced by fipa\_DateTime\_Compose(), and fipa\_datetime\_Parse().

**4.41.2.9 int fipa\_DateTime\_s::year**

Definition at line 236 of file fipa\_acl.h.

Referenced by fipa\_DateTime\_Compose().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl.h](#)

## 4.42 fipa\_expression\_s Struct Reference

```
#include <fipa_acl.h>
```

### Data Structures

- union [content\\_u](#)

### Data Fields

- enum [fipa\\_expression\\_type\\_e](#) type
- union [fipa\\_expression\\_s::content\\_u](#) content

#### 4.42.1 Detailed Description

Definition at line 193 of file [fipa\\_acl.h](#).

#### 4.42.2 Field Documentation

##### 4.42.2.1 union [fipa\\_expression\\_s::content\\_u](#) [fipa\\_expression\\_s::content](#)

Referenced by [fipa\\_expression\\_Compose\(\)](#), [fipa\\_expression\\_Copy\(\)](#), [fipa\\_expression\\_Destroy\(\)](#), [MC\\_AclGetConversationID\(\)](#), and [MC\\_AclSetConversationID\(\)](#).

##### 4.42.2.2 enum [fipa\\_expression\\_type\\_e](#) [fipa\\_expression\\_s::type](#)

Definition at line 195 of file [fipa\\_acl.h](#).

Referenced by [fipa\\_expression\\_Compose\(\)](#), [fipa\\_expression\\_Copy\(\)](#), [fipa\\_expression\\_Destroy\(\)](#), [fipa\\_expression\\_Parse\(\)](#), and [MC\\_AclSetConversationID\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl.h](#)

## 4.43 fipa\_list\_t Struct Reference

```
#include <fipa_comm.h>
```

### Data Fields

- void \* [data](#)
- struct [fipa\\_list\\_t](#) \* [next](#)

#### 4.43.1 Detailed Description

Definition at line 136 of file [fipa\\_comm.h](#).

#### 4.43.2 Field Documentation

##### 4.43.2.1 void\* fipa\_list\_t::data

Definition at line 137 of file [fipa\\_comm.h](#).

##### 4.43.2.2 struct fipa\_list\_t\* fipa\_list\_t::next

Definition at line 138 of file [fipa\\_comm.h](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_comm.h](#)

## 4.44 fipa\_message\_string\_s Struct Reference

```
#include <fipa_acl.h>
```

### Data Fields

- char \* [message](#)
- char \* [parse](#)

#### 4.44.1 Detailed Description

Definition at line 143 of file fipa\_acl.h.

#### 4.44.2 Field Documentation

##### 4.44.2.1 char\* fipa\_message\_string\_s::message

Definition at line 145 of file fipa\_acl.h.

Referenced by `acc_connection_Thread()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandleReceived()`, `fipa_message_string_Copy()`, and `fipa_message_string_Destroy()`.

##### 4.44.2.2 char\* fipa\_message\_string\_s::parse

Definition at line 146 of file fipa\_acl.h.

Referenced by `acc_connection_Thread()`, `fipa_agent_identifier_Parse()`, `fipa_CheckNextToken()`, `fipa_datetime_Parse()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandleReceived()`, `fipa_GetAtom()`, `fipa_GetNextWord()`, `fipa_GetWholeToken()`, `fipa_message_string_Copy()`, `fipa_string_Parse()`, and `fipa_word_Parse()`.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl.h](#)

## 4.45 fipa\_number\_s Struct Reference

```
#include <fipa_acl.h>
```

### Data Fields

- char \* [str](#)

### 4.45.1 Detailed Description

Definition at line 264 of file fipa\_acl.h.

### 4.45.2 Field Documentation

#### 4.45.2.1 char\* fipa\_number\_s::str

Definition at line 266 of file fipa\_acl.h.

Referenced by fipa\_number\_Compose(), fipa\_number\_Copy(), and fipa\_number\_Destroy().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl.h](#)



## 4.46 fipa\_string\_s Struct Reference

```
#include <fipa_acl.h>
```

### Data Fields

- char \* [content](#)

### 4.46.1 Detailed Description

Definition at line 223 of file fipa\_acl.h.

### 4.46.2 Field Documentation

#### 4.46.2.1 char\* fipa\_string\_s::content

Definition at line 225 of file fipa\_acl.h.

Referenced by fipa\_string\_Compose(), fipa\_string\_Copy(), fipa\_string\_Destroy(), fipa\_string\_Parse(), MC\_AclGetContent(), MC\_AclGetConversationID(), MC\_AclSetContent(), and MC\_AclSetConversationID().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl.h](#)

## 4.47 fipa\_url\_s Struct Reference

```
#include <fipa_acl.h>
```

### Data Fields

- char \* [str](#)

### 4.47.1 Detailed Description

Definition at line 253 of file fipa\_acl.h.

### 4.47.2 Field Documentation

#### 4.47.2.1 char\* fipa\_url\_s::str

Definition at line 255 of file fipa\_acl.h.

Referenced by `acc_connection_Thread()`, `fipa_envelope_Compose__from()`, `fipa_envelope_Compose__intended_receiver()`, `fipa_envelope_Compose__to()`, `fipa_envelope_HandleReceived()`, `fipa_envelope_ParseAddresses()`, `fipa_url_Compose()`, `fipa_url_Copy()`, `fipa_url_Destroy()`, `MC_AclAddReceiver()`, `MC_AclAddReplyTo()`, `MC_AclGetSender()`, `MC_AclSend()`, and `MC_AclSetSender()`.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl.h](#)

## 4.48 fipa\_url\_sequence\_s Struct Reference

```
#include <fipa_acl.h>
```

### Data Fields

- `int num`
- `struct fipa_url_s ** urls`

### 4.48.1 Detailed Description

Definition at line 154 of file `fipa_acl.h`.

### 4.48.2 Field Documentation

#### 4.48.2.1 `int fipa_url_sequence_s::num`

Definition at line 156 of file `fipa_acl.h`.

Referenced by `fipa_agent_identifier_Compose()`, `fipa_envelope_Compose__from()`, `fipa_envelope_Compose__intended_receiver()`, `fipa_envelope_Compose__to()`, `fipa_envelope_ParseAddresses()`, `fipa_url_sequence_Compose()`, `fipa_url_sequence_Copy()`, `fipa_url_sequence_Destroy()`, `MC_AclAddReceiver()`, `MC_AclAddReplyTo()`, `MC_AclSend()`, and `MC_AclSetSender()`.

#### 4.48.2.2 `struct fipa_url_s** fipa_url_sequence_s::urls`

Definition at line 157 of file `fipa_acl.h`.

Referenced by `acc_connection_Thread()`, `fipa_envelope_Compose__from()`, `fipa_envelope_Compose__intended_receiver()`, `fipa_envelope_Compose__to()`, `fipa_envelope_ParseAddresses()`, `fipa_url_sequence_Compose()`, `fipa_url_sequence_Copy()`, `fipa_url_sequence_Destroy()`, `MC_AclAddReceiver()`, `MC_AclAddReplyTo()`, `MC_AclGetSender()`, `MC_AclSend()`, and `MC_AclSetSender()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/include/fipa_acl.h`

## 4.49 fipa\_word\_s Struct Reference

```
#include <fipa_acl.h>
```

### Data Fields

- char \* [content](#)

### 4.49.1 Detailed Description

Definition at line 212 of file fipa\_acl.h.

### 4.49.2 Field Documentation

#### 4.49.2.1 char\* fipa\_word\_s::content

Definition at line 214 of file fipa\_acl.h.

Referenced by fipa\_agent\_identifier\_Parse(), fipa\_agent\_identifier\_set\_Parse(), fipa\_message\_parameter\_Parse(), fipa\_message\_type\_Parse(), fipa\_protocol\_type\_Parse(), fipa\_url\_Parse(), fipa\_url\_sequence\_Parse(), fipa\_word\_Compose(), fipa\_word\_Copy(), and fipa\_word\_Destroy().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/fipa\\_acl.h](#)

## 4.50 foo\_c Class Reference

### Public Member Functions

- `foo_c` (float *f*, int *b*)
- `~foo_c` ()
- `int get_bar` ()
- `float get_foo` ()
- `void set_bar` (int *b*)
- `void set_foo` (float *f*)
- `void set_foobar` (float *f*, int *b*=0)
- `int get_barfoo` ()

### Protected Member Functions

- `int get_global` ()

### Static Protected Attributes

- static `int global`

### Private Attributes

- `float foo`
- `int bar`
- `int barfoo`

#### 4.50.1 Detailed Description

Definition at line 1 of file class.cxx.

#### 4.50.2 Constructor & Destructor Documentation

##### 4.50.2.1 `foo_c::foo_c` (float *f*, int *b*)

##### 4.50.2.2 `foo_c::~~foo_c` ()

Definition at line 81 of file class.cxx.

#### 4.50.3 Member Function Documentation

##### 4.50.3.1 `int foo_c::get_bar` () `[inline]`

Definition at line 13 of file class.cxx.

References `bar`.

**4.50.3.2 int foo\_c::get\_barfoo () [inline]**

Definition at line 67 of file class.cxx.

References barfoo.

**4.50.3.3 float foo\_c::get\_foo () [inline]**

Definition at line 20 of file class.cxx.

References foo.

**4.50.3.4 int foo\_c::get\_global () [inline, protected]**

Definition at line 54 of file class.cxx.

References global.

**4.50.3.5 void foo\_c::set\_bar (int *b*) [inline]**

Definition at line 27 of file class.cxx.

References bar.

**4.50.3.6 void foo\_c::set\_foo (float *f*) [inline]**

Definition at line 34 of file class.cxx.

References foo.

**4.50.3.7 void foo\_c::set\_foobar (float *f*, int *b* = 0) [inline]**

Definition at line 41 of file class.cxx.

References bar, and foo.

**4.50.4 Field Documentation****4.50.4.1 int foo\_c::bar [private]**

Definition at line 4 of file class.cxx.

Referenced by get\_bar(), set\_bar(), and set\_foobar().

**4.50.4.2 int foo\_c::barfoo [private]**

Definition at line 61 of file class.cxx.

Referenced by get\_barfoo().

#### 4.50.4.3 float foo\_c::foo [private]

Definition at line 3 of file class.cxx.

Referenced by get\_foo(), set\_foo(), and set\_foobar().

#### 4.50.4.4 int foo\_c::global [static, protected]

Definition at line 50 of file class.cxx.

Referenced by get\_global().

The documentation for this class was generated from the following file:

- /home/dko/Projects/mobilec/src/mxml-2.2.2/test/[class.cxx](#)

## 4.51 foo\_s Struct Reference

### Public Member Functions

- [foo\\_s](#) (float [f](#), int [b](#))
- [~foo\\_s](#) ()
- int [get\\_bar](#) ()
- float [get\\_foo](#) ()
- void [set\\_bar](#) (int [b](#))
- void [set\\_foo](#) (float [f](#))

### Data Fields

- float [foo](#)
- int [bar](#)

#### 4.51.1 Detailed Description

Definition at line 1 of file struct.cxx.

#### 4.51.2 Constructor & Destructor Documentation

##### 4.51.2.1 foo\_s::foo\_s (float *f*, int *b*)

Definition at line 39 of file struct.cxx.

References [bar](#), and [foo](#).

##### 4.51.2.2 foo\_s::~~foo\_s ()

Definition at line 46 of file struct.cxx.

#### 4.51.3 Member Function Documentation

##### 4.51.3.1 int foo\_s::get\_bar () [inline]

Definition at line 11 of file struct.cxx.

References [bar](#).

##### 4.51.3.2 float foo\_s::get\_foo () [inline]

Definition at line 18 of file struct.cxx.

References [foo](#).



**4.51.3.3 void foo\_s::set\_bar (int *b*) [inline]**

Definition at line 25 of file struct.cxx.

References bar.

**4.51.3.4 void foo\_s::set\_foo (float *f*) [inline]**

Definition at line 32 of file struct.cxx.

References foo.

**4.51.4 Field Documentation****4.51.4.1 int foo\_s::bar**

Definition at line 4 of file struct.cxx.

Referenced by foo\_s(), get\_bar(), and set\_bar().

**4.51.4.2 float foo\_s::foo**

Definition at line 3 of file struct.cxx.

Referenced by foo\_s(), get\_foo(), and set\_foo().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/mxml-2.2.2/test/[struct.cxx](#)

## 4.52 havege\_state Struct Reference

HAVEGE state structure.

```
#include <havege.h>
```

### Data Fields

- [int PT1](#)
- [int PT2](#)
- [int offset](#) [2]
- [int pool](#) [COLLECT\_SIZE]
- [int WALK](#) [8192]

### 4.52.1 Detailed Description

HAVEGE state structure.

Definition at line 12 of file havege.h.

### 4.52.2 Field Documentation

#### 4.52.2.1 int havege\_state::offset[2]

Definition at line 14 of file havege.h.

Referenced by `havege_fill()`, and `havege_rand()`.

#### 4.52.2.2 int havege\_state::pool[COLLECT\_SIZE]

Definition at line 15 of file havege.h.

Referenced by `havege_rand()`.

#### 4.52.2.3 int havege\_state::PT1

Definition at line 14 of file havege.h.

Referenced by `havege_fill()`.

#### 4.52.2.4 int havege\_state::PT2

Definition at line 14 of file havege.h.

Referenced by `havege_fill()`.

#### 4.52.2.5 int havege\_state::WALK[8192]

Definition at line 16 of file havege.h.

Referenced by `havege_fill()`.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/havege.h](#)

## 4.53 host\_id\_s Struct Reference

```
#include <host_id.h>
```

### Data Fields

- char \* [hostname](#)
- [int](#) port

### 4.53.1 Detailed Description

Definition at line 38 of file `host_id.h`.

### 4.53.2 Field Documentation

#### 4.53.2.1 char\* host\_id\_s::hostname

Definition at line 40 of file `host_id.h`.

#### 4.53.2.2 int host\_id\_s::port

Definition at line 41 of file `host_id.h`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/include/host\_id.h`

## 4.54 hr\_time Struct Reference

timer structure

```
#include <timing.h>
```

### Data Fields

- unsigned char [opaque](#) [32]

#### 4.54.1 Detailed Description

timer structure

Definition at line 10 of file `timing.h`.

#### 4.54.2 Field Documentation

##### 4.54.2.1 unsigned char hr\_time::opaque[32]

Definition at line 12 of file `timing.h`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/timing.h`

## 4.55 interpreter\_variable\_data\_s Struct Reference

```
#include <interpreter_variable_data.h>
```

### Data Fields

- char \* [name](#)
- int [size](#)
- ChType\_t [data\\_type](#)
- int [array\\_dim](#)
- int \* [array\\_extent](#)
- void \* [data](#)

### 4.55.1 Detailed Description

Definition at line 41 of file `interpreter_variable_data.h`.

### 4.55.2 Field Documentation

#### 4.55.2.1 int interpreter\_variable\_data\_s::array\_dim

Definition at line 45 of file `interpreter_variable_data.h`.

Referenced by `agent_AddPersistentVariable()`, `agent_xml_compose__data()`, `agent_xml_compose__row()`, `agent_xml_parse__data()`, `agent_xml_parse__row()`, `interpreter_variable_data_Copy()`, `interpreter_variable_data_Initialize()`, `interpreter_variable_data_InitializeFromAgent()`, `interpreter_variable_data_New()`, `MC_AgentReturnArrayDim()`, `MC_AgentReturnArrayExtent()`, `MC_AgentReturnArrayNum()`, `MC_AgentReturnIsArray()`, `MC_AgentVariableRetrieveInfo()`, and `MC_GetAgentReturnData()`.

#### 4.55.2.2 int\* interpreter\_variable\_data\_s::array\_extent

Definition at line 46 of file `interpreter_variable_data.h`.

Referenced by `agent_AddPersistentVariable()`, `agent_xml_compose__row()`, `agent_xml_parse__row()`, `interpreter_variable_data_Copy()`, `interpreter_variable_data_Destroy()`, `interpreter_variable_data_Initialize()`, `interpreter_variable_data_InitializeFromAgent()`, `interpreter_variable_data_New()`, `MC_AgentReturnArrayExtent()`, `MC_AgentVariableRetrieveInfo()`, and `MC_GetAgentReturnData()`.

#### 4.55.2.3 void\* interpreter\_variable\_data\_s::data

Definition at line 47 of file `interpreter_variable_data.h`.

Referenced by `agent_AddPersistentVariable()`, `agent_xml_compose__data()`, `agent_xml_compose__row()`, `agent_xml_parse__data()`, `agent_xml_parse__row()`, `interpreter_variable_data_Copy()`, `interpreter_variable_data_Destroy()`, `interpreter_variable_data_Initialize()`, `interpreter_variable_data_InitializeFromAgent()`, `interpreter_variable_data_New()`, `MC_AgentReturnDataGetSymbolAddr()`, `MC_AgentVariableRetrieve()`, `MC_AgentVariableRetrieveInfo()`, and `MC_SaveData_chdl()`.

#### 4.55.2.4 ChType\_t interpreter\_variable\_data\_s::data\_type

Definition at line 44 of file interpreter\_variable\_data.h.

Referenced by agent\_AddPersistentVariable(), agent\_xml\_compose\_\_data(), agent\_xml\_compose\_\_row(), agent\_xml\_parse\_\_data(), agent\_xml\_parse\_\_row(), interpreter\_variable\_data\_Copy(), interpreter\_variable\_data\_Initialize(), interpreter\_variable\_data\_InitializeFromAgent(), interpreter\_variable\_data\_New(), MC\_AgentReturnDataSize(), MC\_AgentReturnDataType(), MC\_GetAgentReturnData(), and MC\_SaveData\_chdl().

#### 4.55.2.5 char\* interpreter\_variable\_data\_s::name

Definition at line 42 of file interpreter\_variable\_data.h.

Referenced by agent\_AddPersistentVariable(), agent\_xml\_compose\_\_data(), agent\_xml\_parse\_\_data(), interpreter\_variable\_data\_Copy(), interpreter\_variable\_data\_Destroy(), interpreter\_variable\_data\_Initialize(), interpreter\_variable\_data\_InitializeFromAgent(), interpreter\_variable\_data\_New(), and MC\_SaveData\_chdl().

#### 4.55.2.6 int interpreter\_variable\_data\_s::size

Definition at line 43 of file interpreter\_variable\_data.h.

Referenced by agent\_AddPersistentVariable(), interpreter\_variable\_data\_Copy(), interpreter\_variable\_data\_Initialize(), interpreter\_variable\_data\_InitializeFromAgent(), interpreter\_variable\_data\_New(), and MC\_SaveData\_chdl().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/interpreter\\_variable\\_data.h](#)

## 4.56 list\_s Struct Reference

```
#include <list.h>
```

### Data Fields

- [listNode\\_p listhead](#)
- [int size](#)

### 4.56.1 Detailed Description

Definition at line 41 of file list.h.

### 4.56.2 Field Documentation

#### 4.56.2.1 listNode\_p list\_s::listhead

Definition at line 43 of file list.h.

Referenced by `barrier_queue_Add()`, `barrier_queue_Get()`, `ListAdd()`, `ListDelete()`, `ListGetHead()`, `ListInitialize()`, `ListPop()`, `ListSearch()`, `ListTerminate()`, `syncListAddNode()`, and `syncListFind()`.

#### 4.56.2.2 int list\_s::size

Definition at line 44 of file list.h.

Referenced by `barrier_queue_Delete()`, `list_pGetSize()`, `ListAdd()`, `ListDelete()`, `ListInitialize()`, `ListPop()`, `ListSearch()`, `ListTerminate()`, `syncListDelete()`, `syncListDestroy()`, and `syncListRemove()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/mc_list/list.h`



## 4.57 listNode\_s Struct Reference

```
#include <list.h>
```

### Data Fields

- DATA [node\\_data](#)
- struct [listNode\\_s](#) \* [next](#)

### 4.57.1 Detailed Description

Definition at line 32 of file list.h.

### 4.57.2 Field Documentation

#### 4.57.2.1 struct listNode\_s\* listNode\_s::next

Definition at line 35 of file list.h.

Referenced by `AP_QUEUE_STD_DEFN_TEMPLATE()`, `barrier_queue_Add()`, `barrier_queue_Get()`, `df_SearchForService()`, `ListAdd()`, `ListDelete()`, `ListPop()`, `ListSearch()`, `request_handler_DEREGISTER()`, `syncListAddNode()`, and `syncListFind()`.

#### 4.57.2.2 DATA listNode\_s::node\_data

Definition at line 34 of file list.h.

Referenced by `AP_QUEUE_STD_DEFN_TEMPLATE()`, `barrier_queue_Add()`, `barrier_queue_Get()`, `df_SearchForService()`, `ListAdd()`, `ListDelete()`, `ListGetHead()`, `ListPop()`, `ListSearch()`, `request_handler_DEREGISTER()`, `syncListAddNode()`, and `syncListFind()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/mc_list/list.h`

## 4.58 mc\_platform\_s Struct Reference

```
#include <mc_platform.h>
```

### Data Fields

- [int](#) `err`
- `char *` [hostname](#)
- unsigned short [port](#)
- [int](#) `initInterps`
- `message_queue_p` [asm\\_message\\_queue](#)
- `message_queue_p` [message\\_queue](#)
- `agent_queue_p` [agent\\_queue](#)
- `connection_queue_p` [connection\\_queue](#)
- `df_p` [df](#)
- `ams_p` [ams](#)
- `acc_p` [acc](#)
- `cmd_prompt_p` [cmd\\_prompt](#)
- `syncList_p` [syncList](#)
- `barrier_queue_p` [barrier\\_queue](#)
- `listen_thread_arg_p` [listen\\_thread\\_arg](#)
- `listen_thread_arg_p` [client\\_thread\\_arg](#)
- [int](#) `default_agentstatus`
- [int](#) `stack_size` [MC\_THREAD\_ALL]
- `ChOptions_t *` [interp\\_options](#)
- `COND_T *` [MC\\_signal\\_cond](#)
- `COND_T *` [MC\\_sync\\_cond](#)
- `MUTEX_T *` [MC\\_signal\\_lock](#)
- `MUTEX_T *` [MC\\_sync\\_lock](#)
- `enum` [MC\\_Signal\\_e](#) [MC\\_signal](#)
- `enum` [MC\\_SteerCommand\\_e](#) [MC\\_steer\\_command](#)
- `MUTEX_T *` [MC\\_steer\\_lock](#)
- `COND_T *` [MC\\_steer\\_cond](#)
- [int](#) `giant`
- `MUTEX_T *` [giant\\_lock](#)
- `COND_T *` [giant\\_cond](#)
- [int](#) `quit`
- `MUTEX_T *` [quit\\_lock](#)
- `COND_T *` [quit\\_cond](#)
- [int](#) `sockfd`
- `char` [private\\_key](#) [1210]
- [MCAgency\\_t](#) [agency](#)
- `interpreter_queue_p` [interpreter\\_queue](#)
- [int](#) `bluetooth`

### 4.58.1 Detailed Description

Definition at line 49 of file `mc_platform.h`.

## 4.58.2 Field Documentation

### 4.58.2.1 acc\_p mc\_platform\_s::acc

Definition at line 72 of file mc\_platform.h.

Referenced by acc\_connection\_Thread(), acc\_MessageHandlerThread(), acc\_Start(), acc\_Thread(), listen\_Thread(), MC\_End(), mc\_platform\_Destroy(), mc\_platform\_Initialize(), and message\_Send().

### 4.58.2.2 MCAgency\_t mc\_platform\_s::agency

Definition at line 114 of file mc\_platform.h.

Referenced by agent\_RunChScriptThread(), handler\_COMPOSE\_SEND(), listen\_Thread(), mc\_platform\_Initialize(), and message\_send\_Thread().

### 4.58.2.3 agent\_queue\_p mc\_platform\_s::agent\_queue

Definition at line 67 of file mc\_platform.h.

Referenced by acc\_connection\_Thread(), acc\_MessageHandlerThread(), handler\_FLUSH\_AGENTS(), handler\_PRINTLIST\_AGENTS(), MC\_AddAgent(), MC\_AddStationaryAgent(), MC\_FindAgentByID(), MC\_FindAgentByName(), MC\_GetAllAgents(), mc\_platform\_Destroy(), mc\_platform\_Initialize(), MC\_RetrieveAgent(), MC\_SendAgent(), MC\_SendAgentFile(), MC\_SendAgentMigrationMessageFile(), MC\_WaitAgent(), and MC\_WaitRetrieveAgent().

### 4.58.2.4 ams\_p mc\_platform\_s::ams

Definition at line 71 of file mc\_platform.h.

Referenced by acc\_MessageHandlerThread(), agent\_RunChScriptThread(), ams\_Start(), ams\_Thread(), MC\_AddAgent(), MC\_End(), mc\_platform\_Destroy(), mc\_platform\_Initialize(), MC\_SendAgent(), MC\_SendAgentFile(), MC\_SendAgentMigrationMessageFile(), and MC\_SetAgentStatus().

### 4.58.2.5 message\_queue\_p mc\_platform\_s::asm\_message\_queue

Definition at line 62 of file mc\_platform.h.

### 4.58.2.6 barrier\_queue\_p mc\_platform\_s::barrier\_queue

Definition at line 79 of file mc\_platform.h.

Referenced by MC\_Barrier(), MC\_BarrierDelete(), MC\_BarrierInit(), mc\_platform\_Destroy(), and mc\_platform\_Initialize().

### 4.58.2.7 int mc\_platform\_s::bluetooth

Definition at line 120 of file mc\_platform.h.

Referenced by listen\_Thread(), mc\_platform\_Initialize(), message\_InitializeFromAgent(), and message\_send\_Thread().

#### **4.58.2.8 listen\_thread\_arg\_p mc\_platform\_s::client\_thread\_arg**

Definition at line 82 of file mc\_platform.h.

#### **4.58.2.9 cmd\_prompt\_p mc\_platform\_s::cmd\_prompt**

Definition at line 73 of file mc\_platform.h.

Referenced by cmd\_prompt\_Start(), MC\_End(), mc\_platform\_Destroy(), and mc\_platform\_Initialize().

#### **4.58.2.10 connection\_queue\_p mc\_platform\_s::connection\_queue**

Definition at line 68 of file mc\_platform.h.

Referenced by acc\_Thread(), handler\_PRINT\_CONNECTLIST(), listen\_Thread(), MC\_End(), mc\_platform\_Destroy(), and mc\_platform\_Initialize().

#### **4.58.2.11 int mc\_platform\_s::default\_agentstatus**

Definition at line 84 of file mc\_platform.h.

Referenced by agent\_Initialize(), mc\_platform\_Initialize(), and MC\_SetDefaultAgentStatus().

#### **4.58.2.12 df\_p mc\_platform\_s::df**

Definition at line 70 of file mc\_platform.h.

Referenced by df\_ProcessRequest(), df\_Start(), df\_Thread(), MC\_DeregisterService(), MC\_End(), mc\_platform\_Destroy(), mc\_platform\_Initialize(), MC\_RegisterService(), MC\_SearchForService(), request\_handler\_DEREGISTER(), request\_handler\_REGISTER(), and request\_handler\_SEARCH().

#### **4.58.2.13 int mc\_platform\_s::err**

Definition at line 52 of file mc\_platform.h.

Referenced by agent\_Initialize(), mc\_platform\_Initialize(), message\_InitializeFromAgent(), and message\_InitializeFromString().

#### **4.58.2.14 int mc\_platform\_s::giant**

Definition at line 104 of file mc\_platform.h.

Referenced by acc\_MessageHandlerThread(), acc\_Thread(), MC\_GetAllAgents(), MC\_HaltAgency(), mc\_platform\_Initialize(), MC\_ResetSignal(), MC\_ResumeAgency(), and MC\_WaitSignal().

#### **4.58.2.15 COND\_T\* mc\_platform\_s::giant\_cond**

Definition at line 106 of file mc\_platform.h.

Referenced by acc\_MessageHandlerThread(), acc\_Thread(), mc\_platform\_Destroy(), mc\_platform\_Initialize(), and MC\_ResetSignal().

**4.58.2.16 MUTEX\_T\* mc\_platform\_s::giant\_lock**

Definition at line 105 of file mc\_platform.h.

Referenced by acc\_MessageHandlerThread(), acc\_Thread(), MC\_GetAllAgents(), MC\_HaltAgency(), mc\_platform\_Destroy(), mc\_platform\_Initialize(), MC\_ResetSignal(), MC\_ResumeAgency(), and MC\_WaitSignal().

**4.58.2.17 char\* mc\_platform\_s::hostname**

Definition at line 57 of file mc\_platform.h.

Referenced by agent\_Initialize(), agent\_NewBinary(), fipa\_envelope\_Compose\_\_from(), handler\_COMPOSE\_SEND(), mc\_platform\_Destroy(), mc\_platform\_Initialize(), message\_InitializeFromAgent(), message\_InitializeFromString(), message\_queue\_SendOutgoing(), and udplisten\_Thread().

**4.58.2.18 int mc\_platform\_s::initInterps**

Definition at line 59 of file mc\_platform.h.

Referenced by mc\_platform\_Initialize().

**4.58.2.19 ChOptions\_t\* mc\_platform\_s::interp\_options**

Definition at line 88 of file mc\_platform.h.

Referenced by agent\_RunChScriptThread(), mc\_platform\_Destroy(), and mc\_platform\_Initialize().

**4.58.2.20 interpreter\_queue\_p mc\_platform\_s::interpreter\_queue**

Definition at line 117 of file mc\_platform.h.

Referenced by agent\_Destroy(), agent\_RunChScriptThread(), mc\_platform\_Destroy(), and mc\_platform\_Initialize().

**4.58.2.21 listen\_thread\_arg\_p mc\_platform\_s::listen\_thread\_arg**

Definition at line 81 of file mc\_platform.h.

**4.58.2.22 enum MC\_Signal\_e mc\_platform\_s::MC\_signal**

Definition at line 95 of file mc\_platform.h.

Referenced by acc\_MessageHandlerThread(), acc\_Thread(), agent\_RunChScriptThread(), mc\_platform\_Initialize(), MC\_ResetSignal(), and MC\_WaitSignal().

**4.58.2.23 COND\_T\* mc\_platform\_s::MC\_signal\_cond**

Definition at line 91 of file mc\_platform.h.

Referenced by acc\_MessageHandlerThread(), acc\_Thread(), agent\_RunChScriptThread(), mc\_platform\_Destroy(), mc\_platform\_Initialize(), and MC\_WaitSignal().

**4.58.2.24    `MUTEX_T* mc_platform_s::MC_signal_lock`**

Definition at line 93 of file `mc_platform.h`.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `agent_RunChScriptThread()`, `mc_platform_Destroy()`, `mc_platform_Initialize()`, and `MC_WaitSignal()`.

**4.58.2.25    `enum MC_SteerCommand_e mc_platform_s::MC_steer_command`**

Definition at line 98 of file `mc_platform.h`.

Referenced by `MC_SendSteerCommand()`, `MC_Steer()`, and `MC_SteerControl()`.

**4.58.2.26    `COND_T* mc_platform_s::MC_steer_cond`**

Definition at line 100 of file `mc_platform.h`.

Referenced by `mc_platform_Destroy()`, `mc_platform_Initialize()`, `MC_SendSteerCommand()`, and `MC_SteerControl()`.

**4.58.2.27    `MUTEX_T* mc_platform_s::MC_steer_lock`**

Definition at line 99 of file `mc_platform.h`.

Referenced by `mc_platform_Destroy()`, `mc_platform_Initialize()`, `MC_SendSteerCommand()`, `MC_Steer()`, and `MC_SteerControl()`.

**4.58.2.28    `COND_T* mc_platform_s::MC_sync_cond`**

Definition at line 92 of file `mc_platform.h`.

Referenced by `mc_platform_Destroy()`, and `mc_platform_Initialize()`.

**4.58.2.29    `MUTEX_T* mc_platform_s::MC_sync_lock`**

Definition at line 94 of file `mc_platform.h`.

Referenced by `mc_platform_Destroy()`, and `mc_platform_Initialize()`.

**4.58.2.30    `message_queue_p mc_platform_s::message_queue`**

Definition at line 66 of file `mc_platform.h`.

Referenced by `acc_connection_Thread()`, `acc_MessageHandlerThread()`, `handler_PRINTLIST_MESSAGE()`, `MC_End()`, `MC_LoadAgentFromFile()`, `mc_platform_Destroy()`, `mc_platform_Initialize()`, and `MC_SendAgentMigrationMessage()`.

**4.58.2.31    `unsigned short mc_platform_s::port`**

Definition at line 58 of file `mc_platform.h`.

Referenced by `acc_connection_Thread()`, `agent_Initialize()`, `agent_NewBinary()`, `fipa_envelope_Compose__from()`, `listen_Thread()`, `mc_platform_Initialize()`, `message_InitializeFromAgent()`, `message_InitializeFromString()`, `message_queue_SendOutgoing()`, and `udplisten_Thread()`.

#### 4.58.2.32 `char mc_platform_s::private_key[1210]`

Definition at line 112 of file `mc_platform.h`.

Referenced by `acc_connection_Thread()`, `listen_Thread()`, and `MC_AclSend()`.

#### 4.58.2.33 `int mc_platform_s::quit`

Definition at line 108 of file `mc_platform.h`.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `ams_ManageAgentList()`, `ams_Thread()`, `df_Thread()`, `handler_QUIT()`, `listen_Thread()`, `MC_End()`, `MC_MainLoop()`, and `mc_platform_Initialize()`.

#### 4.58.2.34 `COND_T* mc_platform_s::quit_cond`

Definition at line 110 of file `mc_platform.h`.

Referenced by `handler_QUIT()`, `MC_End()`, `MC_MainLoop()`, `mc_platform_Destroy()`, and `mc_platform_Initialize()`.

#### 4.58.2.35 `MUTEX_T* mc_platform_s::quit_lock`

Definition at line 109 of file `mc_platform.h`.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `ams_ManageAgentList()`, `ams_Thread()`, `df_Thread()`, `handler_QUIT()`, `MC_End()`, `MC_MainLoop()`, `mc_platform_Destroy()`, and `mc_platform_Initialize()`.

#### 4.58.2.36 `int mc_platform_s::sockfd`

Definition at line 111 of file `mc_platform.h`.

Referenced by `listen_Thread()`, and `mc_platform_Destroy()`.

#### 4.58.2.37 `int mc_platform_s::stack_size[MC_THREAD_ALL]`

Definition at line 86 of file `mc_platform.h`.

Referenced by `acc_Start()`, `agent_RunChScript()`, `ams_Start()`, `cmd_prompt_Start()`, `df_Start()`, and `mc_platform_Initialize()`.

#### 4.58.2.38 `syncList_p mc_platform_s::syncList`

Definition at line 78 of file `mc_platform.h`.

Referenced by `MC_CondBroadcast()`, `MC_CondReset()`, `MC_CondSignal()`, `MC_CondWait()`, `MC_MutexLock()`, `MC_MutexUnlock()`, `mc_platform_Destroy()`, `mc_platform_Initialize()`, `MC_SemaphorePost()`, `MC_SemaphoreWait()`, `MC_SyncDelete()`, and `MC_SyncInit()`.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/mc\\_platform.h](#)



## 4.59 mc\_rwlock\_s Struct Reference

```
#include <mc_rwlock.h>
```

### Data Fields

- [int num\\_readers](#)
- [int write\\_flag](#)
- [int write\\_request](#)
- [MUTEX\\_T \\* lock](#)
- [COND\\_T \\* cond](#)

### 4.59.1 Detailed Description

Definition at line 39 of file mc\_rwlock.h.

### 4.59.2 Field Documentation

#### 4.59.2.1 COND\_T\* mc\_rwlock\_s::cond

Definition at line 44 of file mc\_rwlock.h.

Referenced by [mc\\_rwlock\\_destroy\(\)](#), [mc\\_rwlock\\_init\(\)](#), [mc\\_rwlock\\_rdlock\(\)](#), [mc\\_rwlock\\_rdunlock\(\)](#), [mc\\_rwlock\\_wrlock\(\)](#), and [mc\\_rwlock\\_wrunlock\(\)](#).

#### 4.59.2.2 MUTEX\_T\* mc\_rwlock\_s::lock

Definition at line 43 of file mc\_rwlock.h.

Referenced by [mc\\_rwlock\\_destroy\(\)](#), [mc\\_rwlock\\_init\(\)](#), [mc\\_rwlock\\_rdlock\(\)](#), [mc\\_rwlock\\_rdunlock\(\)](#), [mc\\_rwlock\\_wrlock\(\)](#), and [mc\\_rwlock\\_wrunlock\(\)](#).

#### 4.59.2.3 int mc\_rwlock\_s::num\_readers

Definition at line 40 of file mc\_rwlock.h.

Referenced by [mc\\_rwlock\\_init\(\)](#), [mc\\_rwlock\\_rdlock\(\)](#), [mc\\_rwlock\\_rdunlock\(\)](#), and [mc\\_rwlock\\_wrlock\(\)](#).

#### 4.59.2.4 int mc\_rwlock\_s::write\_flag

Definition at line 41 of file mc\_rwlock.h.

Referenced by [mc\\_rwlock\\_init\(\)](#), [mc\\_rwlock\\_rdlock\(\)](#), [mc\\_rwlock\\_wrlock\(\)](#), and [mc\\_rwlock\\_wrunlock\(\)](#).

#### 4.59.2.5 int mc\_rwlock\_s::write\_request

Definition at line 42 of file mc\_rwlock.h.

Referenced by [mc\\_rwlock\\_init\(\)](#), [mc\\_rwlock\\_rdlock\(\)](#), and [mc\\_rwlock\\_wrlock\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/mc\\_rwlock.h](#)

## 4.60 MCAgencyOptions\_s Struct Reference

User modifiable agency options.

```
#include <libmc.h>
```

### Data Fields

- [int threads](#)
- [int default\\_agent\\_status](#)
- [int modified](#)
- [int enable\\_security](#)
- unsigned char [passphrase](#) [32]
- [int stack\\_size](#) [MC\_THREAD\_ALL]
- char \* [known\\_host\\_filename](#)
- char \* [priv\\_key\\_filename](#)
- [int initInterps](#)
- [int bluetooth](#)
- ChOptions\_t \* [ch\\_options](#)

### 4.60.1 Detailed Description

User modifiable agency options.

Definition at line 254 of file libmc.h.

### 4.60.2 Field Documentation

#### 4.60.2.1 int MCAgencyOptions\_s::bluetooth

Definition at line 267 of file libmc.h.

Referenced by MC\_Initialize(), and MC\_InitializeAgencyOptions().

#### 4.60.2.2 ChOptions\_t\* MCAgencyOptions\_s::ch\_options

Definition at line 268 of file libmc.h.

Referenced by MC\_Initialize().

#### 4.60.2.3 int MCAgencyOptions\_s::default\_agent\_status

Default agent status

Definition at line 256 of file libmc.h.

Referenced by MC\_Initialize(), and MC\_InitializeAgencyOptions().

#### 4.60.2.4 int MCAgencyOptions\_s::enable\_security

security enable flag

Definition at line 258 of file libmc.h.

#### 4.60.2.5 int MCAgencyOptions\_s::initInterps

Definition at line 266 of file libmc.h.

Referenced by MC\_Initialize(), and MC\_InitializeAgencyOptions().

#### 4.60.2.6 char\* MCAgencyOptions\_s::known\_host\_filename

Definition at line 264 of file libmc.h.

Referenced by MC\_Initialize(), and MC\_InitializeAgencyOptions().

#### 4.60.2.7 int MCAgencyOptions\_s::modified

unused

Definition at line 257 of file libmc.h.

Referenced by MC\_InitializeAgencyOptions().

#### 4.60.2.8 unsigned char MCAgencyOptions\_s::passphrase[32]

security enable flag

Definition at line 259 of file libmc.h.

Referenced by MC\_Initialize(), and MC\_InitializeAgencyOptions().

#### 4.60.2.9 char\* MCAgencyOptions\_s::priv\_key\_filename

Definition at line 265 of file libmc.h.

Referenced by MC\_Initialize(), and MC\_InitializeAgencyOptions().

#### 4.60.2.10 int MCAgencyOptions\_s::stack\_size[MC\_THREAD\_ALL]

If the stack size is -1, use system def.

Definition at line 262 of file libmc.h.

Referenced by MC\_Initialize(), and MC\_InitializeAgencyOptions().

#### 4.60.2.11 int MCAgencyOptions\_s::threads

Threads to start

Definition at line 255 of file libmc.h.

Referenced by MC\_Initialize(), MC\_InitializeAgencyOptions(), MC\_SetThreadOff(), MC\_SetThreadOn(), MC\_SetThreadsAllOff(), and MC\_SetThreadsAllOn().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/include/[libmc.h](#)

## 4.61 md2\_context Struct Reference

MD2 context structure.

```
#include <md2.h>
```

### Data Fields

- unsigned char [cksum](#) [16]
- unsigned char [state](#) [48]
- unsigned char [buffer](#) [16]
- unsigned char [ipad](#) [64]
- unsigned char [opad](#) [64]
- [int left](#)

#### 4.61.1 Detailed Description

MD2 context structure.

Definition at line 10 of file md2.h.

#### 4.61.2 Field Documentation

##### 4.61.2.1 unsigned char md2\_context::buffer[16]

data block being processed

Definition at line 14 of file md2.h.

##### 4.61.2.2 unsigned char md2\_context::cksum[16]

checksum of the data block

Definition at line 12 of file md2.h.

##### 4.61.2.3 unsigned char md2\_context::ipad[64]

HMAC: inner padding

Definition at line 16 of file md2.h.

##### 4.61.2.4 int md2\_context::left

amount of data in buffer

Definition at line 18 of file md2.h.

##### 4.61.2.5 unsigned char md2\_context::opad[64]

HMAC: outer padding

Definition at line 17 of file md2.h.

#### 4.61.2.6 unsigned char md2\_context::state[48]

intermediate digest state

Definition at line 13 of file md2.h.

The documentation for this struct was generated from the following file:

- </home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md2.h>

## 4.62 md4\_context Struct Reference

MD4 context structure.

```
#include <md4.h>
```

### Data Fields

- unsigned long [total](#) [2]
- unsigned long [state](#) [4]
- unsigned char [buffer](#) [64]
- unsigned char [ipad](#) [64]
- unsigned char [opad](#) [64]

### 4.62.1 Detailed Description

MD4 context structure.

Definition at line 10 of file md4.h.

### 4.62.2 Field Documentation

#### 4.62.2.1 unsigned char md4\_context::buffer[64]

data block being processed

Definition at line 14 of file md4.h.

#### 4.62.2.2 unsigned char md4\_context::ipad[64]

HMAC: inner padding

Definition at line 16 of file md4.h.

#### 4.62.2.3 unsigned char md4\_context::opad[64]

HMAC: outer padding

Definition at line 17 of file md4.h.

#### 4.62.2.4 unsigned long md4\_context::state[4]

intermediate digest state

Definition at line 13 of file md4.h.

#### 4.62.2.5 unsigned long md4\_context::total[2]

number of bytes processed

Definition at line 12 of file md4.h.

The documentation for this struct was generated from the following file:

- </home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md4.h>



## 4.63 md5\_context Struct Reference

MD5 context structure.

```
#include <md5.h>
```

### Data Fields

- unsigned long [total](#) [2]
- unsigned long [state](#) [4]
- unsigned char [buffer](#) [64]
- unsigned char [ipad](#) [64]
- unsigned char [opad](#) [64]

### 4.63.1 Detailed Description

MD5 context structure.

Definition at line 10 of file md5.h.

### 4.63.2 Field Documentation

#### 4.63.2.1 unsigned char md5\_context::buffer[64]

data block being processed

Definition at line 14 of file md5.h.

Referenced by md5\_update().

#### 4.63.2.2 unsigned char md5\_context::ipad[64]

HMAC: inner padding

Definition at line 16 of file md5.h.

Referenced by md5\_hmac\_starts().

#### 4.63.2.3 unsigned char md5\_context::opad[64]

HMAC: outer padding

Definition at line 17 of file md5.h.

Referenced by md5\_hmac\_finish(), and md5\_hmac\_starts().

#### 4.63.2.4 unsigned long md5\_context::state[4]

intermediate digest state

Definition at line 13 of file md5.h.

Referenced by md5\_finish(), md5\_process(), md5\_starts(), and ssl\_calc\_finished().

**4.63.2.5 unsigned long md5\_context::total[2]**

number of bytes processed

Definition at line 12 of file md5.h.

Referenced by md5\_finish(), md5\_starts(), and md5\_update().

The documentation for this struct was generated from the following file:

- </home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md5.h>

## 4.64 message\_s Struct Reference

```
#include <message.h>
```

### Data Fields

- struct sockaddr\_in \* [addr](#)
- int [connect\\_id](#)
- int [message\\_id](#)
- int [isHTTP](#)
- enum [message\\_type\\_e](#) [message\\_type](#)
- enum [http\\_performative\\_e](#) [http\\_type](#)
- mxml\_node\_t \* [xml\\_root](#)
- mxml\_node\_t \* [xml\\_payload](#)
- char \* [message\\_body](#)
- char \* [update\\_name](#)
- int [update\\_num](#)
- char \* [from\\_address](#)
- char \* [to\\_address](#)
- char \* [target](#)
- int [agent\\_xml\\_flag](#)
- char \* [sending\\_agent\\_name](#)

### 4.64.1 Detailed Description

Definition at line 78 of file message.h.

### 4.64.2 Field Documentation

#### 4.64.2.1 struct sockaddr\_in\* message\_s::addr

Definition at line 80 of file message.h.

Referenced by [message\\_Destroy\(\)](#), [message\\_InitializeFromAgent\(\)](#), [message\\_InitializeFromConnection\(\)](#), [message\\_InitializeFromString\(\)](#), [message\\_New\(\)](#), and [message\\_xml\\_parse\\_\\_message\(\)](#).

#### 4.64.2.2 int message\_s::agent\_xml\_flag

Definition at line 112 of file message.h.

Referenced by [agent\\_Initialize\(\)](#), [message\\_Destroy\(\)](#), [message\\_InitializeFromAgent\(\)](#), and [message\\_New\(\)](#).

#### 4.64.2.3 int message\_s::connect\_id

Definition at line 83 of file message.h.

Referenced by [message\\_InitializeFromConnection\(\)](#), [message\\_InitializeFromString\(\)](#), and [message\\_New\(\)](#).

#### 4.64.2.4 `char* message_s::from_address`

Definition at line 102 of file message.h.

Referenced by `AP_QUEUE_SEARCH_TEMPLATE()`, `message_Destroy()`, `message_InitializeFromAgent()`, `message_InitializeFromConnection()`, `message_InitializeFromString()`, `message_New()`, and `message_xml_parse__message()`.

#### 4.64.2.5 `enum http_performative_e message_s::http_type`

Definition at line 91 of file message.h.

Referenced by `message_New()`.

#### 4.64.2.6 `int message_s::isHTTP`

Definition at line 87 of file message.h.

Referenced by `message_New()`, `mtp_http_ComposeMessage()`, and `mtp_http_CreateMessage()`.

#### 4.64.2.7 `char* message_s::message_body`

Definition at line 97 of file message.h.

Referenced by `acc_connection_Thread()`, `message_Destroy()`, `message_InitializeFromAgent()`, `message_InitializeFromConnection()`, `message_InitializeFromString()`, `message_New()`, `message_send_Thread()`, `mtp_http_ComposeMessage()`, and `mtp_http_CreateMessage()`.

#### 4.64.2.8 `int message_s::message_id`

Definition at line 84 of file message.h.

Referenced by `AP_QUEUE_SEARCH_TEMPLATE()`, `message_InitializeFromAgent()`, `message_InitializeFromConnection()`, `message_InitializeFromString()`, and `message_New()`.

#### 4.64.2.9 `enum message_type_e message_s::message_type`

Definition at line 90 of file message.h.

Referenced by `acc_connection_Thread()`, `acc_MessageHandlerThread()`, `agent_Initialize()`, `MC_AclSend()`, `message_InitializeFromAgent()`, `message_InitializeFromString()`, `message_New()`, and `message_xml_parse__message()`.

#### 4.64.2.10 `char* message_s::sending_agent_name`

Definition at line 113 of file message.h.

Referenced by `message_InitializeFromAgent()`.

#### 4.64.2.11 `char* message_s::target`

Definition at line 105 of file message.h.

Referenced by MC\_AclSend(), message\_Destroy(), message\_InitializeFromAgent(), message\_InitializeFromConnection(), message\_InitializeFromString(), message\_New(), and mtp\_http\_ComposeMessage().

#### 4.64.2.12 char\* message\_s::to\_address

Definition at line 103 of file message.h.

Referenced by acc\_MessageHandlerThread(), AP\_QUEUE\_SEARCH\_TEMPLATE(), MC\_LoadAgentFromFile(), message\_Destroy(), message\_InitializeFromAgent(), message\_InitializeFromConnection(), message\_InitializeFromString(), message\_New(), message\_queue\_SendOutgoing(), message\_send\_Thread(), mtp\_http\_ComposeMessage(), and mtp\_http\_CreateMessage().

#### 4.64.2.13 char\* message\_s::update\_name

Definition at line 99 of file message.h.

Referenced by message\_Destroy(), message\_InitializeFromAgent(), message\_InitializeFromString(), and message\_New().

#### 4.64.2.14 int message\_s::update\_num

Definition at line 100 of file message.h.

Referenced by message\_New().

#### 4.64.2.15 mxml\_node\_t\* message\_s::xml\_payload

Definition at line 94 of file message.h.

Referenced by agent\_Initialize(), MC\_LoadAgentFromFile(), message\_New(), and message\_xml\_parse\_message().

#### 4.64.2.16 mxml\_node\_t\* message\_s::xml\_root

Definition at line 93 of file message.h.

Referenced by acc\_connection\_Thread(), agent\_Initialize(), MC\_LoadAgentFromFile(), message\_Destroy(), message\_InitializeFromAgent(), message\_InitializeFromConnection(), message\_InitializeFromString(), message\_New(), and message\_xml\_parse().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/include/[message.h](#)

## 4.65 message\_send\_arg\_s Struct Reference

```
#include <message.h>
```

### Data Fields

- struct [mc\\_platform\\_s](#) \* [mc\\_platform](#)
- [message\\_p](#) [message](#)
- char \* [privatekey](#)

### 4.65.1 Detailed Description

Definition at line 117 of file [message.h](#).

### 4.65.2 Field Documentation

#### 4.65.2.1 struct [mc\\_platform\\_s](#)\* [message\\_send\\_arg\\_s::mc\\_platform](#)

Definition at line 119 of file [message.h](#).

Referenced by [message\\_Send\(\)](#).

#### 4.65.2.2 [message\\_p](#) [message\\_send\\_arg\\_s::message](#)

Definition at line 120 of file [message.h](#).

Referenced by [message\\_Send\(\)](#).

#### 4.65.2.3 char\* [message\\_send\\_arg\\_s::privatekey](#)

Definition at line 121 of file [message.h](#).

Referenced by [message\\_Send\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/message.h](#)

## 4.66 mpi Struct Reference

MPI structure.

```
#include <bignum.h>
```

### Data Fields

- `int s`
- `int n`
- `t_int * p`

### 4.66.1 Detailed Description

MPI structure.

Definition at line 48 of file bignum.h.

### 4.66.2 Field Documentation

#### 4.66.2.1 `int mpi::n`

total # of limbs

Definition at line 51 of file bignum.h.

Referenced by `debug_print_mpi()`, `mpi_add_abs()`, `mpi_add_int()`, `mpi_cmp_abs()`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_div_int()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_grow()`, `mpi_init()`, `mpi_is_prime()`, `mpi_lsb()`, `mpi_lset()`, `mpi_mod_int()`, `mpi_montmul()`, `mpi_montred()`, `mpi_msb()`, `mpi_mul_int()`, `mpi_mul_mpi()`, `mpi_shift_l()`, `mpi_shift_r()`, `mpi_sub_abs()`, `mpi_sub_int()`, `mpi_write_string()`, and `x509parse_cert_info()`.

#### 4.66.2.2 `t_int* mpi::p`

pointer to limbs

Definition at line 52 of file bignum.h.

Referenced by `debug_print_mpi()`, `dhm_make_params()`, `dhm_make_public()`, `mpi_add_abs()`, `mpi_add_int()`, `mpi_cmp_abs()`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_div_int()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_grow()`, `mpi_init()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_lsb()`, `mpi_lset()`, `mpi_mod_int()`, `mpi_montg_init()`, `mpi_montmul()`, `mpi_montred()`, `mpi_msb()`, `mpi_mul_int()`, `mpi_mul_mpi()`, `mpi_read_binary()`, `mpi_read_string()`, `mpi_shift_l()`, `mpi_shift_r()`, `mpi_sub_abs()`, `mpi_sub_int()`, `mpi_write_binary()`, `mpi_write_string()`, and `rsa_check_pubkey()`.

#### 4.66.2.3 `int mpi::s`

integer sign

Definition at line 50 of file bignum.h.

Referenced by `mpi_add_int()`, `mpi_add_mpi()`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_div_int()`, `mpi_div_mpi()`, `mpi_free()`, `mpi_gcd()`, `mpi_init()`, `mpi_is_prime()`, `mpi_lset()`, `mpi_montred()`,

`mpi_mul_int()`, `mpi_mul_mpi()`, `mpi_read_string()`, `mpi_sub_int()`, `mpi_sub_mpi()`, and `mpi_write_string()`.

The documentation for this struct was generated from the following file:

- </home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/bignum.h>



## 4.67 mtp\_http\_content\_s Struct Reference

```
#include <mtp_http.h>
```

### Data Fields

- char \* [content\\_type](#)
- void \* [data](#)

### 4.67.1 Detailed Description

Definition at line 106 of file mtp\_http.h.

### 4.67.2 Field Documentation

#### 4.67.2.1 char\* mtp\_http\_content\_s::content\_type

Definition at line 108 of file mtp\_http.h.

Referenced by MC\_AclSend(), mtp\_http\_CreateMessage(), mtp\_http\_Destroy(), and mtp\_http\_Parse().

#### 4.67.2.2 void\* mtp\_http\_content\_s::data

Definition at line 109 of file mtp\_http.h.

Referenced by acc\_connection\_Thread(), MC\_AclSend(), mtp\_http\_CreateMessage(), mtp\_http\_Destroy(), and mtp\_http\_Parse().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/include/[mtp\\_http.h](#)

## 4.68 mtp\_http\_s Struct Reference

```
#include <mtp_http.h>
```

### Data Fields

- enum [http\\_status\\_code\\_e](#) `http_status_code`
- enum [http\\_performative\\_e](#) `http_performative`
- char \* [http\\_version](#)
- char \* [host](#)
- char \* [return\\_code](#)
- char \* [target](#)
- char \* [date](#)
- char \* [server](#)
- char \* [accept\\_ranges](#)
- char \* [content\\_length](#)
- char \* [connection](#)
- char \* [content\\_type](#)
- char \* [user\\_agent](#)
- char \* [cache\\_control](#)
- char \* [mime\\_version](#)
- [int](#) `response_code`
- char \* [response\\_string](#)
- [int](#) `message_parts`
- char \* [boundary](#)
- struct [mtp\\_http\\_content\\_s](#) \* `content`
- [int](#) `header_length`

### 4.68.1 Detailed Description

Definition at line 112 of file `mtp_http.h`.

### 4.68.2 Field Documentation

#### 4.68.2.1 char\* mtp\_http\_s::accept\_ranges

Definition at line 123 of file `mtp_http.h`.

Referenced by `mtp_http_Destroy()`.

#### 4.68.2.2 char\* mtp\_http\_s::boundary

Definition at line 139 of file `mtp_http.h`.

Referenced by `mtp_http_Destroy()`, and `mtp_http_Parse()`.

#### 4.68.2.3 char\* mtp\_http\_s::cache\_control

Definition at line 129 of file `mtp_http.h`.

Referenced by `mtp_http_Destroy()`.

**4.68.2.4 char\* mtp\_http\_s::connection**

Definition at line 125 of file mtp\_http.h.

Referenced by mtp\_http\_Destroy().

**4.68.2.5 struct mtp\_http\_content\_s\* mtp\_http\_s::content**

Definition at line 140 of file mtp\_http.h.

Referenced by acc\_connection\_Thread(), MC\_AclSend(), mtp\_http\_CreateMessage(), mtp\_http\_Destroy(), mtp\_http\_New(), and mtp\_http\_Parse().

**4.68.2.6 char\* mtp\_http\_s::content\_length**

Definition at line 124 of file mtp\_http.h.

Referenced by mtp\_http\_Destroy(), mtp\_http\_InitializeFromConnection(), and mtp\_http\_Parse().

**4.68.2.7 char\* mtp\_http\_s::content\_type**

Definition at line 126 of file mtp\_http.h.

Referenced by mtp\_http\_Destroy(), and mtp\_http\_Parse().

**4.68.2.8 char\* mtp\_http\_s::date**

Definition at line 121 of file mtp\_http.h.

Referenced by mtp\_http\_Destroy().

**4.68.2.9 int mtp\_http\_s::header\_length**

Definition at line 143 of file mtp\_http.h.

Referenced by mtp\_http\_InitializeFromConnection(), and mtp\_http\_ParseHeader().

**4.68.2.10 char\* mtp\_http\_s::host**

Definition at line 117 of file mtp\_http.h.

Referenced by MC\_AclSend(), mtp\_http\_CreateMessage(), and mtp\_http\_Destroy().

**4.68.2.11 enum http\_performative\_e mtp\_http\_s::http\_performative**

Definition at line 115 of file mtp\_http.h.

Referenced by acc\_connection\_Thread(), http\_ParseRequest(), mtp\_http\_InitializeFromConnection(), and mtp\_http\_Parse().

**4.68.2.12 enum http\_status\_code\_e mtp\_http\_s::http\_status\_code**

Definition at line 114 of file mtp\_http.h.

**4.68.2.13 char\* mtp\_http\_s::http\_version**

Definition at line 116 of file mtp\_http.h.

Referenced by mtp\_http\_Destroy().

**4.68.2.14 int mtp\_http\_s::message\_parts**

Definition at line 138 of file mtp\_http.h.

Referenced by acc\_connection\_Thread(), MC\_AclSend(), mtp\_http\_CreateMessage(), mtp\_http\_Destroy(), and mtp\_http\_Parse().

**4.68.2.15 char\* mtp\_http\_s::mime\_version**

Definition at line 130 of file mtp\_http.h.

Referenced by mtp\_http\_Destroy().

**4.68.2.16 int mtp\_http\_s::response\_code**

Definition at line 133 of file mtp\_http.h.

Referenced by http\_ParseRequest().

**4.68.2.17 char\* mtp\_http\_s::response\_string**

Definition at line 134 of file mtp\_http.h.

Referenced by http\_ParseRequest(), and mtp\_http\_Destroy().

**4.68.2.18 char\* mtp\_http\_s::return\_code**

Definition at line 118 of file mtp\_http.h.

Referenced by mtp\_http\_Destroy().

**4.68.2.19 char\* mtp\_http\_s::server**

Definition at line 122 of file mtp\_http.h.

Referenced by mtp\_http\_Destroy().

**4.68.2.20 char\* mtp\_http\_s::target**

Definition at line 119 of file mtp\_http.h.

Referenced by `acc_connection_Thread()`, `http_ParseRequest()`, `MC_AclSend()`, `mtp_http_CreateMessage()`, and `mtp_http_Destroy()`.

#### 4.68.2.21 `char* mtp_http_s::user_agent`

Definition at line 127 of file `mtp_http.h`.

Referenced by `mtp_http_Destroy()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/include/mtp_http.h`

## 4.69 mxml\_attr\_s Struct Reference

```
#include <mxml.h>
```

### Data Fields

- char \* [name](#)
- char \* [value](#)

### 4.69.1 Detailed Description

Definition at line 86 of file mxml.h.

### 4.69.2 Field Documentation

#### 4.69.2.1 char\* mxml\_attr\_s::name

Definition at line 88 of file mxml.h.

Referenced by [mxml\\_write\\_node\(\)](#), [mxmlDelete\(\)](#), [mxmlElementGetAttr\(\)](#), and [mxmlElementSetAttr\(\)](#).

#### 4.69.2.2 char\* mxml\_attr\_s::value

Definition at line 89 of file mxml.h.

Referenced by [mxml\\_write\\_node\(\)](#), [mxmlDelete\(\)](#), [mxmlElementGetAttr\(\)](#), and [mxmlElementSetAttr\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml.h](#)

## 4.70 mxml\_custom\_s Struct Reference

```
#include <mxml.h>
```

### Data Fields

- void \* [data](#)
- void(\* [destroy](#) )(void \*)

### 4.70.1 Detailed Description

Definition at line 105 of file mxml.h.

### 4.70.2 Field Documentation

#### 4.70.2.1 void\* mxml\_custom\_s::data

Definition at line 107 of file mxml.h.

Referenced by mxmlDelete(), mxmlNewCustom(), and mxmlSetCustom().

#### 4.70.2.2 void(\* mxml\_custom\_s::destroy)(void \*)

Definition at line 108 of file mxml.h.

Referenced by mxmlDelete(), mxmlNewCustom(), and mxmlSetCustom().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml.h](#)

## 4.71 mxml\_fdbuf\_s Struct Reference

### Data Fields

- `int fd`
- `unsigned char * current`
- `unsigned char * end`
- `unsigned char buffer [8192]`

### 4.71.1 Detailed Description

Definition at line 84 of file mxml-file.c.

### 4.71.2 Field Documentation

#### 4.71.2.1 `unsigned char mxml_fdbuf_s::buffer[8192]`

Definition at line 87 of file mxml-file.c.

Referenced by `mxml_fd_read()`, `mxml_fd_write()`, `mxmlLoadFd()`, and `mxmlSaveFd()`.

#### 4.71.2.2 `unsigned char* mxml_fdbuf_s::current`

Definition at line 87 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_fd_putc()`, `mxml_fd_read()`, `mxml_fd_write()`, `mxmlLoadFd()`, and `mxmlSaveFd()`.

#### 4.71.2.3 `unsigned char * mxml_fdbuf_s::end`

Definition at line 87 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_fd_putc()`, `mxml_fd_read()`, `mxmlLoadFd()`, and `mxmlSaveFd()`.

#### 4.71.2.4 `int mxml_fdbuf_s::fd`

Definition at line 86 of file mxml-file.c.

Referenced by `mxml_fd_read()`, `mxml_fd_write()`, `mxmlLoadFd()`, and `mxmlSaveFd()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-file.c`



## 4.72 mxml\_index\_s Struct Reference

```
#include <mxml.h>
```

### Data Fields

- char \* [attr](#)
- int [num\\_nodes](#)
- int [alloc\\_nodes](#)
- int [cur\\_node](#)
- [mxml\\_node\\_t](#) \*\* [nodes](#)

### 4.72.1 Detailed Description

Definition at line 133 of file mxml.h.

### 4.72.2 Field Documentation

#### 4.72.2.1 int mxml\_index\_s::alloc\_nodes

Definition at line 137 of file mxml.h.

Referenced by [mxmlIndexDelete\(\)](#), and [mxmlIndexNew\(\)](#).

#### 4.72.2.2 char\* mxml\_index\_s::attr

Definition at line 135 of file mxml.h.

Referenced by [index\\_compare\(\)](#), [index\\_find\(\)](#), [mxmlIndexDelete\(\)](#), [mxmlIndexFind\(\)](#), and [mxmlIndexNew\(\)](#).

#### 4.72.2.3 int mxml\_index\_s::cur\_node

Definition at line 138 of file mxml.h.

Referenced by [mxmlIndexEnum\(\)](#), [mxmlIndexFind\(\)](#), and [mxmlIndexReset\(\)](#).

#### 4.72.2.4 mxml\_node\_t\*\* mxml\_index\_s::nodes

Definition at line 139 of file mxml.h.

Referenced by [index\\_sort\(\)](#), [mxmlIndexDelete\(\)](#), [mxmlIndexEnum\(\)](#), [mxmlIndexFind\(\)](#), [mxmlIndexNew\(\)](#), and [mxmlIndexReset\(\)](#).

#### 4.72.2.5 int mxml\_index\_s::num\_nodes

Definition at line 136 of file mxml.h.

Referenced by [main\(\)](#), [mxmlIndexEnum\(\)](#), [mxmlIndexFind\(\)](#), [mxmlIndexNew\(\)](#), and [mxmlIndexReset\(\)](#).

The documentation for this struct was generated from the following file:

- </home/dko/Projects/mobilec/src/mxml-2.2.2/mxml.h>

## 4.73 mxml\_node\_s Struct Reference

```
#include <mxml.h>
```

### Data Fields

- [mxml\\_type\\_t](#) type
- struct [mxml\\_node\\_s](#) \* next
- struct [mxml\\_node\\_s](#) \* prev
- struct [mxml\\_node\\_s](#) \* parent
- struct [mxml\\_node\\_s](#) \* child
- struct [mxml\\_node\\_s](#) \* last\_child
- [mxml\\_value\\_t](#) value

### 4.73.1 Detailed Description

Definition at line 122 of file mxml.h.

### 4.73.2 Field Documentation

#### 4.73.2.1 struct mxml\_node\_s\* mxml\_node\_s::child

Definition at line 128 of file mxml.h.

Referenced by `add_variable()`, `agent_xml_parse__fill_row_data()`, `fipa_envelope_HandleAclRepresentation()`, `fipa_envelope_HandleComments()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandlePayloadEncoding()`, `fipa_envelope_HandlePayloadLength()`, `fipa_envelope_ParseAddresses()`, `fipa_envelope_ParseAgentIdentifier()`, `main()`, `mxml_write_node()`, `mxmlAdd()`, `mxmlDelete()`, `mxmlRemove()`, `mxmlWalkNext()`, `scan_file()`, `sort_node()`, `whitespace_cb()`, `write_documentation()`, `write_element()`, and `xml_get_text()`.

#### 4.73.2.2 struct mxml\_node\_s\* mxml\_node\_s::last\_child

Definition at line 129 of file mxml.h.

Referenced by `add_variable()`, `main()`, `mxmlAdd()`, `mxmlRemove()`, `mxmlWalkPrev()`, and `scan_file()`.

#### 4.73.2.3 struct mxml\_node\_s\* mxml\_node\_s::next

Definition at line 125 of file mxml.h.

Referenced by `add_variable()`, `main()`, `mxml_write_node()`, `mxmlAdd()`, `mxmlFindElement()`, `mxmlRemove()`, `mxmlWalkNext()`, `scan_file()`, `sort_node()`, `xml_get_next_element()`, and `xml_get_text()`.

#### 4.73.2.4 struct mxml\_node\_s\* mxml\_node\_s::parent

Definition at line 127 of file mxml.h.

Referenced by `agent_xml_parse__data()`, `mxml_load_data()`, `mxmlAdd()`, `mxmlRemove()`, `mxmlWalkNext()`, `mxmlWalkPrev()`, `sort_node()`, `whitespace_cb()`, `ws_cb()`, and `xml_find_sibling()`.

#### 4.73.2.5 struct mxml\_node\_s\* mxml\_node\_s::prev

Definition at line 126 of file mxml.h.

Referenced by mxml\_write\_node(), mxmlAdd(), mxmlRemove(), and mxmlWalkPrev().

#### 4.73.2.6 mxml\_type\_t mxml\_node\_s::type

Definition at line 124 of file mxml.h.

Referenced by agent\_xml\_parse\_\_fill\_row\_data(), fipa\_envelope\_HandleAclRepresentation(), fipa\_envelope\_HandleComments(), fipa\_envelope\_HandleDate(), fipa\_envelope\_HandlePayloadEncoding(), fipa\_envelope\_HandlePayloadLength(), fipa\_envelope\_ParseAddresses(), fipa\_envelope\_ParseAgentIdentifier(), main(), mxml\_new(), mxml\_write\_node(), mxmlDelete(), mxmlElementGetAttr(), mxmlElementSetAttr(), mxmlFindElement(), mxmlSetCustom(), mxmlSetElement(), mxmlSetInteger(), mxmlSetOpaque(), mxmlSetReal(), mxmlSetText(), mxmlSetTextf(), write\_element(), xml\_get\_cdata(), xml\_get\_element\_name(), xml\_get\_next\_element(), and xml\_get\_text().

#### 4.73.2.7 mxml\_value\_t mxml\_node\_s::value

Definition at line 130 of file mxml.h.

Referenced by add\_variable(), agent\_xml\_parse\_\_fill\_row\_data(), fipa\_envelope\_HandleAclRepresentation(), fipa\_envelope\_HandleComments(), fipa\_envelope\_HandleDate(), fipa\_envelope\_HandlePayloadEncoding(), fipa\_envelope\_HandlePayloadLength(), fipa\_envelope\_ParseAddresses(), fipa\_envelope\_ParseAgentIdentifier(), index\_compare(), index\_find(), main(), mxml\_get\_entity(), mxml\_load\_data(), mxml\_parse\_element(), mxml\_write\_node(), mxmlDelete(), mxmlElementGetAttr(), mxmlElementSetAttr(), mxmlFindElement(), mxmlIndexNew(), mxmlNewCustom(), mxmlNewElement(), mxmlNewInteger(), mxmlNewOpaque(), mxmlNewReal(), mxmlNewText(), mxmlNewTextf(), mxmlSetCustom(), mxmlSetElement(), mxmlSetInteger(), mxmlSetOpaque(), mxmlSetReal(), mxmlSetText(), mxmlSetTextf(), scan\_file(), sort\_node(), type\_cb(), update\_comment(), whitespace\_cb(), write\_element(), ws\_cb(), xml\_get\_element\_name(), and xml\_get\_text().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/mxml-2.2.2/[mxml.h](#)

## 4.74 mxml\_text\_s Struct Reference

```
#include <mxml.h>
```

### Data Fields

- `int whitespace`
- `char * string`

#### 4.74.1 Detailed Description

Definition at line 99 of file mxml.h.

#### 4.74.2 Field Documentation

##### 4.74.2.1 `char* mxml_text_s::string`

Definition at line 102 of file mxml.h.

Referenced by `add_variable()`, `agent_xml_parse__fill_row_data()`, `fipa_envelope_HandleAclRepresentation()`, `fipa_envelope_HandleComments()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandlePayloadEncoding()`, `fipa_envelope_HandlePayloadLength()`, `fipa_envelope_ParseAddresses()`, `fipa_envelope_ParseAgentIdentifier()`, `main()`, `mxml_write_node()`, `mxmlDelete()`, `mxmlNewText()`, `mxmlNewTextf()`, `mxmlSetText()`, `mxmlSetTextf()`, `scan_file()`, `update_comment()`, `write_element()`, and `xml_get_text()`.

##### 4.74.2.2 `int mxml_text_s::whitespace`

Definition at line 101 of file mxml.h.

Referenced by `add_variable()`, `main()`, `mxml_write_node()`, `mxmlNewText()`, `mxmlNewTextf()`, `mxmlSetText()`, `mxmlSetTextf()`, `scan_file()`, and `write_element()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml.h`

## 4.75 mxml\_value\_s Struct Reference

```
#include <mxml.h>
```

### Data Fields

- `char * name`
- `int num_attrs`
- `mxml_attr_t * attrs`

### 4.75.1 Detailed Description

Definition at line 92 of file `mxml.h`.

### 4.75.2 Field Documentation

#### 4.75.2.1 `mxml_attr_t* mxml_value_s::attrs`

Definition at line 96 of file `mxml.h`.

Referenced by `mxml_write_node()`, `mxmlDelete()`, `mxmlElementGetAttr()`, and `mxmlElementSetAttr()`.

#### 4.75.2.2 `char* mxml_value_s::name`

Definition at line 94 of file `mxml.h`.

Referenced by `index_compare()`, `index_find()`, `main()`, `mxml_get_entity()`, `mxml_load_data()`, `mxml_parse_element()`, `mxml_write_node()`, `mxmlDelete()`, `mxmlElementSetAttr()`, `mxmlFindElement()`, `mxmlIndexNew()`, `mxmlNewElement()`, `mxmlSetElement()`, `scan_file()`, `sort_node()`, `type_cb()`, `update_comment()`, `whitespace_cb()`, `ws_cb()`, `xml_get_element_name()`, and `xml_get_text()`.

#### 4.75.2.3 `int mxml_value_s::num_attrs`

Definition at line 95 of file `mxml.h`.

Referenced by `mxml_write_node()`, `mxmlDelete()`, `mxmlElementGetAttr()`, and `mxmlElementSetAttr()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/mxml-2.2.2/mxml.h`

## 4.76 mxml\_value\_u Union Reference

```
#include <mxml.h>
```

### Data Fields

- [mxml\\_element\\_t element](#)
- [int integer](#)
- [char \\* opaque](#)
- [double real](#)
- [mxml\\_text\\_t text](#)
- [mxml\\_custom\\_t custom](#)

### 4.76.1 Detailed Description

Definition at line 112 of file mxml.h.

### 4.76.2 Field Documentation

#### 4.76.2.1 mxml\_custom\_t mxml\_value\_u::custom

Definition at line 119 of file mxml.h.

Referenced by [mxmlDelete\(\)](#), [mxmlNewCustom\(\)](#), and [mxmlSetCustom\(\)](#).

#### 4.76.2.2 mxml\_element\_t mxml\_value\_u::element

Definition at line 114 of file mxml.h.

Referenced by [index\\_compare\(\)](#), [index\\_find\(\)](#), [main\(\)](#), [mxml\\_get\\_entity\(\)](#), [mxml\\_load\\_data\(\)](#), [mxml\\_parse\\_element\(\)](#), [mxml\\_write\\_node\(\)](#), [mxmlDelete\(\)](#), [mxmlElementGetAttr\(\)](#), [mxmlElementSetAttr\(\)](#), [mxmlFindElement\(\)](#), [mxmlIndexNew\(\)](#), [mxmlNewElement\(\)](#), [mxmlSetElement\(\)](#), [scan\\_file\(\)](#), [sort\\_node\(\)](#), [type\\_cb\(\)](#), [update\\_comment\(\)](#), [whitespace\\_cb\(\)](#), [ws\\_cb\(\)](#), [xml\\_get\\_element\\_name\(\)](#), and [xml\\_get\\_text\(\)](#).

#### 4.76.2.3 int mxml\_value\_u::integer

Definition at line 115 of file mxml.h.

Referenced by [main\(\)](#), [mxml\\_write\\_node\(\)](#), [mxmlNewInteger\(\)](#), and [mxmlSetInteger\(\)](#).

#### 4.76.2.4 char\* mxml\_value\_u::opaque

Definition at line 116 of file mxml.h.

Referenced by [main\(\)](#), [mxml\\_write\\_node\(\)](#), [mxmlDelete\(\)](#), [mxmlNewOpaque\(\)](#), and [mxmlSetOpaque\(\)](#).

#### 4.76.2.5 double mxml\_value\_u::real

Definition at line 117 of file mxml.h.

Referenced by [main\(\)](#), [mxml\\_write\\_node\(\)](#), [mxmlNewReal\(\)](#), and [mxmlSetReal\(\)](#).

#### 4.76.2.6 mxml\_text\_t mxml\_value\_u::text

Definition at line 118 of file mxml.h.

Referenced by add\_variable(), agent\_xml\_parse\_\_fill\_row\_data(), fipa\_envelope\_HandleAclRepresentation(), fipa\_envelope\_HandleComments(), fipa\_envelope\_HandleDate(), fipa\_envelope\_HandlePayloadEncoding(), fipa\_envelope\_HandlePayloadLength(), fipa\_envelope\_ParseAddresses(), fipa\_envelope\_ParseAgentIdentifier(), main(), mxml\_write\_node(), mxmlDelete(), mxmlNewText(), mxmlNewTextf(), mxmlSetText(), mxmlSetTextf(), scan\_file(), update\_comment(), write\_element(), and xml\_get\_text().

The documentation for this union was generated from the following file:

- /home/dko/Projects/mobilec/src/mxml-2.2.2/[mxml.h](#)



## 4.77 options Struct Reference

### Data Fields

- [int opmode](#)
- [int iomode](#)
- [char \\* server\\_name](#)
- [int server\\_port](#)
- [int command](#)
- [int buffer\\_size](#)
- [int max\\_bytes](#)
- [int debug\\_level](#)
- [int conn\\_timeout](#)
- [int max\\_connections](#)
- [int session\\_reuse](#)
- [int session\\_lifetime](#)
- [int force\\_cipher](#) [2]

### 4.77.1 Detailed Description

Definition at line 75 of file `ssl_test.c`.

### 4.77.2 Field Documentation

#### 4.77.2.1 `int options::buffer_size`

Definition at line 82 of file `ssl_test.c`.

Referenced by `main()`, and `ssl_test()`.

#### 4.77.2.2 `int options::command`

Definition at line 81 of file `ssl_test.c`.

Referenced by `main()`, and `ssl_test()`.

#### 4.77.2.3 `int options::conn_timeout`

Definition at line 85 of file `ssl_test.c`.

Referenced by `main()`, and `ssl_test()`.

#### 4.77.2.4 `int options::debug_level`

Definition at line 84 of file `ssl_test.c`.

Referenced by `main()`.

**4.77.2.5 int options::force\_cipher[2]**

Definition at line 89 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

**4.77.2.6 int options::iomode**

Definition at line 78 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

**4.77.2.7 int options::max\_bytes**

Definition at line 83 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

**4.77.2.8 int options::max\_connections**

Definition at line 86 of file ssl\_test.c.

Referenced by main().

**4.77.2.9 int options::opmode**

Definition at line 77 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

**4.77.2.10 char\* options::server\_name**

Definition at line 79 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

**4.77.2.11 int options::server\_port**

Definition at line 80 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

**4.77.2.12 int options::session\_lifetime**

Definition at line 88 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

**4.77.2.13 int options::session\_reuse**

Definition at line 87 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/ssl\\_test.c](#)

## 4.78 rsa\_context Struct Reference

RSA context structure.

```
#include <rsa.h>
```

### Data Fields

- [int ver](#)
- [int len](#)
- [mpi N](#)
- [mpi E](#)
- [mpi D](#)
- [mpi P](#)
- [mpi Q](#)
- [mpi DP](#)
- [mpi DQ](#)
- [mpi QP](#)
- [mpi RN](#)
- [mpi RP](#)
- [mpi RQ](#)
- [int padding](#)
- [int hash\\_id](#)
- [int\(\\* f\\_rng\)\(void \\*\)](#)
- [void \\* p\\_rng](#)

### 4.78.1 Detailed Description

RSA context structure.

Definition at line 56 of file `rsa.h`.

### 4.78.2 Field Documentation

#### 4.78.2.1 `mpi rsa_context::D`

private exponent

Definition at line 64 of file `rsa.h`.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, `main()`, `rsa_check_privkey()`, `rsa_decryption()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_self_test()`, and `x509parse_key()`.

#### 4.78.2.2 `mpi rsa_context::DP`

$D \% (P - 1)$

Definition at line 67 of file `rsa.h`.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, `main()`, `rsa_decryption()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_self_test()`, and `x509parse_key()`.

#### 4.78.2.3 mpi rsa\_context::DQ

$D \% (Q - 1)$

Definition at line 68 of file rsa.h.

Referenced by generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), main(), rsa\_decryption(), rsa\_free(), rsa\_gen\_key(), rsa\_private(), rsa\_self\_test(), and x509parse\_key().

#### 4.78.2.4 mpi rsa\_context::E

public exponent

Definition at line 62 of file rsa.h.

Referenced by d2i\_RSA\_PUBKEY(), debug\_print\_crt(), generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), main(), rsa\_check\_privkey(), rsa\_check\_pubkey(), rsa\_decryption(), rsa\_encryption(), rsa\_free(), rsa\_gen\_key(), rsa\_public(), rsa\_self\_test(), x509parse\_crt(), and x509parse\_key().

#### 4.78.2.5 int(\* rsa\_context::f\_rng)(void \*)

RNG function

Definition at line 77 of file rsa.h.

Referenced by rsa\_gen\_key(), and rsa\_init().

#### 4.78.2.6 int rsa\_context::hash\_id

hash identifier

Definition at line 76 of file rsa.h.

Referenced by rsa\_init().

#### 4.78.2.7 int rsa\_context::len

size(N) in chars

Definition at line 59 of file rsa.h.

Referenced by d2i\_RSA\_PUBKEY(), main(), rsa\_decryption(), rsa\_encryption(), rsa\_gen\_key(), rsa\_pkcs1\_decrypt(), rsa\_pkcs1\_encrypt(), rsa\_pkcs1\_sign(), rsa\_pkcs1\_verify(), rsa\_private(), rsa\_public(), rsa\_self\_test(), ssl\_parse\_certificate\_verify(), ssl\_parse\_client\_key\_exchange(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_certificate\_verify(), ssl\_write\_client\_key\_exchange(), ssl\_write\_server\_key\_exchange(), x509parse\_crt(), and x509parse\_key().

#### 4.78.2.8 mpi rsa\_context::N

public modulus

Definition at line 61 of file rsa.h.

Referenced by d2i\_RSA\_PUBKEY(), debug\_print\_crt(), generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), main(), rsa\_check\_privkey(), rsa\_check\_pubkey(), rsa\_decryption(), rsa\_

encryption(), rsa\_free(), rsa\_gen\_key(), rsa\_private(), rsa\_public(), rsa\_self\_test(), x509parse\_cert\_info(), x509parse\_crt(), and x509parse\_key().

#### 4.78.2.9 `mpi rsa_context::P`

1st prime factor

Definition at line 65 of file rsa.h.

Referenced by generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), main(), rsa\_check\_privkey(), rsa\_decryption(), rsa\_free(), rsa\_gen\_key(), rsa\_private(), rsa\_self\_test(), and x509parse\_key().

#### 4.78.2.10 `void* rsa_context::p_rng`

RNG parameter

Definition at line 78 of file rsa.h.

Referenced by rsa\_gen\_key(), and rsa\_init().

#### 4.78.2.11 `int rsa_context::padding`

1.5 or OAEP/PSS

Definition at line 75 of file rsa.h.

Referenced by rsa\_init(), rsa\_pkcs1\_decrypt(), rsa\_pkcs1\_encrypt(), rsa\_pkcs1\_sign(), and rsa\_pkcs1\_verify().

#### 4.78.2.12 `mpi rsa_context::Q`

2nd prime factor

Definition at line 66 of file rsa.h.

Referenced by generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), main(), rsa\_check\_privkey(), rsa\_decryption(), rsa\_free(), rsa\_gen\_key(), rsa\_private(), rsa\_self\_test(), and x509parse\_key().

#### 4.78.2.13 `mpi rsa_context::QP`

$1 / (Q \% P)$

Definition at line 69 of file rsa.h.

Referenced by generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), main(), rsa\_decryption(), rsa\_free(), rsa\_gen\_key(), rsa\_private(), rsa\_self\_test(), and x509parse\_key().

#### 4.78.2.14 `mpi rsa_context::RN`

cached  $R^2 \bmod N$

Definition at line 71 of file rsa.h.

Referenced by rsa\_free(), rsa\_private(), and rsa\_public().

**4.78.2.15 mpi rsa\_context::RP**

cached  $R^2 \bmod P$

Definition at line 72 of file rsa.h.

Referenced by rsa\_free(), and rsa\_private().

**4.78.2.16 mpi rsa\_context::RQ**

cached  $R^2 \bmod Q$

Definition at line 73 of file rsa.h.

Referenced by rsa\_free(), and rsa\_private().

**4.78.2.17 int rsa\_context::ver**

always 0

Definition at line 58 of file rsa.h.

Referenced by x509parse\_key().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/[rsa.h](#)

## 4.79 sha1\_context Struct Reference

SHA-1 context structure.

```
#include <sha1.h>
```

### Data Fields

- unsigned long [total](#) [2]
- unsigned long [state](#) [5]
- unsigned char [buffer](#) [64]
- unsigned char [ipad](#) [64]
- unsigned char [opad](#) [64]

### 4.79.1 Detailed Description

SHA-1 context structure.

Definition at line 10 of file sha1.h.

### 4.79.2 Field Documentation

#### 4.79.2.1 unsigned char sha1\_context::buffer[64]

data block being processed

Definition at line 14 of file sha1.h.

Referenced by sha1\_update().

#### 4.79.2.2 unsigned char sha1\_context::ipad[64]

HMAC: inner padding

Definition at line 16 of file sha1.h.

Referenced by sha1\_hmac\_starts().

#### 4.79.2.3 unsigned char sha1\_context::opad[64]

HMAC: outer padding

Definition at line 17 of file sha1.h.

Referenced by sha1\_hmac\_finish(), and sha1\_hmac\_starts().

#### 4.79.2.4 unsigned long sha1\_context::state[5]

intermediate digest state

Definition at line 13 of file sha1.h.

Referenced by sha1\_finish(), sha1\_process(), sha1\_starts(), and ssl\_calc\_finished().



**4.79.2.5 unsigned long sha1\_context::total[2]**

number of bytes processed

Definition at line 12 of file sha1.h.

Referenced by sha1\_finish(), sha1\_starts(), and sha1\_update().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/[sha1.h](#)

## 4.80 sha2\_context Struct Reference

SHA-256 context structure.

```
#include <sha2.h>
```

### Data Fields

- unsigned long [total](#) [2]
- unsigned long [state](#) [8]
- unsigned char [buffer](#) [64]
- unsigned char [ipad](#) [64]
- unsigned char [opad](#) [64]
- [int is224](#)

### 4.80.1 Detailed Description

SHA-256 context structure.

Definition at line 10 of file sha2.h.

### 4.80.2 Field Documentation

#### 4.80.2.1 unsigned char sha2\_context::buffer[64]

data block being processed

Definition at line 14 of file sha2.h.

Referenced by sha2\_update().

#### 4.80.2.2 unsigned char sha2\_context::ipad[64]

HMAC: inner padding

Definition at line 16 of file sha2.h.

Referenced by sha2\_hmac\_starts().

#### 4.80.2.3 int sha2\_context::is224

0 => SHA-256, else SHA-224

Definition at line 18 of file sha2.h.

Referenced by sha2\_finish(), sha2\_hmac\_finish(), and sha2\_starts().

#### 4.80.2.4 unsigned char sha2\_context::opad[64]

HMAC: outer padding

Definition at line 17 of file sha2.h.

Referenced by sha2\_hmac\_finish(), and sha2\_hmac\_starts().

**4.80.2.5 unsigned long sha2\_context::state[8]**

intermediate digest state

Definition at line 13 of file sha2.h.

Referenced by sha2\_finish(), sha2\_process(), and sha2\_starts().

**4.80.2.6 unsigned long sha2\_context::total[2]**

number of bytes processed

Definition at line 12 of file sha2.h.

Referenced by sha2\_finish(), sha2\_starts(), and sha2\_update().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/[sha2.h](#)

## 4.81 sha4\_context Struct Reference

SHA-512 context structure.

```
#include <sha4.h>
```

### Data Fields

- unsigned int64 [total](#) [2]
- unsigned int64 [state](#) [8]
- unsigned char [buffer](#) [128]
- unsigned char [ipad](#) [128]
- unsigned char [opad](#) [128]
- [int is384](#)

### 4.81.1 Detailed Description

SHA-512 context structure.

Definition at line 18 of file sha4.h.

### 4.81.2 Field Documentation

#### 4.81.2.1 unsigned char sha4\_context::buffer[128]

data block being processed

Definition at line 22 of file sha4.h.

Referenced by sha4\_update().

#### 4.81.2.2 unsigned char sha4\_context::ipad[128]

HMAC: inner padding

Definition at line 24 of file sha4.h.

Referenced by sha4\_hmac\_starts().

#### 4.81.2.3 int sha4\_context::is384

0 => SHA-512, else SHA-384

Definition at line 26 of file sha4.h.

Referenced by sha4\_finish(), sha4\_hmac\_finish(), and sha4\_starts().

#### 4.81.2.4 unsigned char sha4\_context::opad[128]

HMAC: outer padding

Definition at line 25 of file sha4.h.

Referenced by sha4\_hmac\_finish(), and sha4\_hmac\_starts().

**4.81.2.5 unsigned int64 sha4\_context::state[8]**

intermediate digest state

Definition at line 21 of file sha4.h.

Referenced by sha4\_finish(), sha4\_process(), and sha4\_starts().

**4.81.2.6 unsigned int64 sha4\_context::total[2]**

number of bytes processed

Definition at line 20 of file sha4.h.

Referenced by sha4\_finish(), sha4\_starts(), and sha4\_update().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/[sha4.h](#)

## 4.82 stationary\_agent\_info\_s Struct Reference

```
#include <libmc.h>
```

### Data Fields

- void \* [args](#)
- struct [agent\\_s](#) \* [agent](#)
- [MCAgency\\_t](#) [attr](#)
- [MCAgency\\_t](#) [agency](#)
- [THREAD\\_T](#) [thread](#)

### 4.82.1 Detailed Description

Definition at line 271 of file libmc.h.

### 4.82.2 Field Documentation

#### 4.82.2.1 MCAgency\_t stationary\_agent\_info\_s::agency

Definition at line 275 of file libmc.h.

Referenced by [MC\\_AddStationaryAgent\(\)](#), and [MC\\_AgentInfo\\_GetAgency\(\)](#).

#### 4.82.2.2 struct agent\_s\* stationary\_agent\_info\_s::agent

Definition at line 273 of file libmc.h.

Referenced by [MC\\_AddStationaryAgent\(\)](#), and [MC\\_AgentInfo\\_GetAgent\(\)](#).

#### 4.82.2.3 void\* stationary\_agent\_info\_s::args

Definition at line 272 of file libmc.h.

Referenced by [MC\\_AddStationaryAgent\(\)](#), and [MC\\_AgentInfo\\_GetAgentArgs\(\)](#).

#### 4.82.2.4 MCAgency\_t stationary\_agent\_info\_s::attr

Definition at line 274 of file libmc.h.

Referenced by [MC\\_AddStationaryAgent\(\)](#).

#### 4.82.2.5 THREAD\_T stationary\_agent\_info\_s::thread

Definition at line 276 of file libmc.h.

Referenced by [MC\\_AddStationaryAgent\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/src/include/libmc.h](#)

## 4.83 syncList\_s Struct Reference

```
#include <sync_list.h>
```

### Data Fields

- `RWLOCK_T * lock`
- `MUTEX_T * giant_lock`
- `list_p list`
- `int size`

### 4.83.1 Detailed Description

Definition at line 26 of file `sync_list.h`.

### 4.83.2 Field Documentation

#### 4.83.2.1 `MUTEX_T* syncList_s::giant_lock`

Definition at line 28 of file `sync_list.h`.

Referenced by `MC_SyncDelete()`, `MC_SyncInit()`, `syncListDestroy()`, and `syncListInit()`.

#### 4.83.2.2 `list_p syncList_s::list`

Definition at line 30 of file `sync_list.h`.

Referenced by `syncListAddNode()`, `syncListDelete()`, `syncListDestroy()`, `syncListFind()`, `syncListInit()`, and `syncListRemove()`.

#### 4.83.2.3 `RWLOCK_T* syncList_s::lock`

Definition at line 27 of file `sync_list.h`.

Referenced by `syncListAddNode()`, `syncListDelete()`, `syncListDestroy()`, `syncListFind()`, `syncListInit()`, and `syncListRemove()`.

#### 4.83.2.4 `int syncList_s::size`

Definition at line 31 of file `sync_list.h`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/src/mc_sync/sync_list.h`

## 4.84 syncListNode\_s Struct Reference

```
#include <sync_list.h>
```

### Data Fields

- `MUTEX_T * lock`
- `COND_T * cond`
- `SEMAPHORE_T * sem`
- `int id`
- `int signalled`

#### 4.84.1 Detailed Description

Definition at line 16 of file `sync_list.h`.

#### 4.84.2 Field Documentation

##### 4.84.2.1 `COND_T* syncListNode_s::cond`

Definition at line 18 of file `sync_list.h`.

Referenced by `MC_CondBroadcast()`, `MC_CondSignal()`, `MC_CondWait()`, `syncListNodeDestroy()`, `syncListNodeInit()`, and `syncListNodeNew()`.

##### 4.84.2.2 `int syncListNode_s::id`

Definition at line 20 of file `sync_list.h`.

Referenced by `MC_SyncInit()`, `syncListAddNode()`, `syncListDelete()`, and `syncListRemove()`.

##### 4.84.2.3 `MUTEX_T* syncListNode_s::lock`

Definition at line 17 of file `sync_list.h`.

Referenced by `MC_CondBroadcast()`, `MC_CondReset()`, `MC_CondSignal()`, `MC_CondWait()`, `MC_MutexLock()`, `MC_MutexUnlock()`, `MC_SyncDelete()`, `syncListNodeDestroy()`, `syncListNodeInit()`, and `syncListNodeNew()`.

##### 4.84.2.4 `SEMAPHORE_T* syncListNode_s::sem`

Definition at line 19 of file `sync_list.h`.

Referenced by `MC_SemaphorePost()`, `MC_SemaphoreWait()`, `syncListNodeDestroy()`, `syncListNodeInit()`, and `syncListNodeNew()`.

##### 4.84.2.5 `int syncListNode_s::signalled`

Definition at line 21 of file `sync_list.h`.



Referenced by MC\_CondBroadcast(), MC\_CondReset(), MC\_CondSignal(), MC\_CondWait(), and syncListNodeNew().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/src/mc\_sync/[sync\\_list.h](#)



## Chapter 5

# File Documentation

### 5.1 /home/dko/Projects/mobilec/src/acc.c File Reference

```
#include <sys/socket.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <netdb.h>
#include <sys/un.h>
#include <unistd.h>
#include <sys/time.h>
#include <pthread.h>
#include "config.h"
#include <stdlib.h>
#include "include/acc.h"
#include "macros.h"
#include <sys/types.h>
#include "ap_queue_template.h"
#include "../mc_list/list.h"
#include "connection.h"
#include "fipa_acl.h"
#include "agent_mailbox.h"
#include "message.h"
#include "interpreter_variable_data.h"
#include <semaphore.h>
#include <errno.h>
#include "acc.h"
#include "ams.h"
```

```
#include "barrier.h"
#include "cmd_prompt.h"
#include "data_structures.h"
#include "df.h"
#include "libmc.h"
#include "../mc_sync/sync_list.h"
#include "../security/asm.h"
#include "mtp_http.h"
#include <mxml.h>
#include <stdio.h>
#include <string.h>
#include <sys/stat.h>
#include "xyssl-0.9/include/xyssl/havege.h"
#include "xyssl-0.9/include/xyssl/bignum.h"
#include "xyssl-0.9/include/xyssl/rsa.h"
#include "xyssl-0.9/include/xyssl/sha2.h"
#include "xyssl-0.9/include/xyssl/sha1.h"
#include "xyssl-0.9/include/xyssl/aes.h"
#include "xyssl-0.9/include/xyssl/md5.h"
#include "agent.h"
```

## Defines

- #define [BACKLOG](#) 200
- #define [CONN\\_THREADS](#) 40
- #define [CONNECT\\_THREAD\\_EXIT](#)()
- #define [BUFLen](#) 512
- #define [UDPPort](#) 8866

## Functions

- [acc\\_p acc\\_Initialize](#) (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#))
- [int acc\\_Destroy](#) ([acc\\_p](#) [acc](#))
- [void \\* acc\\_MessageHandlerThread](#) ([void](#) \*[arg](#))
- [void \\* acc\\_Thread](#) ([void](#) \*[arg](#))
- [void \\* acc\\_connection\\_Thread](#) ([void](#) \*[arg](#))
- [void acc\\_Start](#) ([mc\\_platform\\_p](#) [mc\\_platform](#))
- [int auth\\_conn\\_rece\\_key](#) ([int](#) [sockfd](#), [char](#) \*[peer\\_name](#), [int](#) \*[nonce](#), [unsigned char](#) \*[aes\\_key](#), [char](#) \*[privkey](#), [char](#) \*[known\\_host\\_filename](#))
- [void \\* listen\\_Thread](#) ([void](#) \*[arg](#))
- [void \\* udplisten\\_Thread](#) ([void](#) \*[arg](#))

## 5.1.1 Define Documentation

### 5.1.1.1 #define BACKLOG 200

Definition at line 74 of file acc.c.

Referenced by listen\_Thread().

### 5.1.1.2 #define BUFLLEN 512

Definition at line 914 of file acc.c.

Referenced by udplisten\_Thread().

### 5.1.1.3 #define CONN\_THREADS 40

Definition at line 297 of file acc.c.

Referenced by acc\_Thread().

### 5.1.1.4 #define CONNECT\_THREAD\_EXIT()

**Value:**

```
free(arg); \
    MUTEX_LOCK(&acc->conn_thread_lock); \
    acc->num_conn_threads--; \
    COND_SIGNAL(&acc->conn_thread_cond); \
    MUTEX_UNLOCK(&acc->conn_thread_lock); \
    THREAD_EXIT();
```

Definition at line 389 of file acc.c.

Referenced by acc\_connection\_Thread().

### 5.1.1.5 #define UDPPORT 8866

Definition at line 915 of file acc.c.

Referenced by udplisten\_Thread().

## 5.1.2 Function Documentation

### 5.1.2.1 void\* acc\_connection\_Thread (void \*arg)

Definition at line 399 of file acc.c.

References mc\_platform\_s::acc, fipa\_agent\_identifier\_s::addresses, agent\_mailbox\_Post(), mc\_platform\_s::agent\_queue, AGENT\_UPDATE, CANCEL, CONNECT\_THREAD\_EXIT, connection\_Destroy(), mtp\_http\_s::content, mtp\_http\_content\_s::data, FIPA\_ACL, fipa\_acl\_envelope\_Destroy(), fipa\_acl\_envelope\_New(), fipa\_acl\_message\_Destroy(), fipa\_acl\_message\_New(), fipa\_acl\_Parse(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_envelope\_Parse(), fipa\_message\_string\_Destroy(), fipa\_message\_string\_New(), mtp\_http\_s::http\_performative, HTTP\_POST, HTTP\_PUT, agent\_s::mailbox, agent\_s::mc\_platform, mc\_platform, fipa\_message\_string\_s::message, message\_s::message\_body,

message\_Destroy(), message\_New(), mtp\_http\_s::message\_parts, mc\_platform\_s::message\_queue, message\_s::message\_type, message\_xml\_parse(), MOBILE\_AGENT, mtp\_http\_Destroy(), mtp\_http\_InitializeFromConnection(), mtp\_http\_New(), MXML\_NO\_CALLBACK, mxxmlLoadString(), N\_UNDRSTD, fipa\_agent\_identifier\_s::name, fipa\_agent\_identifier\_set\_s::num, fipa\_acl\_envelope\_s::num\_params, fipa\_acl\_envelope\_s::params, fipa\_message\_string\_s::parse, mc\_platform\_s::port, mc\_platform\_s::private\_key, QUER\_IF, QUER\_REF, RELAY, REQUEST, RETURN\_MSG, fipa\_url\_s::str, strdup, SUBSCRIBE, mtp\_http\_s::target, fipa\_acl\_Param\_s::to, fipa\_url\_sequence\_s::urls, and message\_s::xml\_root.

Referenced by acc\_Thread().

#### 5.1.2.2 int acc\_Destroy (acc\_p acc)

Definition at line 102 of file acc.c.

References COND\_DESTROY, MC\_SUCCESS, and MUTEX\_DESTROY.

Referenced by mc\_platform\_Destroy().

#### 5.1.2.3 acc\_p acc\_Initialize (struct mc\_platform\_s \* mc\_platform)

Definition at line 77 of file acc.c.

References COND\_INIT, COND\_T, MUTEX\_INIT, and MUTEX\_T.

Referenced by mc\_platform\_Initialize().

#### 5.1.2.4 void\* acc\_MessageHandlerThread (void \* arg)

Definition at line 118 of file acc.c.

References mc\_platform\_s::acc, agent\_Initialize(), mc\_platform\_s::agent\_queue, agent\_s::agent\_status, AGENT\_UPDATE, mc\_platform\_s::ams, CANCEL, COND\_BROADCAST, COND\_WAIT, agent\_s::datastate, FIPA\_ACL, mc\_platform\_s::giant, mc\_platform\_s::giant\_cond, mc\_platform\_s::giant\_lock, agent\_s::lock, MC\_AGENT\_NEUTRAL, mc\_platform, MC\_RECV\_AGENT, MC\_RECV\_MESSAGE, MC\_RECV\_RETURN, mc\_platform\_s::MC\_signal, mc\_platform\_s::MC\_signal\_cond, mc\_platform\_s::MC\_signal\_lock, message\_Destroy(), mc\_platform\_s::message\_queue, message\_Send(), message\_s::message\_type, MOBILE\_AGENT, MUTEX\_LOCK, MUTEX\_UNLOCK, N\_UNDRSTD, agent\_s::name, agent\_datastate\_s::persistent, QUER\_IF, QUER\_REF, mc\_platform\_s::quit, mc\_platform\_s::quit\_lock, RELAY, REQUEST, RETURN\_MSG, SUBSCRIBE, THREAD\_EXIT, and message\_s::to\_address.

Referenced by acc\_Start().

#### 5.1.2.5 void acc\_Start (mc\_platform\_p mc\_platform)

Definition at line 575 of file acc.c.

References mc\_platform\_s::acc, acc\_MessageHandlerThread(), acc\_Thread(), listen\_Thread(), MC\_THREAD\_ACC, mc\_platform\_s::stack\_size, THREAD\_CREATE, and udplisten\_Thread().

Referenced by mc\_platform\_Initialize().

**5.1.2.6 void\* acc\_Thread (void \* arg)**

Definition at line 300 of file acc.c.

References mc\_platform\_s::acc, acc\_connection\_Thread(), COND\_BROADCAST, COND\_WAIT, CONN\_THREADS, mc\_platform\_s::connection\_queue, mc\_platform\_s::giant, mc\_platform\_s::giant\_cond, mc\_platform\_s::giant\_lock, mc\_platform, MC\_RECV\_CONNECTION, mc\_platform\_s::MC\_signal, mc\_platform\_s::MC\_signal\_cond, mc\_platform\_s::MC\_signal\_lock, MUTEX\_LOCK, MUTEX\_UNLOCK, mc\_platform\_s::quit, mc\_platform\_s::quit\_lock, THREAD\_CREATE, THREAD\_DETACH, and THREAD\_EXIT.

Referenced by acc\_Start().

**5.1.2.7 int auth\_conn\_rece\_key (int sockfd, char \* peer\_name, int \* nonce, unsigned char \* aes\_key, char \* privkey, char \* known\_host\_filename)**

Definition at line 624 of file acc.c.

References read\_known\_host\_file(), and reply\_migration\_process().

Referenced by listen\_Thread().

**5.1.2.8 void\* listen\_Thread (void \* arg)**

Definition at line 663 of file acc.c.

References mc\_platform\_s::acc, connection\_s::addr, connection\_s::AES\_key, mc\_platform\_s::agency, auth\_conn\_rece\_key(), BACKLOG, mc\_platform\_s::bluetooth, connection\_s::clientfd, COND\_BROADCAST, connection\_s::connect\_id, connection\_New(), mc\_platform\_s::connection\_queue, agency\_s::known\_host\_filename, mc\_platform, MUTEX\_LOCK, MUTEX\_UNLOCK, connection\_s::nonce, mc\_platform\_s::port, agency\_s::portno, mc\_platform\_s::private\_key, mc\_platform\_s::quit, connection\_s::remote\_hostname, connection\_s::serverfd, SOCKET\_ERROR, mc\_platform\_s::sockfd, and THREAD\_EXIT.

Referenced by acc\_Start().

**5.1.2.9 void\* udplisten\_Thread (void \* arg)**

Definition at line 918 of file acc.c.

References buf, BUFLen, mc\_platform\_s::hostname, mc\_platform, PACKAGE\_VERSION, mc\_platform\_s::port, and UDPPORT.

Referenced by acc\_Start().

## 5.2 /home/dko/Projects/mobilec/src/agent.c File Reference

```
#include <unistd.h>
#include "config.h"
#include <embedch.h>
#include "include/libmc.h"
#include "macros.h"
#include "agent_datastate.h"
#include "agent_task.h"
#include "libmc.h"
#include "message.h"
#include "data_structures.h"
#include "agent_mailbox.h"
#include "include/mc_platform.h"
#include <ch.h>
#include "include/xml_parser.h"
```

### Functions

- [int agent\\_AddPersistentVariable](#) ([agent\\_p](#) agent, [int](#) task\_num, const char \*var\_name)
- [agent\\_p agent\\_Copy](#) (const [agent\\_p](#) agent)
- [agent\\_p agent\\_New](#) (void)
- [agent\\_p agent\\_NewBinary](#) (struct [mc\\_platform\\_s](#) \*mc\_platform)
- [agent\\_p agent\\_Initialize](#) (struct [mc\\_platform\\_s](#) \*mc\_platform, [message\\_p](#) message, [int](#) id)
- [int agent\\_Destroy](#) ([agent\\_p](#) agent)
- [void agent\\_RunChScript](#) ([agent\\_p](#) agent, [mc\\_platform\\_p](#) mc\_platform)
- [void \\* agent\\_ChScriptInitVar](#) ([ChInterp\\_t](#) \*interp)
- [void \\* agent\\_RunChScriptThread](#) (void \*ChAgent)

### 5.2.1 Function Documentation

#### 5.2.1.1 [int agent\\_AddPersistentVariable](#) ([agent\\_p](#) agent, [int](#) task\_num, const char \* var\_name)

Definition at line 52 of file agent.c.

References [agent\\_s::agent\\_interp](#), [agent\\_task\\_s::agent\\_variable\\_list](#), [interpreter\\_variable\\_data\\_s::array\\_dim](#), [interpreter\\_variable\\_data\\_s::array\\_extent](#), [CH\\_DATATYPE\\_SIZE](#), [CHECK\\_NULL](#), [interpreter\\_variable\\_data\\_s::data](#), [interpreter\\_variable\\_data\\_s::data\\_type](#), [agent\\_s::datastate](#), [MC\\_ERR](#), [agent\\_s::name](#), [interpreter\\_variable\\_data\\_s::name](#), [interpreter\\_variable\\_data\\_s::size](#), [size](#), [strdup](#), [agent\\_datastate\\_s::task\\_progress](#), and [agent\\_datastate\\_s::tasks](#).

#### 5.2.1.2 [void\\* agent\\_ChScriptInitVar](#) ([ChInterp\\_t](#) \* interp)

Definition at line 491 of file agent.c.



References MC\_AclAddReceiver\_chdl(), MC\_AclAddReplyTo\_chdl(), MC\_AclDestroy\_chdl(), MC\_AclGetContent\_chdl(), MC\_AclGetConversationID\_chdl(), MC\_AclGetPerformative\_chdl(), MC\_AclGetProtocol\_chdl(), MC\_AclGetSender\_chdl(), MC\_AclNew\_chdl(), MC\_AclPost\_chdl(), MC\_AclReply\_chdl(), MC\_AclRetrieve\_chdl(), MC\_AclSend\_chdl(), MC\_AclSetContent\_chdl(), MC\_AclSetConversationID\_chdl(), MC\_AclSetPerformative\_chdl(), MC\_AclSetProtocol\_chdl(), MC\_AclSetSender\_chdl(), MC\_AclWaitRetrieve\_chdl(), MC\_AddAgent\_chdl(), MC\_AgentAddTask\_chdl(), MC\_AgentVariableRetrieve\_chdl(), MC\_AgentVariableSave\_chdl(), MC\_Barrier\_chdl(), MC\_BarrierDelete\_chdl(), MC\_BarrierInit\_chdl(), MC\_CallAgentFunc\_chdl(), MC\_ComposeAgent\_chdl(), MC\_ComposeAgentFromFile\_chdl(), MC\_ComposeAgentFromFileWithWorkgroup\_chdl(), MC\_ComposeAgentWithWorkgroup\_chdl(), MC\_CondBroadcast\_chdl(), MC\_CondReset\_chdl(), MC\_CondSignal\_chdl(), MC\_CondWait\_chdl(), MC\_DeleteAgent\_chdl(), MC\_DeleteAgentWG\_chdl(), MC\_DeregisterService\_chdl(), MC\_DestroyServiceSearchResult\_chdl(), MC\_End\_chdl(), MC\_FindAgentByID\_chdl(), MC\_FindAgentByName\_chdl(), MC\_GetAgentID\_chdl(), MC\_GetAgentName\_chdl(), MC\_GetAgentStatus\_chdl(), MC\_GetAgentXMLString\_chdl(), MC\_GetTimeOfDay\_chdl(), MC\_HaltAgency\_chdl(), MC\_MigrateAgent\_chdl(), MC\_MutexLock\_chdl(), MC\_MutexUnlock\_chdl(), MC\_PrintAgentCode\_chdl(), MC\_RegisterService\_chdl(), MC\_ResumeAgency\_chdl(), MC\_RetrieveAgent\_chdl(), MC\_RetrieveAgentCode\_chdl(), MC\_SaveData\_chdl(), MC\_SearchForService\_chdl(), MC\_SemaphorePost\_chdl(), MC\_SemaphoreWait\_chdl(), MC\_SendAgentMigrationMessage\_chdl(), MC\_SendAgentMigrationMessageFile\_chdl(), MC\_SendSteerCommand\_chdl(), MC\_SetAgentStatus\_chdl(), MC\_SetDefaultAgentStatus\_chdl(), MC\_SyncDelete\_chdl(), MC\_SyncInit\_chdl(), MC\_TerminateAgent\_chdl(), and MC\_TerminateAgentWG\_chdl().

Referenced by AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), and mc\_platform\_Initialize().

### 5.2.1.3 agent\_p agent\_Copy (const agent\_p agent)

Definition at line 134 of file agent.c.

References agent\_datastate\_Copy(), agent\_s::agent\_interp, agent\_mailbox\_New(), agent\_s::agent\_persistent, agent\_s::agent\_status, agent\_s::agent\_type, agent\_s::arrival\_time, agent\_s::datastate, agent\_s::home, agent\_s::home\_port, agent\_s::id, agent\_s::lock, agent\_s::mailbox, MUTEX\_INIT, MUTEX\_LOCK, MUTEX\_T, agent\_s::name, agent\_s::orphan, agent\_s::owner, agent\_s::return\_data, and agent\_s::run\_lock.

Referenced by MC\_CopyAgent().

### 5.2.1.4 int agent\_Destroy (agent\_p agent)

Definition at line 413 of file agent.c.

References agent\_datastate\_Destroy(), agent\_s::agent\_interp, agent\_mailbox\_Destroy(), agent\_s::agent\_status, agent\_s::datastate, agent\_s::home, mc\_platform\_s::interpreter\_queue, agent\_s::lock, agent\_s::mailbox, MC\_AGENT\_NEUTRAL, agent\_s::mc\_platform, MC\_SUCCESS, MUTEX\_DESTROY, MUTEX\_LOCK, agent\_s::name, agent\_s::owner, agent\_s::run\_lock, agent\_s::sender, and agent\_s::wg\_code.

Referenced by agent\_Initialize().

### 5.2.1.5 agent\_p agent\_Initialize (struct mc\_platform\_s \* mc\_platform, message\_p message, int id)

Definition at line 292 of file agent.c.

References agent\_s::agent\_address, agent\_datastate\_New(), agent\_Destroy(), agent\_mailbox\_New(), agent\_s::agent\_pipe\_active, agent\_s::agent\_pipe\_ready\_to\_read, agent\_s::agent\_ready\_to\_send,

agent\_s::agent\_script\_ready, agent\_s::agent\_status, agent\_s::agent\_thread\_id, agent\_s::agent\_type, message\_s::agent\_xml\_flag, agent\_xml\_parse(), agent\_s::arrival\_time, agent\_s::datastate, mc\_platform\_s::default\_agentstatus, mc\_platform\_s::err, agent\_s::home, mc\_platform\_s::hostname, agent\_s::id, agent\_s::lock, agent\_s::mailbox, MC\_ERR\_PARSE, agent\_s::mc\_platform, MC\_REMOTE\_AGENT, MC\_RETURN\_AGENT, MC\_WAIT\_CH, message\_s::message\_type, MOBILE\_AGENT, MUTEX\_DESTROY, MUTEX\_INIT, MUTEX\_T, agent\_s::orphan, mc\_platform\_s::port, RETURN\_MSG, agent\_s::run\_lock, agent\_s::sender, agent\_datastate\_s::xml\_agent\_root, message\_s::xml\_payload, message\_s::xml\_root, and agent\_datastate\_s::xml\_root.

Referenced by acc\_MessageHandlerThread(), MC\_SendAgent(), MC\_SendAgentFile(), and MC\_SendAgentMigrationMessageFile().

#### 5.2.1.6 agent\_p agent\_New (void)

Definition at line 197 of file agent.c.

References agent\_s::lock, MUTEX\_INIT, MUTEX\_NEW, and agent\_s::run\_lock.

Referenced by MC\_ComposeAgentWithWorkgroup().

#### 5.2.1.7 agent\_p agent\_NewBinary (struct mc\_platform\_s \* mc\_platform)

Definition at line 220 of file agent.c.

References agent\_s::agent\_address, agent\_mailbox\_New(), agent\_s::agent\_pipe\_active, agent\_s::agent\_pipe\_ready\_to\_read, agent\_s::agent\_ready\_to\_send, agent\_s::agent\_script\_ready, agent\_s::agent\_status, agent\_s::agent\_thread\_id, agent\_s::arrival\_time, agent\_s::binary, agent\_s::home, mc\_platform\_s::hostname, agent\_s::id, agent\_s::lock, agent\_s::mailbox, MC\_AGENT\_ACTIVE, agent\_s::mc\_platform, MUTEX\_INIT, MUTEX\_T, agent\_s::orphan, mc\_platform\_s::port, agent\_s::run\_lock, and agent\_s::sender.

Referenced by MC\_AddStationaryAgent().

#### 5.2.1.8 void agent\_RunChScript (agent\_p agent, mc\_platform\_p mc\_platform)

Definition at line 453 of file agent.c.

References agent\_RunChScriptThread(), agent\_s::agent\_status, agent\_s::agent\_thread, MC\_AGENT\_ACTIVE, agent\_s::mc\_platform, MC\_THREAD\_AGENT, mc\_platform\_s::stack\_size, and THREAD\_CREATE.

Referenced by ams\_ManageAgentList().

#### 5.2.1.9 void\* agent\_RunChScriptThread (void \* ChAgent)

Definition at line 971 of file agent.c.

References mc\_platform\_s::agency, agent\_s::agent\_interp, agency\_s::agentInitCallback, agency\_s::agentInitUserData, mc\_platform\_s::ams, COND\_SIGNAL, agent\_s::datastate, agent\_s::id, mc\_platform\_s::interp\_options, mc\_platform\_s::interpreter\_queue, interpreter\_queue\_CreateRetrieve(), interpreter\_variable\_data\_Destroy(), interpreter\_variable\_data\_Initialize(), interpreter\_variable\_data\_InitializeFromAgent(), MC\_AGENT\_NEUTRAL, MC\_EXEC\_AGENT, agent\_s::mc\_platform, mc\_platform, MC\_RETURN\_AGENT, mc\_platform\_s::MC\_signal, mc\_platform\_s::MC\_signal\_cond, mc\_platform\_s::MC\_signal\_lock, MC\_WAIT\_FINISHED, MC\_WAIT\_MESSGSEND, MUTEX\_LOCK, MUTEX\_UNLOCK, agent\_s::name, SIGNAL, agent\_datastate\_s::task\_progress, and WARN.

Referenced by agent\_RunChScript().

## 5.3 /home/dko/Projects/mobilec/src/agent\_datastate.c File Reference

```
#include "config.h"
#include <mxml.h>
#include "include/agent_datastate.h"
#include "agent_task.h"
#include "include/macros.h"
#include "include/mc_error.h"
```

### Functions

- [agent\\_datastate\\_p agent\\_datastate\\_Copy](#) (const [agent\\_datastate\\_p](#) datastate)
- [agent\\_datastate\\_p agent\\_datastate\\_New](#) (void)
- [int agent\\_datastate\\_Destroy](#) ([agent\\_datastate\\_p](#) agent\_datastate)

### 5.3.1 Function Documentation

#### 5.3.1.1 [agent\\_datastate\\_p agent\\_datastate\\_Copy](#) (const [agent\\_datastate\\_p](#) datastate)

Definition at line 47 of file `agent_datastate.c`.

References `agent_datastate_s::agent_code`, `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_datastate_New()`, `agent_task_Copy()`, `agent_datastate_s::init_agent_status`, `agent_datastate_s::number_of_tasks`, `agent_datastate_s::persistent`, `agent_datastate_s::return_data`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

Referenced by `agent_Copy()`.

#### 5.3.1.2 [int agent\\_datastate\\_Destroy](#) ([agent\\_datastate\\_p](#) agent\_datastate)

Definition at line 136 of file `agent_datastate.c`.

References `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_task_Destroy()`, `MC_SUCCESS`, `mxmlDelete()`, `agent_datastate_s::number_of_tasks`, `agent_datastate_s::tasks`, and `agent_datastate_s::xml_root`.

Referenced by `agent_Destroy()`.

#### 5.3.1.3 [agent\\_datastate\\_p agent\\_datastate\\_New](#) (void)

Definition at line 115 of file `agent_datastate.c`.

References `agent_datastate_s::agent_code`, `CHECK_NULL`, `agent_datastate_s::init_agent_status`, `agent_datastate_s::number_of_tasks`, `agent_datastate_s::persistent`, `agent_datastate_s::progress_modifier`, `agent_datastate_s::return_data`, `agent_datastate_s::task_progress`, `agent_datastate_s::tasks`, `agent_datastate_s::xml_agent_root`, and `agent_datastate_s::xml_root`.

Referenced by `agent_datastate_Copy()`, `agent_Initialize()`, and `MC_ComposeAgentWithWorkgroup()`.

## 5.4 /home/dko/Projects/mobilec/src/agent\_mailbox.c File Reference

```
#include "config.h"
#include "include/data_structures.h"
```

### Functions

- [agent\\_mailbox\\_p agent\\_mailbox\\_New](#) (void)
- [agent\\_mailbox\\_p agent\\_mailbox\\_Copy](#) (agent\_mailbox\_p src)
- [int agent\\_mailbox\\_Destroy](#) (agent\_mailbox\_t \*mailbox)
- [int agent\\_mailbox\\_Post](#) (agent\_mailbox\_p mailbox, [fipa\\_acl\\_message\\_t](#) \*message)
- [fipa\\_acl\\_message\\_t \\* agent\\_mailbox\\_Retrieve](#) (agent\_mailbox\_p mailbox)
- [fipa\\_acl\\_message\\_t \\* agent\\_mailbox\\_WaitRetrieve](#) (agent\_mailbox\_p mailbox)

### 5.4.1 Function Documentation

#### 5.4.1.1 agent\_mailbox\_p agent\_mailbox\_Copy (agent\_mailbox\_p src)

Definition at line 21 of file agent\_mailbox.c.

References [agent\\_mailbox\\_New\(\)](#), and [agent\\_mailbox\\_s::mail\\_queue](#).

#### 5.4.1.2 int agent\_mailbox\_Destroy (agent\_mailbox\_t \* mailbox)

Definition at line 29 of file agent\_mailbox.c.

References [agent\\_mailbox\\_s::mail\\_queue](#).

Referenced by [agent\\_Destroy\(\)](#).

#### 5.4.1.3 agent\_mailbox\_p agent\_mailbox\_New (void)

Definition at line 12 of file agent\_mailbox.c.

References [agent\\_mailbox\\_s::mail\\_queue](#).

Referenced by [agent\\_Copy\(\)](#), [agent\\_Initialize\(\)](#), [agent\\_mailbox\\_Copy\(\)](#), and [agent\\_NewBinary\(\)](#).

#### 5.4.1.4 int agent\_mailbox\_Post (agent\_mailbox\_p mailbox, fipa\_acl\_message\_t \* message)

Definition at line 38 of file agent\_mailbox.c.

References [agent\\_mailbox\\_s::mail\\_queue](#).

Referenced by [acc\\_connection\\_Thread\(\)](#), and [MC\\_AclPost\(\)](#).

#### 5.4.1.5 fipa\_acl\_message\_t\* agent\_mailbox\_Retrieve (agent\_mailbox\_p mailbox)

Definition at line 44 of file agent\_mailbox.c.

References [agent\\_mailbox\\_s::mail\\_queue](#).

Referenced by [agent\\_mailbox\\_WaitRetrieve\(\)](#), and [MC\\_AclRetrieve\(\)](#).

**5.4.1.6 fipa\_acl\_message\_t\* agent\_mailbox\_WaitRetrieve (agent\_mailbox\_p *mailbox*)**

Definition at line 49 of file agent\_mailbox.c.

References agent\_mailbox\_Retrieve(), COND\_WAIT, agent\_mailbox\_s::mail\_queue, MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_AclWaitRetrieve().

## 5.5 /home/dko/Projects/mobilec/src/agent\_return\_data.c File Reference

```
#include "config.h"
#include "include/interpreter_variable_data.h"
#include "include/agent.h"
```

### Functions

- [interpreter\\_variable\\_data\\_p interpreter\\_variable\\_data\\_New](#) (void)
- [interpreter\\_variable\\_data\\_p interpreter\\_variable\\_data\\_InitializeFromAgent](#) (agent\_p agent)
- [interpreter\\_variable\\_data\\_p interpreter\\_variable\\_data\\_Initialize](#) (agent\_p agent, const char \*varname)
- [int interpreter\\_variable\\_data\\_Destroy](#) (interpreter\_variable\_data\_p agent\_variable\_data)
- [interpreter\\_variable\\_data\\_p interpreter\\_variable\\_data\\_Copy](#) (interpreter\_variable\_data\_p src)

### 5.5.1 Function Documentation

#### 5.5.1.1 [interpreter\\_variable\\_data\\_p interpreter\\_variable\\_data\\_Copy](#) ([interpreter\\_variable\\_data\\_p](#) src)

Definition at line 235 of file agent\_return\_data.c.

References [interpreter\\_variable\\_data\\_s::array\\_dim](#), [interpreter\\_variable\\_data\\_s::array\\_extent](#), [interpreter\\_variable\\_data\\_s::data](#), [interpreter\\_variable\\_data\\_s::data\\_type](#), [interpreter\\_variable\\_data\\_New\(\)](#), [interpreter\\_variable\\_data\\_s::name](#), [interpreter\\_variable\\_data\\_s::size](#), and [strdup](#).

Referenced by [agent\\_task\\_Copy\(\)](#).

#### 5.5.1.2 [int interpreter\\_variable\\_data\\_Destroy](#) ([interpreter\\_variable\\_data\\_p](#) agent\_variable\_data)

Definition at line 216 of file agent\_return\_data.c.

References [interpreter\\_variable\\_data\\_s::array\\_extent](#), [interpreter\\_variable\\_data\\_s::data](#), [MC\\_SUCCESS](#), and [interpreter\\_variable\\_data\\_s::name](#).

Referenced by [agent\\_RunChScriptThread\(\)](#), and [agent\\_task\\_Destroy\(\)](#).

#### 5.5.1.3 [interpreter\\_variable\\_data\\_p interpreter\\_variable\\_data\\_Initialize](#) ([agent\\_p](#) agent, const char \* varname)

Definition at line 145 of file agent\_return\_data.c.

References [agent\\_s::agent\\_interp](#), [interpreter\\_variable\\_data\\_s::array\\_dim](#), [interpreter\\_variable\\_data\\_s::array\\_extent](#), [CH\\_DATATYPE\\_SIZE](#), [CHECK\\_NULL](#), [interpreter\\_variable\\_data\\_s::data](#), [interpreter\\_variable\\_data\\_s::data\\_type](#), [MUTEX\\_LOCK](#), [MUTEX\\_UNLOCK](#), [interpreter\\_variable\\_data\\_s::name](#), [agent\\_s::run\\_lock](#), [interpreter\\_variable\\_data\\_s::size](#), [size](#), and [strdup](#).

Referenced by [agent\\_RunChScriptThread\(\)](#).

#### 5.5.1.4 `interpreter_variable_data_p` `interpreter_variable_data_InitializeFromAgent` (`agent_p agent`)

Definition at line 61 of file `agent_return_data.c`.

References `agent_s::agent_interp`, `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `CH_DATATYPE_SIZE`, `CHECK_NULL`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, `agent_s::datastate`, `interpreter_variable_data_s::name`, `interpreter_variable_data_s::size`, `size`, `strdup`, `agent_datastate_s::task_progress`, `agent_datastate_s::tasks`, and `agent_task_s::var_name`.

Referenced by `agent_RunChScriptThread()`.

#### 5.5.1.5 `interpreter_variable_data_p` `interpreter_variable_data_New` (`void`)

Definition at line 46 of file `agent_return_data.c`.

References `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `CHECK_NULL`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, `interpreter_variable_data_s::name`, and `interpreter_variable_data_s::size`.

Referenced by `agent_xml_parse__data()`, `interpreter_variable_data_Copy()`, and `MC_SaveData_chdl()`.



## 5.6 /home/dko/Projects/mobilec/src/agent\_task.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "include/agent_task.h"
#include "interpreter_variable_data.h"
#include "data_structures.h"
#include "include/mc_error.h"
```

### Functions

- [agent\\_task\\_p agent\\_task\\_New](#) (void)
- [agent\\_task\\_p agent\\_task\\_Copy](#) (agent\_task\_p task)
- [int agent\\_task\\_Destroy](#) (agent\_task\_p agent\_task)

#### 5.6.1 Function Documentation

##### 5.6.1.1 agent\_task\_p agent\_task\_Copy (agent\_task\_p task)

Definition at line 66 of file agent\_task.c.

References [agent\\_task\\_s::agent\\_return\\_data](#), [agent\\_task\\_s::agent\\_variable\\_list](#), [agent\\_task\\_s::code\\_id](#), [agent\\_task\\_s::init\\_agent\\_status](#), [interpreter\\_variable\\_data\\_Copy\(\)](#), [ListAdd\(\)](#), [ListSearch\(\)](#), [agent\\_task\\_s::num\\_saved\\_variables](#), [agent\\_task\\_s::number\\_of\\_elements](#), [agent\\_task\\_s::persistent](#), [agent\\_task\\_s::saved\\_variables](#), [agent\\_task\\_s::server\\_name](#), [agent\\_task\\_s::size\\_of\\_element\\_array](#), [strdup](#), and [agent\\_task\\_s::var\\_name](#).

Referenced by [agent\\_datastate\\_Copy\(\)](#).

##### 5.6.1.2 int agent\_task\_Destroy (agent\_task\_p agent\_task)

Definition at line 132 of file agent\_task.c.

References [agent\\_task\\_s::agent\\_return\\_data](#), [agent\\_task\\_s::agent\\_variable\\_list](#), [agent\\_task\\_s::code\\_id](#), [interpreter\\_variable\\_data\\_Destroy\(\)](#), [MC\\_SUCCESS](#), [agent\\_task\\_s::saved\\_variables](#), [agent\\_task\\_s::server\\_name](#), and [agent\\_task\\_s::var\\_name](#).

Referenced by [agent\\_datastate\\_Destroy\(\)](#).

##### 5.6.1.3 agent\_task\_p agent\_task\_New (void)

Definition at line 48 of file agent\_task.c.

References [agent\\_task\\_s::agent\\_variable\\_list](#), [agent\\_task\\_s::num\\_saved\\_variables](#), and [agent\\_task\\_s::saved\\_variables](#).

Referenced by [agent\\_xml\\_parse\\_\\_tasks\(\)](#), [MC\\_AgentAddTask\(\)](#), and [MC\\_ComposeAgentWithWorkgroup\(\)](#).

## 5.7 /home/dko/Projects/mobilec/src/ams.c File Reference

```
#include "config.h"
#include "include/ams.h"
#include "macros.h"
#include "include/agent.h"
#include "include/data_structures.h"
#include "include/mc_platform.h"
```

### Functions

- [int ams\\_Destroy](#) (ams\_p ams)
- [ams\\_p ams\\_Initialize](#) (mc\_platform\_p mc\_platform)
- [void ams\\_Print](#) (ams\_p ams)
- [int ams\\_ManageAgentList](#) (ams\_p ams)
- [void ams\\_Start](#) (mc\_platform\_p mc\_platform)
- [void \\* ams\\_Thread](#) (void \*arg)

### 5.7.1 Function Documentation

#### 5.7.1.1 int ams\_Destroy (ams\_p ams)

Definition at line 46 of file ams.c.

References COND\_DESTROY, MC\_SUCCESS, and MUTEX\_DESTROY.

Referenced by mc\_platform\_Destroy().

#### 5.7.1.2 ams\_p ams\_Initialize (mc\_platform\_p mc\_platform)

Definition at line 61 of file ams.c.

References CHECK\_NULL, COND\_INIT, COND\_T, MUTEX\_INIT, and MUTEX\_T.

Referenced by mc\_platform\_Initialize().

#### 5.7.1.3 int ams\_ManageAgentList (ams\_p ams)

Definition at line 120 of file ams.c.

References [agent\\_RunChScript\(\)](#), [agent\\_s::agent\\_status](#), [agent\\_s::binary](#), [ListSearch\(\)](#), [agent\\_s::lock](#), [MC\\_AGENT\\_ACTIVE](#), [MC\\_AGENT\\_NEUTRAL](#), [MC\\_TerminateAgent\(\)](#), [MC\\_WAIT\\_CH](#), [MC\\_WAIT\\_FINISHED](#), [MC\\_WAIT\\_MESSGSEND](#), [message\\_Destroy\(\)](#), [message\\_InitializeFromAgent\(\)](#), [message\\_New\(\)](#), [MUTEX\\_LOCK](#), [MUTEX\\_UNLOCK](#), [agent\\_s::name](#), [agent\\_s::orphan](#), [mc\\_platform\\_s::quit](#), [mc\\_platform\\_s::quit\\_lock](#), and [agent\\_s::run\\_lock](#).

Referenced by [ams\\_Thread\(\)](#).

**5.7.1.4 void ams\_Print (ams\_p *ams*)**

Definition at line 88 of file ams.c.

References agent\_s::agent\_status, agent\_s::connect\_id, agent\_s::id, ListSearch(), MUTEX\_LOCK, and MUTEX\_UNLOCK.

**5.7.1.5 void ams\_Start (mc\_platform\_p *mc\_platform*)**

Definition at line 227 of file ams.c.

References mc\_platform\_s::ams, ams\_Thread(), MC\_THREAD\_AMS, mc\_platform\_s::stack\_size, and THREAD\_CREATE.

Referenced by mc\_platform\_Initialize().

**5.7.1.6 void\* ams\_Thread (void \* *arg*)**

Definition at line 258 of file ams.c.

References mc\_platform\_s::ams, ams\_ManageAgentList(), COND\_BROADCAST, COND\_WAIT, mc\_platform, MUTEX\_LOCK, MUTEX\_UNLOCK, mc\_platform\_s::quit, mc\_platform\_s::quit\_lock, and THREAD\_EXIT.

Referenced by ams\_Start().

## 5.8 /home/dko/Projects/mobilec/src/barrier.c File Reference

```
#include "config.h"
#include "include/barrier.h"
#include "macros.h"
#include "../mc_list/list.h"
#include "mc_rwlock.h"
#include "include/mc_error.h"
```

### Functions

- [barrier\\_node\\_p barrier\\_node\\_Initialize](#) ([int id](#), [int num\\_registered](#))
- [int barrier\\_node\\_Destroy](#) ([barrier\\_node\\_p node](#))
- [int barrier\\_queue\\_Add](#) ([barrier\\_queue\\_p list](#), [barrier\\_node\\_p node](#))
- [int barrier\\_queue\\_Delete](#) ([int id](#), [barrier\\_queue\\_p list](#))
- [int barrier\\_queue\\_Destroy](#) ([barrier\\_queue\\_p queue](#))
- [barrier\\_node\\_p barrier\\_queue\\_Get](#) ([barrier\\_queue\\_p list](#), [int id](#))
- [barrier\\_queue\\_p barrier\\_queue\\_New](#) ([void](#))
- [barrier\\_node\\_p barrier\\_queue\\_Pop](#) ([barrier\\_queue\\_p queue](#))

### 5.8.1 Function Documentation

#### 5.8.1.1 [int barrier\\_node\\_Destroy](#) ([barrier\\_node\\_p node](#))

Definition at line 70 of file barrier.c.

References [barrier\\_node\\_s::cond](#), [COND\\_DESTROY](#), [barrier\\_node\\_s::lock](#), [MC\\_SUCCESS](#), and [MUTEX\\_DESTROY](#).

Referenced by [barrier\\_queue\\_Delete\(\)](#), and [barrier\\_queue\\_Destroy\(\)](#).

#### 5.8.1.2 [barrier\\_node\\_p barrier\\_node\\_Initialize](#) ([int id](#), [int num\\_registered](#))

Definition at line 45 of file barrier.c.

References [CHECK\\_NULL](#), [barrier\\_node\\_s::cond](#), [COND\\_INIT](#), [COND\\_T](#), [barrier\\_node\\_s::id](#), [barrier\\_node\\_s::lock](#), [MUTEX\\_INIT](#), [MUTEX\\_T](#), [node](#), [barrier\\_node\\_s::num\\_registered](#), and [barrier\\_node\\_s::num\\_waiting](#).

Referenced by [MC\\_BarrierInit\(\)](#).

#### 5.8.1.3 [int barrier\\_queue\\_Add](#) ([barrier\\_queue\\_p list](#), [barrier\\_node\\_p node](#))

Definition at line 87 of file barrier.c.

References [DATA](#), [barrier\\_node\\_s::id](#), [barrier\\_queue\\_s::list](#), [ListAdd\(\)](#), [list\\_s::listhead](#), [barrier\\_queue\\_s::lock](#), [MC\\_SUCCESS](#), [MC\\_WARN\\_DUPLICATE](#), [listNode\\_s::next](#), [listNode\\_s::node\\_data](#), [RWLOCK\\_WRLOCK](#), [RWLOCK\\_WRUNLOCK](#), and [barrier\\_queue\\_s::size](#).

Referenced by [MC\\_BarrierInit\(\)](#).

#### 5.8.1.4 int barrier\_queue\_Delete (int *id*, barrier\_queue\_p *list*)

Definition at line 111 of file barrier.c.

References barrier\_node\_Destroy(), barrier\_node\_s::id, barrier\_queue\_s::list, ListDelete(), ListSearch(), barrier\_queue\_s::lock, MC\_ERR\_NOT\_FOUND, MC\_SUCCESS, RWLOCK\_WRLOCK, RWLOCK\_WRUNLOCK, barrier\_queue\_s::size, and list\_s::size.

Referenced by MC\_BarrierDelete().

#### 5.8.1.5 int barrier\_queue\_Destroy (barrier\_queue\_p *queue*)

Definition at line 131 of file barrier.c.

References barrier\_node\_Destroy(), barrier\_queue\_Pop(), barrier\_queue\_s::list, ListTerminate(), barrier\_queue\_s::lock, MC\_SUCCESS, node, and RWLOCK\_DESTROY.

Referenced by mc\_platform\_Destroy().

#### 5.8.1.6 barrier\_node\_p barrier\_queue\_Get (barrier\_queue\_p *list*, int *id*)

Definition at line 145 of file barrier.c.

References barrier\_queue\_s::list, list\_s::listhead, barrier\_queue\_s::lock, listNode\_s::next, listNode\_s::node\_data, RWLOCK\_RDLOCK, and RWLOCK\_RDUNLOCK.

Referenced by MC\_Barrier(), and MC\_BarrierInit().

#### 5.8.1.7 barrier\_queue\_p barrier\_queue\_New (void)

Definition at line 162 of file barrier.c.

References CHECK\_NULL, barrier\_queue\_s::list, ListInitialize(), barrier\_queue\_s::lock, RWLOCK\_INIT, and RWLOCK\_T.

Referenced by mc\_platform\_Initialize().

#### 5.8.1.8 barrier\_node\_p barrier\_queue\_Pop (barrier\_queue\_p *queue*)

Definition at line 176 of file barrier.c.

References barrier\_queue\_s::list, ListPop(), and node.

Referenced by barrier\_queue\_Destroy().

## 5.9 /home/dko/Projects/mobilec/src/cmd\_prompt.c File Reference

```
#include <stdio.h>
#include <unistd.h>
#include "config.h"
#include <stdlib.h>
#include <string.h>
#include "include/cmd_prompt.h"
#include "macros.h"
#include "mc_platform.h"
#include "commands.x.h"
```

### Functions

- [cmd\\_prompt\\_p cmd\\_prompt\\_Initialize \(mc\\_platform\\_p mc\\_platform\)](#)
- [int cmd\\_prompt\\_Destroy \(cmd\\_prompt\\_p cmd\\_prompt\)](#)
- [void cmd\\_prompt\\_Start \(mc\\_platform\\_p mc\\_platform\)](#)
- [void \\* cmd\\_prompt\\_Thread \(void \\*arg\)](#)
- [int split\\_string \(char \\*\\*\\*args, const char \\*buf\)](#)
- [int process\\_command \(command\\_t \\*cmd\)](#)
- [int exec\\_command \(command\\_t cmd, mc\\_platform\\_p global\)](#)
- [int dealloc\\_command \(command\\_t \\*cmd\)](#)
- [int handler\\_QUIT \(void \\*arg, mc\\_platform\\_p global\)](#)
- [int handler\\_HELP \(void \\*arg, mc\\_platform\\_p global\)](#)
- [int handler\\_SEND \(void \\*arg, mc\\_platform\\_p global\)](#)
- [int handler\\_PRINT\\_CONNECTLIST \(void \\*arg, mc\\_platform\\_p global\)](#)
- [int handler\\_PRINTLIST\\_MESSAGE \(void \\*arg, mc\\_platform\\_p global\)](#)
- [int handler\\_PRINTLIST\\_AGENTS \(void \\*arg, mc\\_platform\\_p global\)](#)
- [int handler\\_FLUSH\\_AGENTS \(void \\*arg, mc\\_platform\\_p global\)](#)
- [int handler\\_COMPOSE\\_SEND \(void \\*arg, mc\\_platform\\_p global\)](#)

### 5.9.1 Function Documentation

#### 5.9.1.1 [int cmd\\_prompt\\_Destroy \(cmd\\_prompt\\_p \*cmd\\_prompt\*\)](#)

Definition at line 129 of file cmd\_prompt.c.

References MC\_SUCCESS.

Referenced by mc\_platform\_Destroy().

#### 5.9.1.2 [cmd\\_prompt\\_p cmd\\_prompt\\_Initialize \(mc\\_platform\\_p \*mc\\_platform\*\)](#)

Definition at line 121 of file cmd\_prompt.c.

Referenced by mc\_platform\_Initialize().

**5.9.1.3 void cmd\_prompt\_Start (mc\_platform\_p mc\_platform)**

Definition at line 136 of file cmd\_prompt.c.

References mc\_platform\_s::cmd\_prompt, cmd\_prompt\_Thread(), MC\_THREAD\_CP, mc\_platform\_s::stack\_size, cmd\_prompt\_s::thread, and THREAD\_CREATE.

Referenced by mc\_platform\_Initialize().

**5.9.1.4 void\* cmd\_prompt\_Thread (void \* arg)**

Definition at line 168 of file cmd\_prompt.c.

References command\_s::args, buf, dealloc\_command(), exec\_command(), command\_s::index, mc\_platform, command\_s::num\_args, process\_command(), and split\_string().

Referenced by cmd\_prompt\_Start().

**5.9.1.5 int dealloc\_command (command\_t \* cmd)**

Definition at line 317 of file cmd\_prompt.c.

References command\_s::args, and command\_s::num\_args.

Referenced by cmd\_prompt\_Thread().

**5.9.1.6 int exec\_command (command\_t cmd, mc\_platform\_p global)**

Definition at line 301 of file cmd\_prompt.c.

References command\_s::args, cmd\_handlers, command\_s::index, and command\_s::num\_args.

Referenced by cmd\_prompt\_Thread().

**5.9.1.7 int handler\_COMPOSE\_SEND (void \* arg, mc\_platform\_p global)**

Definition at line 414 of file cmd\_prompt.c.

References mc\_platform\_s::agency, command\_s::args, command\_descriptions, mc\_platform\_s::hostname, MC\_AddAgent(), MC\_ComposeAgentFromFile(), and command\_s::num\_args.

**5.9.1.8 int handler\_FLUSH\_AGENTS (void \* arg, mc\_platform\_p global)**

Definition at line 408 of file cmd\_prompt.c.

References mc\_platform\_s::agent\_queue, and agent\_queue\_Flush().

**5.9.1.9 int handler\_HELP (void \* arg, mc\_platform\_p global)**

Definition at line 341 of file cmd\_prompt.c.

References command\_s::args, command\_cmds, command\_descriptions, and command\_s::num\_args.

**5.9.1.10 int handler\_PRINT\_CONNECTLIST (void \* *arg*, mc\_platform\_p *global*)**

Definition at line 390 of file cmd\_prompt.c.

References mc\_platform\_s::connection\_queue.

**5.9.1.11 int handler\_PRINTLIST\_AGENTS (void \* *arg*, mc\_platform\_p *global*)**

Definition at line 402 of file cmd\_prompt.c.

References mc\_platform\_s::agent\_queue.

**5.9.1.12 int handler\_PRINTLIST\_MESSAGE (void \* *arg*, mc\_platform\_p *global*)**

Definition at line 396 of file cmd\_prompt.c.

References mc\_platform\_s::message\_queue.

**5.9.1.13 int handler\_QUIT (void \* *arg*, mc\_platform\_p *global*)**

Definition at line 332 of file cmd\_prompt.c.

References COND\_BROADCAST, MUTEX\_LOCK, MUTEX\_UNLOCK, mc\_platform\_s::quit, mc\_platform\_s::quit\_cond, and mc\_platform\_s::quit\_lock.

**5.9.1.14 int handler\_SEND (void \* *arg*, mc\_platform\_p *global*)**

Definition at line 379 of file cmd\_prompt.c.

References command\_s::args, command\_descriptions, MC\_SendAgentFile(), and command\_s::num\_args.

**5.9.1.15 int process\_command (command\_t \* *cmd*)**

Definition at line 283 of file cmd\_prompt.c.

References command\_s::args, command\_cmds, command\_s::index, and command\_s::num\_args.

Referenced by cmd\_prompt\_Thread().

**5.9.1.16 int split\_string (char \*\*\* *args*, const char \* *buf*)**

Definition at line 235 of file cmd\_prompt.c.

References int.

Referenced by cmd\_prompt\_Thread().



## 5.10 /home/dko/Projects/mobilec/src/connection.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "include/connection.h"
#include "include/mc_error.h"
#include "include/macros.h"
```

### Functions

- [int connection\\_Destroy](#) ([connection\\_p](#) connection)
- [connection\\_p connection\\_New](#) (void)
- [connection\\_p connection\\_Copy](#) ([connection\\_p](#) connection)

#### 5.10.1 Function Documentation

##### 5.10.1.1 [connection\\_p connection\\_Copy](#) ([connection\\_p](#) *connection*)

Definition at line 80 of file connection.c.

References [connection\\_s::addr](#), [connection\\_s::clientfd](#), [connection\\_s::connect\\_id](#), [connection\\_New\(\)](#), [connection\\_s::remote\\_hostname](#), [connection\\_s::serverfd](#), and [strdup](#).

##### 5.10.1.2 [int connection\\_Destroy](#) ([connection\\_p](#) *connection*)

Definition at line 48 of file connection.c.

References [connection\\_s::clientfd](#), [MC\\_SUCCESS](#), and [connection\\_s::remote\\_hostname](#).

Referenced by [acc\\_connection\\_Thread\(\)](#).

##### 5.10.1.3 [connection\\_p connection\\_New](#) (void)

Definition at line 69 of file connection.c.

Referenced by [connection\\_Copy\(\)](#), and [listen\\_Thread\(\)](#).

## 5.11 /home/dko/Projects/mobilec/src/data\_structures.c File Reference

```
#include "config.h"
#include "include/ap_queue_template.h"
#include "macros.h"
#include "mc_error.h"
#include <embedch.h>
#include "include/data_structures.h"
#include "include/interpreter_variable_data.h"
#include "include/agent.h"
```

### Functions

- [AP\\_QUEUE\\_SEARCH\\_TEMPLATE](#) (AP\_QUEUE\_REMOVE\_TEMPLATE(connection\_queue, [Search](#), AP\_QUEUE\_REMOVE\_TEMPLATE(connection, AP\_QUEUE\_REMOVE\_TEMPLATE(int,([node](#)->connect\_id==key)))
- [AP\\_QUEUE\\_SEARCH\\_TEMPLATE](#) (AP\_QUEUE\_REMOVE\_TEMPLATE(agent\_variable\_list, AP\_QUEUE\_REMOVE\_TEMPLATE([Search](#), AP\_QUEUE\_REMOVE\_TEMPLATE(interpreter\_variable\_data, char \*,(!strcmp([node](#)->name, key))))
- [AP\\_QUEUE\\_STD\\_DEFN\\_TEMPLATE](#) (AP\_QUEUE\_SEARCH\_TEMPLATE(agent\_queue, AP\_QUEUE\_SEARCH\_TEMPLATE(agent)
- [int agent\\_queue\\_Flush](#) (agent\_queue\_p queue)
- [AP\\_QUEUE\\_STD\\_DEFN\\_TEMPLATE](#) (fipa\_acl\_message\_p mail\_queue\_SearchReceivers(mail\_queue\_p mail\_queue, fipa\_acl\_message)
- [AP\\_QUEUE\\_STD\\_DEFN\\_TEMPLATE](#) (mailbox\_queue, agent\_mailbox)
- [int AP\\_GENERIC\\_Destroy](#) (AP\_GENERIC\_t \*blah)
- [AP\\_GENERIC\\_p AP\\_GENERIC\\_Copy](#) (AP\_GENERIC\_p AP\_GENERIC)

### 5.11.1 Function Documentation

#### 5.11.1.1 int agent\_queue\_Flush (agent\_queue\_p queue)

Definition at line 197 of file data\_structures.c.

References [agent\\_s::agent\\_status](#), [agent\\_s::id](#), [ListSearch\(\)](#), [MUTEX\\_LOCK](#), [MUTEX\\_UNLOCK](#), [agent\\_s::name](#), and [node](#).

Referenced by [handler\\_FLUSH\\_AGENTS\(\)](#).

#### 5.11.1.2 AP\_GENERIC\_p AP\_GENERIC\_Copy (AP\_GENERIC\_p AP\_GENERIC)

Definition at line 295 of file data\_structures.c.

#### 5.11.1.3 int AP\_GENERIC\_Destroy (AP\_GENERIC\_t \*blah)

Definition at line 291 of file data\_structures.c.

**5.11.1.4 AP\_QUEUE\_SEARCH\_TEMPLATE (AP\_QUEUE\_REMOVE\_TEMPLATE(  
*agent\_variable\_list*, AP\_QUEUE\_REMOVE\_TEMPLATE( *Search*,  
AP\_QUEUE\_REMOVE\_TEMPLATE( *interpreter\_variable\_data*, char \*,  
(!strcmp(node->name, key))))**

Definition at line 84 of file data\_structures.c.

References message\_s::from\_address, ListSearch(), message\_s::message\_id, MUTEX\_LOCK, MUTEX\_UNLOCK, node, and message\_s::to\_address.

**5.11.1.5 AP\_QUEUE\_SEARCH\_TEMPLATE (AP\_QUEUE\_REMOVE\_TEMPLATE(  
*connection\_queue*, Search, AP\_QUEUE\_REMOVE\_TEMPLATE( *connection*,  
AP\_QUEUE\_REMOVE\_TEMPLATE( *int*, (node->connect\_id==key))**

Definition at line 45 of file data\_structures.c.

References connection\_s::connect\_id, ListSearch(), MUTEX\_LOCK, MUTEX\_UNLOCK, node, and connection\_s::remote\_hostname.

**5.11.1.6 AP\_QUEUE\_STD\_DEFN\_TEMPLATE (mailbox\_queue, agent\_mailbox)**

Definition at line 253 of file data\_structures.c.

References agent\_ChScriptInitVar(), COND\_SIGNAL, ListPop(), MUTEX\_LOCK, and MUTEX\_UNLOCK.

**5.11.1.7 AP\_QUEUE\_STD\_DEFN\_TEMPLATE (fipa\_acl\_message\_p  
mail\_queue\_SearchReceivers( mail\_queue\_p mail\_queue, fipa\_acl\_message)**

Definition at line 214 of file data\_structures.c.

References fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, MUTEX\_LOCK, MUTEX\_UNLOCK, fipa\_agent\_identifier\_s::name, listNode\_s::next, node, listNode\_s::node\_data, and fipa\_acl\_message\_s::receiver.

**5.11.1.8 AP\_QUEUE\_STD\_DEFN\_TEMPLATE (AP\_QUEUE\_SEARCH\_TEMPLATE(  
*agent\_queue*, AP\_QUEUE\_SEARCH\_TEMPLATE( *agent*)**

Definition at line 145 of file data\_structures.c.

References agent\_s::agent\_status, agent\_s::id, ListSearch(), MUTEX\_LOCK, MUTEX\_UNLOCK, agent\_s::name, and node.

## 5.12 /home/dko/Projects/mobilec/src/df.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include "config.h"
#include "include/mc_platform.h"
#include "include/df.h"
#include "../mc_list/list.h"
#include "../include/macros.h"
#include "df_request.x.h"
```

### Defines

- #define [REQUEST](#)(name, string, description)

### Functions

- [int df\\_Add](#) (struct df\_s \*df, struct df\_node\_s \*[node](#))
- [int df\\_AddRequest](#) (struct df\_s \*df, struct df\_request\_list\_node\_s \*[node](#))
- [int df\\_Destroy](#) (df\_p df)
- df\_p [df\\_Initialize](#) ([mc\\_platform\\_p](#) mc\_platform)
- [int df\\_ProcessRequest](#) (struct [mc\\_platform\\_s](#) \*global)
- [int df\\_SearchForService](#) (df\_p df, const char \*searchstring, char \*\*\*agent\_names, char \*\*\*service\_names, [int](#) \*\*agent\_ids, [int](#) \*num\_entries)
- void [df\\_Start](#) ([mc\\_platform\\_p](#) mc\_platform)
- [int df\\_request\\_list\\_node\\_Destroy](#) (df\_request\_list\_node\_p [node](#))
- df\_request\_list\_node\_p [df\\_request\\_list\\_node\\_New](#) (void)
- [int df\\_request\\_list\\_Destroy](#) (df\_request\_list\_p df\_request\_list)
- df\_request\_list\_p [df\\_request\\_list\\_New](#) (void)
- df\_request\_list\_node\_p [df\\_request\\_list\\_Pop](#) (df\_request\_list\_p requests)
- df\_request\_search\_p [df\\_request\\_search\\_New](#) (void)
- [int df\\_request\\_search\\_Destroy](#) (df\_request\_search\_p [node](#))
- [int df\\_node\\_Destroy](#) (df\_node\_p df\_node)
- void \* [df\\_Thread](#) (void \*arg)
- [int request\\_handler\\_REGISTER](#) (struct [mc\\_platform\\_s](#) \*global, void \*data)
- [int request\\_handler\\_SEARCH](#) (struct [mc\\_platform\\_s](#) \*global, void \*data)
- [int request\\_handler\\_SUBSCRIBE](#) (struct [mc\\_platform\\_s](#) \*global, void \*data)
- [int request\\_handler\\_DEREGISTER](#) (struct [mc\\_platform\\_s](#) \*global, void \*data)

### 5.12.1 Define Documentation

#### 5.12.1.1 #define REQUEST(name, string, description)

Value:

```
if ( !strcmp(request->command, string ) ) { \
    return_code = MC_SUCCESS; \
    handler_code = request_handler_##name( \
        global, \
        request->data ); \
    request_code = REQUEST_##name; \
} else
```

Referenced by `acc_connection_Thread()`, and `acc_MessageHandlerThread()`.

## 5.12.2 Function Documentation

### 5.12.2.1 `int df_Add (struct df_s * df, struct df_node_s * node)`

Definition at line 50 of file `df.c`.

References `ListAdd()`, `MC_SUCCESS`, and `SIGNAL`.

Referenced by `request_handler_REGISTER()`.

### 5.12.2.2 `int df_AddRequest (struct df_s * df, struct df_request_list_node_s * node)`

Definition at line 66 of file `df.c`.

References `ListAdd()`, and `SIGNAL`.

Referenced by `MC_DeregisterService()`, `MC_RegisterService()`, and `MC_SearchForService()`.

### 5.12.2.3 `int df_Destroy (df_p df)`

Definition at line 83 of file `df.c`.

References `COND_DESTROY`, `df_node_Destroy()`, `df_request_list_Destroy()`, `ListPop()`, `ListTerminate()`, `MC_SUCCESS`, `MUTEX_DESTROY`, and `MUTEX_LOCK`.

Referenced by `mc_platform_Destroy()`.

### 5.12.2.4 `df_p df_Initialize (mc_platform_p mc_platform)`

Definition at line 103 of file `df.c`.

References `COND_INIT`, `COND_T`, `df_request_list_New()`, `ListInitialize()`, `MUTEX_INIT`, and `MUTEX_T`.

Referenced by `mc_platform_Initialize()`.

### 5.12.2.5 `int df_node_Destroy (df_node_p df_node)`

Definition at line 418 of file `df.c`.

References `MC_SUCCESS`, and `MUTEX_LOCK`.

Referenced by `df_Destroy()`.

**5.12.2.6 int df\_ProcessRequest (struct mc\_platform\_s \* *global*)**

Definition at line 134 of file df.c.

References mc\_platform\_s::df, df\_request\_list\_Pop(), MC\_ERR\_EMPTY, and MC\_ERR\_INVALID.

Referenced by df\_Thread().

**5.12.2.7 int df\_request\_list\_Destroy (df\_request\_list\_p *df\_request\_list*)**

Definition at line 323 of file df.c.

References COND\_DESTROY, df\_request\_list\_node\_Destroy(), ListPop(), ListTerminate(), MC\_SUCCESS, MUTEX\_DESTROY, and node.

Referenced by df\_Destroy().

**5.12.2.8 df\_request\_list\_p df\_request\_list\_New (void)**

Definition at line 349 of file df.c.

References CHECK\_NULL, COND\_INIT, COND\_T, ListInitialize(), MUTEX\_INIT, and MUTEX\_T.

Referenced by df\_Initialize().

**5.12.2.9 int df\_request\_list\_node\_Destroy (df\_request\_list\_node\_p *node*)**

Definition at line 294 of file df.c.

References COND\_DESTROY, MC\_SUCCESS, and MUTEX\_DESTROY.

Referenced by df\_request\_list\_Destroy(), and MC\_SearchForService().

**5.12.2.10 df\_request\_list\_node\_p df\_request\_list\_node\_New (void)**

Definition at line 305 of file df.c.

References CHECK\_NULL, COND\_INIT, COND\_T, MUTEX\_INIT, MUTEX\_T, and node.

Referenced by MC\_DeregisterService(), MC\_RegisterService(), and MC\_SearchForService().

**5.12.2.11 df\_request\_list\_node\_p df\_request\_list\_Pop (df\_request\_list\_p *requests*)**

Definition at line 374 of file df.c.

References ListPop(), MUTEX\_LOCK, MUTEX\_UNLOCK, and node.

Referenced by df\_ProcessRequest().

**5.12.2.12 int df\_request\_search\_Destroy (df\_request\_search\_p *node*)**

Definition at line 405 of file df.c.

References COND\_DESTROY, MC\_SUCCESS, and MUTEX\_DESTROY.

Referenced by MC\_SearchForService().

**5.12.2.13 df\_request\_search\_p df\_request\_search\_New (void)**

Definition at line 390 of file df.c.

References CHECK\_NULL, COND\_INIT, COND\_T, MUTEX\_INIT, MUTEX\_T, and search.

Referenced by MC\_SearchForService().

**5.12.2.14 int df\_SearchForService (df\_p df, const char \* searchstring, char \*\*\* agent\_names, char \*\*\* service\_names, int \*\* agent\_ids, int \* num\_entries)**

Definition at line 178 of file df.c.

References MC\_ERR\_NOT\_FOUND, MC\_SUCCESS, MUTEX\_LOCK, MUTEX\_UNLOCK, listNode\_s::next, and listNode\_s::node\_data.

Referenced by request\_handler\_SEARCH().

**5.12.2.15 void df\_Start (mc\_platform\_p mc\_platform)**

Definition at line 263 of file df.c.

References mc\_platform\_s::df, df\_Thread(), MC\_THREAD\_DF, mc\_platform\_s::stack\_size, and THREAD\_CREATE.

Referenced by mc\_platform\_Initialize().

**5.12.2.16 void\* df\_Thread (void \* arg)**

Definition at line 432 of file df.c.

References COND\_BROADCAST, COND\_WAIT, mc\_platform\_s::df, df\_ProcessRequest(), MC\_SUCCESS, MUTEX\_LOCK, MUTEX\_UNLOCK, mc\_platform\_s::quit, mc\_platform\_s::quit\_lock, and THREAD\_EXIT.

Referenced by df\_Start().

**5.12.2.17 int request\_handler\_DEREGISTER (struct mc\_platform\_s \* global, void \* data)**

Definition at line 526 of file df.c.

References mc\_platform\_s::df, MC\_SUCCESS, MUTEX\_LOCK, MUTEX\_UNLOCK, listNode\_s::next, node, and listNode\_s::node\_data.

**5.12.2.18 int request\_handler\_REGISTER (struct mc\_platform\_s \* global, void \* data)**

Definition at line 496 of file df.c.

References mc\_platform\_s::df, and df\_Add().

**5.12.2.19 int request\_handler\_SEARCH (struct mc\_platform\_s \* global, void \* data)**

Definition at line 502 of file df.c.

References mc\_platform\_s::df, df\_SearchForService(), MC\_SUCCESS, search, and SIGNAL.

**5.12.2.20** `int request_handler_SUBSCRIBE (struct mc_platform_s * global, void * data)`

Definition at line 521 of file df.c.



## 5.13 /home/dko/Projects/mobilec/src/dynstring.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "include/dynstring.h"
```

### Functions

- [dynstring\\_t \\* dynstring\\_New](#) (void)
- [int dynstring\\_Append](#) (dynstring\_t \*msg, char \*str)
- [int dynstring\\_Destroy](#) (dynstring\_t \*dynstring)

#### 5.13.1 Function Documentation

##### 5.13.1.1 int dynstring\_Append (dynstring\_t \* msg, char \* str)

Definition at line 30 of file dynstring.c.

References COMPOSE\_BLOCKSIZE, dynstring\_s::len, dynstring\_s::message, and dynstring\_s::size.

Referenced by fipa\_acl\_Compose(), fipa\_agent\_identifier\_Compose(), fipa\_agent\_identifier\_set\_Compose(), fipa\_DateTime\_Compose(), fipa\_envelope\_Compose\_\_from(), fipa\_number\_Compose(), fipa\_performative\_Compose(), fipa\_protocol\_Compose(), fipa\_string\_Compose(), fipa\_url\_Compose(), fipa\_url\_sequence\_Compose(), fipa\_word\_Compose(), message\_send\_Thread(), mtp\_http\_CreateMessage(), and mtp\_http\_InitializeFromConnection().

##### 5.13.1.2 int dynstring\_Destroy (dynstring\_t \* dynstring)

Definition at line 56 of file dynstring.c.

References dynstring\_s::message.

Referenced by fipa\_envelope\_Compose\_\_from(), MC\_AclSend(), message\_send\_Thread(), mtp\_http\_CreateMessage(), and mtp\_http\_InitializeFromConnection().

##### 5.13.1.3 dynstring\_t\* dynstring\_New (void)

Definition at line 14 of file dynstring.c.

References COMPOSE\_BLOCKSIZE, dynstring\_s::len, dynstring\_s::message, and dynstring\_s::size.

Referenced by fipa\_acl\_Compose(), fipa\_envelope\_Compose\_\_from(), message\_send\_Thread(), mtp\_http\_CreateMessage(), and mtp\_http\_InitializeFromConnection().

## 5.14 /home/dko/Projects/mobilec/src/fipa\_acl.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <strings.h>
#include "include/fipa_acl.h"
#include "dynstring.h"
#include "include/mc_error.h"
#include "include/macros.h"
```

### Defines

- #define [FREEMEM](#)(x) if (x != NULL) free(x)

### Functions

- [fipa\\_acl\\_message\\_t \\* fipa\\_acl\\_message\\_New](#) (void)
- [int fipa\\_acl\\_message\\_Destroy](#) ([fipa\\_acl\\_message\\_t](#) \*message)
- [fipa\\_acl\\_message\\_t \\* fipa\\_acl\\_message\\_Copy](#) ([fipa\\_acl\\_message\\_t](#) \*src)
- [fipa\\_message\\_string\\_t \\* fipa\\_message\\_string\\_New](#) (void)
- [int fipa\\_message\\_string\\_Destroy](#) ([fipa\\_message\\_string\\_t](#) \*message)
- [fipa\\_message\\_string\\_t \\* fipa\\_message\\_string\\_Copy](#) ([fipa\\_message\\_string\\_t](#) \*src)
- [fipa\\_url\\_sequence\\_t \\* fipa\\_url\\_sequence\\_New](#) (void)
- [int fipa\\_url\\_sequence\\_Destroy](#) ([fipa\\_url\\_sequence\\_t](#) \*sequence)
- [fipa\\_url\\_sequence\\_t \\* fipa\\_url\\_sequence\\_Copy](#) ([fipa\\_url\\_sequence\\_t](#) \*src)
- [fipa\\_agent\\_identifier\\_set\\_t \\* fipa\\_agent\\_identifier\\_set\\_New](#) (void)
- [int fipa\\_agent\\_identifier\\_set\\_Destroy](#) ([fipa\\_agent\\_identifier\\_set\\_t](#) \*idset)
- [fipa\\_agent\\_identifier\\_set\\_t \\* fipa\\_agent\\_identifier\\_set\\_Copy](#) ([fipa\\_agent\\_identifier\\_set\\_t](#) \*src)
- [fipa\\_agent\\_identifier\\_t \\* fipa\\_agent\\_identifier\\_New](#) (void)
- [int fipa\\_agent\\_identifier\\_Destroy](#) ([fipa\\_agent\\_identifier\\_t](#) \*id)
- [fipa\\_agent\\_identifier\\_t \\* fipa\\_agent\\_identifier\\_Copy](#) ([fipa\\_agent\\_identifier\\_t](#) \*src)
- [fipa\\_expression\\_t \\* fipa\\_expression\\_New](#) (void)
- [int fipa\\_expression\\_Destroy](#) ([fipa\\_expression\\_t](#) \*expr)
- [fipa\\_expression\\_t \\* fipa\\_expression\\_Copy](#) ([fipa\\_expression\\_t](#) \*src)
- [fipa\\_word\\_t \\* fipa\\_word\\_New](#) (void)
- [int fipa\\_word\\_Destroy](#) ([fipa\\_word\\_t](#) \*word)
- [fipa\\_word\\_t \\* fipa\\_word\\_Copy](#) ([fipa\\_word\\_t](#) \*src)
- [fipa\\_string\\_t \\* fipa\\_string\\_New](#) (void)
- [int fipa\\_string\\_Destroy](#) ([fipa\\_string\\_t](#) \*str)
- [fipa\\_string\\_t \\* fipa\\_string\\_Copy](#) ([fipa\\_string\\_t](#) \*src)
- [fipa\\_DateTime\\_t \\* fipa\\_DateTime\\_New](#) (void)
- [int fipa\\_DateTime\\_Destroy](#) ([fipa\\_DateTime\\_t](#) \*dt)
- [fipa\\_DateTime\\_t \\* fipa\\_DateTime\\_Copy](#) ([fipa\\_DateTime\\_t](#) \*src)
- [fipa\\_url\\_t \\* fipa\\_url\\_New](#) (void)

- `int fipa_url_Destroy (fipa_url_t *url)`
- `fipa_url_t * fipa_url_Copy (fipa_url_t *src)`
- `fipa_number_t * fipa_number_New (void)`
- `int fipa_number_Destroy (fipa_number_t *number)`
- `fipa_number_t * fipa_number_Copy (fipa_number_t *src)`
- `int fipa_acl_Parse (fipa_acl_message_p acl, fipa_message_string_p message)`
- `int fipa_message_parameter_Parse (fipa_acl_message_p acl, fipa_message_string_p message)`
- `int fipa_protocol_type_Parse (enum fipa_protocol_e *protocol, fipa_message_string_p message)`
- `int fipa_message_type_Parse (enum fipa_performative_e *performative, fipa_message_string_p message)`
- `int fipa_GetAtom (fipa_message_string_p message, char expected_atom)`
- `int fipa_word_Parse (fipa_word_t **word, fipa_message_string_p message)`
- `int fipa_CheckNextToken (const fipa_message_string_p message, const char *token)`
- `int fipa_expression_Parse (fipa_expression_t **expression, fipa_message_string_p message)`
- `int fipa_GetNextWord (char **word, const fipa_message_string_p message)`
- `int fipa_GetWholeToken (char **word, fipa_message_string_p message)`
- `int fipa_datetime_Parse (fipa_DateTime_p *datetime, fipa_message_string_p message)`
- `int fipa_string_Parse (fipa_string_p *fipa_string, fipa_message_string_p message)`
- `int fipa_agent_identifier_Parse (fipa_agent_identifier_p *aid, fipa_message_string_p message)`
- `int fipa_url_sequence_Parse (fipa_url_sequence_p *urls, fipa_message_string_p message)`
- `int fipa_url_Parse (fipa_url_p *url, fipa_message_string_p message)`
- `int fipa_agent_identifier_set_Parse (fipa_agent_identifier_set_p *agent_ids, fipa_message_string_p message)`
- `int fipa_acl_Compose (dynstring_t **msg, fipa_acl_message_t *acl)`
- `int fipa_protocol_Compose (dynstring_t *msg, enum fipa_protocol_e protocol)`
- `int fipa_performative_Compose (dynstring_t *msg, enum fipa_performative_e performative)`
- `int fipa_url_sequence_Compose (dynstring_t *msg, fipa_url_sequence_t *urls)`
- `int fipa_agent_identifier_set_Compose (dynstring_t *msg, fipa_agent_identifier_set_t *ids)`
- `int fipa_agent_identifier_Compose (dynstring_t *msg, fipa_agent_identifier_t *id)`
- `int fipa_expression_Compose (dynstring_t *msg, fipa_expression_t *expr)`
- `int fipa_word_Compose (dynstring_t *msg, fipa_word_t *word)`
- `int fipa_string_Compose (dynstring_t *msg, fipa_string_t *string)`
- `int fipa_DateTime_Compose (dynstring_t *msg, fipa_DateTime_t *date)`
- `int fipa_url_Compose (dynstring_t *msg, fipa_url_t *url)`
- `int fipa_number_Compose (dynstring_t *msg, fipa_number_t *number)`
- `struct fipa_acl_message_s * fipa_Reply (struct fipa_acl_message_s *acl)`

## 5.14.1 Define Documentation

### 5.14.1.1 #define FREEMEM(x) if (x != NULL) free(x)

Definition at line 53 of file fipa\_acl.c.

Referenced by fipa\_expression\_Destroy().

## 5.14.2 Function Documentation

### 5.14.2.1 `int fipa_acl_Compose (dynstring_t ** msg, fipa_acl_message_t * acl)`

Definition at line 1233 of file `fipa_acl.c`.

References `fipa_acl_message_s::content`, `fipa_acl_message_s::conversation_id`, `dynstring_Append()`, `dynstring_New()`, `fipa_acl_message_s::encoding`, `fipa_agent_identifier_Compose()`, `fipa_agent_identifier_set_Compose()`, `fipa_DateTime_Compose()`, `fipa_expression_Compose()`, `fipa_performative_Compose()`, `fipa_protocol_Compose()`, `FIPA_PROTOCOL_NONE`, `fipa_string_Compose()`, `fipa_acl_message_s::in_reply_to`, `fipa_acl_message_s::language`, `fipa_acl_message_s::ontology`, `fipa_acl_message_s::performative`, `fipa_acl_message_s::protocol`, `fipa_acl_message_s::receiver`, `fipa_acl_message_s::reply_by`, `fipa_acl_message_s::reply_to`, `fipa_acl_message_s::reply_with`, and `fipa_acl_message_s::sender`.

Referenced by `MC_AclSend()`.

### 5.14.2.2 `fipa_acl_message_t* fipa_acl_message_Copy (fipa_acl_message_t * src)`

Definition at line 86 of file `fipa_acl.c`.

References `fipa_acl_message_s::content`, `fipa_acl_message_s::conversation_id`, `fipa_acl_message_s::encoding`, `fipa_acl_message_New()`, `fipa_agent_identifier_Copy()`, `fipa_agent_identifier_set_Copy()`, `fipa_DateTime_Copy()`, `fipa_expression_Copy()`, `fipa_string_Copy()`, `fipa_acl_message_s::in_reply_to`, `fipa_acl_message_s::language`, `fipa_acl_message_s::ontology`, `fipa_acl_message_s::performative`, `fipa_acl_message_s::protocol`, `fipa_acl_message_s::receiver`, `fipa_acl_message_s::reply_by`, `fipa_acl_message_s::reply_to`, `fipa_acl_message_s::reply_with`, and `fipa_acl_message_s::sender`.

Referenced by `MC_AclSend()`.

### 5.14.2.3 `int fipa_acl_message_Destroy (fipa_acl_message_t * message)`

Definition at line 65 of file `fipa_acl.c`.

References `fipa_acl_message_s::content`, `fipa_acl_message_s::conversation_id`, `fipa_acl_message_s::encoding`, `fipa_agent_identifier_Destroy()`, `fipa_agent_identifier_set_Destroy()`, `fipa_DateTime_Destroy()`, `fipa_expression_Destroy()`, `fipa_string_Destroy()`, `fipa_acl_message_s::in_reply_to`, `fipa_acl_message_s::language`, `fipa_acl_message_s::ontology`, `fipa_acl_message_s::receiver`, `fipa_acl_message_s::reply_by`, `fipa_acl_message_s::reply_to`, `fipa_acl_message_s::reply_with`, and `fipa_acl_message_s::sender`.

Referenced by `acc_connection_Thread()`, and `MC_AclDestroy()`.

### 5.14.2.4 `fipa_acl_message_t* fipa_acl_message_New (void)`

Definition at line 57 of file `fipa_acl.c`.

Referenced by `acc_connection_Thread()`, `fipa_acl_message_Copy()`, `fipa_Reply()`, and `MC_AclNew()`.

### 5.14.2.5 `int fipa_acl_Parse (fipa_acl_message_p acl, fipa_message_string_p message)`

Definition at line 465 of file `fipa_acl.c`.

References `fipa_GetAtom()`, `fipa_message_parameter_Parse()`, `fipa_message_type_Parse()`, `MC_ERR_PARSE`, and `fipa_acl_message_s::performative`.

Referenced by `acc_connection_Thread()`.

#### 5.14.2.6 `int fipa_agent_identifier_Compose (dynstring_t * msg, fipa_agent_identifier_t * id)`

Definition at line 1447 of file `fipa_acl.c`.

References `fipa_agent_identifier_s::addresses`, `dynstring_Append()`, `fipa_agent_identifier_set_Compose()`, `fipa_url_sequence_Compose()`, `fipa_agent_identifier_s::name`, `fipa_agent_identifier_set_s::num`, `fipa_url_sequence_s::num`, and `fipa_agent_identifier_s::resolvers`.

Referenced by `fipa_acl_Compose()`, and `fipa_agent_identifier_set_Compose()`.

#### 5.14.2.7 `fipa_agent_identifier_t* fipa_agent_identifier_Copy (fipa_agent_identifier_t * src)`

Definition at line 236 of file `fipa_acl.c`.

References `fipa_agent_identifier_s::addresses`, `fipa_agent_identifier_New()`, `fipa_agent_identifier_set_Copy()`, `fipa_url_sequence_Copy()`, `fipa_agent_identifier_s::name`, `fipa_agent_identifier_s::resolvers`, and `strdup`.

Referenced by `fipa_acl_message_Copy()`, `fipa_acl_Param_Copy()`, `fipa_agent_identifier_set_Copy()`, and `fipa_Reply()`.

#### 5.14.2.8 `int fipa_agent_identifier_Destroy (fipa_agent_identifier_t * id)`

Definition at line 224 of file `fipa_acl.c`.

References `fipa_agent_identifier_s::addresses`, `fipa_agent_identifier_set_Destroy()`, `fipa_url_sequence_Destroy()`, `fipa_agent_identifier_s::name`, and `fipa_agent_identifier_s::resolvers`.

Referenced by `fipa_acl_message_Destroy()`, `fipa_acl_Param_Destroy()`, `fipa_agent_identifier_set_Destroy()`, and `MC_AclSetSender()`.

#### 5.14.2.9 `fipa_agent_identifier_t* fipa_agent_identifier_New (void)`

Definition at line 216 of file `fipa_acl.c`.

Referenced by `fipa_agent_identifier_Copy()`, `fipa_envelope_ParseAgentIdentifier()`, `MC_AclAddReceiver()`, `MC_AclAddReplyTo()`, and `MC_AclSetSender()`.

#### 5.14.2.10 `int fipa_agent_identifier_Parse (fipa_agent_identifier_p * aid, fipa_message_string_p message)`

Definition at line 1033 of file `fipa_acl.c`.

References `CHECK_NULL`, `fipa_word_s::content`, `fipa_agent_identifier_set_Parse()`, `fipa_GetAtom()`, `fipa_url_sequence_Parse()`, `fipa_word_Destroy()`, `fipa_word_Parse()`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

Referenced by `fipa_agent_identifier_set_Parse()`, and `fipa_message_parameter_Parse()`.

#### 5.14.2.11 **int fipa\_agent\_identifier\_set\_Compose** (dynstring\_t \* *msg*, fipa\_agent\_identifier\_set\_t \* *ids*)

Definition at line 1434 of file fipa\_acl.c.

References dynstring\_Append(), fipa\_agent\_identifier\_Compose(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, and fipa\_agent\_identifier\_set\_s::num.

Referenced by fipa\_acl\_Compose(), and fipa\_agent\_identifier\_Compose().

#### 5.14.2.12 **fipa\_agent\_identifier\_set\_t\* fipa\_agent\_identifier\_set\_Copy** (fipa\_agent\_identifier\_set\_t \* *src*)

Definition at line 195 of file fipa\_acl.c.

References fipa\_agent\_identifier\_Copy(), fipa\_agent\_identifier\_set\_New(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_agent\_identifier\_set\_s::num, and fipa\_agent\_identifier\_set\_s::retain\_order.

Referenced by fipa\_acl\_message\_Copy(), fipa\_acl\_Param\_Copy(), fipa\_agent\_identifier\_Copy(), and fipa\_Reply().

#### 5.14.2.13 **int fipa\_agent\_identifier\_set\_Destroy** (fipa\_agent\_identifier\_set\_t \* *idset*)

Definition at line 183 of file fipa\_acl.c.

References fipa\_agent\_identifier\_Destroy(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, and fipa\_agent\_identifier\_set\_s::num.

Referenced by fipa\_acl\_message\_Destroy(), fipa\_acl\_Param\_Destroy(), and fipa\_agent\_identifier\_Destroy().

#### 5.14.2.14 **fipa\_agent\_identifier\_set\_t\* fipa\_agent\_identifier\_set\_New** (void)

Definition at line 175 of file fipa\_acl.c.

Referenced by fipa\_agent\_identifier\_set\_Copy(), fipa\_envelope\_HandleIntendedReceiver(), fipa\_envelope\_HandleTo(), fipa\_Reply(), MC\_AclAddReceiver(), and MC\_AclAddReplyTo().

#### 5.14.2.15 **int fipa\_agent\_identifier\_set\_Parse** (fipa\_agent\_identifier\_set\_p \* *agent\_ids*, fipa\_message\_string\_p *message*)

Definition at line 1192 of file fipa\_acl.c.

References fipa\_word\_s::content, fipa\_agent\_identifier\_Parse(), fipa\_GetAtom(), fipa\_word\_Parse(), MC\_ERR\_PARSE, MC\_SUCCESS, and fipa\_agent\_identifier\_set\_s::retain\_order.

Referenced by fipa\_agent\_identifier\_Parse(), and fipa\_message\_parameter\_Parse().

#### 5.14.2.16 **int fipa\_CheckNextToken** (const fipa\_message\_string\_p *message*, const char \* *token*)

Definition at line 695 of file fipa\_acl.c.

References fipa\_message\_string\_s::parse.

Referenced by fipa\_expression\_Parse().

**5.14.2.17 int fipa\_DateTime\_Compose (dynstring\_t \* msg, fipa\_DateTime\_t \* date)**

Definition at line 1520 of file fipa\_acl.c.

References buf, fipa\_DateTime\_s::day, dynstring\_Append(), fipa\_DateTime\_s::hour, fipa\_DateTime\_s::millisecond, fipa\_DateTime\_s::minute, fipa\_DateTime\_s::month, fipa\_DateTime\_s::second, fipa\_DateTime\_s::sign, and fipa\_DateTime\_s::year.

Referenced by fipa\_acl\_Compose(), and fipa\_expression\_Compose().

**5.14.2.18 fipa\_DateTime\_t\* fipa\_DateTime\_Copy (fipa\_DateTime\_t \* src)**

Definition at line 399 of file fipa\_acl.c.

References fipa\_DateTime\_New().

Referenced by fipa\_acl\_envelope\_Received\_Copy(), fipa\_acl\_message\_Copy(), fipa\_acl\_Param\_Copy(), and fipa\_expression\_Copy().

**5.14.2.19 int fipa\_DateTime\_Destroy (fipa\_DateTime\_t \* dt)**

Definition at line 392 of file fipa\_acl.c.

Referenced by fipa\_acl\_envelope\_Received\_Destroy(), fipa\_acl\_message\_Destroy(), fipa\_acl\_Param\_Destroy(), and fipa\_expression\_Destroy().

**5.14.2.20 fipa\_DateTime\_t\* fipa\_DateTime\_New (void)**

Definition at line 384 of file fipa\_acl.c.

Referenced by fipa\_DateTime\_Copy().

**5.14.2.21 int fipa\_datetime\_Parse (fipa\_DateTime\_p \* datetime, fipa\_message\_string\_p message)**

Definition at line 850 of file fipa\_acl.c.

References buf, fipa\_GetWholeToken(), MC\_ERR\_PARSE, MC\_SUCCESS, fipa\_message\_string\_s::parse, and fipa\_DateTime\_s::sign.

Referenced by fipa\_envelope\_HandleDate(), fipa\_envelope\_HandleReceived(), fipa\_expression\_Parse(), and fipa\_message\_parameter\_Parse().

**5.14.2.22 int fipa\_expression\_Compose (dynstring\_t \* msg, fipa\_expression\_t \* expr)**

Definition at line 1473 of file fipa\_acl.c.

References fipa\_expression\_s::content, fipa\_expression\_s::content\_u::datetime, fipa\_expression\_s::content\_u::expression, fipa\_DateTime\_Compose(), FIPA\_EXPR\_DATETIME, FIPA\_EXPR\_EXPRESSION, FIPA\_EXPR\_NUMBER, FIPA\_EXPR\_STRING, FIPA\_EXPR\_WORD, fipa\_expression\_Compose(), fipa\_number\_Compose(), fipa\_string\_Compose(), fipa\_word\_Compose(), MC\_ERR\_PARSE, fipa\_expression\_s::content\_u::number, fipa\_expression\_s::content\_u::string, fipa\_expression\_s::type, and fipa\_expression\_s::content\_u::word.

Referenced by fipa\_acl\_Compose(), and fipa\_expression\_Compose().

**5.14.2.23 fipa\_expression\_t\* fipa\_expression\_Copy (fipa\_expression\_t \* src)**

Definition at line 287 of file fipa\_acl.c.

References fipa\_expression\_s::content, fipa\_expression\_s::content\_u::datetime, fipa\_expression\_s::content\_u::expression, fipa\_DateTime\_Copy(), FIPA\_EXPR\_DATETIME, FIPA\_EXPR\_EXPRESSION, FIPA\_EXPR\_NUMBER, FIPA\_EXPR\_STRING, FIPA\_EXPR\_WORD, fipa\_expression\_Copy(), fipa\_expression\_Destroy(), fipa\_expression\_New(), fipa\_number\_Copy(), fipa\_string\_Copy(), fipa\_word\_Copy(), fipa\_expression\_s::content\_u::number, fipa\_expression\_s::content\_u::string, fipa\_expression\_s::type, and fipa\_expression\_s::content\_u::word.

Referenced by fipa\_acl\_message\_Copy(), fipa\_expression\_Copy(), and fipa\_Reply().

**5.14.2.24 int fipa\_expression\_Destroy (fipa\_expression\_t \* expr)**

Definition at line 256 of file fipa\_acl.c.

References fipa\_expression\_s::content, fipa\_expression\_s::content\_u::datetime, fipa\_expression\_s::content\_u::expression, fipa\_DateTime\_Destroy(), FIPA\_EXPR\_DATETIME, FIPA\_EXPR\_EXPRESSION, FIPA\_EXPR\_NUMBER, FIPA\_EXPR\_STRING, FIPA\_EXPR\_WORD, fipa\_expression\_Destroy(), fipa\_number\_Destroy(), fipa\_string\_Destroy(), fipa\_word\_Destroy(), FREEMEM, fipa\_expression\_s::content\_u::number, fipa\_expression\_s::content\_u::string, fipa\_expression\_s::type, and fipa\_expression\_s::content\_u::word.

Referenced by fipa\_acl\_message\_Destroy(), fipa\_expression\_Copy(), and fipa\_expression\_Destroy().

**5.14.2.25 fipa\_expression\_t\* fipa\_expression\_New (void)**

Definition at line 248 of file fipa\_acl.c.

Referenced by fipa\_expression\_Copy(), and MC\_AclSetConversationID().

**5.14.2.26 int fipa\_expression\_Parse (fipa\_expression\_t \*\* expression, fipa\_message\_string\_p message)**

Definition at line 714 of file fipa\_acl.c.

References fipa\_CheckNextToken(), fipa\_datetime\_Parse(), FIPA\_EXPR\_DATETIME, FIPA\_EXPR\_EXPRESSION, FIPA\_EXPR\_STRING, FIPA\_EXPR\_WORD, fipa\_expression\_Parse(), fipa\_GetAtom(), fipa\_string\_Parse(), fipa\_word\_Parse(), MC\_ERR\_PARSE, MC\_SUCCESS, and fipa\_expression\_s::type.

Referenced by fipa\_expression\_Parse(), and fipa\_message\_parameter\_Parse().

**5.14.2.27 int fipa\_GetAtom (fipa\_message\_string\_p message, char expected\_atom)**

Definition at line 630 of file fipa\_acl.c.

References MC\_ERR\_PARSE, MC\_SUCCESS, and fipa\_message\_string\_s::parse.

Referenced by fipa\_acl\_Parse(), fipa\_agent\_identifier\_Parse(), fipa\_agent\_identifier\_set\_Parse(), fipa\_expression\_Parse(), fipa\_message\_parameter\_Parse(), fipa\_string\_Parse(), and fipa\_url\_sequence\_Parse().

**5.14.2.28 int fipa\_GetNextWord (char \*\* word, const fipa\_message\_string\_p message)**

Definition at line 765 of file fipa\_acl.c.



References ERR, MC\_SUCCESS, and fipa\_message\_string\_s::parse.

#### 5.14.2.29 int fipa\_GetWholeToken (char \*\* word, fipa\_message\_string\_p message)

Definition at line 813 of file fipa\_acl.c.

References MC\_SUCCESS, and fipa\_message\_string\_s::parse.

Referenced by fipa\_datetime\_Parse().

#### 5.14.2.30 int fipa\_message\_parameter\_Parse (fipa\_acl\_message\_p acl, fipa\_message\_string\_p message)

Definition at line 485 of file fipa\_acl.c.

References fipa\_acl\_message\_s::content, fipa\_word\_s::content, fipa\_acl\_message\_s::conversation\_id, fipa\_acl\_message\_s::encoding, fipa\_agent\_identifier\_Parse(), fipa\_agent\_identifier\_set\_Parse(), fipa\_datetime\_Parse(), fipa\_expression\_Parse(), fipa\_GetAtom(), fipa\_protocol\_type\_Parse(), fipa\_string\_Parse(), fipa\_word\_Destroy(), fipa\_word\_Parse(), fipa\_acl\_message\_s::in\_reply\_to, fipa\_acl\_message\_s::language, MC\_ERR\_PARSE, fipa\_acl\_message\_s::ontology, fipa\_acl\_message\_s::protocol, fipa\_acl\_message\_s::receiver, fipa\_acl\_message\_s::reply\_by, fipa\_acl\_message\_s::reply\_to, fipa\_acl\_message\_s::reply\_with, and fipa\_acl\_message\_s::sender.

Referenced by fipa\_acl\_Parse().

#### 5.14.2.31 fipa\_message\_string\_t\* fipa\_message\_string\_Copy (fipa\_message\_string\_t \* src)

Definition at line 128 of file fipa\_acl.c.

References fipa\_message\_string\_s::message, fipa\_message\_string\_s::parse, and strdup.

#### 5.14.2.32 int fipa\_message\_string\_Destroy (fipa\_message\_string\_t \* message)

Definition at line 118 of file fipa\_acl.c.

References fipa\_message\_string\_s::message.

Referenced by acc\_connection\_Thread().

#### 5.14.2.33 fipa\_message\_string\_t\* fipa\_message\_string\_New (void)

Definition at line 110 of file fipa\_acl.c.

Referenced by acc\_connection\_Thread().

#### 5.14.2.34 int fipa\_message\_type\_Parse (enum fipa\_performative\_e \* performative, fipa\_message\_string\_p message)

Definition at line 568 of file fipa\_acl.c.

References fipa\_word\_s::content, FIPA\_ACCEPT\_PROPOSAL, FIPA\_AGREE, FIPA\_CALL\_FOR\_PROPOSAL, FIPA\_CANCEL, FIPA\_CONFIRM, FIPA\_DISCONFIRM, FIPA\_FAILURE,

FIPA\_INFORM, FIPA\_INFORM\_IF, FIPA\_INFORM\_REF, FIPA\_NOT\_UNDERSTOOD, FIPA\_PROPOGATE, FIPA\_PROPOSE, FIPA\_PROXY, FIPA\_QUERY\_IF, FIPA\_QUERY\_REF, FIPA\_REFUSE, FIPA\_REJECT\_PROPOSAL, FIPA\_REQUEST, FIPA\_REQUEST\_WHEN, FIPA\_REQUEST\_WHENEVER, FIPA\_SUBSCRIBE, fipa\_word\_Destroy(), fipa\_word\_Parse(), and MC\_ERR\_PARSE.

Referenced by fipa\_acl\_Parse().

#### **5.14.2.35 int fipa\_number\_Compose (dynstring\_t \* msg, fipa\_number\_t \* number)**

Definition at line 1548 of file fipa\_acl.c.

References dynstring\_Append(), and fipa\_number\_s::str.

Referenced by fipa\_expression\_Compose().

#### **5.14.2.36 fipa\_number\_t\* fipa\_number\_Copy (fipa\_number\_t \* src)**

Definition at line 455 of file fipa\_acl.c.

References fipa\_number\_New(), fipa\_number\_s::str, and strdup.

Referenced by fipa\_expression\_Copy().

#### **5.14.2.37 int fipa\_number\_Destroy (fipa\_number\_t \* number)**

Definition at line 445 of file fipa\_acl.c.

References fipa\_number\_s::str.

Referenced by fipa\_expression\_Destroy().

#### **5.14.2.38 fipa\_number\_t\* fipa\_number\_New (void)**

Definition at line 437 of file fipa\_acl.c.

Referenced by fipa\_number\_Copy().

#### **5.14.2.39 int fipa\_performative\_Compose (dynstring\_t \* msg, enum fipa\_performative\_e performative)**

Definition at line 1346 of file fipa\_acl.c.

References dynstring\_Append(), FIPA\_ACCEPT\_PROPOSAL, FIPA\_AGREE, FIPA\_CALL\_FOR\_PROPOSAL, FIPA\_CANCEL, FIPA\_CONFIRM, FIPA\_DISCONFIRM, FIPA\_FAILURE, FIPA\_INFORM, FIPA\_INFORM\_IF, FIPA\_INFORM\_REF, FIPA\_NOT\_UNDERSTOOD, FIPA\_PROPOGATE, FIPA\_PROPOSE, FIPA\_PROXY, FIPA\_QUERY\_IF, FIPA\_QUERY\_REF, FIPA\_REFUSE, FIPA\_REJECT\_PROPOSAL, FIPA\_REQUEST, FIPA\_REQUEST\_WHEN, FIPA\_REQUEST\_WHENEVER, FIPA\_SUBSCRIBE, and MC\_ERR\_PARSE.

Referenced by fipa\_acl\_Compose().

#### **5.14.2.40 int fipa\_protocol\_Compose (dynstring\_t \* msg, enum fipa\_protocol\_e protocol)**

Definition at line 1303 of file fipa\_acl.c.

References `dynstring_Append()`, `FIPA_PROTOCOL_BROKERING`, `FIPA_PROTOCOL_CONTRACT_NET`, `FIPA_PROTOCOL_DUTCH_AUCTION`, `FIPA_PROTOCOL_ENGLISH_AUCTION`, `FIPA_PROTOCOL_ITERATED_CONTRACT_NET`, `FIPA_PROTOCOL_PROPOSE`, `FIPA_PROTOCOL_QUERY`, `FIPA_PROTOCOL_RECRUITING`, `FIPA_PROTOCOL_REQUEST`, `FIPA_PROTOCOL_REQUEST_WHEN`, `FIPA_PROTOCOL_SUBSCRIBE`, and `MC_ERR_PARSE`.

Referenced by `fipa_acl_Compose()`.

#### 5.14.2.41 `int fipa_protocol_type_Parse (enum fipa_protocol_e * protocol, fipa_message_string_p message)`

Definition at line 528 of file `fipa_acl.c`.

References `fipa_word_s::content`, `FIPA_PROTOCOL_BROKERING`, `FIPA_PROTOCOL_CONTRACT_NET`, `FIPA_PROTOCOL_DUTCH_AUCTION`, `FIPA_PROTOCOL_ENGLISH_AUCTION`, `FIPA_PROTOCOL_ITERATED_CONTRACT_NET`, `FIPA_PROTOCOL_PROPOSE`, `FIPA_PROTOCOL_QUERY`, `FIPA_PROTOCOL_RECRUITING`, `FIPA_PROTOCOL_REQUEST`, `FIPA_PROTOCOL_REQUEST_WHEN`, `FIPA_PROTOCOL_SUBSCRIBE`, `fipa_word_Destroy()`, `fipa_word_Parse()`, and `MC_ERR_PARSE`.

Referenced by `fipa_message_parameter_Parse()`.

#### 5.14.2.42 `struct fipa_acl_message_s* fipa_Reply (struct fipa_acl_message_s * acl) [read]`

Definition at line 1556 of file `fipa_acl.c`.

References `fipa_acl_message_s::conversation_id`, `fipa_acl_message_New()`, `fipa_agent_identifier_Copy()`, `fipa_agent_identifier_set_Copy()`, `fipa_agent_identifier_set_New()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, `fipa_expression_Copy()`, `fipa_agent_identifier_set_s::num`, `fipa_acl_message_s::protocol`, `fipa_acl_message_s::receiver`, `fipa_acl_message_s::reply_to`, `fipa_agent_identifier_set_s::retain_order`, and `fipa_acl_message_s::sender`.

Referenced by `MC_AclReply()`.

#### 5.14.2.43 `int fipa_string_Compose (dynstring_t * msg, fipa_string_t * string)`

Definition at line 1511 of file `fipa_acl.c`.

References `fipa_string_s::content`, and `dynstring_Append()`.

Referenced by `fipa_acl_Compose()`, and `fipa_expression_Compose()`.

#### 5.14.2.44 `fipa_string_t* fipa_string_Copy (fipa_string_t * src)`

Definition at line 374 of file `fipa_acl.c`.

References `fipa_string_s::content`, `fipa_string_New()`, and `strdup`.

Referenced by `fipa_acl_message_Copy()`, and `fipa_expression_Copy()`.

#### 5.14.2.45 `int fipa_string_Destroy (fipa_string_t * str)`

Definition at line 364 of file `fipa_acl.c`.

References `fipa_string_s::content`.

Referenced by `fipa_acl_message_Destroy()`, `fipa_expression_Destroy()`, and `MC_AclSetContent()`.

#### **5.14.2.46 `fipa_string_t* fipa_string_New (void)`**

Definition at line 356 of file `fipa_acl.c`.

Referenced by `fipa_string_Copy()`, `MC_AclSetContent()`, and `MC_AclSetConversationID()`.

#### **5.14.2.47 `int fipa_string_Parse (fipa_string_p *fipa_string, fipa_message_string_p message)`**

Definition at line 988 of file `fipa_acl.c`.

References `fipa_string_s::content`, `fipa_GetAtom()`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

Referenced by `fipa_expression_Parse()`, and `fipa_message_parameter_Parse()`.

#### **5.14.2.48 `int fipa_url_Compose (dynstring_t *msg, fipa_url_t *url)`**

Definition at line 1540 of file `fipa_acl.c`.

References `dynstring_Append()`, and `fipa_url_s::str`.

Referenced by `fipa_url_sequence_Compose()`.

#### **5.14.2.49 `fipa_url_t* fipa_url_Copy (fipa_url_t *src)`**

Definition at line 427 of file `fipa_acl.c`.

References `fipa_url_New()`, `fipa_url_s::str`, and `strdup`.

Referenced by `fipa_acl_envelope_Received_Copy()`, and `fipa_url_sequence_Copy()`.

#### **5.14.2.50 `int fipa_url_Destroy (fipa_url_t *url)`**

Definition at line 417 of file `fipa_acl.c`.

References `fipa_url_s::str`.

Referenced by `fipa_acl_envelope_Received_Destroy()`, and `fipa_url_sequence_Destroy()`.

#### **5.14.2.51 `fipa_url_t* fipa_url_New (void)`**

Definition at line 409 of file `fipa_acl.c`.

Referenced by `fipa_envelope_HandleReceived()`, `fipa_envelope_ParseAddresses()`, `fipa_url_Copy()`, `MC_AclAddReceiver()`, `MC_AclAddReplyTo()`, and `MC_AclSetSender()`.

#### **5.14.2.52 `int fipa_url_Parse (fipa_url_p *url, fipa_message_string_p message)`**

Definition at line 1172 of file `fipa_acl.c`.

References `fipa_word_s::content`, `fipa_word_Destroy()`, `fipa_word_Parse()`, and `strdup`.

Referenced by `fipa_url_sequence_Parse()`.

**5.14.2.53 int fipa\_url\_sequence\_Compose (dynstring\_t \* msg, fipa\_url\_sequence\_t \* urls)**

Definition at line 1421 of file fipa\_acl.c.

References dynstring\_Append(), fipa\_url\_Compose(), fipa\_url\_sequence\_s::num, and fipa\_url\_sequence\_s::urls.

Referenced by fipa\_agent\_identifier\_Compose().

**5.14.2.54 fipa\_url\_sequence\_t\* fipa\_url\_sequence\_Copy (fipa\_url\_sequence\_t \* src)**

Definition at line 159 of file fipa\_acl.c.

References fipa\_url\_Copy(), fipa\_url\_sequence\_New(), fipa\_url\_sequence\_s::num, and fipa\_url\_sequence\_s::urls.

Referenced by fipa\_agent\_identifier\_Copy().

**5.14.2.55 int fipa\_url\_sequence\_Destroy (fipa\_url\_sequence\_t \* sequence)**

Definition at line 147 of file fipa\_acl.c.

References fipa\_url\_Destroy(), fipa\_url\_sequence\_s::num, and fipa\_url\_sequence\_s::urls.

Referenced by fipa\_agent\_identifier\_Destroy().

**5.14.2.56 fipa\_url\_sequence\_t\* fipa\_url\_sequence\_New (void)**

Definition at line 139 of file fipa\_acl.c.

Referenced by fipa\_url\_sequence\_Copy(), fipa\_url\_sequence\_Parse(), MC\_AclAddReceiver(), MC\_AclAddReplyTo(), and MC\_AclSetSender().

**5.14.2.57 int fipa\_url\_sequence\_Parse (fipa\_url\_sequence\_p \* urls, fipa\_message\_string\_p message)**

Definition at line 1140 of file fipa\_acl.c.

References fipa\_word\_s::content, fipa\_GetAtom(), fipa\_url\_Parse(), fipa\_url\_sequence\_New(), fipa\_word\_Destroy(), fipa\_word\_Parse(), and MC\_ERR\_PARSE.

Referenced by fipa\_agent\_identifier\_Parse().

**5.14.2.58 int fipa\_word\_Compose (dynstring\_t \* msg, fipa\_word\_t \* word)**

Definition at line 1503 of file fipa\_acl.c.

References fipa\_word\_s::content, and dynstring\_Append().

Referenced by fipa\_expression\_Compose().

**5.14.2.59 fipa\_word\_t\* fipa\_word\_Copy (fipa\_word\_t \* src)**

Definition at line 346 of file fipa\_acl.c.

References fipa\_word\_s::content, fipa\_word\_New(), and strdup.

Referenced by `fipa_expression_Copy()`.

#### **5.14.2.60 `int fipa_word_Destroy (fipa_word_t * word)`**

Definition at line 336 of file `fipa_acl.c`.

References `fipa_word_s::content`.

Referenced by `fipa_agent_identifier_Parse()`, `fipa_expression_Destroy()`, `fipa_message_parameter_Parse()`, `fipa_message_type_Parse()`, `fipa_protocol_type_Parse()`, `fipa_url_Parse()`, and `fipa_url_sequence_Parse()`.

#### **5.14.2.61 `fipa_word_t* fipa_word_New (void)`**

Definition at line 328 of file `fipa_acl.c`.

Referenced by `fipa_word_Copy()`.

#### **5.14.2.62 `int fipa_word_Parse (fipa_word_t ** word, fipa_message_string_p message)`**

Definition at line 653 of file `fipa_acl.c`.

References `CHECK_NULL`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

Referenced by `fipa_agent_identifier_Parse()`, `fipa_agent_identifier_set_Parse()`, `fipa_expression_Parse()`, `fipa_message_parameter_Parse()`, `fipa_message_type_Parse()`, `fipa_protocol_type_Parse()`, `fipa_url_Parse()`, and `fipa_url_sequence_Parse()`.

## 5.15 /home/dko/Projects/mobilec/src/fipa\_envelope.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <mxml.h>
#include <time.h>
#include "include/fipa_acl_envelope.h"
#include "include/mc_error.h"
#include "include/mc_platform.h"
```

### Functions

- [fipa\\_acl\\_envelope\\_Received\\_t \\* fipa\\_acl\\_envelope\\_Received\\_New](#) (void)
- [int fipa\\_acl\\_envelope\\_Received\\_Destroy](#) (fipa\_acl\_envelope\_Received\_t \*received)
- [fipa\\_acl\\_envelope\\_Received\\_t \\* fipa\\_acl\\_envelope\\_Received\\_Copy](#) (fipa\_acl\_envelope\_Received\_t \*received)
- [fipa\\_acl\\_Param\\_t \\* fipa\\_acl\\_Param\\_New](#) (void)
- [int fipa\\_acl\\_Param\\_Destroy](#) (fipa\_acl\_Param\_t \*param)
- [fipa\\_acl\\_Param\\_t \\* fipa\\_acl\\_Param\\_Copy](#) (fipa\_acl\_Param\_t \*param)
- [fipa\\_acl\\_envelope\\_t \\* fipa\\_acl\\_envelope\\_New](#) (void)
- [int fipa\\_acl\\_envelope\\_Destroy](#) (fipa\_acl\_envelope\_t \*envelope)
- [fipa\\_acl\\_envelope\\_t \\* fipa\\_acl\\_envelope\\_Copy](#) (fipa\_acl\_envelope\_t \*envelope)
- [int fipa\\_envelope\\_Parse](#) (struct fipa\_acl\_envelope\_s \*envelope, const char \*message)
- [int fipa\\_envelope\\_HandleEnvelope](#) (struct fipa\_acl\_envelope\_s \*envelope, mxml\_node\_t \*node)
- [int fipa\\_envelope\\_HandleParams](#) (struct fipa\_acl\_envelope\_s \*envelope, mxml\_node\_t \*node)
- [int fipa\\_envelope\\_HandleTo](#) (struct fipa\_acl\_envelope\_s \*envelope, mxml\_node\_t \*param\_node, int cur\_param)
- [int fipa\\_envelope\\_HandleFrom](#) (struct fipa\_acl\_envelope\_s \*envelope, mxml\_node\_t \*param\_node, int cur\_param)
- [int fipa\\_envelope\\_HandleComments](#) (struct fipa\_acl\_envelope\_s \*envelope, mxml\_node\_t \*param\_node, int cur\_param)
- [int fipa\\_envelope\\_HandleAclRepresentation](#) (struct fipa\_acl\_envelope\_s \*envelope, mxml\_node\_t \*param\_node, int cur\_param)
- [int fipa\\_envelope\\_HandlePayloadLength](#) (struct fipa\_acl\_envelope\_s \*envelope, mxml\_node\_t \*param\_node, int cur\_param)
- [int fipa\\_envelope\\_HandlePayloadEncoding](#) (struct fipa\_acl\_envelope\_s \*envelope, mxml\_node\_t \*param\_node, int cur\_param)
- [int fipa\\_envelope\\_HandleDate](#) (struct fipa\_acl\_envelope\_s \*envelope, mxml\_node\_t \*param\_node, int cur\_param)
- [int fipa\\_envelope\\_HandleIntendedReceiver](#) (struct fipa\_acl\_envelope\_s \*envelope, mxml\_node\_t \*param\_node, int cur\_param)
- [int fipa\\_envelope\\_HandleReceived](#) (struct fipa\_acl\_envelope\_s \*envelope, mxml\_node\_t \*param\_node, int cur\_param)
- [int fipa\\_envelope\\_ParseAgentIdentifier](#) (struct fipa\_agent\_identifier\_s \*\*aid, mxml\_node\_t \*agent\_identifier\_node)
- [int fipa\\_envelope\\_ParseAddresses](#) (struct fipa\_agent\_identifier\_s \*aid, mxml\_node\_t \*addresses\_node)

- `int fipa_envelope_ParseResolvers` (struct `fipa_agent_identifier_s` \*aid, `mxml_node_t` \*resolvers\_node)
- `char * fipa_envelope_Compose` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__envelope` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__params` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__to` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__from` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__acl_representation` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__payload_encoding` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__date` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__intended_receiver` (`fipa_acl_message_t` \*fipa\_acl)

## 5.15.1 Function Documentation

### 5.15.1.1 `fipa_acl_envelope_t* fipa_acl_envelope_Copy` (`fipa_acl_envelope_t` \* *envelope*)

Definition at line 150 of file `fipa_envelope.c`.

References `fipa_acl_envelope_New()`, `fipa_acl_Param_Copy()`, `fipa_acl_envelope_s::num_params`, and `fipa_acl_envelope_s::params`.

### 5.15.1.2 `int fipa_acl_envelope_Destroy` (`fipa_acl_envelope_t` \* *envelope*)

Definition at line 137 of file `fipa_envelope.c`.

References `fipa_acl_Param_Destroy()`, `fipa_acl_envelope_s::num_params`, and `fipa_acl_envelope_s::params`.

Referenced by `acc_connection_Thread()`.

### 5.15.1.3 `fipa_acl_envelope_t* fipa_acl_envelope_New` (void)

Definition at line 129 of file `fipa_envelope.c`.

Referenced by `acc_connection_Thread()`, and `fipa_acl_envelope_Copy()`.

### 5.15.1.4 `fipa_acl_envelope_Received_t* fipa_acl_envelope_Received_Copy` (`fipa_acl_envelope_Received_t` \* *received*)

Definition at line 69 of file `fipa_envelope.c`.

References `fipa_acl_envelope_Received_New()`, `fipa_DateTime_Copy()`, `fipa_url_Copy()`, `fipa_acl_envelope_Received_s::received_by`, `fipa_acl_envelope_Received_s::received_date`, `fipa_acl_envelope_Received_s::received_from`, `fipa_acl_envelope_Received_s::received_id`, `fipa_acl_envelope_Received_s::received_via`, and `strdup`.

Referenced by `fipa_acl_Param_Copy()`.

### 5.15.1.5 `int fipa_acl_envelope_Received_Destroy` (`fipa_acl_envelope_Received_t` \* *received*)

Definition at line 57 of file `fipa_envelope.c`.



References `fipa_DateTime_Destroy()`, `fipa_url_Destroy()`, `fipa_acl_envelope_Received_s::received_by`, `fipa_acl_envelope_Received_s::received_date`, `fipa_acl_envelope_Received_s::received_from`, `fipa_acl_envelope_Received_s::received_id`, and `fipa_acl_envelope_Received_s::received_via`.

Referenced by `fipa_acl_Param_Destroy()`.

#### 5.15.1.6 `fipa_acl_envelope_Received_t* fipa_acl_envelope_Received_New (void)`

Definition at line 48 of file `fipa_envelope.c`.

Referenced by `fipa_acl_envelope_Received_Copy()`, and `fipa_envelope_HandleReceived()`.

#### 5.15.1.7 `fipa_acl_Param_t* fipa_acl_Param_Copy (fipa_acl_Param_t * param)`

Definition at line 110 of file `fipa_envelope.c`.

References `fipa_acl_Param_s::acl_representation`, `fipa_acl_Param_s::comments`, `fipa_acl_Param_s::date`, `fipa_acl_envelope_Received_Copy()`, `fipa_acl_Param_New()`, `fipa_agent_identifier_Copy()`, `fipa_agent_identifier_set_Copy()`, `fipa_DateTime_Copy()`, `fipa_acl_Param_s::from`, `fipa_acl_Param_s::intended_receiver`, `fipa_acl_Param_s::payload_encoding`, `fipa_acl_Param_s::payload_length`, `fipa_acl_Param_s::received`, `strdup`, and `fipa_acl_Param_s::to`.

Referenced by `fipa_acl_envelope_Copy()`.

#### 5.15.1.8 `int fipa_acl_Param_Destroy (fipa_acl_Param_t * param)`

Definition at line 93 of file `fipa_envelope.c`.

References `fipa_acl_Param_s::acl_representation`, `fipa_acl_Param_s::comments`, `fipa_acl_Param_s::date`, `fipa_acl_envelope_Received_Destroy()`, `fipa_agent_identifier_Destroy()`, `fipa_agent_identifier_set_Destroy()`, `fipa_DateTime_Destroy()`, `fipa_acl_Param_s::from`, `fipa_acl_Param_s::intended_receiver`, `fipa_acl_Param_s::payload_encoding`, `fipa_acl_Param_s::payload_length`, `fipa_acl_Param_s::received`, and `fipa_acl_Param_s::to`.

Referenced by `fipa_acl_envelope_Destroy()`.

#### 5.15.1.9 `fipa_acl_Param_t* fipa_acl_Param_New (void)`

Definition at line 85 of file `fipa_envelope.c`.

Referenced by `fipa_acl_Param_Copy()`, and `fipa_envelope_HandleTo()`.

#### 5.15.1.10 `char* fipa_envelope_Compose (fipa_acl_message_t * fipa_acl)`

Definition at line 869 of file `fipa_envelope.c`.

References `fipa_envelope_Compose__envelope()`, `MXML_ADD_AFTER`, `MXML_ADD_TO_PARENT`, `MXML_NO_CALLBACK`, `mxmAdd()`, `mxmDelete()`, `mxmLoadString()`, `mxmSaveAllocString()`, and `node`.

Referenced by `MC_AclSend()`.

### 5.15.1.11 **mxml\_node\_t\* fipa\_envelope\_Compose\_\_acl\_representation (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 1089 of file fipa\_envelope.c.

References mxmlNewElement(), mxmlNewText(), and node.

Referenced by fipa\_envelope\_Compose\_\_params().

### 5.15.1.12 **mxml\_node\_t\* fipa\_envelope\_Compose\_\_date (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 1117 of file fipa\_envelope.c.

References buf, mxmlNewElement(), mxmlNewText(), and node.

Referenced by fipa\_envelope\_Compose\_\_params().

### 5.15.1.13 **mxml\_node\_t\* fipa\_envelope\_Compose\_\_envelope (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 894 of file fipa\_envelope.c.

References fipa\_envelope\_Compose\_\_params(), MXML\_ADD\_AFTER, MXML\_ADD\_TO\_PARENT, mxmlAdd(), mxmlNewElement(), and node.

Referenced by fipa\_envelope\_Compose().

### 5.15.1.14 **mxml\_node\_t\* fipa\_envelope\_Compose\_\_from (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 1021 of file fipa\_envelope.c.

References fipa\_agent\_identifier\_s::addresses, buf, dynstring\_Append(), dynstring\_Destroy(), dynstring\_New(), g\_mc\_platform, mc\_platform\_s::hostname, dynstring\_s::message, mxmlNewElement(), mxmlNewText(), fipa\_agent\_identifier\_s::name, fipa\_url\_sequence\_s::num, mc\_platform\_s::port, fipa\_acl\_message\_s::sender, fipa\_url\_s::str, and fipa\_url\_sequence\_s::urls.

Referenced by fipa\_envelope\_Compose\_\_params().

### 5.15.1.15 **mxml\_node\_t\* fipa\_envelope\_Compose\_\_intended\_receiver (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 1146 of file fipa\_envelope.c.

References fipa\_agent\_identifier\_s::addresses, fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, mxmlNewElement(), mxmlNewText(), fipa\_agent\_identifier\_s::name, node, fipa\_url\_sequence\_s::num, fipa\_acl\_message\_s::receiver, fipa\_url\_s::str, and fipa\_url\_sequence\_s::urls.

Referenced by fipa\_envelope\_Compose\_\_params().

### 5.15.1.16 **mxml\_node\_t\* fipa\_envelope\_Compose\_\_params (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 914 of file fipa\_envelope.c.

References fipa\_envelope\_Compose\_\_acl\_representation(), fipa\_envelope\_Compose\_\_date(), fipa\_envelope\_Compose\_\_from(), fipa\_envelope\_Compose\_\_intended\_receiver(), fipa\_envelope\_Compose\_\_payload\_encoding(), fipa\_envelope\_Compose\_\_to(), MXML\_ADD\_AFTER, MXML\_ADD\_TO\_PARENT, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), and node.

Referenced by fipa\_envelope\_Compose\_\_envelope().

#### 5.15.1.17 **mxml\_node\_t\* fipa\_envelope\_Compose\_\_payload\_encoding (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 1103 of file fipa\_envelope.c.

References mxmlNewElement(), mxmlNewText(), and node.

Referenced by fipa\_envelope\_Compose\_\_params().

#### 5.15.1.18 **mxml\_node\_t\* fipa\_envelope\_Compose\_\_to (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 975 of file fipa\_envelope.c.

References fipa\_agent\_identifier\_s::addresses, fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, mxmlNewElement(), mxmlNewText(), fipa\_agent\_identifier\_s::name, node, fipa\_url\_sequence\_s::num, fipa\_agent\_identifier\_set\_s::num, fipa\_acl\_message\_s::receiver, fipa\_url\_s::str, and fipa\_url\_sequence\_s::urls.

Referenced by fipa\_envelope\_Compose\_\_params().

#### 5.15.1.19 **int fipa\_envelope\_HandleAclRepresentation (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)**

Definition at line 381 of file fipa\_envelope.c.

References fipa\_acl\_Param\_s::acl\_representation, mxml\_node\_s::child, MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_TEXT, mxmlFindElement(), fipa\_acl\_envelope\_s::params, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by fipa\_envelope\_HandleParams().

#### 5.15.1.20 **int fipa\_envelope\_HandleComments (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)**

Definition at line 357 of file fipa\_envelope.c.

References mxml\_node\_s::child, fipa\_acl\_Param\_s::comments, MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_TEXT, mxmlFindElement(), fipa\_acl\_envelope\_s::params, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by fipa\_envelope\_HandleParams().

#### 5.15.1.21 **int fipa\_envelope\_HandleDate (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)**

Definition at line 452 of file fipa\_envelope.c.

References mxml\_node\_s::child, fipa\_acl\_Param\_s::date, fipa\_datetime\_Parse(), MC\_ERR\_PARSE, fipa\_message\_string\_s::message, MXML\_DESCEND\_FIRST, MXML\_TEXT, mxmlFindElement(), fipa\_acl\_envelope\_s::params, fipa\_message\_string\_s::parse, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by fipa\_envelope\_HandleParams().

### 5.15.1.22 **int fipa\_envelope\_HandleEnvelope** (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *node*)

Definition at line 178 of file fipa\_envelope.c.

References fipa\_envelope\_HandleParams(), MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, and mxmlFindElement().

Referenced by fipa\_envelope\_Parse().

### 5.15.1.23 **int fipa\_envelope\_HandleFrom** (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)

Definition at line 328 of file fipa\_envelope.c.

References fipa\_envelope\_ParseAgentIdentifier(), fipa\_acl\_Param\_s::from, MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, mxmlFindElement(), and fipa\_acl\_envelope\_s::params.

Referenced by fipa\_envelope\_HandleParams().

### 5.15.1.24 **int fipa\_envelope\_HandleIntendedReceiver** (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)

Definition at line 482 of file fipa\_envelope.c.

References fipa\_agent\_identifier\_set\_New(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_envelope\_ParseAgentIdentifier(), fipa\_acl\_Param\_s::intended\_receiver, MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_NO\_DESCEND, mxmlFindElement(), fipa\_agent\_identifier\_set\_s::num, fipa\_acl\_envelope\_s::params, and fipa\_agent\_identifier\_set\_s::retain\_order.

Referenced by fipa\_envelope\_HandleParams().

### 5.15.1.25 **int fipa\_envelope\_HandleParams** (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *node*)

Definition at line 198 of file fipa\_envelope.c.

References buf, fipa\_envelope\_HandleAclRepresentation(), fipa\_envelope\_HandleComments(), fipa\_envelope\_HandleDate(), fipa\_envelope\_HandleFrom(), fipa\_envelope\_HandleIntendedReceiver(), fipa\_envelope\_HandlePayloadEncoding(), fipa\_envelope\_HandlePayloadLength(), fipa\_envelope\_HandleReceived(), fipa\_envelope\_HandleTo(), MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, mxmlFindElement(), fipa\_acl\_envelope\_s::num\_params, and fipa\_acl\_envelope\_s::params.

Referenced by fipa\_envelope\_HandleEnvelope().

### 5.15.1.26 **int fipa\_envelope\_HandlePayloadEncoding** (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)

Definition at line 428 of file fipa\_envelope.c.

References mxml\_node\_s::child, MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_TEXT, mxmlFindElement(), fipa\_acl\_envelope\_s::params, fipa\_acl\_Param\_s::payload\_encoding, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by fipa\_envelope\_HandleParams().

**5.15.1.27 int fipa\_envelope\_HandlePayloadLength (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)**

Definition at line 405 of file fipa\_envelope.c.

References mxml\_node\_s::child, MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_TEXT, mxmlFindElement(), fipa\_acl\_envelope\_s::params, fipa\_acl\_Param\_s::payload\_length, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by fipa\_envelope\_HandleParams().

**5.15.1.28 int fipa\_envelope\_HandleReceived (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)**

Definition at line 560 of file fipa\_envelope.c.

References fipa\_acl\_envelope\_Received\_New(), fipa\_datetime\_Parse(), fipa\_url\_New(), MC\_ERR\_PARSE, fipa\_message\_string\_s::message, MXML\_DESCEND\_FIRST, mxmlElementGetAttr(), mxmlFindElement(), node, fipa\_acl\_envelope\_s::params, fipa\_message\_string\_s::parse, fipa\_acl\_Param\_s::received, fipa\_acl\_envelope\_Received\_s::received\_by, fipa\_acl\_envelope\_Received\_s::received\_date, fipa\_acl\_envelope\_Received\_s::received\_from, fipa\_acl\_envelope\_Received\_s::received\_id, fipa\_acl\_envelope\_Received\_s::received\_via, and fipa\_url\_s::str.

Referenced by fipa\_envelope\_HandleParams().

**5.15.1.29 int fipa\_envelope\_HandleTo (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)**

Definition at line 250 of file fipa\_envelope.c.

References fipa\_acl\_Param\_New(), fipa\_agent\_identifier\_set\_New(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_envelope\_ParseAgentIdentifier(), MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_NO\_DESCEND, mxmlFindElement(), fipa\_agent\_identifier\_set\_s::num, fipa\_acl\_envelope\_s::params, fipa\_agent\_identifier\_set\_s::retain\_order, and fipa\_acl\_Param\_s::to.

Referenced by fipa\_envelope\_HandleParams().

**5.15.1.30 int fipa\_envelope\_Parse (struct fipa\_acl\_envelope\_s \* *envelope*, const char \* *message*)**

Definition at line 165 of file fipa\_envelope.c.

References fipa\_envelope\_HandleEnvelope(), MXML\_NO\_CALLBACK, mxmlDelete(), and mxmlLoadString().

Referenced by acc\_connection\_Thread().

**5.15.1.31 int fipa\_envelope\_ParseAddresses (struct fipa\_agent\_identifier\_s \* *aid*, mxml\_node\_t \* *addresses\_node*)**

Definition at line 737 of file fipa\_envelope.c.

References fipa\_agent\_identifier\_s::addresses, mxml\_node\_s::child, fipa\_url\_New(), MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_NO\_DESCEND, MXML\_TEXT, mxmlFindElement(), fipa\_url\_sequence\_s::num, fipa\_url\_s::str, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, fipa\_url\_sequence\_s::urls, and mxml\_node\_s::value.

Referenced by `fipa_envelope_ParseAgentIdentifier()`.

**5.15.1.32** `int fipa_envelope_ParseAgentIdentifier (struct fipa_agent_identifier_s ** aid,  
mxml_node_t * agent_identifier_node)`

Definition at line 687 of file `fipa_envelope.c`.

References `mxml_node_s::child`, `fipa_agent_identifier_New()`, `fipa_envelope_ParseAddresses()`, `fipa_envelope_ParseResolvers()`, `MC_ERR_PARSE`, `MXML_DESCEND_FIRST`, `MXML_TEXT`, `mxmlFindElement()`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::type`, and `mxml_node_s::value`.

Referenced by `fipa_envelope_HandleFrom()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleTo()`, and `fipa_envelope_ParseResolvers()`.

**5.15.1.33** `int fipa_envelope_ParseResolvers (struct fipa_agent_identifier_s * aid, mxml_node_t *  
resolvers_node)`

Definition at line 804 of file `fipa_envelope.c`.

References `fipa_agent_identifier_set_s::fipa_agent_identifiers`, `fipa_envelope_ParseAgentIdentifier()`, `MC_ERR_PARSE`, `MXML_DESCEND_FIRST`, `MXML_NO_DESCEND`, `mxmlFindElement()`, `fipa_agent_identifier_set_s::num`, `fipa_agent_identifier_s::resolvers`, and `fipa_agent_identifier_set_s::retain_order`.

Referenced by `fipa_envelope_ParseAgentIdentifier()`.

## 5.16 /home/dko/Projects/mobilec/src/include/acc.h File Reference

```
#include <sys/socket.h>
#include "macros.h"
```

### Functions

- **STRUCT** (acc, struct [mc\\_platform\\_s](#) \*[mc\\_platform](#); [THREAD\\_T](#) [thread](#); [THREAD\\_T](#) [message\\_handler\\_thread](#); [THREAD\\_T](#) [listen\\_thread](#); [THREAD\\_T](#) [udplisten\\_thread](#); [int](#) [waiting](#); [MUTEX\\_T](#) \*[waiting\\_lock](#); [COND\\_T](#) \*[waiting\\_cond](#); [int](#) [num\\_conn\\_threads](#); [THREAD\\_T](#) [conn\\_thread](#); [MUTEX\\_T](#) [conn\\_thread\\_lock](#); [COND\\_T](#) [conn\\_thread\\_cond](#); struct [connection\\_thread\\_arg\\_s](#) \*[connection\\_thread\\_arg](#); [int](#) [num\\_msg\\_threads](#); [THREAD\\_T](#) [msg\\_thread](#); [MUTEX\\_T](#) [msg\\_thread\\_lock](#); [COND\\_T](#) [msg\\_thread\\_cond](#);) **STRUCT**([listen\\_thread\\_arg](#)
- **STRUCT** ([connection\\_thread\\_arg](#), struct [mc\\_platform\\_s](#) \*[mc\\_platform](#); struct [connection\\_s](#) \*[connection](#);) [acc\\_p](#) [acc\\_Initialize](#)(struct [mc\\_platform\\_s](#) \*[mc\\_platform](#))
- [int](#) [acc\\_Destroy](#) ([acc\\_p](#) [acc](#))
- void [acc\\_Start](#) (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#))
- void \* [acc\\_MessageHandlerThread](#) (void \*[arg](#))
- void \* [acc\\_Thread](#) (void \*[arg](#))
- void \* [acc\\_connection\\_Thread](#) (void \*[arg](#))
- void \* [listen\\_Thread](#) (void \*[arg](#))
- void \* [udplisten\\_Thread](#) (void \*[arg](#))

### Variables

- [u\\_long](#) [port](#)
- struct [sockaddr\\_in](#) \* [addr](#)
- unsigned long [int](#) [client\\_fd](#)
- unsigned long [int](#) [server\\_fd](#)

### 5.16.1 Function Documentation

#### 5.16.1.1 void\* acc\_connection\_Thread (void \* arg)

Definition at line 399 of file [acc.c](#).

References [mc\\_platform\\_s::acc](#), [fipa\\_agent\\_identifier\\_s::addresses](#), [agent\\_mailbox\\_Post\(\)](#), [mc\\_platform\\_s::agent\\_queue](#), [AGENT\\_UPDATE](#), [CANCEL](#), [CONNECT\\_THREAD\\_EXIT](#), [connection\\_Destroy\(\)](#), [mtp\\_http\\_s::content](#), [mtp\\_http\\_content\\_s::data](#), [FIPA\\_ACL](#), [fipa\\_acl\\_envelope\\_Destroy\(\)](#), [fipa\\_acl\\_envelope\\_New\(\)](#), [fipa\\_acl\\_message\\_Destroy\(\)](#), [fipa\\_acl\\_message\\_New\(\)](#), [fipa\\_acl\\_Parse\(\)](#), [fipa\\_agent\\_identifier\\_set\\_s::fipa\\_agent\\_identifiers](#), [fipa\\_envelope\\_Parse\(\)](#), [fipa\\_message\\_string\\_Destroy\(\)](#), [fipa\\_message\\_string\\_New\(\)](#), [mtp\\_http\\_s::http\\_performative](#), [HTTP\\_POST](#), [HTTP\\_PUT](#), [agent\\_s::mailbox](#), [agent\\_s::mc\\_platform](#), [mc\\_platform](#), [fipa\\_message\\_string\\_s::message](#), [message\\_s::message\\_body](#), [message\\_Destroy\(\)](#), [message\\_New\(\)](#), [mtp\\_http\\_s::message\\_parts](#), [mc\\_platform\\_s::message\\_queue](#), [message\\_s::message\\_type](#), [message\\_xml\\_parse\(\)](#), [MOBILE\\_AGENT](#), [mtp\\_http\\_Destroy\(\)](#), [mtp\\_http\\_InitializeFromConnection\(\)](#), [mtp\\_http\\_New\(\)](#), [MXML\\_NO\\_CALLBACK](#), [mxmlloadString\(\)](#), [N\\_UNDRSTD](#), [fipa\\_agent\\_identifier\\_s::name](#), [fipa\\_agent\\_identifier\\_set\\_s::num](#), [fipa\\_acl\\_envelope\\_s::num\\_params](#), [fipa\\_acl\\_envelope\\_s::params](#), [fipa\\_message\\_string\\_s::parse](#), [mc\\_platform\\_s::port](#), [mc\\_platform\\_s::private\\_key](#), [QUER\\_IF](#), [QUER\\_REF](#), [RELAY](#), [REQUEST](#), [RETURN\\_MSG](#), [fipa\\_url\\_s::str](#), [strdup](#), [SUBSCRIBE](#), [mtp\\_http\\_s::target](#), [fipa\\_acl\\_Param\\_s::to](#), [fipa\\_url\\_sequence\\_s::urls](#), and [message\\_s::xml\\_root](#).

Referenced by `acc_Thread()`.

#### 5.16.1.2 `int acc_Destroy (acc_p acc)`

Definition at line 102 of file `acc.c`.

References `COND_DESTROY`, `MC_SUCCESS`, and `MUTEX_DESTROY`.

Referenced by `mc_platform_Destroy()`.

#### 5.16.1.3 `void* acc_MessageHandlerThread (void * arg)`

Definition at line 118 of file `acc.c`.

References `mc_platform_s::acc`, `agent_Initialize()`, `mc_platform_s::agent_queue`, `agent_s::agent_status`, `AGENT_UPDATE`, `mc_platform_s::ams`, `CANCEL`, `COND_BROADCAST`, `COND_WAIT`, `agent_s::datastate`, `FIPA_ACL`, `mc_platform_s::giant`, `mc_platform_s::giant_cond`, `mc_platform_s::giant_lock`, `agent_s::lock`, `MC_AGENT_NEUTRAL`, `mc_platform`, `MC_RECV_AGENT`, `MC_RECV_MESSAGE`, `MC_RECV_RETURN`, `mc_platform_s::MC_signal`, `mc_platform_s::MC_signal_cond`, `mc_platform_s::MC_signal_lock`, `message_Destroy()`, `mc_platform_s::message_queue`, `message_Send()`, `message_s::message_type`, `MOBILE_AGENT`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `N_UNDRSTD`, `agent_s::name`, `agent_datastate_s::persistent`, `QUER_IF`, `QUER_REF`, `mc_platform_s::quit`, `mc_platform_s::quit_lock`, `RELAY`, `REQUEST`, `RETURN_MSG`, `SUBSCRIBE`, `THREAD_EXIT`, and `message_s::to_address`.

Referenced by `acc_Start()`.

#### 5.16.1.4 `void acc_Start (struct mc_platform_s * mc_platform)`

#### 5.16.1.5 `void* acc_Thread (void * arg)`

Definition at line 300 of file `acc.c`.

References `mc_platform_s::acc`, `acc_connection_Thread()`, `COND_BROADCAST`, `COND_WAIT`, `CONN_THREADS`, `mc_platform_s::connection_queue`, `mc_platform_s::giant`, `mc_platform_s::giant_cond`, `mc_platform_s::giant_lock`, `mc_platform`, `MC_RECV_CONNECTION`, `mc_platform_s::MC_signal`, `mc_platform_s::MC_signal_cond`, `mc_platform_s::MC_signal_lock`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_platform_s::quit`, `mc_platform_s::quit_lock`, `THREAD_CREATE`, `THREAD_DETACH`, and `THREAD_EXIT`.

Referenced by `acc_Start()`.

#### 5.16.1.6 `void* listen_Thread (void * arg)`

Definition at line 663 of file `acc.c`.

References `mc_platform_s::acc`, `connection_s::addr`, `connection_s::AES_key`, `mc_platform_s::agency`, `auth_conn_rece_key()`, `BACKLOG`, `mc_platform_s::bluetooth`, `connection_s::clientfd`, `COND_BROADCAST`, `connection_s::connect_id`, `connection_New()`, `mc_platform_s::connection_queue`, `agency_s::known_host_filename`, `mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `connection_s::nonce`, `mc_platform_s::port`, `agency_s::portno`, `mc_platform_s::private_key`, `mc_platform_s::quit`, `connection_s::remote_hostname`, `connection_s::serverfd`, `SOCKET_ERROR`, `mc_platform_s::sockfd`, and `THREAD_EXIT`.

Referenced by `acc_Start()`.



**5.16.1.7** **STRUCT** (**connection\_thread\_arg**, **struct mc\_platform\_s \*mc\_platform**; **struct connection\_s \*connection**;

**5.16.1.8** **STRUCT** (**acc**, **struct mc\_platform\_s \*mc\_platform**; **THREAD\_T thread**; **THREAD\_T message\_handler\_thread**; **THREAD\_T listen\_thread**; **THREAD\_T udplisten\_thread**; **int waiting**; **MUTEX\_T \*waiting\_lock**; **COND\_T \*waiting\_cond**; **int num\_conn\_threads**; **THREAD\_T conn\_thread**; **MUTEX\_T conn\_thread\_lock**; **COND\_T conn\_thread\_cond**; **struct connection\_thread\_arg\_s \*connection\_thread\_arg**; **int num\_msg\_threads**; **THREAD\_T msg\_thread**; **MUTEX\_T msg\_thread\_lock**; **COND\_T msg\_thread\_cond**;

**5.16.1.9** **void\*** **udplisten\_Thread** (**void \*arg**)

Definition at line 918 of file acc.c.

References **buf**, **BUFLen**, **mc\_platform\_s::hostname**, **mc\_platform**, **PACKAGE\_VERSION**, **mc\_platform\_s::port**, and **UDPPORT**.

Referenced by **acc\_Start()**.

## 5.16.2 Variable Documentation

**5.16.2.1** **struct sockaddr\_in\*** **addr**

Definition at line 78 of file acc.h.

**5.16.2.2** **unsigned long int** **client\_fd**

Definition at line 79 of file acc.h.

Referenced by **main()**, and **ssl\_test()**.

**5.16.2.3** **u\_long** **port**

Definition at line 77 of file acc.h.

Referenced by **MC\_AclSend()**, **MC\_MigrateAgent\_chdl()**, **MC\_SendAgentMigrationMessage\_chdl()**, **MC\_SendAgentMigrationMessageFile\_chdl()**, **message\_send\_Thread()**, and **message\_xml\_parse\_\_message()**.

**5.16.2.4** **unsigned long int** **server\_fd**

Definition at line 80 of file acc.h.

Referenced by **main()**, and **ssl\_test()**.

## 5.17 /home/dko/Projects/mobilec/src/include/agent.h File Reference

```
#include "agent_datastate.h"
#include "agent_task.h"
#include "libmc.h"
#include "message.h"
#include "macros.h"
#include "data_structures.h"
#include "agent_mailbox.h"
```

### Data Structures

- struct [agent\\_s](#)

### Defines

- #define [AGENT\\_T](#)

### Typedefs

- typedef struct [agent\\_s](#) [agent\\_t](#)
- typedef [agent\\_t](#) \* [MCAgent\\_t](#)
- typedef [agent\\_t](#) \* [agent\\_p](#)

### Functions

- [int agent\\_AddPersistentVariable](#) ([agent\\_p](#) agent, [int](#) task\_num, const char \*var\_name)
- [agent\\_p agent\\_New](#) (void)
- [agent\\_p agent\\_NewBinary](#) (struct [mc\\_platform\\_s](#) \*mc\_platform)
- [agent\\_p agent\\_Copy](#) (const [agent\\_p](#) agent)
- [int agent\\_Destroy](#) ([agent\\_p](#) agent)
- [agent\\_p agent\\_Initialize](#) (struct [mc\\_platform\\_s](#) \*mc\_platform, [message\\_p](#) message, [int](#) id)
- void [agent\\_RunChScript](#) ([MCAgent\\_t](#) agent, struct [mc\\_platform\\_s](#) \*global)
- void \* [agent\\_RunChScriptThread](#) (void \*agent)
- void \* [agent\\_ChScriptInitVar](#) ([ChInterp\\_t](#) \*interp)

#### 5.17.1 Define Documentation

##### 5.17.1.1 #define AGENT\_T

Definition at line 108 of file agent.h.

## 5.17.2 Typedef Documentation

### 5.17.2.1 typedef agent\_t\* agent\_p

Definition at line 111 of file agent.h.

### 5.17.2.2 typedef struct agent\_s agent\_t

Definition at line 109 of file agent.h.

### 5.17.2.3 typedef agent\_t\* MCAgent\_t

Definition at line 110 of file agent.h.

## 5.17.3 Function Documentation

### 5.17.3.1 int agent\_AddPersistentVariable (agent\_p agent, int task\_num, const char \* var\_name)

Definition at line 52 of file agent.c.

References agent\_s::agent\_interp, agent\_task\_s::agent\_variable\_list, interpreter\_variable\_data\_s::array\_dim, interpreter\_variable\_data\_s::array\_extent, CH\_DATATYPE\_SIZE, CHECK\_NULL, interpreter\_variable\_data\_s::data, interpreter\_variable\_data\_s::data\_type, agent\_s::datastate, MC\_ERR, agent\_s::name, interpreter\_variable\_data\_s::name, interpreter\_variable\_data\_s::size, size, strdup, agent\_datastate\_s::task\_progress, and agent\_datastate\_s::tasks.

### 5.17.3.2 void\* agent\_ChScriptInitVar (ChInterp\_t \* interp)

Definition at line 491 of file agent.c.

References MC\_AclAddReceiver\_chdl(), MC\_AclAddReplyTo\_chdl(), MC\_AclDestroy\_chdl(), MC\_AclGetContent\_chdl(), MC\_AclGetConversationID\_chdl(), MC\_AclGetPerformative\_chdl(), MC\_AclGetProtocol\_chdl(), MC\_AclGetSender\_chdl(), MC\_AclNew\_chdl(), MC\_AclPost\_chdl(), MC\_AclReply\_chdl(), MC\_AclRetrieve\_chdl(), MC\_AclSend\_chdl(), MC\_AclSetContent\_chdl(), MC\_AclSetConversationID\_chdl(), MC\_AclSetPerformative\_chdl(), MC\_AclSetProtocol\_chdl(), MC\_AclSetSender\_chdl(), MC\_AclWaitRetrieve\_chdl(), MC\_AddAgent\_chdl(), MC\_AgentAddTask\_chdl(), MC\_AgentVariableRetrieve\_chdl(), MC\_AgentVariableSave\_chdl(), MC\_Barrier\_chdl(), MC\_BarrierDelete\_chdl(), MC\_BarrierInit\_chdl(), MC\_CallAgentFunc\_chdl(), MC\_ComposeAgent\_chdl(), MC\_ComposeAgentFromFile\_chdl(), MC\_ComposeAgentFromFileWithWorkgroup\_chdl(), MC\_ComposeAgentWithWorkgroup\_chdl(), MC\_CondBroadcast\_chdl(), MC\_CondReset\_chdl(), MC\_CondSignal\_chdl(), MC\_CondWait\_chdl(), MC\_DeleteAgent\_chdl(), MC\_DeleteAgentWG\_chdl(), MC\_DeregisterService\_chdl(), MC\_DestroyServiceSearchResult\_chdl(), MC\_End\_chdl(), MC\_FindAgentByID\_chdl(), MC\_FindAgentByName\_chdl(), MC\_GetAgentID\_chdl(), MC\_GetAgentName\_chdl(), MC\_GetAgentStatus\_chdl(), MC\_GetAgentXMLString\_chdl(), MC\_GetTimeOfDay\_chdl(), MC\_HaltAgency\_chdl(), MC\_MigrateAgent\_chdl(), MC\_MutexLock\_chdl(), MC\_MutexUnlock\_chdl(), MC\_PrintAgentCode\_chdl(), MC\_RegisterService\_chdl(), MC\_ResumeAgency\_chdl(), MC\_RetrieveAgent\_chdl(), MC\_RetrieveAgentCode\_chdl(), MC\_SaveData\_chdl(), MC\_SearchForService\_chdl(), MC\_SemaphorePost\_chdl(), MC\_SemaphoreWait\_chdl(), MC\_SendAgentMigrationMessage\_chdl(), MC\_SendAgentMigrationMessageFile\_chdl(), MC\_SendSteerCommand\_chdl(), MC\_SetAgentStatus\_chdl(), MC\_SetDefaultAgentStatus\_chdl(), MC\_SyncDelete\_chdl(), MC\_SyncInit\_chdl(), MC\_TerminateAgent\_chdl(), and MC\_TerminateAgentWG\_chdl().

Referenced by AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), and mc\_platform\_Initialize().

### 5.17.3.3 agent\_p agent\_Copy (const agent\_p agent)

Definition at line 134 of file agent.c.

References agent\_datastate\_Copy(), agent\_s::agent\_interp, agent\_mailbox\_New(), agent\_s::agent\_persistent, agent\_s::agent\_status, agent\_s::agent\_type, agent\_s::arrival\_time, agent\_s::datastate, agent\_s::home, agent\_s::home\_port, agent\_s::id, agent\_s::lock, agent\_s::mailbox, MUTEX\_INIT, MUTEX\_LOCK, MUTEX\_T, agent\_s::name, agent\_s::orphan, agent\_s::owner, agent\_s::return\_data, and agent\_s::run\_lock.

Referenced by MC\_CopyAgent().

### 5.17.3.4 int agent\_Destroy (agent\_p agent)

Definition at line 413 of file agent.c.

References agent\_datastate\_Destroy(), agent\_s::agent\_interp, agent\_mailbox\_Destroy(), agent\_s::agent\_status, agent\_s::datastate, agent\_s::home, mc\_platform\_s::interpreter\_queue, agent\_s::lock, agent\_s::mailbox, MC\_AGENT\_NEUTRAL, agent\_s::mc\_platform, MC\_SUCCESS, MUTEX\_DESTROY, MUTEX\_LOCK, agent\_s::name, agent\_s::owner, agent\_s::run\_lock, agent\_s::sender, and agent\_s::wg\_code.

Referenced by agent\_Initialize().

### 5.17.3.5 agent\_p agent\_Initialize (struct mc\_platform\_s \* mc\_platform, message\_p message, int id)

Definition at line 292 of file agent.c.

References agent\_s::agent\_address, agent\_datastate\_New(), agent\_Destroy(), agent\_mailbox\_New(), agent\_s::agent\_pipe\_active, agent\_s::agent\_pipe\_ready\_to\_read, agent\_s::agent\_ready\_to\_send, agent\_s::agent\_script\_ready, agent\_s::agent\_status, agent\_s::agent\_thread\_id, agent\_s::agent\_type, message\_s::agent\_xml\_flag, agent\_xml\_parse(), agent\_s::arrival\_time, agent\_s::datastate, mc\_platform\_s::default\_agentstatus, mc\_platform\_s::err, agent\_s::home, mc\_platform\_s::hostname, agent\_s::id, agent\_s::lock, agent\_s::mailbox, MC\_ERR\_PARSE, agent\_s::mc\_platform, MC\_REMOTE\_AGENT, MC\_RETURN\_AGENT, MC\_WAIT\_CH, message\_s::message\_type, MOBILE\_AGENT, MUTEX\_DESTROY, MUTEX\_INIT, MUTEX\_T, agent\_s::orphan, mc\_platform\_s::port, RETURN\_MSG, agent\_s::run\_lock, agent\_s::sender, agent\_datastate\_s::xml\_agent\_root, message\_s::xml\_payload, message\_s::xml\_root, and agent\_datastate\_s::xml\_root.

Referenced by acc\_MessageHandlerThread(), MC\_SendAgent(), MC\_SendAgentFile(), and MC\_SendAgentMigrationMessageFile().

### 5.17.3.6 agent\_p agent\_New (void)

Definition at line 197 of file agent.c.

References agent\_s::lock, MUTEX\_INIT, MUTEX\_NEW, and agent\_s::run\_lock.

Referenced by MC\_ComposeAgentWithWorkgroup().

**5.17.3.7 agent\_p agent\_NewBinary (struct mc\_platform\_s \* mc\_platform)**

Definition at line 220 of file agent.c.

References agent\_s::agent\_address, agent\_mailbox\_New(), agent\_s::agent\_pipe\_active, agent\_s::agent\_pipe\_ready\_to\_read, agent\_s::agent\_ready\_to\_send, agent\_s::agent\_script\_ready, agent\_s::agent\_status, agent\_s::agent\_thread\_id, agent\_s::arrival\_time, agent\_s::binary, agent\_s::home, mc\_platform\_s::binary, agent\_s::hostname, agent\_s::id, agent\_s::lock, agent\_s::mailbox, MC\_AGENT\_ACTIVE, agent\_s::mc\_platform, MUTEX\_INIT, MUTEX\_T, agent\_s::orphan, mc\_platform\_s::port, agent\_s::run\_lock, and agent\_s::sender.

Referenced by MC\_AddStationaryAgent().

**5.17.3.8 void agent\_RunChScript (MCAgent\_t agent, struct mc\_platform\_s \* global)****5.17.3.9 void\* agent\_RunChScriptThread (void \* agent)**

Definition at line 971 of file agent.c.

References mc\_platform\_s::agency, agent\_s::agent\_interp, agency\_s::agentInitCallback, agency\_s::agentInitUserData, mc\_platform\_s::ams, COND\_SIGNAL, agent\_s::datastate, agent\_s::id, mc\_platform\_s::interp\_options, mc\_platform\_s::interpreter\_queue, interpreter\_queue\_CreateRetrieve(), interpreter\_variable\_data\_Destroy(), interpreter\_variable\_data\_Initialize(), interpreter\_variable\_data\_InitializeFromAgent(), MC\_AGENT\_NEUTRAL, MC\_EXEC\_AGENT, agent\_s::mc\_platform, mc\_platform, MC\_RETURN\_AGENT, mc\_platform\_s::MC\_signal, mc\_platform\_s::MC\_signal\_cond, mc\_platform\_s::MC\_signal\_lock, MC\_WAIT\_FINISHED, MC\_WAIT\_MESSGSEND, MUTEX\_LOCK, MUTEX\_UNLOCK, agent\_s::name, SIGNAL, agent\_datastate\_s::task\_progress, and WARN.

Referenced by agent\_RunChScript().

## 5.18 /home/dko/Projects/mobilec/src/include/agent\_datastate.h File Reference

```
#include <mxml.h>
#include "agent_task.h"
```

### Data Structures

- struct [agent\\_datastate\\_s](#)

### Typedefs

- typedef struct [agent\\_datastate\\_s](#) [agent\\_datastate\\_t](#)
- typedef [agent\\_datastate\\_t](#) \* [agent\\_datastate\\_p](#)

### Functions

- [agent\\_datastate\\_p](#) [agent\\_datastate\\_Copy](#) (const [agent\\_datastate\\_p](#) *datastate*)
- [agent\\_datastate\\_p](#) [agent\\_datastate\\_New](#) (void)
- int [agent\\_datastate\\_Destroy](#) ([agent\\_datastate\\_p](#) *agent\_datastate*)

#### 5.18.1 Typedef Documentation

##### 5.18.1.1 typedef [agent\\_datastate\\_t](#)\* [agent\\_datastate\\_p](#)

Definition at line 73 of file [agent\\_datastate.h](#).

##### 5.18.1.2 typedef struct [agent\\_datastate\\_s](#) [agent\\_datastate\\_t](#)

#### 5.18.2 Function Documentation

##### 5.18.2.1 [agent\\_datastate\\_p](#) [agent\\_datastate\\_Copy](#) (const [agent\\_datastate\\_p](#) *datastate*)

Definition at line 47 of file [agent\\_datastate.c](#).

References [agent\\_datastate\\_s::agent\\_code](#), [agent\\_datastate\\_s::agent\\_code\\_ids](#), [agent\\_datastate\\_s::agent\\_codes](#), [agent\\_datastate\\_New\(\)](#), [agent\\_task\\_Copy\(\)](#), [agent\\_datastate\\_s::init\\_agent\\_status](#), [agent\\_datastate\\_s::number\\_of\\_tasks](#), [agent\\_datastate\\_s::persistent](#), [agent\\_datastate\\_s::return\\_data](#), [agent\\_datastate\\_s::task\\_progress](#), and [agent\\_datastate\\_s::tasks](#).

Referenced by [agent\\_Copy\(\)](#).

##### 5.18.2.2 int [agent\\_datastate\\_Destroy](#) ([agent\\_datastate\\_p](#) *agent\_datastate*)

Definition at line 136 of file [agent\\_datastate.c](#).

References [agent\\_datastate\\_s::agent\\_code\\_ids](#), [agent\\_datastate\\_s::agent\\_codes](#), [agent\\_task\\_Destroy\(\)](#), [MC\\_SUCCESS](#), [mxmlDelete\(\)](#), [agent\\_datastate\\_s::number\\_of\\_tasks](#), [agent\\_datastate\\_s::tasks](#), and [agent\\_datastate\\_s::xml\\_root](#).

Referenced by agent\_Destroy().

### 5.18.2.3 agent\_datastate\_p agent\_datastate\_New (void)

Definition at line 115 of file agent\_datastate.c.

References agent\_datastate\_s::agent\_code, CHECK\_NULL, agent\_datastate\_s::init\_agent\_status, agent\_datastate\_s::number\_of\_tasks, agent\_datastate\_s::persistent, agent\_datastate\_s::progress\_modifier, agent\_datastate\_s::return\_data, agent\_datastate\_s::task\_progress, agent\_datastate\_s::tasks, agent\_datastate\_s::xml\_agent\_root, and agent\_datastate\_s::xml\_root.

Referenced by agent\_datastate\_Copy(), agent\_Initialize(), and MC\_ComposeAgentWithWorkgroup().

## 5.19 /home/dko/Projects/mobilec/src/include/agent\_lib.h File Reference

### Functions

- EXPORTCH [int MC\\_AclGetProtocol\\_chdl](#) (void \*varg)
- EXPORTCH [char \\* MC\\_AclGetConversationID\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclGetPerformative\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclGetSender\\_chdl](#) (void \*varg)
- EXPORTCH [char \\* MC\\_AclGetContent\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclDestroy\\_chdl](#) (void \*varg)
- EXPORTCH [void \\* MC\\_AclNew\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclPost\\_chdl](#) (void \*varg)
- EXPORTCH [void \\* MC\\_AclReply\\_chdl](#) (void \*varg)
- EXPORTCH [void \\* MC\\_AclRetrieve\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSend\\_chdl](#) (void \*varg)
- EXPORTCH [void \\* MC\\_AclWaitRetrieve\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSetProtocol\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSetConversationID\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSetPerformative\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSetSender\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclAddReceiver\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclAddReplyTo\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSetContent\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AddAgent\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AgentAddTask\\_chdl](#) (void \*varg)
- EXPORTCH [const void \\* MC\\_AgentVariableRetrieve\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AgentVariableSave\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_Barrier\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_BarrierDelete\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_BarrierInit\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_CallAgentFunc\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_ComposeAgent\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_ComposeAgentWithWorkgroup\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_ComposeAgentFromFile\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_ComposeAgentFromFileWithWorkgroup\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_CondBroadcast\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_CondWait\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_CondReset\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_CondSignal\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_DeleteAgent\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_DeleteAgentWG\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_DeregisterService\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_DestroyServiceSearchResult\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_End\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_FindAgentByID\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_FindAgentByName\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_GetAgentID\\_chdl](#) (void \*varg)
- EXPORTCH [char \\* MC\\_GetAgentName\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_GetAgentNumTasks\\_chdl](#) (void \*varg)



- EXPORTCH char \* [MC\\_GetAgentXMLString\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_GetTimeOfDay\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_HaltAgency\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_PrintAgentCode\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_MigrateAgent\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_MutexLock\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_MutexUnlock\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_RegisterService\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_ResumeAgency\\_chdl](#) (void \*varg)
- EXPORTCH MC\_Agent\_t [MC\\_RetrieveAgent\\_chdl](#) (void \*varg)
- EXPORTCH char \* [MC\\_RetrieveAgentCode\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_SearchForService\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_SemaphorePost\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_SemaphoreWait\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_SendSteerCommand\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_TerminateAgent\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_TerminateAgentWG\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_GetAgentStatus\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_SaveData\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_SendAgentMigrationMessage\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_SendAgentMigrationMessageFile\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_SetAgentStatus\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_SetDefaultAgentStatus\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_SyncDelete\\_chdl](#) (void \*varg)
- EXPORTCH int [MC\\_SyncInit\\_chdl](#) (void \*varg)

## 5.19.1 Function Documentation

### 5.19.1.1 EXPORTCH int MC\_AclAddReceiver\_chdl (void \* *varg*)

Definition at line 2873 of file libmc.c.

References [MC\\_AclAddReceiver\(\)](#).

Referenced by [agent\\_ChScriptInitVar\(\)](#).

### 5.19.1.2 EXPORTCH int MC\_AclAddReplyTo\_chdl (void \* *varg*)

Definition at line 2893 of file libmc.c.

References [MC\\_AclAddReplyTo\(\)](#).

Referenced by [agent\\_ChScriptInitVar\(\)](#).

### 5.19.1.3 EXPORTCH int MC\_AclDestroy\_chdl (void \* *varg*)

Definition at line 2605 of file libmc.c.

References [MC\\_AclDestroy\(\)](#).

Referenced by [agent\\_ChScriptInitVar\(\)](#).

**5.19.1.4 EXPORTCH char\* MC\_AclGetContent\_chdl (void \* *varg*)**

Definition at line 2784 of file libmc.c.

References MC\_AclGetContent().

Referenced by agent\_ChScriptInitVar().

**5.19.1.5 EXPORTCH char\* MC\_AclGetConversationID\_chdl (void \* *varg*)**

Definition at line 2735 of file libmc.c.

References MC\_AclGetConversationID().

Referenced by agent\_ChScriptInitVar().

**5.19.1.6 EXPORTCH int MC\_AclGetPerformative\_chdl (void \* *varg*)**

Definition at line 2750 of file libmc.c.

References MC\_AclGetPerformative().

Referenced by agent\_ChScriptInitVar().

**5.19.1.7 EXPORTCH int MC\_AclGetProtocol\_chdl (void \* *varg*)**

Definition at line 2720 of file libmc.c.

References MC\_AclGetProtocol().

Referenced by agent\_ChScriptInitVar().

**5.19.1.8 EXPORTCH int MC\_AclGetSender\_chdl (void \* *varg*)**

Definition at line 2765 of file libmc.c.

References MC\_AclGetSender().

Referenced by agent\_ChScriptInitVar().

**5.19.1.9 EXPORTCH void\* MC\_AclNew\_chdl (void \* *varg*)**

Definition at line 2620 of file libmc.c.

References MC\_AclNew().

Referenced by agent\_ChScriptInitVar().

**5.19.1.10 EXPORTCH int MC\_AclPost\_chdl (void \* *varg*)**

Definition at line 2628 of file libmc.c.

References MC\_AclPost().

Referenced by agent\_ChScriptInitVar().

**5.19.1.11 EXPORTCH void\* MC\_AclReply\_chdl (void \* *varg*)**

Definition at line 2646 of file libmc.c.

References MC\_AclReply().

Referenced by agent\_ChScriptInitVar().

**5.19.1.12 EXPORTCH void\* MC\_AclRetrieve\_chdl (void \* *varg*)**

Definition at line 2662 of file libmc.c.

References MC\_AclRetrieve().

Referenced by agent\_ChScriptInitVar().

**5.19.1.13 EXPORTCH int MC\_AclSend\_chdl (void \* *varg*)**

Definition at line 2678 of file libmc.c.

References CHECK\_NULL, MC\_AclSend(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.14 EXPORTCH int MC\_AclSetContent\_chdl (void \* *varg*)**

Definition at line 2913 of file libmc.c.

References fipa\_acl\_message\_s::content, and MC\_AclSetContent().

Referenced by agent\_ChScriptInitVar().

**5.19.1.15 EXPORTCH int MC\_AclSetConversationID\_chdl (void \* *varg*)**

Definition at line 2817 of file libmc.c.

References MC\_AclSetConversationID().

Referenced by agent\_ChScriptInitVar().

**5.19.1.16 EXPORTCH int MC\_AclSetPerformative\_chdl (void \* *varg*)**

Definition at line 2835 of file libmc.c.

References MC\_AclSetPerformative(), and fipa\_acl\_message\_s::performative.

Referenced by agent\_ChScriptInitVar().

**5.19.1.17 EXPORTCH int MC\_AclSetProtocol\_chdl (void \* *varg*)**

Definition at line 2799 of file libmc.c.

References MC\_AclSetProtocol(), and fipa\_acl\_message\_s::protocol.

Referenced by agent\_ChScriptInitVar().

**5.19.1.18 EXPORTCH int MC\_AclSetSender\_chdl (void \* *varg*)**

Definition at line 2853 of file libmc.c.

References MC\_AclSetSender().

Referenced by agent\_ChScriptInitVar().

**5.19.1.19 EXPORTCH void\* MC\_AclWaitRetrieve\_chdl (void \* *varg*)**

Definition at line 2702 of file libmc.c.

References MC\_AclWaitRetrieve().

Referenced by agent\_ChScriptInitVar().

**5.19.1.20 EXPORTCH int MC\_AddAgent\_chdl (void \* *varg*)**

Definition at line 2933 of file libmc.c.

References CHECK\_NULL, MC\_AddAgent(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.21 EXPORTCH int MC\_AgentAddTask\_chdl (void \* *varg*)**

Definition at line 2957 of file libmc.c.

References MC\_AgentAddTask().

Referenced by agent\_ChScriptInitVar().

**5.19.1.22 EXPORTCH const void\* MC\_AgentVariableRetrieve\_chdl (void \* *varg*)**

Definition at line 2991 of file libmc.c.

References MC\_AgentVariableRetrieve().

Referenced by agent\_ChScriptInitVar().

**5.19.1.23 EXPORTCH int MC\_AgentVariableSave\_chdl (void \* *varg*)**

Definition at line 3015 of file libmc.c.

References MC\_AgentVariableSave().

Referenced by agent\_ChScriptInitVar().

**5.19.1.24 EXPORTCH int MC\_Barrier\_chdl (void \* *varg*)**

Definition at line 3067 of file libmc.c.

References CHECK\_NULL, MC\_Barrier(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.25 EXPORTCH int MC\_BarrierDelete\_chdl (void \* *varg*)**

Definition at line 3090 of file libmc.c.

References CHECK\_NULL, MC\_BarrierDelete(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.26 EXPORTCH int MC\_BarrierInit\_chdl (void \* *varg*)**

Definition at line 3113 of file libmc.c.

References CHECK\_NULL, MC\_BarrierInit(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.27 EXPORTCH int MC\_CallAgentFunc\_chdl (void \* *varg*)**

Definition at line 3037 of file libmc.c.

References MC\_CallAgentFuncVar().

Referenced by agent\_ChScriptInitVar().

**5.19.1.28 EXPORTCH MCAgent\_t MC\_ComposeAgent\_chdl (void \* *varg*)**

Definition at line 3161 of file libmc.c.

References MC\_ComposeAgentWithWorkgroup().

Referenced by agent\_ChScriptInitVar().

**5.19.1.29 EXPORTCH MCAgent\_t MC\_ComposeAgentFromFile\_chdl (void \* *varg*)**

Definition at line 3245 of file libmc.c.

References MC\_ComposeAgentFromFileWithWorkgroup().

Referenced by agent\_ChScriptInitVar().

**5.19.1.30 EXPORTCH MCAgent\_t MC\_ComposeAgentFromFileWithWorkgroup\_chdl (void \* *varg*)**

Definition at line 3286 of file libmc.c.

References MC\_ComposeAgentFromFileWithWorkgroup().

Referenced by agent\_ChScriptInitVar().

**5.19.1.31 EXPORTCH MCAgent\_t MC\_ComposeAgentWithWorkgroup\_chdl (void \* *varg*)**

Definition at line 3202 of file libmc.c.

References MC\_ComposeAgentS().

Referenced by agent\_ChScriptInitVar().

**5.19.1.32 EXPORTCH int MC\_CondBroadcast\_chdl (void \* *varg*)**

Definition at line 3138 of file libmc.c.

References CHECK\_NULL, MC\_CondBroadcast(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.33 EXPORTCH int MC\_CondReset\_chdl (void \* *varg*)**

Definition at line 3352 of file libmc.c.

References CHECK\_NULL, MC\_CondReset(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.34 EXPORTCH int MC\_CondSignal\_chdl (void \* *varg*)**

Definition at line 3329 of file libmc.c.

References CHECK\_NULL, MC\_CondSignal(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.35 EXPORTCH int MC\_CondWait\_chdl (void \* *varg*)**

Definition at line 3375 of file libmc.c.

References CHECK\_NULL, MC\_CondWait(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.36 EXPORTCH int MC\_DeleteAgent\_chdl (void \* *varg*)**

Definition at line 3397 of file libmc.c.

References MC\_DeleteAgent(), MC\_ERR\_NOT\_FOUND, and MC\_FindAgentByName().

Referenced by agent\_ChScriptInitVar().

**5.19.1.37 EXPORTCH int MC\_DeleteAgentWG\_chdl (void \* *varg*)**

Definition at line 3417 of file libmc.c.

References MC\_DeleteAgentWG(), MC\_ERR\_NOT\_FOUND, and MC\_FindAgentByName().

Referenced by agent\_ChScriptInitVar().

**5.19.1.38 EXPORTCH int MC\_DeregisterService\_chdl (void \* *varg*)**

Definition at line 3466 of file libmc.c.

References CHECK\_NULL, MC\_DeregisterService(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.39 EXPORTCH int MC\_DestroyServiceSearchResult\_chdl (void \* *varg*)**

Definition at line 3439 of file libmc.c.

References MC\_DestroyServiceSearchResult().

Referenced by agent\_ChScriptInitVar().

**5.19.1.40 EXPORTCH int MC\_End\_chdl (void \* *varg*)**

Definition at line 3493 of file libmc.c.

References CHECK\_NULL, MC\_End(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.41 EXPORTCH MCAgent\_t MC\_FindAgentByID\_chdl (void \* *varg*)**

Definition at line 3510 of file libmc.c.

References CHECK\_NULL, MC\_FindAgentByID(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.42 EXPORTCH MCAgent\_t MC\_FindAgentByName\_chdl (void \* *varg*)**

Definition at line 3533 of file libmc.c.

References CHECK\_NULL, MC\_FindAgentByName(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.43 EXPORTCH int MC\_GetAgentID\_chdl (void \* *varg*)**

Definition at line 3580 of file libmc.c.

References MC\_GetAgentID().

Referenced by agent\_ChScriptInitVar().

**5.19.1.44 EXPORTCH char\* MC\_GetAgentName\_chdl (void \* *varg*)**

Definition at line 3596 of file libmc.c.

References MC\_GetAgentName().

Referenced by agent\_ChScriptInitVar().

**5.19.1.45 EXPORTCH int MC\_GetAgentNumTasks\_chdl (void \* *varg*)**

Definition at line 3612 of file libmc.c.

References MC\_GetAgentNumTasks().

**5.19.1.46 EXPORTCH int MC\_GetAgentStatus\_chdl (void \* *varg*)**

Definition at line 3628 of file libmc.c.

References MC\_GetAgentStatus().

Referenced by agent\_ChScriptInitVar().

**5.19.1.47 EXPORTCH char\* MC\_GetAgentXMLString\_chdl (void \* *varg*)**

Definition at line 3644 of file libmc.c.

References MC\_GetAgentXMLString().

Referenced by agent\_ChScriptInitVar().

**5.19.1.48 EXPORTCH int MC\_GetTimeOfDay\_chdl (void \* *varg*)**

Definition at line 3660 of file libmc.c.

Referenced by agent\_ChScriptInitVar().

**5.19.1.49 EXPORTCH int MC\_HaltAgency\_chdl (void \* *varg*)**

Definition at line 3675 of file libmc.c.

References CHECK\_NULL, MC\_HaltAgency(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.50 EXPORTCH int MC\_MigrateAgent\_chdl (void \* *varg*)**

Definition at line 3692 of file libmc.c.

References MC\_MigrateAgent(), and port.

Referenced by agent\_ChScriptInitVar().

**5.19.1.51 EXPORTCH int MC\_MutexLock\_chdl (void \* *varg*)**

Definition at line 3711 of file libmc.c.

References CHECK\_NULL, MC\_MutexLock(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.19.1.52 EXPORTCH int MC\_MutexUnlock\_chdl (void \* *varg*)**

Definition at line 3735 of file libmc.c.

References CHECK\_NULL, MC\_MutexUnlock(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().



**5.19.1.53 EXPORTCH int MC\_PrintAgentCode\_chdl (void \* *varg*)**

Definition at line 3759 of file libmc.c.

References MC\_PrintAgentCode().

Referenced by agent\_ChScriptInitVar().

**5.19.1.54 EXPORTCH int MC\_RegisterService\_chdl (void \* *varg*)**

Definition at line 3775 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_RegisterService().

Referenced by agent\_ChScriptInitVar().

**5.19.1.55 EXPORTCH int MC\_ResumeAgency\_chdl (void \* *varg*)**

Definition at line 3812 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_ResumeAgency().

Referenced by agent\_ChScriptInitVar().

**5.19.1.56 EXPORTCH MCAgent\_t MC\_RetrieveAgent\_chdl (void \* *varg*)**

Definition at line 3830 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_RetrieveAgent().

Referenced by agent\_ChScriptInitVar().

**5.19.1.57 EXPORTCH char\* MC\_RetrieveAgentCode\_chdl (void \* *varg*)**

Definition at line 3847 of file libmc.c.

References MC\_RetrieveAgentCode().

Referenced by agent\_ChScriptInitVar().

**5.19.1.58 EXPORTCH int MC\_SaveData\_chdl (void \* *varg*)**

Definition at line 3863 of file libmc.c.

References agent\_task\_s::agent\_variable\_list, interpreter\_variable\_data\_s::data, interpreter\_variable\_data\_s::data\_type, agent\_s::datastate, interpreter\_variable\_data\_New(), interpreter\_variable\_data\_s::name, interpreter\_variable\_data\_s::size, size, strdup, agent\_datastate\_s::task\_progress, and agent\_datastate\_s::tasks.

Referenced by agent\_ChScriptInitVar().

**5.19.1.59 EXPORTCH int MC\_SearchForService\_chdl (void \* *varg*)**

Definition at line 3898 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_SearchForService().

Referenced by agent\_ChScriptInitVar().

#### **5.19.1.60 EXPORTCH int MC\_SemaphorePost\_chdl (void \* *varg*)**

Definition at line 3939 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_SemaphorePost().

Referenced by agent\_ChScriptInitVar().

#### **5.19.1.61 EXPORTCH int MC\_SemaphoreWait\_chdl (void \* *varg*)**

Definition at line 3963 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_SemaphoreWait().

Referenced by agent\_ChScriptInitVar().

#### **5.19.1.62 EXPORTCH int MC\_SendAgentMigrationMessage\_chdl (void \* *varg*)**

Definition at line 3987 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, MC\_SendAgentMigrationMessage(), and port.

Referenced by agent\_ChScriptInitVar().

#### **5.19.1.63 EXPORTCH int MC\_SendAgentMigrationMessageFile\_chdl (void \* *varg*)**

Definition at line 4012 of file libmc.c.

References MC\_SendAgentMigrationMessageFile(), and port.

Referenced by agent\_ChScriptInitVar().

#### **5.19.1.64 EXPORTCH int MC\_SendSteerCommand\_chdl (void \* *varg*)**

Definition at line 4032 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_SendSteerCommand().

Referenced by agent\_ChScriptInitVar().

#### **5.19.1.65 EXPORTCH int MC\_SetAgentStatus\_chdl (void \* *varg*)**

Definition at line 4055 of file libmc.c.

References MC\_SetAgentStatus().

Referenced by agent\_ChScriptInitVar().

#### **5.19.1.66 EXPORTCH int MC\_SetDefaultAgentStatus\_chdl (void \* *varg*)**

Definition at line 4073 of file libmc.c.

References CHECK\_NULL, int, agency\_s::mc\_platform, and MC\_SetDefaultAgentStatus().

Referenced by agent\_ChScriptInitVar().

#### **5.19.1.67 EXPORTCH int MC\_SyncDelete\_chdl (void \* *varg*)**

Definition at line 4096 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_SyncDelete().

Referenced by agent\_ChScriptInitVar().

#### **5.19.1.68 EXPORTCH int MC\_SyncInit\_chdl (void \* *varg*)**

Definition at line 4119 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_SyncInit().

Referenced by agent\_ChScriptInitVar().

#### **5.19.1.69 EXPORTCH int MC\_TerminateAgent\_chdl (void \* *varg*)**

Definition at line 4143 of file libmc.c.

References MC\_ERR\_NOT\_FOUND, MC\_FindAgentByName(), and MC\_TerminateAgent().

Referenced by agent\_ChScriptInitVar().

#### **5.19.1.70 EXPORTCH int MC\_TerminateAgentWG\_chdl (void \* *varg*)**

Definition at line 4164 of file libmc.c.

References MC\_ERR\_NOT\_FOUND, MC\_FindAgentByName(), and MC\_TerminateAgentWG().

Referenced by agent\_ChScriptInitVar().

## 5.20 /home/dko/Projects/mobilec/src/include/agent\_mailbox.h File Reference

### Data Structures

- struct [agent\\_mailbox\\_s](#)

### Typedefs

- typedef struct [agent\\_mailbox\\_s](#) [agent\\_mailbox\\_t](#)
- typedef [agent\\_mailbox\\_t](#) \* [agent\\_mailbox\\_p](#)

### Functions

- [agent\\_mailbox\\_p](#) [agent\\_mailbox\\_New](#) (void)
- [agent\\_mailbox\\_p](#) [agent\\_mailbox\\_Copy](#) ([agent\\_mailbox\\_p](#) src)
- int [agent\\_mailbox\\_Destroy](#) ([agent\\_mailbox\\_t](#) \*mailbox)
- int [agent\\_mailbox\\_Post](#) ([agent\\_mailbox\\_p](#) mailbox, [fipa\\_acl\\_message\\_t](#) \*message)
- [fipa\\_acl\\_message\\_t](#) \* [agent\\_mailbox\\_Retrieve](#) ([agent\\_mailbox\\_p](#) mailbox)
- [fipa\\_acl\\_message\\_t](#) \* [agent\\_mailbox\\_WaitRetrieve](#) ([agent\\_mailbox\\_p](#) mailbox)

#### 5.20.1 Typedef Documentation

##### 5.20.1.1 typedef [agent\\_mailbox\\_t](#)\* [agent\\_mailbox\\_p](#)

Definition at line 12 of file [agent\\_mailbox.h](#).

##### 5.20.1.2 typedef struct [agent\\_mailbox\\_s](#) [agent\\_mailbox\\_t](#)

#### 5.20.2 Function Documentation

##### 5.20.2.1 [agent\\_mailbox\\_p](#) [agent\\_mailbox\\_Copy](#) ([agent\\_mailbox\\_p](#) src)

Definition at line 21 of file [agent\\_mailbox.c](#).

References [agent\\_mailbox\\_New\(\)](#), and [agent\\_mailbox\\_s::mail\\_queue](#).

##### 5.20.2.2 int [agent\\_mailbox\\_Destroy](#) ([agent\\_mailbox\\_t](#) \* mailbox)

Definition at line 29 of file [agent\\_mailbox.c](#).

References [agent\\_mailbox\\_s::mail\\_queue](#).

Referenced by [agent\\_Destroy\(\)](#).

##### 5.20.2.3 [agent\\_mailbox\\_p](#) [agent\\_mailbox\\_New](#) (void)

Definition at line 12 of file [agent\\_mailbox.c](#).

References [agent\\_mailbox\\_s::mail\\_queue](#).

Referenced by agent\_Copy(), agent\_Initialize(), agent\_mailbox\_Copy(), and agent\_NewBinary().

#### 5.20.2.4 int agent\_mailbox\_Post (agent\_mailbox\_p *mailbox*, fipa\_acl\_message\_t \* *message*)

Definition at line 38 of file agent\_mailbox.c.

References agent\_mailbox\_s::mail\_queue.

Referenced by acc\_connection\_Thread(), and MC\_AclPost().

#### 5.20.2.5 fipa\_acl\_message\_t\* agent\_mailbox\_Retrieve (agent\_mailbox\_p *mailbox*)

Definition at line 44 of file agent\_mailbox.c.

References agent\_mailbox\_s::mail\_queue.

Referenced by agent\_mailbox\_WaitRetrieve(), and MC\_AclRetrieve().

#### 5.20.2.6 fipa\_acl\_message\_t\* agent\_mailbox\_WaitRetrieve (agent\_mailbox\_p *mailbox*)

Definition at line 49 of file agent\_mailbox.c.

References agent\_mailbox\_Retrieve(), COND\_WAIT, agent\_mailbox\_s::mail\_queue, MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_AclWaitRetrieve().

## 5.21 /home/dko/Projects/mobilec/src/include/agent\_task.h File Reference

```
#include "interpreter_variable_data.h"
#include "data_structures.h"
```

### Data Structures

- struct [agent\\_task\\_s](#)

### Typedefs

- typedef struct [agent\\_task\\_s](#) [agent\\_task\\_t](#)
- typedef [agent\\_task\\_t](#) \* [agent\\_task\\_p](#)

### Functions

- [agent\\_task\\_p](#) [agent\\_task\\_New](#) (void)
- [agent\\_task\\_p](#) [agent\\_task\\_Copy](#) ([agent\\_task\\_p](#) task)
- int [agent\\_task\\_Destroy](#) ([agent\\_task\\_p](#) agent\_task)

#### 5.21.1 Typedef Documentation

##### 5.21.1.1 typedef [agent\\_task\\_t](#)\* [agent\\_task\\_p](#)

Definition at line 59 of file [agent\\_task.h](#).

##### 5.21.1.2 typedef struct [agent\\_task\\_s](#) [agent\\_task\\_t](#)

#### 5.21.2 Function Documentation

##### 5.21.2.1 [agent\\_task\\_p](#) [agent\\_task\\_Copy](#) ([agent\\_task\\_p](#) task)

Definition at line 66 of file [agent\\_task.c](#).

References [agent\\_task\\_s::agent\\_return\\_data](#), [agent\\_task\\_s::agent\\_variable\\_list](#), [agent\\_task\\_s::code\\_id](#), [agent\\_task\\_s::init\\_agent\\_status](#), [interpreter\\_variable\\_data\\_Copy\(\)](#), [ListAdd\(\)](#), [ListSearch\(\)](#), [agent\\_task\\_s::num\\_saved\\_variables](#), [agent\\_task\\_s::number\\_of\\_elements](#), [agent\\_task\\_s::persistent](#), [agent\\_task\\_s::saved\\_variables](#), [agent\\_task\\_s::server\\_name](#), [agent\\_task\\_s::size\\_of\\_element\\_array](#), [strdup](#), and [agent\\_task\\_s::var\\_name](#).

Referenced by [agent\\_datastate\\_Copy\(\)](#).

##### 5.21.2.2 int [agent\\_task\\_Destroy](#) ([agent\\_task\\_p](#) agent\_task)

Definition at line 132 of file [agent\\_task.c](#).

References `agent_task_s::agent_return_data`, `agent_task_s::agent_variable_list`, `agent_task_s::code_id`, `interpreter_variable_data_Destroy()`, `MC_SUCCESS`, `agent_task_s::saved_variables`, `agent_task_s::server_name`, and `agent_task_s::var_name`.

Referenced by `agent_datastate_Destroy()`.

### 5.21.2.3 `agent_task_p agent_task_New (void)`

Definition at line 48 of file `agent_task.c`.

References `agent_task_s::agent_variable_list`, `agent_task_s::num_saved_variables`, and `agent_task_s::saved_variables`.

Referenced by `agent_xml_parse__tasks()`, `MC_AgentAddTask()`, and `MC_ComposeAgentWithWorkgroup()`.

## 5.22 /home/dko/Projects/mobilec/src/include/ams.h File Reference

```
#include "macros.h"
```

### Functions

- **STRUCT** (ams, struct [mc\\_platform\\_s](#) \*[mc\\_platform](#);MUTEX\_T \*[runflag\\_lock](#);COND\_T \*[runflag\\_cond](#);int [run](#);int [waiting](#);MUTEX\_T \*[waiting\\_lock](#);COND\_T \*[waiting\\_cond](#);THREAD\_T [thread](#);) [ams\\_p](#) [ams\\_Initialize](#)(struct [mc\\_platform\\_s](#) \*\_[mc\\_platform](#))
- int [ams\\_Destroy](#) ([ams\\_p](#) [ams](#))
- int [ams\\_RemoveFinishedAgents](#) ([ams\\_p](#) [ams](#))
- int [ams\\_ManageAgentList](#) ([ams\\_p](#) [ams](#))
- void [ams\\_Print](#) ([ams\\_p](#) [ams](#))
- void [ams\\_Start](#) (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#))
- void \* [ams\\_Thread](#) (void \*[arg](#))

### 5.22.1 Function Documentation

#### 5.22.1.1 int [ams\\_Destroy](#) ([ams\\_p](#) [ams](#))

Definition at line 46 of file [ams.c](#).

References [COND\\_DESTROY](#), [MC\\_SUCCESS](#), and [MUTEX\\_DESTROY](#).

Referenced by [mc\\_platform\\_Destroy\(\)](#).

#### 5.22.1.2 int [ams\\_ManageAgentList](#) ([ams\\_p](#) [ams](#))

Definition at line 120 of file [ams.c](#).

References [agent\\_RunChScript\(\)](#), [agent\\_s::agent\\_status](#), [agent\\_s::binary](#), [ListSearch\(\)](#), [agent\\_s::lock](#), [MC\\_AGENT\\_ACTIVE](#), [MC\\_AGENT\\_NEUTRAL](#), [MC\\_TerminateAgent\(\)](#), [MC\\_WAIT\\_CH](#), [MC\\_WAIT\\_FINISHED](#), [MC\\_WAIT\\_MESSGSEND](#), [message\\_Destroy\(\)](#), [message\\_InitializeFromAgent\(\)](#), [message\\_New\(\)](#), [MUTEX\\_LOCK](#), [MUTEX\\_UNLOCK](#), [agent\\_s::name](#), [agent\\_s::orphan](#), [mc\\_platform\\_s::name](#), [agent\\_s::quit](#), [mc\\_platform\\_s::quit\\_lock](#), and [agent\\_s::run\\_lock](#).

Referenced by [ams\\_Thread\(\)](#).

#### 5.22.1.3 void [ams\\_Print](#) ([ams\\_p](#) [ams](#))

Definition at line 88 of file [ams.c](#).

References [agent\\_s::agent\\_status](#), [agent\\_s::connect\\_id](#), [agent\\_s::id](#), [ListSearch\(\)](#), [MUTEX\\_LOCK](#), and [MUTEX\\_UNLOCK](#).

#### 5.22.1.4 int [ams\\_RemoveFinishedAgents](#) ([ams\\_p](#) [ams](#))

#### 5.22.1.5 void [ams\\_Start](#) (struct [mc\\_platform\\_s](#) \* [mc\\_platform](#))

#### 5.22.1.6 void\* [ams\\_Thread](#) (void \* [arg](#))

Definition at line 258 of file [ams.c](#).



References mc\_platform\_s::ams, ams\_ManageAgentList(), COND\_BROADCAST, COND\_WAIT, mc\_platform, MUTEX\_LOCK, MUTEX\_UNLOCK, mc\_platform\_s::quit, mc\_platform\_s::quit\_lock, and THREAD\_EXIT.

Referenced by ams\_Start().

**5.22.1.7** **STRUCT** (ams, struct mc\_platform\_s \*mc\_platform;MUTEX\_T \*runflag\_lock;COND\_T \*runflag\_cond;int run;int waiting;MUTEX\_T \*waiting\_lock;COND\_T \*waiting\_cond;THREAD\_T thread;)

## 5.23 /home/dko/Projects/mobilec/src/include/ap\_queue\_template.h File Reference

```
#include "macros.h"
#include "mc_error.h"
#include <embedch.h>
```

### Data Structures

- struct [AP\\_GENERIC\\_s](#)

### Defines

- #define [AP\\_QUEUE\\_DECL\\_TEMPLATE](#)(name, node\_type)
- #define [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#)(name, func\_name, return\_type, search\_type) return\_type name##\_##\_func\_name(name##\_p name, const search\_type key);
- #define [AP\\_QUEUE\\_STD\\_DEFN\\_TEMPLATE](#)(name, node\_type)
- #define [AP\\_QUEUE\\_SEARCH\\_TEMPLATE](#)(name, func\_name, node\_type, search\_type, search\_expression)
- #define [AP\\_QUEUE\\_REMOVE\\_TEMPLATE](#)(name, func\_name, node\_type, search\_type, search\_expression)

### Typedefs

- typedef struct [AP\\_GENERIC\\_s](#) [AP\\_GENERIC\\_t](#)
- typedef [AP\\_GENERIC\\_t](#) \* [AP\\_GENERIC\\_p](#)

### Functions

- [AP\\_GENERIC\\_p](#) [interpreter\\_queue\\_CreateRetrieve](#) (struct [interpreter\\_queue\\_s](#) \*queue, [ChOptions\\_t](#) \*interp\_options)

#### 5.23.1 Define Documentation

##### 5.23.1.1 #define [AP\\_QUEUE\\_DECL\\_TEMPLATE](#)(name, node\_type)

##### Value:

```
typedef struct name##_s \
{ \
    int size; \
    list_p list; \
    MUTEX_T* lock; \
    COND_T* cond; \
} name##_t; \
\
typedef name##_t* name##_p; \
\
name##_p name##_New( void ); \
int name##_Destroy( name##_p name ); \
```

```

int name##_Add( name##_p name, struct node_type##_s* node ); \
name##_p name##_Copy( name##_p name ); \
struct node_type##_s* name##_Pop( name##_p name ); \
struct node_type##_s* name##_WaitPop( name##_p name ); \
struct node_type##_s* name##_SearchIndex( name##_p name, int index ); \
int name##_RemoveIndex( name##_p name, int index );

```

Definition at line 46 of file ap\_queue\_template.h.

### 5.23.1.2 #define AP\_QUEUE\_GENERIC\_DECL\_TEMPLATE(name, func\_name, return\_type, search\_type) return\_type name##\_##\_func\_name(name##\_p name, const search\_type key);

Definition at line 66 of file ap\_queue\_template.h.

### 5.23.1.3 #define AP\_QUEUE\_REMOVE\_TEMPLATE(name, func\_name, node\_type, search\_type, search\_expression)

**Value:**

```

int name##_##_func_name( name##_p name, const search_type key ) \
{ \
    int err_code = MC_ERR_NOT_FOUND; \
    struct listNode_s* parsenode; \
    struct node_type##_s* node; \
    node = NULL; \
    \
    MUTEX_LOCK( name->lock ); \
    if ( name->list->listhead == NULL ) { \
        MUTEX_UNLOCK( name->lock ); \
        return MC_ERR_NOT_FOUND; \
    } \
    for( \
        parsenode = (listNode_t*)name->list->listhead; \
        parsenode->next != NULL; \
        parsenode = (listNode_t*)parsenode->next \
    ) \
    { \
        node = (node_type##_t*)parsenode->node_data; \
        if ( search_expression ) { \
            break; \
            err_code = MC_SUCCESS; \
        } \
    } \
    MUTEX_UNLOCK( name->lock ); \
    return err_code; \
}

```

Definition at line 224 of file ap\_queue\_template.h.

### 5.23.1.4 #define AP\_QUEUE\_SEARCH\_TEMPLATE(name, func\_name, node\_type, search\_type, search\_expression)

**Value:**

```

struct node_type##_s* name##_##_func_name( name##_p name, const search_type key ) \
{ \
    listNode_t* parsenode; \
    struct node_type##_s* node; \
}

```

```

struct node_type##_s* ret = NULL; \
node = NULL; \
\
MUTEX_LOCK(name->lock); \
if (name->list->listhead == NULL) { \
    MUTEX_UNLOCK(name->lock); \
    return NULL; \
} \
for( \
    parsenode = (listNode_t*)name->list->listhead; \
    parsenode != NULL; \
    parsenode = (listNode_t*)parsenode->next \
) \
{ \
    node = (node_type##_t*)parsenode->node_data; \
    if (search_expression){ \
        ret = node; \
        break; \
    } \
} \
MUTEX_UNLOCK(name->lock); \
return ret; \
}

```

Definition at line 194 of file ap\_queue\_template.h.

#### 5.23.1.5 #define AP\_QUEUE\_STD\_DEFN\_TEMPLATE(name, node\_type)

Definition at line 69 of file ap\_queue\_template.h.

### 5.23.2 Typedef Documentation

#### 5.23.2.1 typedef AP\_GENERIC\_t\* AP\_GENERIC\_p

Definition at line 43 of file ap\_queue\_template.h.

#### 5.23.2.2 typedef struct AP\_GENERIC\_s AP\_GENERIC\_t

Definition at line 42 of file ap\_queue\_template.h.

### 5.23.3 Function Documentation

#### 5.23.3.1 AP\_GENERIC\_p interpreter\_queue\_CreateRetrieve (struct interpreter\_queue\_s \* queue, ChOptions\_t \* interp\_options)

Referenced by agent\_RunChScriptThread().

## 5.24 /home/dko/Projects/mobilec/src/include/barrier.h File Reference

```
#include "macros.h"
#include "../mc_list/list.h"
#include <stdio.h>
#include <stdlib.h>
```

### Data Structures

- struct [barrier\\_node\\_s](#)
- struct [barrier\\_queue\\_s](#)

### Typedefs

- typedef struct [barrier\\_node\\_s](#) [barrier\\_node\\_t](#)
- typedef [barrier\\_node\\_t](#) \* [barrier\\_node\\_p](#)
- typedef struct [barrier\\_queue\\_s](#) [barrier\\_queue\\_t](#)
- typedef [barrier\\_queue\\_t](#) \* [barrier\\_queue\\_p](#)

### Functions

- [barrier\\_node\\_p](#) [barrier\\_node\\_Initialize](#) ([int](#) id, [int](#) num\_registered)
- [int](#) [barrier\\_node\\_Destroy](#) ([barrier\\_node\\_p](#) node)
- [int](#) [barrier\\_queue\\_Add](#) ([barrier\\_queue\\_p](#) list, [barrier\\_node\\_p](#) node)
- [int](#) [barrier\\_queue\\_Delete](#) ([int](#) id, [barrier\\_queue\\_p](#) list)
- [int](#) [barrier\\_queue\\_Destroy](#) ([barrier\\_queue\\_p](#) queue)
- [barrier\\_node\\_p](#) [barrier\\_queue\\_Get](#) ([barrier\\_queue\\_p](#) list, [int](#) id)
- [barrier\\_queue\\_p](#) [barrier\\_queue\\_New](#) (void)
- [barrier\\_node\\_p](#) [barrier\\_queue\\_Pop](#) ([barrier\\_queue\\_p](#) queue)

#### 5.24.1 Typedef Documentation

##### 5.24.1.1 typedef [barrier\\_node\\_t](#)\* [barrier\\_node\\_p](#)

Definition at line 49 of file barrier.h.

##### 5.24.1.2 typedef struct [barrier\\_node\\_s](#) [barrier\\_node\\_t](#)

##### 5.24.1.3 typedef [barrier\\_queue\\_t](#)\* [barrier\\_queue\\_p](#)

Definition at line 57 of file barrier.h.

#### 5.24.1.4 typedef struct barrier\_queue\_s barrier\_queue\_t

### 5.24.2 Function Documentation

#### 5.24.2.1 int barrier\_node\_Destroy (barrier\_node\_p node)

Definition at line 70 of file barrier.c.

References barrier\_node\_s::cond, COND\_DESTROY, barrier\_node\_s::lock, MC\_SUCCESS, and MUTEX\_DESTROY.

Referenced by barrier\_queue\_Delete(), and barrier\_queue\_Destroy().

#### 5.24.2.2 barrier\_node\_p barrier\_node\_Initialize (int id, int num\_registered)

Definition at line 45 of file barrier.c.

References CHECK\_NULL, barrier\_node\_s::cond, COND\_INIT, COND\_T, barrier\_node\_s::id, barrier\_node\_s::lock, MUTEX\_INIT, MUTEX\_T, node, barrier\_node\_s::num\_registered, and barrier\_node\_s::num\_waiting.

Referenced by MC\_BarrierInit().

#### 5.24.2.3 int barrier\_queue\_Add (barrier\_queue\_p list, barrier\_node\_p node)

Definition at line 87 of file barrier.c.

References DATA, barrier\_node\_s::id, barrier\_queue\_s::list, ListAdd(), list\_s::listhead, barrier\_queue\_s::lock, MC\_SUCCESS, MC\_WARN\_DUPLICATE, listNode\_s::next, listNode\_s::node\_data, RWLOCK\_WRLock, RWLOCK\_WRunLock, and barrier\_queue\_s::size.

Referenced by MC\_BarrierInit().

#### 5.24.2.4 int barrier\_queue\_Delete (int id, barrier\_queue\_p list)

Definition at line 111 of file barrier.c.

References barrier\_node\_Destroy(), barrier\_node\_s::id, barrier\_queue\_s::list, ListDelete(), ListSearch(), barrier\_queue\_s::lock, MC\_ERR\_NOT\_FOUND, MC\_SUCCESS, RWLOCK\_WRLock, RWLOCK\_WRunLock, barrier\_queue\_s::size, and list\_s::size.

Referenced by MC\_BarrierDelete().

#### 5.24.2.5 int barrier\_queue\_Destroy (barrier\_queue\_p queue)

Definition at line 131 of file barrier.c.

References barrier\_node\_Destroy(), barrier\_queue\_Pop(), barrier\_queue\_s::list, ListTerminate(), barrier\_queue\_s::lock, MC\_SUCCESS, node, and RWLOCK\_DESTROY.

Referenced by mc\_platform\_Destroy().

#### 5.24.2.6 barrier\_node\_p barrier\_queue\_Get (barrier\_queue\_p list, int id)

Definition at line 145 of file barrier.c.

References `barrier_queue_s::list`, `list_s::listhead`, `barrier_queue_s::lock`, `listNode_s::next`, `listNode_s::node_data`, `RWLOCK_RDLOCK`, and `RWLOCK_RDUNLOCK`.

Referenced by `MC_Barrier()`, and `MC_BarrierInit()`.

#### 5.24.2.7 `barrier_queue_p barrier_queue_New (void)`

Definition at line 162 of file `barrier.c`.

References `CHECK_NULL`, `barrier_queue_s::list`, `ListInitialize()`, `barrier_queue_s::lock`, `RWLOCK_INIT`, and `RWLOCK_T`.

Referenced by `mc_platform_Initialize()`.

#### 5.24.2.8 `barrier_node_p barrier_queue_Pop (barrier_queue_p queue)`

Definition at line 176 of file `barrier.c`.

References `barrier_queue_s::list`, `ListPop()`, and `node`.

Referenced by `barrier_queue_Destroy()`.

## 5.25 /home/dko/Projects/mobilec/src/include/cmd\_prompt.h File Reference

```
#include "macros.h"
```

### Data Structures

- struct [cmd\\_prompt\\_s](#)
- struct [command\\_s](#)

### Defines

- `#define` [CMDLINE\\_SIZE](#) 80

### Typedefs

- typedef struct [cmd\\_prompt\\_s](#) [cmd\\_prompt\\_t](#)
- typedef [cmd\\_prompt\\_t](#) \* [cmd\\_prompt\\_p](#)
- typedef struct [command\\_s](#) [command\\_t](#)
- typedef [command\\_t](#) \* [command\\_p](#)

### Functions

- [int](#) [cmd\\_prompt\\_Destroy](#) ([cmd\\_prompt\\_p](#) cmd\_prompt)
- [cmd\\_prompt\\_p](#) [cmd\\_prompt\\_Initialize](#) (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#))
- void [cmd\\_prompt\\_Start](#) (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#))
- [int](#) [split\\_string](#) (char \*\*\*args, const char \*buf)
- [int](#) [process\\_command](#) ([command\\_t](#) \*cmd)
- void \* [cmd\\_prompt\\_Thread](#) (void \*arg)
- [int](#) [exec\\_command](#) ([command\\_t](#) cmd, struct [mc\\_platform\\_s](#) \*global)
- [int](#) [dealloc\\_command](#) ([command\\_t](#) \*cmd)

#### 5.25.1 Define Documentation

##### 5.25.1.1 `#define` [CMDLINE\\_SIZE](#) 80

Definition at line 37 of file [cmd\\_prompt.h](#).

#### 5.25.2 Typedef Documentation

##### 5.25.2.1 typedef [cmd\\_prompt\\_t](#)\* [cmd\\_prompt\\_p](#)

Definition at line 44 of file [cmd\\_prompt.h](#).



**5.25.2.2** `typedef struct cmd_prompt_s cmd_prompt_t`

**5.25.2.3** `typedef command_t* command_p`

Definition at line 51 of file cmd\_prompt.h.

**5.25.2.4** `typedef struct command_s command_t`

### 5.25.3 Function Documentation

**5.25.3.1** `int cmd_prompt_Destroy (cmd_prompt_p cmd_prompt)`

Definition at line 129 of file cmd\_prompt.c.

References MC\_SUCCESS.

Referenced by mc\_platform\_Destroy().

**5.25.3.2** `cmd_prompt_p cmd_prompt_Initialize (struct mc_platform_s * mc_platform)`

**5.25.3.3** `void cmd_prompt_Start (struct mc_platform_s * mc_platform)`

**5.25.3.4** `void* cmd_prompt_Thread (void * arg)`

Definition at line 168 of file cmd\_prompt.c.

References command\_s::args, buf, dealloc\_command(), exec\_command(), command\_s::index, mc\_platform, command\_s::num\_args, process\_command(), and split\_string().

Referenced by cmd\_prompt\_Start().

**5.25.3.5** `int dealloc_command (command_t * cmd)`

Definition at line 317 of file cmd\_prompt.c.

References command\_s::args, and command\_s::num\_args.

Referenced by cmd\_prompt\_Thread().

**5.25.3.6** `int exec_command (command_t cmd, struct mc_platform_s * global)`

**5.25.3.7** `int process_command (command_t * cmd)`

Definition at line 283 of file cmd\_prompt.c.

References command\_s::args, command\_cmds, command\_s::index, and command\_s::num\_args.

Referenced by cmd\_prompt\_Thread().

**5.25.3.8** `int split_string (char *** args, const char * buf)`

Definition at line 235 of file cmd\_prompt.c.

References int.

Referenced by `cmd_prompt_Thread()`.

## 5.26 /home/dko/Projects/mobilec/src/include/commands.h File Reference

```
#include "macros.h"
#include "mc_platform.h"
#include "commands.x.h"
```

### Defines

- #define [COMMAND](#)(name, cmd, desc) [int](#) handler\_##name (void \*arg, [mc\\_platform\\_p](#) global);
- #define [COMMAND](#)(name, cmd, description) [COMMAND\\_##name](#) ,
- #define [COMMAND](#)(name, cmd, description) &handler\_##name ,
- #define [COMMAND](#)(name, cmd, description) description ,
- #define [COMMAND](#)(name, cmd, description) cmd ,

### Typedefs

- typedef [int](#)(\* [cmd\\_handler\\_t](#))(void \*, [mc\\_platform\\_p](#) global)

### Enumerations

- enum [command\\_indices\\_e](#)

### Functions

- [COMMAND](#) (QUIT,"quit","This command ends the MobileC application.") [COMMAND](#)(HELP
- This command displays [help](#) for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList [COMMAND](#) (PRINTLIST\_-MESSAGE,"printlist\_message","This command prints all the list entries on the linked list:\n \ message\_queue") [COMMAND](#)(PRINTLIST\_AGENTS
- This command displays [help](#) for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n [COMMAND](#) (FLUSH\_AGENTS,"flush\_agents","This command flushes all of the agents on the system which are still\n\ on the agent list.\n") [COMMAND](#)(COMPOSE\_SEND

### Variables

- [cmd\\_handler\\_t](#) [cmd\\_handlers](#) []
- char \* [command\\_descriptions](#) []
- char \* [command\\_cmds](#) []

#### 5.26.1 Define Documentation

##### 5.26.1.1 #define [COMMAND](#)(name, cmd, description) cmd ,

Definition at line 48 of file commands.h.

### 5.26.1.2 **#define COMMAND(name, cmd, description) description ,**

Definition at line 48 of file commands.h.

### 5.26.1.3 **#define COMMAND(name, cmd, description) &handler\_##name ,**

Definition at line 48 of file commands.h.

### 5.26.1.4 **#define COMMAND(name, cmd, description) COMMAND\_##name ,**

Definition at line 48 of file commands.h.

### 5.26.1.5 **#define COMMAND(name, cmd, desc) int handler\_##name (void \*arg, mc\_platform\_p global);**

Definition at line 48 of file commands.h.

## 5.26.2 **Typedef Documentation**

### 5.26.2.1 **typedef int(\* cmd\_handler\_t)(void \*, mc\_platform\_p global)**

Definition at line 45 of file commands.h.

## 5.26.3 **Enumeration Type Documentation**

### 5.26.3.1 **enum command\_indices\_e**

Definition at line 47 of file commands.h.

## 5.26.4 **Function Documentation**

- 5.26.4.1** This command displays help for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n command\_indices\_e::COMMAND (FLUSH\_AGENTS, "flush\_agents", "This command flushes all of the agents on the system which are still\n\n on the agent list.\n")

### **Type Constraints**

- 5.26.4.2** This command displays help for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList command\_indices\_e::COMMAND (PRINTLIST\_MESSAGE, "printlist\_message", "This command prints all the list entries on the linked list:\n\n message\_queue")

### **Type Constraints**

**5.26.4.3** `command_indices_e::COMMAND (QUIT, "quit", "This command ends the MobileC application.")`

## **5.26.5 Variable Documentation**

**5.26.5.1** `cmd_handler_t cmd_handlers[ ]`

Definition at line 54 of file commands.h.

Referenced by `exec_command()`.

**5.26.5.2** `char* command_cmds[ ]`

Definition at line 68 of file commands.h.

Referenced by `handler_HELP()`, and `process_command()`.

**5.26.5.3** `char* command_descriptions[ ]`

Definition at line 61 of file commands.h.

Referenced by `handler_COMPOSE_SEND()`, `handler_HELP()`, and `handler_SEND()`.

## 5.27 /home/dko/Projects/mobilec/src/include/commands.x.h File Reference

### Functions

- [COMMAND](#) (QUIT,"quit","This command ends the MobileC application.") [COMMAND](#)(HELP
- This command displays [help](#) for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList [COMMAND](#) (PRINTLIST\_MESSAGE,"printlist\_message","This command prints all the list entries on the linked list:\n \ message\_queue") [COMMAND](#)(PRINTLIST\_AGENTS
- This command displays [help](#) for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n [COMMAND](#) (FLUSH\_AGENTS,"flush\_agents","This command flushes all of the agents on the system which are still\n\ on the agent list.\n") [COMMAND](#)(COMPOSE\_SEND

### Variables

- [help](#)
- This command displays [help](#) for any command Usage [is](#)
- This command displays [help](#) for any command Usage [send](#)
- This command displays [help](#) for any command Usage This command sends an agent XML file to a host Usage [print\\_connectlist](#)
- This command displays [help](#) for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList [printlist\\_agents](#)
- This command displays [help](#) for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n [compose\\_send](#)

### 5.27.1 Function Documentation

- 5.27.1.1 This command displays help for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n [COMMAND](#) (FLUSH\_AGENTS, "flush\_agents", "This command flushes all of the agents on the system which are still\n\ on the agent list.\n")**

#### Type Constraints

- 5.27.1.2 This command displays help for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList [COMMAND](#) (PRINTLIST\_MESSAGE, "printlist\_message", "This command prints all the list entries on the linked list:\n \ message\_queue")**

#### Type Constraints

**5.27.1.3 COMMAND (QUIT, "quit", "This command ends the MobileC application.")**

## **5.27.2 Variable Documentation**

**5.27.2.1 This command displays help for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n compose\_send**

Definition at line 62 of file commands.x.h.

### **5.27.2.2 help**

Definition at line 40 of file commands.x.h.

**5.27.2.3 This command displays help for any command Usage This command sends an agent XML file to a host Usage is**

Definition at line 40 of file commands.x.h.

**5.27.2.4 This command displays help for any command Usage This command sends an agent XML file to a host Usage print\_connectlist**

Definition at line 40 of file commands.x.h.

**5.27.2.5 This command displays help for any command Usage This command sends an agent XML file to a host Usage This command prints the list entries in the connectList printlist\_agents**

Definition at line 55 of file commands.x.h.

**5.27.2.6 This command displays help for any command Usage send**

Definition at line 40 of file commands.x.h.

Referenced by `extract_nonce_from_MA()`, `initiate_migration_process()`, `MC_End()`, `message_send_Thread()`, `mtp_http_InitializeFromConnection()`, `reply_migration_process()`, and `send_AES_en_MA()`.

## 5.28 /home/dko/Projects/mobilec/src/include/connection.h File Reference

```
#include <sys/types.h>
#include <netinet/in.h>
```

### Data Structures

- struct [connection\\_s](#)

### Typedefs

- typedef struct [connection\\_s](#) [connection\\_t](#)
- typedef [connection\\_t](#) \* [connection\\_p](#)

### Functions

- [connection\\_p](#) [connection\\_New](#) (void)
- int [connection\\_Destroy](#) ([connection\\_p](#) connection)
- [connection\\_p](#) [connection\\_Copy](#) ([connection\\_p](#) connection)
- void [connection\\_Close](#) ([connection\\_p](#) connection)
- void [connection\\_SetHosts](#) ([connection\\_p](#) connection, char \*remote\_host, char \*local\_host)
- int [connection\\_CreateMessage](#) ([connection\\_p](#) connection)
- int [connection\\_SendtoConnection](#) ([connection\\_p](#) connection, char \*send\_buffer)
- int [connection\\_RecvFromConnection](#) ([connection\\_p](#) connection, struct [mc\\_platform\\_s](#) \*global)

### 5.28.1 Typedef Documentation

#### 5.28.1.1 typedef [connection\\_t](#)\* [connection\\_p](#)

Definition at line 66 of file [connection.h](#).

#### 5.28.1.2 typedef struct [connection\\_s](#) [connection\\_t](#)

### 5.28.2 Function Documentation

#### 5.28.2.1 void [connection\\_Close](#) ([connection\\_p](#) *connection*)

#### 5.28.2.2 [connection\\_p](#) [connection\\_Copy](#) ([connection\\_p](#) *connection*)

Definition at line 80 of file [connection.c](#).

References [connection\\_s::addr](#), [connection\\_s::clientfd](#), [connection\\_s::connect\\_id](#), [connection\\_New\(\)](#), [connection\\_s::remote\\_hostname](#), [connection\\_s::serverfd](#), and [strdup](#).



**5.28.2.3 int connection\_CreateMessage (connection\_p *connection*)****5.28.2.4 int connection\_Destroy (connection\_p *connection*)**

Definition at line 48 of file connection.c.

References connection\_s::clientfd, MC\_SUCCESS, and connection\_s::remote\_hostname.

Referenced by acc\_connection\_Thread().

**5.28.2.5 connection\_p connection\_New (void)**

Definition at line 69 of file connection.c.

Referenced by connection\_Copy(), and listen\_Thread().

**5.28.2.6 int connection\_RecvFromConnection (connection\_p *connection*, struct mc\_platform\_s \* *global*)****5.28.2.7 int connection\_SendtoConnection (connection\_p *connection*, char \* *send\_buffer*)****5.28.2.8 void connection\_SetHosts (connection\_p *connection*, char \* *remote\_host*, char \* *local\_host*)**

## 5.29 /home/dko/Projects/mobilec/src/include/data\_structures.h File Reference

```
#include "ap_queue_template.h"
#include "../mc_list/list.h"
#include "connection.h"
#include "fipa_acl.h"
#include "agent_mailbox.h"
#include "message.h"
#include "interpreter_variable_data.h"
```

### Functions

- [AP\\_QUEUE\\_DECL\\_TEMPLATE](#) (connection\_queue, connection) [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#)(connection\_queue
- [connection\\_t](#) [int](#) [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#) (connection\_queue, [Remove](#), [int](#), [int](#)) [int](#) [connection\\_queue\\_Print](#)(connection\_queue\_p clist)
- [AP\\_QUEUE\\_DECL\\_TEMPLATE](#) (message\_queue, message) [int](#) [message\\_queue\\_Print](#)(message\_queue\_p queue)
- [AP\\_QUEUE\\_DECL\\_TEMPLATE](#) (agent\_queue, agent) [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#)(agent\_queue
- struct [agent\\_s](#) [int](#) [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#) (agent\_queue, [SearchName](#), struct [agent\\_s](#) \*, char \*) [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#)(agent\_queue
- struct [agent\\_s](#) [int](#) [int](#) [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#) (agent\_queue, [RemoveName](#), [int](#), char \*) [int](#) [agent\\_queue\\_Print](#)(agent\_queue\_p queue)
- [int](#) [agent\\_queue\\_Flush](#) (agent\_queue\_p queue)
- [AP\\_QUEUE\\_DECL\\_TEMPLATE](#) (agent\_variable\_list, interpreter\_variable\_data) [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#)(agent\_variable\_list
- [interpreter\\_variable\\_data\\_t](#) char \* [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#) (agent\_variable\_list, [Remove](#), [int](#), char \*) [AP\\_QUEUE\\_DECL\\_TEMPLATE](#)(mail\_queue
- [interpreter\\_variable\\_data\\_t](#) char \*fipa\_acl\_message [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#) (mail\_queue, [SearchReceivers](#), [fipa\\_acl\\_message\\_p](#), char \*) [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#)(mail\_queue
- [interpreter\\_variable\\_data\\_t](#) char \*fipa\_acl\_message char \* [AP\\_QUEUE\\_DECL\\_TEMPLATE](#) (mailbox\_queue, agent\_mailbox) [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#)(mailbox\_queue
- [interpreter\\_variable\\_data\\_t](#) char \*fipa\_acl\_message char char \* [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#) (mailbox\_queue, [SearchID](#), [agent\\_mailbox\\_p](#), [int](#)) [AP\\_QUEUE\\_GENERIC\\_DECL\\_TEMPLATE](#)(mailbox\_queue
- [interpreter\\_variable\\_data\\_t](#) char \*fipa\_acl\_message char char [int](#) [AP\\_QUEUE\\_DECL\\_TEMPLATE](#) (interpreter\_queue, [AP\\_GENERIC](#)) [AP\\_GENERIC\\_p](#) [AP\\_GENERIC\\_Copy](#)([AP\\_GENERIC\\_p](#) [AP\\_GENERIC](#))
- [int](#) [AP\\_GENERIC\\_Destroy](#) ([AP\\_GENERIC\\_t](#) \*blah)

### Variables

- [Search](#)
- struct [agent\\_s](#) [int](#) [Remove](#)

- struct [agent\\_s](#) int int
- [interpreter\\_variable\\_data\\_t](#) char \*fipa\_acl\_message [RemoveName](#)
- [interpreter\\_variable\\_data\\_t](#) char \*fipa\_acl\_message char \* [SearchName](#)
- [interpreter\\_variable\\_data\\_t](#) char \*fipa\_acl\_message char [agent\\_mailbox\\_p](#)
- [interpreter\\_variable\\_data\\_t](#) char \*fipa\_acl\_message char char \* [RemoveID](#)

## 5.29.1 Function Documentation

### 5.29.1.1 int agent\_queue\_Flush (agent\_queue\_p *queue*)

Definition at line 197 of file data\_structures.c.

References [agent\\_s::agent\\_status](#), [agent\\_s::id](#), [ListSearch\(\)](#), [MUTEX\\_LOCK](#), [MUTEX\\_UNLOCK](#), [agent\\_s::name](#), and [node](#).

Referenced by [handler\\_FLUSH\\_AGENTS\(\)](#).

### 5.29.1.2 int AP\_GENERIC\_Destroy (AP\_GENERIC\_t \* *blah*)

Definition at line 291 of file data\_structures.c.

### 5.29.1.3 interpreter\_variable\_data\_t char\* fipa\_acl\_message char char int AP\_QUEUE\_DECL\_TEMPLATE (interpreter\_queue, AP\_GENERIC)

#### Type Constraints

### 5.29.1.4 interpreter\_variable\_data\_t char\* fipa\_acl\_message char\* AP\_QUEUE\_DECL\_TEMPLATE (mailbox\_queue, agent\_mailbox)

#### Type Constraints

### 5.29.1.5 AP\_QUEUE\_DECL\_TEMPLATE (agent\_variable\_list, interpreter\_variable\_data)

### 5.29.1.6 AP\_QUEUE\_DECL\_TEMPLATE (agent\_queue, agent)

### 5.29.1.7 AP\_QUEUE\_DECL\_TEMPLATE (message\_queue, message)

### 5.29.1.8 AP\_QUEUE\_DECL\_TEMPLATE (connection\_queue, connection)

### 5.29.1.9 interpreter\_variable\_data\_t char\* fipa\_acl\_message char char\* AP\_QUEUE\_ GENERIC\_DECL\_TEMPLATE (mailbox\_queue, SearchID, agent\_mailbox\_p, int)

#### Type Constraints

**5.29.1.10** `interpreter_variable_data_t char* fipa_acl_message AP_QUEUE_GENERIC_DECL_TEMPLATE (mail_queue, SearchReceivers, fipa_acl_message_p, char *)`

**5.29.1.11** `interpreter_variable_data_t char* AP_QUEUE_GENERIC_DECL_TEMPLATE (agent_variable_list, Remove, int, char *)`

#### Type Constraints

**5.29.1.12** `struct agent_s int int AP_QUEUE_GENERIC_DECL_TEMPLATE (agent_queue, RemoveName, int, char *)`

#### Type Constraints

**5.29.1.13** `struct agent_s int AP_QUEUE_GENERIC_DECL_TEMPLATE (agent_queue, SearchName, struct agent_s *, char *)` **[read]**

#### Type Constraints

**5.29.1.14** `connection_t int AP_QUEUE_GENERIC_DECL_TEMPLATE (connection_queue, Remove, int, int)`

#### Type Constraints

## 5.29.2 Variable Documentation

**5.29.2.1** `interpreter_variable_data_t char* fipa_acl_message char agent_mailbox_p`

Definition at line 164 of file data\_structures.h.

**5.29.2.2** `interpreter_variable_data_t char *fipa_acl_message char char int`

Definition at line 101 of file data\_structures.h.

Referenced by `aes_en_de()`, `default_callback()`, `main()`, `MC_SetDefaultAgentStatus_chdl()`, `mpi_lsb()`, `mpi_read_string()`, `net_accept()`, `rece_de_msg()`, `remove_nonce_from_MA()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_verify()`, `send_AES_en_MA()`, `sha4_finish()`, `sha4_update()`, `split_string()`, and `ssl_parse_certificate()`.

**5.29.2.3** `struct agent_s int Remove`

Definition at line 101 of file data\_structures.h.

**5.29.2.4 interpreter\_variable\_data\_t char\* fipa\_acl\_message char char\* RemoveID**

Definition at line 178 of file data\_structures.h.

**5.29.2.5 interpreter\_variable\_data\_t char\* fipa\_acl\_message RemoveName**

Definition at line 151 of file data\_structures.h.

**5.29.2.6 Search**

Definition at line 53 of file data\_structures.h.

**5.29.2.7 interpreter\_variable\_data\_t char\* fipa\_acl\_message char\* SearchName**

Definition at line 164 of file data\_structures.h.

## 5.30 /home/dko/Projects/mobilec/src/include/df.h File Reference

```
#include "../mc_list/list.h"
#include "../include/macros.h"
#include "df_request.x.h"
```

### Defines

- #define [REQUEST](#)(name, string, description) REQUEST\_##name,
- #define [REQUEST](#)(name, string, description) int request\_handler\_##name (struct [mc\\_platform\\_s](#) \*global, void\* data);

### Enumerations

- enum [df\\_request\\_list\\_index\\_e](#)
- enum [service\\_types\\_e](#) {  
[ZERO](#), [MISC](#), [INSERT](#), [SOME](#),  
[TYPES](#), [HERE](#) }

### Functions

- [REQUEST](#) (SUBSCRIBE,"subscribe","Subscribe to a directory facilitator") [REQUEST](#)(REGISTER
- Register services with the directory facilitator [REQUEST](#) (DEREGISTER,"deregister","Deregisters mobile agent services from a directory facilitator.") [REQUEST](#)(SEARCH
- [STRUCT](#) (df\_request\_list\_node, MUTEX\_T \*[lock](#);COND\_T \*[cond](#);const char \*command;void \*data;int data\_size;) [STRUCT](#)(df\_request\_list
- [STRUCT](#) (df\_search\_results, char \*\*agent\_names;char \*\*service\_names;int \*agent\_ids;int num\_results;) [STRUCT](#)(df\_request\_search
- [STRUCT](#) (df\_node, MUTEX\_T \*[lock](#);int agent\_id;char \*agent\_name;int num\_services;char \*\*service\_names;enum [service\\_types\\_e](#) service\_types;) [STRUCT](#)(df
- [STRUCT](#) (df\_deregister, int agent\_id;char \*service\_name;) int df\_Add(struct df\_s \*df
- int df\_AddRequest (struct df\_s \*df, struct df\_request\_list\_node\_s \*[node](#))
- int df\_Destroy (df\_p df)
- df\_p df\_Initialize (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#))
- int df\_ProcessRequest (struct [mc\\_platform\\_s](#) \*global)
- int df\_SearchForService (df\_p df, const char \*searchstring, char \*\*\*agent\_names, char \*\*\*service\_names, int \*\*agent\_ids, int \*num\_entries)
- void df\_Start (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#))
- int df\_node\_Destroy (df\_node\_p df\_node)
- int df\_request\_list\_node\_Destroy (df\_request\_list\_node\_p [node](#))
- df\_request\_list\_node\_p df\_request\_list\_node\_New (void)
- int df\_request\_list\_Destroy (df\_request\_list\_p df\_request\_list)
- df\_request\_list\_p df\_request\_list\_New (void)
- df\_request\_list\_node\_p df\_request\_list\_Pop (df\_request\_list\_p requests)
- df\_request\_search\_p df\_request\_search\_New (void)
- int df\_request\_search\_Destroy (df\_request\_search\_p [node](#))
- void \* df\_Thread (void \*arg)

## Variables

- `MUTEX_T * lock`
- `COND_T * cond`
- `int size`
- `list_p request_list`
- `char * search_string`
- `df_search_results_p search_results`
- `struct mc_platform_s * mc_platform`
- `list_p service_list`
- `int num_entries`
- `int waiting`
- `MUTEX_T * waiting_lock`
- `COND_T * waiting_cond`
- `THREAD_T thread`
- `struct df_node_s * node`

### 5.30.1 Define Documentation

#### 5.30.1.1 `#define REQUEST(name, string, description) int request_handler_##name (struct mc_platform_s *global, void* data);`

Definition at line 193 of file df.h.

#### 5.30.1.2 `#define REQUEST(name, string, description) REQUEST_##name,`

Definition at line 193 of file df.h.

### 5.30.2 Enumeration Type Documentation

#### 5.30.2.1 `enum df_request_list_index_e`

Definition at line 46 of file df.h.

#### 5.30.2.2 `enum service_types_e`

Enumerator:

***ZERO***  
***MISC***  
***INSERT***  
***SOME***  
***TYPES***  
***HERE***

Definition at line 53 of file df.h.

### 5.30.3 Function Documentation

#### 5.30.3.1 `int df_AddRequest (struct df_s * df, struct df_request_list_node_s * node)`

Definition at line 66 of file df.c.

References ListAdd(), and SIGNAL.

Referenced by MC\_DeregisterService(), MC\_RegisterService(), and MC\_SearchForService().

#### 5.30.3.2 `int df_Destroy (df_p df)`

Definition at line 83 of file df.c.

References COND\_DESTROY, df\_node\_Destroy(), df\_request\_list\_Destroy(), ListPop(), ListTerminate(), MC\_SUCCESS, MUTEX\_DESTROY, and MUTEX\_LOCK.

Referenced by mc\_platform\_Destroy().

#### 5.30.3.3 `df_p df_Initialize (struct mc_platform_s * mc_platform)`

#### 5.30.3.4 `int df_node_Destroy (df_node_p df_node)`

Definition at line 418 of file df.c.

References MC\_SUCCESS, and MUTEX\_LOCK.

Referenced by df\_Destroy().

#### 5.30.3.5 `int df_ProcessRequest (struct mc_platform_s * global)`

Definition at line 134 of file df.c.

References mc\_platform\_s::df, df\_request\_list\_Pop(), MC\_ERR\_EMPTY, and MC\_ERR\_INVALID.

Referenced by df\_Thread().

#### 5.30.3.6 `int df_request_list_Destroy (df_request_list_p df_request_list)`

Definition at line 323 of file df.c.

References COND\_DESTROY, df\_request\_list\_node\_Destroy(), ListPop(), ListTerminate(), MC\_SUCCESS, MUTEX\_DESTROY, and node.

Referenced by df\_Destroy().

#### 5.30.3.7 `df_request_list_p df_request_list_New (void)`

Definition at line 349 of file df.c.

References CHECK\_NULL, COND\_INIT, COND\_T, ListInitialize(), MUTEX\_INIT, and MUTEX\_T.

Referenced by df\_Initialize().



**5.30.3.8 int df\_request\_list\_node\_Destroy (df\_request\_list\_node\_p node)**

Definition at line 294 of file df.c.

References COND\_DESTROY, MC\_SUCCESS, and MUTEX\_DESTROY.

Referenced by df\_request\_list\_Destroy(), and MC\_SearchForService().

**5.30.3.9 df\_request\_list\_node\_p df\_request\_list\_node\_New (void)**

Definition at line 305 of file df.c.

References CHECK\_NULL, COND\_INIT, COND\_T, MUTEX\_INIT, MUTEX\_T, and node.

Referenced by MC\_DeregisterService(), MC\_RegisterService(), and MC\_SearchForService().

**5.30.3.10 df\_request\_list\_node\_p df\_request\_list\_Pop (df\_request\_list\_p requests)**

Definition at line 374 of file df.c.

References ListPop(), MUTEX\_LOCK, MUTEX\_UNLOCK, and node.

Referenced by df\_ProcessRequest().

**5.30.3.11 int df\_request\_search\_Destroy (df\_request\_search\_p node)**

Definition at line 405 of file df.c.

References COND\_DESTROY, MC\_SUCCESS, and MUTEX\_DESTROY.

Referenced by MC\_SearchForService().

**5.30.3.12 df\_request\_search\_p df\_request\_search\_New (void)**

Definition at line 390 of file df.c.

References CHECK\_NULL, COND\_INIT, COND\_T, MUTEX\_INIT, MUTEX\_T, and search.

Referenced by MC\_SearchForService().

**5.30.3.13 int df\_SearchForService (df\_p df, const char \* searchstring, char \*\*\* agent\_names, char \*\*\* service\_names, int \*\* agent\_ids, int \* num\_entries)**

Definition at line 178 of file df.c.

References MC\_ERR\_NOT\_FOUND, MC\_SUCCESS, MUTEX\_LOCK, MUTEX\_UNLOCK, listNode\_s::next, and listNode\_s::node\_data.

Referenced by request\_handler\_SEARCH().

**5.30.3.14 void df\_Start (struct mc\_platform\_s \* mc\_platform)****5.30.3.15 void\* df\_Thread (void \* arg)**

Definition at line 432 of file df.c.

References COND\_BROADCAST, COND\_WAIT, mc\_platform\_s::df, df\_ProcessRequest(), MC\_SUCCESS, MUTEX\_LOCK, MUTEX\_UNLOCK, mc\_platform\_s::quit, mc\_platform\_s::quit\_lock, and THREAD\_EXIT.

Referenced by df\_Start().

**5.30.3.16 Register services with the directory facilitator df\_request\_list\_index\_e::REQUEST (DEREGISTER, "deregister", "Deregisters mobile agent services from a directory facilitator.")**

#### Type Constraints

**5.30.3.17 df\_request\_list\_index\_e::REQUEST (SUBSCRIBE, "subscribe", "Subscribe to a directory facilitator")**

**5.30.3.18 STRUCT (df\_deregister, int agent\_id;char \*service\_name;)**

**5.30.3.19 STRUCT (df\_node, MUTEX\_T \*lock;int agent\_id;char \*agent\_name;int num\_services;char \*\*service\_names;enum service\_types\_e service\_types;)**

**5.30.3.20 STRUCT (df\_search\_results, char \*\*agent\_names;char \*\*service\_names;int \*agent\_ids;int num\_results;)**

**5.30.3.21 STRUCT (df\_request\_list\_node, MUTEX\_T \*lock;COND\_T \*cond;const char \*command;void \*data;int data\_size;)**

### 5.30.4 Variable Documentation

**5.30.4.1 COND\_T \* cond**

Definition at line 74 of file df.h.

**5.30.4.2 MUTEX\_T \* lock**

Definition at line 73 of file df.h.

**5.30.4.3 struct mc\_platform\_s\* mc\_platform**

Definition at line 107 of file df.h.

Referenced by acc\_connection\_Thread(), acc\_MessageHandlerThread(), acc\_Thread(), agent\_RunChScriptThread(), ams\_Thread(), cmd\_prompt\_Thread(), listen\_Thread(), MC\_Initialize(), mc\_platform\_Initialize(), MC\_WaitAgent(), message\_send\_Thread(), and udplisten\_Thread().

**5.30.4.4 struct df\_node\_s\* node**

Definition at line 129 of file df.h.

Referenced by `add_variable()`, `agent_queue_Flush()`, `agent_xml_compose()`, `agent_xml_compose__agent_code()`, `agent_xml_compose__agent_data()`, `agent_xml_compose__create_row_nodes()`, `agent_xml_compose__data()`, `agent_xml_compose__gaf_message()`, `agent_xml_compose__home()`, `agent_xml_compose__message()`, `agent_xml_compose__mobile_agent()`, `agent_xml_compose__name()`, `agent_xml_compose__owner()`, `agent_xml_compose__row()`, `agent_xml_compose__task()`, `agent_xml_compose__tasks()`, `agent_xml_compose__wg_code()`, `AP_QUEUE_SEARCH_TEMPLATE()`, `AP_QUEUE_STD_DEFN_TEMPLATE()`, `barrier_node_Initialize()`, `barrier_queue_Destroy()`, `barrier_queue_Pop()`, `df_request_list_Destroy()`, `df_request_list_node_New()`, `df_request_list_Pop()`, `fipa_envelope_Compose()`, `fipa_envelope_Compose__acl_representation()`, `fipa_envelope_Compose__date()`, `fipa_envelope_Compose__envelope()`, `fipa_envelope_Compose__intended_receiver()`, `fipa_envelope_Compose__params()`, `fipa_envelope_Compose__payload_encoding()`, `fipa_envelope_Compose__to()`, `fipa_envelope_HandleReceived()`, `main()`, `MC_Barrier()`, `MC_BarrierInit()`, `MC_SyncInit()`, `mxml_new()`, `mxmlNewCustom()`, `mxmlNewElement()`, `mxmlNewInteger()`, `mxmlNewOpaque()`, `mxmlNewReal()`, `mxmlNewText()`, `mxmlNewTextf()`, `request_handler_DEREGISTER()`, `scan_file()`, `syncListNew()`, `write_element()`, `xml_get_deep_child()`, and `xml_new_cdata()`.

#### 5.30.4.5 `int num_entries`

Definition at line 113 of file `df.h`.

#### 5.30.4.6 `df_request_list_p request_list`

Definition at line 76 of file `df.h`.

#### 5.30.4.7 `df_search_results_p search_results`

Definition at line 91 of file `df.h`.

#### 5.30.4.8 `char* search_string`

Definition at line 90 of file `df.h`.

#### 5.30.4.9 `list_p service_list`

Definition at line 110 of file `df.h`.

#### 5.30.4.10 `int size`

Definition at line 75 of file `df.h`.

Referenced by `agent_AddPersistentVariable()`, `agent_xml_compose__create_row_nodes()`, `interpreter_variable_data_Initialize()`, `interpreter_variable_data_InitializeFromAgent()`, `MC_AgentReturnDataSize()`, `MC_GetAgentReturnData()`, `MC_SaveData_chdl()`, `MC_WaitAgent()`, `mxml_vsnprintf()`, `receive_AES_en_MA()`, and `send_AES_en_MA()`.

#### 5.30.4.11 `THREAD_T thread`

Definition at line 117 of file `df.h`.

**5.30.4.12 int waiting**

Definition at line 114 of file df.h.

**5.30.4.13 COND\_T\* waiting\_cond**

Definition at line 116 of file df.h.

**5.30.4.14 MUTEX\_T\* waiting\_lock**

Definition at line 115 of file df.h.

## 5.31 /home/dko/Projects/mobilec/src/include/df\_request.x.h File Reference

### Functions

- [REQUEST](#) (SUBSCRIBE,"subscribe","Subscribe to a directory facilitator") [REQUEST](#)(REGISTER
- Register services with the directory facilitator [REQUEST](#) (DEREGISTER,"deregister","Deregisters mobile agent services from a directory facilitator.") [REQUEST](#)(SEARCH

### Variables

- [register](#)
- Register services with the directory facilitator [search](#)

### 5.31.1 Function Documentation

- 5.31.1.1 Register services with the directory facilitator [REQUEST](#) (DEREGISTER, "deregister", "Deregisters mobile agent services from a directory facilitator.")**

#### Type Constraints

- 5.31.1.2 [REQUEST](#) (SUBSCRIBE, "subscribe", "Subscribe to a directory facilitator")**

### 5.31.2 Variable Documentation

#### 5.31.2.1 [register](#)

Definition at line 43 of file df\_request.x.h.

#### 5.31.2.2 Register services with the directory facilitator [search](#)

Definition at line 54 of file df\_request.x.h.

Referenced by [df\\_request\\_search\\_New\(\)](#), [MC\\_SearchForService\(\)](#), and [request\\_handler\\_SEARCH\(\)](#).

## 5.32 /home/dko/Projects/mobilec/src/include/dynstring.h File Reference

### Data Structures

- struct [dynstring\\_s](#)

### Defines

- #define [COMPOSE\\_BLOCKSIZE](#) 1024

### Typedefs

- typedef struct [dynstring\\_s](#) [dynstring\\_t](#)
- typedef [dynstring\\_t](#) \* [dynstring\\_p](#)

### Functions

- [dynstring\\_t](#) \* [dynstring\\_New](#) (void)
- [int](#) [dynstring\\_Append](#) ([dynstring\\_t](#) \*msg, char \*str)
- [int](#) [dynstring\\_Destroy](#) ([dynstring\\_t](#) \*dynstring)

#### 5.32.1 Define Documentation

##### 5.32.1.1 #define COMPOSE\_BLOCKSIZE 1024

Definition at line 8 of file [dynstring.h](#).

Referenced by [dynstring\\_Append\(\)](#), and [dynstring\\_New\(\)](#).

#### 5.32.2 Typedef Documentation

##### 5.32.2.1 typedef [dynstring\\_t](#)\* [dynstring\\_p](#)

Definition at line 14 of file [dynstring.h](#).

##### 5.32.2.2 typedef struct [dynstring\\_s](#) [dynstring\\_t](#)

#### 5.32.3 Function Documentation

##### 5.32.3.1 [int](#) [dynstring\\_Append](#) ([dynstring\\_t](#) \*msg, char \*str)

Definition at line 30 of file [dynstring.c](#).

References [COMPOSE\\_BLOCKSIZE](#), [dynstring\\_s::len](#), [dynstring\\_s::message](#), and [dynstring\\_s::size](#).

Referenced by [fipa\\_acl\\_Compose\(\)](#), [fipa\\_agent\\_identifier\\_Compose\(\)](#), [fipa\\_agent\\_identifier\\_set\\_Compose\(\)](#), [fipa\\_DateTime\\_Compose\(\)](#), [fipa\\_envelope\\_Compose\\_\\_from\(\)](#), [fipa\\_number\\_Compose\(\)](#),

fipa\_performative\_Compose(), fipa\_protocol\_Compose(), fipa\_string\_Compose(), fipa\_url\_Compose(), fipa\_url\_sequence\_Compose(), fipa\_word\_Compose(), message\_send\_Thread(), mtp\_http\_CreateMessage(), and mtp\_http\_InitializeFromConnection().

#### 5.32.3.2 int dynstring\_Destroy (dynstring\_t \* *dynstring*)

Definition at line 56 of file dynstring.c.

References dynstring\_s::message.

Referenced by fipa\_envelope\_Compose\_\_from(), MC\_AclSend(), message\_send\_Thread(), mtp\_http\_CreateMessage(), and mtp\_http\_InitializeFromConnection().

#### 5.32.3.3 dynstring\_t\* dynstring\_New (void)

Definition at line 14 of file dynstring.c.

References COMPOSE\_BLOCKSIZE, dynstring\_s::len, dynstring\_s::message, and dynstring\_s::size.

Referenced by fipa\_acl\_Compose(), fipa\_envelope\_Compose\_\_from(), message\_send\_Thread(), mtp\_http\_CreateMessage(), and mtp\_http\_InitializeFromConnection().

## 5.33 /home/dko/Projects/mobilec/src/include/fipa\_acl.h File Reference

```
#include "dynstring.h"
```

### Data Structures

- struct [fipa\\_acl\\_message\\_s](#)
- struct [fipa\\_message\\_string\\_s](#)
- struct [fipa\\_url\\_sequence\\_s](#)
- struct [fipa\\_agent\\_identifier\\_set\\_s](#)
- struct [fipa\\_agent\\_identifier\\_s](#)
- struct [fipa\\_expression\\_s](#)
- union [fipa\\_expression\\_s::content\\_u](#)
- struct [fipa\\_word\\_s](#)
- struct [fipa\\_string\\_s](#)
- struct [fipa\\_DateTime\\_s](#)
- struct [fipa\\_url\\_s](#)
- struct [fipa\\_number\\_s](#)

### Typedefs

- typedef enum [fipa\\_expression\\_type\\_e](#) [fipa\\_expression\\_type\\_t](#)
- typedef struct [fipa\\_acl\\_message\\_s](#) [fipa\\_acl\\_message\\_t](#)
- typedef [fipa\\_acl\\_message\\_t](#) \* [fipa\\_acl\\_message\\_p](#)
- typedef struct [fipa\\_message\\_string\\_s](#) [fipa\\_message\\_string\\_t](#)
- typedef [fipa\\_message\\_string\\_t](#) \* [fipa\\_message\\_string\\_p](#)
- typedef struct [fipa\\_url\\_sequence\\_s](#) [fipa\\_url\\_sequence\\_t](#)
- typedef [fipa\\_url\\_sequence\\_t](#) \* [fipa\\_url\\_sequence\\_p](#)
- typedef struct [fipa\\_agent\\_identifier\\_set\\_s](#) [fipa\\_agent\\_identifier\\_set\\_t](#)
- typedef [fipa\\_agent\\_identifier\\_set\\_t](#) \* [fipa\\_agent\\_identifier\\_set\\_p](#)
- typedef struct [fipa\\_agent\\_identifier\\_s](#) [fipa\\_agent\\_identifier\\_t](#)
- typedef [fipa\\_agent\\_identifier\\_t](#) \* [fipa\\_agent\\_identifier\\_p](#)
- typedef struct [fipa\\_expression\\_s](#) [fipa\\_expression\\_t](#)
- typedef [fipa\\_expression\\_t](#) \* [fipa\\_expression\\_p](#)
- typedef struct [fipa\\_word\\_s](#) [fipa\\_word\\_t](#)
- typedef [fipa\\_word\\_t](#) \* [fipa\\_word\\_p](#)
- typedef struct [fipa\\_string\\_s](#) [fipa\\_string\\_t](#)
- typedef [fipa\\_string\\_t](#) \* [fipa\\_string\\_p](#)
- typedef struct [fipa\\_DateTime\\_s](#) [fipa\\_DateTime\\_t](#)
- typedef [fipa\\_DateTime\\_t](#) \* [fipa\\_DateTime\\_p](#)
- typedef struct [fipa\\_url\\_s](#) [fipa\\_url\\_t](#)
- typedef [fipa\\_url\\_t](#) \* [fipa\\_url\\_p](#)
- typedef struct [fipa\\_number\\_s](#) [fipa\\_number\\_t](#)
- typedef [fipa\\_number\\_t](#) \* [fipa\\_number\\_p](#)



## Enumerations

- enum `fipa_performative_e` {  
`FIPA_ERROR = -1, FIPA_ZERO, FIPA_ACCEPT_PROPOSAL, FIPA_AGREE,`  
`FIPA_CANCEL, FIPA_CALL_FOR_PROPOSAL, FIPA_CONFIRM, FIPA_DISCONFIRM,`  
`FIPA_FAILURE, FIPA_INFORM, FIPA_INFORM_IF, FIPA_INFORM_REF,`  
`FIPA_NOT_UNDERSTOOD, FIPA_PROPOGATE, FIPA_PROPOSE, FIPA_PROXY,`  
`FIPA_QUERY_IF, FIPA_QUERY_REF, FIPA_REFUSE, FIPA_REJECT_PROPOSAL,`  
`FIPA_REQUEST, FIPA_REQUEST_WHEN, FIPA_REQUEST_WHENEVER, FIPA_-`  
`SUBSCRIBE,`  
`FIPA_ERROR = -1, FIPA_ZERO, FIPA_ACCEPT_PROPOSAL, FIPA_AGREE,`  
`FIPA_CANCEL, FIPA_CALL_FOR_PROPOSAL, FIPA_CONFIRM, FIPA_DISCONFIRM,`  
`FIPA_FAILURE, FIPA_INFORM, FIPA_INFORM_IF, FIPA_INFORM_REF,`  
`FIPA_NOT_UNDERSTOOD, FIPA_PROPOGATE, FIPA_PROPOSE, FIPA_PROXY,`  
`FIPA_QUERY_IF, FIPA_QUERY_REF, FIPA_REFUSE, FIPA_REJECT_PROPOSAL,`  
`FIPA_REQUEST, FIPA_REQUEST_WHEN, FIPA_REQUEST_WHENEVER, FIPA_-`  
`SUBSCRIBE }`
- enum `fipa_protocol_e` {  
`FIPA_PROTOCOL_ERROR = -1, FIPA_PROTOCOL_NONE, FIPA_PROTOCOL_REQUEST,`  
`FIPA_PROTOCOL_QUERY,`  
`FIPA_PROTOCOL_REQUEST_WHEN, FIPA_PROTOCOL_CONTRACT_NET, FIPA_-`  
`PROTOCOL_ITERATED_CONTRACT_NET, FIPA_PROTOCOL_ENGLISH_AUCTION,`  
`FIPA_PROTOCOL_DUTCH_AUCTION, FIPA_PROTOCOL_BROKERING, FIPA_-`  
`PROTOCOL_RECRUITING, FIPA_PROTOCOL_SUBSCRIBE,`  
`FIPA_PROTOCOL_PROPOSE, FIPA_PROTOCOL_END }`
- enum `fipa_expression_type_e` {  
`FIPA_EXPR_ZERO, FIPA_EXPR_WORD, FIPA_EXPR_STRING, FIPA_EXPR_NUMBER,`  
`FIPA_EXPR_DATETIME, FIPA_EXPR_EXPRESSION }`

## Functions

- `int fipa_performative_Compose (dynstring_t *msg, enum fipa_performative_e performative)`
- `int fipa_protocol_Compose (dynstring_t *msg, enum fipa_protocol_e protocol)`
- `fipa_acl_message_t * fipa_acl_message_New (void)`
- `int fipa_acl_message_Destroy (fipa_acl_message_t *message)`
- `fipa_acl_message_t * fipa_acl_message_Copy (fipa_acl_message_t *src)`
- `int fipa_acl_Compose (dynstring_t **msg, fipa_acl_message_t *acl)`
- `fipa_message_string_t * fipa_message_string_New (void)`
- `int fipa_message_string_Destroy (fipa_message_string_t *message)`
- `fipa_message_string_t * fipa_message_string_Copy (fipa_message_string_t *src)`
- `fipa_url_sequence_t * fipa_url_sequence_New (void)`
- `int fipa_url_sequence_Destroy (fipa_url_sequence_t *sequence)`
- `fipa_url_sequence_t * fipa_url_sequence_Copy (fipa_url_sequence_t *src)`
- `int fipa_url_sequence_Compose (dynstring_t *msg, fipa_url_sequence_t *urls)`
- `fipa_agent_identifier_set_t * fipa_agent_identifier_set_New (void)`
- `int fipa_agent_identifier_set_Destroy (fipa_agent_identifier_set_t *idset)`

- [fipa\\_agent\\_identifier\\_set\\_t \\* fipa\\_agent\\_identifier\\_set\\_Copy \(fipa\\_agent\\_identifier\\_set\\_t \\*src\)](#)
- [int fipa\\_agent\\_identifier\\_set\\_Compose \(dynstring\\_t \\*msg, fipa\\_agent\\_identifier\\_set\\_t \\*ids\)](#)
- [fipa\\_agent\\_identifier\\_t \\* fipa\\_agent\\_identifier\\_New \(void\)](#)
- [int fipa\\_agent\\_identifier\\_Destroy \(fipa\\_agent\\_identifier\\_t \\*id\)](#)
- [fipa\\_agent\\_identifier\\_t \\* fipa\\_agent\\_identifier\\_Copy \(fipa\\_agent\\_identifier\\_t \\*src\)](#)
- [int fipa\\_agent\\_identifier\\_Compose \(dynstring\\_t \\*msg, fipa\\_agent\\_identifier\\_t \\*id\)](#)
- [fipa\\_expression\\_t \\* fipa\\_expression\\_New \(void\)](#)
- [int fipa\\_expression\\_Destroy \(fipa\\_expression\\_t \\*expr\)](#)
- [fipa\\_expression\\_t \\* fipa\\_expression\\_Copy \(fipa\\_expression\\_t \\*src\)](#)
- [int fipa\\_expression\\_Compose \(dynstring\\_t \\*msg, fipa\\_expression\\_t \\*expr\)](#)
- [fipa\\_word\\_t \\* fipa\\_word\\_New \(void\)](#)
- [int fipa\\_word\\_Destroy \(fipa\\_word\\_t \\*word\)](#)
- [fipa\\_word\\_t \\* fipa\\_word\\_Copy \(fipa\\_word\\_t \\*src\)](#)
- [int fipa\\_word\\_Compose \(dynstring\\_t \\*msg, fipa\\_word\\_t \\*word\)](#)
- [fipa\\_string\\_t \\* fipa\\_string\\_New \(void\)](#)
- [int fipa\\_string\\_Destroy \(fipa\\_string\\_t \\*str\)](#)
- [fipa\\_string\\_t \\* fipa\\_string\\_Copy \(fipa\\_string\\_t \\*src\)](#)
- [int fipa\\_string\\_Compose \(dynstring\\_t \\*msg, fipa\\_string\\_t \\*string\)](#)
- [fipa\\_DateTime\\_t \\* fipa\\_DateTime\\_New \(void\)](#)
- [int fipa\\_DateTime\\_Destroy \(fipa\\_DateTime\\_t \\*dt\)](#)
- [fipa\\_DateTime\\_t \\* fipa\\_DateTime\\_Copy \(fipa\\_DateTime\\_t \\*src\)](#)
- [int fipa\\_DateTime\\_Compose \(dynstring\\_t \\*msg, fipa\\_DateTime\\_t \\*date\)](#)
- [fipa\\_url\\_t \\* fipa\\_url\\_New \(void\)](#)
- [int fipa\\_url\\_Destroy \(fipa\\_url\\_t \\*url\)](#)
- [fipa\\_url\\_t \\* fipa\\_url\\_Copy \(fipa\\_url\\_t \\*src\)](#)
- [int fipa\\_url\\_Compose \(dynstring\\_t \\*msg, fipa\\_url\\_t \\*url\)](#)
- [fipa\\_number\\_t \\* fipa\\_number\\_New \(void\)](#)
- [int fipa\\_number\\_Destroy \(fipa\\_number\\_t \\*number\)](#)
- [fipa\\_number\\_t \\* fipa\\_number\\_Copy \(fipa\\_number\\_t \\*src\)](#)
- [int fipa\\_number\\_Compose \(dynstring\\_t \\*msg, fipa\\_number\\_t \\*number\)](#)
- [int fipa\\_acl\\_Parse \(struct fipa\\_acl\\_message\\_s \\*acl, fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_message\\_parameter\\_Parse \(struct fipa\\_acl\\_message\\_s \\*acl, fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_protocol\\_type\\_Parse \(enum fipa\\_protocol\\_e \\*protocol, fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_message\\_type\\_Parse \(enum fipa\\_performative\\_e \\*performative, fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_GetAtom \(fipa\\_message\\_string\\_p message, char expected\\_atom\)](#)
- [int fipa\\_word\\_Parse \(fipa\\_word\\_t \\*\\*word, fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_CheckNextToken \(const fipa\\_message\\_string\\_p message, const char \\*token\)](#)
- [int fipa\\_expression\\_Parse \(fipa\\_expression\\_t \\*\\*expression, fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_GetNextWord \(char \\*\\*word, const fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_GetWholeToken \(char \\*\\*word, const fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_datetime\\_Parse \(fipa\\_DateTime\\_p \\*datetime, fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_string\\_Parse \(fipa\\_string\\_p \\*fipa\\_string, fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_agent\\_identifier\\_Parse \(fipa\\_agent\\_identifier\\_p \\*aid, fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_agent\\_identifier\\_set\\_Parse \(fipa\\_agent\\_identifier\\_set\\_p \\*agent\\_ids, fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_url\\_sequence\\_Parse \(fipa\\_url\\_sequence\\_p \\*urls, fipa\\_message\\_string\\_p message\)](#)
- [int fipa\\_url\\_Parse \(fipa\\_url\\_p \\*url, fipa\\_message\\_string\\_p message\)](#)
- [struct fipa\\_acl\\_message\\_s \\* fipa\\_Reply \(struct fipa\\_acl\\_message\\_s \\*acl\)](#)

### 5.33.1 Typedef Documentation

#### 5.33.1.1 typedef fipa\_acl\_message\_t\* fipa\_acl\_message\_p

Definition at line 136 of file fipa\_acl.h.

#### 5.33.1.2 typedef struct fipa\_acl\_message\_s fipa\_acl\_message\_t

#### 5.33.1.3 typedef fipa\_agent\_identifier\_t\* fipa\_agent\_identifier\_p

Definition at line 186 of file fipa\_acl.h.

#### 5.33.1.4 typedef fipa\_agent\_identifier\_set\_t\* fipa\_agent\_identifier\_set\_p

Definition at line 172 of file fipa\_acl.h.

#### 5.33.1.5 typedef struct fipa\_agent\_identifier\_set\_s fipa\_agent\_identifier\_set\_t

#### 5.33.1.6 typedef struct fipa\_agent\_identifier\_s fipa\_agent\_identifier\_t

#### 5.33.1.7 typedef fipa\_DateTime\_t\* fipa\_DateTime\_p

Definition at line 246 of file fipa\_acl.h.

#### 5.33.1.8 typedef struct fipa\_DateTime\_s fipa\_DateTime\_t

#### 5.33.1.9 typedef fipa\_expression\_t\* fipa\_expression\_p

Definition at line 205 of file fipa\_acl.h.

#### 5.33.1.10 typedef struct fipa\_expression\_s fipa\_expression\_t

#### 5.33.1.11 typedef enum fipa\_expression\_type\_e fipa\_expression\_type\_t

#### 5.33.1.12 typedef fipa\_message\_string\_t\* fipa\_message\_string\_p

Definition at line 148 of file fipa\_acl.h.

#### 5.33.1.13 typedef struct fipa\_message\_string\_s fipa\_message\_string\_t

#### 5.33.1.14 typedef fipa\_number\_t\* fipa\_number\_p

Definition at line 268 of file fipa\_acl.h.

#### 5.33.1.15 typedef struct fipa\_number\_s fipa\_number\_t

#### 5.33.1.16 typedef fipa\_string\_t\* fipa\_string\_p

Definition at line 227 of file fipa\_acl.h.

**5.33.1.17** `typedef struct fipa_string_s fipa_string_t`

**5.33.1.18** `typedef fipa_url_t* fipa_url_p`

Definition at line 257 of file fipa\_acl.h.

**5.33.1.19** `typedef fipa_url_sequence_t* fipa_url_sequence_p`

Definition at line 159 of file fipa\_acl.h.

**5.33.1.20** `typedef struct fipa_url_sequence_s fipa_url_sequence_t`

**5.33.1.21** `typedef struct fipa_url_s fipa_url_t`

**5.33.1.22** `typedef fipa_word_t* fipa_word_p`

Definition at line 216 of file fipa\_acl.h.

**5.33.1.23** `typedef struct fipa_word_s fipa_word_t`

## **5.33.2 Enumeration Type Documentation**

**5.33.2.1** `enum fipa_expression_type_e`

Enumerator:

*FIPA\_EXPR\_ZERO*

*FIPA\_EXPR\_WORD*

*FIPA\_EXPR\_STRING*

*FIPA\_EXPR\_NUMBER*

*FIPA\_EXPR\_DATETIME*

*FIPA\_EXPR\_EXPRESSION*

Definition at line 94 of file fipa\_acl.h.

**5.33.2.2** `enum fipa_performative_e`

Enumerator:

*FIPA\_ERROR*

*FIPA\_ZERO*

*FIPA\_ACCEPT\_PROPOSAL*

*FIPA\_AGREE*

*FIPA\_CANCEL*

*FIPA\_CALL\_FOR\_PROPOSAL*

*FIPA\_CONFIRM*

*FIPA\_DISCONFIRM*

*FIPA\_FAILURE*  
*FIPA\_INFORM*  
*FIPA\_INFORM\_IF*  
*FIPA\_INFORM\_REF*  
*FIPA\_NOT\_UNDERSTOOD*  
*FIPA\_PROPOGATE*  
*FIPA\_PROPOSE*  
*FIPA\_PROXY*  
*FIPA\_QUERY\_IF*  
*FIPA\_QUERY\_REF*  
*FIPA\_REFUSE*  
*FIPA\_REJECT\_PROPOSAL*  
*FIPA\_REQUEST*  
*FIPA\_REQUEST\_WHEN*  
*FIPA\_REQUEST\_WHENEVER*  
*FIPA\_SUBSCRIBE*  
*FIPA\_ERROR*  
*FIPA\_ZERO*  
*FIPA\_ACCEPT\_PROPOSAL*  
*FIPA\_AGREE*  
*FIPA\_CANCEL*  
*FIPA\_CALL\_FOR\_PROPOSAL*  
*FIPA\_CONFIRM*  
*FIPA\_DISCONFIRM*  
*FIPA\_FAILURE*  
*FIPA\_INFORM*  
*FIPA\_INFORM\_IF*  
*FIPA\_INFORM\_REF*  
*FIPA\_NOT\_UNDERSTOOD*  
*FIPA\_PROPOGATE*  
*FIPA\_PROPOSE*  
*FIPA\_PROXY*  
*FIPA\_QUERY\_IF*  
*FIPA\_QUERY\_REF*  
*FIPA\_REFUSE*  
*FIPA\_REJECT\_PROPOSAL*  
*FIPA\_REQUEST*  
*FIPA\_REQUEST\_WHEN*  
*FIPA\_REQUEST\_WHENEVER*  
*FIPA\_SUBSCRIBE*

Definition at line 41 of file fipa\_acl.h.

### 5.33.2.3 enum fipa\_protocol\_e

Enumerator:

*FIPA\_PROTOCOL\_ERROR*  
*FIPA\_PROTOCOL\_NONE*  
*FIPA\_PROTOCOL\_REQUEST*  
*FIPA\_PROTOCOL\_QUERY*  
*FIPA\_PROTOCOL\_REQUEST\_WHEN*  
*FIPA\_PROTOCOL\_CONTRACT\_NET*  
*FIPA\_PROTOCOL\_ITERATED\_CONTRACT\_NET*  
*FIPA\_PROTOCOL\_ENGLISH\_AUCTION*  
*FIPA\_PROTOCOL\_DUTCH\_AUCTION*  
*FIPA\_PROTOCOL\_BROKERING*  
*FIPA\_PROTOCOL\_RECRUITING*  
*FIPA\_PROTOCOL\_SUBSCRIBE*  
*FIPA\_PROTOCOL\_PROPOSE*  
*FIPA\_PROTOCOL\_END*

Definition at line 72 of file fipa\_acl.h.

## 5.33.3 Function Documentation

### 5.33.3.1 int fipa\_acl\_Compose (dynstring\_t \*\* msg, fipa\_acl\_message\_t \* acl)

Definition at line 1233 of file fipa\_acl.c.

References fipa\_acl\_message\_s::content, fipa\_acl\_message\_s::conversation\_id, dynstring\_Append(), dynstring\_New(), fipa\_acl\_message\_s::encoding, fipa\_agent\_identifier\_Compose(), fipa\_agent\_identifier\_set\_Compose(), fipa\_DateTime\_Compose(), fipa\_expression\_Compose(), fipa\_performative\_Compose(), fipa\_protocol\_Compose(), FIPA\_PROTOCOL\_NONE, fipa\_string\_Compose(), fipa\_acl\_message\_s::in\_reply\_to, fipa\_acl\_message\_s::language, fipa\_acl\_message\_s::ontology, fipa\_acl\_message\_s::performative, fipa\_acl\_message\_s::protocol, fipa\_acl\_message\_s::receiver, fipa\_acl\_message\_s::reply\_by, fipa\_acl\_message\_s::reply\_to, fipa\_acl\_message\_s::reply\_with, and fipa\_acl\_message\_s::sender.

Referenced by MC\_AclSend().

### 5.33.3.2 fipa\_acl\_message\_t\* fipa\_acl\_message\_Copy (fipa\_acl\_message\_t \* src)

Definition at line 86 of file fipa\_acl.c.

References fipa\_acl\_message\_s::content, fipa\_acl\_message\_s::conversation\_id, fipa\_acl\_message\_s::encoding, fipa\_acl\_message\_New(), fipa\_agent\_identifier\_Copy(), fipa\_agent\_identifier\_set\_Copy(), fipa\_DateTime\_Copy(), fipa\_expression\_Copy(), fipa\_string\_Copy(), fipa\_acl\_message\_s::in\_reply\_to, fipa\_acl\_message\_s::language, fipa\_acl\_message\_s::ontology, fipa\_acl\_message\_s::performative, fipa\_acl\_message\_s::protocol, fipa\_acl\_message\_s::receiver, fipa\_acl\_message\_s::reply\_by, fipa\_acl\_message\_s::reply\_to, fipa\_acl\_message\_s::reply\_with, and fipa\_acl\_message\_s::sender.

Referenced by MC\_AclSend().

**5.33.3.3 int fipa\_acl\_message\_Destroy (fipa\_acl\_message\_t \* message)**

Definition at line 65 of file fipa\_acl.c.

References fipa\_acl\_message\_s::content, fipa\_acl\_message\_s::conversation\_id, fipa\_acl\_message\_s::encoding, fipa\_agent\_identifier\_Destroy(), fipa\_agent\_identifier\_set\_Destroy(), fipa\_DateTime\_Destroy(), fipa\_expression\_Destroy(), fipa\_string\_Destroy(), fipa\_acl\_message\_s::in\_reply\_to, fipa\_acl\_message\_s::language, fipa\_acl\_message\_s::ontology, fipa\_acl\_message\_s::receiver, fipa\_acl\_message\_s::reply\_by, fipa\_acl\_message\_s::reply\_to, fipa\_acl\_message\_s::reply\_with, and fipa\_acl\_message\_s::sender.

Referenced by acc\_connection\_Thread(), and MC\_AclDestroy().

**5.33.3.4 fipa\_acl\_message\_t\* fipa\_acl\_message\_New (void)**

Definition at line 57 of file fipa\_acl.c.

Referenced by acc\_connection\_Thread(), fipa\_acl\_message\_Copy(), fipa\_Reply(), and MC\_AclNew().

**5.33.3.5 int fipa\_acl\_Parse (struct fipa\_acl\_message\_s \* acl, fipa\_message\_string\_p message)****5.33.3.6 int fipa\_agent\_identifier\_Compose (dynstring\_t \* msg, fipa\_agent\_identifier\_t \* id)**

Definition at line 1447 of file fipa\_acl.c.

References fipa\_agent\_identifier\_s::addresses, dynstring\_Append(), fipa\_agent\_identifier\_set\_Compose(), fipa\_url\_sequence\_Compose(), fipa\_agent\_identifier\_s::name, fipa\_agent\_identifier\_set\_s::num, fipa\_url\_sequence\_s::num, and fipa\_agent\_identifier\_s::resolvers.

Referenced by fipa\_acl\_Compose(), and fipa\_agent\_identifier\_set\_Compose().

**5.33.3.7 fipa\_agent\_identifier\_t\* fipa\_agent\_identifier\_Copy (fipa\_agent\_identifier\_t \* src)**

Definition at line 236 of file fipa\_acl.c.

References fipa\_agent\_identifier\_s::addresses, fipa\_agent\_identifier\_New(), fipa\_agent\_identifier\_set\_Copy(), fipa\_url\_sequence\_Copy(), fipa\_agent\_identifier\_s::name, fipa\_agent\_identifier\_s::resolvers, and strdup.

Referenced by fipa\_acl\_message\_Copy(), fipa\_acl\_Param\_Copy(), fipa\_agent\_identifier\_set\_Copy(), and fipa\_Reply().

**5.33.3.8 int fipa\_agent\_identifier\_Destroy (fipa\_agent\_identifier\_t \* id)**

Definition at line 224 of file fipa\_acl.c.

References fipa\_agent\_identifier\_s::addresses, fipa\_agent\_identifier\_set\_Destroy(), fipa\_url\_sequence\_Destroy(), fipa\_agent\_identifier\_s::name, and fipa\_agent\_identifier\_s::resolvers.

Referenced by fipa\_acl\_message\_Destroy(), fipa\_acl\_Param\_Destroy(), fipa\_agent\_identifier\_set\_Destroy(), and MC\_AclSetSender().

**5.33.3.9 fipa\_agent\_identifier\_t\* fipa\_agent\_identifier\_New (void)**

Definition at line 216 of file fipa\_acl.c.

Referenced by `fipa_agent_identifier_Copy()`, `fipa_envelope_ParseAgentIdentifier()`, `MC_AclAddReceiver()`, `MC_AclAddReplyTo()`, and `MC_AclSetSender()`.

#### 5.33.3.10 `int fipa_agent_identifier_Parse (fipa_agent_identifier_p * aid, fipa_message_string_p message)`

Definition at line 1033 of file `fipa_acl.c`.

References `CHECK_NULL`, `fipa_word_s::content`, `fipa_agent_identifier_set_Parse()`, `fipa_GetAtom()`, `fipa_url_sequence_Parse()`, `fipa_word_Destroy()`, `fipa_word_Parse()`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

Referenced by `fipa_agent_identifier_set_Parse()`, and `fipa_message_parameter_Parse()`.

#### 5.33.3.11 `int fipa_agent_identifier_set_Compose (dynstring_t * msg, fipa_agent_identifier_set_t * ids)`

Definition at line 1434 of file `fipa_acl.c`.

References `dynstring_Append()`, `fipa_agent_identifier_Compose()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, and `fipa_agent_identifier_set_s::num`.

Referenced by `fipa_acl_Compose()`, and `fipa_agent_identifier_Compose()`.

#### 5.33.3.12 `fipa_agent_identifier_set_t* fipa_agent_identifier_set_Copy (fipa_agent_identifier_set_t * src)`

Definition at line 195 of file `fipa_acl.c`.

References `fipa_agent_identifier_Copy()`, `fipa_agent_identifier_set_New()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, `fipa_agent_identifier_set_s::num`, and `fipa_agent_identifier_set_s::retain_order`.

Referenced by `fipa_acl_message_Copy()`, `fipa_acl_Param_Copy()`, `fipa_agent_identifier_Copy()`, and `fipa_Reply()`.

#### 5.33.3.13 `int fipa_agent_identifier_set_Destroy (fipa_agent_identifier_set_t * idset)`

Definition at line 183 of file `fipa_acl.c`.

References `fipa_agent_identifier_Destroy()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, and `fipa_agent_identifier_set_s::num`.

Referenced by `fipa_acl_message_Destroy()`, `fipa_acl_Param_Destroy()`, and `fipa_agent_identifier_Destroy()`.

#### 5.33.3.14 `fipa_agent_identifier_set_t* fipa_agent_identifier_set_New (void)`

Definition at line 175 of file `fipa_acl.c`.

Referenced by `fipa_agent_identifier_set_Copy()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleTo()`, `fipa_Reply()`, `MC_AclAddReceiver()`, and `MC_AclAddReplyTo()`.



**5.33.3.15 int fipa\_agent\_identifier\_set\_Parse (fipa\_agent\_identifier\_set\_p \* *agent\_ids*, fipa\_message\_string\_p *message*)**

Definition at line 1192 of file fipa\_acl.c.

References fipa\_word\_s::content, fipa\_agent\_identifier\_Parse(), fipa\_GetAtom(), fipa\_word\_Parse(), MC\_ERR\_PARSE, MC\_SUCCESS, and fipa\_agent\_identifier\_set\_s::retain\_order.

Referenced by fipa\_agent\_identifier\_Parse(), and fipa\_message\_parameter\_Parse().

**5.33.3.16 int fipa\_CheckNextToken (const fipa\_message\_string\_p *message*, const char \* *token*)**

Definition at line 695 of file fipa\_acl.c.

References fipa\_message\_string\_s::parse.

Referenced by fipa\_expression\_Parse().

**5.33.3.17 int fipa\_DateTime\_Compose (dynstring\_t \* *msg*, fipa\_DateTime\_t \* *date*)**

Definition at line 1520 of file fipa\_acl.c.

References buf, fipa\_DateTime\_s::day, dynstring\_Append(), fipa\_DateTime\_s::hour, fipa\_DateTime\_s::millisecond, fipa\_DateTime\_s::minute, fipa\_DateTime\_s::month, fipa\_DateTime\_s::second, fipa\_DateTime\_s::sign, and fipa\_DateTime\_s::year.

Referenced by fipa\_acl\_Compose(), and fipa\_expression\_Compose().

**5.33.3.18 fipa\_DateTime\_t\* fipa\_DateTime\_Copy (fipa\_DateTime\_t \* *src*)**

Definition at line 399 of file fipa\_acl.c.

References fipa\_DateTime\_New().

Referenced by fipa\_acl\_envelope\_Received\_Copy(), fipa\_acl\_message\_Copy(), fipa\_acl\_Param\_Copy(), and fipa\_expression\_Copy().

**5.33.3.19 int fipa\_DateTime\_Destroy (fipa\_DateTime\_t \* *dt*)**

Definition at line 392 of file fipa\_acl.c.

Referenced by fipa\_acl\_envelope\_Received\_Destroy(), fipa\_acl\_message\_Destroy(), fipa\_acl\_Param\_Destroy(), and fipa\_expression\_Destroy().

**5.33.3.20 fipa\_DateTime\_t\* fipa\_DateTime\_New (void)**

Definition at line 384 of file fipa\_acl.c.

Referenced by fipa\_DateTime\_Copy().

**5.33.3.21 int fipa\_datetime\_Parse (fipa\_DateTime\_p \* *datetime*, fipa\_message\_string\_p *message*)**

Definition at line 850 of file fipa\_acl.c.

References `buf`, `fipa_GetWholeToken()`, `MC_ERR_PARSE`, `MC_SUCCESS`, `fipa_message_string_s::parse`, and `fipa_DateTime_s::sign`.

Referenced by `fipa_envelope_HandleDate()`, `fipa_envelope_HandleReceived()`, `fipa_expression_Parse()`, and `fipa_message_parameter_Parse()`.

### 5.33.3.22 `int fipa_expression_Compose (dynstring_t * msg, fipa_expression_t * expr)`

Definition at line 1473 of file `fipa_acl.c`.

References `fipa_expression_s::content`, `fipa_expression_s::content_u::datetime`, `fipa_expression_s::content_u::expression`, `fipa_DateTime_Compose()`, `FIPA_EXPR_DATETIME`, `FIPA_EXPR_EXPRESSION`, `FIPA_EXPR_NUMBER`, `FIPA_EXPR_STRING`, `FIPA_EXPR_WORD`, `fipa_expression_Compose()`, `fipa_number_Compose()`, `fipa_string_Compose()`, `fipa_word_Compose()`, `MC_ERR_PARSE`, `fipa_expression_s::content_u::number`, `fipa_expression_s::content_u::string`, `fipa_expression_s::type`, and `fipa_expression_s::content_u::word`.

Referenced by `fipa_acl_Compose()`, and `fipa_expression_Compose()`.

### 5.33.3.23 `fipa_expression_t* fipa_expression_Copy (fipa_expression_t * src)`

Definition at line 287 of file `fipa_acl.c`.

References `fipa_expression_s::content`, `fipa_expression_s::content_u::datetime`, `fipa_expression_s::content_u::expression`, `fipa_DateTime_Copy()`, `FIPA_EXPR_DATETIME`, `FIPA_EXPR_EXPRESSION`, `FIPA_EXPR_NUMBER`, `FIPA_EXPR_STRING`, `FIPA_EXPR_WORD`, `fipa_expression_Copy()`, `fipa_expression_Destroy()`, `fipa_expression_New()`, `fipa_number_Copy()`, `fipa_string_Copy()`, `fipa_word_Copy()`, `fipa_expression_s::content_u::number`, `fipa_expression_s::content_u::string`, `fipa_expression_s::type`, and `fipa_expression_s::content_u::word`.

Referenced by `fipa_acl_message_Copy()`, `fipa_expression_Copy()`, and `fipa_Reply()`.

### 5.33.3.24 `int fipa_expression_Destroy (fipa_expression_t * expr)`

Definition at line 256 of file `fipa_acl.c`.

References `fipa_expression_s::content`, `fipa_expression_s::content_u::datetime`, `fipa_expression_s::content_u::expression`, `fipa_DateTime_Destroy()`, `FIPA_EXPR_DATETIME`, `FIPA_EXPR_EXPRESSION`, `FIPA_EXPR_NUMBER`, `FIPA_EXPR_STRING`, `FIPA_EXPR_WORD`, `fipa_expression_Destroy()`, `fipa_number_Destroy()`, `fipa_string_Destroy()`, `fipa_word_Destroy()`, `FREEMEM`, `fipa_expression_s::content_u::number`, `fipa_expression_s::content_u::string`, `fipa_expression_s::type`, and `fipa_expression_s::content_u::word`.

Referenced by `fipa_acl_message_Destroy()`, `fipa_expression_Copy()`, and `fipa_expression_Destroy()`.

### 5.33.3.25 `fipa_expression_t* fipa_expression_New (void)`

Definition at line 248 of file `fipa_acl.c`.

Referenced by `fipa_expression_Copy()`, and `MC_AclSetConversationID()`.

**5.33.3.26 int fipa\_expression\_Parse (fipa\_expression\_t \*\* *expression*, fipa\_message\_string\_p *message*)**

Definition at line 714 of file fipa\_acl.c.

References fipa\_CheckNextToken(), fipa\_datetime\_Parse(), FIPA\_EXPR\_DATETIME, FIPA\_EXPR\_EXPRESSION, FIPA\_EXPR\_STRING, FIPA\_EXPR\_WORD, fipa\_expression\_Parse(), fipa\_GetAtom(), fipa\_string\_Parse(), fipa\_word\_Parse(), MC\_ERR\_PARSE, MC\_SUCCESS, and fipa\_expression\_s::type.

Referenced by fipa\_expression\_Parse(), and fipa\_message\_parameter\_Parse().

**5.33.3.27 int fipa\_GetAtom (fipa\_message\_string\_p *message*, char *expected\_atom*)**

Definition at line 630 of file fipa\_acl.c.

References MC\_ERR\_PARSE, MC\_SUCCESS, and fipa\_message\_string\_s::parse.

Referenced by fipa\_acl\_Parse(), fipa\_agent\_identifier\_Parse(), fipa\_agent\_identifier\_set\_Parse(), fipa\_expression\_Parse(), fipa\_message\_parameter\_Parse(), fipa\_string\_Parse(), and fipa\_url\_sequence\_Parse().

**5.33.3.28 int fipa\_GetNextWord (char \*\* *word*, const fipa\_message\_string\_p *message*)**

Definition at line 765 of file fipa\_acl.c.

References ERR, MC\_SUCCESS, and fipa\_message\_string\_s::parse.

**5.33.3.29 int fipa\_GetWholeToken (char \*\* *word*, const fipa\_message\_string\_p *message*)**

Definition at line 813 of file fipa\_acl.c.

References MC\_SUCCESS, and fipa\_message\_string\_s::parse.

Referenced by fipa\_datetime\_Parse().

**5.33.3.30 int fipa\_message\_parameter\_Parse (struct fipa\_acl\_message\_s \* *acl*, fipa\_message\_string\_p *message*)****5.33.3.31 fipa\_message\_string\_t\* fipa\_message\_string\_Copy (fipa\_message\_string\_t \* *src*)**

Definition at line 128 of file fipa\_acl.c.

References fipa\_message\_string\_s::message, fipa\_message\_string\_s::parse, and strdup.

**5.33.3.32 int fipa\_message\_string\_Destroy (fipa\_message\_string\_t \* *message*)**

Definition at line 118 of file fipa\_acl.c.

References fipa\_message\_string\_s::message.

Referenced by acc\_connection\_Thread().

**5.33.3.33 fipa\_message\_string\_t\* fipa\_message\_string\_New (void)**

Definition at line 110 of file fipa\_acl.c.

Referenced by `acc_connection_Thread()`.

#### **5.33.3.34 `int fipa_message_type_Parse (enum fipa_performative_e * performative, fipa_message_string_p message)`**

Definition at line 568 of file `fipa_acl.c`.

References `fipa_word_s::content`, `FIPA_ACCEPT_PROPOSAL`, `FIPA_AGREE`, `FIPA_CALL_FOR_PROPOSAL`, `FIPA_CANCEL`, `FIPA_CONFIRM`, `FIPA_DISCONFIRM`, `FIPA_FAILURE`, `FIPA_INFORM`, `FIPA_INFORM_IF`, `FIPA_INFORM_REF`, `FIPA_NOT_UNDERSTOOD`, `FIPA_PROPOGATE`, `FIPA_PROPOSE`, `FIPA_PROXY`, `FIPA_QUERY_IF`, `FIPA_QUERY_REF`, `FIPA_REFUSE`, `FIPA_REJECT_PROPOSAL`, `FIPA_REQUEST`, `FIPA_REQUEST_WHEN`, `FIPA_REQUEST_WHENEVER`, `FIPA_SUBSCRIBE`, `fipa_word_Destroy()`, `fipa_word_Parse()`, and `MC_ERR_PARSE`.

Referenced by `fipa_acl_Parse()`.

#### **5.33.3.35 `int fipa_number_Compose (dynstring_t * msg, fipa_number_t * number)`**

Definition at line 1548 of file `fipa_acl.c`.

References `dynstring_Append()`, and `fipa_number_s::str`.

Referenced by `fipa_expression_Compose()`.

#### **5.33.3.36 `fipa_number_t* fipa_number_Copy (fipa_number_t * src)`**

Definition at line 455 of file `fipa_acl.c`.

References `fipa_number_New()`, `fipa_number_s::str`, and `strdup`.

Referenced by `fipa_expression_Copy()`.

#### **5.33.3.37 `int fipa_number_Destroy (fipa_number_t * number)`**

Definition at line 445 of file `fipa_acl.c`.

References `fipa_number_s::str`.

Referenced by `fipa_expression_Destroy()`.

#### **5.33.3.38 `fipa_number_t* fipa_number_New (void)`**

Definition at line 437 of file `fipa_acl.c`.

Referenced by `fipa_number_Copy()`.

#### **5.33.3.39 `int fipa_performative_Compose (dynstring_t * msg, enum fipa_performative_e performative)`**

Definition at line 1346 of file `fipa_acl.c`.

References `dynstring_Append()`, `FIPA_ACCEPT_PROPOSAL`, `FIPA_AGREE`, `FIPA_CALL_FOR_PROPOSAL`, `FIPA_CANCEL`, `FIPA_CONFIRM`, `FIPA_DISCONFIRM`, `FIPA_FAILURE`,

FIPA\_INFORM, FIPA\_INFORM\_IF, FIPA\_INFORM\_REF, FIPA\_NOT\_UNDERSTOOD, FIPA\_PROPOGATE, FIPA\_PROPOSE, FIPA\_PROXY, FIPA\_QUERY\_IF, FIPA\_QUERY\_REF, FIPA\_REFUSE, FIPA\_REJECT\_PROPOSAL, FIPA\_REQUEST, FIPA\_REQUEST\_WHEN, FIPA\_REQUEST\_WHENEVER, FIPA\_SUBSCRIBE, and MC\_ERR\_PARSE.

Referenced by fipa\_acl\_Compose().

#### 5.33.3.40 int fipa\_protocol\_Compose (dynstring\_t \* msg, enum fipa\_protocol\_e protocol)

Definition at line 1303 of file fipa\_acl.c.

References dynstring\_Append(), FIPA\_PROTOCOL\_BROKERING, FIPA\_PROTOCOL\_CONTRACT\_NET, FIPA\_PROTOCOL\_DUTCH\_AUCTION, FIPA\_PROTOCOL\_ENGLISH\_AUCTION, FIPA\_PROTOCOL\_ITERATED\_CONTRACT\_NET, FIPA\_PROTOCOL\_PROPOSE, FIPA\_PROTOCOL\_QUERY, FIPA\_PROTOCOL\_RECRUITING, FIPA\_PROTOCOL\_REQUEST, FIPA\_PROTOCOL\_REQUEST\_WHEN, FIPA\_PROTOCOL\_SUBSCRIBE, and MC\_ERR\_PARSE.

Referenced by fipa\_acl\_Compose().

#### 5.33.3.41 int fipa\_protocol\_type\_Parse (enum fipa\_protocol\_e \* protocol, fipa\_message\_string\_p message)

Definition at line 528 of file fipa\_acl.c.

References fipa\_word\_s::content, FIPA\_PROTOCOL\_BROKERING, FIPA\_PROTOCOL\_CONTRACT\_NET, FIPA\_PROTOCOL\_DUTCH\_AUCTION, FIPA\_PROTOCOL\_ENGLISH\_AUCTION, FIPA\_PROTOCOL\_ITERATED\_CONTRACT\_NET, FIPA\_PROTOCOL\_PROPOSE, FIPA\_PROTOCOL\_QUERY, FIPA\_PROTOCOL\_RECRUITING, FIPA\_PROTOCOL\_REQUEST, FIPA\_PROTOCOL\_REQUEST\_WHEN, FIPA\_PROTOCOL\_SUBSCRIBE, fipa\_word\_Destroy(), fipa\_word\_Parse(), and MC\_ERR\_PARSE.

Referenced by fipa\_message\_parameter\_Parse().

#### 5.33.3.42 struct fipa\_acl\_message\_s\* fipa\_Reply (struct fipa\_acl\_message\_s \* acl) [read]

Definition at line 1556 of file fipa\_acl.c.

References fipa\_acl\_message\_s::conversation\_id, fipa\_acl\_message\_New(), fipa\_agent\_identifier\_Copy(), fipa\_agent\_identifier\_set\_Copy(), fipa\_agent\_identifier\_set\_New(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_expression\_Copy(), fipa\_agent\_identifier\_set\_s::num, fipa\_acl\_message\_s::protocol, fipa\_acl\_message\_s::receiver, fipa\_acl\_message\_s::reply\_to, fipa\_agent\_identifier\_set\_s::retain\_order, and fipa\_acl\_message\_s::sender.

Referenced by MC\_AclReply().

#### 5.33.3.43 int fipa\_string\_Compose (dynstring\_t \* msg, fipa\_string\_t \* string)

Definition at line 1511 of file fipa\_acl.c.

References fipa\_string\_s::content, and dynstring\_Append().

Referenced by fipa\_acl\_Compose(), and fipa\_expression\_Compose().

**5.33.3.44 fipa\_string\_t\* fipa\_string\_Copy (fipa\_string\_t \* src)**

Definition at line 374 of file fipa\_acl.c.

References fipa\_string\_s::content, fipa\_string\_New(), and strdup.

Referenced by fipa\_acl\_message\_Copy(), and fipa\_expression\_Copy().

**5.33.3.45 int fipa\_string\_Destroy (fipa\_string\_t \* str)**

Definition at line 364 of file fipa\_acl.c.

References fipa\_string\_s::content.

Referenced by fipa\_acl\_message\_Destroy(), fipa\_expression\_Destroy(), and MC\_AclSetContent().

**5.33.3.46 fipa\_string\_t\* fipa\_string\_New (void)**

Definition at line 356 of file fipa\_acl.c.

Referenced by fipa\_string\_Copy(), MC\_AclSetContent(), and MC\_AclSetConversationID().

**5.33.3.47 int fipa\_string\_Parse (fipa\_string\_p \* fipa\_string, fipa\_message\_string\_p message)**

Definition at line 988 of file fipa\_acl.c.

References fipa\_string\_s::content, fipa\_GetAtom(), MC\_ERR\_PARSE, MC\_SUCCESS, and fipa\_message\_string\_s::parse.

Referenced by fipa\_expression\_Parse(), and fipa\_message\_parameter\_Parse().

**5.33.3.48 int fipa\_url\_Compose (dynstring\_t \* msg, fipa\_url\_t \* url)**

Definition at line 1540 of file fipa\_acl.c.

References dynstring\_Append(), and fipa\_url\_s::str.

Referenced by fipa\_url\_sequence\_Compose().

**5.33.3.49 fipa\_url\_t\* fipa\_url\_Copy (fipa\_url\_t \* src)**

Definition at line 427 of file fipa\_acl.c.

References fipa\_url\_New(), fipa\_url\_s::str, and strdup.

Referenced by fipa\_acl\_envelope\_Received\_Copy(), and fipa\_url\_sequence\_Copy().

**5.33.3.50 int fipa\_url\_Destroy (fipa\_url\_t \* url)**

Definition at line 417 of file fipa\_acl.c.

References fipa\_url\_s::str.

Referenced by fipa\_acl\_envelope\_Received\_Destroy(), and fipa\_url\_sequence\_Destroy().

**5.33.3.51 fipa\_url\_t\* fipa\_url\_New (void)**

Definition at line 409 of file fipa\_acl.c.

Referenced by fipa\_envelope\_HandleReceived(), fipa\_envelope\_ParseAddresses(), fipa\_url\_Copy(), MC\_AclAddReceiver(), MC\_AclAddReplyTo(), and MC\_AclSetSender().

**5.33.3.52 int fipa\_url\_Parse (fipa\_url\_p \* url, fipa\_message\_string\_p message)**

Definition at line 1172 of file fipa\_acl.c.

References fipa\_word\_s::content, fipa\_word\_Destroy(), fipa\_word\_Parse(), and strdup.

Referenced by fipa\_url\_sequence\_Parse().

**5.33.3.53 int fipa\_url\_sequence\_Compose (dynstring\_t \* msg, fipa\_url\_sequence\_t \* urls)**

Definition at line 1421 of file fipa\_acl.c.

References dynstring\_Append(), fipa\_url\_Compose(), fipa\_url\_sequence\_s::num, and fipa\_url\_sequence\_s::urls.

Referenced by fipa\_agent\_identifier\_Compose().

**5.33.3.54 fipa\_url\_sequence\_t\* fipa\_url\_sequence\_Copy (fipa\_url\_sequence\_t \* src)**

Definition at line 159 of file fipa\_acl.c.

References fipa\_url\_Copy(), fipa\_url\_sequence\_New(), fipa\_url\_sequence\_s::num, and fipa\_url\_sequence\_s::urls.

Referenced by fipa\_agent\_identifier\_Copy().

**5.33.3.55 int fipa\_url\_sequence\_Destroy (fipa\_url\_sequence\_t \* sequence)**

Definition at line 147 of file fipa\_acl.c.

References fipa\_url\_Destroy(), fipa\_url\_sequence\_s::num, and fipa\_url\_sequence\_s::urls.

Referenced by fipa\_agent\_identifier\_Destroy().

**5.33.3.56 fipa\_url\_sequence\_t\* fipa\_url\_sequence\_New (void)**

Definition at line 139 of file fipa\_acl.c.

Referenced by fipa\_url\_sequence\_Copy(), fipa\_url\_sequence\_Parse(), MC\_AclAddReceiver(), MC\_AclAddReplyTo(), and MC\_AclSetSender().

**5.33.3.57 int fipa\_url\_sequence\_Parse (fipa\_url\_sequence\_p \* urls, fipa\_message\_string\_p message)**

Definition at line 1140 of file fipa\_acl.c.

References fipa\_word\_s::content, fipa\_GetAtom(), fipa\_url\_Parse(), fipa\_url\_sequence\_New(), fipa\_word\_Destroy(), fipa\_word\_Parse(), and MC\_ERR\_PARSE.

Referenced by `fipa_agent_identifier_Parse()`.

#### **5.33.3.58** `int fipa_word_Compose (dynstring_t * msg, fipa_word_t * word)`

Definition at line 1503 of file `fipa_acl.c`.

References `fipa_word_s::content`, and `dynstring_Append()`.

Referenced by `fipa_expression_Compose()`.

#### **5.33.3.59** `fipa_word_t* fipa_word_Copy (fipa_word_t * src)`

Definition at line 346 of file `fipa_acl.c`.

References `fipa_word_s::content`, `fipa_word_New()`, and `strdup`.

Referenced by `fipa_expression_Copy()`.

#### **5.33.3.60** `int fipa_word_Destroy (fipa_word_t * word)`

Definition at line 336 of file `fipa_acl.c`.

References `fipa_word_s::content`.

Referenced by `fipa_agent_identifier_Parse()`, `fipa_expression_Destroy()`, `fipa_message_parameter_Parse()`, `fipa_message_type_Parse()`, `fipa_protocol_type_Parse()`, `fipa_url_Parse()`, and `fipa_url_sequence_Parse()`.

#### **5.33.3.61** `fipa_word_t* fipa_word_New (void)`

Definition at line 328 of file `fipa_acl.c`.

Referenced by `fipa_word_Copy()`.

#### **5.33.3.62** `int fipa_word_Parse (fipa_word_t ** word, fipa_message_string_p message)`

Definition at line 653 of file `fipa_acl.c`.

References `CHECK_NULL`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

Referenced by `fipa_agent_identifier_Parse()`, `fipa_agent_identifier_set_Parse()`, `fipa_expression_Parse()`, `fipa_message_parameter_Parse()`, `fipa_message_type_Parse()`, `fipa_protocol_type_Parse()`, `fipa_url_Parse()`, and `fipa_url_sequence_Parse()`.



## 5.34 /home/dko/Projects/mobilec/src/include/fipa\_acl\_envelope.h File Reference

```
#include "fipa_acl.h"
```

### Data Structures

- struct [fipa\\_acl\\_envelope\\_Received\\_s](#)
- struct [fipa\\_acl\\_Param\\_s](#)
- struct [fipa\\_acl\\_envelope\\_s](#)

### Typedefs

- typedef struct [fipa\\_acl\\_envelope\\_Received\\_s](#) [fipa\\_acl\\_envelope\\_Received\\_t](#)
- typedef struct [fipa\\_acl\\_Param\\_s](#) [fipa\\_acl\\_Param\\_t](#)
- typedef struct [fipa\\_acl\\_envelope\\_s](#) [fipa\\_acl\\_envelope\\_t](#)
- typedef [fipa\\_acl\\_envelope\\_t](#) \* [fipa\\_acl\\_envelope\\_p](#)

### Functions

- [fipa\\_acl\\_envelope\\_Received\\_t](#) \* [fipa\\_acl\\_envelope\\_Received\\_New](#) (void)
- int [fipa\\_acl\\_envelope\\_Received\\_Destroy](#) ([fipa\\_acl\\_envelope\\_Received\\_t](#) \*received)
- [fipa\\_acl\\_envelope\\_Received\\_t](#) \* [fipa\\_acl\\_envelope\\_Received\\_Copy](#) ([fipa\\_acl\\_envelope\\_Received\\_t](#) \*received)
- [fipa\\_acl\\_Param\\_t](#) \* [fipa\\_acl\\_Param\\_New](#) (void)
- int [fipa\\_acl\\_Param\\_Destroy](#) ([fipa\\_acl\\_Param\\_t](#) \*param)
- [fipa\\_acl\\_Param\\_t](#) \* [fipa\\_acl\\_Param\\_Copy](#) ([fipa\\_acl\\_Param\\_t](#) \*param)
- [fipa\\_acl\\_envelope\\_t](#) \* [fipa\\_acl\\_envelope\\_New](#) (void)
- int [fipa\\_acl\\_envelope\\_Destroy](#) ([fipa\\_acl\\_envelope\\_t](#) \*envelope)
- [fipa\\_acl\\_envelope\\_t](#) \* [fipa\\_acl\\_envelope\\_Copy](#) ([fipa\\_acl\\_envelope\\_t](#) \*envelope)
- int [fipa\\_envelope\\_Parse](#) (struct [fipa\\_acl\\_envelope\\_s](#) \*envelope, const char \*message)
- int [fipa\\_envelope\\_HandleEnvelope](#) (struct [fipa\\_acl\\_envelope\\_s](#) \*envelope, [mxml\\_node\\_t](#) \*node)
- int [fipa\\_envelope\\_HandleParams](#) (struct [fipa\\_acl\\_envelope\\_s](#) \*envelope, [mxml\\_node\\_t](#) \*node)
- int [fipa\\_envelope\\_HandleTo](#) (struct [fipa\\_acl\\_envelope\\_s](#) \*envelope, [mxml\\_node\\_t](#) \*param\_node, int cur\_param)
- int [fipa\\_envelope\\_HandleFrom](#) (struct [fipa\\_acl\\_envelope\\_s](#) \*envelope, [mxml\\_node\\_t](#) \*param\_node, int cur\_param)
- int [fipa\\_envelope\\_HandleComments](#) (struct [fipa\\_acl\\_envelope\\_s](#) \*envelope, [mxml\\_node\\_t](#) \*param\_node, int cur\_param)
- int [fipa\\_envelope\\_HandleAclRepresentation](#) (struct [fipa\\_acl\\_envelope\\_s](#) \*envelope, [mxml\\_node\\_t](#) \*param\_node, int cur\_param)
- int [fipa\\_envelope\\_HandlePayloadLength](#) (struct [fipa\\_acl\\_envelope\\_s](#) \*envelope, [mxml\\_node\\_t](#) \*param\_node, int cur\_param)
- int [fipa\\_envelope\\_HandlePayloadEncoding](#) (struct [fipa\\_acl\\_envelope\\_s](#) \*envelope, [mxml\\_node\\_t](#) \*param\_node, int cur\_param)
- int [fipa\\_envelope\\_HandleDate](#) (struct [fipa\\_acl\\_envelope\\_s](#) \*envelope, [mxml\\_node\\_t](#) \*param\_node, int cur\_param)
- int [fipa\\_envelope\\_HandleIntendedReceiver](#) (struct [fipa\\_acl\\_envelope\\_s](#) \*envelope, [mxml\\_node\\_t](#) \*param\_node, int cur\_param)

- `int fipa_envelope_HandleReceived` (struct `fipa_acl_envelope_s` \*envelope, `mxml_node_t` \*param\_node, `int` cur\_param)
- `int fipa_envelope_ParseAgentIdentifier` (struct `fipa_agent_identifier_s` \*\*aid, `mxml_node_t` \*agent\_identifier\_node)
- `int fipa_envelope_ParseAddresses` (struct `fipa_agent_identifier_s` \*aid, `mxml_node_t` \*addresses\_node)
- `int fipa_envelope_ParseResolvers` (struct `fipa_agent_identifier_s` \*aid, `mxml_node_t` \*resolvers\_node)
- `char * fipa_envelope_Compose` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__envelope` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__params` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__to` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__from` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__acl_representation` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__payload_encoding` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__date` (`fipa_acl_message_t` \*fipa\_acl)
- `mxml_node_t * fipa_envelope_Compose__intended_receiver` (`fipa_acl_message_t` \*fipa\_acl)

### 5.34.1 Typedef Documentation

#### 5.34.1.1 `typedef fipa_acl_envelope_t* fipa_acl_envelope_p`

Definition at line 45 of file `fipa_acl_envelope.h`.

#### 5.34.1.2 `typedef struct fipa_acl_envelope_Received_s fipa_acl_envelope_Received_t`

#### 5.34.1.3 `typedef struct fipa_acl_envelope_s fipa_acl_envelope_t`

#### 5.34.1.4 `typedef struct fipa_acl_Param_s fipa_acl_Param_t`

### 5.34.2 Function Documentation

#### 5.34.2.1 `fipa_acl_envelope_t* fipa_acl_envelope_Copy (fipa_acl_envelope_t * envelope)`

Definition at line 150 of file `fipa_envelope.c`.

References `fipa_acl_envelope_New()`, `fipa_acl_Param_Copy()`, `fipa_acl_envelope_s::num_params`, and `fipa_acl_envelope_s::params`.

#### 5.34.2.2 `int fipa_acl_envelope_Destroy (fipa_acl_envelope_t * envelope)`

Definition at line 137 of file `fipa_envelope.c`.

References `fipa_acl_Param_Destroy()`, `fipa_acl_envelope_s::num_params`, and `fipa_acl_envelope_s::params`.

Referenced by `acc_connection_Thread()`.

#### 5.34.2.3 fipa\_acl\_envelope\_t\* fipa\_acl\_envelope\_New (void)

Definition at line 129 of file fipa\_envelope.c.

Referenced by acc\_connection\_Thread(), and fipa\_acl\_envelope\_Copy().

#### 5.34.2.4 fipa\_acl\_envelope\_Received\_t\* fipa\_acl\_envelope\_Received\_Copy (fipa\_acl\_envelope\_Received\_t \* *received*)

Definition at line 69 of file fipa\_envelope.c.

References fipa\_acl\_envelope\_Received\_New(), fipa\_DateTime\_Copy(), fipa\_url\_Copy(), fipa\_acl\_envelope\_Received\_s::received\_by, fipa\_acl\_envelope\_Received\_s::received\_date, fipa\_acl\_envelope\_Received\_s::received\_from, fipa\_acl\_envelope\_Received\_s::received\_id, fipa\_acl\_envelope\_Received\_s::received\_via, and strdup.

Referenced by fipa\_acl\_Param\_Copy().

#### 5.34.2.5 int fipa\_acl\_envelope\_Received\_Destroy (fipa\_acl\_envelope\_Received\_t \* *received*)

Definition at line 57 of file fipa\_envelope.c.

References fipa\_DateTime\_Destroy(), fipa\_url\_Destroy(), fipa\_acl\_envelope\_Received\_s::received\_by, fipa\_acl\_envelope\_Received\_s::received\_date, fipa\_acl\_envelope\_Received\_s::received\_from, fipa\_acl\_envelope\_Received\_s::received\_id, and fipa\_acl\_envelope\_Received\_s::received\_via.

Referenced by fipa\_acl\_Param\_Destroy().

#### 5.34.2.6 fipa\_acl\_envelope\_Received\_t\* fipa\_acl\_envelope\_Received\_New (void)

Definition at line 48 of file fipa\_envelope.c.

Referenced by fipa\_acl\_envelope\_Received\_Copy(), and fipa\_envelope\_HandleReceived().

#### 5.34.2.7 fipa\_acl\_Param\_t\* fipa\_acl\_Param\_Copy (fipa\_acl\_Param\_t \* *param*)

Definition at line 110 of file fipa\_envelope.c.

References fipa\_acl\_Param\_s::acl\_representation, fipa\_acl\_Param\_s::comments, fipa\_acl\_Param\_s::date, fipa\_acl\_envelope\_Received\_Copy(), fipa\_acl\_Param\_New(), fipa\_agent\_identifier\_Copy(), fipa\_agent\_identifier\_set\_Copy(), fipa\_DateTime\_Copy(), fipa\_acl\_Param\_s::from, fipa\_acl\_Param\_s::intended\_receiver, fipa\_acl\_Param\_s::payload\_encoding, fipa\_acl\_Param\_s::payload\_length, fipa\_acl\_Param\_s::received, strdup, and fipa\_acl\_Param\_s::to.

Referenced by fipa\_acl\_envelope\_Copy().

#### 5.34.2.8 int fipa\_acl\_Param\_Destroy (fipa\_acl\_Param\_t \* *param*)

Definition at line 93 of file fipa\_envelope.c.

References fipa\_acl\_Param\_s::acl\_representation, fipa\_acl\_Param\_s::comments, fipa\_acl\_Param\_s::date, fipa\_acl\_envelope\_Received\_Destroy(), fipa\_agent\_identifier\_Destroy(), fipa\_agent\_identifier\_set\_Destroy(), fipa\_DateTime\_Destroy(), fipa\_acl\_Param\_s::from, fipa\_acl\_Param\_s::intended\_receiver, fipa\_acl\_Param\_s::payload\_encoding, fipa\_acl\_Param\_s::payload\_length, fipa\_acl\_Param\_s::received, and fipa\_acl\_Param\_s::to.

Referenced by `fipa_acl_envelope_Destroy()`.

#### 5.34.2.9 `fipa_acl_Param_t* fipa_acl_Param_New (void)`

Definition at line 85 of file `fipa_envelope.c`.

Referenced by `fipa_acl_Param_Copy()`, and `fipa_envelope_HandleTo()`.

#### 5.34.2.10 `char* fipa_envelope_Compose (fipa_acl_message_t * fipa_acl)`

Definition at line 869 of file `fipa_envelope.c`.

References `fipa_envelope_Compose__envelope()`, `MXML_ADD_AFTER`, `MXML_ADD_TO_PARENT`, `MXML_NO_CALLBACK`, `mxmlAdd()`, `mxmlDelete()`, `mxmlLoadString()`, `mxmlSaveAllocString()`, and `node`.

Referenced by `MC_AclSend()`.

#### 5.34.2.11 `mxml_node_t* fipa_envelope_Compose__acl_representation (fipa_acl_message_t * fipa_acl)`

Definition at line 1089 of file `fipa_envelope.c`.

References `mxmlNewElement()`, `mxmlNewText()`, and `node`.

Referenced by `fipa_envelope_Compose__params()`.

#### 5.34.2.12 `mxml_node_t* fipa_envelope_Compose__date (fipa_acl_message_t * fipa_acl)`

Definition at line 1117 of file `fipa_envelope.c`.

References `buf`, `mxmlNewElement()`, `mxmlNewText()`, and `node`.

Referenced by `fipa_envelope_Compose__params()`.

#### 5.34.2.13 `mxml_node_t* fipa_envelope_Compose__envelope (fipa_acl_message_t * fipa_acl)`

Definition at line 894 of file `fipa_envelope.c`.

References `fipa_envelope_Compose__params()`, `MXML_ADD_AFTER`, `MXML_ADD_TO_PARENT`, `mxmlAdd()`, `mxmlNewElement()`, and `node`.

Referenced by `fipa_envelope_Compose()`.

#### 5.34.2.14 `mxml_node_t* fipa_envelope_Compose__from (fipa_acl_message_t * fipa_acl)`

Definition at line 1021 of file `fipa_envelope.c`.

References `fipa_agent_identifier_s::addresses`, `buf`, `dynstring_Append()`, `dynstring_Destroy()`, `dynstring_New()`, `g_mc_platform`, `mc_platform_s::hostname`, `dynstring_s::message`, `mxmlNewElement()`, `mxmlNewText()`, `fipa_agent_identifier_s::name`, `fipa_url_sequence_s::num`, `mc_platform_s::port`, `fipa_acl_message_s::sender`, `fipa_url_s::str`, and `fipa_url_sequence_s::urls`.

Referenced by `fipa_envelope_Compose__params()`.

**5.34.2.15 mxmml\_node\_t\* fipa\_envelope\_Compose\_\_intended\_receiver (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 1146 of file fipa\_envelope.c.

References fipa\_agent\_identifier\_s::addresses, fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, mxmml-NewElement(), mxmmlNewText(), fipa\_agent\_identifier\_s::name, node, fipa\_url\_sequence\_s::num, fipa\_acl\_message\_s::receiver, fipa\_url\_s::str, and fipa\_url\_sequence\_s::urls.

Referenced by fipa\_envelope\_Compose\_\_params().

**5.34.2.16 mxmml\_node\_t\* fipa\_envelope\_Compose\_\_params (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 914 of file fipa\_envelope.c.

References fipa\_envelope\_Compose\_\_acl\_representation(), fipa\_envelope\_Compose\_\_date(), fipa\_envelope\_Compose\_\_from(), fipa\_envelope\_Compose\_\_intended\_receiver(), fipa\_envelope\_Compose\_\_payload\_encoding(), fipa\_envelope\_Compose\_\_to(), MXML\_ADD\_AFTER, MXML\_ADD\_TO\_PARENT, mxmmlAdd(), mxmmlElementSetAttr(), mxmmlNewElement(), and node.

Referenced by fipa\_envelope\_Compose\_\_envelope().

**5.34.2.17 mxmml\_node\_t\* fipa\_envelope\_Compose\_\_payload\_encoding (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 1103 of file fipa\_envelope.c.

References mxmmlNewElement(), mxmmlNewText(), and node.

Referenced by fipa\_envelope\_Compose\_\_params().

**5.34.2.18 mxmml\_node\_t\* fipa\_envelope\_Compose\_\_to (fipa\_acl\_message\_t \* *fipa\_acl*)**

Definition at line 975 of file fipa\_envelope.c.

References fipa\_agent\_identifier\_s::addresses, fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, mxmml-NewElement(), mxmmlNewText(), fipa\_agent\_identifier\_s::name, node, fipa\_url\_sequence\_s::num, fipa\_agent\_identifier\_set\_s::num, fipa\_acl\_message\_s::receiver, fipa\_url\_s::str, and fipa\_url\_sequence\_s::urls.

Referenced by fipa\_envelope\_Compose\_\_params().

**5.34.2.19 int fipa\_envelope\_HandleAclRepresentation (struct fipa\_acl\_envelope\_s \* *envelope*, mxmml\_node\_t \* *param\_node*, int *cur\_param*)**

Definition at line 381 of file fipa\_envelope.c.

References fipa\_acl\_Param\_s::acl\_representation, mxmml\_node\_s::child, MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_TEXT, mxmmlFindElement(), fipa\_acl\_envelope\_s::params, mxmml\_text\_s::string, mxmml\_value\_u::text, mxmml\_node\_s::type, and mxmml\_node\_s::value.

Referenced by fipa\_envelope\_HandleParams().

### 5.34.2.20 **int fipa\_envelope\_HandleComments** (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)

Definition at line 357 of file fipa\_envelope.c.

References mxml\_node\_s::child, fipa\_acl\_Param\_s::comments, MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_TEXT, mxmlFindElement(), fipa\_acl\_envelope\_s::params, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by fipa\_envelope\_HandleParams().

### 5.34.2.21 **int fipa\_envelope\_HandleDate** (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)

Definition at line 452 of file fipa\_envelope.c.

References mxml\_node\_s::child, fipa\_acl\_Param\_s::date, fipa\_datetime\_Parse(), MC\_ERR\_PARSE, fipa\_message\_string\_s::message, MXML\_DESCEND\_FIRST, MXML\_TEXT, mxmlFindElement(), fipa\_acl\_envelope\_s::params, fipa\_message\_string\_s::parse, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by fipa\_envelope\_HandleParams().

### 5.34.2.22 **int fipa\_envelope\_HandleEnvelope** (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *node*)

Definition at line 178 of file fipa\_envelope.c.

References fipa\_envelope\_HandleParams(), MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, and mxmlFindElement().

Referenced by fipa\_envelope\_Parse().

### 5.34.2.23 **int fipa\_envelope\_HandleFrom** (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)

Definition at line 328 of file fipa\_envelope.c.

References fipa\_envelope\_ParseAgentIdentifier(), fipa\_acl\_Param\_s::from, MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, mxmlFindElement(), and fipa\_acl\_envelope\_s::params.

Referenced by fipa\_envelope\_HandleParams().

### 5.34.2.24 **int fipa\_envelope\_HandleIntendedReceiver** (struct fipa\_acl\_envelope\_s \* *envelope*, mxml\_node\_t \* *param\_node*, int *cur\_param*)

Definition at line 482 of file fipa\_envelope.c.

References fipa\_agent\_identifier\_set\_New(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_envelope\_ParseAgentIdentifier(), fipa\_acl\_Param\_s::intended\_receiver, MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_NO\_DESCEND, mxmlFindElement(), fipa\_agent\_identifier\_set\_s::num, fipa\_acl\_envelope\_s::params, and fipa\_agent\_identifier\_set\_s::retain\_order.

Referenced by fipa\_envelope\_HandleParams().

**5.34.2.25** `int fipa_envelope_HandleParams (struct fipa_acl_envelope_s * envelope, mxml_node_t * node)`

Definition at line 198 of file fipa\_envelope.c.

References `buf`, `fipa_envelope_HandleAclRepresentation()`, `fipa_envelope_HandleComments()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandleFrom()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandlePayloadEncoding()`, `fipa_envelope_HandlePayloadLength()`, `fipa_envelope_HandleReceived()`, `fipa_envelope_HandleTo()`, `MC_ERR_PARSE`, `MXML_DESCEND_FIRST`, `mxmFindElement()`, `fipa_acl_envelope_s::num_params`, and `fipa_acl_envelope_s::params`.

Referenced by `fipa_envelope_HandleEnvelope()`.

**5.34.2.26** `int fipa_envelope_HandlePayloadEncoding (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)`

Definition at line 428 of file fipa\_envelope.c.

References `mxml_node_s::child`, `MC_ERR_PARSE`, `MXML_DESCEND_FIRST`, `MXML_TEXT`, `mxmFindElement()`, `fipa_acl_envelope_s::params`, `fipa_acl_Param_s::payload_encoding`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::type`, and `mxml_node_s::value`.

Referenced by `fipa_envelope_HandleParams()`.

**5.34.2.27** `int fipa_envelope_HandlePayloadLength (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)`

Definition at line 405 of file fipa\_envelope.c.

References `mxml_node_s::child`, `MC_ERR_PARSE`, `MXML_DESCEND_FIRST`, `MXML_TEXT`, `mxmFindElement()`, `fipa_acl_envelope_s::params`, `fipa_acl_Param_s::payload_length`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::type`, and `mxml_node_s::value`.

Referenced by `fipa_envelope_HandleParams()`.

**5.34.2.28** `int fipa_envelope_HandleReceived (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)`

Definition at line 560 of file fipa\_envelope.c.

References `fipa_acl_envelope_Received_New()`, `fipa_datetime_Parse()`, `fipa_url_New()`, `MC_ERR_PARSE`, `fipa_message_string_s::message`, `MXML_DESCEND_FIRST`, `mxmlElementGetAttr()`, `mxmFindElement()`, `node`, `fipa_acl_envelope_s::params`, `fipa_message_string_s::parse`, `fipa_acl_Param_s::received`, `fipa_acl_envelope_Received_s::received_by`, `fipa_acl_envelope_Received_s::received_date`, `fipa_acl_envelope_Received_s::received_from`, `fipa_acl_envelope_Received_s::received_id`, `fipa_acl_envelope_Received_s::received_via`, and `fipa_url_s::str`.

Referenced by `fipa_envelope_HandleParams()`.

**5.34.2.29** `int fipa_envelope_HandleTo (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)`

Definition at line 250 of file fipa\_envelope.c.

References `fipa_acl_Param_New()`, `fipa_agent_identifier_set_New()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, `fipa_envelope_ParseAgentIdentifier()`, `MC_ERR_PARSE`, `MXML_DESCEND_FIRST`,

MXML\_NO\_DESCEND, mxmlFindElement(), fipa\_agent\_identifier\_set\_s::num, fipa\_acl\_envelope\_s::params, fipa\_agent\_identifier\_set\_s::retain\_order, and fipa\_acl\_Param\_s::to.

Referenced by fipa\_envelope\_HandleParams().

#### 5.34.2.30 int fipa\_envelope\_Parse (struct fipa\_acl\_envelope\_s \* *envelope*, const char \* *message*)

Definition at line 165 of file fipa\_envelope.c.

References fipa\_envelope\_HandleEnvelope(), MXML\_NO\_CALLBACK, mxmlDelete(), and mxmlLoadString().

Referenced by acc\_connection\_Thread().

#### 5.34.2.31 int fipa\_envelope\_ParseAddresses (struct fipa\_agent\_identifier\_s \* *aid*, mxml\_node\_t \* *addresses\_node*)

Definition at line 737 of file fipa\_envelope.c.

References fipa\_agent\_identifier\_s::addresses, mxml\_node\_s::child, fipa\_url\_New(), MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_NO\_DESCEND, MXML\_TEXT, mxmlFindElement(), fipa\_url\_sequence\_s::num, fipa\_url\_s::str, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, fipa\_url\_sequence\_s::urls, and mxml\_node\_s::value.

Referenced by fipa\_envelope\_ParseAgentIdentifier().

#### 5.34.2.32 int fipa\_envelope\_ParseAgentIdentifier (struct fipa\_agent\_identifier\_s \*\* *aid*, mxml\_node\_t \* *agent\_identifier\_node*)

Definition at line 687 of file fipa\_envelope.c.

References mxml\_node\_s::child, fipa\_agent\_identifier\_New(), fipa\_envelope\_ParseAddresses(), fipa\_envelope\_ParseResolvers(), MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_TEXT, mxmlFindElement(), mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by fipa\_envelope\_HandleFrom(), fipa\_envelope\_HandleIntendedReceiver(), fipa\_envelope\_HandleTo(), and fipa\_envelope\_ParseResolvers().

#### 5.34.2.33 int fipa\_envelope\_ParseResolvers (struct fipa\_agent\_identifier\_s \* *aid*, mxml\_node\_t \* *resolvers\_node*)

Definition at line 804 of file fipa\_envelope.c.

References fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_envelope\_ParseAgentIdentifier(), MC\_ERR\_PARSE, MXML\_DESCEND\_FIRST, MXML\_NO\_DESCEND, mxmlFindElement(), fipa\_agent\_identifier\_set\_s::num, fipa\_agent\_identifier\_s::resolvers, and fipa\_agent\_identifier\_set\_s::retain\_order.

Referenced by fipa\_envelope\_ParseAgentIdentifier().



## 5.35 /home/dko/Projects/mobilec/src/include/fipa\_comm.h File Reference

```
#include <fipa_acl.h>
```

### Data Structures

- struct [fipa\\_comm\\_message\\_check\\_t](#)
- struct [fipa\\_comm\\_reply\\_t](#)
- struct [fipa\\_comm\\_action\\_t](#)
- struct [fipa\\_comm\\_performative\\_t](#)
- struct [fipa\\_list\\_t](#)
- struct [fipa\\_comm\\_protocol\\_cn\\_t](#)
- struct [fipa\\_comm\\_protocol\\_t](#)
- struct [fipa\\_comm\\_t](#)

### Defines

- #define [false](#) 0
- #define [MC\\_NUM\\_PERFORMATIVES](#) 23

### Typedefs

- typedef [int](#)(\* [fipa\\_comm\\_callback](#) )(char \*received\_content, char \*\*reply\_content)
- typedef [int](#)(\* [protocol\\_contract\\_net\\_callback](#) )(int numBids, char \*\*bids, char \*winners)
- typedef struct [fipa\\_comm\\_message\\_check\\_t](#) [fipa\\_comm\\_message\\_check\\_s](#)
- typedef struct [fipa\\_comm\\_message\\_check\\_t](#) \* [fipa\\_comm\\_message\\_check\\_p](#)
- typedef struct [fipa\\_comm\\_reply\\_t](#) [fipa\\_comm\\_reply\\_s](#)
- typedef struct [fipa\\_comm\\_reply\\_t](#) \* [fipa\\_comm\\_reply\\_p](#)
- typedef struct [fipa\\_comm\\_action\\_t](#) [fipa\\_comm\\_action\\_s](#)
- typedef struct [fipa\\_comm\\_action\\_t](#) \* [fipa\\_comm\\_action\\_p](#)
- typedef struct [fipa\\_comm\\_performative\\_t](#) [fipa\\_comm\\_performative\\_s](#)
- typedef struct [fipa\\_comm\\_performative\\_t](#) \* [fipa\\_comm\\_performative\\_p](#)
- typedef struct [fipa\\_list\\_t](#) [fipa\\_list\\_s](#)
- typedef struct [fipa\\_list\\_t](#) \* [fipa\\_list\\_p](#)
- typedef struct [fipa\\_comm\\_protocol\\_cn\\_t](#) [fipa\\_comm\\_protocol\\_cn\\_s](#)
- typedef struct [fipa\\_comm\\_protocol\\_cn\\_t](#) \* [fipa\\_comm\\_protocol\\_cn\\_p](#)
- typedef struct [fipa\\_comm\\_protocol\\_t](#) \* [fipa\\_comm\\_protocol\\_p](#)
- typedef struct [fipa\\_comm\\_protocol\\_t](#) [fipa\\_comm\\_protocol\\_s](#)
- typedef struct [fipa\\_comm\\_t](#) [fipa\\_comm\\_s](#)
- typedef struct [fipa\\_comm\\_t](#) \* [fipa\\_comm\\_p](#)
- typedef struct [fipa\\_comm\\_t](#) [MCFIPAComm\\_s](#)
- typedef struct [fipa\\_comm\\_t](#) \* [MCFIPAComm\\_p](#)

## Enumerations

- enum `fipa_comm_type_e` {  
`FIPA_COMM_ACTION`, `FIPA_COMM_CHECK`, `FIPA_COMM_REPLY`, `FIPA_COMM_CALLBACK`,  
`FIPA_COMM_END` }
- enum `fipa_comm_logic_type_e` { `FIPA_COMM_LOGIC_ERROR` = -1, `FIPA_COMM_LOGIC_OR`, `FIPA_COMM_LOGIC_AND` }
- enum `fipa_comm_check_type_e` {  
`FIPA_COMM_CHECK_ERROR` = -1, `FIPA_COMM_CHECK_SENDER_NAME`, `FIPA_COMM_CHECK_SENDER_ADDRESS`, `FIPA_COMM_CHECK_CONTENT`,  
`FIPA_COMM_CHECK_CONVERSATIONID`, `FIPA_COMM_CHECK_LANGUAGE`, `FIPA_COMM_CHECK_ONTOLOGY`, `FIPA_COMM_CHECK_ENCODING` }
- enum `fipa_comm_action_type_e` {  
`FIPA_COMM_ACTION_ERROR` = -1, `FIPA_COMM_ACTION_BOTH`, `FIPA_COMM_ACTION_REPLY`, `FIPA_COMM_ACTION_CALLBACK`,  
`FIPA_COMM_ACTION_NONE` }
- enum `fipa_comm_protocol_cn_state_e` {  
`FIPA_COMM_PROTOCOL_CN_ERROR` = -1, `FIPA_COMM_PROTOCOL_CN_START`, `FIPA_COMM_PROTOCOL_CN_CFP`, `FIPA_COMM_PROTOCOL_CN_BID`,  
`FIPA_COMM_PROTOCOL_CN_RESULT`, `FIPA_COMM_PROTOCOL_CN_DONE` }

## Functions

- `fipa_comm_p mc_FIPAComm_New ()`
- `int mc_FIPAComm_HandleMessage (fipa_comm_p fcomm, fipa_acl_message_p acl)`
- `int mc_FIPAComm_HandleMessageOnce (fipa_comm_p fcomm)`
- `int mc_FIPAComm_HandleMessageWait (fipa_comm_p fcomm)`
- `void mc_FIPAComm_Destroy (fipa_comm_p)`
- `int mc_FIPAComm_RegisterPerformative (fipa_comm_p fcomm, enum fipa_performative_e type)`
- `int mc_FIPAComm_DeregisterPerformative (fipa_comm_p fcomm, enum fipa_performative_e type)`
- `int mc_FIPAComm_SetDefaultReply (fipa_comm_p fcomm, enum fipa_performative_e type, char *content)`
- `int mc_FIPAComm_SetPerformativeDefaultReply (fipa_comm_p fcomm, enum fipa_performative_e ptype, enum fipa_performative_e rtype, char *content)`
- `int mc_FIPAComm_SetAgent (fipa_comm_p fcomm, void *agent, char *agent_name, char *agent_address)`
- `int mc_FIPAComm_RegisterAction (fipa_comm_p fcomm, enum fipa_performative_e pType, enum fipa_comm_action_type_e aType,...)`
- `int mc_FIPAComm_RegisterActionS (fipa_performative_e type, fipa_comm_action_p act)`
- `fipa_comm_protocol_p mc_FIPAComm_New ()`
- `int mc_FIPAComm_RegisterProtocolCallback (fipa_comm_p fcomm, enum fipa_protocol_e protocolType, fipa_comm_callback func)`
- `int mc_FIPAComm_RegisterProtocol (fipa_comm_p fcomm, enum fipa_protocol_e protocolType, char *protocolID, void *info, char *network)`
- `void fipa_comm_protocol_info_COPY (fipa_comm_protocol_p proto, void *info)`
- `int fipa_list_Add (fipa_list_p *list, fipa_acl_message_p acl)`
- `int mc_FIPAComm_StartProtocol (fipa_comm_p fcomm, enum fipa_protocol_e type, char *id)`
- `int mc_FIPAComm_HandleProtocol (fipa_comm_p fcomm, fipa_acl_message_p acl)`

- [fipa\\_comm\\_protocol\\_p mc\\_FIPAComm\\_AddNewProtocol](#) ([fipa\\_comm\\_p](#) fcomm, enum [fipa\\_protocol\\_e](#) type, char \*id)
- [int mc\\_FIPAComm\\_FSM\\_CN](#) ([fipa\\_comm\\_p](#) fcomm, [fipa\\_acl\\_message\\_p](#) acl)
- [fipa\\_comm\\_protocol\\_p mc\\_FIPAComm\\_NewProtocol](#) (enum [fipa\\_protocol\\_e](#) type, char \*id)
- [int mc\\_FIPAComm\\_SetProtocolState](#) ([fipa\\_comm\\_protocol\\_p](#) protocol, char state)

### 5.35.1 Define Documentation

#### 5.35.1.1 #define false 0

Definition at line 11 of file [fipa\\_comm.h](#).

#### 5.35.1.2 #define MC\_NUM\_PERFORMATIVES 23

Definition at line 14 of file [fipa\\_comm.h](#).

### 5.35.2 Typedef Documentation

#### 5.35.2.1 typedef struct fipa\_comm\_action\_t\* fipa\_comm\_action\_p

Definition at line 109 of file [fipa\\_comm.h](#).

#### 5.35.2.2 typedef struct fipa\_comm\_action\_t fipa\_comm\_action\_s

Definition at line 108 of file [fipa\\_comm.h](#).

#### 5.35.2.3 typedef int(\* fipa\_comm\_callback)(char \*received\_content, char \*\*reply\_content)

Definition at line 17 of file [fipa\\_comm.h](#).

#### 5.35.2.4 typedef struct fipa\_comm\_message\_check\_t\* fipa\_comm\_message\_check\_p

Definition at line 80 of file [fipa\\_comm.h](#).

#### 5.35.2.5 typedef struct fipa\_comm\_message\_check\_t fipa\_comm\_message\_check\_s

Definition at line 79 of file [fipa\\_comm.h](#).

#### 5.35.2.6 typedef struct fipa\_comm\_t\* fipa\_comm\_p

Definition at line 193 of file [fipa\\_comm.h](#).

#### 5.35.2.7 typedef struct fipa\_comm\_performative\_t\* fipa\_comm\_performative\_p

Definition at line 121 of file [fipa\\_comm.h](#).

**5.35.2.8 typedef struct fipa\_comm\_performative\_t fipa\_comm\_performative\_s**

Definition at line 120 of file fipa\_comm.h.

**5.35.2.9 typedef struct fipa\_comm\_protocol\_cn\_t\* fipa\_comm\_protocol\_cn\_p**

Definition at line 156 of file fipa\_comm.h.

**5.35.2.10 typedef struct fipa\_comm\_protocol\_cn\_t fipa\_comm\_protocol\_cn\_s**

Definition at line 155 of file fipa\_comm.h.

**5.35.2.11 typedef struct fipa\_comm\_protocol\_t\* fipa\_comm\_protocol\_p**

Definition at line 172 of file fipa\_comm.h.

**5.35.2.12 typedef struct fipa\_comm\_protocol\_t fipa\_comm\_protocol\_s**

Definition at line 173 of file fipa\_comm.h.

**5.35.2.13 typedef struct fipa\_comm\_reply\_t\* fipa\_comm\_reply\_p**

Definition at line 93 of file fipa\_comm.h.

**5.35.2.14 typedef struct fipa\_comm\_reply\_t fipa\_comm\_reply\_s**

Definition at line 92 of file fipa\_comm.h.

**5.35.2.15 typedef struct fipa\_comm\_t fipa\_comm\_s**

Definition at line 192 of file fipa\_comm.h.

**5.35.2.16 typedef struct fipa\_list\_t\* fipa\_list\_p**

Definition at line 141 of file fipa\_comm.h.

**5.35.2.17 typedef struct fipa\_list\_t fipa\_list\_s**

Definition at line 140 of file fipa\_comm.h.

**5.35.2.18 typedef struct fipa\_comm\_t\* MCFIPAComm\_p**

Definition at line 195 of file fipa\_comm.h.

### 5.35.2.19 typedef struct fipa\_comm\_t MCFIPAComm\_s

Definition at line 194 of file fipa\_comm.h.

### 5.35.2.20 typedef int(\* protocol\_contract\_net\_callback)(int numBids, char \*\*bids, char \*winners)

Definition at line 18 of file fipa\_comm.h.

## 5.35.3 Enumeration Type Documentation

### 5.35.3.1 enum fipa\_comm\_action\_type\_e

Enumerator:

*FIPA\_COMM\_ACTION\_ERROR*  
*FIPA\_COMM\_ACTION\_BOTH*  
*FIPA\_COMM\_ACTION\_REPLY*  
*FIPA\_COMM\_ACTION\_CALLBACK*  
*FIPA\_COMM\_ACTION\_NONE*

Definition at line 45 of file fipa\_comm.h.

### 5.35.3.2 enum fipa\_comm\_check\_type\_e

Enumerator:

*FIPA\_COMM\_CHECK\_ERROR*  
*FIPA\_COMM\_CHECK\_SENDER\_NAME*  
*FIPA\_COMM\_CHECK\_SENDER\_ADDRESS*  
*FIPA\_COMM\_CHECK\_CONTENT*  
*FIPA\_COMM\_CHECK\_CONVERSATIONID*  
*FIPA\_COMM\_CHECK\_LANGUAGE*  
*FIPA\_COMM\_CHECK\_ONTOLOGY*  
*FIPA\_COMM\_CHECK\_ENCODING*

Definition at line 34 of file fipa\_comm.h.

### 5.35.3.3 enum fipa\_comm\_logic\_type\_e

Enumerator:

*FIPA\_COMM\_LOGIC\_ERROR*  
*FIPA\_COMM\_LOGIC\_OR*  
*FIPA\_COMM\_LOGIC\_AND*

Definition at line 28 of file fipa\_comm.h.

#### 5.35.3.4 enum fipa\_comm\_protocol\_cn\_state\_e

Enumerator:

*FIPA\_COMM\_PROTOCOL\_CN\_ERROR*

*FIPA\_COMM\_PROTOCOL\_CN\_START*

*FIPA\_COMM\_PROTOCOL\_CN\_CFP*

*FIPA\_COMM\_PROTOCOL\_CN\_BID*

*FIPA\_COMM\_PROTOCOL\_CN\_RESULT*

*FIPA\_COMM\_PROTOCOL\_CN\_DONE*

Definition at line 127 of file fipa\_comm.h.

#### 5.35.3.5 enum fipa\_comm\_type\_e

Enumerator:

*FIPA\_COMM\_ACTION*

*FIPA\_COMM\_CHECK*

*FIPA\_COMM\_REPLY*

*FIPA\_COMM\_CALLBACK*

*FIPA\_COMM\_END*

Definition at line 20 of file fipa\_comm.h.



### 5.35.4 Function Documentation

- 5.35.4.1 void fipa\_comm\_protocol\_info\_COPY (fipa\_comm\_protocol\_p *proto*, void \* *info*)
- 5.35.4.2 int fipa\_list\_Add (fipa\_list\_p \* *list*, fipa\_acl\_message\_p *acl*)
- 5.35.4.3 fipa\_comm\_protocol\_p mc\_FIPAComm\_AddNewProtocol (fipa\_comm\_p *fcomm*, enum fipa\_protocol\_e *type*, char \* *id*)
- 5.35.4.4 int mc\_FIPAComm\_DeregisterPerformative (fipa\_comm\_p *fcomm*, enum fipa\_performative\_e *type*)
- 5.35.4.5 void mc\_FIPAComm\_Destroy (fipa\_comm\_p)
- 5.35.4.6 int mc\_FIPAComm\_FSM\_CN (fipa\_comm\_p *fcomm*, fipa\_acl\_message\_p *acl*)
- 5.35.4.7 int mc\_FIPAComm\_HandleMessage (fipa\_comm\_p *fcomm*, fipa\_acl\_message\_p *acl*)
- 5.35.4.8 int mc\_FIPAComm\_HandleMessageOnce (fipa\_comm\_p *fcomm*)
- 5.35.4.9 int mc\_FIPAComm\_HandleMessageWait (fipa\_comm\_p *fcomm*)
- 5.35.4.10 int mc\_FIPAComm\_HandleProtocol (fipa\_comm\_p *fcomm*, fipa\_acl\_message\_p *acl*)
- 5.35.4.11 fipa\_comm\_p mc\_FIPAComm\_New ()
- 5.35.4.12 fipa\_comm\_protocol\_p mc\_FIPAComm\_NewProtocol (enum fipa\_protocol\_e *type*, char \* *id*)
- 5.35.4.13 int mc\_FIPAComm\_RegisterAction (fipa\_comm\_p *fcomm*, enum fipa\_performative\_e *pType*, enum fipa\_comm\_action\_type\_e *aType*, ...)
- 5.35.4.14 int mc\_FIPAComm\_RegisterActionS (fipa\_performative\_e *type*, fipa\_comm\_action\_p *act*)
- 5.35.4.15 int mc\_FIPAComm\_RegisterPerformative (fipa\_comm\_p *fcomm*, enum fipa\_performative\_e *type*)
- 5.35.4.16 int mc\_FIPAComm\_RegisterProtocol (fipa\_comm\_p *fcomm*, enum fipa\_protocol\_e *protocolType*, char \* *protocolID*, void \* *info*, char \* *network*)
- 5.35.4.17 int mc\_FIPAComm\_RegisterProtocolCallback (fipa\_comm\_p *fcomm*, enum fipa\_protocol\_e *protocolType*, fipa\_comm\_callback *func*)
- 5.35.4.18 int mc\_FIPAComm\_SetAgent (fipa\_comm\_p *fcomm*, void \* *agent*, char \* *agent\_name*, char \* *agent\_address*)
- 5.35.4.19 int mc\_FIPAComm\_SetDefaultReply (fipa\_comm\_p *fcomm*, enum fipa\_performative\_e *type*, char \* *content*)
- 5.35.4.20 int mc\_FIPAComm\_SetPerformativeDefaultReply (fipa\_comm\_p *fcomm*, enum fipa\_performative\_e *ptype*, enum fipa\_performative\_e *rtype*, char \* *content*)
- 5.35.4.21 int mc\_FIPAComm\_SetProtocolState (fipa\_comm\_protocol\_p *protocol*, char *state*)
- 5.35.4.22 int mc\_FIPAComm\_StartProtocol (fipa\_comm\_p *fcomm*, enum fipa\_protocol\_e *type*, char \* *id*)
- 5.35.4.23 fipa\_comm\_protocol\_p mc\_FIPAProtocol\_New ()



## 5.36 /home/dko/Projects/mobilec/src/include/host\_id.h File Reference

### Data Structures

- struct [host\\_id\\_s](#)

### Typedefs

- typedef struct [host\\_id\\_s](#) [host\\_id\\_t](#)
- typedef [host\\_id\\_t](#) \* [host\\_id\\_p](#)

### Functions

- [host\\_id\\_p](#) [host\\_id\\_Initialize](#) (const char \*hostname, [int](#) port)
- void [host\\_id\\_Destroy](#) ([host\\_id\\_p](#) host\_id)
- [int](#) [host\\_id\\_Compare](#) ([host\\_id\\_p](#) host1, [host\\_id\\_p](#) host2)

#### 5.36.1 Typedef Documentation

##### 5.36.1.1 typedef [host\\_id\\_t](#)\* [host\\_id\\_p](#)

Definition at line 43 of file [host\\_id.h](#).

##### 5.36.1.2 typedef struct [host\\_id\\_s](#) [host\\_id\\_t](#)

#### 5.36.2 Function Documentation

##### 5.36.2.1 [int](#) [host\\_id\\_Compare](#) ([host\\_id\\_p](#) *host1*, [host\\_id\\_p](#) *host2*)

##### 5.36.2.2 void [host\\_id\\_Destroy](#) ([host\\_id\\_p](#) *host\_id*)

##### 5.36.2.3 [host\\_id\\_p](#) [host\\_id\\_Initialize](#) (const char \* *hostname*, [int](#) *port*)

## 5.37 /home/dko/Projects/mobilec/src/include/interpreter\_variable\_data.h File Reference

```
#include <ch.h>
```

### Data Structures

- struct [interpreter\\_variable\\_data\\_s](#)

### Typedefs

- typedef struct [interpreter\\_variable\\_data\\_s](#) [interpreter\\_variable\\_data\\_t](#)
- typedef [interpreter\\_variable\\_data\\_t](#) \* [interpreter\\_variable\\_data\\_p](#)

### Functions

- [interpreter\\_variable\\_data\\_p](#) [interpreter\\_variable\\_data\\_New](#) (void)
- [interpreter\\_variable\\_data\\_p](#) [interpreter\\_variable\\_data\\_Copy](#) ([interpreter\\_variable\\_data\\_p](#) src)
- [interpreter\\_variable\\_data\\_p](#) [interpreter\\_variable\\_data\\_InitializeFromAgent](#) (struct [agent\\_s](#) \*agent)
- [interpreter\\_variable\\_data\\_p](#) [interpreter\\_variable\\_data\\_Initialize](#) (struct [agent\\_s](#) \*agent, const char \*varname)
- int [interpreter\\_variable\\_data\\_Destroy](#) ([interpreter\\_variable\\_data\\_p](#) agent\_return\_data)

#### 5.37.1 Typedef Documentation

##### 5.37.1.1 typedef [interpreter\\_variable\\_data\\_t](#)\* [interpreter\\_variable\\_data\\_p](#)

Definition at line 51 of file [interpreter\\_variable\\_data.h](#).

##### 5.37.1.2 typedef struct [interpreter\\_variable\\_data\\_s](#) [interpreter\\_variable\\_data\\_t](#)

Definition at line 50 of file [interpreter\\_variable\\_data.h](#).

#### 5.37.2 Function Documentation

##### 5.37.2.1 [interpreter\\_variable\\_data\\_p](#) [interpreter\\_variable\\_data\\_Copy](#) ([interpreter\\_variable\\_data\\_p](#) src)

Definition at line 235 of file [agent\\_return\\_data.c](#).

References [interpreter\\_variable\\_data\\_s::array\\_dim](#), [interpreter\\_variable\\_data\\_s::array\\_extent](#), [interpreter\\_variable\\_data\\_s::data](#), [interpreter\\_variable\\_data\\_s::data\\_type](#), [interpreter\\_variable\\_data\\_New\(\)](#), [interpreter\\_variable\\_data\\_s::name](#), [interpreter\\_variable\\_data\\_s::size](#), and [strdup](#).

Referenced by [agent\\_task\\_Copy\(\)](#).

**5.37.2.2 int interpreter\_variable\_data\_Destroy (interpreter\_variable\_data\_p agent\_return\_data)**

Definition at line 216 of file agent\_return\_data.c.

References interpreter\_variable\_data\_s::array\_extent, interpreter\_variable\_data\_s::data, MC\_SUCCESS, and interpreter\_variable\_data\_s::name.

Referenced by agent\_RunChScriptThread(), and agent\_task\_Destroy().

**5.37.2.3 interpreter\_variable\_data\_p interpreter\_variable\_data\_Initialize (struct agent\_s \* agent, const char \* varname)****5.37.2.4 interpreter\_variable\_data\_p interpreter\_variable\_data\_InitializeFromAgent (struct agent\_s \* agent)****5.37.2.5 interpreter\_variable\_data\_p interpreter\_variable\_data\_New (void)**

Definition at line 46 of file agent\_return\_data.c.

References interpreter\_variable\_data\_s::array\_dim, interpreter\_variable\_data\_s::array\_extent, CHECK\_NULL, interpreter\_variable\_data\_s::data, interpreter\_variable\_data\_s::data\_type, interpreter\_variable\_data\_s::name, and interpreter\_variable\_data\_s::size.

Referenced by agent\_xml\_parse\_\_data(), interpreter\_variable\_data\_Copy(), and MC\_SaveData\_chdl().

## 5.38 /home/dko/Projects/mobilec/src/include/libmc.h File Reference

MobileC api header file.

```
#include "macros.h"
#include <embedch.h>
```

### Data Structures

- struct [agency\\_s](#)  
*The agency handle.*
- struct [MCAgencyOptions\\_s](#)  
*User modifiable agency options.*
- struct [stationary\\_agent\\_info\\_s](#)

### Defines

- #define [EXPORTMC](#)
- #define [MC\\_Wait](#)(arg1) MC\_MainLoop(arg1)
- #define [\\_ERROR\\_CODE\\_E\\_](#)
- #define [AGENT\\_T](#)
- #define [\\_FIPA\\_PERFORMATIVE\\_E\\_](#)

### Typedefs

- typedef [int](#)(\* [MC\\_AgentInitCallbackFunc\\_t](#))(ChInterp\_t interp, struct [agent\\_s](#) \*agent, void \*user\_data)
- typedef enum [error\\_code\\_e](#) [error\\_code\\_t](#)
- typedef enum [MC\\_SteerCommand\\_e](#) [MC\\_SteerCommand\\_t](#)  
*Available commands for MC\_Steer.*
- typedef struct [agency\\_s](#) [agency\\_t](#)  
*The agency handle.*
- typedef [agency\\_t](#) \* [agency\\_p](#)
- typedef [agency\\_p](#) [MCAgency\\_t](#)
- typedef struct [MCAgencyOptions\\_s](#) [MCAgencyOptions\\_t](#)  
*User modifiable agency options.*
- typedef struct [stationary\\_agent\\_info\\_s](#) [stationary\\_agent\\_info\\_t](#)
- typedef struct [agent\\_s](#) [agent\\_t](#)
- typedef [agent\\_t](#) \* [MCAgent\\_t](#)
- typedef [agent\\_t](#) \* [agent\\_p](#)

## Enumerations

- enum `error_code_e` {  
`MC_SUCCESS = 0, MC_ERR, MC_ERR_CONNECT, MC_ERR_PARSE,`  
`MC_ERR_EMPTY, MC_ERR_INVALID, MC_ERR_INVALID_ARGS, MC_ERR_NOT_FOUND,`  
`MC_ERR_MEMORY, MC_ERR_SEND, MC_WARN_DUPLICATE, MC_SUCCESS = 0,`  
`ERR, MC_ERR_CONNECT, MC_ERR_PARSE, MC_ERR_EMPTY,`  
`MC_ERR_INVALID, MC_ERR_INVALID_ARGS, MC_ERR_NOT_FOUND, MC_ERR-`  
`MEMORY,`  
`MC_ERR_SEND, MC_WARN_DUPLICATE }`
- enum `MC_ThreadIndex_e` {  
`MC_THREAD_DF = 0, MC_THREAD_AMS, MC_THREAD_ACC, MC_THREAD_CP,`  
`MC_THREAD_AGENT, MC_THREAD_ALL }`  
*MobileC Module indices.*
- enum `MC_SteerCommand_e` { `MC_RUN = 0, MC_SUSPEND, MC_RESTART, MC_STOP }`  
*Available commands for MC\_Steer.*
- enum `MC_Signal_e` {  
`MC_NO_SIGNAL = 0x00, MC_RECV_CONNECTION = 0x01, MC_RECV_MESSAGE = 0x02,`  
`MC_RECV_AGENT = 0x04,`  
`MC_RECV_RETURN = 0x08, MC_EXEC_AGENT = 0x10, MC_ALL_SIGNALS = 0x20 }`  
*MobileC system signals.*
- enum `MC_AgentType_e` { `MC_NONE = -1, MC_REMOTE_AGENT = 0, MC_LOCAL_AGENT,`  
`MC_RETURN_AGENT }`
- enum `MC_AgentStatus_e` {  
`MC_NO_STATUS = -1, MC_WAIT_CH = 0, MC_WAIT_MESSGSEND, MC_AGENT_ACTIVE,`  
`MC_AGENT_NEUTRAL, MC_AGENT_SUSPENDED, MC_WAIT_FINISHED }`  
*An agent's current execution state.*
- enum `fipa_performative_e` {  
`FIPA_ERROR = -1, FIPA_ZERO, FIPA_ACCEPT_PROPOSAL, FIPA_AGREE,`  
`FIPA_CANCEL, FIPA_CALL_FOR_PROPOSAL, FIPA_CONFIRM, FIPA_DISCONFIRM,`  
`FIPA_FAILURE, FIPA_INFORM, FIPA_INFORM_IF, FIPA_INFORM_REF,`  
`FIPA_NOT_UNDERSTOOD, FIPA_PROPOGATE, FIPA_PROPOSE, FIPA_PROXY,`  
`FIPA_QUERY_IF, FIPA_QUERY_REF, FIPA_REFUSE, FIPA_REJECT_PROPOSAL,`  
`FIPA_REQUEST, FIPA_REQUEST_WHEN, FIPA_REQUEST_WHENEVER, FIPA-`  
`SUBSCRIBE,`  
`FIPA_ERROR = -1, FIPA_ZERO, FIPA_ACCEPT_PROPOSAL, FIPA_AGREE,`  
`FIPA_CANCEL, FIPA_CALL_FOR_PROPOSAL, FIPA_CONFIRM, FIPA_DISCONFIRM,`  
`FIPA_FAILURE, FIPA_INFORM, FIPA_INFORM_IF, FIPA_INFORM_REF,`  
`FIPA_NOT_UNDERSTOOD, FIPA_PROPOGATE, FIPA_PROPOSE, FIPA_PROXY,`  
`FIPA_QUERY_IF, FIPA_QUERY_REF, FIPA_REFUSE, FIPA_REJECT_PROPOSAL,`  
`FIPA_REQUEST, FIPA_REQUEST_WHEN, FIPA_REQUEST_WHENEVER, FIPA-`  
`SUBSCRIBE }`

## Functions

- EXPORTMC [int MC\\_AclDestroy](#) (struct [fipa\\_acl\\_message\\_s](#) \*message)  
*Destroy a FIPA ACL message.*
- EXPORTMC struct [fipa\\_acl\\_message\\_s](#) \* [MC\\_AclNew](#) (void)  
*Allocate a new ACL Message.*
- EXPORTMC [int MC\\_AclPost](#) ([MCAgent\\_t](#) agent, struct [fipa\\_acl\\_message\\_s](#) \*message)  
*Post ACL message to agent.*
- EXPORTMC struct [fipa\\_acl\\_message\\_s](#) \* [MC\\_AclReply](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl\_message)  
*Reply to an ACL message.*
- EXPORTMC struct [fipa\\_acl\\_message\\_s](#) \* [MC\\_AclRetrieve](#) ([MCAgent\\_t](#) agent)  
*Retrieve an ACL message.*
- EXPORTMC [int MC\\_AclSend](#) ([MCAgency\\_t](#) attr, struct [fipa\\_acl\\_message\\_s](#) \*acl)  
*Send a composed ACL Message.*
- EXPORTMC struct [fipa\\_acl\\_message\\_s](#) \* [MC\\_AclWaitRetrieve](#) ([MCAgent\\_t](#) agent)  
*Wait for and retrieve an ACL message.*
- EXPORTMC enum [fipa\\_protocol\\_e](#) [MC\\_AclGetProtocol](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl)
- EXPORTMC char \* [MC\\_AclGetConversationID](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl)
- EXPORTMC enum [fipa\\_performative\\_e](#) [MC\\_AclGetPerformative](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl)
- EXPORTMC [int MC\\_AclGetSender](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl, char \*\*name, char \*\*address)
- EXPORTMC const char \* [MC\\_AclGetContent](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl)
- EXPORTMC [int MC\\_AclSetProtocol](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl, enum [fipa\\_protocol\\_e](#) performative)
- EXPORTMC [int MC\\_AclSetConversationID](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl, char \*id)
- EXPORTMC [int MC\\_AclSetPerformative](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl, enum [fipa\\_performative\\_e](#) performative)
- EXPORTMC [int MC\\_AclSetSender](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl, const char \*name, const char \*address)
- EXPORTMC [int MC\\_AclAddReceiver](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl, const char \*name, const char \*address)
- EXPORTMC [int MC\\_AclAddReplyTo](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl, const char \*name, const char \*address)
- EXPORTMC [int MC\\_AclSetContent](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl, const char \*content)
- EXPORTMC [int MC\\_AddAgent](#) ([MCAgency\\_t](#) attr, [MCAgent\\_t](#) agent)  
*Add an agent to the agency 'attr'.*
- [int MC\\_AddStationaryAgent](#) ([MCAgency\\_t](#) agency, void \*(\*agent\_thread)([stationary\\_agent\\_info\\_t](#) \*), const char \*name, void \*agent\_args)  
*Add a stationary agent to the agency.*
- [int MC\\_AddAgentInitCallback](#) ([MCAgency\\_t](#) agency, [MC\\_AgentInitCallbackFunc\\_t](#) function, void \*user\_data)  
*Add an agent initialization callback function.*

- [int MC\\_AgentAddTask](#) ([MCAgent\\_t](#) agent, const char \*code, const char \*return\_var\_name, const char \*server, [int](#) persistent)  
*Add a new task to an already existing agent.*
- [int MC\\_AgentAddTaskFromFile](#) ([MCAgent\\_t](#) agent, const char \*filename, const char \*return\_var\_name, const char \*server, [int](#) persistent)  
*Add a new task to an already existing agent.*
- [MCAgency\\_t MC\\_AgentInfo\\_GetAgency](#) ([stationary\\_agent\\_info\\_t](#) \*stationary\_agent\_info)  
*Get the host Mobile-C agency handle from a "stationary\_agent\_info\_t" data structure.*
- [MCAgent\\_t MC\\_AgentInfo\\_GetAgent](#) ([stationary\\_agent\\_info\\_t](#) \*stationary\_agent\_info)  
*Retrieve an agent handle from the stationary agent info from within a stationary agent.*
- void \* [MC\\_AgentInfo\\_GetAgentArgs](#) ([stationary\\_agent\\_info\\_t](#) \*stationary\_agent\_info)  
*Retrieve the agent argument supplied as the "agent\_args" parameter in the [MC\\_AddStationaryAgent\(\)](#) function.*
- [int MC\\_AgentReturnArrayDim](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Get the dimension of an agent's return data, if it is an array.*
- [int MC\\_AgentReturnArrayExtent](#) ([MCAgent\\_t](#) agent, [int](#) task\_num, [int](#) index)  
*Get the extent of a dimension of an agent's return data, if it is an array.*
- [int MC\\_AgentReturnArrayNum](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Get the total number of elements of the agent return data.*
- const void \* [MC\\_AgentReturnDataGetSymbolAddr](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Get a pointer to the agent's return data.*
- [size\\_t MC\\_AgentReturnDataSize](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Get the size of an agent's return data type.*
- [int MC\\_AgentReturnDataType](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Get the type of an agent's return data.*
- [int MC\\_AgentReturnIsArray](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Check to see whether or not an agent's return data is an array.*
- const void \* [MC\\_AgentVariableRetrieve](#) ([MCAgent\\_t](#) agent, const char \*var\_name, [int](#) task\_num)  
*Retrieve a pointer to a previously saved variable.*
- [int MC\\_AgentVariableRetrieveInfo](#) ([MCAgent\\_t](#) agent, const char \*var\_name, [int](#) task\_num, const void \*\*data, [int](#) \*dim, const [int](#) \*\*extent)  
*Retrieve a info about a previously saved variable.*
- [int MC\\_AgentVariableSave](#) ([MCAgent\\_t](#) agent, const char \*var\_name)  
*Mark an agent variable for saving.*

- EXPORTMC [int MC\\_Barrier](#) ([MCAgency\\_t](#) attr, [int](#) id)
- EXPORTMC [int MC\\_BarrierDelete](#) ([MCAgency\\_t](#) attr, [int](#) id)  
*Find and delete an initialized MobileC Barrier.*
- EXPORTMC [int MC\\_BarrierInit](#) ([MCAgency\\_t](#) attr, [int](#) id, [int](#) num\_procs)  
*Initialize a MobileC Barrier.*
- EXPORTMC [int MC\\_CallAgentFunc](#) ([MCAgent\\_t](#) agent, const char \*funcName, void \*returnVal, [int](#) numArgs,...)  
*Use custom ChOptions\_t type for internal Ch interpreter.*
- EXPORTMC [int MC\\_CallAgentFuncArg](#) ([MCAgent\\_t](#) agent, const char \*funcName, void \*returnVal, void \*arg)  
*Calls a function defined in an agent.*
- EXPORTMC [int MC\\_CallAgentFuncV](#) ([MCAgent\\_t](#) agent, const char \*funcName, void \*returnVal, va\_list ap)  
*Calls a function defined in an agent.*
- EXPORTMC [int MC\\_CallAgentFuncVar](#) ([MCAgent\\_t](#) agent, const char \*funcName, void \*returnVal, ChVaList\_t arglist)
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgent](#) (const char \*name, const char \*home, const char \*owner, const char \*code, const char \*return\_var\_name, const char \*server, [int](#) persistent)  
*Compose a new agent dynamically without using a prewritten XML file.*
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgentS](#) (const char \*name, const char \*home, const char \*owner, const char \*code, const char \*return\_var\_name, const char \*server, [int](#) persistent, const char \*workgroup\_code)  
*Compose a new agent dynamically without using a prewritten XML file.*
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgentWithWorkgroup](#) (const char \*name, const char \*home, const char \*owner, const char \*code, const char \*return\_var\_name, const char \*server, [int](#) persistent, const char \*workgroup\_code)  
*Compose a new agent dynamically without using a prewritten XML file.*
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgentFromFile](#) (const char \*name, const char \*home, const char \*owner, const char \*filename, const char \*return\_var\_name, const char \*server, [int](#) persistent)  
*Compose a new agent dynamically from a source code file.*
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgentFromFileS](#) (const char \*name, const char \*home, const char \*owner, const char \*filename, const char \*return\_var\_name, const char \*server, [int](#) persistent, const char \*workgroup\_code)  
*Compose a new agent dynamically from a source code file.*
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgentFromFileWithWorkgroup](#) (const char \*name, const char \*home, const char \*owner, const char \*filename, const char \*return\_var\_name, const char \*server, [int](#) persistent, const char \*workgroup\_code)  
*Compose a new agent dynamically from a source code file.*
- EXPORTMC [int MC\\_CondBroadcast](#) ([MCAgency\\_t](#) attr, [int](#) id)  
*Wakes up all agents/threads waiting on a condition variable.*



- EXPORTMC [int MC\\_CondSignal](#) ([MCAgency\\_t](#) attr, [int](#) id)  
*Wakes up at least one thread waiting on a condition variable.*
- EXPORTMC [int MC\\_CondReset](#) ([MCAgency\\_t](#) attr, [int](#) id)  
*Reset a previously signalled MobileC condition variable.*
- EXPORTMC [int MC\\_CondWait](#) ([MCAgency\\_t](#) attr, [int](#) id)  
*Wait on a MobileC synchronization variable.*
- [int MC\\_CopyAgent](#) ([MCAgent\\_t](#) \*agent\_out, const [MCAgent\\_t](#) agent\_in)  
*Performs a deep-copy of an agent structure.*
- EXPORTMC [int MC\\_DeleteAgent](#) ([MCAgent\\_t](#) agent)  
*Stop and remove an agent.*
- EXPORTMC [int MC\\_DeleteAgentWG](#) ([MCAgent\\_t](#) calling\_agent, [MCAgent\\_t](#) agent)  
*Stop and remove an agent in the same workgroup.*
- EXPORTMC [int MC\\_End](#) ([MCAgency\\_t](#) attr)  
*End an agency.*
- [int MC\\_DestroyServiceSearchResult](#) (char \*\*agentName, char \*\*serviceName, [int](#) \*agentID, [int](#) numResult)  
*Free memory allocated by a Service Search operation.*
- EXPORTMC [MCAgent\\_t MC\\_FindAgentByName](#) ([MCAgency\\_t](#) attr, const char \*name)  
*Find an agent by its name.*
- EXPORTMC [MCAgent\\_t MC\\_FindAgentByID](#) ([MCAgency\\_t](#) attr, [int](#) ID)  
*Find an agent by its id.*
- EXPORTMC void \* [MC\\_GetAgentExecEngine](#) ([MCAgent\\_t](#) agent)  
*Retrieve an agent's Ch interpreter.*
- EXPORTMC [int MC\\_GetAgentID](#) ([MCAgent\\_t](#) agent)  
*Retrieve an agent's id.*
- EXPORTMC char \* [MC\\_GetAgentName](#) ([MCAgent\\_t](#) agent)
- EXPORTMC [int MC\\_GetAgentNumTasks](#) ([MCAgent\\_t](#) agent)  
*Retrive the number of tasks an agent has.*
- EXPORTMC [int MC\\_GetAgentReturnData](#) ([MCAgent\\_t](#) agent, [int](#) task\_num, void \*\*data, [int](#) \*dim, [int](#) \*\*extent)  
*Get an agent's return data.*
- EXPORTMC [int MC\\_GetAgentStatus](#) ([MCAgent\\_t](#) agent)  
*Get an agent's current status.*
- EXPORTMC enum [MC\\_AgentType\\_e MC\\_GetAgentType](#) ([MCAgent\\_t](#) agent)

*Get an agent's type.*

- EXPORTMC char \* [MC\\_GetAgentXMLString](#) ([MCAgent\\_t](#) agent)  
*Get an agent's xml string.*
- EXPORTMC int [MC\\_HaltAgency](#) ([MCAgency\\_t](#) agency)  
*Halt an agency: Do not process new entries in queues.*
- EXPORTMC [MCAgency\\_t](#) [MC\\_Initialize](#) (int port, [MCAgencyOptions\\_t](#) \*options)  
*Initialize and start a MobileC agency.*
- EXPORTMC int [MC\\_InitializeAgencyOptions](#) (struct [MCAgencyOptions\\_s](#) \*options)  
*Initialize MobileC options.*
- EXPORTMC int [MC\\_MainLoop](#) ([MCAgency\\_t](#) attr)  
*Wait indefinitely.*
- EXPORTMC int [MC\\_LoadAgentFromFile](#) ([MCAgency\\_t](#) attr, const char \*filename)  
*Load an agent from a file into an agency.*
- EXPORTMC int [MC\\_MigrateAgent](#) ([MCAgent\\_t](#) agent, const char \*hostname, int port)  
*Migrates a running agent to another host.*
- EXPORTMC int [MC\\_MutexLock](#) ([MCAgency\\_t](#) attr, int id)  
*Locks a MobileC synchronization variable as a mutex.*
- EXPORTMC int [MC\\_MutexUnlock](#) ([MCAgency\\_t](#) attr, int id)
- EXPORTMC int [MC\\_RegisterService](#) ([MCAgency\\_t](#) agency, [MCAgent\\_t](#) agent, int agentID, const char \*agentName, char \*\*serviceNames, int numServices)  
*Register a new service with the Directory Facilitator.*
- EXPORTMC int [MC\\_ResumeAgency](#) ([MCAgency\\_t](#) agency)  
*Resumes a halted agency.*
- EXPORTMC [MCAgent\\_t](#) [MC\\_RetrieveAgent](#) ([MCAgency\\_t](#) attr)  
*Retrieves the oldest agent from an agency.*
- EXPORTMC int [MC\\_SemaphorePost](#) ([MCAgency\\_t](#) attr, int id)  
*Post to a MobileC synchronization variable semaphore.*
- EXPORTMC int [MC\\_SemaphoreWait](#) ([MCAgency\\_t](#) attr, int id)  
*Decreases a MobileC synchronization variable semaphore count by one.*
- EXPORTMC int [MC\\_SetDefaultAgentStatus](#) ([MCAgency\\_t](#) agency, enum [MC\\_AgentStatus\\_e](#) status)  
*Sets default incoming agent status.*
- EXPORTMC int [MC\\_SetThreadOn](#) ([MCAgencyOptions\\_t](#) \*options, enum [MC\\_ThreadIndex\\_e](#) index)  
*Sets a MobileC thread to "on" status.*

- EXPORTMC [int MC\\_SetThreadsAllOn](#) (MCAgencyOptions\_t \*options)  
*Set all Mobile-C threads on.*
- EXPORTMC [int MC\\_SetThreadOff](#) (MCAgencyOptions\_t \*options, enum [MC\\_ThreadIndex\\_e](#) index)  
*Sets a MobileC thread to "off" status.*
- EXPORTMC [int MC\\_SetThreadsAllOff](#) (MCAgencyOptions\_t \*options)  
*Set all MobileC threads to 'off' status.*
- EXPORTMC [int MC\\_PrintAgentCode](#) (MCAgent\_t agent)  
*Prints an agents code to stdout.*
- EXPORTMC [char \\* MC\\_RetrieveAgentCode](#) (MCAgent\_t agent)  
*Retrieves an agent's Ch code.*
- EXPORTMC [int MC\\_ResetSignal](#) (MCAgency\_t attr)  
*Reset a MobileC signal.*
- EXPORTMC [int MC\\_SearchForService](#) (MCAgency\_t attr, const char \*searchString, char \*\*\*agentNames, char \*\*\*serviceNames, [int \\*\\*agentIDs](#), [int \\*numResults](#))  
*Search the directory facilitator for a service.*
- EXPORTMC [int MC\\_SendAgent](#) (MCAgency\_t attr, const char \*message)  
*Sends an agent migration message.*
- EXPORTMC [int MC\\_SendAgentFile](#) (MCAgency\_t attr, const char \*filename)  
*Sends an agent migration message.*
- EXPORTMC [int MC\\_SendAgentMigrationMessage](#) (MCAgency\_t attr, const char \*message, const char \*hostname, [int port](#))  
*Sends an agent migration message.*
- EXPORTMC [int MC\\_SendAgentMigrationMessageFile](#) (MCAgency\_t attr, const char \*filename, const char \*hostname, [int port](#))  
*Sends an agent migration message.*
- EXPORTMC [int MC\\_SetAgentStatus](#) (MCAgent\_t agent, enum [MC\\_AgentStatus\\_e](#) status)  
*Set an agent's status.*
- EXPORTMC [int MC\\_Steer](#) (MCAgency\_t attr, [int>\(\\*funcptr\)\(void \\*data\)](#), void \*arg)  
*Set up a steerable algorithm.*
- EXPORTMC [enum MC\\_SteerCommand\\_e MC\\_SteerControl](#) (void)  
*The MobileC user-algorithm steering function.*
- EXPORTMC [int MC\\_SyncDelete](#) (MCAgency\_t attr, [int id](#))  
*Deletes a previously initialized synchronization variable.*

- EXPORTMC [int MC\\_SyncInit](#) ([MCAgency\\_t](#) attr, [int](#) id)  
*Initializes a new MobileC synchronization variable.*
- EXPORTMC [int MC\\_TerminateAgent](#) ([MCAgent\\_t](#) agent)  
*Halt a running agent.*
- EXPORTMC [int MC\\_TerminateAgentWG](#) ([MCAgent\\_t](#) calling\_agent, [MCAgent\\_t](#) agent)
- EXPORTMC [int MC\\_WaitAgent](#) ([MCAgency\\_t](#) attr)  
*Wait indefinitely.*
- EXPORTMC [MCAgent\\_t MC\\_WaitRetrieveAgent](#) ([MCAgency\\_t](#) attr)  
*Wait and retrieve an agent.*
- EXPORTMC [int MC\\_WaitSignal](#) ([MCAgency\\_t](#) attr, [int](#) signals)  
*Wait for a MobileC signal.*

### 5.38.1 Detailed Description

MobileC api header file.

Definition in file [libmc.h](#).

### 5.38.2 Define Documentation

#### 5.38.2.1 `#define _ERROR_CODE_E_`

Definition at line 162 of file [libmc.h](#).

#### 5.38.2.2 `#define _FIPA_PERFORMATIVE_E_`

Definition at line 304 of file [libmc.h](#).

#### 5.38.2.3 `#define AGENT_T`

Definition at line 280 of file [libmc.h](#).

#### 5.38.2.4 `#define EXPORTMC`

Definition at line 140 of file [libmc.h](#).

#### 5.38.2.5 `#define MC_Wait(arg1) MC_MainLoop(arg1)`

Definition at line 145 of file [libmc.h](#).

### 5.38.3 Typedef Documentation

#### 5.38.3.1 typedef agency\_t\* agency\_p

Definition at line 248 of file libmc.h.

#### 5.38.3.2 typedef struct agency\_s agency\_t

The agency handle.

#### 5.38.3.3 typedef agent\_t\* agent\_p

Definition at line 283 of file libmc.h.

#### 5.38.3.4 typedef struct agent\_s agent\_t

Definition at line 281 of file libmc.h.

#### 5.38.3.5 typedef enum error\_code\_e error\_code\_t

#### 5.38.3.6 typedef int(\* MC\_AgentInitCallbackFunc\_t)(ChInterp\_t interp, struct agent\_s \*agent, void \*user\_data)

Definition at line 156 of file libmc.h.

#### 5.38.3.7 typedef enum MC\_SteerCommand\_e MC\_SteerCommand\_t

Available commands for MC\_Steer.

#### 5.38.3.8 typedef agency\_p MCAgency\_t

Definition at line 249 of file libmc.h.

#### 5.38.3.9 typedef struct MCAgencyOptions\_s MCAgencyOptions\_t

User modifiable agency options.

#### 5.38.3.10 typedef agent\_t\* MCAgent\_t

Definition at line 282 of file libmc.h.

5.38.3.11 typedef struct stationary\_agent\_info\_s stationary\_agent\_info\_t

## 5.38.4 Enumeration Type Documentation

5.38.4.1 enum error\_code\_e

Enumerator:

*MC\_SUCCESS*  
*MC\_ERR*  
*MC\_ERR\_CONNECT*  
*MC\_ERR\_PARSE*  
*MC\_ERR\_EMPTY*  
*MC\_ERR\_INVALID*  
*MC\_ERR\_INVALID\_ARGS*  
*MC\_ERR\_NOT\_FOUND*  
*MC\_ERR\_MEMORY*  
*MC\_ERR\_SEND*  
*MC\_WARN\_DUPLICATE*  
*MC\_SUCCESS*  
*ERR*  
*MC\_ERR\_CONNECT*  
*MC\_ERR\_PARSE*  
*MC\_ERR\_EMPTY*  
*MC\_ERR\_INVALID*  
*MC\_ERR\_INVALID\_ARGS*  
*MC\_ERR\_NOT\_FOUND*  
*MC\_ERR\_MEMORY*  
*MC\_ERR\_SEND*  
*MC\_WARN\_DUPLICATE*

Definition at line 167 of file libmc.h.

5.38.4.2 enum fipa\_performative\_e

Enumerator:

*FIPA\_ERROR*  
*FIPA\_ZERO*  
*FIPA\_ACCEPT\_PROPOSAL*  
*FIPA\_AGREE*  
*FIPA\_CANCEL*  
*FIPA\_CALL\_FOR\_PROPOSAL*  
*FIPA\_CONFIRM*  
*FIPA\_DISCONFIRM*

*FIPA\_FAILURE*  
*FIPA\_INFORM*  
*FIPA\_INFORM\_IF*  
*FIPA\_INFORM\_REF*  
*FIPA\_NOT\_UNDERSTOOD*  
*FIPA\_PROPOGATE*  
*FIPA\_PROPOSE*  
*FIPA\_PROXY*  
*FIPA\_QUERY\_IF*  
*FIPA\_QUERY\_REF*  
*FIPA\_REFUSE*  
*FIPA\_REJECT\_PROPOSAL*  
*FIPA\_REQUEST*  
*FIPA\_REQUEST\_WHEN*  
*FIPA\_REQUEST\_WHENEVER*  
*FIPA\_SUBSCRIBE*  
*FIPA\_ERROR*  
*FIPA\_ZERO*  
*FIPA\_ACCEPT\_PROPOSAL*  
*FIPA\_AGREE*  
*FIPA\_CANCEL*  
*FIPA\_CALL\_FOR\_PROPOSAL*  
*FIPA\_CONFIRM*  
*FIPA\_DISCONFIRM*  
*FIPA\_FAILURE*  
*FIPA\_INFORM*  
*FIPA\_INFORM\_IF*  
*FIPA\_INFORM\_REF*  
*FIPA\_NOT\_UNDERSTOOD*  
*FIPA\_PROPOGATE*  
*FIPA\_PROPOSE*  
*FIPA\_PROXY*  
*FIPA\_QUERY\_IF*  
*FIPA\_QUERY\_REF*  
*FIPA\_REFUSE*  
*FIPA\_REJECT\_PROPOSAL*  
*FIPA\_REQUEST*  
*FIPA\_REQUEST\_WHEN*  
*FIPA\_REQUEST\_WHENEVER*  
*FIPA\_SUBSCRIBE*

Definition at line 305 of file libmc.h.

#### 5.38.4.3 enum MC\_AgentStatus\_e

An agent's current execution state.

**Enumerator:**

*MC\_NO\_STATUS*  
*MC\_WAIT\_CH* Waiting to be started  
*MC\_WAIT\_MESSGSEND* Finished, waiting to migrate  
*MC\_AGENT\_ACTIVE* Running  
*MC\_AGENT\_NEUTRAL* Not running, but do not flush  
*MC\_AGENT\_SUSPENDED* Unused  
*MC\_WAIT\_FINISHED* Finished, waiting to be flushed

Definition at line 291 of file libmc.h.

#### 5.38.4.4 enum MC\_AgentType\_e

**Enumerator:**

*MC\_NONE*  
*MC\_REMOTE\_AGENT*  
*MC\_LOCAL\_AGENT*  
*MC\_RETURN\_AGENT*

Definition at line 286 of file libmc.h.

#### 5.38.4.5 enum MC\_Signal\_e

MobileC system signals.

**Note**

Each signal is activated after the corresponding action. i.e. The 'MC\_RECV\_MESSAGE' signal is activated after a message is received.

**See also**

[MC\\_WaitSignal\(\)](#), [MC\\_ResetSignal](#)

**Enumerator:**

*MC\_NO\_SIGNAL* No signal has been received  
*MC\_RECV\_CONNECTION* A Connection has been received  
*MC\_RECV\_MESSAGE* A Message has been received from a connection  
*MC\_RECV\_AGENT* An normal agent has been parsed from a message  
*MC\_RECV\_RETURN* A return agent has been parsed from a message  
*MC\_EXEC\_AGENT* A normal agent has been executed.  
*MC\_ALL\_SIGNALS* Catch any of the above signals

Definition at line 212 of file libmc.h.



#### 5.38.4.6 enum MC\_SteerCommand\_e

Available commands for MC\_Steer.

##### Enumerator:

*MC\_RUN* Continue the algorithm  
*MC\_SUSPEND* Suspend/pause the algorithm  
*MC\_RESTART* Restart the algorithm from the beginning  
*MC\_STOP* Stop the algorithm

Definition at line 197 of file libmc.h.

#### 5.38.4.7 enum MC\_ThreadIndex\_e

MobileC Module indices.

##### Enumerator:

*MC\_THREAD\_DF* Directory Facilitator  
*MC\_THREAD\_AMS* Agent Managment system  
*MC\_THREAD\_ACC* Agency communications  
*MC\_THREAD\_CP* Command Prompt  
*MC\_THREAD\_AGENT* Agent threads  
*MC\_THREAD\_ALL*

Definition at line 185 of file libmc.h.

### 5.38.5 Function Documentation

#### 5.38.5.1 EXPORTMC int MC\_AclAddReceiver (struct fipa\_acl\_message\_s \* *acl*, const char \* *name*, const char \* *address*)

Definition at line 318 of file libmc.c.

References fipa\_agent\_identifier\_s::addresses, fipa\_agent\_identifier\_New(), fipa\_agent\_identifier\_set\_-New(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_url\_New(), fipa\_url\_sequence\_New(), fipa\_agent\_identifier\_s::name, fipa\_url\_sequence\_s::num, fipa\_agent\_identifier\_set\_s::num, fipa\_acl\_message\_s::receiver, fipa\_acl\_message\_s::receiver\_num, fipa\_url\_s::str, strdup, and fipa\_url\_sequence\_s::urls.

Referenced by MC\_AclAddReceiver\_chdl().

#### 5.38.5.2 EXPORTMC int MC\_AclAddReplyTo (struct fipa\_acl\_message\_s \* *acl*, const char \* *name*, const char \* *address*)

Definition at line 354 of file libmc.c.

References fipa\_agent\_identifier\_s::addresses, fipa\_agent\_identifier\_New(), fipa\_agent\_identifier\_set\_-New(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_url\_New(), fipa\_url\_sequence\_New(), fipa\_agent\_identifier\_s::name, fipa\_url\_sequence\_s::num, fipa\_agent\_identifier\_set\_s::num, fipa\_acl\_message\_s::reply\_to, fipa\_url\_s::str, strdup, and fipa\_url\_sequence\_s::urls.

Referenced by MC\_AclAddReplyTo\_chdl().

### 5.38.5.3 EXPORTMC int MC\_AclDestroy (struct fipa\_acl\_message\_s \* *message*)

Destroy a FIPA ACL message.

#### Parameters

*message* The ACL message to destroy

#### Returns

0 on success, error code on failure.

Definition at line 81 of file libmc.c.

References fipa\_acl\_message\_Destroy().

Referenced by MC\_AclDestroy\_chdl().

### 5.38.5.4 EXPORTMC const char\* MC\_AclGetContent (struct fipa\_acl\_message\_s \* *acl*)

Definition at line 246 of file libmc.c.

References fipa\_string\_s::content, and fipa\_acl\_message\_s::content.

Referenced by MC\_AclGetContent\_chdl().

### 5.38.5.5 EXPORTMC char\* MC\_AclGetConversationID (struct fipa\_acl\_message\_s \* *acl*)

Definition at line 221 of file libmc.c.

References fipa\_string\_s::content, fipa\_expression\_s::content, fipa\_acl\_message\_s::conversation\_id, and fipa\_expression\_s::content\_u::string.

Referenced by MC\_AclGetConversationID\_chdl().

### 5.38.5.6 EXPORTMC enum fipa\_performative\_e MC\_AclGetPerformative (struct fipa\_acl\_message\_s \* *acl*)

Definition at line 227 of file libmc.c.

References fipa\_acl\_message\_s::performative.

Referenced by MC\_AclGetPerformative\_chdl().

### 5.38.5.7 EXPORTMC enum fipa\_protocol\_e MC\_AclGetProtocol (struct fipa\_acl\_message\_s \* *acl*)

Definition at line 215 of file libmc.c.

References fipa\_acl\_message\_s::performative.

Referenced by MC\_AclGetProtocol\_chdl().

### 5.38.5.8 EXPORTMC int MC\_AclGetSender (struct fipa\_acl\_message\_s \* *acl*, char \*\* *name*, char \*\* *address*)

Definition at line 233 of file libmc.c.

References `fipa_agent_identifier_s::addresses`, `fipa_agent_identifier_s::name`, `fipa_acl_message_s::sender`, `fipa_url_s::str`, `strdup`, and `fipa_url_sequence_s::urls`.

Referenced by `MC_AclGetSender_chdl()`.

#### 5.38.5.9 EXPORTMC struct fipa\_acl\_message\_s\* MC\_AclNew (void) [read]

Allocate a new ACL Message.

##### Returns

A newly allocated and empty ACL message.

Definition at line 87 of file `libmc.c`.

References `fipa_acl_message_New()`.

Referenced by `MC_AclNew_chdl()`.

#### 5.38.5.10 EXPORTMC int MC\_AclPost (MCAgent\_t agent, struct fipa\_acl\_message\_s \* message)

Post ACL message to agent.

##### Parameters

*agent* The agent to post the message to

*message* The message to post

##### Returns

0 if successful, or `error_code_t` type.

Definition at line 92 of file `libmc.c`.

References `agent_mailbox_Post()`, and `agent_s::mailbox`.

Referenced by `MC_AclPost_chdl()`, and `MC_AclSend()`.

#### 5.38.5.11 EXPORTMC struct fipa\_acl\_message\_s\* MC\_AclReply (struct fipa\_acl\_message\_s \* acl\_message) [read]

Reply to an ACL message.

##### Parameters

*acl\_message* The incoming acl message to reply to

##### Returns

A newly allocated ACL message

##### Note

This function simply generates a new ACL message with the 'receiver' field automatically set to the 'sender' field of the incoming message.

Definition at line 98 of file libmc.c.

References fipa\_Reply().

Referenced by MC\_AclReply\_chdl().

#### 5.38.5.12 EXPORTMC struct fipa\_acl\_message\_s\* MC\_AclRetrieve (MCAgent\_t *agent*) [read]

Retrieve an ACL message.

##### Parameters

*agent* Agent to retrieve message from.

##### Returns

an ACL message struct on success or NULL on failure

Definition at line 104 of file libmc.c.

References agent\_mailbox\_Retrieve(), and agent\_s::mailbox.

Referenced by MC\_AclRetrieve\_chdl().

#### 5.38.5.13 EXPORTMC int MC\_AclSend (MCAgency\_t *attr*, struct fipa\_acl\_message\_s \* *acl*)

Send a composed ACL Message.

##### Parameters

*attr* An initialized and running MobileC agency

*acl* An allocated and fully composed ACL message.

##### Returns

0 if successful, error code on failure.

Definition at line 110 of file libmc.c.

References fipa\_agent\_identifier\_s::addresses, mtp\_http\_s::content, mtp\_http\_content\_s::content\_type, mtp\_http\_content\_s::data, dynstring\_Destroy(), FIPA\_ACL, fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_envelope\_Compose(), mtp\_http\_s::host, http\_to\_hostport(), MC\_AclPost(), MC\_FindAgentByName(), agency\_s::mc\_platform, dynstring\_s::message, mtp\_http\_s::message\_parts, message\_Send(), message\_s::message\_type, mtp\_http\_CreateMessage(), mtp\_http\_Destroy(), mtp\_http\_New(), fipa\_agent\_identifier\_s::name, fipa\_url\_sequence\_s::num, fipa\_agent\_identifier\_set\_s::num, port, mc\_platform\_s::private\_key, fipa\_acl\_message\_s::receiver, fipa\_url\_s::str, strdup, mtp\_http\_s::target, message\_s::target, and fipa\_url\_sequence\_s::urls.

Referenced by MC\_AclSend\_chdl().

#### 5.38.5.14 EXPORTMC int MC\_AclSetContent (struct fipa\_acl\_message\_s \* *acl*, const char \* *content*)

Definition at line 389 of file libmc.c.

References `fipa_string_s::content`, `fipa_acl_message_s::content`, `fipa_string_Destroy()`, `fipa_string_New()`, and `strdup`.

Referenced by `MC_AclSetContent_chdl()`.

#### 5.38.5.15 EXPORTMC int MC\_AclSetConversationID (struct `fipa_acl_message_s` \* *acl*, char \* *id*)

Definition at line 268 of file `libmc.c`.

References `fipa_string_s::content`, `fipa_expression_s::content`, `fipa_acl_message_s::conversation_id`, `FIPA_EXPR_STRING`, `fipa_expression_New()`, `fipa_string_New()`, `strdup`, `fipa_expression_s::content_u::string`, and `fipa_expression_s::type`.

Referenced by `MC_AclSetConversationID_chdl()`.

#### 5.38.5.16 EXPORTMC int MC\_AclSetPerformative (struct `fipa_acl_message_s` \* *acl*, enum `fipa_performative_e` *performative*)

Definition at line 287 of file `libmc.c`.

References `fipa_acl_message_s::performative`.

Referenced by `MC_AclSetPerformative_chdl()`.

#### 5.38.5.17 EXPORTMC int MC\_AclSetProtocol (struct `fipa_acl_message_s` \* *acl*, enum `fipa_protocol_e` *performative*)

Definition at line 252 of file `libmc.c`.

References `FIPA_PROTOCOL_END`, `FIPA_PROTOCOL_ERROR`, and `fipa_acl_message_s::protocol`.

Referenced by `MC_AclSetProtocol_chdl()`.

#### 5.38.5.18 EXPORTMC int MC\_AclSetSender (struct `fipa_acl_message_s` \* *acl*, const char \* *name*, const char \* *address*)

Definition at line 295 of file `libmc.c`.

References `fipa_agent_identifier_s::addresses`, `fipa_agent_identifier_Destroy()`, `fipa_agent_identifier_New()`, `fipa_url_New()`, `fipa_url_sequence_New()`, `fipa_agent_identifier_s::name`, `fipa_url_sequence_s::num`, `fipa_acl_message_s::sender`, `fipa_url_s::str`, `strdup`, and `fipa_url_sequence_s::urls`.

Referenced by `MC_AclSetSender_chdl()`.

#### 5.38.5.19 EXPORTMC struct `fipa_acl_message_s`\* MC\_AclWaitRetrieve (MCAgent\_t *agent*) [read]

Wait for and retrieve an ACL message.

##### Parameters

*agent* Agent to retrieve message from.

##### Returns

an ACL message struct on success or NULL on failure

Definition at line 208 of file libmc.c.

References agent\_mailbox\_WaitRetrieve(), and agent\_s::mailbox.

Referenced by MC\_AclWaitRetrieve\_chdl().

#### 5.38.5.20 EXPORTMC int MC\_AddAgent (MCAgency\_t *attr*, MCAgent\_t *agent*)

Add an agent to the agency '*attr*'.

##### Parameters

*attr* a MobileC agency

*agent* An initialized MobileC agent

##### Returns

0 if successful, or error\_code\_t type

Definition at line 406 of file libmc.c.

References mc\_platform\_s::agent\_queue, mc\_platform\_s::ams, COND\_SIGNAL, agency\_s::mc\_platform, agent\_s::mc\_platform, MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by handler\_COMPOSE\_SEND(), and MC\_AddAgent\_chdl().

#### 5.38.5.21 int MC\_AddAgentInitCallback (MCAgency\_t *agency*, MC\_AgentInitCallbackFunc\_t *function*, void \* *user\_data*)

Add an agent initialization callback function.

##### Parameters

*agency* a MobileC agency

*function* The callback function to add. This callback function will be called after an interpreter is initialized for an agent but before the agent is executed.

*user\_data* A pointer to any user data that should be made available to the callback function

##### Returns

0 if successful, or error\_code\_t type

Definition at line 447 of file libmc.c.

References agency\_s::agentInitCallback, and agency\_s::agentInitUserData.

#### 5.38.5.22 int MC\_AddStationaryAgent (MCAgency\_t *agency*, void (\*)(stationary\_agent\_info\_t \*) *agent\_thread*, const char \* *name*, void \* *agent\_args*)

Add a stationary agent to the agency.

##### Parameters

*agency* a MobileC agency

*agent\_thread* The binary-space agent function

*name* The new agent's name

*agent\_args* user arguments to pass to the agent

### Returns

0 if successful, or `error_code_t` type

Definition at line 419 of file `libmc.c`.

References `stationary_agent_info_s::agency`, `stationary_agent_info_s::agent`, `agent_NewBinary()`, `mc_platform_s::agent_queue`, `stationary_agent_info_s::args`, `stationary_agent_info_s::attr`, `agency_s::mc_platform`, `agent_s::name`, `strdup`, `stationary_agent_info_s::thread`, and `THREAD_CREATE`.

#### 5.38.5.23 `int MC_AgentAddTask (MCAgent_t agent, const char * code, const char * return_var_name, const char * server, int persistent)`

Add a new task to an already existing agent.

### Parameters

*agent* An existing agent

*code* The C code to use in the new task

*return\_var\_name* The name of the the return variable, or NULL if there is no return variable.

*server* The server to perform the task on.

*persistent* Whether or not the task is persistent. '1' indicates persistent, and '0' indicates not-persistent (the default)

### Returns

Returns a valid agency handle or NULL on failure.

Definition at line 472 of file `libmc.c`.

References `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_task_New()`, `agent_task_s::code_id`, `agent_s::datastate`, `agent_datastate_s::number_of_tasks`, `agent_task_s::persistent`, `agent_task_s::server_name`, `strdup`, `agent_datastate_s::tasks`, and `agent_task_s::var_name`.

Referenced by `MC_AgentAddTask_chdl()`, and `MC_AgentAddTaskFromFile()`.

#### 5.38.5.24 `int MC_AgentAddTaskFromFile (MCAgent_t agent, const char * filename, const char * return_var_name, const char * server, int persistent)`

Add a new task to an already existing agent.

### Parameters

*agent* An existing agent

*filename* The C code to use in the new task

*return\_var\_name* The name of the the return variable, or NULL if there is no return variable.

*server* The server to perform the task on.

***persistent*** Whether or not the task is persistent. '1' indicates persistent, and '0' indicates not-persistent (the default)

### Returns

Returns a valid agency handle or NULL on failure.

Definition at line 545 of file libmc.c.

References MC\_AgentAddTask().

#### 5.38.5.25 MCAgency\_t MC\_AgentInfo\_GetAgency (stationary\_agent\_info\_t \* stationary\_agent\_info)

Get the host Mobile-C agency handle from a "stationary\_agent\_info\_t" data structure.

### Parameters

***stationary\_agent\_info*** A stationary\_agent\_info\_t type variable provided to a stationary agent function

### Returns

Returns a valid agency handle or NULL on failure.

Definition at line 457 of file libmc.c.

References stationary\_agent\_info\_s::agency.

#### 5.38.5.26 MCAgent\_t MC\_AgentInfo\_GetAgent (stationary\_agent\_info\_t \* stationary\_agent\_info)

Retrieve an agent handle from the stationary agent info from within a stationary agent.

### Parameters

***stationary\_agent\_info*** A stationary\_agent\_info\_t type variable provided to a stationary agent function

### Returns

Returns a valid agent handle or NULL on failure.

## 5.38.6 Examples

The following example demonstrates usage of [MC\\_AgentInfo\\_GetAgent\(\)](#)

Definition at line 462 of file libmc.c.

References stationary\_agent\_info\_s::agent.

#### 5.38.6.1 void\* MC\_AgentInfo\_GetAgentArgs (stationary\_agent\_info\_t \* stationary\_agent\_info)

Retrieve the agent argument supplied as the "agent\_args" parameter in the [MC\\_AddStationaryAgent\(\)](#) function.



**Parameters**

*stationary\_agent\_info* A stationary\_agent\_info\_t type variable provided to a stationary agent function

**Returns**

Returns a valid pointer or NULL on failure.

Definition at line 467 of file libmc.c.

References stationary\_agent\_info\_s::args.

**5.38.6.2 int MC\_AgentReturnArrayDim (MCAgent\_t agent, int task\_num)**

Get the dimension of an agent's return data, if it is an array.

**Parameters**

*agent* A return agent

*task* task to retrieve the agent return data size from

**Returns**

Returns a positive integer on success or -1 on failure.

Definition at line 603 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::array\_dim, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

**5.38.6.3 int MC\_AgentReturnArrayExtent (MCAgent\_t agent, int task\_num, int index)**

Get the extent of a dimension of an agent's return data, if it is an array.

**Parameters**

*agent* A return agent

*task* task to retrieve the agent return data size from

*index* The dimension index to get the extent

**Returns**

Returns a positive integer on success or -1 on failure.

Definition at line 579 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::array\_dim, interpreter\_variable\_data\_s::array\_extent, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

**5.38.6.4 int MC\_AgentReturnArrayNum (MCAgent\_t agent, int task\_num)**

Get the total number of elements of the agent return data.

**Parameters**

*agent* A return agent

*task* task to retrieve the agent return data size from

*index* The dimension index to get the extent

**Returns**

Returns a positive integer on success or -1 on failure.

Definition at line 621 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::array\_dim, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

**5.38.6.5 const void\* MC\_AgentReturnDataGetSymbolAddr (MCAgent\_t agent, int task\_num)**

Get a pointer to the agent's return data.

**Parameters**

*agent* A return agent

*task* task to retrieve the agent return data from

**Returns**

Returns a pointer to the beginning of the task's return data on success. Returns NULL on failure.

Definition at line 651 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::data, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

**5.38.6.6 size\_t MC\_AgentReturnDataSize (MCAgent\_t agent, int task\_num)**

Get the size of an agent's return data type.

**Parameters**

*agent* A return agent

*task* task to retrieve the agent return data type size from

**Returns**

Returns a positive integer on success or -1 on failure.

Definition at line 668 of file libmc.c.

References agent\_task\_s::agent\_return\_data, CH\_DATATYPE\_SIZE, interpreter\_variable\_data\_s::data\_type, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, size, and agent\_datastate\_s::tasks.

### 5.38.6.7 int MC\_AgentReturnDataType (MCAgent\_t *agent*, int *task\_num*)

Get the type of an agent's return data.

#### Parameters

*agent* A return agent

*task* task to retrieve the agent return data type from

#### Returns

Returns a positive integer on success or -1 on failure. The return type is of type ChType\_t, defined in ch.h

Definition at line 691 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::data\_type, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

### 5.38.6.8 int MC\_AgentReturnIsArray (MCAgent\_t *agent*, int *task\_num*)

Check to see whether or not an agent's return data is an array.

#### Parameters

*agent* A return agent

*task* task to check to see if the return data is an array.

#### Returns

Returns 1 if the return variable is an array, 0 if it is not an array, or -1 on failure.

Definition at line 701 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::array\_dim, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

### 5.38.6.9 const void\* MC\_AgentVariableRetrieve (MCAgent\_t *agent*, const char \* *var\_name*, int *task\_num*)

Retrieve a pointer to a previously saved variable.

#### Parameters

*agent* A MobileC agent.

*var\_name* The name of the saved variable that has previously been saved.

*task\_num* The previous completed task from which to retrieve the saved variable.

#### Returns

A pointer to the data on success or NULL on failure.

### 5.38.7 Examples

The following example demonstrates usage of [MC\\_AgentVariableRetrieve\(\)](#) from agent space.

Definition at line 722 of file libmc.c.

References `agent_task_s::agent_variable_list`, `interpreter_variable_data_s::data`, `agent_s::datastate`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

Referenced by `MC_AgentVariableRetrieve_chdl()`.

#### 5.38.7.1 `int MC_AgentVariableRetrieveInfo (MCAgent_t agent, const char * var_name, int task_num, const void ** data, int * dim, const int ** extent)`

Retrieve a info about a previously saved variable.

##### Parameters

*agent* A MobileC agent.

*var\_name* The name of the saved variable that has previously been saved.

*task\_num* The previous completed task from which to retrieve the saved variable.

*data* (Output) The Variable Data

*dim* (Output) The dimension of the data array

*extent* (Output) The extents of the output array

##### Returns

Error code.

Definition at line 740 of file libmc.c.

References `agent_task_s::agent_variable_list`, `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `interpreter_variable_data_s::data`, `agent_s::datastate`, `MC_ERR_NOT_FOUND`, `MC_SUCCESS`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

#### 5.38.7.2 `int MC_AgentVariableSave (MCAgent_t agent, const char * var_name)`

Mark an agent variable for saving.

##### Parameters

*agent* A MobileC agent.

*var\_name* The name of the variable to mark for saving.

##### Returns

0 on success, non-zero on failure.

##### See also

`test1.xml`

### 5.38.8 Examples

See `agent_saved_variables_example/test1.xml` for an example of usage of this api function.

### 5.38.9 Examples

The following example demonstrates usage of `MC_AgentVariableSave()` from agent space.

Definition at line 761 of file `libmc.c`.

References `agent_s::datastate`, `MC_ERR_MEMORY`, `agent_task_s::num_saved_variables`, `agent_task_s::saved_variables`, `strdup`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

Referenced by `MC_AgentVariableSave_chdl()`.

#### 5.38.9.1 EXPORTMC int MC\_Barrier (MCAgency\_t attr, int id)

Definition at line 782 of file `libmc.c`.

References `mc_platform_s::barrier_queue`, `barrier_queue_Get()`, `barrier_node_s::cond`, `COND_BROADCAST`, `COND_WAIT`, `barrier_node_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MC_SUCCESS`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `node`, `barrier_node_s::num_registered`, and `barrier_node_s::num_waiting`.

Referenced by `MC_Barrier_chdl()`.

#### 5.38.9.2 EXPORTMC int MC\_BarrierDelete (MCAgency\_t attr, int id)

Find and delete an initialized MobileC Barrier.

##### Parameters

- attr* A running MobileC agency
- id* The id of the barrier node to delete

##### Returns

returns 0 on success, error if the node is not found or other failure.

Definition at line 822 of file `libmc.c`.

References `mc_platform_s::barrier_queue`, `barrier_queue_Delete()`, and `agency_s::mc_platform`.

Referenced by `MC_BarrierDelete_chdl()`.

#### 5.38.9.3 EXPORTMC int MC\_BarrierInit (MCAgency\_t attr, int id, int num\_procs)

Initialize a MobileC Barrier.

##### Parameters

- attr* A running MobileC agency
- id* The requested barrier id

*num\_procs* The number of agents/threads/processes that will wait on the barrier

### Returns

The allocated barrier id. May differ from the requested id if it is already in use.

## 5.38.10 Examples

The following example demonstrates an agent which sets up an MC\_Barrier.

Definition at line 808 of file libmc.c.

References `barrier_node_Initialize()`, `mc_platform_s::barrier_queue`, `barrier_queue_Add()`, `barrier_queue_Get()`, `MC_ERR`, `agency_s::mc_platform`, `MC_SUCCESS`, and `node`.

Referenced by `MC_BarrierInit_chdl()`.

**5.38.10.1 EXPORTMC int MC\_CallAgentFunc (MCAgent\_t agent, const char \*funcName, void \*returnVal, int numArgs, ...)**

Use custom ChOptions\_t type for internal Ch interpreter.

### Parameters

*attr* A running MobileC agency

*options* Initialized Ch options structure

### Returns

0 on success, `error_code_t` type on failure Calls a function defined in an agent

### Parameters

*agent* An initialized and executed MobileC agent

*funcName* The name of the function to call

*returnVal* The agent function's return value

*numArgs* The number of arguments supplied to the agent function

... A variable argument list to be supplied to the agent function

### Returns

0 if successful, `error_code_t` type on failure

## 5.38.11 Example

Definition at line 828 of file libmc.c.

References `agent_s::agent_interp`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `agent_s::run_lock`.

**5.38.11.1 EXPORTMC int MC\_CallAgentFuncArg (MCAgent\_t *agent*, const char \**funcName*, void \**returnVal*, void \**arg*)**

Calls a function defined in an agent.

**Parameters**

*agent* An initialized and executed MobileC agent

*funcName* The name of the function to call

*returnVal* The agent function's return value

*arg* The agent functions argument

**Note**

The agent function must be of the form 'void\* func(void\* arg);'

**Returns**

0 if successful, error\_code\_t type on failure

**5.38.12 Example**

Definition at line 849 of file libmc.c.

References agent\_s::agent\_interp, MUTEX\_LOCK, MUTEX\_UNLOCK, and agent\_s::run\_lock.

**5.38.12.1 EXPORTMC int MC\_CallAgentFuncV (MCAgent\_t *agent*, const char \**funcName*, void \**returnVal*, va\_list *ap*)**

Calls a function defined in an agent.

**Parameters**

*agent* An initialized and executed MobileC agent

*funcName* The name of the function to call

*returnVal* The agent function's return value

*ap* A variable argument list

**Returns**

0 if successful, error\_code\_t type on failure

Definition at line 868 of file libmc.c.

References agent\_s::agent\_interp, MUTEX\_LOCK, MUTEX\_UNLOCK, and agent\_s::run\_lock.

**5.38.12.2 EXPORTMC int MC\_CallAgentFuncVar (MCAgent\_t *agent*, const char \**funcName*, void \**returnVal*, ChVaList\_t *arglist*)**

Definition at line 888 of file libmc.c.

References agent\_s::agent\_interp, MUTEX\_LOCK, MUTEX\_UNLOCK, and agent\_s::run\_lock.

Referenced by MC\_CallAgentFunc\_chdl().

### 5.38.12.3 EXPORTMC MCAgent\_t MC\_ComposeAgent (const char \* *name*, const char \* *home*, const char \* *owner*, const char \* *code*, const char \* *return\_var\_name*, const char \* *server*, int *persistent*)

Compose a new agent dynamically without using a prewritten XML file.

#### Parameters

*name* The desired name of the new agent.

*home* The home of the new agent.

*owner* The owner of the new agent.

*code* The agent code

*return\_var\_name* The name of the agent's return variable. Set to "no-return" if no return variable is desired.

*server* The target destination of the agent.

*persistent* A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

#### Returns

This function returns a valid MCAgent\_t structure on success or NULL on failure.

Definition at line 928 of file libmc.c.

References MC\_ComposeAgentWithWorkgroup().

### 5.38.12.4 EXPORTMC MCAgent\_t MC\_ComposeAgentFromFile (const char \* *name*, const char \* *home*, const char \* *owner*, const char \* *filename*, const char \* *return\_var\_name*, const char \* *server*, int *persistent*)

Compose a new agent dynamically from a source code file.

#### Parameters

*filename* The filename of the source file

*name* The desired name of the new agent.

*home* The home of the new agent.

*owner* The owner of the new agent.

*code* The agent code

*return\_var\_name* The name of the agent's return variable. Set to "no-return" if no return variable is desired.

*server* The target destination of the agent.

*persistent* A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

#### Returns

This function returns a valid MCAgent\_t structure on success or NULL on failure.

Definition at line 1054 of file libmc.c.

References MC\_ComposeAgentFromFileWithWorkgroup().

Referenced by handler\_COMPOSE\_SEND().



### 5.38.12.5 EXPORTMC MCAgent\_t MC\_ComposeAgentFromFileS (const char \* *name*, const char \* *home*, const char \* *owner*, const char \* *filename*, const char \* *return\_var\_name*, const char \* *server*, int *persistent*, const char \* *workgroup\_code*)

Compose a new agent dynamically from a source code file.

#### Parameters

*filename* The filename of the source file

*name* The desired name of the new agent.

*home* The home of the new agent.

*owner* The owner of the new agent.

*code* The agent code

*return\_var\_name* The name of the agent's return variable. Set to "no-return" if no return variable is desired.

*server* The target destination of the agent.

*workgroup\_code* The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

*persitent* A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

#### Returns

This function returns a valid MCAgent\_t structure on success or NULL on failure.

#### Note

This function is deprecated. Please use the MC\_ComposeAgentFromFileWithWorkgroup function instead.

Definition at line 1076 of file libmc.c.

References MC\_ComposeAgentFromFileWithWorkgroup().

### 5.38.12.6 EXPORTMC MCAgent\_t MC\_ComposeAgentFromFileWithWorkgroup (const char \* *name*, const char \* *home*, const char \* *owner*, const char \* *filename*, const char \* *return\_var\_name*, const char \* *server*, int *persistent*, const char \* *workgroup\_code*)

Compose a new agent dynamically from a source code file.

#### Parameters

*filename* The filename of the source file

*name* The desired name of the new agent.

*home* The home of the new agent.

*owner* The owner of the new agent.

*code* The agent code

*return\_var\_name* The name of the agent's return variable. Set to "no-return" if no return variable is desired.

*server* The target destination of the agent.

***workgroup\_code*** The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

***persistent*** A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

### Returns

This function returns a valid `MCAgent_t` structure on success or `NULL` on failure.

Definition at line 1101 of file `libmc.c`.

References `MC_ComposeAgentWithWorkgroup()`.

Referenced by `MC_ComposeAgentFromFile()`, `MC_ComposeAgentFromFile_chdl()`, `MC_ComposeAgentFromFileS()`, and `MC_ComposeAgentFromFileWithWorkgroup_chdl()`.

**5.38.12.7 EXPORTMC MCAgent\_t MC\_ComposeAgentS (const char \* *name*, const char \* *home*, const char \* *owner*, const char \* *code*, const char \* *return\_var\_name*, const char \* *server*, int *persistent*, const char \* *workgroup\_code*)**

Compose a new agent dynamically without using a prewritten XML file.

### Parameters

***name*** The desired name of the new agent.

***home*** The home of the new agent.

***owner*** The owner of the new agent.

***code*** The agent code

***return\_var\_name*** The name of the agent's return variable. Set to "no-return" if no return variable is desired.

***server*** The target destination of the agent.

***workgroup\_code*** The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

***persistent*** A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

### Returns

This function returns a valid `MCAgent_t` structure on success or `NULL` on failure.

### Note

This function is deprecated. Please use the `MC_ComposeAgentWithWorkgroup` function instead.

Definition at line 951 of file `libmc.c`.

References `MC_ComposeAgentWithWorkgroup()`.

Referenced by `MC_ComposeAgentWithWorkgroup_chdl()`.

### 5.38.12.8 EXPORTMC MCAgent\_t MC\_ComposeAgentWithWorkgroup (const char \* *name*, const char \* *home*, const char \* *owner*, const char \* *code*, const char \* *return\_var\_name*, const char \* *server*, int *persistent*, const char \* *workgroup\_code*)

Compose a new agent dynamically without using a prewritten XML file.

#### Parameters

***name*** The desired name of the new agent.

***home*** The home of the new agent.

***owner*** The owner of the new agent.

***code*** The agent code

***return\_var\_name*** The name of the agent's return variable. Set to "no-return" if no return variable is desired.

***server*** The target destination of the agent.

***workgroup\_code*** The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

***persitent*** A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

#### Returns

This function returns a valid MCAgent\_t structure on success or NULL on failure.

#### Note

This function is deprecated

Definition at line 977 of file libmc.c.

References agent\_datastate\_s::agent\_code, agent\_datastate\_s::agent\_code\_ids, agent\_datastate\_s::agent\_codes, agent\_datastate\_New(), agent\_New(), agent\_s::agent\_status, agent\_task\_New(), agent\_s::agent\_type, agent\_s::datastate, agent\_s::home, MC\_LOCAL\_AGENT, MC\_WAIT\_MESSGSEND, agent\_s::name, agent\_datastate\_s::number\_of\_tasks, agent\_s::orphan, agent\_s::owner, agent\_task\_s::persistent, agent\_task\_s::server\_name, strdup, agent\_datastate\_s::tasks, agent\_task\_s::var\_name, and agent\_s::wg\_code.

Referenced by MC\_ComposeAgent(), MC\_ComposeAgent\_chdl(), MC\_ComposeAgentFromFileWithWorkgroup(), and MC\_ComposeAgentS().

### 5.38.12.9 EXPORTMC int MC\_CondBroadcast (MCAgency\_t *attr*, int *id*)

Wakes up all agents/threads waiting on a condition variable.

#### Parameters

***attr*** A MobileC agency

***id*** Synchronization variable id to broadcast to

#### See also

[MC\\_SyncInit\(\)](#), [MC\\_CondSignal\(\)](#)

**Returns**

0 on success, `error_code_t` type on failure

Definition at line 1145 of file `libmc.c`.

References `syncListNode_s::cond`, `COND_BROADCAST`, `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `syncListNode_s::signalled`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_CondBroadcast_chdl()`.

**5.38.12.10 EXPORTMC int MC\_CondReset (MCAgency\_t *attr*, int *id*)**

Reset a previously signalled MobileC condition variable.

**Parameters**

*attr* A MobileC Agency

*id* The synchronization variable id to reset

**See also**

[MC\\_SyncInit\(\)](#)

**Returns**

0 on success, `error_code_t` type on failure

Definition at line 1197 of file `libmc.c`.

References `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `syncListNode_s::signalled`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_CondReset_chdl()`.

**5.38.12.11 EXPORTMC int MC\_CondSignal (MCAgency\_t *attr*, int *id*)**

Wakes up at least one thread waiting on a condition variable.

**Parameters**

*attr* A MobileC agency

*id* synchronization variable id

**See also**

[MC\\_SyncInit\(\)](#)

**Returns**

0 on success, `error_code_t` type on failure

### 5.38.13 Example

The following example demonstrates the agent-space version of the function, which is nearly identical to the binary space api function.

Definition at line 1160 of file libmc.c.

References `syncListNode_s::cond`, `COND_SIGNAL`, `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `syncListNode_s::signalled`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_CondSignal_chdl()`.

#### 5.38.13.1 EXPORTMC int MC\_CondWait (MCAgency\_t *attr*, int *id*)

Wait on a MobileC synchronization variable.

##### Parameters

*attr* A MobileC agency

*id* a synchronization variable id

##### See also

[MC\\_SyncInit\(\)](#)

##### Returns

0 on success, `error_code_t` type on failure

### 5.38.14 Example

The following example demonstrates the agent-space version of this function.

Definition at line 1175 of file libmc.c.

References `syncListNode_s::cond`, `COND_WAIT`, `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `syncListNode_s::signalled`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_CondWait_chdl()`.

#### 5.38.14.1 int MC\_CopyAgent (MCAgent\_t \* *agent\_out*, const MCAgent\_t *agent\_in*)

Performs a deep-copy of an agent structure.

##### Parameters

*agent\_out* A pointer to the agent to copy to.

*agent\_in* The agent to copy

**Returns**

0 on success, `error_code_t` type on failure.

Definition at line 1215 of file `libmc.c`.

References `agent_Copy()`, and `MC_SUCCESS`.

**5.38.14.2 EXPORTMC int MC\_DeleteAgent (MCAgent\_t *agent*)**

Stop and remove an agent.

**Parameters**

*agent* An agent in any state (running, waiting, etc)

**Returns**

0 on success, `error_code_t` type on failure

Definition at line 1222 of file `libmc.c`.

References `CHECK_NULL`, `MC_ERR_INVALID`, `MC_ERR_INVALID_ARGS`, `MC_SUCCESS`, `MC_TerminateAgent()`, and `agent_s::wg_code`.

Referenced by `MC_DeleteAgent_chdl()`.

**5.38.14.3 EXPORTMC int MC\_DeleteAgentWG (MCAgent\_t *calling\_agent*, MCAgent\_t *agent*)**

Stop and remove an agent in the same workgroup.

**Parameters**

*calling\_agent* The calling agent

*agent* An agent in any state (running, waiting, etc)

**Note**

The agents must belong to the same workgroup.

**Returns**

0 on success, `error_code_t` type on failure

Definition at line 1239 of file `libmc.c`.

References `CHECK_NULL`, `MC_ERR_INVALID`, `MC_ERR_INVALID_ARGS`, `MC_SUCCESS`, `MC_TerminateAgentWG()`, and `agent_s::wg_code`.

Referenced by `MC_DeleteAgentWG_chdl()`.

**5.38.14.4 int MC\_DestroyServiceSearchResult (char \*\* *agentName*, char \*\* *serviceName*, int \* *agentID*, int *numResult*)**

Free memory allocated by a Service Search operation.

**Parameters**

*agentName* agent names returned by a search operation.  
*serviceName* service names return by a search operation.  
*agentID* list of agent id's returned by a search operation.  
*numResult* The number of hits returned by a search operation.

**Returns**

0 on success, error code on failure.

Definition at line 1259 of file libmc.c.

Referenced by MC\_DestroyServiceSearchResult\_chdl().

**5.38.14.5 EXPORTMC int MC\_End (MCAgency\_t attr)**

End an agency.

**Parameters**

*attr* A running agency

**Returns**

0 on success, error\_code\_t type on failure

**5.38.15 Example**

Definition at line 1308 of file libmc.c.

References mc\_platform\_s::acc, mc\_platform\_s::ams, buf, mc\_platform\_s::cmd\_prompt, COND\_BROADCAST, COND\_SIGNAL, mc\_platform\_s::connection\_queue, mc\_platform\_s::df, GET\_THREAD\_MODE, agency\_s::hostName, agency\_s::mc\_platform, mc\_platform\_Destroy(), MC\_THREAD\_ACC, MC\_THREAD\_AMS, MC\_THREAD\_CP, MC\_THREAD\_DF, mc\_platform\_s::message\_queue, MUTEX\_LOCK, MUTEX\_UNLOCK, agency\_s::portno, mc\_platform\_s::quit, mc\_platform\_s::quit\_cond, mc\_platform\_s::quit\_lock, send, cmd\_prompt\_s::thread, THREAD\_CANCEL, THREAD\_JOIN, and agency\_s::threads.

Referenced by MC\_End\_chdl().

**5.38.15.1 EXPORTMC MCAgent\_t MC\_FindAgentByID (MCAgency\_t attr, int ID)**

Find an agent by its id.

**Parameters**

*attr* the agency to search  
*ID* the id to search for

**Returns**

a valid agent on success, NULL on failure

Definition at line 1423 of file libmc.c.

References mc\_platform\_s::agent\_queue, and agency\_s::mc\_platform.

Referenced by MC\_FindAgentByID\_chdl().

#### 5.38.15.2 EXPORTMC MCAgent\_t MC\_FindAgentByName (MCAgency\_t *attr*, const char \* *name*)

Find an agent by its name.

##### Parameters

*attr* a running agency

*name* name to search for

##### Returns

a valid agent on success or NULL on failure

### 5.38.16 Example

Definition at line 1410 of file libmc.c.

References mc\_platform\_s::agent\_queue, and agency\_s::mc\_platform.

Referenced by MC\_AclSend(), MC\_DeleteAgent\_chdl(), MC\_DeleteAgentWG\_chdl(), MC\_FindAgentByName\_chdl(), MC\_TerminateAgent\_chdl(), and MC\_TerminateAgentWG\_chdl().

#### 5.38.16.1 EXPORTMC void\* MC\_GetAgentExecEngine (MCAgent\_t *agent*)

Retrieve an agent's Ch interpreter.

##### Parameters

*agent* a valid agent

##### Returns

a Ch interpreter of type 'ChInterp\_t' on success, or NULL on failure.

Definition at line 1478 of file libmc.c.

References agent\_s::agent\_interp.

#### 5.38.16.2 EXPORTMC int MC\_GetAgentID (MCAgent\_t *agent*)

Retrieve an agent's id.

Definition at line 1484 of file libmc.c.

References agent\_s::id.

Referenced by MC\_GetAgentID\_chdl().



**5.38.16.3 EXPORTMC char\* MC\_GetAgentName (MCAgent\_t *agent*)**

Definition at line 1492 of file libmc.c.

References agent\_s::lock, MUTEX\_LOCK, MUTEX\_UNLOCK, and agent\_s::name.

Referenced by MC\_GetAgentName\_chdl().

**5.38.16.4 EXPORTMC int MC\_GetAgentNumTasks (MCAgent\_t *agent*)**

Retrive the number of tasks an agent has.

**5.38.17 Example**

Definition at line 1575 of file libmc.c.

References agent\_s::datastate, and agent\_datastate\_s::number\_of\_tasks.

Referenced by MC\_GetAgentNumTasks\_chdl().

**5.38.17.1 EXPORTMC int MC\_GetAgentReturnData (MCAgent\_t *agent*, int *task\_num*, void \*\*  
*data*, int \* *dim*, int \*\* *extent*)**

Get an agent's return data.

**Parameters**

***agent*** a valid agent

***task\_num*** the task for which to retrieve the return data. The task must already be completed.

***data*** the return data. May be multi dimensional array.

***dim*** the number of dimensions of the return array.

***extent*** the extent of each one of the array dimensions.

**5.38.18 Example**

This file demonstrates the retrieval of agent return data from an agent

This is the agent which gets the data

Definition at line 1510 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::array\_dim, interpreter\_variable\_data\_s::array\_extent, CH\_DATATYPE\_SIZE, interpreter\_variable\_data\_s::data\_type, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, size, and agent\_datastate\_s::tasks.

**5.38.18.1 EXPORTMC int MC\_GetAgentStatus (MCAgent\_t *agent*)**

Get an agent's current status.

**Returns**

returns type 'enum MC\_AgentStatus\_e'

Definition at line 1456 of file libmc.c.

References agent\_s::agent\_status, agent\_s::lock, MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_GetAgentStatus\_chdl().

**5.38.18.2 EXPORTMC enum MC\_AgentType\_e MC\_GetAgentType (MCAgent\_t *agent*)**

Get an agent's type.

**Returns**

returns type 'enum MC\_AgentType\_e'

Definition at line 1581 of file libmc.c.

References agent\_s::agent\_type.

**5.38.18.3 EXPORTMC char\* MC\_GetAgentXMLString (MCAgent\_t *agent*)**

Get an agent's xml string.

**Returns**

a malloc'd character string containing the agent's xml code

Definition at line 1466 of file libmc.c.

References agent\_s::datastate, mxmlSaveAllocString(), and agent\_datastate\_s::xml\_agent\_root.

Referenced by MC\_GetAgentXMLString\_chdl().

**5.38.18.4 EXPORTMC int MC\_HaltAgency (MCAgency\_t *agency*)**

Halt an agency: Do not process new entries in queues.

**Parameters**

*agency* A handle to a running MobileC agency.

**Returns**

0 on success, non-zero on failure.

Definition at line 1627 of file libmc.c.

References mc\_platform\_s::giant, mc\_platform\_s::giant\_lock, agency\_s::mc\_platform, MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_GetAllAgents(), and MC\_HaltAgency\_chdl().

**5.38.18.5 EXPORTMC MCAgency\_t MC\_Initialize (int port, MCAgencyOptions\_t \* options)**

Initialize and start a MobileC agency.

**Parameters**

*port* the TCP port the agency should bind to

*options* initialized MobileC options or NULL for default options

**Returns**

a handle to a running MobileC agency or NULL on failure

**5.38.19 Example**

Definition at line 1636 of file libmc.c.

References agency\_s::agentInitCallback, agency\_s::agentInitUserData, MCAgencyOptions\_s::bluetooth, agency\_s::bluetooth, buf, MCAgencyOptions\_s::ch\_options, agency\_s::client, MCAgencyOptions\_s::default\_agent\_status, agency\_s::default\_agentstatus, f, agency\_s::hostName, MCAgencyOptions\_s::initInterps, agency\_s::initInterps, agency\_s::known\_host\_filename, MCAgencyOptions\_s::known\_host\_filename, MC\_InitializeAgencyOptions(), mc\_platform, agency\_s::mc\_platform, mc\_platform\_Initialize(), MC\_THREAD\_ALL, MCAgencyOptions\_s::passphrase, agency\_s::portno, agency\_s::priv\_key\_filename, MCAgencyOptions\_s::priv\_key\_filename, read\_encrypted\_file(), agency\_s::server, MCAgencyOptions\_s::stack\_size, agency\_s::stack\_size, strdup, MCAgencyOptions\_s::threads, and agency\_s::threads.

**5.38.19.1 EXPORTMC int MC\_InitializeAgencyOptions (struct MCAgencyOptions\_s \* options)**

Initialize MobileC options.

**Parameters**

*options* options to initialize.

**Returns**

0 on success, error\_code\_t on failure

**Note**

MobileC options should be initialized with this function before any of its members are modified.

**5.38.20 Example**

Definition at line 1749 of file libmc.c.

References MCAgencyOptions\_s::bluetooth, MCAgencyOptions\_s::default\_agent\_status, MCAgencyOptions\_s::initInterps, MCAgencyOptions\_s::known\_host\_filename, MC\_THREAD\_ALL, MC\_WAIT\_CH, MCAgencyOptions\_s::modified, MCAgencyOptions\_s::passphrase,

MCAgencyOptions\_s::priv\_key\_filename, MCAgencyOptions\_s::stack\_size, and MCAgencyOptions\_s::threads.

Referenced by MC\_Initialize().

#### 5.38.20.1 EXPORTMC int MC\_LoadAgentFromFile (MCAgency\_t attr, const char \*filename)

Load an agent from a file into an agency.

##### Parameters

**agency** A valid and running Mobile-C agency  
**filename** Filename containing the agent to load

##### Returns

0 on success, non-zero on failure.

Definition at line 1771 of file libmc.c.

References buf, agency\_s::mc\_platform, message\_Destroy(), message\_InitializeFromString(), message\_New(), mc\_platform\_s::message\_queue, MXML\_DESCEND, mxmlFindElement(), mxmlLoadString(), message\_s::to\_address, message\_s::xml\_payload, and message\_s::xml\_root.

#### 5.38.20.2 EXPORTMC int MC\_MainLoop (MCAgency\_t attr)

Wait indefinitely.

##### Note

This function is intended to block the calling thread forever.

Definition at line 2535 of file libmc.c.

References COND\_WAIT, agency\_s::mc\_platform, MUTEX\_LOCK, MUTEX\_UNLOCK, mc\_platform\_s::quit, mc\_platform\_s::quit\_cond, and mc\_platform\_s::quit\_lock.

#### 5.38.20.3 EXPORTMC int MC\_MigrateAgent (MCAgent\_t agent, const char \*hostname, int port)

Migrates a running agent to another host.

##### Parameters

**agent** The agent to migrate  
**hostname** The new host to migrate the agent to  
**port** The new port to migrate the agent to

##### Returns

0 on success, error\_code\_t type on failure.

Definition at line 1841 of file libmc.c.

References agent\_s::datastate, agent\_datastate\_s::progress\_modifier, agent\_task\_s::server\_name, agent\_datastate\_s::task\_progress, and agent\_datastate\_s::tasks.

Referenced by MC\_MigrateAgent\_chdl().

#### 5.38.20.4 EXPORTMC int MC\_MutexLock (MCAgency\_t *attr*, int *id*)

Locks a MobileC synchronization variable as a mutex.

##### Parameters

*attr* a MobileC agency handle

*id* the synchronization variable id to lock

##### Returns

0 on success, error\_code\_t type on failure

#### 5.38.21 Example

Consider the following agents, which use the agent-space version of this api function. Note that the 'sleep' agent is sent first, followed by the 'wake' agent.

Definition at line 1863 of file libmc.c.

References syncListNode\_s::lock, agency\_s::mc\_platform, MUTEX\_LOCK, mc\_platform\_s::syncList, and syncListFind().

Referenced by MC\_MutexLock\_chdl().

#### 5.38.21.1 EXPORTMC int MC\_MutexUnlock (MCAgency\_t *attr*, int *id*)

Definition at line 1875 of file libmc.c.

References syncListNode\_s::lock, agency\_s::mc\_platform, MUTEX\_UNLOCK, mc\_platform\_s::syncList, and syncListFind().

Referenced by MC\_MutexUnlock\_chdl().

#### 5.38.21.2 EXPORTMC int MC\_PrintAgentCode (MCAgent\_t *agent*)

Prints an agents code to stdout.

##### Returns

0 on success, error\_code\_t on failure

Definition at line 1887 of file libmc.c.

References agent\_datastate\_s::agent\_code, agent\_s::datastate, agent\_s::lock, MUTEX\_LOCK, MUTEX\_UNLOCK, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::task\_progress.

Referenced by MC\_PrintAgentCode\_chdl().

### 5.38.21.3 EXPORTMC int MC\_RegisterService (MCAgency\_t *agency*, MCAgent\_t *agent*, int *agentID*, const char \* *agentName*, char \*\* *serviceNames*, int *numServices*)

Register a new service with the Directory Facilitator.

#### Parameters

*agency* a MobileC agency handle  
*agent* (OPTIONAL: See note) a MobileC agent  
*agentID* (OPTIONAL: See note) a MobileC agent id  
*agentName* (OPTIONAL: See note) a MobileC agent name  
*serviceNames* an array of character strings of service names  
*numServices* the number of services described in 'serviceNames'

#### Returns

0 on success, error\_code\_t type on failure

#### Note

Three of the input arguments are optional. The function expects as input the arguments 'agent XOR (agentID AND agentName)'.

### 5.38.22 Example

Definition at line 1903 of file libmc.c.

References CHECK\_NULL, mc\_platform\_s::df, df\_AddRequest(), df\_request\_list\_node\_New(), agent\_s::id, MC\_ERR\_INVALID\_ARGS, MC\_ERR\_MEMORY, agency\_s::mc\_platform, MUTEX\_INIT, MUTEX\_T, and agent\_s::name.

Referenced by MC\_RegisterService\_chdl().

### 5.38.22.1 EXPORTMC int MC\_ResetSignal (MCAgency\_t *attr*)

Reset a MobileC signal.

#### Returns

0 on success, error\_code\_t on failure

#### See also

[MC\\_WaitSignal\(\)](#)

Definition at line 2035 of file libmc.c.

References COND\_SIGNAL, mc\_platform\_s::giant, mc\_platform\_s::giant\_cond, mc\_platform\_s::giant\_lock, MC\_NO\_SIGNAL, agency\_s::mc\_platform, mc\_platform\_s::MC\_signal, MUTEX\_LOCK, and MUTEX\_UNLOCK.

### 5.38.22.2 EXPORTMC int MC\_ResumeAgency (MCAgency\_t *agency*)

Resumes a halted agency.

#### Parameters

*agency* An agency previously halted with the [MC\\_HaltAgency\(\)](#) function.

#### Returns

0 on success, non-zero on failure.

Definition at line 1982 of file libmc.c.

References `mc_platform_s::giant`, `mc_platform_s::giant_lock`, `agency_s::mc_platform`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

Referenced by `MC_GetAllAgents()`, and `MC_ResumeAgency_chdl()`.

### 5.38.22.3 EXPORTMC MCAgent\_t MC\_RetrieveAgent (MCAgency\_t *attr*)

Retrieves the oldest agent from an agency.

#### Returns

a valid agent or NULL on failure

Definition at line 1991 of file libmc.c.

References `mc_platform_s::agent_queue`, `agent_s::agent_status`, `ListSearch()`, `MC_AGENT_NEUTRAL`, `agency_s::mc_platform`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

Referenced by `MC_RetrieveAgent_chdl()`.

### 5.38.22.4 EXPORTMC char\* MC\_RetrieveAgentCode (MCAgent\_t *agent*)

Retrieves an agent's Ch code.

#### Returns

a malloc'd character string on success, NULL on failure

Definition at line 2019 of file libmc.c.

References `agent_datastate_s::agent_code`, `buf`, `agent_s::datastate`, `agent_s::lock`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `agent_datastate_s::task_progress`.

Referenced by `MC_RetrieveAgentCode_chdl()`.

### 5.38.22.5 EXPORTMC int MC\_SearchForService (MCAgency\_t *attr*, const char \* *searchString*, char \*\*\* *agentNames*, char \*\*\* *serviceNames*, int \*\* *agentIDs*, int \* *numResults*)

Search the directory facilitator for a service.

#### Returns

0 on success, `error_code_t` on failure

**Parameters**

*attr* (input) a MobileC agency handle  
*searchString* (input) substring to search services for  
*agentNames* (return) array of agent names with matching services  
*serviceNames* (return) array of matching service names  
*agentIDs* (return) array of matching agent IDs  
*numResults* (return) number of matching results

**5.38.23 Example**

Definition at line 2046 of file libmc.c.

References CHECK\_NULL, COND\_SLEEP\_ACTION, mc\_platform\_s::df, df\_AddRequest(), df\_request\_list\_node\_Destroy(), df\_request\_list\_node\_New(), df\_request\_search\_Destroy(), df\_request\_search\_New(), MC\_ERR\_MEMORY, agency\_s::mc\_platform, MC\_SUCCESS, and search.

Referenced by MC\_SearchForService\_chdl().

**5.38.23.1 EXPORTMC int MC\_SemaphorePost (MCAgency\_t *attr*, int *id*)**

Post to a MobileC synchronization variable semaphore.

**Parameters**

*attr* a MobileC agency handle  
*id* the synchronization variable id to post to

**Returns**

0 on success, error\_code\_t type on failure

**5.38.24 Example**

Definition at line 2096 of file libmc.c.

References agency\_s::mc\_platform, syncListNode\_s::sem, SEMAPHORE\_POST, mc\_platform\_s::syncList, and syncListFind().

Referenced by MC\_SemaphorePost\_chdl().

**5.38.24.1 EXPORTMC int MC\_SemaphoreWait (MCAgency\_t *attr*, int *id*)**

Decreases a MobileC synchronization variable semaphore count by one.

**Parameters**

*attr* a MobileC agency handle



*id* synchronization variable id to wait on

### Returns

0 on MC\_SUCCESS, error\_code\_t type of failure

### Note

If the semaphore count is already zero, this function will block until another thread posts to the semaphore.

## 5.38.25 Example

Definition at line 2108 of file libmc.c.

References `agency_s::mc_platform`, `syncListNode_s::sem`, `SEMAPHORE_WAIT`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_SemaphoreWait_chdl()`.

### 5.38.25.1 EXPORTMC int MC\_SendAgent (MCAgency\_t *attr*, const char \* *message*)

Sends an agent migration message.

#### Parameters

*attr* a MobileC agency handle

*message* a valid MobileC xml agent migration message

Definition at line 2130 of file libmc.c.

References `agent_Initialize()`, `mc_platform_s::agent_queue`, `agent_s::agent_status`, `mc_platform_s::ams`, `COND_BROADCAST`, `MC_ERR`, `agency_s::mc_platform`, `MC_WAIT_MESSGSEND`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

### 5.38.25.2 EXPORTMC int MC\_SendAgentFile (MCAgency\_t *attr*, const char \* *filename*)

Sends an agent migration message.

#### Parameters

*attr* a MobileC agency handle

*filename* file containing a valid MobileC xml agent migration message

Definition at line 2178 of file libmc.c.

References `agent_Initialize()`, `mc_platform_s::agent_queue`, `agent_s::agent_status`, `mc_platform_s::ams`, `buf`, `COND_BROADCAST`, `agency_s::mc_platform`, `MC_WAIT_MESSGSEND`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

Referenced by `handler_SEND()`.

### 5.38.25.3 EXPORTMC int MC\_SendAgentMigrationMessage (MCAgency\_t *attr*, const char \* *message*, const char \* *hostname*, int *port*)

Sends an agent migration message.

#### Parameters

*attr* a MobileC agency handle  
*message* a valid MobileC xml agent migration message  
*hostname* host to send the message to  
*port* port to send the message to

Definition at line 2244 of file libmc.c.

References MC\_ERR, agency\_s::mc\_platform, message\_Destroy(), message\_InitializeFromString(), message\_New(), and mc\_platform\_s::message\_queue.

Referenced by MC\_SendAgentMigrationMessage\_chdl().

### 5.38.25.4 EXPORTMC int MC\_SendAgentMigrationMessageFile (MCAgency\_t *attr*, const char \* *filename*, const char \* *hostname*, int *port*)

Sends an agent migration message.

#### Parameters

*attr* a MobileC agency handle  
*filename* file containing a valid MobileC xml agent migration message  
*hostname* hostname to send the agent to  
*port* port to send the agent to

Definition at line 2278 of file libmc.c.

References agent\_Initialize(), mc\_platform\_s::agent\_queue, agent\_s::agent\_status, mc\_platform\_s::ams, buf, COND\_BROADCAST, agency\_s::mc\_platform, MC\_WAIT\_MESSGSEND, message\_Destroy(), message\_InitializeFromString(), message\_New(), MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_SendAgentMigrationMessageFile\_chdl().

### 5.38.25.5 EXPORTMC int MC\_SetAgentStatus (MCAgent\_t *agent*, enum MC\_AgentStatus\_e *status*)

Set an agent's status.

#### Parameters

*agent* a MobileC agent  
*status* agent status of type 'enum MC\_AgentStatus\_e'

#### Returns

0 on success, or error\_code\_t on failure

Definition at line 2358 of file libmc.c.

References `agent_s::agent_status`, `mc_platform_s::ams`, `COND_SIGNAL`, `agent_s::lock`, `agent_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `agent_s::orphan`.

Referenced by `MC_SetAgentStatus_chdl()`.

#### 5.38.25.6 EXPORTMC int MC\_SetDefaultAgentStatus (MCAgency\_t *agency*, enum MC\_AgentStatus\_e *status*)

Sets default incoming agent status.

##### Parameters

*agency* a MobileC agency handle

*status* the status to set all incoming agents

##### Returns

0 on success, `error_type_t` on failure

##### Note

using this function will override any status the incoming agent attempts to set for itself.

Definition at line 2373 of file libmc.c.

References `mc_platform_s::default_agentstatus`, and `agency_s::mc_platform`.

Referenced by `MC_SetDefaultAgentStatus_chdl()`.

#### 5.38.25.7 EXPORTMC int MC\_SetThreadOff (MCAgencyOptions\_t \* *options*, enum MC\_ThreadIndex\_e *index*)

Sets a MobileC thread to "off" status.

##### Parameters

*options* MobileC options previously initialized with [MC\\_InitializeAgencyOptions\(\)](#)

*index* the thread to set

##### Returns

0 on success, `error_code_t` on failure

##### Note

This function must be called before [MC\\_Initialize\(\)](#). Once an agency is started with `MC_Initialize`, the `MC_SetThread` functions will have no effect.

Definition at line 2400 of file libmc.c.

References `SET_THREAD_OFF`, and `MCAgencyOptions_s::threads`.

#### 5.38.25.8 EXPORTMC int MC\_SetThreadOn (MCAgencyOptions\_t \* *options*, enum MC\_ThreadIndex\_e *index*)

Sets a MobileC thread to "on" status.

##### Parameters

*options* MobileC options previously initialized with [MC\\_InitializeAgencyOptions\(\)](#)

*index* the thread to set

##### Returns

0 on success, error\_code\_t on failure

##### Note

This function must be called before [MC\\_Initialize\(\)](#). Once an agency is started with MC\_Initialize, the MC\_SetThread functions will have no effect.

Definition at line 2383 of file libmc.c.

References SET\_THREAD\_ON, and MCAgencyOptions\_s::threads.

#### 5.38.25.9 EXPORTMC int MC\_SetThreadsAllOff (MCAgencyOptions\_t \* *options*)

Set all MobileC threads to 'off' status.

##### Parameters

*options* a MobileC options structure initialized with with the [MC\\_InitializeAgencyOptions\(\)](#) function.

##### Returns

0 on success, error code on failure.

Definition at line 2407 of file libmc.c.

References MC\_THREAD\_ALL, SET\_THREAD\_OFF, and MCAgencyOptions\_s::threads.

#### 5.38.25.10 EXPORTMC int MC\_SetThreadsAllOn (MCAgencyOptions\_t \* *options*)

Set all Mobile-C threads on.

##### Parameters

*options* MobileC options structure, initialized with [MC\\_InitializeAgencyOptions\(\)](#)

##### Returns

0 on success, error code on failure.

Definition at line 2390 of file libmc.c.

References MC\_THREAD\_ALL, SET\_THREAD\_ON, and MCAgencyOptions\_s::threads.

**5.38.25.11 EXPORTMC int MC\_Steer (MCAgency\_t *attr*, int(\*) (void \*data) *funcptr*, void \* *arg*)**

Set up a steerable algorithm.

**Parameters**

*attr* a MobileC agency handle  
*funcptr* a function pointer to the algorithm  
*arg* an argument for the algorithm function

**Returns**

0 on success, error\_code\_t on failure

**Note**

The algorithm function must contain a call to MC\_SteerControl in order for the algorithm to be steerable.

**5.38.26 Example**

Definition at line 2417 of file libmc.c.

References agency\_s::mc\_platform, MC\_RESTART, MC\_RUN, mc\_platform\_s::MC\_steer\_command, mc\_platform\_s::MC\_steer\_lock, MUTEX\_LOCK, and MUTEX\_UNLOCK.

**5.38.26.1 EXPORTMC enum MC\_SteerCommand\_e MC\_SteerControl (void)**

The MobileC user-algorithm steering function.

**Returns**

The current steering command

**Note**

This function belongs inside a user's steerable algorithm.

**See also**

[MC\\_Steer\(\)](#)

Definition at line 2436 of file libmc.c.

References COND\_WAIT, mc\_platform\_s::MC\_steer\_command, mc\_platform\_s::MC\_steer\_cond, mc\_platform\_s::MC\_steer\_lock, MC\_SUSPEND, MUTEX\_LOCK, and MUTEX\_UNLOCK.

**5.38.26.2 EXPORTMC int MC\_SyncDelete (MCAgency\_t *attr*, int *id*)**

Deletes a previously initialized synchronization variable.

**Parameters**

*attr* a MobileC agency handle

*id* the sync variable id to delete

### Returns

0 on success, or `error_code_t` on failure

Definition at line 2452 of file `libmc.c`.

References `syncList_s::giant_lock`, `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_platform_s::syncList`, `syncListFind()`, `syncListNodeDestroy()`, and `syncListRemove()`.

Referenced by `MC_SyncDelete_chdl()`.

### 5.38.26.3 EXPORTMC int MC\_SyncInit (MCAgency\_t attr, int id)

Initializes a new MobileC synchronization variable.

### Parameters

*attr* a MobileC agency handle

*id* the requested sync variable id

### Returns

new sync variable's id. May be different than the requested id.

### Note

Each synchronization variable may be used as a mutex, condition variable, or semaphore. However, it should only be used as one type of synchronization variable per instance, or undefined behaviour may result.

### See also

[MC\\_MutexLock\(\)](#), [MC\\_MutexUnlock\(\)](#), [MC\\_CondWait\(\)](#), [MC\\_CondSignal\(\)](#), [MC\\_CondBroadcast](#), [MC\\_SemaphorePost\(\)](#), [MC\\_SemaphoreWait\(\)](#)

Definition at line 2482 of file `libmc.c`.

References `syncList_s::giant_lock`, `syncListNode_s::id`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `node`, `mc_platform_s::syncList`, `syncListAddNode()`, `syncListFind()`, and `syncListNodeNew()`.

Referenced by `MC_SyncInit_chdl()`.

### 5.38.26.4 EXPORTMC int MC\_TerminateAgent (MCAgent\_t agent)

Halt a running agent.

### Returns

0 on success, `error_code_t` on failure

Definition at line 2507 of file `libmc.c`.

References `agent_s::agent_interp`.

Referenced by `ams_ManageAgentList()`, `MC_DeleteAgent()`, and `MC_TerminateAgent_chdl()`.

**5.38.26.5 EXPORTMC int MC\_TerminateAgentWG (MCAgent\_t *calling\_agent*, MCAgent\_t *agent*)**

Definition at line 2517 of file libmc.c.

References agent\_s::agent\_interp, MC\_ERR\_INVALID\_ARGS, and agent\_s::wg\_code.

Referenced by MC\_DeleteAgentWG(), and MC\_TerminateAgentWG\_chdl().

**5.38.26.6 EXPORTMC int MC\_WaitAgent (MCAgency\_t *attr*)**

Wait indefinitely.

**Note**

This function is intended to block the calling thread forever. Wait for an agent arrival event  
This function blocks until an agent arrival signal is triggered, at which point it unblocks.

Definition at line 2546 of file libmc.c.

References mc\_platform\_s::agent\_queue, COND\_WAIT, mc\_platform, agency\_s::mc\_platform, MUTEX\_LOCK, MUTEX\_UNLOCK, and size.

**5.38.26.7 EXPORTMC MCAgent\_t MC\_WaitRetrieveAgent (MCAgency\_t *attr*)**

Wait and retrieve an agent.

**Returns**

a valid MobileC agent on success, or NULL on failure

**Note**

This function blocks until the arrival of an agent. The agent is retrieved after it is initialized, but before it is executed.

Definition at line 2566 of file libmc.c.

References mc\_platform\_s::agent\_queue, ListSearch(), agency\_s::mc\_platform, MC\_RECV\_AGENT, MC\_WaitSignal(), MUTEX\_LOCK, and MUTEX\_UNLOCK.

**5.38.26.8 EXPORTMC int MC\_WaitSignal (MCAgency\_t *attr*, int *signals*)**

Wait for a MobileC signal.

**Parameters**

*attr* a MobileC agency handle

*signals* a flag of signals to wait for, of type 'enum MC\_Signal\_e'

**Returns**

0 on success, error\_code\_t on failure

**Note**

the parameter 'signals' may be something like 'MC\_RECV\_MESSAGE | MC\_RECV\_AGENT', etc.

### 5.38.27 Example

Definition at line 2584 of file libmc.c.

References COND\_WAIT, mc\_platform\_s::giant, mc\_platform\_s::giant\_lock, agency\_s::mc\_platform, mc\_platform\_s::MC\_signal, mc\_platform\_s::MC\_signal\_cond, mc\_platform\_s::MC\_signal\_lock, MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_WaitRetrieveAgent().



## 5.39 /home/dko/Projects/mobilec/src/include/macros.h File Reference

```
#include <pthread.h>
#include <semaphore.h>
#include "config.h"
#include <errno.h>
```

### Defines

- #define [GET\\_THREAD\\_MODE](#)(a, b) ( (a & (1<<b)) / (1<<b) )
- #define [SET\\_THREAD\\_ON](#)(a, b) a = (a | (1<<b))
- #define [SET\\_THREAD\\_OFF](#)(a, b) a = (a & (~(1<<b)))
- #define [STRUCT](#)(name, members)
- #define [SOCKET\\_ERROR](#)()
- #define [PTHREAD\\_STACK\\_SIZE](#) 131072
- #define [THREAD\\_CREATE](#)(thread\_handle, function, arg)
- #define [THREAD\\_CANCEL](#)(thread\_handle) pthread\_cancel( thread\_handle )
- #define [THREAD\\_JOIN](#)(thread\_handle) pthread\_join( thread\_handle, NULL )
- #define [THREAD\\_DETACH](#)(thread\_handle)
- #define [THREAD\\_EXIT](#)() pthread\_exit(NULL)
- #define [MUTEX\\_T](#) pthread\_mutex\_t
- #define [MUTEX\\_INIT](#)(mutex) pthread\_mutex\_init(mutex, NULL)
- #define [MUTEX\\_DESTROY](#)(mutex) pthread\_mutex\_destroy(mutex)
- #define [MUTEX\\_LOCK](#)(mutex)
- #define [MUTEX\\_UNLOCK](#)(mutex) pthread\_mutex\_unlock( mutex )
- #define [MUTEX\\_NEW](#)(mutex)
- #define [COND\\_T](#) pthread\_cond\_t
- #define [COND\\_INIT](#)(cond) pthread\_cond\_init([cond](#), NULL)
- #define [COND\\_DESTROY](#)(cond) pthread\_cond\_destroy([cond](#))
- #define [COND\\_WAIT](#)(cond, mutex) pthread\_cond\_wait([cond](#), mutex )
- #define [COND\\_SLEEP](#)(cond, mutex, test)
- #define [COND\\_RESET](#)(cond, mutex) pthread\_mutex\_unlock( mutex );
- #define [COND\\_SLEEP\\_ACTION](#)(cond, mutex, action)
- #define [SIGNAL](#)(cond, mutex, action)
- #define [COND\\_BROADCAST](#)(cond) pthread\_cond\_broadcast( [cond](#) )
- #define [COND\\_SIGNAL](#)(cond) pthread\_cond\_signal( [cond](#) )
- #define [SEMAPHORE\\_T](#) sem\_t
- #define [SEMAPHORE\\_INIT](#)(sem) sem\_init(sem, 0, 0)
- #define [SEMAPHORE\\_DESTROY](#)(sem) sem\_destroy(sem)
- #define [SEMAPHORE\\_WAIT](#)(sem) sem\_wait(sem)
- #define [SEMAPHORE\\_POST](#)(sem) sem\_post(sem)
- #define [RWLOCK\\_T](#) mc\_rwlock\_t
- #define [RWLOCK\\_INIT](#)(rwlock) mc\_rwlock\_init(rwlock)
- #define [RWLOCK\\_DESTROY](#)(rwlock) mc\_rwlock\_destroy(rwlock)
- #define [RWLOCK\\_RDLOCK](#)(rwlock) mc\_rwlock\_rdlock(rwlock)
- #define [RWLOCK\\_RDUNLOCK](#)(rwlock) mc\_rwlock\_rdunlock(rwlock)
- #define [RWLOCK\\_WRLOCK](#)(rwlock) mc\_rwlock\_wrlock(rwlock)

- #define [RWLOCK\\_WRUNLOCK](#)(rwlock) mc\_rwlock\_wrunlock(rwlock)
- #define [WAKE\\_QUEUE](#)(queue, action)
- #define [SLEEP\\_QUEUE](#)(queue)
- #define [SLEEP\\_RESET](#)(queue) pthread\_mutex\_unlock( queue->thread\_mutex )
- #define [CHECK\\_NULL](#)(var, action)
- #define [WARN](#)(message)
- #define [CH\\_DATATYPE\\_SIZE](#)(type, size)
- #define [CH\\_DATATYPE\\_STRING](#)(type, string)
- #define [CH\\_DATATYPE\\_VALUE\\_STRING](#)(type, string, p)
- #define [CH\\_STRING\\_DATATYPE](#)(string, type)
- #define [CH\\_DATATYPE\\_STR\\_TO\\_VAL](#)(type, string, val)

### 5.39.1 Define Documentation

#### 5.39.1.1 #define CH\_DATATYPE\_SIZE(type, size)

**Value:**

```

switch(type) {
    case CH_CHARTYPE:
        size = sizeof(char);
        break;
    case CH_INTTYPE:
        size = sizeof(int);
        break;
    case CH_UINTTYPE:
        size = sizeof(unsigned int);
        break;
    case CH_SHORTTYPE:
        size = sizeof(short);
        break;
    case CH_USHORTTYPE:
        size = sizeof(unsigned short);
        break;
    case CH_FLOATTYPE:
        size = sizeof(float);
        break;
    case CH_DOUBLETYPE:
        size = sizeof(double);
        break;
    default:
        fprintf(stderr, "Unknown data type: %d at %s:%d",
            type, __FILE__, __LINE__);
        size=0;
}

```

Definition at line 490 of file macros.h.

Referenced by `agent_AddPersistentVariable()`, `agent_xml_compose__create_row_nodes()`, `agent_xml_parse__data()`, `agent_xml_parse__fill_row_data()`, `agent_xml_parse__row()`, `interpreter_variable_data_Initialize()`, `interpreter_variable_data_InitializeFromAgent()`, `MC_AgentReturnDataSize()`, and `MC_GetAgentReturnData()`.

#### 5.39.1.2 #define CH\_DATATYPE\_STR\_TO\_VAL(type, string, val)

**Value:**

```

switch (type) { \
    case CH_INTTYPE: \
        *(int*)val = atoi(string); \
    break; \
    case CH_UINTTYPE: \
        *(unsigned int*)val = atoi(string); \
    break; \
    case CH_SHORTTYPE: \
        *(short*)val = (short)atoi(string); /*FIXME*/ \
    break; \
    case CH_USHORTTYPE: \
        *(unsigned short*)val = (unsigned short)atoi(string); /*F
        IXME*/ \
    break; \
    case CH_FLOATTYPE: \
        *(float*)val = strtod(string, NULL); \
    break; \
    case CH_DOUBLETTYPE: \
        *(double*)val = strtod(string, NULL); \
    break; \
    default: \
        fprintf(stderr, \
            "Unsupported data type: %d %s:%d\n", \
            type, __FILE__, __LINE__ ); \
}

```

Definition at line 604 of file macros.h.

Referenced by agent\_xml\_parse\_\_data().

### 5.39.1.3 #define CH\_DATATYPE\_STRING(type, string)

**Value:**

```

switch(type) {
    case CH_CHARTYPE:
        strcpy(string, "char");
    break;
    case CH_INTTYPE:
        strcpy(string, "int");
    break;
    case CH_UINTTYPE:
        strcpy(string, "unsigned int");
    break;
    case CH_SHORTTYPE:
        strcpy(string, "short");
    break;
    case CH_USHORTTYPE:
        strcpy(string, "unsigned short");
    break;
    case CH_FLOATTYPE:
        strcpy(string, "float");
    break;
    case CH_DOUBLETTYPE:
        strcpy(string, "double");
    break;
    default:
        fprintf(stderr,
            "Unsupported data type: %d %s:%d\n",
            type, __FILE__, __LINE__ );
}

```

Definition at line 521 of file macros.h.

Referenced by agent\_xml\_compose\_\_data().

#### 5.39.1.4 #define CH\_DATATYPE\_VALUE\_STRING(type, string, p)

**Value:**

```

switch(type) {
    case CH_CHARTYPE:
        sprintf(string, "%c", *((char*)p));
        break;
    case CH_INTTYPE:
        sprintf(string, "%d", *((int*)p));
        break;
    case CH_UINTTYPE:
        sprintf(string, "%d", *((unsigned int*)p));
        break;
    case CH_SHORTTYPE:
        sprintf(string, "%d", *((short*)p));
        break;
    case CH_USHORTTYPE:
        sprintf(string, "%d", *((unsigned short*)p));
        break;
    case CH_FLOATTYPE:
        sprintf(string, "%f", *((float*)p));
        break;
    case CH_DOUBLETTYPE:
        sprintf(string, "%f", *((double*)p));
        break;
    default:
        fprintf(stderr,
            "Unsupported data type: %d %s:%d\n",
            type, __FILE__, __LINE__);
}

```

Definition at line 553 of file macros.h.

Referenced by agent\_xml\_compose\_\_create\_row\_nodes(), and agent\_xml\_compose\_\_data().

#### 5.39.1.5 #define CH\_STRING\_DATATYPE(string, type)

**Value:**

```

if (!strcmp(string, "int")) {
    type = CH_INTTYPE;
} else if (!strcmp(string, "float")) {
    type = CH_FLOATTYPE;
} else if (!strcmp(string, "double")) {
    type = CH_DOUBLETTYPE;
} else if (!strcmp(string, "unsigned int")) {
    type = CH_UINTTYPE;
} else if (!strcmp(string, "short")) {
    type = CH_SHORTTYPE;
} else if (!strcmp(string, "unsigned short")) {
    type = CH_USHORTTYPE;
} else if (!strcmp(string, "char")) {
    type = CH_CHARTYPE;
} else {
    fprintf(stderr,
        "Unsupported data type: %d %s:%d\n",
        type, __FILE__, __LINE__);
}

```

Definition at line 582 of file macros.h.

Referenced by agent\_xml\_parse\_\_data().

**5.39.1.6 #define CHECK\_NULL(var, action)****Value:**

```

if ( var == NULL ) {
    fprintf(stderr, "Pointer var is null: expected otherwise.\n");
    fprintf(stderr, "Error occured at %s:%d", __FILE__, __LINE__);
    action;
}

```

Definition at line 473 of file macros.h.

Referenced by agent\_AddPersistentVariable(), agent\_datastate\_New(), agent\_xml\_parse\_home(), agent\_xml\_parse\_name(), agent\_xml\_parse\_owner(), agent\_xml\_parse\_sender(), agent\_xml\_parse\_task(), ams\_Initialize(), barrier\_node\_Initialize(), barrier\_queue\_New(), df\_request\_list\_New(), df\_request\_list\_node\_New(), df\_request\_search\_New(), fipa\_agent\_identifier\_Parse(), fipa\_word\_Parse(), http\_ParseExpression(), interpreter\_variable\_data\_Initialize(), interpreter\_variable\_data\_InitializeFromAgent(), interpreter\_variable\_data\_New(), MC\_AclSend\_chdl(), MC\_AddAgent\_chdl(), MC\_Barrier\_chdl(), MC\_BarrierDelete\_chdl(), MC\_BarrierInit\_chdl(), MC\_CondBroadcast\_chdl(), MC\_CondReset\_chdl(), MC\_CondSignal\_chdl(), MC\_CondWait\_chdl(), MC\_DeleteAgent(), MC\_DeleteAgentWG(), MC\_DeregisterService\_chdl(), MC\_End\_chdl(), MC\_FindAgentByID\_chdl(), MC\_FindAgentByName\_chdl(), MC\_HaltAgency\_chdl(), MC\_MutexLock\_chdl(), MC\_MutexUnlock\_chdl(), mc\_platform\_Initialize(), MC\_RegisterService(), MC\_RegisterService\_chdl(), MC\_ResumeAgency\_chdl(), MC\_RetrieveAgent\_chdl(), mc\_rwlock\_init(), MC\_SearchForService(), MC\_SearchForService\_chdl(), MC\_SemaphorePost\_chdl(), MC\_SemaphoreWait\_chdl(), MC\_SendAgentMigrationMessage\_chdl(), MC\_SendSteerCommand\_chdl(), MC\_SetDefaultAgentStatus\_chdl(), MC\_SyncDelete\_chdl(), MC\_SyncInit\_chdl(), message\_InitializeFromAgent(), message\_InitializeFromConnection(), message\_InitializeFromString(), message\_New(), message\_queue\_SendOutgoing(), message\_send\_Thread(), message\_xml\_parse\_message(), mtp\_http\_InitializeFromConnection(), mtp\_http\_New(), syncListNodeInit(), syncListNodeNew(), xml\_get\_cdata(), xml\_get\_text(), and xml\_new\_cdata().

**5.39.1.7 #define COND\_BROADCAST(cond) pthread\_cond\_broadcast( cond )**

Definition at line 225 of file macros.h.

Referenced by acc\_MessageHandlerThread(), acc\_Thread(), ams\_Thread(), df\_Thread(), handler\_QUIT(), listen\_Thread(), MC\_Barrier(), MC\_CondBroadcast(), MC\_End(), MC\_SendAgent(), MC\_SendAgentFile(), MC\_SendAgentMigrationMessageFile(), and MC\_SendSteerCommand().

**5.39.1.8 #define COND\_DESTROY(cond) pthread\_cond\_destroy(cond)**

Definition at line 201 of file macros.h.

Referenced by acc\_Destroy(), ams\_Destroy(), barrier\_node\_Destroy(), df\_Destroy(), df\_request\_list\_Destroy(), df\_request\_list\_node\_Destroy(), df\_request\_search\_Destroy(), mc\_platform\_Destroy(), mc\_rwlock\_destroy(), and syncListNodeDestroy().

**5.39.1.9 #define COND\_INIT(cond) pthread\_cond\_init(cond, NULL)**

Definition at line 198 of file macros.h.

Referenced by acc\_Initialize(), ams\_Initialize(), barrier\_node\_Initialize(), df\_Initialize(), df\_request\_list\_New(), df\_request\_list\_node\_New(), df\_request\_search\_New(), mc\_platform\_Initialize(), mc\_rwlock\_init(), syncListNodeInit(), and syncListNodeNew().

**5.39.1.10 #define COND\_RESET(cond, mutex) pthread\_mutex\_unlock( mutex );**

Definition at line 213 of file macros.h.

**5.39.1.11 #define COND\_SIGNAL(cond) pthread\_cond\_signal( cond )**

Definition at line 227 of file macros.h.

Referenced by agent\_RunChScriptThread(), AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), MC\_AddAgent(), MC\_CondSignal(), MC\_End(), MC\_ResetSignal(), mc\_rwlock\_rdunlock(), mc\_rwlock\_wrunlock(), and MC\_SetAgentStatus().

**5.39.1.12 #define COND\_SLEEP(cond, mutex, test)**

**Value:**

```
if (pthread_mutex_lock( mutex )) \
    printf("pthread lock error: %s:%d\n", __FILE__, __LINE__); \
if (!test) { \
    pthread_cond_wait( cond, mutex ); \
}
```

Definition at line 207 of file macros.h.

**5.39.1.13 #define COND\_SLEEP\_ACTION(cond, mutex, action)**

**Value:**

```
if (pthread_mutex_lock( mutex )) \
    printf("pthread lock error: %s:%d\n", __FILE__, __LINE__); \
action; \
pthread_cond_wait( cond, mutex );
```

Definition at line 215 of file macros.h.

Referenced by MC\_SearchForService().

**5.39.1.14 #define COND\_T pthread\_cond\_t**

Definition at line 196 of file macros.h.

Referenced by acc\_Initialize(), ams\_Initialize(), barrier\_node\_Initialize(), df\_Initialize(), df\_request\_list\_New(), df\_request\_list\_node\_New(), df\_request\_search\_New(), mc\_platform\_Initialize(), mc\_rwlock\_init(), syncListNodeInit(), and syncListNodeNew().

**5.39.1.15 #define COND\_WAIT(cond, mutex) pthread\_cond\_wait(cond, mutex )**

Definition at line 204 of file macros.h.

Referenced by acc\_MessageHandlerThread(), acc\_Thread(), agent\_mailbox\_WaitRetrieve(), ams\_Thread(), df\_Thread(), MC\_Barrier(), MC\_CondWait(), MC\_MainLoop(), mc\_platform\_Initialize(), mc\_rwlock\_rdlock(), mc\_rwlock\_wrlock(), MC\_SterControl(), MC\_WaitAgent(), MC\_WaitSignal(), and message\_Send().

**5.39.1.16 #define GET\_THREAD\_MODE(a, b) ( (a & (1<<b)) / (1<<b) )**

Definition at line 112 of file macros.h.

Referenced by MC\_End(), and mc\_platform\_Initialize().

**5.39.1.17 #define MUTEX\_DESTROY(mutex) pthread\_mutex\_destroy(mutex)**

Definition at line 179 of file macros.h.

Referenced by acc\_Destroy(), agent\_Destroy(), agent\_Initialize(), ams\_Destroy(), barrier\_node\_Destroy(), df\_Destroy(), df\_request\_list\_Destroy(), df\_request\_list\_node\_Destroy(), df\_request\_search\_Destroy(), mc\_platform\_Destroy(), mc\_rwlock\_destroy(), syncListDestroy(), and syncListNodeDestroy().

**5.39.1.18 #define MUTEX\_INIT(mutex) pthread\_mutex\_init(mutex, NULL)**

Definition at line 176 of file macros.h.

Referenced by acc\_Initialize(), agent\_Copy(), agent\_Initialize(), agent\_New(), agent\_NewBinary(), ams\_Initialize(), barrier\_node\_Initialize(), df\_Initialize(), df\_request\_list\_New(), df\_request\_list\_node\_New(), df\_request\_search\_New(), mc\_platform\_Initialize(), MC\_RegisterService(), mc\_rwlock\_init(), syncListInit(), syncListNodeInit(), and syncListNodeNew().

**5.39.1.19 #define MUTEX\_LOCK(mutex)**

**Value:**

```
if (pthread_mutex_lock( mutex )) \
    fprintf(stderr, "pthread lock error: %s:%d\n", __FILE__, __LINE__)
```

Definition at line 182 of file macros.h.

Referenced by acc\_MessageHandlerThread(), acc\_Thread(), agent\_Copy(), agent\_Destroy(), agent\_mailbox\_WaitRetrieve(), agent\_queue\_Flush(), agent\_RunChScriptThread(), ams\_ManageAgentList(), ams\_Print(), ams\_Thread(), AP\_QUEUE\_SEARCH\_TEMPLATE(), AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), df\_Destroy(), df\_node\_Destroy(), df\_request\_list\_Pop(), df\_SearchForService(), df\_Thread(), handler\_QUIT(), interpreter\_variable\_data\_Initialize(), listen\_Thread(), MC\_AddAgent(), MC\_Barrier(), MC\_CallAgentFunc(), MC\_CallAgentFuncArg(), MC\_CallAgentFuncV(), MC\_CallAgentFuncVar(), MC\_CondBroadcast(), MC\_CondReset(), MC\_CondSignal(), MC\_CondWait(), MC\_End(), MC\_GetAgentName(), MC\_GetAgentStatus(), MC\_GetAllAgents(), MC\_HaltAgency(), MC\_MainLoop(), MC\_MutexLock(), mc\_platform\_Initialize(), MC\_PrintAgentCode(), MC\_ResetSignal(), MC\_ResumeAgency(), MC\_RetrieveAgent(), MC\_RetrieveAgentCode(), mc\_rwlock\_rdlock(), mc\_rwlock\_rdunlock(), mc\_rwlock\_wrlock(), mc\_rwlock\_wrunlock(), MC\_SendAgent(), MC\_SendAgentFile(), MC\_SendAgentMigrationMessageFile(), MC\_SendSteerCommand(), MC\_SetAgentStatus(), MC\_Steer(), MC\_SteerControl(), MC\_SyncDelete(), MC\_SyncInit(), MC\_WaitAgent(), MC\_WaitRetrieveAgent(), MC\_WaitSignal(), message\_queue\_SendOutgoing(), message\_Send(), and request\_handler\_DEREGISTER().

**5.39.1.20 #define MUTEX\_NEW(mutex)**

**Value:**

```
mutex = (pthread_mutex_t*)malloc(sizeof(pthread_mutex_t)); \
    if (mutex == NULL) \
        fprintf(stderr, "Memory Error. %s:%d\n", __FILE__, __LINE__); \
```

Definition at line 187 of file macros.h.

Referenced by agent\_New().

#### 5.39.1.21 **#define MUTEX\_T pthread\_mutex\_t**

Definition at line 174 of file macros.h.

Referenced by acc\_Initialize(), agent\_Copy(), agent\_Initialize(), agent\_NewBinary(), ams\_Initialize(), barrier\_node\_Initialize(), df\_Initialize(), df\_request\_list\_New(), df\_request\_list\_node\_New(), df\_request\_search\_New(), mc\_platform\_Initialize(), MC\_RegisterService(), mc\_rwlock\_init(), syncListInit(), syncListNodeInit(), and syncListNodeNew().

#### 5.39.1.22 **#define MUTEX\_UNLOCK(mutex) pthread\_mutex\_unlock( mutex )**

Definition at line 185 of file macros.h.

Referenced by acc\_MessageHandlerThread(), acc\_Thread(), agent\_mailbox\_WaitRetrieve(), agent\_queue\_Flush(), agent\_RunChScriptThread(), ams\_ManageAgentList(), ams\_Print(), ams\_Thread(), AP\_QUEUE\_SEARCH\_TEMPLATE(), AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), df\_request\_list\_Pop(), df\_SearchForService(), df\_Thread(), handler\_QUIT(), interpreter\_variable\_data\_Initialize(), listen\_Thread(), MC\_AddAgent(), MC\_Barrier(), MC\_CallAgentFunc(), MC\_CallAgentFuncArg(), MC\_CallAgentFuncV(), MC\_CallAgentFuncVar(), MC\_CondBroadcast(), MC\_CondReset(), MC\_CondSignal(), MC\_CondWait(), MC\_End(), MC\_GetAgentName(), MC\_GetAgentStatus(), MC\_GetAllAgents(), MC\_HaltAgency(), MC\_MainLoop(), MC\_MutexUnlock(), mc\_platform\_Initialize(), MC\_PrintAgentCode(), MC\_ResetSignal(), MC\_ResumeAgency(), MC\_RetrieveAgent(), MC\_RetrieveAgentCode(), mc\_rwlock\_rdlock(), mc\_rwlock\_rdunlock(), mc\_rwlock\_wrlock(), mc\_rwlock\_wrunlock(), MC\_SendAgent(), MC\_SendAgentFile(), MC\_SendAgentMigrationMessageFile(), MC\_SendSteerCommand(), MC\_SetAgentStatus(), MC\_Steer(), MC\_SteerControl(), MC\_SyncDelete(), MC\_SyncInit(), MC\_WaitAgent(), MC\_WaitRetrieveAgent(), MC\_WaitSignal(), message\_queue\_SendOutgoing(), message\_Send(), and request\_handler\_DEREGISTER().

#### 5.39.1.23 **#define PTHREAD\_STACK\_SIZE 131072**

Definition at line 139 of file macros.h.

#### 5.39.1.24 **#define RWLOCK\_DESTROY(rwlock) mc\_rwlock\_destroy(rwlock)**

Definition at line 269 of file macros.h.

Referenced by barrier\_queue\_Destroy(), and syncListDestroy().

#### 5.39.1.25 **#define RWLOCK\_INIT(rwlock) mc\_rwlock\_init(rwlock)**

Definition at line 261 of file macros.h.

Referenced by barrier\_queue\_New(), and syncListInit().

#### 5.39.1.26 **#define RWLOCK\_RDLOCK(rwlock) mc\_rwlock\_rdlock(rwlock)**

Definition at line 287 of file macros.h.

Referenced by barrier\_queue\_Get(), and syncListFind().



**5.39.1.27 #define RWLOCK\_RDUNLOCK(rwlock) mc\_rwlock\_rdunlock(rwlock)**

Definition at line 289 of file macros.h.

Referenced by barrier\_queue\_Get(), and syncListFind().

**5.39.1.28 #define RWLOCK\_T mc\_rwlock\_t**

Definition at line 254 of file macros.h.

Referenced by barrier\_queue\_New(), and syncListInit().

**5.39.1.29 #define RWLOCK\_WRLOCK(rwlock) mc\_rwlock\_wrlock(rwlock)**

Definition at line 291 of file macros.h.

Referenced by barrier\_queue\_Add(), barrier\_queue\_Delete(), syncListAddNode(), syncListDelete(), syncListDestroy(), and syncListRemove().

**5.39.1.30 #define RWLOCK\_WRUNLOCK(rwlock) mc\_rwlock\_wrunlock(rwlock)**

Definition at line 293 of file macros.h.

Referenced by barrier\_queue\_Add(), barrier\_queue\_Delete(), syncListAddNode(), syncListDelete(), syncListDestroy(), and syncListRemove().

**5.39.1.31 #define SEMAPHORE\_DESTROY(sem) sem\_destroy(sem)**

Definition at line 239 of file macros.h.

Referenced by syncListNodeDestroy().

**5.39.1.32 #define SEMAPHORE\_INIT(sem) sem\_init(sem, 0, 0)**

Definition at line 236 of file macros.h.

Referenced by syncListNodeInit(), and syncListNodeNew().

**5.39.1.33 #define SEMAPHORE\_POST(sem) sem\_post(sem)**

Definition at line 244 of file macros.h.

Referenced by MC\_SemaphorePost().

**5.39.1.34 #define SEMAPHORE\_T sem\_t**

Definition at line 234 of file macros.h.

Referenced by syncListNodeInit(), and syncListNodeNew().

**5.39.1.35 #define SEMAPHORE\_WAIT(sem) sem\_wait(sem)**

Definition at line 242 of file macros.h.

Referenced by MC\_SemaphoreWait().

**5.39.1.36 #define SET\_THREAD\_OFF(a, b) a = (a & ~(1<<b)))**

Definition at line 116 of file macros.h.

Referenced by MC\_SetThreadOff(), and MC\_SetThreadsAllOff().

**5.39.1.37 #define SET\_THREAD\_ON(a, b) a = (a | (1<<b))**

Definition at line 115 of file macros.h.

Referenced by MC\_SetThreadOn(), and MC\_SetThreadsAllOn().

**5.39.1.38 #define SIGNAL(cond, mutex, action)**

**Value:**

```
pthread_mutex_lock( mutex ); \
action; \
pthread_cond_signal( cond ); \
pthread_mutex_unlock( mutex )
```

Definition at line 220 of file macros.h.

Referenced by agent\_RunChScriptThread(), df\_Add(), df\_AddRequest(), and request\_handler\_SEARCH().

**5.39.1.39 #define SLEEP\_QUEUE(queue)**

**Value:**

```
if (pthread_mutex_lock( queue->thread_mutex )) \
printf("pthread lock error: %s:%d\n", __FILE__, __LINE__); \
pthread_cond_wait( queue->touched_signal, queue->thread_mutex )
```

Definition at line 307 of file macros.h.

**5.39.1.40 #define SLEEP\_RESET(queue) pthread\_mutex\_unlock( queue->thread\_mutex )**

Definition at line 311 of file macros.h.

**5.39.1.41 #define SOCKET\_ERROR()**

**Value:**

```
printf("Socket error. %s:%d\n", __FILE__, __LINE__, errno); \
sleep(500)
```

Definition at line 132 of file macros.h.

Referenced by listen\_Thread(), mc\_platform\_Destroy(), message\_InitializeFromConnection(), message\_send\_Thread(), mtp\_http\_InitializeFromConnection(), net\_bind(), and net\_connect().

#### 5.39.1.42 #define STRUCT(name, members)

**Value:**

```
typedef struct name##_s { \
    members \
} name##_t; \
typedef name##_t* name##_p;
```

Definition at line 120 of file macros.h.

#### 5.39.1.43 #define THREAD\_CANCEL(thread\_handle) pthread\_cancel( thread\_handle )

Definition at line 155 of file macros.h.

Referenced by MC\_End().

#### 5.39.1.44 #define THREAD\_CREATE(thread\_handle, function, arg)

**Value:**

```
while(pthread_create( \
    thread_handle, \
    &attr, \
    function, \
    (void*) arg \
) < 0) { \
    printf("pthread_create failed. Trying again...\n"); \
}
```

Definition at line 145 of file macros.h.

Referenced by acc\_Start(), acc\_Thread(), agent\_RunChScript(), ams\_Start(), cmd\_prompt\_Start(), df\_Start(), MC\_AddStationaryAgent(), and message\_Send().

#### 5.39.1.45 #define THREAD\_DETACH(thread\_handle)

**Value:**

```
if(pthread_detach(thread_handle) < 0) { \
    printf("pthread_detach failed. %s:%d\n", __FILE__, __LINE__); \
}
```

Definition at line 161 of file macros.h.

Referenced by acc\_Thread(), and message\_Send().

#### 5.39.1.46 #define THREAD\_EXIT() pthread\_exit(NULL)

Definition at line 166 of file macros.h.

Referenced by acc\_MessageHandlerThread(), acc\_Thread(), ams\_Thread(), df\_Thread(), and listen\_Thread().

**5.39.1.47 #define THREAD\_JOIN(thread\_handle) pthread\_join( thread\_handle, NULL )**

Definition at line 158 of file macros.h.

Referenced by MC\_End().

**5.39.1.48 #define WAKE\_QUEUE(queue, action)**

**Value:**

```
if (pthread_mutex_trylock( queue->lock ) == 0) {    \
    action;                                           \
    pthread_cond_signal( queue->cond);               \
    pthread_mutex_unlock( queue->lock);              \
}
```

Definition at line 301 of file macros.h.

**5.39.1.49 #define WARN(message)**

**Value:**

```
fprintf(stderr, "\nWARNING: "); \
    fprintf(stderr, message ); \
    fprintf(stderr, " %s:%d\n", __FILE__, __LINE__ )
```

Definition at line 480 of file macros.h.

Referenced by agent\_RunChScriptThread(), and message\_InitializeFromAgent().

## 5.40 /home/dko/Projects/mobilec/src/include/mc\_error.h File Reference

### Typedefs

- typedef enum [error\\_code\\_e](#) [error\\_code\\_t](#)

### Enumerations

- enum [error\\_code\\_e](#) {  
    [MC\\_SUCCESS](#) = 0, [MC\\_ERR](#), [MC\\_ERR\\_CONNECT](#), [MC\\_ERR\\_PARSE](#),  
    [MC\\_ERR\\_EMPTY](#), [MC\\_ERR\\_INVALID](#), [MC\\_ERR\\_INVALID\\_ARGS](#), [MC\\_ERR\\_NOT\\_FOUND](#),  
    [MC\\_ERR\\_MEMORY](#), [MC\\_ERR\\_SEND](#), [MC\\_WARN\\_DUPLICATE](#), [MC\\_SUCCESS](#) = 0,  
    [ERR](#), [MC\\_ERR\\_CONNECT](#), [MC\\_ERR\\_PARSE](#), [MC\\_ERR\\_EMPTY](#),  
    [MC\\_ERR\\_INVALID](#), [MC\\_ERR\\_INVALID\\_ARGS](#), [MC\\_ERR\\_NOT\\_FOUND](#), [MC\\_ERR\\_MEMORY](#),  
    [MC\\_ERR\\_SEND](#), [MC\\_WARN\\_DUPLICATE](#) }

#### 5.40.1 Typedef Documentation

##### 5.40.1.1 typedef enum [error\\_code\\_e](#) [error\\_code\\_t](#)

#### 5.40.2 Enumeration Type Documentation

##### 5.40.2.1 enum [error\\_code\\_e](#)

Enumerator:

*[MC\\_SUCCESS](#)*  
*[MC\\_ERR](#)*  
*[MC\\_ERR\\_CONNECT](#)*  
*[MC\\_ERR\\_PARSE](#)*  
*[MC\\_ERR\\_EMPTY](#)*  
*[MC\\_ERR\\_INVALID](#)*  
*[MC\\_ERR\\_INVALID\\_ARGS](#)*  
*[MC\\_ERR\\_NOT\\_FOUND](#)*  
*[MC\\_ERR\\_MEMORY](#)*  
*[MC\\_ERR\\_SEND](#)*  
*[MC\\_WARN\\_DUPLICATE](#)*  
*[MC\\_SUCCESS](#)*  
*[ERR](#)*  
*[MC\\_ERR\\_CONNECT](#)*  
*[MC\\_ERR\\_PARSE](#)*  
*[MC\\_ERR\\_EMPTY](#)*

*MC\_ERR\_INVALID*  
*MC\_ERR\_INVALID\_ARGS*  
*MC\_ERR\_NOT\_FOUND*  
*MC\_ERR\_MEMORY*  
*MC\_ERR\_SEND*  
*MC\_WARN\_DUPLICATE*

Definition at line 41 of file mc\_error.h.

## 5.41 /home/dko/Projects/mobilec/src/include/mc\_platform.h File Reference

```
#include "acc.h"
#include "ams.h"
#include "barrier.h"
#include "cmd_prompt.h"
#include "config.h"
#include "connection.h"
#include "data_structures.h"
#include "df.h"
#include "libmc.h"
#include "../mc_sync/sync_list.h"
#include ../include/macros.h"
#include ../mc_list/list.h"
#include ../include/mc_rwlock.h"
#include ../include/ap_queue_template.h"
#include ../include/data_structures.h"
#include "asm_node.h"
```

### Data Structures

- struct [mc\\_platform\\_s](#)

### Typedefs

- typedef struct [mc\\_platform\\_s](#) [mc\\_platform\\_t](#)
- typedef [mc\\_platform\\_t](#) \* [mc\\_platform\\_p](#)

### Functions

- [mc\\_platform\\_p](#) [mc\\_platform\\_Initialize](#) ([MCAgency\\_t](#) agency, [ChOptions\\_t](#) \*ch\_options)
- [int](#) [mc\\_platform\\_Destroy](#) ([mc\\_platform\\_p](#) mc\_platform)

#### 5.41.1 Typedef Documentation

##### 5.41.1.1 typedef [mc\\_platform\\_t](#)\* [mc\\_platform\\_p](#)

Definition at line 124 of file [mc\\_platform.h](#).

### 5.41.1.2 typedef struct mc\_platform\_s mc\_platform\_t

Definition at line 123 of file mc\_platform.h.

## 5.41.2 Function Documentation

### 5.41.2.1 int mc\_platform\_Destroy (mc\_platform\_p mc\_platform)

Definition at line 249 of file mc\_platform.c.

References mc\_platform\_s::acc, acc\_Destroy(), mc\_platform\_s::agent\_queue, mc\_platform\_s::ams, ams\_Destroy(), mc\_platform\_s::barrier\_queue, barrier\_queue\_Destroy(), mc\_platform\_s::cmd\_prompt, cmd\_prompt\_Destroy(), COND\_DESTROY, mc\_platform\_s::connection\_queue, mc\_platform\_s::df, df\_Destroy(), mc\_platform\_s::giant\_cond, mc\_platform\_s::giant\_lock, mc\_platform\_s::hostname, mc\_platform\_s::interp\_options, mc\_platform\_s::interpreter\_queue, mc\_platform\_s::MC\_signal\_cond, mc\_platform\_s::MC\_signal\_lock, mc\_platform\_s::MC\_steer\_cond, mc\_platform\_s::MC\_steer\_lock, MC\_SUCCESS, mc\_platform\_s::MC\_sync\_cond, mc\_platform\_s::MC\_sync\_lock, mc\_platform\_s::message\_queue, MUTEX\_DESTROY, mc\_platform\_s::quit\_cond, mc\_platform\_s::quit\_lock, SOCKET\_ERROR, mc\_platform\_s::sockfd, mc\_platform\_s::syncList, and syncListDestroy().

Referenced by MC\_End().

### 5.41.2.2 mc\_platform\_p mc\_platform\_Initialize (MCAgency\_t agency, ChOptions\_t \* ch\_options)

Definition at line 54 of file mc\_platform.c.

References mc\_platform\_s::acc, acc\_Initialize(), acc\_Start(), mc\_platform\_s::agency, agent\_ChScriptInitVar(), mc\_platform\_s::agent\_queue, mc\_platform\_s::ams, ams\_Initialize(), ams\_Start(), mc\_platform\_s::barrier\_queue, barrier\_queue\_New(), agency\_s::bluetooth, mc\_platform\_s::bluetooth, CHECK\_NULL, mc\_platform\_s::cmd\_prompt, cmd\_prompt\_Initialize(), cmd\_prompt\_Start(), COND\_INIT, COND\_T, COND\_WAIT, mc\_platform\_s::connection\_queue, agency\_s::default\_agentstatus, mc\_platform\_s::default\_agentstatus, DEFAULT\_HOSTNAME\_LENGTH, mc\_platform\_s::df, df\_Initialize(), df\_Start(), mc\_platform\_s::err, GET\_THREAD\_MODE, mc\_platform\_s::giant, mc\_platform\_s::giant\_cond, mc\_platform\_s::giant\_lock, mc\_platform\_s::hostname, agency\_s::initInterps, mc\_platform\_s::initInterps, mc\_platform\_s::interp\_options, mc\_platform\_s::interpreter\_queue, agency\_s::last\_error, MC\_ERR\_MEMORY, MC\_NO\_SIGNAL, mc\_platform, mc\_platform\_s::MC\_signal, mc\_platform\_s::MC\_signal\_cond, mc\_platform\_s::MC\_signal\_lock, mc\_platform\_s::MC\_steer\_cond, mc\_platform\_s::MC\_steer\_lock, mc\_platform\_s::MC\_sync\_cond, mc\_platform\_s::MC\_sync\_lock, MC\_THREAD\_ACC, MC\_THREAD\_ALL, MC\_THREAD\_AMS, MC\_THREAD\_CP, MC\_THREAD\_DF, mc\_platform\_s::message\_queue, MUTEX\_INIT, MUTEX\_LOCK, MUTEX\_T, MUTEX\_UNLOCK, mc\_platform\_s::port, agency\_s::portno, mc\_platform\_s::quit, mc\_platform\_s::quit\_cond, mc\_platform\_s::quit\_lock, agency\_s::stack\_size, mc\_platform\_s::stack\_size, mc\_platform\_s::syncList, syncListInit(), and agency\_s::threads.

Referenced by MC\_Initialize().



## 5.42 /home/dko/Projects/mobilec/src/include/mc\_rwlock.h File Reference

```
#include "macros.h"
```

### Data Structures

- struct [mc\\_rwlock\\_s](#)

### Typedefs

- typedef struct [mc\\_rwlock\\_s](#) [mc\\_rwlock\\_t](#)
- typedef [mc\\_rwlock\\_t](#) \* [mc\\_rwlock\\_p](#)

### Functions

- [int mc\\_rwlock\\_init](#) ([mc\\_rwlock\\_p](#) *rwlock*)
- [int mc\\_rwlock\\_destroy](#) ([mc\\_rwlock\\_p](#) *rwlock*)
- [int mc\\_rwlock\\_rdlock](#) ([mc\\_rwlock\\_p](#) *rwlock*)
- [int mc\\_rwlock\\_rdunlock](#) ([mc\\_rwlock\\_p](#) *rwlock*)
- [int mc\\_rwlock\\_wrlock](#) ([mc\\_rwlock\\_p](#) *rwlock*)
- [int mc\\_rwlock\\_wrunlock](#) ([mc\\_rwlock\\_p](#) *rwlock*)

#### 5.42.1 Typedef Documentation

##### 5.42.1.1 typedef [mc\\_rwlock\\_t](#)\* [mc\\_rwlock\\_p](#)

Definition at line 46 of file [mc\\_rwlock.h](#).

##### 5.42.1.2 typedef struct [mc\\_rwlock\\_s](#) [mc\\_rwlock\\_t](#)

#### 5.42.2 Function Documentation

##### 5.42.2.1 [int mc\\_rwlock\\_destroy](#) ([mc\\_rwlock\\_p](#) *rwlock*)

Definition at line 66 of file [mc\\_rwlock.c](#).

References [mc\\_rwlock\\_s::cond](#), [COND\\_DESTROY](#), [mc\\_rwlock\\_s::lock](#), and [MUTEX\\_DESTROY](#).

##### 5.42.2.2 [int mc\\_rwlock\\_init](#) ([mc\\_rwlock\\_p](#) *rwlock*)

Definition at line 46 of file [mc\\_rwlock.c](#).

References [CHECK\\_NULL](#), [mc\\_rwlock\\_s::cond](#), [COND\\_INIT](#), [COND\\_T](#), [mc\\_rwlock\\_s::lock](#), [MC\\_ERR\\_MEMORY](#), [MUTEX\\_INIT](#), [MUTEX\\_T](#), [mc\\_rwlock\\_s::num\\_readers](#), [mc\\_rwlock\\_s::write\\_flag](#), and [mc\\_rwlock\\_s::write\\_request](#).

#### 5.42.2.3 `int mc_rwlock_rdlock (mc_rwlock_p rwlock)`

Definition at line 79 of file `mc_rwlock.c`.

References `mc_rwlock_s::cond`, `COND_WAIT`, `mc_rwlock_s::lock`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_rwlock_s::num_readers`, `mc_rwlock_s::write_flag`, and `mc_rwlock_s::write_request`.

#### 5.42.2.4 `int mc_rwlock_rdunlock (mc_rwlock_p rwlock)`

Definition at line 94 of file `mc_rwlock.c`.

References `mc_rwlock_s::cond`, `COND_SIGNAL`, `mc_rwlock_s::lock`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `mc_rwlock_s::num_readers`.

#### 5.42.2.5 `int mc_rwlock_wrlock (mc_rwlock_p rwlock)`

Definition at line 107 of file `mc_rwlock.c`.

References `mc_rwlock_s::cond`, `COND_WAIT`, `mc_rwlock_s::lock`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_rwlock_s::num_readers`, `mc_rwlock_s::write_flag`, and `mc_rwlock_s::write_request`.

#### 5.42.2.6 `int mc_rwlock_wrunlock (mc_rwlock_p rwlock)`

Definition at line 126 of file `mc_rwlock.c`.

References `mc_rwlock_s::cond`, `COND_SIGNAL`, `mc_rwlock_s::lock`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `mc_rwlock_s::write_flag`.

## 5.43 /home/dko/Projects/mobilec/src/include/message.h File Reference

```
#include "mtp_http.h"
#include <mxml.h>
#include <netinet/in.h>
#include "config.h"
#include "security/interface.h"
```

### Data Structures

- struct [message\\_s](#)
- struct [message\\_send\\_arg\\_s](#)

### Typedefs

- typedef enum [message\\_type\\_e](#) [message\\_type\\_t](#)
- typedef struct [message\\_s](#) [message\\_t](#)
- typedef [message\\_t](#) \* [message\\_p](#)
- typedef struct [message\\_send\\_arg\\_s](#) [message\\_send\\_arg\\_t](#)

### Enumerations

- enum [message\\_type\\_e](#) {  
    [RELAY](#), [REQUEST](#), [SUBSCRIBE](#), [CANCEL](#),  
    [N\\_UNDRSTD](#), [MOBILE\\_AGENT](#), [QUER\\_IF](#), [QUER\\_REF](#),  
    [AGENT\\_UPDATE](#), [RETURN\\_MSG](#), [FIPA\\_ACL](#), [ENCRYPTED\\_DATA](#),  
    [ENCRYPTION\\_INITIALIZE](#), [REQUEST\\_ENCRYPTION\\_INITIALIZE](#), [NUM\\_MESSAGE\\_-](#)  
    [TYPE](#), [MSG\\_SEC](#) }

### Functions

- [message\\_p](#) [message\\_New](#) (void)
- [message\\_p](#) [message\\_Copy](#) ([message\\_p](#) src)
- int [message\\_InitializeFromAgent](#) (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#), [message\\_p](#) message, struct [agent\\_s](#) \*[agent](#))
- int [message\\_InitializeFromConnection](#) (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#), [message\\_p](#) message, struct [connection\\_s](#) \*[connection](#))
- int [message\\_InitializeFromString](#) (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#), [message\\_p](#) message, const char \*[string](#), const char \*[destination\\_host](#), int [destination\\_port](#), const char \*[target](#))
- int [message\\_Destroy](#) ([message\\_p](#) message)
- int [auth\\_rece\\_send\\_msg](#) (int [sockfd](#), char \*[hostname](#), char \*[message](#), char \*[privkey](#), char \*[known\\_-](#)  
    [host\\_filename](#))
- int [message\\_Send](#) (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#), [message\\_p](#) message, char \*[privatekey](#))
- void \* [message\\_send\\_Thread](#) (void \*[arg](#))
- int [http\\_to\\_hostport](#) (const char \*[http\\_str](#), char \*\*[host](#), int \*[port](#), char \*\*[target](#))

### 5.43.1 Typedef Documentation

#### 5.43.1.1 typedef message\_t\* message\_p

Definition at line 115 of file message.h.

#### 5.43.1.2 typedef struct message\_send\_arg\_s message\_send\_arg\_t

#### 5.43.1.3 typedef struct message\_s message\_t

#### 5.43.1.4 typedef enum message\_type\_e message\_type\_t

### 5.43.2 Enumeration Type Documentation

#### 5.43.2.1 enum message\_type\_e

Enumerator:

*RELAY*  
*REQUEST*  
*SUBSCRIBE*  
*CANCEL*  
*N\_UNDRSTD*  
*MOBILE\_AGENT*  
*QUER\_IF*  
*QUER\_REF*  
*AGENT\_UPDATE*  
*RETURN\_MSG*  
*FIPA\_ACL*  
*ENCRYPTED\_DATA*  
*ENCRYPTION\_INITIALIZE*  
*REQUEST\_ENCRYPTION\_INITIALIZE*  
*NUM\_MESSAGE\_TYPE*  
*MSG\_SEC*

Definition at line 52 of file message.h.

### 5.43.3 Function Documentation

#### 5.43.3.1 int auth\_rece\_send\_msg (int sockfd, char \* hostname, char \* message, char \* privkey, char \* known\_host\_filename)

Definition at line 453 of file message.c.

References `aes_en_de()`, `initiate_migration_process()`, `read_known_host_file()`, and `send_AES_en_MA()`.

Referenced by `message_send_Thread()`.

**5.43.3.2 int http\_to\_hostport (const char \* *http\_str*, char \*\* *host*, int \* *port*, char \*\* *target*)**

Definition at line 298 of file message.c.

References MC\_ERR\_PARSE.

Referenced by MC\_AclSend().

**5.43.3.3 message\_p message\_Copy (message\_p *src*)**

Definition at line 97 of file message.c.

**5.43.3.4 int message\_Destroy (message\_p *message*)**

Definition at line 414 of file message.c.

References message\_s::addr, message\_s::agent\_xml\_flag, message\_s::from\_address, MC\_SUCCESS, message\_s::message\_body, mxmlDelete(), message\_s::target, message\_s::to\_address, message\_s::update\_name, and message\_s::xml\_root.

Referenced by acc\_connection\_Thread(), acc\_MessageHandlerThread(), ams\_ManageAgentList(), MC\_LoadAgentFromFile(), MC\_SendAgent(), MC\_SendAgentFile(), MC\_SendAgentMigrationMessage(), MC\_SendAgentMigrationMessageFile(), and message\_InitializeFromConnection().

**5.43.3.5 int message\_InitializeFromAgent (struct mc\_platform\_s \* *mc\_platform*, message\_p *message*, struct agent\_s \* *agent*)**

Referenced by ams\_ManageAgentList().

**5.43.3.6 int message\_InitializeFromConnection (struct mc\_platform\_s \* *mc\_platform*, message\_p *message*, struct connection\_s \* *connection*)****5.43.3.7 int message\_InitializeFromString (struct mc\_platform\_s \* *mc\_platform*, message\_p *message*, const char \* *string*, const char \* *destination\_host*, int *destination\_port*, const char \* *target*)**

Referenced by MC\_LoadAgentFromFile(), MC\_SendAgent(), MC\_SendAgentFile(), MC\_SendAgentMigrationMessage(), and MC\_SendAgentMigrationMessageFile().

**5.43.3.8 message\_p message\_New (void)**

Definition at line 73 of file message.c.

References message\_s::addr, message\_s::agent\_xml\_flag, CHECK\_NULL, message\_s::connect\_id, message\_s::from\_address, message\_s::http\_type, message\_s::isHTTP, message\_s::message\_body, message\_s::message\_id, message\_s::message\_type, message\_s::target, message\_s::to\_address, message\_s::update\_name, message\_s::update\_num, message\_s::xml\_payload, and message\_s::xml\_root.

Referenced by acc\_connection\_Thread(), ams\_ManageAgentList(), MC\_LoadAgentFromFile(), MC\_SendAgent(), MC\_SendAgentFile(), MC\_SendAgentMigrationMessage(), MC\_SendAgentMigrationMessageFile(), and mtp\_http\_CreateMessage().

#### 5.43.3.9 `int message_Send (struct mc_platform_s * mc_platform, message_p message, char * privatekey)`

Definition at line 575 of file message.c.

References `mc_platform_s::acc`, `COND_WAIT`, `message_send_arg_s::mc_platform`, `message_send_arg_s::message`, `message_send_Thread()`, `MSG_THREADS`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `message_send_arg_s::privatekey`, `THREAD_CREATE`, and `THREAD_DETACH`.

Referenced by `acc_MessageHandlerThread()`, `MC_AclSend()`, and `message_queue_SendOutgoing()`.

#### 5.43.3.10 `void* message_send_Thread (void * arg)`

Definition at line 648 of file message.c.

References `mc_platform_s::agency`, `auth_rece_send_msg()`, `mc_platform_s::bluetooth`, `buf`, `CHECK_NULL`, `dynstring_Append()`, `dynstring_Destroy()`, `dynstring_New()`, `agency_s::known_host_filename`, `mc_platform`, `dynstring_s::message`, `message_s::message_body`, `MSG_THREAD_EXIT`, `mtp_http_ComposeMessage()`, `mtp_http_Destroy()`, `mtp_http_New()`, `mtp_http_Parse()`, `port`, `send`, `SOCKET_ERROR`, `SOCKET_INPUT_SIZE`, `strtok_r`, and `message_s::to_address`.

Referenced by `message_Send()`.

## 5.44 /home/dko/Projects/mobilec/src/include/mobilec.h File Reference

```
#include <dynstring.h>
#include <fipa_acl.h>
#include <fipa_comm.h>
```

## 5.45 /home/dko/Projects/mobilec/src/include/mtp\_http.h File Reference

### Data Structures

- struct [mtp\\_http\\_content\\_s](#)
- struct [mtp\\_http\\_s](#)

### Defines

- #define [SOCKET\\_INPUT\\_SIZE](#) 4096

### Typedefs

- typedef struct [mtp\\_http\\_content\\_s](#) [mtp\\_http\\_content\\_t](#)
- typedef struct [mtp\\_http\\_s](#) [mtp\\_http\\_t](#)
- typedef [mtp\\_http\\_t](#) \* [mtp\\_http\\_p](#)

### Enumerations

- enum [http\\_status\\_code\\_e](#) {  
    [CONTINUE](#) = 100, [SWITCHING\\_PROTOCOLS](#), [PROCESSING](#), [OK](#) = 200,  
    [CREATED](#), [ACCEPTED](#), [NON\\_AUTHORITATIVE\\_INFORMATION](#), [NO\\_CONTENT](#),  
    [RESET\\_CONTENT](#), [PARTIAL\\_CONTENT](#), [MULTI\\_STATUS](#), [BAD\\_REQUEST](#) = 400,  
    [UNAUTHORIZED](#), [PAYMENT\\_REQUIRED](#), [FORBIDDEN](#), [NOT\\_FOUND](#),  
    [METHOD\\_NOT\\_ALLOWED](#), [NOT\\_ACCEPTABLE](#), [PROXY\\_AUTHENTICATION\\_REQUIRED](#), [REQUEST\\_TIMEOUT](#),  
    [CONFLICT](#), [GONE](#), [LENGTH\\_REQUIRED](#), [PRECONDITION\\_FAILED](#),  
    [REQUEST\\_ENTITY\\_TOO\\_LARGE](#), [REQUEST\\_URI\\_TOO\\_LONG](#), [UNSUPPORTED\\_MEDIA\\_TYPE](#), [REQUESTED\\_RANGE\\_NOT\\_SATISFIABLE](#),  
    [EXPECTATION\\_FAILED](#), [UNPROCESSABLE\\_ENTITY](#), [LOCKED](#), [FAILED\\_DEPENDANCY](#),  
    [UNORDERED\\_COLLECTION](#), [UPGRADE\\_REQUIRED](#), [RETRY\\_WITH](#) }  
    *http return status codes*
- enum [http\\_performative\\_e](#) {  
    [HTTP\\_PERFORMATIVE\\_UNDEF](#) = -1, [HTTP\\_PERFORMATIVE\\_ZERO](#) = 0, [HTTP\\_HEAD](#),  
    [HTTP\\_GET](#),  
    [HTTP\\_POST](#), [HTTP\\_PUT](#), [HTTP\\_DELETE](#), [HTTP\\_TRACE](#),  
    [HTTP\\_OPTIONS](#), [HTTP\\_CONNECT](#), [HTTP\\_RESPONSE](#), [HTTP\\_NUM\\_PERFORMATIVES](#) }  
    *http 'verbs'*



## Functions

- const char \* [http\\_GetExpression](#) (const char \*string, char \*\*expr)  
*Parse an html expression.*
- int [http\\_ParseExpression](#) (const char \*expression\_string, char \*\*name, char \*\*value)  
*Parse an expression into its name and value.*
- const char \* [http\\_ParseRequest](#) (mtp\_http\_p http, const char \*string)
- const char \* [http\\_GetToken](#) (const char \*string, char \*\*token)
- int [mtp\\_http\\_Destroy](#) (mtp\_http\_p http)
- int [mtp\\_http\\_InitializeFromConnection](#) (struct mtp\_http\_s \*http, struct connection\_s \*connection, char \*privatekey)
- mtp\_http\_p [mtp\\_http\\_New](#) (void)
- const char \* [mtp\\_http\\_ParseHeader](#) (struct mtp\_http\_s \*http, const char \*string)
- int [mtp\\_http\\_Parse](#) (struct mtp\_http\_s \*http, const char \*string)
- int [mtp\\_http\\_ComposeMessage](#) (struct message\_s \*message)
- struct message\_s \* [mtp\\_http\\_CreateMessage](#) (mtp\_http\_t \*mtp\_http, char \*hostname, int port)

### 5.45.1 Define Documentation

#### 5.45.1.1 #define SOCKET\_INPUT\_SIZE 4096

Definition at line 147 of file mtp\_http.h.

Referenced by [message\\_InitializeFromConnection\(\)](#), [message\\_send\\_Thread\(\)](#), and [mtp\\_http\\_InitializeFromConnection\(\)](#).

### 5.45.2 Typedef Documentation

#### 5.45.2.1 typedef struct mtp\_http\_content\_s mtp\_http\_content\_t

#### 5.45.2.2 typedef mtp\_http\_t\* mtp\_http\_p

Definition at line 145 of file mtp\_http.h.

#### 5.45.2.3 typedef struct mtp\_http\_s mtp\_http\_t

### 5.45.3 Enumeration Type Documentation

#### 5.45.3.1 enum http\_performative\_e

http 'verbs'

Enumerator:

*HTTP\_PERFORMATIVE\_UNDEF*  
*HTTP\_PERFORMATIVE\_ZERO*  
*HTTP\_HEAD*  
*HTTP\_GET*

*HTTP\_POST*  
*HTTP\_PUT*  
*HTTP\_DELETE*  
*HTTP\_TRACE*  
*HTTP\_OPTIONS*  
*HTTP\_CONNECT*  
*HTTP\_RESPONSE*  
*HTTP\_NUM\_PERFORMATIVES*

Definition at line 90 of file mtp\_http.h.

#### 5.45.3.2 enum http\_status\_code\_e

http return status codes

**Enumerator:**

*CONTINUE*  
*SWITCHING\_PROTOCOLS*  
*PROCESSING*  
*OK*  
*CREATED*  
*ACCEPTED*  
*NON\_AUTHORITATIVE\_INFORMATION*  
*NO\_CONTENT*  
*RESET\_CONTENT*  
*PARTIAL\_CONTENT*  
*MULTI\_STATUS*  
*BAD\_REQUEST*  
*UNAUTHORIZED*  
*PAYMENT\_REQUIRED*  
*FORBIDDEN*  
*NOT\_FOUND*  
*METHOD\_NOT\_ALLOWED*  
*NOT\_ACCEPTABLE*  
*PROXY\_AUTHENTICATION\_REQUIRED*  
*REQUEST\_TIMEOUT*  
*CONFLICT*  
*GONE*  
*LENGTH\_REQUIRED*  
*PRECONDITION\_FAILED*  
*REQUEST\_ENTITY\_TOO\_LARGE*  
*REQUEST\_URI\_TOO\_LONG*

***UNSUPPORTED\_MEDIA\_TYPE***  
***REQUESTED\_RANGE\_NOT\_SATISFIABLE***  
***EXPECTATION\_FAILED***  
***UNPROCESSABLE\_ENTITY***  
***LOCKED***  
***FAILED\_DEPENDANCY***  
***UNORDERED\_COLLECTION***  
***UPGRADE\_REQUIRED***  
***RETRY\_WITH***

Definition at line 45 of file mtp\_http.h.

## 5.45.4 Function Documentation

### 5.45.4.1 `const char* http_GetExpression (const char * string, char ** expr)`

Parse an html expression.

#### Parameters

*string* (input) The html block of text: Will parse the first expression pointed to by 'string'.  
*expr* (output) The allocated expression

#### Returns

A pointer to the next expression segment of the string block, or NULL.

Definition at line 349 of file mtp\_http.c.

Referenced by mtp\_http\_Parse(), and mtp\_http\_ParseHeader().

### 5.45.4.2 `const char* http_GetToken (const char * string, char ** token)`

Definition at line 790 of file mtp\_http.c.

References cur.

Referenced by http\_ParseRequest().

### 5.45.4.3 `int http_ParseExpression (const char * expression_string, char ** name, char ** value)`

Parse an expression into its name and value.

#### Parameters

*expression\_string* (input) The expression  
*name* (output) An allocated name string or NULL  
*value* (output) An allocated value string or NULL

#### Returns

error\_code\_t type

**Note**

an http expression is something like 'Date: Mon, 23 May 2005 22:38:34 GMT'  
 ' where 'Date' is the name and the remainder of the string is the value

Definition at line 408 of file mtp\_http.c.

References CHECK\_NULL, MC\_ERR\_PARSE, and MC\_SUCCESS.

Referenced by mtp\_http\_Parse(), and mtp\_http\_ParseHeader().

**5.45.4.4 const char\* http\_ParseRequest (mtp\_http\_p *http*, const char \* *string*)**

Definition at line 701 of file mtp\_http.c.

References cur, HTTP\_CONNECT, HTTP\_DELETE, HTTP\_GET, http\_GetToken(), HTTP\_HEAD, HTTP\_OPTIONS, mtp\_http\_s::http\_performative, HTTP\_PERFORMATIVE\_UNDEF, HTTP\_POST, HTTP\_PUT, HTTP\_RESPONSE, HTTP\_TRACE, mtp\_http\_s::response\_code, mtp\_http\_s::response\_string, and mtp\_http\_s::target.

Referenced by mtp\_http\_ParseHeader().

**5.45.4.5 int mtp\_http\_ComposeMessage (struct message\_s \* *message*)**

Referenced by message\_send\_Thread().

**5.45.4.6 struct message\_s\* mtp\_http\_CreateMessage (mtp\_http\_t \* *mtp\_http*, char \* *hostname*, int *port*) [read]**

Definition at line 875 of file mtp\_http.c.

References buf, mtp\_http\_s::content, mtp\_http\_content\_s::content\_type, mtp\_http\_content\_s::data, dynstring\_Append(), dynstring\_Destroy(), dynstring\_New(), mtp\_http\_s::host, message\_s::isHTTP, dynstring\_s::len, dynstring\_s::message, message\_s::message\_body, message\_New(), mtp\_http\_s::message\_parts, PACKAGE\_VERSION, mtp\_http\_s::target, and message\_s::to\_address.

Referenced by MC\_AclSend().

**5.45.4.7 int mtp\_http\_Destroy (mtp\_http\_p *http*)**

Definition at line 54 of file mtp\_http.c.

References mtp\_http\_s::accept\_ranges, mtp\_http\_s::boundary, mtp\_http\_s::cache\_control, mtp\_http\_s::connection, mtp\_http\_s::content, mtp\_http\_s::content\_length, mtp\_http\_content\_s::content\_type, mtp\_http\_s::content\_type, mtp\_http\_content\_s::data, mtp\_http\_s::date, mtp\_http\_s::host, mtp\_http\_s::http\_version, mtp\_http\_s::message\_parts, mtp\_http\_s::mime\_version, mtp\_http\_s::response\_string, mtp\_http\_s::return\_code, SAFE\_FREE, mtp\_http\_s::server, mtp\_http\_s::target, and mtp\_http\_s::user\_agent.

Referenced by acc\_connection\_Thread(), MC\_AclSend(), message\_send\_Thread(), and mtp\_http\_InitializeFromConnection().

**5.45.4.8 int mtp\_http\_InitializeFromConnection (struct mtp\_http\_s \* *http*, struct connection\_s \* *connection*, char \* *privatekey*)**

Referenced by acc\_connection\_Thread().

#### 5.45.4.9 mtp\_http\_p mtp\_http\_New (void)

Definition at line 89 of file mtp\_http.c.

References CHECK\_NULL, and mtp\_http\_s::content.

Referenced by acc\_connection\_Thread(), MC\_AclSend(), message\_send\_Thread(), and mtp\_http\_InitializeFromConnection().

#### 5.45.4.10 int mtp\_http\_Parse (struct mtp\_http\_s \* *http*, const char \* *string*)

Definition at line 551 of file mtp\_http.c.

References mtp\_http\_s::boundary, mtp\_http\_s::content, mtp\_http\_s::content\_length, mtp\_http\_content\_s::content\_type, mtp\_http\_s::content\_type, mtp\_http\_content\_s::data, http\_GetExpression(), HTTP\_HEAD, http\_ParseExpression(), mtp\_http\_s::http\_performative, HTTP\_POST, HTTP\_PUT, HTTP\_RESPONSE, MC\_SUCCESS, mtp\_http\_s::message\_parts, mtp\_http\_ParseHeader(), and strdup.

Referenced by message\_send\_Thread(), and mtp\_http\_InitializeFromConnection().

#### 5.45.4.11 const char\* mtp\_http\_ParseHeader (struct mtp\_http\_s \* *http*, const char \* *string*)

Definition at line 467 of file mtp\_http.c.

References mtp\_http\_s::header\_length, http\_GetExpression(), HTTP\_PARSE\_EXPR, http\_ParseExpression(), http\_ParseRequest(), MC\_SUCCESS, and SAFE\_FREE.

Referenced by mtp\_http\_InitializeFromConnection(), and mtp\_http\_Parse().

## 5.46 /home/dko/Projects/mobilec/src/include/xml\_compose.h File Reference

```
#include "agent.h"
```

### Functions

- [mxml\\_node\\_t \\* agent\\_xml\\_compose \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_gaf\\_message \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_message \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_mobile\\_agent \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_agent\\_data \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_name \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_owner \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_home \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_wg\\_code \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_tasks \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_task \(agent\\_p agent, int index\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_data \(agent\\_p agent, int index, interpreter\\_variable\\_data\\_t \\*interp\\_variable\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_agent\\_code \(agent\\_p agent, int index\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_row \(interpreter\\_variable\\_data\\_t \\*interp\\_variable, int index\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_create\\_row\\_nodes \(void \\*data, int index, int \\*extent, ChType\\_t type, int dim, int extent\\_index\)](#)

### 5.46.1 Function Documentation

#### 5.46.1.1 mxml\_node\_t\* agent\_xml\_compose (agent\_p agent)

Definition at line 46 of file `xml_compose.c`.

References `agent_xml_compose__gaf_message()`, `MXML_ADD_AFTER`, `MXML_ADD_TO_PARENT`, `MXML_NO_CALLBACK`, `mxmlAdd()`, `mxmlLoadString()`, and `node`.

Referenced by `message_InitializeFromAgent()`.

#### 5.46.1.2 mxml\_node\_t\* agent\_xml\_compose\_\_agent\_code (agent\_p agent, int index)

Definition at line 521 of file `xml_compose.c`.

References `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_s::datastate`, `MXML_NO_PARENT`, `mxmlElementSetAttr()`, `mxmlNewElement()`, `node`, and `xml_new_cdata()`.

Referenced by `agent_xml_compose__tasks()`.

#### 5.46.1.3 mxml\_node\_t\* agent\_xml\_compose\_\_agent\_data (agent\_p agent)

Definition at line 150 of file `xml_compose.c`.

References `agent_xml_compose__home()`, `agent_xml_compose__name()`, `agent_xml_compose__owner()`, `agent_xml_compose__tasks()`, `agent_xml_compose__wg_code()`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlNewElement()`, and `node`.

Referenced by agent\_xml\_compose\_\_mobile\_agent().

#### 5.46.1.4 **mxml\_node\_t\* agent\_xml\_compose\_\_create\_row\_nodes** (void \* *data*, int *index*, int \* *extent*, ChType\_t *type*, int *dim*, int *extent\_index*)

Definition at line 572 of file xml\_compose.c.

References agent\_xml\_compose\_\_create\_row\_nodes(), buf, CH\_DATATYPE\_SIZE, CH\_DATATYPE\_VALUE\_STRING, MXML\_ADD\_AFTER, MXML\_ADD\_TO\_PARENT, MXML\_NO\_PARENT, mxm-  
lAdd(), mxmlElementSetAttr(), mxmlNewElement(), mxmlNewText(), node, and size.

Referenced by agent\_xml\_compose\_\_create\_row\_nodes(), and agent\_xml\_compose\_\_row().

#### 5.46.1.5 **mxml\_node\_t\* agent\_xml\_compose\_\_data** (agent\_p *agent*, int *index*, interpreter\_variable\_data\_t \* *interp\_variable*)

Definition at line 447 of file xml\_compose.c.

References agent\_xml\_compose\_\_row(), interpreter\_variable\_data\_s::array\_dim, buf, CH\_DATATYPE\_STRING, CH\_DATATYPE\_VALUE\_STRING, interpreter\_variable\_data\_s::data, interpreter\_variable\_data\_s::data\_type, MXML\_ADD\_AFTER, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), interpreter\_variable\_data\_s::name, and node.

Referenced by agent\_xml\_compose\_\_task().

#### 5.46.1.6 **mxml\_node\_t\* agent\_xml\_compose\_\_gaf\_message** (agent\_p *agent*)

Definition at line 66 of file xml\_compose.c.

References agent\_xml\_compose\_\_message(), MXML\_ADD\_AFTER, mxmlAdd(), mxmlNewElement(), and node.

Referenced by agent\_xml\_compose().

#### 5.46.1.7 **mxml\_node\_t\* agent\_xml\_compose\_\_home** (agent\_p *agent*)

Definition at line 253 of file xml\_compose.c.

References agent\_s::home, mxmlNewElement(), mxmlNewText(), and node.

Referenced by agent\_xml\_compose\_\_agent\_data().

#### 5.46.1.8 **mxml\_node\_t\* agent\_xml\_compose\_\_message** (agent\_p *agent*)

Definition at line 85 of file xml\_compose.c.

References agent\_s::agent\_type, agent\_xml\_compose\_\_mobile\_agent(), MC\_LOCAL\_AGENT, MC\_REMOTE\_AGENT, MC\_RETURN\_AGENT, MXML\_ADD\_AFTER, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), and node.

Referenced by agent\_xml\_compose\_\_gaf\_message().

#### 5.46.1.9 **mxml\_node\_t\* agent\_xml\_compose\_\_mobile\_agent** (agent\_p *agent*)

Definition at line 129 of file xml\_compose.c.

References `agent_xml_compose__agent_data()`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlNewElement()`, and `node`.

Referenced by `agent_xml_compose__message()`.

#### 5.46.1.10 `mxml_node_t* agent_xml_compose__name (agent_p agent)`

Definition at line 221 of file `xml_compose.c`.

References `mxmlNewElement()`, `mxmlNewText()`, `agent_s::name`, and `node`.

Referenced by `agent_xml_compose__agent_data()`.

#### 5.46.1.11 `mxml_node_t* agent_xml_compose__owner (agent_p agent)`

Definition at line 237 of file `xml_compose.c`.

References `mxmlNewElement()`, `mxmlNewText()`, `node`, and `agent_s::owner`.

Referenced by `agent_xml_compose__agent_data()`.

#### 5.46.1.12 `mxml_node_t* agent_xml_compose__row (interpreter_variable_data_t * interp_variable, int index)`

Definition at line 549 of file `xml_compose.c`.

References `agent_xml_compose__create_row_nodes()`, `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, and `node`.

Referenced by `agent_xml_compose__data()`.

#### 5.46.1.13 `mxml_node_t* agent_xml_compose__task (agent_p agent, int index)`

Definition at line 342 of file `xml_compose.c`.

References `agent_task_s::agent_return_data`, `agent_task_s::agent_variable_list`, `agent_xml_compose__data()`, `buf`, `agent_task_s::code_id`, `agent_s::datastate`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlElementSetAttr()`, `mxmlNewElement()`, `node`, `agent_task_s::persistent`, `agent_datastate_s::persistent`, `agent_task_s::server_name`, `agent_datastate_s::tasks`, and `agent_task_s::var_name`.

Referenced by `agent_xml_compose__tasks()`.

#### 5.46.1.14 `mxml_node_t* agent_xml_compose__tasks (agent_p agent)`

Definition at line 285 of file `xml_compose.c`.

References `agent_xml_compose__agent_code()`, `agent_xml_compose__task()`, `buf`, `agent_s::datastate`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlElementSetAttr()`, `mxmlNewElement()`, `node`, `agent_datastate_s::number_of_tasks`, and `agent_datastate_s::task_progress`.

Referenced by `agent_xml_compose__agent_data()`.



**5.46.1.15 mxml\_node\_t\* agent\_xml\_compose\_\_wg\_code (agent\_p *agent*)**

Definition at line 269 of file xml\_compose.c.

References `mxmlNewElement()`, `mxmlNewText()`, `node`, and `agent_s::wg_code`.

Referenced by `agent_xml_compose__agent_data()`.

## 5.47 /home/dko/Projects/mobilec/src/include/xml\_helper.h File Reference

```
#include <mxml.h>
```

### Functions

- [mxml\\_node\\_t \\* xml\\_find\\_sibling](#) (const [mxml\\_node\\_t](#) \**node*, const char \**sibling\_name*)
- char \* [xml\\_get\\_cdata](#) (const [mxml\\_node\\_t](#) \**node*)
- [mxml\\_node\\_t \\* xml\\_get\\_child](#) (const [mxml\\_node\\_t](#) \**node*, const char \**child\_name*, [int](#) *descend*)
- [mxml\\_node\\_t \\* xml\\_get\\_deep\\_child](#) (const [mxml\\_node\\_t](#) \**parent*, const char \*\**child\_path*)
- [mxml\\_node\\_t \\* xml\\_get\\_next\\_element](#) (const [mxml\\_node\\_t](#) \**node*)
- char \* [xml\\_get\\_text](#) (const [mxml\\_node\\_t](#) \**node*)
- char \* [xml\\_get\\_element\\_name](#) (const [mxml\\_node\\_t](#) \**node*)
- [mxml\\_node\\_t \\* xml\\_new\\_cdata](#) ([mxml\\_node\\_t](#) \**parent*, const char \**text*)
- const char \* [whitespace\\_cb](#) ([mxml\\_node\\_t](#) \**node*, [int](#) *where*)

### 5.47.1 Function Documentation

#### 5.47.1.1 const char \* whitespace\_cb (mxml\_node\_t \* *node*, int *where*)

Definition at line 572 of file testmxml.c.

References [mxml\\_node\\_s::child](#), [mxml\\_value\\_u::element](#), [MXML\\_WS\\_AFTER\\_CLOSE](#), [MXML\\_WS\\_AFTER\\_OPEN](#), [MXML\\_WS\\_BEFORE\\_CLOSE](#), [MXML\\_WS\\_BEFORE\\_OPEN](#), [mxml\\_value\\_s::name](#), [mxml\\_node\\_s::parent](#), and [mxml\\_node\\_s::value](#).

Referenced by [main\(\)](#).

#### 5.47.1.2 mxml\_node\_t\* xml\_find\_sibling (const mxml\_node\_t \* *node*, const char \* *sibling\_name*)

Definition at line 54 of file xml\_helper.c.

References [MXML\\_NO\\_DESCEND](#), [mxmlFindElement\(\)](#), [mxml\\_node\\_s::parent](#), and [xml\\_get\\_element\\_name\(\)](#).

#### 5.47.1.3 char\* xml\_get\_cdata (const mxml\_node\_t \* *node*)

Definition at line 76 of file xml\_helper.c.

References [buf](#), [CHECK\\_NULL](#), [MXML\\_ELEMENT](#), [mxml\\_node\\_s::type](#), and [xml\\_get\\_element\\_name\(\)](#).

Referenced by [xml\\_get\\_text\(\)](#).

#### 5.47.1.4 mxml\_node\_t\* xml\_get\_child (const mxml\_node\_t \* *node*, const char \* *child\_name*, int *descend*)

Definition at line 109 of file xml\_helper.c.

References [mxmlFindElement\(\)](#).

Referenced by `agent_return_xml_parse()`, `agent_xml_parse__agent_data()`, `agent_xml_parse__data()`, `agent_xml_parse__mobile_agent()`, `message_xml_parse()`, `message_xml_parse__message()`, and `xml_get_deep_child()`.

#### 5.47.1.5 `mxml_node_t* xml_get_deep_child (const mxml_node_t * parent, const char ** child_path)`

Definition at line 128 of file `xml_helper.c`.

References `MXML_NO_DESCEND`, `node`, and `xml_get_child()`.

#### 5.47.1.6 `char* xml_get_element_name (const mxml_node_t * node)`

Definition at line 222 of file `xml_helper.c`.

References `mxml_value_u::element`, `MXML_ELEMENT`, `mxml_value_s::name`, `mxml_node_s::type`, and `mxml_node_s::value`.

Referenced by `agent_xml_parse__data()`, `agent_xml_parse__mobile_agent()`, `agent_xml_parse__row()`, `message_xml_parse()`, `xml_find_sibling()`, and `xml_get_cdata()`.

#### 5.47.1.7 `mxml_node_t* xml_get_next_element (const mxml_node_t * node)`

Definition at line 142 of file `xml_helper.c`.

References `MXML_ELEMENT`, `mxml_node_s::next`, and `mxml_node_s::type`.

#### 5.47.1.8 `char* xml_get_text (const mxml_node_t * node)`

Definition at line 160 of file `xml_helper.c`.

References `CHECK_NULL`, `mxml_node_s::child`, `mxml_value_u::element`, `MXML_ELEMENT`, `MXML_TEXT`, `mxml_value_s::name`, `mxml_node_s::next`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::type`, `mxml_node_s::value`, and `xml_get_cdata()`.

Referenced by `agent_xml_parse__agent_code()`, `agent_xml_parse__home()`, `agent_xml_parse__name()`, `agent_xml_parse__owner()`, `agent_xml_parse__sender()`, and `agent_xml_parse__wg_code()`.

#### 5.47.1.9 `mxml_node_t* xml_new_cdata (mxml_node_t * parent, const char * text)`

Definition at line 235 of file `xml_helper.c`.

References `CHECK_NULL`, `mxmlNewElement()`, and `node`.

Referenced by `agent_xml_compose__agent_code()`.

## 5.48 /home/dko/Projects/mobilec/src/include/xml\_parser.h File Reference

```
#include <mxml.h>
#include "macros.h"
#include "agent.h"
```

### Functions

- [STRUCT](#) ([xml\\_parser](#), [mxml\\_node\\_t](#) \*root; [mxml\\_node\\_t](#) \*node;)
- [error\\_code\\_t agent\\_xml\\_parse](#) ([agent\\_p](#) agent)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_mobile\\_agent](#) ([agent\\_p](#) agent, [xml\\_parser\\_p](#) xml\_parser)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_agent\\_data](#) ([agent\\_p](#) agent, [xml\\_parser\\_p](#) xml\_parser)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_name](#) ([agent\\_p](#) agent, [xml\\_parser\\_p](#) xml\_parser)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_owner](#) ([agent\\_p](#) agent, [xml\\_parser\\_p](#) xml\_parser)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_home](#) ([agent\\_p](#) agent, [xml\\_parser\\_p](#) xml\_parser)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_sender](#) ([agent\\_p](#) agent, [xml\\_parser\\_p](#) xml\_parser)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_wg\\_code](#) ([agent\\_p](#) agent, [xml\\_parser\\_p](#) xml\_parser)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_tasks](#) ([agent\\_p](#) agent, [xml\\_parser\\_p](#) xml\_parser)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_task](#) ([agent\\_p](#) agent, [xml\\_parser\\_p](#) xml\_parser, [int](#) index)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_data](#) ([agent\\_p](#) agent, [xml\\_parser\\_p](#) xml\_parser, [int](#) index)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_row](#) ([interpreter\\_variable\\_data\\_t](#) \*interp\_variable, [xml\\_parser\\_p](#) xml\_parser, [int](#) index)
- [void agent\\_xml\\_parse\\_\\_fill\\_row\\_data](#) ([void](#) \*data, [ChType\\_t](#) type, [int](#) \*extent, [const mxml\\_node\\_t](#) \*node, [int](#) \*index)
- [error\\_code\\_t agent\\_xml\\_parse\\_\\_agent\\_code](#) ([agent\\_p](#) agent, [int](#) index, [xml\\_parser\\_p](#) xml\_parser)
- [error\\_code\\_t message\\_xml\\_parse](#) ([message\\_p](#) message)
- [error\\_code\\_t message\\_xml\\_parse\\_\\_message](#) ([message\\_p](#) message, [xml\\_parser\\_p](#) xml\_parser)

### 5.48.1 Function Documentation

#### 5.48.1.1 [error\\_code\\_t agent\\_xml\\_parse](#) ([agent\\_p](#) agent)

Definition at line 52 of file [xml\\_parser.c](#).

References [agent\\_xml\\_parse\\_\\_mobile\\_agent\(\)](#), [agent\\_s::datastate](#), [MC\\_SUCCESS](#), and [agent\\_datastate\\_s::xml\\_agent\\_root](#).

Referenced by [agent\\_initialize\(\)](#).

#### 5.48.1.2 [error\\_code\\_t agent\\_xml\\_parse\\_\\_agent\\_code](#) ([agent\\_p](#) agent, [int](#) index, [xml\\_parser\\_p](#) xml\_parser)

Definition at line 872 of file [xml\\_parser.c](#).

References [agent\\_datastate\\_s::agent\\_code](#), [agent\\_datastate\\_s::agent\\_code\\_ids](#), [agent\\_datastate\\_s::agent\\_codes](#), [agent\\_task\\_s::code\\_id](#), [agent\\_s::datastate](#), [MC\\_SUCCESS](#), [mxmlElementGetAttr\(\)](#), [agent\\_datastate\\_s::number\\_of\\_tasks](#), [agent\\_datastate\\_s::task\\_progress](#), [agent\\_datastate\\_s::tasks](#), and [xml\\_get\\_text\(\)](#).

Referenced by [agent\\_xml\\_parse\\_\\_tasks\(\)](#).

**5.48.1.3 error\_code\_t agent\_xml\_parse\_\_agent\_data (agent\_p agent, xml\_parser\_p xml\_parser)**

Definition at line 93 of file xml\_parser.c.

References agent\_xml\_parse\_\_home(), agent\_xml\_parse\_\_name(), agent\_xml\_parse\_\_owner(), agent\_xml\_parse\_\_sender(), agent\_xml\_parse\_\_tasks(), agent\_xml\_parse\_\_wg\_code(), MC\_ERR\_PARSE, MC\_SUCCESS, and xml\_get\_child().

Referenced by agent\_xml\_parse\_\_mobile\_agent().

**5.48.1.4 error\_code\_t agent\_xml\_parse\_\_data (agent\_p agent, xml\_parser\_p xml\_parser, int index)**

Definition at line 538 of file xml\_parser.c.

References agent\_task\_s::agent\_return\_data, agent\_task\_s::agent\_variable\_list, agent\_xml\_parse\_\_row(), interpreter\_variable\_data\_s::array\_dim, CH\_DATATYPE\_SIZE, CH\_DATATYPE\_STR\_TO\_VAL, CH\_STRING\_DATATYPE, interpreter\_variable\_data\_s::data, interpreter\_variable\_data\_s::data\_type, agent\_s::datastate, interpreter\_variable\_data\_New(), MC\_ERR\_PARSE, MC\_SUCCESS, mxmlElementGetAttr(), interpreter\_variable\_data\_s::name, mxml\_node\_s::parent, agent\_task\_s::persistent, agent\_datastate\_s::tasks, xml\_get\_child(), and xml\_get\_element\_name().

Referenced by agent\_xml\_parse\_\_task().

**5.48.1.5 void agent\_xml\_parse\_\_fill\_row\_data (void \* data, ChType\_t type, int \* extent, const mxml\_node\_t \* node, int \* index)**

Definition at line 751 of file xml\_parser.c.

References agent\_xml\_parse\_\_fill\_row\_data(), buf, CH\_DATATYPE\_SIZE, mxml\_node\_s::child, MXML\_DESCEND\_FIRST, MXML\_ELEMENT, MXML\_TEXT, mxmlFindElement(), mxml\_text\_s::string, strtok\_r, mxml\_value\_u::text, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by agent\_xml\_parse\_\_fill\_row\_data(), and agent\_xml\_parse\_\_row().

**5.48.1.6 error\_code\_t agent\_xml\_parse\_\_home (agent\_p agent, xml\_parser\_p xml\_parser)**

Definition at line 219 of file xml\_parser.c.

References CHECK\_NULL, agent\_s::home, MC\_SUCCESS, and xml\_get\_text().

Referenced by agent\_return\_xml\_parse(), and agent\_xml\_parse\_\_agent\_data().

**5.48.1.7 error\_code\_t agent\_xml\_parse\_\_mobile\_agent (agent\_p agent, xml\_parser\_p xml\_parser)**

Definition at line 65 of file xml\_parser.c.

References agent\_xml\_parse\_\_agent\_data(), MC\_ERR\_PARSE, xml\_get\_child(), and xml\_get\_element\_name().

Referenced by agent\_xml\_parse().

**5.48.1.8 error\_code\_t agent\_xml\_parse\_\_name (agent\_p agent, xml\_parser\_p xml\_parser)**

Definition at line 166 of file xml\_parser.c.

References CHECK\_NULL, MC\_ERR\_PARSE, MC\_SUCCESS, agent\_s::name, and xml\_get\_text().

Referenced by agent\_return\_xml\_parse(), and agent\_xml\_parse\_\_agent\_data().

#### 5.48.1.9 error\_code\_t agent\_xml\_parse\_\_owner (agent\_p agent, xml\_parser\_p xml\_parser)

Definition at line 192 of file xml\_parser.c.

References CHECK\_NULL, MC\_SUCCESS, agent\_s::owner, and xml\_get\_text().

Referenced by agent\_return\_xml\_parse(), and agent\_xml\_parse\_\_agent\_data().

#### 5.48.1.10 error\_code\_t agent\_xml\_parse\_\_row (interpreter\_variable\_data\_t \* interp\_variable, xml\_parser\_p xml\_parser, int index)

Definition at line 684 of file xml\_parser.c.

References agent\_xml\_parse\_\_fill\_row\_data(), interpreter\_variable\_data\_s::array\_dim, interpreter\_variable\_data\_s::array\_extent, CH\_DATATYPE\_SIZE, interpreter\_variable\_data\_s::data, interpreter\_variable\_data\_s::data\_type, MC\_SUCCESS, and xml\_get\_element\_name().

Referenced by agent\_xml\_parse\_\_data().

#### 5.48.1.11 error\_code\_t agent\_xml\_parse\_\_sender (agent\_p agent, xml\_parser\_p xml\_parser)

Definition at line 245 of file xml\_parser.c.

References CHECK\_NULL, MC\_SUCCESS, agent\_s::sender, and xml\_get\_text().

Referenced by agent\_xml\_parse\_\_agent\_data().

#### 5.48.1.12 error\_code\_t agent\_xml\_parse\_\_task (agent\_p agent, xml\_parser\_p xml\_parser, int index)

Definition at line 442 of file xml\_parser.c.

References agent\_xml\_parse\_\_data(), CHECK\_NULL, agent\_task\_s::code\_id, agent\_s::datastate, MC\_ERR\_PARSE, MC\_SUCCESS, MXML\_DESCEND\_FIRST, MXML\_NO\_DESCEND, mxmlElementGetAttr(), mxmlFindElement(), agent\_task\_s::persistent, agent\_task\_s::server\_name, strdup, agent\_datastate\_s::tasks, and agent\_task\_s::var\_name.

Referenced by agent\_xml\_parse\_\_tasks().

#### 5.48.1.13 error\_code\_t agent\_xml\_parse\_\_tasks (agent\_p agent, xml\_parser\_p xml\_parser)

Definition at line 304 of file xml\_parser.c.

References agent\_datastate\_s::agent\_code, agent\_datastate\_s::agent\_code\_ids, agent\_datastate\_s::agent\_codes, agent\_task\_New(), agent\_xml\_parse\_\_agent\_code(), agent\_xml\_parse\_\_task(), buf, agent\_s::datastate, MC\_ERR\_PARSE, MXML\_DESCEND, MXML\_DESCEND\_FIRST, MXML\_NO\_DESCEND, mxmlElementGetAttr(), mxmlFindElement(), agent\_datastate\_s::number\_of\_tasks, agent\_datastate\_s::task\_progress, and agent\_datastate\_s::tasks.

Referenced by agent\_return\_xml\_parse(), and agent\_xml\_parse\_\_agent\_data().

**5.48.1.14 error\_code\_t agent\_xml\_parse\_\_wg\_code (agent\_p agent, xml\_parser\_p xml\_parser)**

Definition at line 273 of file xml\_parser.c.

References MC\_SUCCESS, agent\_s::wg\_code, and xml\_get\_text().

Referenced by agent\_xml\_parse\_\_agent\_data().

**5.48.1.15 error\_code\_t message\_xml\_parse (message\_p message)**

Definition at line 948 of file xml\_parser.c.

References MC\_ERR\_PARSE, message\_xml\_parse\_\_message(), MXML\_DESCEND, MXML\_NO\_DESCEND, mxmlFindElement(), xml\_get\_child(), xml\_get\_element\_name(), and message\_s::xml\_root.

Referenced by acc\_connection\_Thread(), message\_InitializeFromConnection(), and message\_InitializeFromString().

**5.48.1.16 error\_code\_t message\_xml\_parse\_\_message (message\_p message, xml\_parser\_p xml\_parser)**

Definition at line 1003 of file xml\_parser.c.

References message\_s::addr, buf, CHECK\_NULL, ENCRYPTED\_DATA, ENCRYPTION\_INITIALIZE, FIPA\_ACL, message\_s::from\_address, MC\_ERR\_PARSE, MC\_SUCCESS, message\_s::message\_type, MOBILE\_AGENT, mxmlElementGetAttr(), port, REQUEST\_ENCRYPTION\_INITIALIZE, RETURN\_MSG, strtok\_r, xml\_get\_child(), and message\_s::xml\_payload.

Referenced by message\_xml\_parse().

**5.48.1.17 STRUCT (xml\_parser, mxml\_node\_t \*root;mxml\_node\_t \*node;)**

## 5.49 /home/dko/Projects/mobilec/src/libmc.c File Reference

```
#include "config.h"
#include <unistd.h>
#include <pthread.h>
#include <sys/socket.h>
#include <netdb.h>
#include <embedch.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <sys/time.h>
#include "include/libmc.h"
#include "include/macros.h"
#include "include/mc_platform.h"
#include "include/message.h"
#include "include/data_structures.h"
#include "include/fipa_acl_envelope.h"
#include "include/fipa_acl.h"
#include "include/agent.h"
#include "include/agent_task.h"
#include "include/agent_lib.h"
```

### Functions

- EXPORTMC [int MC\\_AclDestroy](#) (struct [fipa\\_acl\\_message\\_s](#) \*message)  
*Destroy a FIPA ACL message.*
- EXPORTMC [fipa\\_acl\\_message\\_t](#) \* [MC\\_AclNew](#) (void)  
*Allocate a new ACL Message.*
- EXPORTMC [int MC\\_AclPost](#) ([MCAgent\\_t](#) agent, struct [fipa\\_acl\\_message\\_s](#) \*message)  
*Post ACL message to agent.*
- EXPORTMC [fipa\\_acl\\_message\\_t](#) \* [MC\\_AclReply](#) ([fipa\\_acl\\_message\\_t](#) \*acl\_message)  
*Reply to an ACL message.*
- EXPORTMC [fipa\\_acl\\_message\\_t](#) \* [MC\\_AclRetrieve](#) ([MCAgent\\_t](#) agent)  
*Retrieve an ACL message.*
- EXPORTMC [int MC\\_AclSend](#) ([MCAgency\\_t](#) attr, [fipa\\_acl\\_message\\_t](#) \*acl)  
*Send a composed ACL Message.*
- EXPORTMC [fipa\\_acl\\_message\\_t](#) \* [MC\\_AclWaitRetrieve](#) ([MCAgent\\_t](#) agent)



*Wait for and retrieve an ACL message.*

- EXPORTMC enum [fipa\\_protocol\\_e](#) [MC\\_AclGetProtocol](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl)
- EXPORTMC char \* [MC\\_AclGetConversationID](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl)
- EXPORTMC enum [fipa\\_performative\\_e](#) [MC\\_AclGetPerformative](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl)
- EXPORTMC int [MC\\_AclGetSender](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl, char \*\*name, char \*\*address)
- EXPORTMC const char \* [MC\\_AclGetContent](#) (struct [fipa\\_acl\\_message\\_s](#) \*acl)
- EXPORTMC int [MC\\_AclSetProtocol](#) ([fipa\\_acl\\_message\\_t](#) \*acl, enum [fipa\\_protocol\\_e](#) protocol)
- EXPORTMC int [MC\\_AclSetConversationID](#) ([fipa\\_acl\\_message\\_t](#) \*acl, char \*id)
- EXPORTMC int [MC\\_AclSetPerformative](#) ([fipa\\_acl\\_message\\_t](#) \*acl, enum [fipa\\_performative\\_e](#) performative)
- EXPORTMC int [MC\\_AclSetSender](#) ([fipa\\_acl\\_message\\_t](#) \*acl, const char \*name, const char \*address)
- EXPORTMC int [MC\\_AclAddReceiver](#) ([fipa\\_acl\\_message\\_t](#) \*acl, const char \*name, const char \*address)
- EXPORTMC int [MC\\_AclAddReplyTo](#) ([fipa\\_acl\\_message\\_t](#) \*acl, const char \*name, const char \*address)
- EXPORTMC int [MC\\_AclSetContent](#) ([fipa\\_acl\\_message\\_t](#) \*acl, const char \*content)
- EXPORTMC int [MC\\_AddAgent](#) ([MCAgency\\_t](#) attr, [MCAgent\\_t](#) agent)

*Add an agent to the agency 'attr'.*

- EXPORTMC int [MC\\_AddStationaryAgent](#) ([MCAgency\\_t](#) agency, void \*(\*agent\_thread)([stationary\\_agent\\_info\\_t](#) \*), const char \*name, void \*agent\_args)

*Add a stationary agent to the agency.*

- EXPORTMC int [MC\\_AddAgentInitCallback](#) ([MCAgency\\_t](#) agency, [MC\\_AgentInitCallbackFunc\\_t](#) function, void \*user\_data)

*Add an agent initialization callback function.*

- EXPORTMC [MCAgency\\_t](#) [MC\\_AgentInfo\\_GetAgency](#) ([stationary\\_agent\\_info\\_t](#) \*stationary\_agent\_info)

*Get the host Mobile-C agency handle from a "stationary\_agent\_info\_t" data structure.*

- EXPORTMC [MCAgent\\_t](#) [MC\\_AgentInfo\\_GetAgent](#) ([stationary\\_agent\\_info\\_t](#) \*stationary\_agent\_info)

*Retrieve an agent handle from the stationary agent info from within a stationary agent.*

- EXPORTMC void \* [MC\\_AgentInfo\\_GetAgentArgs](#) ([stationary\\_agent\\_info\\_t](#) \*stationary\_agent\_info)

*Retrieve the agent argument supplied as the "agent\_args" parameter in the [MC\\_AddStationaryAgent\(\)](#) function.*

- EXPORTMC int [MC\\_AgentAddTask](#) ([MCAgent\\_t](#) agent, const char \*code, const char \*return\_var\_name, const char \*server, int persistent)

*Add a new task to an already existing agent.*

- EXPORTMC int [MC\\_AgentAddTaskFromFile](#) ([MCAgent\\_t](#) agent, const char \*filename, const char \*return\_var\_name, const char \*server, int persistent)

*Add a new task to an already existing agent.*

- EXPORTMC int [MC\\_AgentReturnArrayExtent](#) ([MCAgent\\_t](#) agent, int task\_num, int index)

*Get the extent of a dimension of an agent's return data, if it is an array.*

- EXPORTMC [int MC\\_AgentReturnArrayDim](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Get the dimension of an agent's return data, if it is an array.*
- EXPORTMC [int MC\\_AgentReturnArrayNum](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Get the total number of elements of the agent return data.*
- EXPORTMC [const void \\*](#) [MC\\_AgentReturnDataGetSymbolAddr](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Get a pointer to the agent's return data.*
- EXPORTMC [size\\_t MC\\_AgentReturnDataSize](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Get the size of an agent's return data type.*
- EXPORTMC [int MC\\_AgentReturnDataType](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Get the type of an agent's return data.*
- EXPORTMC [int MC\\_AgentReturnIsArray](#) ([MCAgent\\_t](#) agent, [int](#) task\_num)  
*Check to see whether or not an agent's return data is an array.*
- EXPORTMC [const void \\*](#) [MC\\_AgentVariableRetrieve](#) ([MCAgent\\_t](#) agent, [const char \\*](#)var\_name, [int](#) task\_num)  
*Retrieve a pointer to a previously saved variable.*
- EXPORTMC [int MC\\_AgentVariableRetrieveInfo](#) ([MCAgent\\_t](#) agent, [const char \\*](#)var\_name, [int](#) task\_num, [const void \\*\\*](#)data, [int \\*](#)dim, [const int \\*\\*](#)extent)  
*Retrieve a info about a previously saved variable.*
- EXPORTMC [int MC\\_AgentVariableSave](#) ([MCAgent\\_t](#) agent, [const char \\*](#)var\_name)  
*Mark an agent variable for saving.*
- EXPORTMC [int MC\\_Barrier](#) ([MCAgency\\_t](#) attr, [int](#) id)
- EXPORTMC [int MC\\_BarrierInit](#) ([MCAgency\\_t](#) attr, [int](#) id, [int](#) num\_procs)  
*Initialize a MobileC Barrier.*
- EXPORTMC [int MC\\_BarrierDelete](#) ([MCAgency\\_t](#) attr, [int](#) id)  
*Find and delete an initialized MobileC Barrier.*
- EXPORTMC [int MC\\_CallAgentFunc](#) ([MCAgent\\_t](#) agent, [const char \\*](#)funcName, [void \\*](#)returnVal, [int](#) numArgs,...)  
*Use custom ChOptions\_t type for internal Ch interpreter.*
- EXPORTMC [int MC\\_CallAgentFuncArg](#) ([MCAgent\\_t](#) agent, [const char \\*](#)funcName, [void \\*](#)returnVal, [void \\*](#)arg)  
*Calls a function defined in an agent.*
- EXPORTMC [int MC\\_CallAgentFuncV](#) ([MCAgent\\_t](#) agent, [const char \\*](#)funcName, [void \\*](#)returnVal, [va\\_list](#) ap)  
*Calls a function defined in an agent.*

- EXPORTMC [int MC\\_CallAgentFuncVar](#) ([MCAgent\\_t](#) agent, const char \*funcName, void \*returnVal, ChVaList\_t varg)
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgent](#) (const char \*name, const char \*home, const char \*owner, const char \*code, const char \*return\_var\_name, const char \*server, [int](#) persistent)  
*Compose a new agent dynamically without using a prewritten XML file.*
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgentS](#) (const char \*name, const char \*home, const char \*owner, const char \*code, const char \*return\_var\_name, const char \*server, [int](#) persistent, const char \*workgroup\_code)  
*Compose a new agent dynamically without using a prewritten XML file.*
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgentWithWorkgroup](#) (const char \*name, const char \*home, const char \*owner, const char \*code, const char \*return\_var\_name, const char \*server, [int](#) persistent, const char \*workgroup\_code)  
*Compose a new agent dynamically without using a prewritten XML file.*
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgentFromFile](#) (const char \*name, const char \*home, const char \*owner, const char \*filename, const char \*return\_var\_name, const char \*server, [int](#) persistent)  
*Compose a new agent dynamically from a source code file.*
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgentFromFileS](#) (const char \*name, const char \*home, const char \*owner, const char \*filename, const char \*return\_var\_name, const char \*server, [int](#) persistent, const char \*workgroup\_code)  
*Compose a new agent dynamically from a source code file.*
- EXPORTMC [MCAgent\\_t MC\\_ComposeAgentFromFileWithWorkgroup](#) (const char \*name, const char \*home, const char \*owner, const char \*filename, const char \*return\_var\_name, const char \*server, [int](#) persistent, const char \*workgroup\_code)  
*Compose a new agent dynamically from a source code file.*
- EXPORTMC [int MC\\_CondBroadcast](#) ([MCAgency\\_t](#) attr, [int](#) id)  
*Wakes up all agents/threads waiting on a condition variable.*
- EXPORTMC [int MC\\_CondSignal](#) ([MCAgency\\_t](#) attr, [int](#) id)  
*Wakes up at least one thread waiting on a condition variable.*
- EXPORTMC [int MC\\_CondWait](#) ([MCAgency\\_t](#) attr, [int](#) id)  
*Wait on a MobileC synchronization variable.*
- EXPORTMC [int MC\\_CondReset](#) ([MCAgency\\_t](#) attr, [int](#) id)  
*Reset a previously signalled MobileC condition variable.*
- EXPORTMC [int MC\\_CopyAgent](#) ([MCAgent\\_t](#) \*agent\_out, const [MCAgent\\_t](#) agent\_in)  
*Performs a deep-copy of an agent structure.*
- EXPORTMC [int MC\\_DeleteAgent](#) ([MCAgent\\_t](#) agent)  
*Stop and remove an agent.*
- EXPORTMC [int MC\\_DeleteAgentWG](#) ([MCAgent\\_t](#) calling\_agent, [MCAgent\\_t](#) agent)  
*Stop and remove an agent in the same workgroup.*

- EXPORTMC [int MC\\_DestroyServiceSearchResult](#) (char \*\*agentName, char \*\*serviceName, [int](#) \*agentID, [int](#) numResult)  
*Free memory allocated by a Service Search operation.*
- EXPORTMC [int MC\\_DeregisterService](#) ([MCAgency\\_t](#) agency, [int](#) agentID, const char \*serviceName)
- EXPORTMC [int MC\\_End](#) ([MCAgency\\_t](#) agency)  
*End an agency.*
- EXPORTMC [MCAgent\\_t MC\\_FindAgentByName](#) ([MCAgency\\_t](#) attr, const char \*name)  
*Find an agent by its name.*
- EXPORTMC [MCAgent\\_t MC\\_FindAgentByID](#) ([MCAgency\\_t](#) attr, [int](#) ID)  
*Find an agent by its id.*
- EXPORTMC [time\\_t MC\\_GetAgentArrivalTime](#) ([MCAgent\\_t](#) agent)
- EXPORTMC [int MC\\_GetAgentStatus](#) ([MCAgent\\_t](#) agent)  
*Get an agent's current status.*
- EXPORTMC char \* [MC\\_GetAgentXMLString](#) ([MCAgent\\_t](#) agent)  
*Get an agent's xml string.*
- EXPORTMC void \* [MC\\_GetAgentExecEngine](#) ([MCAgent\\_t](#) agent)  
*Retrieve an agent's Ch interpreter.*
- EXPORTMC [int MC\\_GetAgentID](#) ([MCAgent\\_t](#) agent)  
*Retrieve an agent's id.*
- EXPORTMC char \* [MC\\_GetAgentName](#) ([MCAgent\\_t](#) agent)
- EXPORTMC [int MC\\_GetAgentReturnData](#) ([MCAgent\\_t](#) agent, [int](#) task\_num, void \*\*data, [int](#) \*dim, [int](#) \*\*extent)  
*Get an agent's return data.*
- EXPORTMC [int MC\\_GetAgentNumTasks](#) ([MCAgent\\_t](#) agent)  
*Retrive the number of tasks an agent has.*
- EXPORTMC enum [MC\\_AgentType\\_e MC\\_GetAgentType](#) ([MCAgent\\_t](#) agent)  
*Get an agent's type.*
- EXPORTMC [int MC\\_GetAllAgents](#) ([MCAgency\\_t](#) attr, [MCAgent\\_t](#) \*\*agents, [int](#) \*num\_agents)
- EXPORTMC [int MC\\_HaltAgency](#) ([MCAgency\\_t](#) attr)  
*Halt an agency: Do not process new entries in queues.*
- EXPORTMC [MCAgency\\_t MC\\_Initialize](#) ([int](#) port, [MCAgencyOptions\\_t](#) \*options)  
*Initialize and start a MobileC agency.*
- EXPORTMC [int MC\\_InitializeAgencyOptions](#) (struct [MCAgencyOptions\\_s](#) \*options)  
*Initialize MobileC options.*

- EXPORTMC [int MC\\_LoadAgentFromFile](#) (MCAgency\_t attr, const char \*filename)  
*Load an agent from a file into an agency.*
- EXPORTMC [int MC\\_MigrateAgent](#) (MCAgent\_t agent, const char \*hostname, [int port](#))  
*Migrates a running agent to another host.*
- EXPORTMC [int MC\\_MutexLock](#) (MCAgency\_t attr, [int id](#))  
*Locks a MobileC synchronization variable as a mutex.*
- EXPORTMC [int MC\\_MutexUnlock](#) (MCAgency\_t attr, [int id](#))
- EXPORTMC [int MC\\_PrintAgentCode](#) (MCAgent\_t agent)  
*Prints an agents code to stdout.*
- EXPORTMC [int MC\\_RegisterService](#) (MCAgency\_t agency, MCAgent\_t agent, [int agentID](#), const char \*agentName, char \*\*serviceNames, [int numServices](#))  
*Register a new service with the Directory Facilitator.*
- EXPORTMC [int MC\\_ResumeAgency](#) (MCAgency\_t attr)  
*Resumes a halted agency.*
- EXPORTMC [MCAgent\\_t MC\\_RetrieveAgent](#) (MCAgency\_t attr)  
*Retrieves the oldest agent from an agency.*
- EXPORTMC char \* [MC\\_RetrieveAgentCode](#) (MCAgent\_t agent)  
*Retrieves an agent's Ch code.*
- EXPORTMC [int MC\\_ResetSignal](#) (MCAgency\_t attr)  
*Reset a MobileC signal.*
- EXPORTMC [int MC\\_SearchForService](#) (MCAgency\_t attr, const char \*searchString, char \*\*\*agentNames, char \*\*\*serviceNames, [int \\*\\*agentIDs](#), [int \\*numResults](#))  
*Search the directory facilitator for a service.*
- EXPORTMC [int MC\\_SemaphorePost](#) (MCAgency\_t attr, [int id](#))  
*Post to a MobileC synchronization variable semaphore.*
- EXPORTMC [int MC\\_SemaphoreWait](#) (MCAgency\_t attr, [int id](#))  
*Decreases a MobileC synchronization variable semaphore count by one.*
- EXPORTMC [int MC\\_SendCh](#) (MCAgency\_t attr, const char \*filename, const char \*remotehost, [int port](#))
- EXPORTMC [int MC\\_SendAgent](#) (MCAgency\_t attr, const char \*string)  
*Sends an agent migration message.*
- EXPORTMC [int MC\\_SendAgentFile](#) (MCAgency\_t attr, const char \*filename)  
*Sends an agent migration message.*
- EXPORTMC [int MC\\_SendAgentMigrationMessage](#) (MCAgency\_t attr, const char \*string, const char \*hostname, [int port](#))  
*Sends an agent migration message.*

- EXPORTMC [int MC\\_SendAgentMigrationMessageFile](#) (MCAgency\_t attr, const char \*filename, const char \*hostname, [int port](#))  
*Sends an agent migration message.*
- EXPORTMC [int MC\\_SendSteerCommand](#) (MCAgency\_t attr, enum [MC\\_SteerCommand\\_e](#) cmd)
- EXPORTMC [int MC\\_SetAgentStatus](#) (MCAgent\_t agent, enum [MC\\_AgentStatus\\_e](#) status)  
*Set an agent's status.*
- EXPORTMC [int MC\\_SetDefaultAgentStatus](#) (MCAgency\_t agency, enum [MC\\_AgentStatus\\_e](#) status)  
*Sets default incoming agent status.*
- EXPORTMC [int MC\\_SetThreadOn](#) (MCAgencyOptions\_t \*options, enum [MC\\_ThreadIndex\\_e](#) index)  
*Sets a MobileC thread to "on" status.*
- EXPORTMC [int MC\\_SetThreadsAllOn](#) (MCAgencyOptions\_t \*options)  
*Set all Mobile-C threads on.*
- EXPORTMC [int MC\\_SetThreadOff](#) (MCAgencyOptions\_t \*options, enum [MC\\_ThreadIndex\\_e](#) index)  
*Sets a MobileC thread to "off" status.*
- EXPORTMC [int MC\\_SetThreadsAllOff](#) (MCAgencyOptions\_t \*options)  
*Set all MobileC threads to 'off' status.*
- EXPORTMC [int MC\\_Steer](#) (MCAgency\_t attr, [int](#)(\*funcptr)(void \*data), void \*arg)  
*Set up a steerable algorithm.*
- EXPORTMC enum [MC\\_SteerCommand\\_e](#) [MC\\_SteerControl](#) (void)  
*The MobileC user-algorithm steering function.*
- EXPORTMC [int MC\\_SyncDelete](#) (MCAgency\_t attr, [int id](#))  
*Deletes a previously initialized synchronization variable.*
- EXPORTMC [int MC\\_SyncInit](#) (MCAgency\_t attr, [int id](#))  
*Initializes a new MobileC synchronization variable.*
- EXPORTMC [int MC\\_TerminateAgent](#) (MCAgent\_t agent)  
*Halt a running agent.*
- EXPORTMC [int MC\\_TerminateAgentWG](#) (MCAgent\_t calling\_agent, MCAgent\_t agent)
- EXPORTMC [int MC\\_MainLoop](#) (MCAgency\_t attr)  
*Wait indefinitely.*
- EXPORTMC [int MC\\_WaitAgent](#) (MCAgency\_t attr)  
*Wait indefinitely.*
- EXPORTMC [MCAgent\\_t MC\\_WaitRetrieveAgent](#) (MCAgency\_t attr)

*Wait and retrieve an agent.*

- EXPORTMC [int MC\\_WaitSignal](#) (MCAgency\_t attr, int signals)

*Wait for a MobileC signal.*

- EXPORTMC [int MC\\_AclDestroy\\_chdl](#) (void \*varg)
- EXPORTMC [void \\* MC\\_AclNew\\_chdl](#) (void \*varg)
- EXPORTMC [int MC\\_AclPost\\_chdl](#) (void \*varg)
- EXPORTCH [void \\* MC\\_AclReply\\_chdl](#) (void \*varg)
- EXPORTCH [void \\* MC\\_AclRetrieve\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSend\\_chdl](#) (void \*varg)
- EXPORTCH [void \\* MC\\_AclWaitRetrieve\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclGetProtocol\\_chdl](#) (void \*varg)
- EXPORTCH [char \\* MC\\_AclGetConversationID\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclGetPerformative\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclGetSender\\_chdl](#) (void \*varg)
- EXPORTCH [char \\* MC\\_AclGetContent\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSetProtocol\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSetConversationID\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSetPerformative\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSetSender\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclAddReceiver\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclAddReplyTo\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AclSetContent\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AddAgent\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AgentAddTask\\_chdl](#) (void \*varg)
- EXPORTCH [const void \\* MC\\_AgentVariableRetrieve\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_AgentVariableSave\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_CallAgentFunc\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_Barrier\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_BarrierDelete\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_BarrierInit\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_CondBroadcast\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_ComposeAgent\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_ComposeAgentWithWorkgroup\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_ComposeAgentFromFile\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_ComposeAgentFromFileWithWorkgroup\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_CondSignal\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_CondReset\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_CondWait\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_DeleteAgent\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_DeleteAgentWG\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_DestroyServiceSearchResult\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_DeregisterService\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_End\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_FindAgentByID\\_chdl](#) (void \*varg)
- EXPORTCH [MCAgent\\_t MC\\_FindAgentByName\\_chdl](#) (void \*varg)
- EXPORTCH [time\\_t MC\\_GetAgentArrivalTime\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_GetAgentID\\_chdl](#) (void \*varg)
- EXPORTCH [char \\* MC\\_GetAgentName\\_chdl](#) (void \*varg)

- EXPORTCH [int MC\\_GetAgentNumTasks\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_GetAgentStatus\\_chdl](#) (void \*varg)
- EXPORTCH [char \\* MC\\_GetAgentXMLString\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_GetTimeOfDay\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_HaltAgency\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_MigrateAgent\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_MutexLock\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_MutexUnlock\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_PrintAgentCode\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_RegisterService\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_ResumeAgency\\_chdl](#) (void \*varg)
- EXPORTCH [MC\\_Agent\\_t MC\\_RetrieveAgent\\_chdl](#) (void \*varg)
- EXPORTCH [char \\* MC\\_RetrieveAgentCode\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_SaveData\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_SearchForService\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_SemaphorePost\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_SemaphoreWait\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_SendAgentMigrationMessage\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_SendAgentMigrationMessageFile\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_SendSteerCommand\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_SetAgentStatus\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_SetDefaultAgentStatus\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_SyncDelete\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_SyncInit\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_TerminateAgent\\_chdl](#) (void \*varg)
- EXPORTCH [int MC\\_TerminateAgentWG\\_chdl](#) (void \*varg)

## Variables

- [mc\\_platform\\_p g\\_mc\\_platform](#)

## 5.49.1 Function Documentation

### 5.49.1.1 EXPORTMC int MC\_AclAddReceiver (fipa\_acl\_message\_t \* *acl*, const char \* *name*, const char \* *address*)

Definition at line 318 of file libmc.c.

References [fipa\\_agent\\_identifier\\_s::addresses](#), [fipa\\_agent\\_identifier\\_New\(\)](#), [fipa\\_agent\\_identifier\\_set\\_-New\(\)](#), [fipa\\_agent\\_identifier\\_set\\_s::fipa\\_agent\\_identifiers](#), [fipa\\_url\\_New\(\)](#), [fipa\\_url\\_sequence\\_New\(\)](#), [fipa\\_agent\\_identifier\\_s::name](#), [fipa\\_url\\_sequence\\_s::num](#), [fipa\\_agent\\_identifier\\_set\\_s::num](#), [fipa\\_acl\\_message\\_s::receiver](#), [fipa\\_acl\\_message\\_s::receiver\\_num](#), [fipa\\_url\\_s::str](#), [strdup](#), and [fipa\\_url\\_sequence\\_s::urls](#).

Referenced by [MC\\_AclAddReceiver\\_chdl\(\)](#).

### 5.49.1.2 EXPORTCH int MC\_AclAddReceiver\_chdl (void \* *varg*)

Definition at line 2873 of file libmc.c.

References [MC\\_AclAddReceiver\(\)](#).

Referenced by [agent\\_ChScriptInitVar\(\)](#).



**5.49.1.3 EXPORTMC int MC\_AclAddReplyTo (fipa\_acl\_message\_t \* *acl*, const char \* *name*, const char \* *address*)**

Definition at line 354 of file libmc.c.

References fipa\_agent\_identifier\_s::addresses, fipa\_agent\_identifier\_New(), fipa\_agent\_identifier\_set\_New(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_url\_New(), fipa\_url\_sequence\_New(), fipa\_agent\_identifier\_s::name, fipa\_url\_sequence\_s::num, fipa\_agent\_identifier\_set\_s::num, fipa\_acl\_message\_s::reply\_to, fipa\_url\_s::str, strdup, and fipa\_url\_sequence\_s::urls.

Referenced by MC\_AclAddReplyTo\_chdl().

**5.49.1.4 EXPORTCH int MC\_AclAddReplyTo\_chdl (void \* *varg*)**

Definition at line 2893 of file libmc.c.

References MC\_AclAddReplyTo().

Referenced by agent\_ChScriptInitVar().

**5.49.1.5 EXPORTMC int MC\_AclDestroy (struct fipa\_acl\_message\_s \* *message*)**

Destroy a FIPA ACL message.

**Parameters**

*message* The ACL message to destroy

**Returns**

0 on success, error code on failure.

Definition at line 81 of file libmc.c.

References fipa\_acl\_message\_Destroy().

Referenced by MC\_AclDestroy\_chdl().

**5.49.1.6 EXPORTMC int MC\_AclDestroy\_chdl (void \* *varg*)**

Definition at line 2605 of file libmc.c.

References MC\_AclDestroy().

Referenced by agent\_ChScriptInitVar().

**5.49.1.7 EXPORTMC const char\* MC\_AclGetContent (struct fipa\_acl\_message\_s \* *acl*)**

Definition at line 246 of file libmc.c.

References fipa\_string\_s::content, and fipa\_acl\_message\_s::content.

Referenced by MC\_AclGetContent\_chdl().

**5.49.1.8 EXPORTCH char\* MC\_AclGetContent\_chdl (void \* *varg*)**

Definition at line 2784 of file libmc.c.

References MC\_AclGetContent().

Referenced by agent\_ChScriptInitVar().

**5.49.1.9 EXPORTMC char\* MC\_AclGetConversationID (struct fipa\_acl\_message\_s \* *acl*)**

Definition at line 221 of file libmc.c.

References fipa\_string\_s::content, fipa\_expression\_s::content, fipa\_acl\_message\_s::conversation\_id, and fipa\_expression\_s::content\_u::string.

Referenced by MC\_AclGetConversationID\_chdl().

**5.49.1.10 EXPORTCH char\* MC\_AclGetConversationID\_chdl (void \* *varg*)**

Definition at line 2735 of file libmc.c.

References MC\_AclGetConversationID().

Referenced by agent\_ChScriptInitVar().

**5.49.1.11 EXPORTMC enum fipa\_performative\_e MC\_AclGetPerformative (struct fipa\_acl\_message\_s \* *acl*)**

Definition at line 227 of file libmc.c.

References fipa\_acl\_message\_s::performative.

Referenced by MC\_AclGetPerformative\_chdl().

**5.49.1.12 EXPORTCH int MC\_AclGetPerformative\_chdl (void \* *varg*)**

Definition at line 2750 of file libmc.c.

References MC\_AclGetPerformative().

Referenced by agent\_ChScriptInitVar().

**5.49.1.13 EXPORTMC enum fipa\_protocol\_e MC\_AclGetProtocol (struct fipa\_acl\_message\_s \* *acl*)**

Definition at line 215 of file libmc.c.

References fipa\_acl\_message\_s::performative.

Referenced by MC\_AclGetProtocol\_chdl().

**5.49.1.14 EXPORTCH int MC\_AclGetProtocol\_chdl (void \* *varg*)**

Definition at line 2720 of file libmc.c.

References MC\_AclGetProtocol().

Referenced by agent\_ChScriptInitVar().

#### 5.49.1.15 EXPORTMC int MC\_AclGetSender (struct fipa\_acl\_message\_s \* *acl*, char \*\* *name*, char \*\* *address*)

Definition at line 233 of file libmc.c.

References fipa\_agent\_identifier\_s::addresses, fipa\_agent\_identifier\_s::name, fipa\_acl\_message\_s::sender, fipa\_url\_s::str, strdup, and fipa\_url\_sequence\_s::urls.

Referenced by MC\_AclGetSender\_chdl().

#### 5.49.1.16 EXPORTCH int MC\_AclGetSender\_chdl (void \* *varg*)

Definition at line 2765 of file libmc.c.

References MC\_AclGetSender().

Referenced by agent\_ChScriptInitVar().

#### 5.49.1.17 EXPORTMC fipa\_acl\_message\_t\* MC\_AclNew (void) [read]

Allocate a new ACL Message.

##### Returns

A newly allocated and empty ACL message.

Definition at line 87 of file libmc.c.

References fipa\_acl\_message\_New().

Referenced by MC\_AclNew\_chdl().

#### 5.49.1.18 EXPORTMC void\* MC\_AclNew\_chdl (void \* *varg*)

Definition at line 2620 of file libmc.c.

References MC\_AclNew().

Referenced by agent\_ChScriptInitVar().

#### 5.49.1.19 EXPORTMC int MC\_AclPost (MCAgent\_t *agent*, struct fipa\_acl\_message\_s \* *message*)

Post ACL message to agent.

##### Parameters

*agent* The agent to post the message to

*message* The message to post

##### Returns

0 if successful, or error\_code\_t type.

Definition at line 92 of file libmc.c.

References agent\_mailbox\_Post(), and agent\_s::mailbox.

Referenced by MC\_AclPost\_chdl(), and MC\_AclSend().

#### 5.49.1.20 EXPORTMC int MC\_AclPost\_chdl (void \* *varg*)

Definition at line 2628 of file libmc.c.

References MC\_AclPost().

Referenced by agent\_ChScriptInitVar().

#### 5.49.1.21 EXPORTMC fipa\_acl\_message\_t\* MC\_AclReply (struct fipa\_acl\_message\_s \* *acl\_message*) [read]

Reply to an ACL message.

##### Parameters

*acl\_message* The incoming acl message to reply to

##### Returns

A newly allocated ACL message

##### Note

This function simply generates a new ACL message with the 'receiver' field automatically set to the 'sender' field of the incoming message.

Definition at line 98 of file libmc.c.

References fipa\_Reply().

Referenced by MC\_AclReply\_chdl().

#### 5.49.1.22 EXPORTCH void\* MC\_AclReply\_chdl (void \* *varg*)

Definition at line 2646 of file libmc.c.

References MC\_AclReply().

Referenced by agent\_ChScriptInitVar().

#### 5.49.1.23 EXPORTMC fipa\_acl\_message\_t\* MC\_AclRetrieve (MCAgent\_t *agent*) [read]

Retrieve an ACL message.

##### Parameters

*agent* Agent to retrieve message from.

##### Returns

an ACL message struct on success or NULL on failure

Definition at line 104 of file libmc.c.

References agent\_mailbox\_Retrieve(), and agent\_s::mailbox.

Referenced by MC\_AclRetrieve\_chdl().

#### 5.49.1.24 EXPORTCH void\* MC\_AclRetrieve\_chdl (void \* *varg*)

Definition at line 2662 of file libmc.c.

References MC\_AclRetrieve().

Referenced by agent\_ChScriptInitVar().

#### 5.49.1.25 EXPORTMC int MC\_AclSend (MCAgency\_t *attr*, struct fipa\_acl\_message\_s \* *acl*)

Send a composed ACL Message.

##### Parameters

*attr* An initialized and running MobileC agency

*acl* An allocated and fully composed ACL message.

##### Returns

0 if successful, error code on failure.

Definition at line 110 of file libmc.c.

References fipa\_agent\_identifier\_s::addresses, mtp\_http\_s::content, mtp\_http\_content\_s::content\_type, mtp\_http\_content\_s::data, dynstring\_Destroy(), FIPA\_ACL, fipa\_acl\_Compose(), fipa\_acl\_message\_Copy(), fipa\_agent\_identifier\_set\_s::fipa\_agent\_identifiers, fipa\_envelope\_Compose(), mtp\_http\_s::host, http\_to\_hostport(), MC\_AclPost(), MC\_FindAgentByName(), agency\_s::mc\_platform, dynstring\_s::message, mtp\_http\_s::message\_parts, message\_Send(), message\_s::message\_type, mtp\_http\_CreateMessage(), mtp\_http\_Destroy(), mtp\_http\_New(), fipa\_agent\_identifier\_s::name, fipa\_url\_sequence\_s::num, fipa\_agent\_identifier\_set\_s::num, port, mc\_platform\_s::private\_key, fipa\_acl\_message\_s::receiver, fipa\_url\_s::str, strdup, mtp\_http\_s::target, message\_s::target, and fipa\_url\_sequence\_s::urls.

Referenced by MC\_AclSend\_chdl().

#### 5.49.1.26 EXPORTCH int MC\_AclSend\_chdl (void \* *varg*)

Definition at line 2678 of file libmc.c.

References CHECK\_NULL, MC\_AclSend(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

#### 5.49.1.27 EXPORTMC int MC\_AclSetContent (fipa\_acl\_message\_t \* *acl*, const char \* *content*)

Definition at line 389 of file libmc.c.

References fipa\_string\_s::content, fipa\_acl\_message\_s::content, fipa\_string\_Destroy(), fipa\_string\_New(), and strdup.

Referenced by MC\_AclSetContent\_chdl().

**5.49.1.28 EXPORTCH int MC\_AclSetContent\_chdl (void \* *varg*)**

Definition at line 2913 of file libmc.c.

References fipa\_acl\_message\_s::content, and MC\_AclSetContent().

Referenced by agent\_ChScriptInitVar().

**5.49.1.29 EXPORTMC int MC\_AclSetConversationID (fipa\_acl\_message\_t \* *acl*, char \* *id*)**

Definition at line 268 of file libmc.c.

References fipa\_string\_s::content, fipa\_expression\_s::content, fipa\_acl\_message\_s::conversation\_id, FIPA\_EXPR\_STRING, fipa\_expression\_New(), fipa\_string\_New(), strdup, fipa\_expression\_s::content\_u::string, and fipa\_expression\_s::type.

Referenced by MC\_AclSetConversationID\_chdl().

**5.49.1.30 EXPORTCH int MC\_AclSetConversationID\_chdl (void \* *varg*)**

Definition at line 2817 of file libmc.c.

References MC\_AclSetConversationID().

Referenced by agent\_ChScriptInitVar().

**5.49.1.31 EXPORTMC int MC\_AclSetPerformative (fipa\_acl\_message\_t \* *acl*, enum fipa\_performative\_e *performative*)**

Definition at line 287 of file libmc.c.

References fipa\_acl\_message\_s::performative.

Referenced by MC\_AclSetPerformative\_chdl().

**5.49.1.32 EXPORTCH int MC\_AclSetPerformative\_chdl (void \* *varg*)**

Definition at line 2835 of file libmc.c.

References MC\_AclSetPerformative(), and fipa\_acl\_message\_s::performative.

Referenced by agent\_ChScriptInitVar().

**5.49.1.33 EXPORTMC int MC\_AclSetProtocol (fipa\_acl\_message\_t \* *acl*, enum fipa\_protocol\_e *protocol*)**

Definition at line 252 of file libmc.c.

References FIPA\_PROTOCOL\_END, FIPA\_PROTOCOL\_ERROR, and fipa\_acl\_message\_s::protocol.

Referenced by MC\_AclSetProtocol\_chdl().

**5.49.1.34 EXPORTCH int MC\_AclSetProtocol\_chdl (void \* *varg*)**

Definition at line 2799 of file libmc.c.

References MC\_AclSetProtocol(), and fipa\_acl\_message\_s::protocol.

Referenced by agent\_ChScriptInitVar().

#### 5.49.1.35 EXPORTMC int MC\_AclSetSender (fipa\_acl\_message\_t \* *acl*, const char \* *name*, const char \* *address*)

Definition at line 295 of file libmc.c.

References fipa\_agent\_identifier\_s::addresses, fipa\_agent\_identifier\_Destroy(), fipa\_agent\_identifier\_New(), fipa\_url\_New(), fipa\_url\_sequence\_New(), fipa\_agent\_identifier\_s::name, fipa\_url\_sequence\_s::num, fipa\_acl\_message\_s::sender, fipa\_url\_s::str, strdup, and fipa\_url\_sequence\_s::urls.

Referenced by MC\_AclSetSender\_chdl().

#### 5.49.1.36 EXPORTCH int MC\_AclSetSender\_chdl (void \* *varg*)

Definition at line 2853 of file libmc.c.

References MC\_AclSetSender().

Referenced by agent\_ChScriptInitVar().

#### 5.49.1.37 EXPORTMC fipa\_acl\_message\_t\* MC\_AclWaitRetrieve (MCAgent\_t *agent*) [read]

Wait for and retrieve an ACL message.

##### Parameters

*agent* Agent to retrieve message from.

##### Returns

an ACL message struct on success or NULL on failure

Definition at line 208 of file libmc.c.

References agent\_mailbox\_WaitRetrieve(), and agent\_s::mailbox.

Referenced by MC\_AclWaitRetrieve\_chdl().

#### 5.49.1.38 EXPORTCH void\* MC\_AclWaitRetrieve\_chdl (void \* *varg*)

Definition at line 2702 of file libmc.c.

References MC\_AclWaitRetrieve().

Referenced by agent\_ChScriptInitVar().

#### 5.49.1.39 EXPORTMC int MC\_AddAgent (MCAgency\_t *attr*, MCAgent\_t *agent*)

Add an agent to the agency 'attr'.

##### Parameters

*attr* a MobileC agency

*agent* An initialized MobileC agent

### Returns

0 if successful, or `error_code_t` type

Definition at line 406 of file `libmc.c`.

References `mc_platform_s::agent_queue`, `mc_platform_s::ams`, `COND_SIGNAL`, `agency_s::mc_platform`, `agent_s::mc_platform`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

Referenced by `handler_COMPOSE_SEND()`, and `MC_AddAgent_chdl()`.

#### 5.49.1.40 EXPORTCH int MC\_AddAgent\_chdl (void \* *varg*)

Definition at line 2933 of file `libmc.c`.

References `CHECK_NULL`, `MC_AddAgent()`, and `agency_s::mc_platform`.

Referenced by `agent_ChScriptInitVar()`.

#### 5.49.1.41 EXPORTMC int MC\_AddAgentInitCallback (MCAgency\_t *agency*, MC\_AgentInitCallbackFunc\_t *function*, void \* *user\_data*)

Add an agent initialization callback function.

### Parameters

*agency* a MobileC agency

*function* The callback function to add. This callback function will be called after an interpreter is initialized for an agent but before the agent is executed.

*user\_data* A pointer to any user data that should be made available to the callback function

### Returns

0 if successful, or `error_code_t` type

Definition at line 447 of file `libmc.c`.

References `agency_s::agentInitCallback`, and `agency_s::agentInitUserData`.

#### 5.49.1.42 EXPORTMC int MC\_AddStationaryAgent (MCAgency\_t *agency*, void \*(\*)(stationary\_agent\_info\_t \*) *agent\_thread*, const char \* *name*, void \* *agent\_args*)

Add a stationary agent to the agency.

### Parameters

*agency* a MobileC agency

*agent\_thread* The binary-space agent function

*name* The new agent's name

*agent\_args* user arguments to pass to the agent



**Returns**

0 if successful, or `error_code_t` type

Definition at line 419 of file `libmc.c`.

References `stationary_agent_info_s::agency`, `stationary_agent_info_s::agent`, `agent_NewBinary()`, `mc_platform_s::agent_queue`, `stationary_agent_info_s::args`, `stationary_agent_info_s::attr`, `agency_s::mc_platform`, `agent_s::name`, `strdup`, `stationary_agent_info_s::thread`, and `THREAD_CREATE`.

#### 5.49.1.43 EXPORTMC int MC\_AgentAddTask (MCAgent\_t agent, const char \* code, const char \* return\_var\_name, const char \* server, int persistent)

Add a new task to an already existing agent.

**Parameters**

*agent* An existing agent

*code* The C code to use in the new task

*return\_var\_name* The name of the the return variable, or NULL if there is no return variable.

*server* The server to perform the task on.

*persistent* Whether or not the task is persistent. '1' indicates persistent, and '0' indicates not-persistent (the default)

**Returns**

Returns a valid agency handle or NULL on failure.

Definition at line 472 of file `libmc.c`.

References `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_task_New()`, `agent_task_s::code_id`, `agent_s::datastate`, `agent_datastate_s::number_of_tasks`, `agent_task_s::persistent`, `agent_task_s::server_name`, `strdup`, `agent_datastate_s::tasks`, and `agent_task_s::var_name`.

Referenced by `MC_AgentAddTask_chdl()`, and `MC_AgentAddTaskFromFile()`.

#### 5.49.1.44 EXPORTCH int MC\_AgentAddTask\_chdl (void \* varg)

Definition at line 2957 of file `libmc.c`.

References `MC_AgentAddTask()`.

Referenced by `agent_ChScriptInitVar()`.

#### 5.49.1.45 EXPORTMC int MC\_AgentAddTaskFromFile (MCAgent\_t agent, const char \* filename, const char \* return\_var\_name, const char \* server, int persistent)

Add a new task to an already existing agent.

**Parameters**

*agent* An existing agent

*filename* The C code to use in the new task

*return\_var\_name* The name of the the return variable, or NULL if there is no return variable.

*server* The server to perform the task on.

*persistent* Whether or not the task is persistent. '1' indicates persistent, and '0' indicates not-persistent (the default)

### Returns

Returns a valid agency handle or NULL on failure.

Definition at line 545 of file libmc.c.

References MC\_AgentAddTask().

#### 5.49.1.46 EXPORTMC MCAgency\_t MC\_AgentInfo\_GetAgency (stationary\_agent\_info\_t \* stationary\_agent\_info)

Get the host Mobile-C agency handle from a "stationary\_agent\_info\_t" data structure.

### Parameters

*stationary\_agent\_info* A stationary\_agent\_info\_t type variable provided to a stationary agent function

### Returns

Returns a valid agency handle or NULL on failure.

Definition at line 457 of file libmc.c.

References stationary\_agent\_info\_s::agency.

#### 5.49.1.47 EXPORTMC MCAgent\_t MC\_AgentInfo\_GetAgent (stationary\_agent\_info\_t \* stationary\_agent\_info)

Retrieve an agent handle from the stationary agent info from within a stationary agent.

### Parameters

*stationary\_agent\_info* A stationary\_agent\_info\_t type variable provided to a stationary agent function

### Returns

Returns a valid agent handle or NULL on failure.

## 5.49.2 Examples

The following example demonstrates usage of [MC\\_AgentInfo\\_GetAgent\(\)](#)

Definition at line 462 of file libmc.c.

References stationary\_agent\_info\_s::agent.

#### 5.49.2.1 EXPORTMC void\* MC\_AgentInfo\_GetAgentArgs (stationary\_agent\_info\_t \* stationary\_agent\_info)

Retrieve the agent argument supplied as the "agent\_args" parameter in the [MC\\_AddStationaryAgent\(\)](#) function.

##### Parameters

*stationary\_agent\_info* A stationary\_agent\_info\_t type variable provided to a stationary agent function

##### Returns

Returns a valid pointer or NULL on failure.

Definition at line 467 of file libmc.c.

References stationary\_agent\_info\_s::args.

#### 5.49.2.2 EXPORTMC int MC\_AgentReturnArrayDim (MCAgent\_t agent, int task\_num)

Get the dimension of an agent's return data, if it is an array.

##### Parameters

*agent* A return agent

*task* task to retrieve the agent return data size from

##### Returns

Returns a positive integer on success or -1 on failure.

Definition at line 603 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::array\_dim, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

#### 5.49.2.3 EXPORTMC int MC\_AgentReturnArrayExtent (MCAgent\_t agent, int task\_num, int index)

Get the extent of a dimension of an agent's return data, if it is an array.

##### Parameters

*agent* A return agent

*task* task to retrieve the agent return data size from

*index* The dimension index to get the extent

##### Returns

Returns a positive integer on success or -1 on failure.

Definition at line 579 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::array\_dim, interpreter\_variable\_data\_s::array\_extent, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

#### 5.49.2.4 EXPORTMC int MC\_AgentReturnArrayNum (MCAgent\_t agent, int task\_num)

Get the total number of elements of the agent return data.

##### Parameters

*agent* A return agent  
*task* task to retrieve the agent return data size from  
*index* The dimension index to get the extent

##### Returns

Returns a positive integer on success or -1 on failure.

Definition at line 621 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::array\_dim, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

#### 5.49.2.5 EXPORTMC const void\* MC\_AgentReturnDataGetSymbolAddr (MCAgent\_t agent, int task\_num)

Get a pointer to the agent's return data.

##### Parameters

*agent* A return agent  
*task* task to retrieve the agent return data from

##### Returns

Returns a pointer to the beginning of the task's return data on success. Returns NULL on failure.

Definition at line 651 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::data, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

#### 5.49.2.6 EXPORTMC size\_t MC\_AgentReturnDataSize (MCAgent\_t agent, int task\_num)

Get the size of an agent's return data type.

##### Parameters

*agent* A return agent  
*task* task to retrieve the agent return data type size from

##### Returns

Returns a positive integer on success or -1 on failure.

Definition at line 668 of file libmc.c.

References agent\_task\_s::agent\_return\_data, CH\_DATATYPE\_SIZE, interpreter\_variable\_data\_s::data\_type, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, size, and agent\_datastate\_s::tasks.

### 5.49.2.7 EXPORTMC int MC\_AgentReturnDataType (MCAgent\_t *agent*, int *task\_num*)

Get the type of an agent's return data.

#### Parameters

*agent* A return agent

*task* task to retrieve the agent return data type from

#### Returns

Returns a positive integer on success or -1 on failure. The return type is of type ChType\_t, defined in ch.h

Definition at line 691 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::data\_type, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

### 5.49.2.8 EXPORTMC int MC\_AgentReturnIsArray (MCAgent\_t *agent*, int *task\_num*)

Check to see whether or not an agent's return data is an array.

#### Parameters

*agent* A return agent

*task* task to check to see if the return data is an array.

#### Returns

Returns 1 if the return variable is an array, 0 if it is not an array, or -1 on failure.

Definition at line 701 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::array\_dim, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::tasks.

### 5.49.2.9 EXPORTMC const void\* MC\_AgentVariableRetrieve (MCAgent\_t *agent*, const char \* *var\_name*, int *task\_num*)

Retrieve a pointer to a previously saved variable.

#### Parameters

*agent* A MobileC agent.

*var\_name* The name of the saved variable that has previously been saved.

*task\_num* The previous completed task from which to retrieve the saved variable.

#### Returns

A pointer to the data on success or NULL on failure.

### 5.49.3 Examples

The following example demonstrates usage of [MC\\_AgentVariableRetrieve\(\)](#) from agent space.

Definition at line 722 of file libmc.c.

References `agent_task_s::agent_variable_list`, `interpreter_variable_data_s::data`, `agent_s::datastate`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

Referenced by `MC_AgentVariableRetrieve_chdl()`.

#### 5.49.3.1 EXPORTCH const void\* MC\_AgentVariableRetrieve\_chdl (void \*varg)

Definition at line 2991 of file libmc.c.

References `MC_AgentVariableRetrieve()`.

Referenced by `agent_ChScriptInitVar()`.

#### 5.49.3.2 EXPORTMC int MC\_AgentVariableRetrieveInfo (MCAgent\_t agent, const char \* var\_name, int task\_num, const void \*\* data, int \* dim, const int \*\* extent)

Retrieve a info about a previously saved variable.

##### Parameters

*agent* A MobileC agent.

*var\_name* The name of the saved variable that has previously been saved.

*task\_num* The previous completed task from which to retrieve the saved variable.

*data* (Output) The Variable Data

*dim* (Output) The dimension of the data array

*extent* (Output) The extents of the output array

##### Returns

Error code.

Definition at line 740 of file libmc.c.

References `agent_task_s::agent_variable_list`, `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `interpreter_variable_data_s::data`, `agent_s::datastate`, `MC_ERR_NOT_FOUND`, `MC_SUCCESS`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

#### 5.49.3.3 EXPORTMC int MC\_AgentVariableSave (MCAgent\_t agent, const char \* var\_name)

Mark an agent variable for saving.

##### Parameters

*agent* A MobileC agent.

*var\_name* The name of the variable to mark for saving.

**Returns**

0 on success, non-zero on failure.

**See also**

test1.xml

**5.49.4 Examples**

See agent\_saved\_variables\_example/test1.xml for an example of usage of this api function.

**5.49.5 Examples**

The following example demonstrates usage of [MC\\_AgentVariableSave\(\)](#) from agent space.

Definition at line 761 of file libmc.c.

References agent\_s::datastate, MC\_ERR\_MEMORY, agent\_task\_s::num\_saved\_variables, agent\_task\_s::saved\_variables, strdup, agent\_datastate\_s::task\_progress, and agent\_datastate\_s::tasks.

Referenced by MC\_AgentVariableSave\_chdl().

**5.49.5.1 EXPORTCH int MC\_AgentVariableSave\_chdl (void \* *varg*)**

Definition at line 3015 of file libmc.c.

References MC\_AgentVariableSave().

Referenced by agent\_ChScriptInitVar().

**5.49.5.2 EXPORTMC int MC\_Barrier (MCAgency\_t *attr*, int *id*)**

Definition at line 782 of file libmc.c.

References mc\_platform\_s::barrier\_queue, barrier\_queue\_Get(), barrier\_node\_s::cond, COND\_BROADCAST, COND\_WAIT, barrier\_node\_s::lock, MC\_ERR\_NOT\_FOUND, agency\_s::mc\_platform, MC\_SUCCESS, MUTEX\_LOCK, MUTEX\_UNLOCK, node, barrier\_node\_s::num\_registered, and barrier\_node\_s::num\_waiting.

Referenced by MC\_Barrier\_chdl().

**5.49.5.3 EXPORTCH int MC\_Barrier\_chdl (void \* *varg*)**

Definition at line 3067 of file libmc.c.

References CHECK\_NULL, MC\_Barrier(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.49.5.4 EXPORTMC int MC\_BarrierDelete (MCAgency\_t *attr*, int *id*)**

Find and delete an initialized MobileC Barrier.

**Parameters**

- attr* A running MobileC agency
- id* The id of the barrier node to delete

**Returns**

returns 0 on success, error if the node is not found or other failure.

Definition at line 822 of file libmc.c.

References mc\_platform\_s::barrier\_queue, barrier\_queue\_Delete(), and agency\_s::mc\_platform.

Referenced by MC\_BarrierDelete\_chdl().

**5.49.5.5 EXPORTCH int MC\_BarrierDelete\_chdl (void \* *varg*)**

Definition at line 3090 of file libmc.c.

References CHECK\_NULL, MC\_BarrierDelete(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.49.5.6 EXPORTMC int MC\_BarrierInit (MCAgency\_t *attr*, int *id*, int *num\_procs*)**

Initialize a MobileC Barrier.

**Parameters**

- attr* A running MobileC agency
- id* The requested barrier id
- num\_procs* The number of agents/threads/processes that will wait on the barrier

**Returns**

The allocated barrier id. May differ from the requested id if it is already in use.

**5.49.6 Examples**

The following example demonstrates an agent which sets up an MC\_Barrier.

Definition at line 808 of file libmc.c.

References barrier\_node\_Initialize(), mc\_platform\_s::barrier\_queue, barrier\_queue\_Add(), barrier\_queue\_Get(), MC\_ERR, agency\_s::mc\_platform, MC\_SUCCESS, and node.

Referenced by MC\_BarrierInit\_chdl().

**5.49.6.1 EXPORTCH int MC\_BarrierInit\_chdl (void \* *varg*)**

Definition at line 3113 of file libmc.c.

References CHECK\_NULL, MC\_BarrierInit(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().



### 5.49.6.2 EXPORTMC int MC\_CallAgentFunc (MCAgent\_t *agent*, const char \**funcName*, void \**returnVal*, int *numArgs*, ...)

Use custom ChOptions\_t type for internal Ch interpreter.

#### Parameters

*attr* A running MobileC agency  
*options* Initialized Ch options structure

#### Returns

0 on success, error\_code\_t type on failure Calls a function defined in an agent

#### Parameters

*agent* An initialized and executed MobileC agent  
*funcName* The name of the function to call  
*returnVal* The agent function's return value  
*numArgs* The number of arguments supplied to the agent function  
... A variable argument list to be supplied to the agent function

#### Returns

0 if successful, error\_code\_t type on failure

## 5.49.7 Example

Definition at line 828 of file libmc.c.

References agent\_s::agent\_interp, MUTEX\_LOCK, MUTEX\_UNLOCK, and agent\_s::run\_lock.

### 5.49.7.1 EXPORTCH int MC\_CallAgentFunc\_chdl (void \* *varg*)

Definition at line 3037 of file libmc.c.

References MC\_CallAgentFuncVar().

Referenced by agent\_ChScriptInitVar().

### 5.49.7.2 EXPORTMC int MC\_CallAgentFuncArg (MCAgent\_t *agent*, const char \**funcName*, void \**returnVal*, void \**arg*)

Calls a function defined in an agent.

#### Parameters

*agent* An initialized and executed MobileC agent  
*funcName* The name of the function to call  
*returnVal* The agent function's return value

*arg* The agent functions argument

#### Note

The agent function must be of the form 'void\* func(void\* arg);'

#### Returns

0 if successful, error\_code\_t type on failure

### 5.49.8 Example

Definition at line 849 of file libmc.c.

References agent\_s::agent\_interp, MUTEX\_LOCK, MUTEX\_UNLOCK, and agent\_s::run\_lock.

#### 5.49.8.1 EXPORTMC int MC\_CallAgentFuncV (MCAgent\_t *agent*, const char \* *funcName*, void \* *returnVal*, va\_list *ap*)

Calls a function defined in an agent.

#### Parameters

*agent* An initialized and executed MobileC agent

*funcName* The name of the function to call

*returnVal* The agent function's return value

*ap* A variable argument list

#### Returns

0 if successful, error\_code\_t type on failure

Definition at line 868 of file libmc.c.

References agent\_s::agent\_interp, MUTEX\_LOCK, MUTEX\_UNLOCK, and agent\_s::run\_lock.

#### 5.49.8.2 EXPORTMC int MC\_CallAgentFuncVar (MCAgent\_t *agent*, const char \* *funcName*, void \* *returnVal*, ChVaList\_t *varg*)

Definition at line 888 of file libmc.c.

References agent\_s::agent\_interp, MUTEX\_LOCK, MUTEX\_UNLOCK, and agent\_s::run\_lock.

Referenced by MC\_CallAgentFunc\_chdl().

#### 5.49.8.3 EXPORTMC MCAgent\_t MC\_ComposeAgent (const char \* *name*, const char \* *home*, const char \* *owner*, const char \* *code*, const char \* *return\_var\_name*, const char \* *server*, int *persistent*)

Compose a new agent dynamically without using a prewritten XML file.

**Parameters**

- name* The desired name of the new agent.
- home* The home of the new agent.
- owner* The owner of the new agent.
- code* The agent code
- return\_var\_name* The name of the agent's return variable. Set to "no-return" if no return variable is desired.
- server* The target destination of the agent.
- persistent* A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

**Returns**

This function returns a valid MCAgent\_t structure on success or NULL on failure.

Definition at line 928 of file libmc.c.

References MC\_ComposeAgentWithWorkgroup().

**5.49.8.4 EXPORTCH MCAgent\_t MC\_ComposeAgent\_chdl (void \* varg)**

Definition at line 3161 of file libmc.c.

References MC\_ComposeAgentWithWorkgroup().

Referenced by agent\_ChScriptInitVar().

**5.49.8.5 EXPORTMC MCAgent\_t MC\_ComposeAgentFromFile (const char \* name, const char \* home, const char \* owner, const char \* filename, const char \* return\_var\_name, const char \* server, int persistent)**

Compose a new agent dynamically from a source code file.

**Parameters**

- filename* The filename of the source file
- name* The desired name of the new agent.
- home* The home of the new agent.
- owner* The owner of the new agent.
- code* The agent code
- return\_var\_name* The name of the agent's return variable. Set to "no-return" if no return variable is desired.
- server* The target destination of the agent.
- persistent* A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

**Returns**

This function returns a valid MCAgent\_t structure on success or NULL on failure.

Definition at line 1054 of file libmc.c.

References MC\_ComposeAgentFromFileWithWorkgroup().

Referenced by handler\_COMPOSE\_SEND().

#### 5.49.8.6 EXPORTCH MCAgent\_t MC\_ComposeAgentFromFile\_chdl (void \* varg)

Definition at line 3245 of file libmc.c.

References MC\_ComposeAgentFromFileWithWorkgroup().

Referenced by agent\_ChScriptInitVar().

#### 5.49.8.7 EXPORTMC MCAgent\_t MC\_ComposeAgentFromFileS (const char \* name, const char \* home, const char \* owner, const char \* filename, const char \* return\_var\_name, const char \* server, int persistent, const char \* workgroup\_code)

Compose a new agent dynamically from a source code file.

##### Parameters

*filename* The filename of the source file

*name* The desired name of the new agent.

*home* The home of the new agent.

*owner* The owner of the new agent.

*code* The agent code

*return\_var\_name* The name of the agent's return variable. Set to "no-return" if no return variable is desired.

*server* The target destination of the agent.

*workgroup\_code* The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

*persistant* A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

##### Returns

This function returns a valid MCAgent\_t structure on success or NULL on failure.

##### Note

This function is deprecated. Please use the MC\_ComposeAgentFromFileWithWorkgroup function instead.

Definition at line 1076 of file libmc.c.

References MC\_ComposeAgentFromFileWithWorkgroup().

#### 5.49.8.8 EXPORTMC MCAgent\_t MC\_ComposeAgentFromFileWithWorkgroup (const char \* name, const char \* home, const char \* owner, const char \* filename, const char \* return\_var\_name, const char \* server, int persistent, const char \* workgroup\_code)

Compose a new agent dynamically from a source code file.

##### Parameters

*filename* The filename of the source file

*name* The desired name of the new agent.

**home** The home of the new agent.

**owner** The owner of the new agent.

**code** The agent code

**return\_var\_name** The name of the agent's return variable. Set to "no-return" if no return variable is desired.

**server** The target destination of the agent.

**workgroup\_code** The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

**persistant** A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

## Returns

This function returns a valid MCAgent\_t structure on success or NULL on failure.

Definition at line 1101 of file libmc.c.

References MC\_ComposeAgentWithWorkgroup().

Referenced by MC\_ComposeAgentFromFile(), MC\_ComposeAgentFromFile\_chdl(), MC\_ComposeAgentFromFileS(), and MC\_ComposeAgentFromFileWithWorkgroup\_chdl().

### 5.49.8.9 EXPORTCH MCAgent\_t MC\_ComposeAgentFromFileWithWorkgroup\_chdl (void \*varg)

Definition at line 3286 of file libmc.c.

References MC\_ComposeAgentFromFileWithWorkgroup().

Referenced by agent\_ChScriptInitVar().

### 5.49.8.10 EXPORTMC MCAgent\_t MC\_ComposeAgentS (const char \* name, const char \* home, const char \* owner, const char \* code, const char \* return\_var\_name, const char \* server, int persistent, const char \* workgroup\_code)

Compose a new agent dynamically without using a prewritten XML file.

## Parameters

**name** The desired name of the new agent.

**home** The home of the new agent.

**owner** The owner of the new agent.

**code** The agent code

**return\_var\_name** The name of the agent's return variable. Set to "no-return" if no return variable is desired.

**server** The target destination of the agent.

**workgroup\_code** The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

**persistant** A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

**Returns**

This function returns a valid `MCAgent_t` structure on success or `NULL` on failure.

**Note**

This function is deprecated. Please use the `MC_ComposeAgentWithWorkgroup` function instead.

Definition at line 951 of file `libmc.c`.

References `MC_ComposeAgentWithWorkgroup()`.

Referenced by `MC_ComposeAgentWithWorkgroup_chdl()`.

#### 5.49.8.11 EXPORTMC MCAgent\_t MC\_ComposeAgentWithWorkgroup (const char \* name, const char \* home, const char \* owner, const char \* code, const char \* return\_var\_name, const char \* server, int persistent, const char \* workgroup\_code)

Compose a new agent dynamically without using a prewritten XML file.

**Parameters**

**name** The desired name of the new agent.

**home** The home of the new agent.

**owner** The owner of the new agent.

**code** The agent code

**return\_var\_name** The name of the agent's return variable. Set to "no-return" if no return variable is desired.

**server** The target destination of the agent.

**workgroup\_code** The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

**persistent** A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

**Returns**

This function returns a valid `MCAgent_t` structure on success or `NULL` on failure.

**Note**

This function is deprecated

Definition at line 977 of file `libmc.c`.

References `agent_datastate_s::agent_code`, `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_datastate_New()`, `agent_New()`, `agent_s::agent_status`, `agent_task_New()`, `agent_s::agent_type`, `agent_s::datastate`, `agent_s::home`, `MC_LOCAL_AGENT`, `MC_WAIT_MESSGSEND`, `agent_s::name`, `agent_datastate_s::number_of_tasks`, `agent_s::orphan`, `agent_s::owner`, `agent_task_s::persistent`, `agent_task_s::server_name`, `strdup`, `agent_datastate_s::tasks`, `agent_task_s::var_name`, and `agent_s::wg_code`.

Referenced by `MC_ComposeAgent()`, `MC_ComposeAgent_chdl()`, `MC_ComposeAgentFromFileWithWorkgroup()`, and `MC_ComposeAgentS()`.

**5.49.8.12 EXPORTCH MC\_Agent\_t MC\_ComposeAgentWithWorkgroup\_chdl (void \* *varg*)**

Definition at line 3202 of file libmc.c.

References MC\_ComposeAgentS().

Referenced by agent\_ChScriptInitVar().

**5.49.8.13 EXPORTMC int MC\_CondBroadcast (MC\_Agency\_t *attr*, int *id*)**

Wakes up all agents/threads waiting on a condition variable.

**Parameters**

*attr* A MobileC agency

*id* Synchronization variable id to broadcast to

**See also**

[MC\\_SyncInit\(\)](#), [MC\\_CondSignal\(\)](#)

**Returns**

0 on success, error\_code\_t type on failure

Definition at line 1145 of file libmc.c.

References syncListNode\_s::cond, COND\_BROADCAST, syncListNode\_s::lock, MC\_ERR\_NOT\_FOUND, agency\_s::mc\_platform, MUTEX\_LOCK, MUTEX\_UNLOCK, syncListNode\_s::signalled, mc\_platform\_s::syncList, and syncListFind().

Referenced by MC\_CondBroadcast\_chdl().

**5.49.8.14 EXPORTCH int MC\_CondBroadcast\_chdl (void \* *varg*)**

Definition at line 3138 of file libmc.c.

References CHECK\_NULL, MC\_CondBroadcast(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.49.8.15 EXPORTMC int MC\_CondReset (MC\_Agency\_t *attr*, int *id*)**

Reset a previously signalled MobileC condition variable.

**Parameters**

*attr* A MobileC Agency

*id* The synchronization variable id to reset

**See also**

[MC\\_SyncInit\(\)](#)

**Returns**

0 on success, error\_code\_t type on failure

Definition at line 1197 of file libmc.c.

References `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `syncListNode_s::signalled`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_CondReset_chdl()`.

#### 5.49.8.16 EXPORTCH int MC\_CondReset\_chdl (void \* *varg*)

Definition at line 3352 of file libmc.c.

References `CHECK_NULL`, `MC_CondReset()`, and `agency_s::mc_platform`.

Referenced by `agent_ChScriptInitVar()`.

#### 5.49.8.17 EXPORTMC int MC\_CondSignal (MCAgency\_t *attr*, int *id*)

Wakes up at least one thread waiting on a condition variable.

##### Parameters

*attr* A MobileC agency

*id* synchronization variable id

##### See also

[MC\\_SyncInit\(\)](#)

##### Returns

0 on success, `error_code_t` type on failure

### 5.49.9 Example

The following example demonstrates the agent-space version of the function, which is nearly identical to the binary space api function.

Definition at line 1160 of file libmc.c.

References `syncListNode_s::cond`, `COND_SIGNAL`, `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `syncListNode_s::signalled`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_CondSignal_chdl()`.

#### 5.49.9.1 EXPORTCH int MC\_CondSignal\_chdl (void \* *varg*)

Definition at line 3329 of file libmc.c.

References `CHECK_NULL`, `MC_CondSignal()`, and `agency_s::mc_platform`.

Referenced by `agent_ChScriptInitVar()`.



### 5.49.9.2 EXPORTMC int MC\_CondWait (MCAgency\_t *attr*, int *id*)

Wait on a MobileC synchronization variable.

#### Parameters

*attr* A MobileC agency

*id* a synchronization variable id

#### See also

[MC\\_SyncInit\(\)](#)

#### Returns

0 on success, error\_code\_t type on failure

### 5.49.10 Example

The following example demonstrates the agent-space version of this function.

Definition at line 1175 of file libmc.c.

References syncListNode\_s::cond, COND\_WAIT, syncListNode\_s::lock, MC\_ERR\_NOT\_FOUND, agency\_s::mc\_platform, MUTEX\_LOCK, MUTEX\_UNLOCK, syncListNode\_s::signalled, mc\_platform\_s::syncList, and syncListFind().

Referenced by MC\_CondWait\_chdl().

#### 5.49.10.1 EXPORTCH int MC\_CondWait\_chdl (void \* *varg*)

Definition at line 3375 of file libmc.c.

References CHECK\_NULL, MC\_CondWait(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

#### 5.49.10.2 EXPORTMC int MC\_CopyAgent (MCAgent\_t \* *agent\_out*, const MCAgent\_t *agent\_in*)

Performs a deep-copy of an agent structure.

#### Parameters

*agent\_out* A pointer to the agent to copy to.

*agent\_in* The agent to copy

#### Returns

0 on success, error\_code\_t type on failure.

Definition at line 1215 of file libmc.c.

References agent\_Copy(), and MC\_SUCCESS.

### 5.49.10.3 EXPORTMC int MC\_DeleteAgent (MCAgent\_t *agent*)

Stop and remove an agent.

#### Parameters

*agent* An agent in any state (running, waiting, etc)

#### Returns

0 on success, error\_code\_t type on failure

Definition at line 1222 of file libmc.c.

References CHECK\_NULL, MC\_ERR\_INVALID, MC\_ERR\_INVALID\_ARGS, MC\_SUCCESS, MC\_TerminateAgent(), and agent\_s::wg\_code.

Referenced by MC\_DeleteAgent\_chdl().

### 5.49.10.4 EXPORTCH int MC\_DeleteAgent\_chdl (void \* *varg*)

Definition at line 3397 of file libmc.c.

References MC\_DeleteAgent(), MC\_ERR\_NOT\_FOUND, and MC\_FindAgentByName().

Referenced by agent\_ChScriptInitVar().

### 5.49.10.5 EXPORTMC int MC\_DeleteAgentWG (MCAgent\_t *calling\_agent*, MCAgent\_t *agent*)

Stop and remove an agent in the same workgroup.

#### Parameters

*calling\_agent* The calling agent

*agent* An agent in any state (running, waiting, etc)

#### Note

The agents must belong to the same workgroup.

#### Returns

0 on success, error\_code\_t type on failure

Definition at line 1239 of file libmc.c.

References CHECK\_NULL, MC\_ERR\_INVALID, MC\_ERR\_INVALID\_ARGS, MC\_SUCCESS, MC\_TerminateAgentWG(), and agent\_s::wg\_code.

Referenced by MC\_DeleteAgentWG\_chdl().

### 5.49.10.6 EXPORTCH int MC\_DeleteAgentWG\_chdl (void \* *varg*)

Definition at line 3417 of file libmc.c.

References MC\_DeleteAgentWG(), MC\_ERR\_NOT\_FOUND, and MC\_FindAgentByName().

Referenced by agent\_ChScriptInitVar().

**5.49.10.7 EXPORTMC int MC\_DeregisterService (MCAgency\_t *agency*, int *agentID*, const char \* *serviceName*)**

Definition at line 1279 of file libmc.c.

References mc\_platform\_s::df, df\_AddRequest(), df\_request\_list\_node\_New(), and agency\_s::mc\_platform.

Referenced by MC\_DeregisterService\_chdl().

**5.49.10.8 EXPORTCH int MC\_DeregisterService\_chdl (void \* *varg*)**

Definition at line 3466 of file libmc.c.

References CHECK\_NULL, MC\_DeregisterService(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.49.10.9 EXPORTMC int MC\_DestroyServiceSearchResult (char \*\* *agentName*, char \*\* *serviceName*, int \* *agentID*, int *numResult*)**

Free memory allocated by a Service Search operation.

**Parameters**

*agentName* agent names returned by a search operation.

*serviceName* service names return by a search operation.

*agentID* list of agent id's returned by a search operation.

*numResult* The number of hits returned by a search operation.

**Returns**

0 on success, error code on failure.

Definition at line 1259 of file libmc.c.

Referenced by MC\_DestroyServiceSearchResult\_chdl().

**5.49.10.10 EXPORTCH int MC\_DestroyServiceSearchResult\_chdl (void \* *varg*)**

Definition at line 3439 of file libmc.c.

References MC\_DestroyServiceSearchResult().

Referenced by agent\_ChScriptInitVar().

**5.49.10.11 EXPORTMC int MC\_End (MCAgency\_t *attr*)**

End an agency.

**Parameters**

*attr* A running agency

**Returns**

0 on success, error\_code\_t type on failure

### 5.49.11 Example

Definition at line 1308 of file libmc.c.

References mc\_platform\_s::acc, mc\_platform\_s::ams, buf, mc\_platform\_s::cmd\_prompt, COND\_BROADCAST, COND\_SIGNAL, mc\_platform\_s::connection\_queue, mc\_platform\_s::df, GET\_THREAD\_MODE, agency\_s::hostName, agency\_s::mc\_platform, mc\_platform\_Destroy(), MC\_THREAD\_ACC, MC\_THREAD\_AMS, MC\_THREAD\_CP, MC\_THREAD\_DF, mc\_platform\_s::message\_queue, MUTEX\_LOCK, MUTEX\_UNLOCK, agency\_s::portno, mc\_platform\_s::quit, mc\_platform\_s::quit\_cond, mc\_platform\_s::quit\_lock, send, cmd\_prompt\_s::thread, THREAD\_CANCEL, THREAD\_JOIN, and agency\_s::threads.

Referenced by MC\_End\_chdl().

#### 5.49.11.1 EXPORTCH int MC\_End\_chdl (void \* *varg*)

Definition at line 3493 of file libmc.c.

References CHECK\_NULL, MC\_End(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

#### 5.49.11.2 EXPORTMC MCAgent\_t MC\_FindAgentByID (MCAgency\_t *attr*, int *ID*)

Find an agent by its id.

##### Parameters

*attr* the agency to search

*ID* the id to search for

##### Returns

a valid agent on success, NULL on failure

Definition at line 1423 of file libmc.c.

References mc\_platform\_s::agent\_queue, and agency\_s::mc\_platform.

Referenced by MC\_FindAgentByID\_chdl().

#### 5.49.11.3 EXPORTCH MCAgent\_t MC\_FindAgentByID\_chdl (void \* *varg*)

Definition at line 3510 of file libmc.c.

References CHECK\_NULL, MC\_FindAgentByID(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

#### 5.49.11.4 EXPORTMC MCAgent\_t MC\_FindAgentByName (MCAgency\_t *attr*, const char \* *name*)

Find an agent by its name.

**Parameters**

*attr* a running agency  
*name* name to search for

**Returns**

a valid agent on success or NULL on failure

**5.49.12 Example**

Definition at line 1410 of file libmc.c.

References mc\_platform\_s::agent\_queue, and agency\_s::mc\_platform.

Referenced by MC\_AclSend(), MC\_DeleteAgent\_chdl(), MC\_DeleteAgentWG\_chdl(), MC\_FindAgentByName\_chdl(), MC\_TerminateAgent\_chdl(), and MC\_TerminateAgentWG\_chdl().

**5.49.12.1 EXPORTCH MCAgent\_t MC\_FindAgentByName\_chdl (void \* *varg*)**

Definition at line 3533 of file libmc.c.

References CHECK\_NULL, MC\_FindAgentByName(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.49.12.2 EXPORTMC time\_t MC\_GetAgentArrivalTime (MCAgent\_t *agent*)**

Definition at line 1441 of file libmc.c.

References agent\_s::arrival\_time.

Referenced by MC\_GetAgentArrivalTime\_chdl().

**5.49.12.3 EXPORTCH time\_t MC\_GetAgentArrivalTime\_chdl (void \* *varg*)**

Definition at line 3560 of file libmc.c.

References MC\_GetAgentArrivalTime().

**5.49.12.4 EXPORTMC void\* MC\_GetAgentExecEngine (MCAgent\_t *agent*)**

Retrieve an agent's Ch interpreter.

**Parameters**

*agent* a valid agent

**Returns**

a Ch interpreter of type 'ChInterp\_t' on success, or NULL on failure.

Definition at line 1478 of file libmc.c.

References agent\_s::agent\_interp.

**5.49.12.5 EXPORTMC int MC\_GetAgentID (MCAgent\_t *agent*)**

Retrieve an agent's id.

Definition at line 1484 of file libmc.c.

References agent\_s::id.

Referenced by MC\_GetAgentID\_chdl().

**5.49.12.6 EXPORTCH int MC\_GetAgentID\_chdl (void \* *varg*)**

Definition at line 3580 of file libmc.c.

References MC\_GetAgentID().

Referenced by agent\_ChScriptInitVar().

**5.49.12.7 EXPORTMC char\* MC\_GetAgentName (MCAgent\_t *agent*)**

Definition at line 1492 of file libmc.c.

References agent\_s::lock, MUTEX\_LOCK, MUTEX\_UNLOCK, and agent\_s::name.

Referenced by MC\_GetAgentName\_chdl().

**5.49.12.8 EXPORTCH char\* MC\_GetAgentName\_chdl (void \* *varg*)**

Definition at line 3596 of file libmc.c.

References MC\_GetAgentName().

Referenced by agent\_ChScriptInitVar().

**5.49.12.9 EXPORTMC int MC\_GetAgentNumTasks (MCAgent\_t *agent*)**

Retrive the number of tasks an agent has.

**5.49.13 Example**

Definition at line 1575 of file libmc.c.

References agent\_s::datastate, and agent\_datastate\_s::number\_of\_tasks.

Referenced by MC\_GetAgentNumTasks\_chdl().

**5.49.13.1 EXPORTCH int MC\_GetAgentNumTasks\_chdl (void \* *varg*)**

Definition at line 3612 of file libmc.c.

References MC\_GetAgentNumTasks().

### 5.49.13.2 EXPORTMC int MC\_GetAgentReturnData (MCAgent\_t *agent*, int *task\_num*, void \*\**data*, int \**dim*, int \*\**extent*)

Get an agent's return data.

#### Parameters

*agent* a valid agent

*task\_num* the task for which to retrieve the return data. The task must already be completed.

*data* the return data. May be multi dimensional array.

*dim* the number of dimensions of the return array.

*extent* the extent of each one of the array dimensions.

### 5.49.14 Example

This file demonstrates the retrieval of agent return data from an agent

This is the agent which gets the data

Definition at line 1510 of file libmc.c.

References agent\_task\_s::agent\_return\_data, interpreter\_variable\_data\_s::array\_dim, interpreter\_variable\_data\_s::array\_extent, CH\_DATATYPE\_SIZE, interpreter\_variable\_data\_s::data\_type, agent\_s::datastate, agent\_datastate\_s::number\_of\_tasks, size, and agent\_datastate\_s::tasks.

#### 5.49.14.1 EXPORTMC int MC\_GetAgentStatus (MCAgent\_t *agent*)

Get an agent's current status.

#### Returns

returns type 'enum MC\_AgentStatus\_e'

Definition at line 1456 of file libmc.c.

References agent\_s::agent\_status, agent\_s::lock, MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_GetAgentStatus\_chdl().

#### 5.49.14.2 EXPORTCH int MC\_GetAgentStatus\_chdl (void \**varg*)

Definition at line 3628 of file libmc.c.

References MC\_GetAgentStatus().

Referenced by agent\_ChScriptInitVar().

**5.49.14.3 EXPORTMC enum MC\_AgentType\_e MC\_GetAgentType (MCAgent\_t *agent*)**

Get an agent's type.

**Returns**

returns type 'enum MC\_AgentType\_e'

Definition at line 1581 of file libmc.c.

References agent\_s::agent\_type.

**5.49.14.4 EXPORTMC char\* MC\_GetAgentXMLString (MCAgent\_t *agent*)**

Get an agent's xml string.

**Returns**

a malloc'd character string containing the agent's xml code

Definition at line 1466 of file libmc.c.

References agent\_s::datastate, mxmlSaveAllocString(), and agent\_datastate\_s::xml\_agent\_root.

Referenced by MC\_GetAgentXMLString\_chdl().

**5.49.14.5 EXPORTCH char\* MC\_GetAgentXMLString\_chdl (void \* *varg*)**

Definition at line 3644 of file libmc.c.

References MC\_GetAgentXMLString().

Referenced by agent\_ChScriptInitVar().

**5.49.14.6 EXPORTMC int MC\_GetAllAgents (MCAgency\_t *attr*, MCAgent\_t \*\* *agents*, int \* *num\_agents*)**

Definition at line 1591 of file libmc.c.

References mc\_platform\_s::agent\_queue, mc\_platform\_s::giant, mc\_platform\_s::giant\_lock, MC\_HaltAgency(), agency\_s::mc\_platform, MC\_ResumeAgency(), MUTEX\_LOCK, and MUTEX\_UNLOCK.

**5.49.14.7 EXPORTCH int MC\_GetTimeOfDay\_chdl (void \* *varg*)**

Definition at line 3660 of file libmc.c.

Referenced by agent\_ChScriptInitVar().

**5.49.14.8 EXPORTMC int MC\_HaltAgency (MCAgency\_t *agency*)**

Halt an agency: Do not process new entries in queues.



**Parameters**

*agency* A handle to a running MobileC agency.

**Returns**

0 on success, non-zero on failure.

Definition at line 1627 of file libmc.c.

References mc\_platform\_s::giant, mc\_platform\_s::giant\_lock, agency\_s::mc\_platform, MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_GetAllAgents(), and MC\_HaltAgency\_chdl().

**5.49.14.9 EXPORTCH int MC\_HaltAgency\_chdl (void \* *varg*)**

Definition at line 3675 of file libmc.c.

References CHECK\_NULL, MC\_HaltAgency(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.49.14.10 EXPORTMC MCAgency\_t MC\_Initialize (int *port*, MCAgencyOptions\_t \* *options*)**

Initialize and start a MobileC agency.

**Parameters**

*port* the TCP port the agency should bind to

*options* initialized MobileC options or NULL for default options

**Returns**

a handle to a running MobileC agency or NULL on failure

**5.49.15 Example**

Definition at line 1636 of file libmc.c.

References agency\_s::agentInitCallback, agency\_s::agentInitUserData, MCAgencyOptions\_s::bluetooth, agency\_s::bluetooth, buf, MCAgencyOptions\_s::ch\_options, agency\_s::client, MCAgencyOptions\_s::default\_agent\_status, agency\_s::default\_agentstatus, f, agency\_s::hostName, MCAgencyOptions\_s::initInterps, agency\_s::initInterps, agency\_s::known\_host\_filename, MCAgencyOptions\_s::known\_host\_filename, MC\_InitializeAgencyOptions(), mc\_platform, agency\_s::mc\_platform, mc\_platform\_Initialize(), MC\_THREAD\_ALL, MCAgencyOptions\_s::passphrase, agency\_s::portno, agency\_s::priv\_key\_filename, MCAgencyOptions\_s::priv\_key\_filename, read\_encrypted\_file(), agency\_s::server, MCAgencyOptions\_s::stack\_size, agency\_s::stack\_size, strdup, MCAgencyOptions\_s::threads, and agency\_s::threads.

**5.49.15.1 EXPORTMC int MC\_InitializeAgencyOptions (struct MCAgencyOptions\_s \* *options*)**

Initialize MobileC options.

**Parameters**

*options* options to initialize.

**Returns**

0 on success, `error_code_t` on failure

**Note**

MobileC options should be initialized with this function before any of its members are modified.

**5.49.16 Example**

Definition at line 1749 of file `libmc.c`.

References `MCAgencyOptions_s::bluetooth`, `MCAgencyOptions_s::default_agent_status`, `MCAgencyOptions_s::initInterps`, `MCAgencyOptions_s::known_host_filename`, `MC_THREAD_ALL`, `MC_WAIT_CH`, `MCAgencyOptions_s::modified`, `MCAgencyOptions_s::passphrase`, `MCAgencyOptions_s::priv_key_filename`, `MCAgencyOptions_s::stack_size`, and `MCAgencyOptions_s::threads`.

Referenced by `MC_Initialize()`.

**5.49.16.1 EXPORTMC int MC\_LoadAgentFromFile (MCAgency\_t attr, const char \*filename)**

Load an agent from a file into an agency.

**Parameters**

*agency* A valid and running Mobile-C agency  
*filename* Filename containing the agent to load

**Returns**

0 on success, non-zero on failure.

Definition at line 1771 of file `libmc.c`.

References `buf`, `agency_s::mc_platform`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, `mc_platform_s::message_queue`, `MXML_DESCEND`, `mxmlFindElement()`, `mxmlLoadString()`, `message_s::to_address`, `message_s::xml_payload`, and `message_s::xml_root`.

**5.49.16.2 EXPORTMC int MC\_MainLoop (MCAgency\_t attr)**

Wait indefinitely.

**Note**

This function is intended to block the calling thread forever.

Definition at line 2535 of file `libmc.c`.

References `COND_WAIT`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_platform_s::quit`, `mc_platform_s::quit_cond`, and `mc_platform_s::quit_lock`.

### 5.49.16.3 EXPORTMC int MC\_MigrateAgent (MCAgent\_t *agent*, const char \* *hostname*, int *port*)

Migrates a running agent to another host.

#### Parameters

*agent* The agent to migrate

*hostname* The new host to migrate the agent to

*port* The new port to migrate the agent to

#### Returns

0 on success, error\_code\_t type on failure.

Definition at line 1841 of file libmc.c.

References agent\_s::datastate, agent\_datastate\_s::progress\_modifier, agent\_task\_s::server\_name, agent\_datastate\_s::task\_progress, and agent\_datastate\_s::tasks.

Referenced by MC\_MigrateAgent\_chdl().

### 5.49.16.4 EXPORTCH int MC\_MigrateAgent\_chdl (void \* *varg*)

Definition at line 3692 of file libmc.c.

References MC\_MigrateAgent(), and port.

Referenced by agent\_ChScriptInitVar().

### 5.49.16.5 EXPORTMC int MC\_MutexLock (MCAgency\_t *attr*, int *id*)

Locks a MobileC synchronization variable as a mutex.

#### Parameters

*attr* a MobileC agency handle

*id* the synchronization variable id to lock

#### Returns

0 on success, error\_code\_t type on failure

## 5.49.17 Example

Consider the following agents, which use the agent-space version of this api function. Note that the 'sleep' agent is sent first, followed by the 'wake' agent.

Definition at line 1863 of file libmc.c.

References syncListNode\_s::lock, agency\_s::mc\_platform, MUTEX\_LOCK, mc\_platform\_s::syncList, and syncListFind().

Referenced by MC\_MutexLock\_chdl().

**5.49.17.1 EXPORTCH int MC\_MutexLock\_chdl (void \* *varg*)**

Definition at line 3711 of file libmc.c.

References CHECK\_NULL, MC\_MutexLock(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.49.17.2 EXPORTMC int MC\_MutexUnlock (MCAgency\_t *attr*, int *id*)**

Definition at line 1875 of file libmc.c.

References syncListNode\_s::lock, agency\_s::mc\_platform, MUTEX\_UNLOCK, mc\_platform\_s::syncList, and syncListFind().

Referenced by MC\_MutexUnlock\_chdl().

**5.49.17.3 EXPORTCH int MC\_MutexUnlock\_chdl (void \* *varg*)**

Definition at line 3735 of file libmc.c.

References CHECK\_NULL, MC\_MutexUnlock(), and agency\_s::mc\_platform.

Referenced by agent\_ChScriptInitVar().

**5.49.17.4 EXPORTMC int MC\_PrintAgentCode (MCAgent\_t *agent*)**

Prints an agents code to stdout.

**Returns**

0 on success, error\_code\_t on failure

Definition at line 1887 of file libmc.c.

References agent\_datastate\_s::agent\_code, agent\_s::datastate, agent\_s::lock, MUTEX\_LOCK, MUTEX\_UNLOCK, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::task\_progress.

Referenced by MC\_PrintAgentCode\_chdl().

**5.49.17.5 EXPORTCH int MC\_PrintAgentCode\_chdl (void \* *varg*)**

Definition at line 3759 of file libmc.c.

References MC\_PrintAgentCode().

Referenced by agent\_ChScriptInitVar().

**5.49.17.6 EXPORTMC int MC\_RegisterService (MCAgency\_t *agency*, MCAgent\_t *agent*, int *agentID*, const char \* *agentName*, char \*\* *serviceNames*, int *numServices*)**

Register a new service with the Directory Facilitator.

**Parameters**

*agency* a MobileC agency handle

*agent* (OPTIONAL: See note) a MobileC agent  
*agentID* (OPTIONAL: See note) a MobileC agent id  
*agentName* (OPTIONAL: See note) a MobileC agent name  
*serviceNames* an array of character strings of service names  
*numServices* the number of services described in 'serviceNames'

### Returns

0 on success, error\_code\_t type on failure

### Note

Three of the input arguments are optional. The function expects as input the arguments 'agent XOR (agentID AND agentName)'.

## 5.49.18 Example

Definition at line 1903 of file libmc.c.

References CHECK\_NULL, mc\_platform\_s::df, df\_AddRequest(), df\_request\_list\_node\_New(), agent\_s::id, MC\_ERR\_INVALID\_ARGS, MC\_ERR\_MEMORY, agency\_s::mc\_platform, MUTEX\_INIT, MUTEX\_T, and agent\_s::name.

Referenced by MC\_RegisterService\_chdl().

### 5.49.18.1 EXPORTCH int MC\_RegisterService\_chdl (void \* varg)

Definition at line 3775 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_RegisterService().

Referenced by agent\_ChScriptInitVar().

### 5.49.18.2 EXPORTMC int MC\_ResetSignal (MCAgency\_t attr)

Reset a MobileC signal.

### Returns

0 on success, error\_code\_t on failure

### See also

[MC\\_WaitSignal\(\)](#)

Definition at line 2035 of file libmc.c.

References COND\_SIGNAL, mc\_platform\_s::giant, mc\_platform\_s::giant\_cond, mc\_platform\_s::giant\_lock, MC\_NO\_SIGNAL, agency\_s::mc\_platform, mc\_platform\_s::MC\_signal, MUTEX\_LOCK, and MUTEX\_UNLOCK.

### 5.49.18.3 EXPORTMC int MC\_ResumeAgency (MCAgency\_t *agency*)

Resumes a halted agency.

#### Parameters

*agency* An agency previously halted with the [MC\\_HaltAgency\(\)](#) function.

#### Returns

0 on success, non-zero on failure.

Definition at line 1982 of file libmc.c.

References mc\_platform\_s::giant, mc\_platform\_s::giant\_lock, agency\_s::mc\_platform, MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_GetAllAgents(), and MC\_ResumeAgency\_chdl().

### 5.49.18.4 EXPORTCH int MC\_ResumeAgency\_chdl (void \* *varg*)

Definition at line 3812 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_ResumeAgency().

Referenced by agent\_ChScriptInitVar().

### 5.49.18.5 EXPORTMC MCAgent\_t MC\_RetrieveAgent (MCAgency\_t *attr*)

Retrieves the oldest agent from an agency.

#### Returns

a valid agent or NULL on failure

Definition at line 1991 of file libmc.c.

References mc\_platform\_s::agent\_queue, agent\_s::agent\_status, ListSearch(), MC\_AGENT\_NEUTRAL, agency\_s::mc\_platform, MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_RetrieveAgent\_chdl().

### 5.49.18.6 EXPORTCH MCAgent\_t MC\_RetrieveAgent\_chdl (void \* *varg*)

Definition at line 3830 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_RetrieveAgent().

Referenced by agent\_ChScriptInitVar().

### 5.49.18.7 EXPORTMC char\* MC\_RetrieveAgentCode (MCAgent\_t *agent*)

Retrieves an agent's Ch code.

#### Returns

a malloc'd character string on success, NULL on failure

Definition at line 2019 of file libmc.c.

References agent\_datastate\_s::agent\_code, buf, agent\_s::datastate, agent\_s::lock, MUTEX\_LOCK, MUTEX\_UNLOCK, and agent\_datastate\_s::task\_progress.

Referenced by MC\_RetrieveAgentCode\_chdl().

#### 5.49.18.8 EXPORTCH char\* MC\_RetrieveAgentCode\_chdl (void \* varg)

Definition at line 3847 of file libmc.c.

References MC\_RetrieveAgentCode().

Referenced by agent\_ChScriptInitVar().

#### 5.49.18.9 EXPORTCH int MC\_SaveData\_chdl (void \* varg)

Definition at line 3863 of file libmc.c.

References agent\_task\_s::agent\_variable\_list, interpreter\_variable\_data\_s::data, interpreter\_variable\_data\_s::data\_type, agent\_s::datastate, interpreter\_variable\_data\_New(), interpreter\_variable\_data\_s::name, interpreter\_variable\_data\_s::size, size, strdup, agent\_datastate\_s::task\_progress, and agent\_datastate\_s::tasks.

Referenced by agent\_ChScriptInitVar().

#### 5.49.18.10 EXPORTMC int MC\_SearchForService (MCAgency\_t attr, const char \* searchString, char \*\*\* agentNames, char \*\*\* serviceNames, int \*\* agentIDs, int \* numResults)

Search the directory facilitator for a service.

##### Returns

0 on success, error\_code\_t on failure

##### Parameters

**attr** (input) a MobileC agency handle

**searchString** (input) substring to search services for

**agentNames** (return) array of agent names with matching services

**serviceNames** (return) array of matching service names

**agentIDs** (return) array of matching agent IDs

**numResults** (return) number of matching results

#### 5.49.19 Example

Definition at line 2046 of file libmc.c.

References CHECK\_NULL, COND\_SLEEP\_ACTION, mc\_platform\_s::df, df\_AddRequest(), df\_request\_list\_node\_Destroy(), df\_request\_list\_node\_New(), df\_request\_search\_Destroy(), df\_request\_search\_New(), MC\_ERR\_MEMORY, agency\_s::mc\_platform, MC\_SUCCESS, and search.

Referenced by MC\_SearchForService\_chdl().

**5.49.19.1 EXPORTCH int MC\_SearchForService\_chdl (void \* *varg*)**

Definition at line 3898 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_SearchForService().

Referenced by agent\_ChScriptInitVar().

**5.49.19.2 EXPORTMC int MC\_SemaphorePost (MCAgency\_t *attr*, int *id*)**

Post to a MobileC synchronization variable semaphore.

**Parameters**

*attr* a MobileC agency handle

*id* the synchronization variable id to post to

**Returns**

0 on success, error\_code\_t type on failure

**5.49.20 Example**

Definition at line 2096 of file libmc.c.

References agency\_s::mc\_platform, syncListNode\_s::sem, SEMAPHORE\_POST, mc\_platform\_s::syncList, and syncListFind().

Referenced by MC\_SemaphorePost\_chdl().

**5.49.20.1 EXPORTCH int MC\_SemaphorePost\_chdl (void \* *varg*)**

Definition at line 3939 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_SemaphorePost().

Referenced by agent\_ChScriptInitVar().

**5.49.20.2 EXPORTMC int MC\_SemaphoreWait (MCAgency\_t *attr*, int *id*)**

Decreases a MobileC synchronization variable semaphore count by one.

**Parameters**

*attr* a MobileC agency handle

*id* synchronization variable id to wait on

**Returns**

0 on MC\_SUCCESS, error\_code\_t type of failure

**Note**

If the semaphore count is already zero, this function will block until another thread posts to the semaphore.



### 5.49.21 Example

Definition at line 2108 of file libmc.c.

References `agency_s::mc_platform`, `syncListNode_s::sem`, `SEMAPHORE_WAIT`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_SemaphoreWait_chdl()`.

#### 5.49.21.1 EXPORTCH int MC\_SemaphoreWait\_chdl (void \* *varg*)

Definition at line 3963 of file libmc.c.

References `CHECK_NULL`, `agency_s::mc_platform`, and `MC_SemaphoreWait()`.

Referenced by `agent_ChScriptInitVar()`.

#### 5.49.21.2 EXPORTMC int MC\_SendAgent (MCAgency\_t *attr*, const char \* *message*)

Sends an agent migration message.

##### Parameters

*attr* a MobileC agency handle

*message* a valid MobileC xml agent migration message

Definition at line 2130 of file libmc.c.

References `agent_Initialize()`, `mc_platform_s::agent_queue`, `agent_s::agent_status`, `mc_platform_s::ams`, `COND_BROADCAST`, `MC_ERR`, `agency_s::mc_platform`, `MC_WAIT_MESSGSEND`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

#### 5.49.21.3 EXPORTMC int MC\_SendAgentFile (MCAgency\_t *attr*, const char \* *filename*)

Sends an agent migration message.

##### Parameters

*attr* a MobileC agency handle

*filename* file containing a valid MobileC xml agent migration message

Definition at line 2178 of file libmc.c.

References `agent_Initialize()`, `mc_platform_s::agent_queue`, `agent_s::agent_status`, `mc_platform_s::ams`, `buf`, `COND_BROADCAST`, `agency_s::mc_platform`, `MC_WAIT_MESSGSEND`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

Referenced by `handler_SEND()`.

#### 5.49.21.4 EXPORTMC int MC\_SendAgentMigrationMessage (MCAgency\_t *attr*, const char \* *message*, const char \* *hostname*, int *port*)

Sends an agent migration message.

**Parameters**

*attr* a MobileC agency handle  
*message* a valid MobileC xml agent migration message  
*hostname* host to send the message to  
*port* port to send the message to

Definition at line 2244 of file libmc.c.

References MC\_ERR, agency\_s::mc\_platform, message\_Destroy(), message\_InitializeFromString(), message\_New(), and mc\_platform\_s::message\_queue.

Referenced by MC\_SendAgentMigrationMessage\_chdl().

**5.49.21.5 EXPORTCH int MC\_SendAgentMigrationMessage\_chdl (void \* *varg*)**

Definition at line 3987 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, MC\_SendAgentMigrationMessage(), and port.

Referenced by agent\_ChScriptInitVar().

**5.49.21.6 EXPORTMC int MC\_SendAgentMigrationMessageFile (MCAgency\_t *attr*, const char \* *filename*, const char \* *hostname*, int *port*)**

Sends an agent migration message.

**Parameters**

*attr* a MobileC agency handle  
*filename* file containing a valid MobileC xml agent migration message  
*hostname* hostname to send the agent to  
*port* port to send the agent to

Definition at line 2278 of file libmc.c.

References agent\_Initialize(), mc\_platform\_s::agent\_queue, agent\_s::agent\_status, mc\_platform\_s::ams, buf, COND\_BROADCAST, agency\_s::mc\_platform, MC\_WAIT\_MESSGSEND, message\_Destroy(), message\_InitializeFromString(), message\_New(), MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_SendAgentMigrationMessageFile\_chdl().

**5.49.21.7 EXPORTCH int MC\_SendAgentMigrationMessageFile\_chdl (void \* *varg*)**

Definition at line 4012 of file libmc.c.

References MC\_SendAgentMigrationMessageFile(), and port.

Referenced by agent\_ChScriptInitVar().

**5.49.21.8 EXPORTMC int MC\_SendCh (MCAgency\_t *attr*, const char \* *filename*, const char \* *remotehost*, int *port*)**

Definition at line 2120 of file libmc.c.

**5.49.21.9 EXPORTMC int MC\_SendSteerCommand (MCAgency\_t *attr*, enum MC\_SteerCommand\_e *cmd*)**

Definition at line 2348 of file libmc.c.

References COND\_BROADCAST, agency\_s::mc\_platform, mc\_platform\_s::MC\_steer\_command, mc\_platform\_s::MC\_steer\_cond, mc\_platform\_s::MC\_steer\_lock, MUTEX\_LOCK, and MUTEX\_UNLOCK.

Referenced by MC\_SendSteerCommand\_chdl().

**5.49.21.10 EXPORTCH int MC\_SendSteerCommand\_chdl (void \* *varg*)**

Definition at line 4032 of file libmc.c.

References CHECK\_NULL, agency\_s::mc\_platform, and MC\_SendSteerCommand().

Referenced by agent\_ChScriptInitVar().

**5.49.21.11 EXPORTMC int MC\_SetAgentStatus (MCAgent\_t *agent*, enum MC\_AgentStatus\_e *status*)**

Set an agent's status.

**Parameters**

*agent* a MobileC agent

*status* agent status of type 'enum MC\_AgentStatus\_e'

**Returns**

0 on success, or error\_code\_t on failure

Definition at line 2358 of file libmc.c.

References agent\_s::agent\_status, mc\_platform\_s::ams, COND\_SIGNAL, agent\_s::lock, agent\_s::mc\_platform, MUTEX\_LOCK, MUTEX\_UNLOCK, and agent\_s::orphan.

Referenced by MC\_SetAgentStatus\_chdl().

**5.49.21.12 EXPORTCH int MC\_SetAgentStatus\_chdl (void \* *varg*)**

Definition at line 4055 of file libmc.c.

References MC\_SetAgentStatus().

Referenced by agent\_ChScriptInitVar().

**5.49.21.13 EXPORTMC int MC\_SetDefaultAgentStatus (MCAgency\_t *agency*, enum MC\_AgentStatus\_e *status*)**

Sets default incoming agent status.

**Parameters**

*agency* a MobileC agency handle

*status* the status to set all incoming agents

### Returns

0 on success, `error_type_t` on failure

### Note

using this function will override any status the incoming agent attempts to set for itself.

Definition at line 2373 of file `libmc.c`.

References `mc_platform_s::default_agentstatus`, and `agency_s::mc_platform`.

Referenced by `MC_SetDefaultAgentStatus_chdl()`.

#### 5.49.21.14 EXPORTCH int MC\_SetDefaultAgentStatus\_chdl (void \* *varg*)

Definition at line 4073 of file `libmc.c`.

References `CHECK_NULL`, `int`, `agency_s::mc_platform`, and `MC_SetDefaultAgentStatus()`.

Referenced by `agent_ChScriptInitVar()`.

#### 5.49.21.15 EXPORTMC int MC\_SetThreadOff (MCAgencyOptions\_t \* *options*, enum MC\_ThreadIndex\_e *index*)

Sets a MobileC thread to "off" status.

### Parameters

*options* MobileC options previously initialized with [MC\\_InitializeAgencyOptions\(\)](#)

*index* the thread to set

### Returns

0 on success, `error_code_t` on failure

### Note

This function must be called before [MC\\_Initialize\(\)](#). Once an agency is started with `MC_Initialize`, the `MC_SetThread` functions will have no effect.

Definition at line 2400 of file `libmc.c`.

References `SET_THREAD_OFF`, and `MCAgencyOptions_s::threads`.

#### 5.49.21.16 EXPORTMC int MC\_SetThreadOn (MCAgencyOptions\_t \* *options*, enum MC\_ThreadIndex\_e *index*)

Sets a MobileC thread to "on" status.

### Parameters

*options* MobileC options previously initialized with [MC\\_InitializeAgencyOptions\(\)](#)

*index* the thread to set

**Returns**

0 on success, error\_code\_t on failure

**Note**

This function must be called before [MC\\_Initialize\(\)](#). Once an agency is started with MC\_Initialize, the MC\_SetThread functions will have no effect.

Definition at line 2383 of file libmc.c.

References SET\_THREAD\_ON, and MCAgencyOptions\_s::threads.

**5.49.21.17 EXPORTMC int MC\_SetThreadsAllOff (MCAgencyOptions\_t \* options)**

Set all MobileC threads to 'off' status.

**Parameters**

*options* a MobileC options structure initialized with with the [MC\\_InitializeAgencyOptions\(\)](#) function.

**Returns**

0 on success, error code on failure.

Definition at line 2407 of file libmc.c.

References MC\_THREAD\_ALL, SET\_THREAD\_OFF, and MCAgencyOptions\_s::threads.

**5.49.21.18 EXPORTMC int MC\_SetThreadsAllOn (MCAgencyOptions\_t \* options)**

Set all Mobile-C threads on.

**Parameters**

*options* MobileC options structure, initialized with [MC\\_InitializeAgencyOptions\(\)](#)

**Returns**

0 on success, error code on failure.

Definition at line 2390 of file libmc.c.

References MC\_THREAD\_ALL, SET\_THREAD\_ON, and MCAgencyOptions\_s::threads.

**5.49.21.19 EXPORTMC int MC\_Steer (MCAgency\_t attr, int(\*) (void \*data) funcptr, void \* arg)**

Set up a steerable algorithm.

**Parameters**

*attr* a MobileC agency handle

*funcptr* a function pointer to the algorithm

*arg* an argument for the algorithm function

**Returns**

0 on success, `error_code_t` on failure

**Note**

The algorithm function must contain a call to `MC_SteerControl` in order for the algorithm to be steerable.

**5.49.22 Example**

Definition at line 2417 of file `libmc.c`.

References `agency_s::mc_platform`, `MC_RESTART`, `MC_RUN`, `mc_platform_s::MC_steer_command`, `mc_platform_s::MC_steer_lock`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

**5.49.22.1 EXPORTMC enum MC\_SteerCommand\_e MC\_SteerControl (void)**

The MobileC user-algorithm steering function.

**Returns**

The current steering command

**Note**

This function belongs inside a user's steerable algorithm.

**See also**

[MC\\_Steer\(\)](#)

Definition at line 2436 of file `libmc.c`.

References `COND_WAIT`, `mc_platform_s::MC_steer_command`, `mc_platform_s::MC_steer_cond`, `mc_platform_s::MC_steer_lock`, `MC_SUSPEND`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

**5.49.22.2 EXPORTMC int MC\_SyncDelete (MCAgency\_t attr, int id)**

Deletes a previously initialized synchronization variable.

**Parameters**

*attr* a MobileC agency handle

*id* the sync variable id to delete

**Returns**

0 on success, or `error_code_t` on failure

Definition at line 2452 of file `libmc.c`.

References `syncList_s::giant_lock`, `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_platform_s::syncList`, `syncListFind()`, `syncListNodeDestroy()`, and `syncListRemove()`.

Referenced by `MC_SyncDelete_chdl()`.

**5.49.22.3 EXPORTCH int MC\_SyncDelete\_chdl (void \* *varg*)**

Definition at line 4096 of file libmc.c.

References `CHECK_NULL`, `agency_s::mc_platform`, and `MC_SyncDelete()`.

Referenced by `agent_ChScriptInitVar()`.

**5.49.22.4 EXPORTMC int MC\_SyncInit (MCAgency\_t *attr*, int *id*)**

Initializes a new MobileC synchronization variable.

**Parameters**

*attr* a MobileC agency handle

*id* the requested sync variable id

**Returns**

new sync variable's id. May be different than the requested id.

**Note**

Each synchronization variable may be used as a mutex, condition variable, or semaphore. However, it should only be used as one type of synchronization variable per instance, or undefined behaviour may result.

**See also**

[MC\\_MutexLock\(\)](#), [MC\\_MutexUnlock\(\)](#), [MC\\_CondWait\(\)](#), [MC\\_CondSignal\(\)](#), [MC\\_CondBroadcast](#), [MC\\_SemaphorePost\(\)](#), [MC\\_SemaphoreWait\(\)](#)

Definition at line 2482 of file libmc.c.

References `syncList_s::giant_lock`, `syncListNode_s::id`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `node`, `mc_platform_s::syncList`, `syncListAddNode()`, `syncListFind()`, and `syncListNodeNew()`.

Referenced by `MC_SyncInit_chdl()`.

**5.49.22.5 EXPORTCH int MC\_SyncInit\_chdl (void \* *varg*)**

Definition at line 4119 of file libmc.c.

References `CHECK_NULL`, `agency_s::mc_platform`, and `MC_SyncInit()`.

Referenced by `agent_ChScriptInitVar()`.

**5.49.22.6 EXPORTMC int MC\_TerminateAgent (MCAgent\_t *agent*)**

Halt a running agent.

**Returns**

0 on success, `error_code_t` on failure

Definition at line 2507 of file libmc.c.

References agent\_s::agent\_interp.

Referenced by ams\_ManageAgentList(), MC\_DeleteAgent(), and MC\_TerminateAgent\_chdl().

#### **5.49.22.7 EXPORTCH int MC\_TerminateAgent\_chdl (void \* *varg*)**

Definition at line 4143 of file libmc.c.

References MC\_ERR\_NOT\_FOUND, MC\_FindAgentByName(), and MC\_TerminateAgent().

Referenced by agent\_ChScriptInitVar().

#### **5.49.22.8 EXPORTMC int MC\_TerminateAgentWG (MCAgent\_t *calling\_agent*, MCAgent\_t *agent*)**

Definition at line 2517 of file libmc.c.

References agent\_s::agent\_interp, MC\_ERR\_INVALID\_ARGS, and agent\_s::wg\_code.

Referenced by MC\_DeleteAgentWG(), and MC\_TerminateAgentWG\_chdl().

#### **5.49.22.9 EXPORTCH int MC\_TerminateAgentWG\_chdl (void \* *varg*)**

Definition at line 4164 of file libmc.c.

References MC\_ERR\_NOT\_FOUND, MC\_FindAgentByName(), and MC\_TerminateAgentWG().

Referenced by agent\_ChScriptInitVar().

#### **5.49.22.10 EXPORTMC int MC\_WaitAgent (MCAgency\_t *attr*)**

Wait indefinitely.

##### **Note**

This function is intended to block the calling thread forever. Wait for an agent arrival event  
This function blocks until an agent arrival signal is triggered, at which point it unblocks.

Definition at line 2546 of file libmc.c.

References mc\_platform\_s::agent\_queue, COND\_WAIT, mc\_platform, agency\_s::mc\_platform, MUTEX\_LOCK, MUTEX\_UNLOCK, and size.

#### **5.49.22.11 EXPORTMC MCAgent\_t MC\_WaitRetrieveAgent (MCAgency\_t *attr*)**

Wait and retrieve an agent.

##### **Returns**

a valid MobileC agent on success, or NULL on failure

##### **Note**

This function blocks until the arrival of an agent. The agent is retrieved after it is initialized, but before it is executed.



Definition at line 2566 of file libmc.c.

References `mc_platform_s::agent_queue`, `ListSearch()`, `agency_s::mc_platform`, `MC_RECV_AGENT`, `MC_WaitSignal()`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

#### 5.49.22.12 EXPORTMC int MC\_WaitSignal (MCAgency\_t *attr*, int *signals*)

Wait for a MobileC signal.

##### Parameters

*attr* a MobileC agency handle

*signals* a flag of signals to wait for, of type 'enum MC\_Signal\_e'

##### Returns

0 on success, `error_code_t` on failure

##### Note

the parameter 'signals' may be something like '`MC_RECV_MESSAGE | MC_RECV_AGENT`', etc.

#### 5.49.23 Example

Definition at line 2584 of file libmc.c.

References `COND_WAIT`, `mc_platform_s::giant`, `mc_platform_s::giant_lock`, `agency_s::mc_platform`, `mc_platform_s::MC_signal`, `mc_platform_s::MC_signal_cond`, `mc_platform_s::MC_signal_lock`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

Referenced by `MC_WaitRetrieveAgent()`.

#### 5.49.24 Variable Documentation

##### 5.49.24.1 mc\_platform\_p g\_mc\_platform

Definition at line 74 of file libmc.c.

Referenced by `fipa_envelope_Compose__from()`.

## 5.50 /home/dko/Projects/mobilec/src/mc\_list/list.c File Reference

```
#include "list.h"
#include <stdio.h>
#include <stdlib.h>
```

### Functions

- [list\\_p ListInitialize](#) (void)
- void [ListTerminate](#) ([list\\_p](#) list)
- int [list\\_pGetSize](#) ([list\\_p](#) list)
- DATA [ListGetHead](#) ([list\\_p](#) list)
- DATA [ListPop](#) ([list\\_p](#) list)
- DATA [ListSearch](#) ([list\\_p](#) list, const int index)
- int [ListAdd](#) ([list\\_p](#) list, DATA data)
- int [ListInsert](#) ([list\\_p](#) list, DATA data, const int index)
- DATA [ListDelete](#) ([list\\_p](#) list, const int index)

### 5.50.1 Function Documentation

#### 5.50.1.1 int list\_pGetSize (list\_p list)

Definition at line 73 of file list.c.

References [list\\_s::size](#).

#### 5.50.1.2 int ListAdd (list\_p list, DATA data)

Definition at line 165 of file list.c.

References [list\\_s::listhead](#), [listNode\\_s::next](#), [listNode\\_s::node\\_data](#), and [list\\_s::size](#).

Referenced by [agent\\_task\\_Copy\(\)](#), [barrier\\_queue\\_Add\(\)](#), [df\\_Add\(\)](#), [df\\_AddRequest\(\)](#), and [syncListAddNode\(\)](#).

#### 5.50.1.3 DATA ListDelete (list\_p list, const int index)

Definition at line 220 of file list.c.

References [DATA](#), [list\\_s::listhead](#), [listNode\\_s::next](#), [listNode\\_s::node\\_data](#), and [list\\_s::size](#).

Referenced by [barrier\\_queue\\_Delete\(\)](#), [syncListDelete\(\)](#), [syncListDestroy\(\)](#), and [syncListRemove\(\)](#).

#### 5.50.1.4 DATA ListGetHead (list\_p list)

Definition at line 79 of file list.c.

References [list\\_s::listhead](#), and [listNode\\_s::node\\_data](#).

**5.50.1.5 list\_p ListInitialize (void)**

Definition at line 33 of file list.c.

References list\_s::listhead, and list\_s::size.

Referenced by barrier\_queue\_New(), df\_Initialize(), df\_request\_list\_New(), and syncListInit().

**5.50.1.6 int ListInsert (list\_p list, DATA data, const int index)**

Definition at line 207 of file list.c.

**5.50.1.7 DATA ListPop (list\_p list)**

Definition at line 89 of file list.c.

References DATA, list\_s::listhead, listNode\_s::next, listNode\_s::node\_data, and list\_s::size.

Referenced by AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), barrier\_queue\_Pop(), df\_Destroy(), df\_request\_list\_Destroy(), and df\_request\_list\_Pop().

**5.50.1.8 DATA ListSearch (list\_p list, const int index)**

Definition at line 124 of file list.c.

References list\_s::listhead, listNode\_s::next, listNode\_s::node\_data, and list\_s::size.

Referenced by agent\_queue\_Flush(), agent\_task\_Copy(), ams\_ManageAgentList(), ams\_Print(), AP\_QUEUE\_SEARCH\_TEMPLATE(), AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), barrier\_queue\_Delete(), MC\_RetrieveAgent(), MC\_WaitRetrieveAgent(), message\_queue\_SendOutgoing(), syncListDelete(), syncListDestroy(), and syncListRemove().

**5.50.1.9 void ListTerminate (list\_p list)**

Definition at line 49 of file list.c.

References list\_s::listhead, and list\_s::size.

Referenced by barrier\_queue\_Destroy(), df\_Destroy(), df\_request\_list\_Destroy(), and syncListDestroy().

## 5.51 /home/dko/Projects/mobilec/src/mc\_list/list.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
```

### Data Structures

- struct [listNode\\_s](#)
- struct [list\\_s](#)

### Defines

- #define [DATA](#) void\*
- #define [QUEUE\\_TEMPLATE](#)(name, node\_type, search\_type, search\_var\_name)

### Typedefs

- typedef struct [listNode\\_s](#) [listNode\\_t](#)
- typedef [listNode\\_t](#) \* [listNode\\_p](#)
- typedef struct [list\\_s](#) [list\\_t](#)
- typedef [list\\_t](#) \* [list\\_p](#)

### Functions

- [list\\_p](#) [ListInitialize](#) (void)
- void [ListTerminate](#) ([list\\_p](#) list)
- int [ListGetSize](#) ([list\\_p](#) list)
- int [ListAdd](#) ([list\\_p](#) list, DATA data)
- int [ListInsert](#) ([list\\_p](#) list, DATA data, const int index)
- DATA [ListGetHead](#) ([list\\_p](#) list)
- DATA [ListPop](#) ([list\\_p](#) list)
- DATA [ListSearch](#) ([list\\_p](#) list, const int index)
- DATA [ListDelete](#) ([list\\_p](#) list, const int index)

#### 5.51.1 Define Documentation

##### 5.51.1.1 #define DATA void\*

Definition at line 30 of file list.h.

Referenced by [barrier\\_queue\\_Add\(\)](#), [ListDelete\(\)](#), [ListPop\(\)](#), and [syncListAddNode\(\)](#).

##### 5.51.1.2 #define QUEUE\_TEMPLATE(name, node\_type, search\_type, search\_var\_name)

Definition at line 68 of file list.h.

## 5.51.2 Typedef Documentation

### 5.51.2.1 typedef list\_t\* list\_p

Definition at line 49 of file list.h.

### 5.51.2.2 typedef struct list\_s list\_t

### 5.51.2.3 typedef listNode\_t\* listNode\_p

Definition at line 39 of file list.h.

### 5.51.2.4 typedef struct listNode\_s listNode\_t

## 5.51.3 Function Documentation

### 5.51.3.1 int ListAdd (list\_p *list*, DATA *data*)

Definition at line 165 of file list.c.

References list\_s::listhead, listNode\_s::next, listNode\_s::node\_data, and list\_s::size.

Referenced by agent\_task\_Copy(), barrier\_queue\_Add(), df\_Add(), df\_AddRequest(), and syncListAddNode().

### 5.51.3.2 DATA ListDelete (list\_p *list*, const int *index*)

Definition at line 220 of file list.c.

References DATA, list\_s::listhead, listNode\_s::next, listNode\_s::node\_data, and list\_s::size.

Referenced by barrier\_queue\_Delete(), syncListDelete(), syncListDestroy(), and syncListRemove().

### 5.51.3.3 DATA ListGetHead (list\_p *list*)

Definition at line 79 of file list.c.

References list\_s::listhead, and listNode\_s::node\_data.

### 5.51.3.4 int ListGetSize (list\_p *list*)

### 5.51.3.5 list\_p ListInitialize (void)

Definition at line 33 of file list.c.

References list\_s::listhead, and list\_s::size.

Referenced by barrier\_queue\_New(), df\_Initialize(), df\_request\_list\_New(), and syncListInit().

### 5.51.3.6 int ListInsert (list\_p *list*, DATA *data*, const int *index*)

Definition at line 207 of file list.c.

### 5.51.3.7 DATA ListPop (*list\_p list*)

Definition at line 89 of file list.c.

References DATA, list\_s::listhead, listNode\_s::next, listNode\_s::node\_data, and list\_s::size.

Referenced by AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), barrier\_queue\_Pop(), df\_Destroy(), df\_request\_list\_Destroy(), and df\_request\_list\_Pop().

### 5.51.3.8 DATA ListSearch (*list\_p list*, *const int index*)

Definition at line 124 of file list.c.

References list\_s::listhead, listNode\_s::next, listNode\_s::node\_data, and list\_s::size.

Referenced by agent\_queue\_Flush(), agent\_task\_Copy(), ams\_ManageAgentList(), ams\_Print(), AP\_QUEUE\_SEARCH\_TEMPLATE(), AP\_QUEUE\_STD\_DEFN\_TEMPLATE(), barrier\_queue\_Delete(), MC\_RetrieveAgent(), MC\_WaitRetrieveAgent(), message\_queue\_SendOutgoing(), syncListDelete(), syncListDestroy(), and syncListRemove().

### 5.51.3.9 void ListTerminate (*list\_p list*)

Definition at line 49 of file list.c.

References list\_s::listhead, and list\_s::size.

Referenced by barrier\_queue\_Destroy(), df\_Destroy(), df\_request\_list\_Destroy(), and syncListDestroy().

## 5.52 /home/dko/Projects/mobilec/src/mc\_platform.c File Reference

```
#include "config.h"
#include <unistd.h>
#include <sys/socket.h>
#include <netdb.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include "include/acc.h"
#include "include/agent.h"
#include "include/mc_platform.h"
#include "include/macros.h"
```

### Defines

- `#define` [DEFAULT\\_HOSTNAME\\_LENGTH](#) 200

### Functions

- `mc_platform_p` `mc_platform_Initialize` (`MCAgency_t` agency, `ChOptions_t` \*ch\_options)
- `int` `mc_platform_Destroy` (`mc_platform_p` platform)

#### 5.52.1 Define Documentation

##### 5.52.1.1 `#define` `DEFAULT_HOSTNAME_LENGTH` 200

Definition at line 52 of file `mc_platform.c`.

Referenced by `mc_platform_Initialize()`.

#### 5.52.2 Function Documentation

##### 5.52.2.1 `int` `mc_platform_Destroy` (`mc_platform_p` platform)

Definition at line 249 of file `mc_platform.c`.

References `mc_platform_s::acc`, `acc_Destroy()`, `mc_platform_s::agent_queue`, `mc_platform_s::ams`, `ams_Destroy()`, `mc_platform_s::barrier_queue`, `barrier_queue_Destroy()`, `mc_platform_s::cmd_prompt`, `cmd_prompt_Destroy()`, `COND_DESTROY`, `mc_platform_s::connection_queue`, `mc_platform_s::df`, `df_Destroy()`, `mc_platform_s::giant_cond`, `mc_platform_s::giant_lock`, `mc_platform_s::hostname`, `mc_platform_s::interp_options`, `mc_platform_s::interpreter_queue`, `mc_platform_s::MC_signal_cond`, `mc_platform_s::MC_signal_lock`, `mc_platform_s::MC_steer_cond`, `mc_platform_s::MC_steer_lock`, `MC_SUCCESS`, `mc_platform_s::MC_sync_cond`, `mc_platform_s::MC_sync_lock`, `mc_platform_s::message_queue`, `MUTEX_DESTROY`, `mc_platform_s::quit_cond`, `mc_platform_s::quit_lock`, `SOCKET_ERROR`, `mc_platform_s::sockfd`, `mc_platform_s::syncList`, and `syncListDestroy()`.

Referenced by `MC_End()`.

### 5.52.2.2 mc\_platform\_p mc\_platform\_Initialize (MCAgency\_t agency, ChOptions\_t \* ch\_options)

Definition at line 54 of file mc\_platform.c.

References mc\_platform\_s::acc, acc\_Initialize(), acc\_Start(), mc\_platform\_s::agency, agent\_ChScriptInitVar(), mc\_platform\_s::agent\_queue, mc\_platform\_s::ams, ams\_Initialize(), ams\_Start(), mc\_platform\_s::barrier\_queue, barrier\_queue\_New(), agency\_s::bluetooth, mc\_platform\_s::bluetooth, CHECK\_NULL, mc\_platform\_s::cmd\_prompt, cmd\_prompt\_Initialize(), cmd\_prompt\_Start(), COND\_INIT, COND\_T, COND\_WAIT, mc\_platform\_s::connection\_queue, agency\_s::default\_agentstatus, mc\_platform\_s::default\_agentstatus, DEFAULT\_HOSTNAME\_LENGTH, mc\_platform\_s::df, df\_Initialize(), df\_Start(), mc\_platform\_s::err, GET\_THREAD\_MODE, mc\_platform\_s::giant, mc\_platform\_s::giant\_cond, mc\_platform\_s::giant\_lock, mc\_platform\_s::hostname, agency\_s::initInterps, mc\_platform\_s::initInterps, mc\_platform\_s::interp\_options, mc\_platform\_s::interpreter\_queue, agency\_s::last\_error, MC\_ERR\_MEMORY, MC\_NO\_SIGNAL, mc\_platform, mc\_platform\_s::MC\_signal, mc\_platform\_s::MC\_signal\_cond, mc\_platform\_s::MC\_signal\_lock, mc\_platform\_s::MC\_steer\_cond, mc\_platform\_s::MC\_steer\_lock, mc\_platform\_s::MC\_sync\_cond, mc\_platform\_s::MC\_sync\_lock, MC\_THREAD\_ACC, MC\_THREAD\_ALL, MC\_THREAD\_AMS, MC\_THREAD\_CP, MC\_THREAD\_DF, mc\_platform\_s::message\_queue, MUTEX\_INIT, MUTEX\_LOCK, MUTEX\_T, MUTEX\_UNLOCK, mc\_platform\_s::port, agency\_s::portno, mc\_platform\_s::quit, mc\_platform\_s::quit\_cond, mc\_platform\_s::quit\_lock, agency\_s::stack\_size, mc\_platform\_s::stack\_size, mc\_platform\_s::syncList, syncListInit(), and agency\_s::threads.

Referenced by MC\_Initialize().



## 5.53 /home/dko/Projects/mobilec/src/mc\_rwlock.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include "include/macros.h"
#include "include/mc_error.h"
#include "include/mc_rwlock.h"
```

### Functions

- [int mc\\_rwlock\\_init \(mc\\_rwlock\\_p rwlock\)](#)
- [int mc\\_rwlock\\_destroy \(mc\\_rwlock\\_p rwlock\)](#)
- [int mc\\_rwlock\\_rdlock \(mc\\_rwlock\\_p rwlock\)](#)
- [int mc\\_rwlock\\_rdunlock \(mc\\_rwlock\\_p rwlock\)](#)
- [int mc\\_rwlock\\_wrlock \(mc\\_rwlock\\_p rwlock\)](#)
- [int mc\\_rwlock\\_wrunlock \(mc\\_rwlock\\_p rwlock\)](#)

### 5.53.1 Function Documentation

#### 5.53.1.1 int mc\_rwlock\_destroy (mc\_rwlock\_p *rwlock*)

Definition at line 66 of file mc\_rwlock.c.

References `mc_rwlock_s::cond`, `COND_DESTROY`, `mc_rwlock_s::lock`, and `MUTEX_DESTROY`.

#### 5.53.1.2 int mc\_rwlock\_init (mc\_rwlock\_p *rwlock*)

Definition at line 46 of file mc\_rwlock.c.

References `CHECK_NULL`, `mc_rwlock_s::cond`, `COND_INIT`, `COND_T`, `mc_rwlock_s::lock`, `MC_ERR_MEMORY`, `MUTEX_INIT`, `MUTEX_T`, `mc_rwlock_s::num_readers`, `mc_rwlock_s::write_flag`, and `mc_rwlock_s::write_request`.

#### 5.53.1.3 int mc\_rwlock\_rdlock (mc\_rwlock\_p *rwlock*)

Definition at line 79 of file mc\_rwlock.c.

References `mc_rwlock_s::cond`, `COND_WAIT`, `mc_rwlock_s::lock`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_rwlock_s::num_readers`, `mc_rwlock_s::write_flag`, and `mc_rwlock_s::write_request`.

#### 5.53.1.4 int mc\_rwlock\_rdunlock (mc\_rwlock\_p *rwlock*)

Definition at line 94 of file mc\_rwlock.c.

References `mc_rwlock_s::cond`, `COND_SIGNAL`, `mc_rwlock_s::lock`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `mc_rwlock_s::num_readers`.

**5.53.1.5 int mc\_rwlock\_wrlock (mc\_rwlock\_p *rwlock*)**

Definition at line 107 of file mc\_rwlock.c.

References mc\_rwlock\_s::cond, COND\_WAIT, mc\_rwlock\_s::lock, MUTEX\_LOCK, MUTEX\_UNLOCK, mc\_rwlock\_s::num\_readers, mc\_rwlock\_s::write\_flag, and mc\_rwlock\_s::write\_request.

**5.53.1.6 int mc\_rwlock\_wrunlock (mc\_rwlock\_p *rwlock*)**

Definition at line 126 of file mc\_rwlock.c.

References mc\_rwlock\_s::cond, COND\_SIGNAL, mc\_rwlock\_s::lock, MUTEX\_LOCK, MUTEX\_UNLOCK, and mc\_rwlock\_s::write\_flag.

## 5.54 /home/dko/Projects/mobilec/src/mc\_sync/sync\_list.c File Reference

```
#include <pthread.h>
#include "sync_list.h"
#include "../mc_list/list.h"
#include "../include/mc_error.h"
```

### Functions

- `int syncListNodeInit` (struct `syncListNode_s` \*`node`)
- struct `syncListNode_s` \* `syncListNodeNew` (void)
- `int syncListNodeDestroy` (struct `syncListNode_s` \*`node`)
- `syncListNode_t` \* `syncListFind` (int `id`, struct `syncList_s` \*`list`)
- struct `syncList_s` \* `syncListInit` (void)
- `int syncListDestroy` (struct `syncList_s` \*`list`)
- `int syncListAddNode` (struct `syncListNode_s` \*`node`, struct `syncList_s` \*`list`)
- `int syncListNew` (int `id`, struct `syncList_s` \*`list`)
- `int syncListDelete` (int `id`, struct `syncList_s` \*`list`)
- `syncListNode_t` \* `syncListRemove` (int `id`, struct `syncList_s` \*`list`)

### 5.54.1 Function Documentation

#### 5.54.1.1 `int syncListAddNode` (struct `syncListNode_s` \* `node`, struct `syncList_s` \* `list`)

Definition at line 107 of file `sync_list.c`.

References `DATA`, `syncListNode_s::id`, `syncList_s::list`, `ListAdd()`, `list_s::listhead`, `syncList_s::lock`, `listNode_s::next`, `listNode_s::node_data`, `RWLOCK_WRLock`, and `RWLOCK_WRunlock`.

Referenced by `MC_SyncInit()`, and `syncListNew()`.

#### 5.54.1.2 `int syncListDelete` (int `id`, struct `syncList_s` \* `list`)

Definition at line 136 of file `sync_list.c`.

References `syncListNode_s::id`, `syncList_s::list`, `ListDelete()`, `ListSearch()`, `syncList_s::lock`, `MC_ERR_NOT_FOUND`, `RWLOCK_WRLock`, `RWLOCK_WRunlock`, `list_s::size`, and `syncListNodeDestroy()`.

#### 5.54.1.3 `int syncListDestroy` (struct `syncList_s` \* `list`)

Definition at line 86 of file `sync_list.c`.

References `syncList_s::giant_lock`, `syncList_s::list`, `ListDelete()`, `ListSearch()`, `ListTerminate()`, `syncList_s::lock`, `MUTEX_DESTROY`, `RWLOCK_DESTROY`, `RWLOCK_WRLock`, `RWLOCK_WRunlock`, `list_s::size`, and `syncListNodeDestroy()`.

Referenced by `mc_platform_Destroy()`.

#### 5.54.1.4 `syncListNode_t* syncListFind (int id, struct syncList_s * list)`

Definition at line 56 of file `sync_list.c`.

References `syncList_s::list`, `list_s::listhead`, `syncList_s::lock`, `listNode_s::next`, `listNode_s::node_data`, `RWLOCK_RDLOCK`, and `RWLOCK_RDUNLOCK`.

Referenced by `MC_CondBroadcast()`, `MC_CondReset()`, `MC_CondSignal()`, `MC_CondWait()`, `MC_MutexLock()`, `MC_MutexUnlock()`, `MC_SemaphorePost()`, `MC_SemaphoreWait()`, `MC_SyncDelete()`, and `MC_SyncInit()`.

#### 5.54.1.5 `struct syncList_s* syncListInit (void) [read]`

Definition at line 72 of file `sync_list.c`.

References `syncList_s::giant_lock`, `syncList_s::list`, `ListInitialize()`, `syncList_s::lock`, `MUTEX_INIT`, `MUTEX_T`, `RWLOCK_INIT`, and `RWLOCK_T`.

Referenced by `mc_platform_Initialize()`.

#### 5.54.1.6 `int syncListNew (int id, struct syncList_s * list)`

Definition at line 126 of file `sync_list.c`.

References `node`, `syncListAddNode()`, and `syncListNodeInit()`.

#### 5.54.1.7 `int syncListNodeDestroy (struct syncListNode_s * node)`

Definition at line 44 of file `sync_list.c`.

References `syncListNode_s::cond`, `COND_DESTROY`, `syncListNode_s::lock`, `MUTEX_DESTROY`, `syncListNode_s::sem`, and `SEMAPHORE_DESTROY`.

Referenced by `MC_SyncDelete()`, `syncListDelete()`, and `syncListDestroy()`.

#### 5.54.1.8 `int syncListNodeInit (struct syncListNode_s * node)`

Definition at line 11 of file `sync_list.c`.

References `CHECK_NULL`, `syncListNode_s::cond`, `COND_INIT`, `COND_T`, `syncListNode_s::lock`, `MUTEX_INIT`, `MUTEX_T`, `syncListNode_s::sem`, `SEMAPHORE_INIT`, and `SEMAPHORE_T`.

Referenced by `syncListNew()`.

#### 5.54.1.9 `struct syncListNode_s* syncListNodeNew (void) [read]`

Definition at line 26 of file `sync_list.c`.

References `CHECK_NULL`, `syncListNode_s::cond`, `COND_INIT`, `COND_T`, `syncListNode_s::lock`, `MUTEX_INIT`, `MUTEX_T`, `syncListNode_s::sem`, `SEMAPHORE_INIT`, `SEMAPHORE_T`, and `syncListNode_s::signalled`.

Referenced by `MC_SyncInit()`.

**5.54.1.10 syncListNode\_t\* syncListRemove (int *id*, struct syncList\_s \* *list*)**

Definition at line 153 of file sync\_list.c.

References syncListNode\_s::id, syncList\_s::list, ListDelete(), ListSearch(), syncList\_s::lock, RWLOCK\_WRLOCK, RWLOCK\_WRUNLOCK, and list\_s::size.

Referenced by MC\_SyncDelete().

## 5.55 /home/dko/Projects/mobilec/src/mc\_sync/sync\_list.h File Reference

```
#include "../include/macros.h"
#include "../mc_list/list.h"
#include "../include/mc_rwlock.h"
```

### Data Structures

- struct [syncListNode\\_s](#)
- struct [syncList\\_s](#)

### Typedefs

- typedef struct [syncListNode\\_s](#) [syncListNode\\_t](#)
- typedef [syncListNode\\_t](#) \* [syncListNode\\_p](#)
- typedef struct [syncList\\_s](#) [syncList\\_t](#)
- typedef [syncList\\_t](#) \* [syncList\\_p](#)

### Functions

- [int syncListNodeInit](#) (struct [syncListNode\\_s](#) \*node)
- [int syncListNodeDestroy](#) (struct [syncListNode\\_s](#) \*node)
- [syncListNode\\_t](#) \* [syncListFind](#) (int id, struct [syncList\\_s](#) \*list)
- struct [syncListNode\\_s](#) \* [syncListNodeNew](#) (void)
- [int syncListDelete](#) (int id, struct [syncList\\_s](#) \*list)
- [syncListNode\\_t](#) \* [syncListRemove](#) (int id, struct [syncList\\_s](#) \*list)
- struct [syncList\\_s](#) \* [syncListInit](#) (void)
- [int syncListDestroy](#) (struct [syncList\\_s](#) \*list)
- [int syncListAddNode](#) (struct [syncListNode\\_s](#) \*node, struct [syncList\\_s](#) \*list)
- [int syncListNew](#) (int id, struct [syncList\\_s](#) \*list)
- [syncListNode\\_t](#) \* [syncListGet](#) (int id, struct [syncList\\_s](#) \*list)

#### 5.55.1 Typedef Documentation

##### 5.55.1.1 typedef [syncList\\_t](#)\* [syncList\\_p](#)

Definition at line 33 of file [sync\\_list.h](#).

##### 5.55.1.2 typedef struct [syncList\\_s](#) [syncList\\_t](#)

##### 5.55.1.3 typedef [syncListNode\\_t](#)\* [syncListNode\\_p](#)

Definition at line 23 of file [sync\\_list.h](#).

#### 5.55.1.4 typedef struct syncListNode\_s syncListNode\_t

### 5.55.2 Function Documentation

#### 5.55.2.1 int syncListAddNode (struct syncListNode\_s \* *node*, struct syncList\_s \* *list*)

Definition at line 107 of file sync\_list.c.

References DATA, syncListNode\_s::id, syncList\_s::list, ListAdd(), list\_s::listhead, syncList\_s::lock, listNode\_s::next, listNode\_s::node\_data, RWLOCK\_WRLock, and RWLOCK\_WRunlock.

Referenced by MC\_SyncInit(), and syncListNew().

#### 5.55.2.2 int syncListDelete (int *id*, struct syncList\_s \* *list*)

Definition at line 136 of file sync\_list.c.

References syncListNode\_s::id, syncList\_s::list, ListDelete(), ListSearch(), syncList\_s::lock, MC\_ERR\_NOT\_FOUND, RWLOCK\_WRLock, RWLOCK\_WRunlock, list\_s::size, and syncListNodeDestroy().

#### 5.55.2.3 int syncListDestroy (struct syncList\_s \* *list*)

Definition at line 86 of file sync\_list.c.

References syncList\_s::giant\_lock, syncList\_s::list, ListDelete(), ListSearch(), ListTerminate(), syncList\_s::lock, MUTEX\_DESTROY, RWLOCK\_DESTROY, RWLOCK\_WRLock, RWLOCK\_WRunlock, list\_s::size, and syncListNodeDestroy().

Referenced by mc\_platform\_Destroy().

#### 5.55.2.4 syncListNode\_t\* syncListFind (int *id*, struct syncList\_s \* *list*)

Definition at line 56 of file sync\_list.c.

References syncList\_s::list, list\_s::listhead, syncList\_s::lock, listNode\_s::next, listNode\_s::node\_data, RWLOCK\_RDLOCK, and RWLOCK\_RDUnlock.

Referenced by MC\_CondBroadcast(), MC\_CondReset(), MC\_CondSignal(), MC\_CondWait(), MC\_MutexLock(), MC\_MutexUnlock(), MC\_SemaphorePost(), MC\_SemaphoreWait(), MC\_SyncDelete(), and MC\_SyncInit().

#### 5.55.2.5 syncListNode\_t\* syncListGet (int *id*, struct syncList\_s \* *list*)

#### 5.55.2.6 struct syncList\_s\* syncListInit (void) [read]

Definition at line 72 of file sync\_list.c.

References syncList\_s::giant\_lock, syncList\_s::list, ListInitialize(), syncList\_s::lock, MUTEX\_INIT, MUTEX\_T, RWLOCK\_INIT, and RWLOCK\_T.

Referenced by mc\_platform\_Initialize().

**5.55.2.7 int syncListNew (int *id*, struct syncList\_s \* *list*)**

Definition at line 126 of file sync\_list.c.

References node, syncListAddNode(), and syncListNodeInit().

**5.55.2.8 int syncListNodeDestroy (struct syncListNode\_s \* *node*)**

Definition at line 44 of file sync\_list.c.

References syncListNode\_s::cond, COND\_DESTROY, syncListNode\_s::lock, MUTEX\_DESTROY, syncListNode\_s::sem, and SEMAPHORE\_DESTROY.

Referenced by MC\_SyncDelete(), syncListDelete(), and syncListDestroy().

**5.55.2.9 int syncListNodeInit (struct syncListNode\_s \* *node*)**

Definition at line 11 of file sync\_list.c.

References CHECK\_NULL, syncListNode\_s::cond, COND\_INIT, COND\_T, syncListNode\_s::lock, MUTEX\_INIT, MUTEX\_T, syncListNode\_s::sem, SEMAPHORE\_INIT, and SEMAPHORE\_T.

Referenced by syncListNew().

**5.55.2.10 struct syncListNode\_s\* syncListNodeNew (void) [read]**

Definition at line 26 of file sync\_list.c.

References CHECK\_NULL, syncListNode\_s::cond, COND\_INIT, COND\_T, syncListNode\_s::lock, MUTEX\_INIT, MUTEX\_T, syncListNode\_s::sem, SEMAPHORE\_INIT, SEMAPHORE\_T, and syncListNode\_s::signalled.

Referenced by MC\_SyncInit().

**5.55.2.11 syncListNode\_t\* syncListRemove (int *id*, struct syncList\_s \* *list*)**

Definition at line 153 of file sync\_list.c.

References syncListNode\_s::id, syncList\_s::list, ListDelete(), ListSearch(), syncList\_s::lock, RWLOCK\_WRLOCK, RWLOCK\_WRUNLOCK, and list\_s::size.

Referenced by MC\_SyncDelete().



## 5.56 /home/dko/Projects/mobilec/src/message.c File Reference

```
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <unistd.h>
#include <netdb.h>
#include <errno.h>
#include "config.h"
#include <mxml.h>
#include "include/libmc.h"
#include "include/agent.h"
#include "include/mc_platform.h"
#include "include/message.h"
#include "include/mtp_http.h"
#include "include/xml_compose.h"
#include "include/xml_parser.h"
#include "include/macros.h"
#include "../include/macros.h"
#include "../include/message.h"
#include "xyssl-0.9/include/xyssl/dhm.h"
#include "xyssl-0.9/include/xyssl/rsa.h"
#include "xyssl-0.9/include/xyssl/aes.h"
```

### Defines

- `#define SOCKET_INPUT_SIZE 4096`
- `#define MSG_THREADS 40`
- `#define MSG_THREAD_EXIT()`

### Functions

- `message_p message_New` (void)
- `message_p message_Copy` (message\_p src)
- `int message_InitializeFromAgent` (mc\_platform\_p mc\_platform, message\_p message, agent\_p agent)
- `int message_InitializeFromConnection` (mc\_platform\_p mc\_platform, message\_p message, connection\_p connection)
- `int http_to_hostport` (const char \*http\_str, char \*\*host, int \*port, char \*\*target)

- `int message_InitializeFromString (mc_platform_p mc_platform, message_p message, const char *string, const char *destination_host, int destination_port, const char *target)`
- `int message_Destroy (message_p message)`
- `int auth_rece_send_msg (int sockfd, char *hostname, char *message, char *privkey, char *known_host_filename)`
- `int message_Send (mc_platform_t *mc_platform, message_p message, char *privatekey)`
- `void * message_send_Thread (void *arg)`

## 5.56.1 Define Documentation

### 5.56.1.1 #define MSG\_THREAD\_EXIT()

#### Value:

```
free(arg); \
message_Destroy(message); \
MUTEX_LOCK(&mc_platform->acc->msg_thread_lock); \
mc_platform->acc->num_msg_threads--; \
COND_SIGNAL(&mc_platform->acc->msg_thread_cond); \
MUTEX_UNLOCK(&mc_platform->acc->msg_thread_lock); \
THREAD_EXIT()
```

Definition at line 637 of file message.c.

Referenced by message\_send\_Thread().

### 5.56.1.2 #define MSG\_THREADS 40

Definition at line 574 of file message.c.

Referenced by message\_Send().

### 5.56.1.3 #define SOCKET\_INPUT\_SIZE 4096

Definition at line 70 of file message.c.

## 5.56.2 Function Documentation

### 5.56.2.1 int auth\_rece\_send\_msg (int sockfd, char \* hostname, char \* message, char \* privkey, char \* known\_host\_filename)

Definition at line 453 of file message.c.

References aes\_en\_de(), initiate\_migration\_process(), read\_known\_host\_file(), and send\_AES\_en\_MA().

Referenced by message\_send\_Thread().

### 5.56.2.2 int http\_to\_hostport (const char \* http\_str, char \*\* host, int \* port, char \*\* target)

Definition at line 298 of file message.c.

References MC\_ERR\_PARSE.

Referenced by MC\_AclSend().

### 5.56.2.3 message\_p message\_Copy (message\_p src)

Definition at line 97 of file message.c.

### 5.56.2.4 int message\_Destroy (message\_p message)

Definition at line 414 of file message.c.

References message\_s::addr, message\_s::agent\_xml\_flag, message\_s::from\_address, MC\_SUCCESS, message\_s::message\_body, mxmlDelete(), message\_s::target, message\_s::to\_address, message\_s::update\_name, and message\_s::xml\_root.

Referenced by acc\_connection\_Thread(), acc\_MessageHandlerThread(), ams\_ManageAgentList(), MC\_LoadAgentFromFile(), MC\_SendAgent(), MC\_SendAgentFile(), MC\_SendAgentMigrationMessage(), MC\_SendAgentMigrationMessageFile(), and message\_InitializeFromConnection().

### 5.56.2.5 int message\_InitializeFromAgent (mc\_platform\_p mc\_platform, message\_p message, agent\_p agent)

Definition at line 105 of file message.c.

References message\_s::addr, agent\_xml\_compose(), message\_s::agent\_xml\_flag, mc\_platform\_s::bluetooth, buf, CHECK\_NULL, agent\_s::datastate, mc\_platform\_s::err, message\_s::from\_address, agent\_s::home, mc\_platform\_s::hostname, MC\_ERR\_MEMORY, MC\_SUCCESS, message\_s::message\_body, message\_s::message\_id, message\_s::message\_type, MOBILE\_AGENT, MXML\_NO\_CALLBACK, mxmlSaveAllocString(), agent\_s::name, agent\_datastate\_s::number\_of\_tasks, mc\_platform\_s::port, message\_s::sending\_agent\_name, agent\_task\_s::server\_name, strdup, strtok\_r, message\_s::target, agent\_datastate\_s::task\_progress, agent\_datastate\_s::tasks, message\_s::to\_address, message\_s::update\_name, WARN, and message\_s::xml\_root.

### 5.56.2.6 int message\_InitializeFromConnection (mc\_platform\_p mc\_platform, message\_p message, connection\_p connection)

Definition at line 207 of file message.c.

References connection\_s::addr, message\_s::addr, CHECK\_NULL, connection\_s::clientfd, connection\_s::connect\_id, message\_s::connect\_id, message\_s::from\_address, MC\_ERR\_CONNECT, MC\_ERR\_PARSE, MC\_SUCCESS, message\_s::message\_body, message\_Destroy(), message\_s::message\_id, message\_xml\_parse(), MXML\_NO\_CALLBACK, mxmlLoadString(), SOCKET\_ERROR, SOCKET\_INPUT\_SIZE, message\_s::target, message\_s::to\_address, and message\_s::xml\_root.

### 5.56.2.7 int message\_InitializeFromString (mc\_platform\_p mc\_platform, message\_p message, const char \* string, const char \* destination\_host, int destination\_port, const char \* target)

Definition at line 333 of file message.c.

References message\_s::addr, CHECK\_NULL, message\_s::connect\_id, mc\_platform\_s::err, message\_s::from\_address, mc\_platform\_s::hostname, MC\_ERR\_MEMORY, MC\_ERR\_PARSE, MC\_SUCCESS, message\_s::message\_body, message\_s::message\_id, message\_s::message\_type, message\_xml\_parse(), MOBILE\_AGENT, MXML\_NO\_CALLBACK, mxmlLoadString(), mc\_platform\_s::port, message\_s::target, message\_s::to\_address, message\_s::update\_name, and message\_s::xml\_root.

### 5.56.2.8 `message_p message_New (void)`

Definition at line 73 of file `message.c`.

References `message_s::addr`, `message_s::agent_xml_flag`, `CHECK_NULL`, `message_s::connect_id`, `message_s::from_address`, `message_s::http_type`, `message_s::isHTTP`, `message_s::message_body`, `message_s::message_id`, `message_s::message_type`, `message_s::target`, `message_s::to_address`, `message_s::update_name`, `message_s::update_num`, `message_s::xml_payload`, and `message_s::xml_root`.

Referenced by `acc_connection_Thread()`, `ams_ManageAgentList()`, `MC_LoadAgentFromFile()`, `MC_SendAgent()`, `MC_SendAgentFile()`, `MC_SendAgentMigrationMessage()`, `MC_SendAgentMigrationMessageFile()`, and `mtp_http_CreateMessage()`.

### 5.56.2.9 `int message_Send (mc_platform_t * mc_platform, message_p message, char * privatekey)`

Definition at line 575 of file `message.c`.

References `mc_platform_s::acc`, `COND_WAIT`, `message_send_arg_s::mc_platform`, `message_send_arg_s::message`, `message_send_Thread()`, `MSG_THREADS`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `message_send_arg_s::privatekey`, `THREAD_CREATE`, and `THREAD_DETACH`.

Referenced by `acc_MessageHandlerThread()`, `MC_AclSend()`, and `message_queue_SendOutgoing()`.

### 5.56.2.10 `void* message_send_Thread (void * arg)`

Definition at line 648 of file `message.c`.

References `mc_platform_s::agency`, `auth_rece_send_msg()`, `mc_platform_s::bluetooth`, `buf`, `CHECK_NULL`, `dynstring_Append()`, `dynstring_Destroy()`, `dynstring_New()`, `agency_s::known_host_filename`, `mc_platform`, `dynstring_s::message`, `message_s::message_body`, `MSG_THREAD_EXIT`, `mtp_http_ComposeMessage()`, `mtp_http_Destroy()`, `mtp_http_New()`, `mtp_http_Parse()`, `port`, `send`, `SOCKET_ERROR`, `SOCKET_INPUT_SIZE`, `strtok_r`, and `message_s::to_address`.

Referenced by `message_Send()`.

## 5.57 /home/dko/Projects/mobilec/src/message\_queue.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include "include/data_structures.h"
#include "include/mc_platform.h"
```

### Functions

- void [message\\_queue\\_SendOutgoing](#) (struct [mc\\_platform\\_s](#) \*[mc\\_platform](#), [message\\_queue\\_p](#) [mqueue](#))

#### 5.57.1 Function Documentation

##### 5.57.1.1 void [message\\_queue\\_SendOutgoing](#) (struct [mc\\_platform\\_s](#) \* [mc\\_platform](#), [message\\_queue\\_p](#) [mqueue](#))

Definition at line 57 of file [message\\_queue.c](#).

References [CHECK\\_NULL](#), [mc\\_platform\\_s::hostname](#), [ListSearch\(\)](#), [message\\_Send\(\)](#), [MUTEX\\_LOCK](#), [MUTEX\\_UNLOCK](#), [mc\\_platform\\_s::port](#), and [message\\_s::to\\_address](#).

## 5.58 /home/dko/Projects/mobilec/src/mtp\_http.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <unistd.h>
#include <errno.h>
#include "config.h"
#include "include/connection.h"
#include "include/mtp_http.h"
#include "include/macros.h"
#include "include/mc_error.h"
#include "include/message.h"
#include "include/dynstring.h"
```

### Defines

- #define [SAFE\\_FREE](#)(elem)
- #define [HTTP\\_PARSE\\_EXPR](#)(parse\_name, struct\_name)
- #define [SAFE\\_FREE](#)(object)

### Functions

- [int mtp\\_http\\_Destroy](#) ([mtp\\_http\\_p](#) http)
- [mtp\\_http\\_p mtp\\_http\\_New](#) (void)
- [int rece\\_de\\_msg](#) (char \*buffer, [connection\\_p](#) con, char \*privatekey)
- [int mtp\\_http\\_InitializeFromConnection](#) ([mtp\\_http\\_p](#) http, [connection\\_p](#) connection, char \*privatekey)
- const char \* [http\\_GetExpression](#) (const char \*string, char \*\*expr)

*Parse an html expression.*

- [int http\\_ParseExpression](#) (const char \*expression\_string, char \*\*name, char \*\*value)

*Parse an expression into its name and value.*

- const char \* [mtp\\_http\\_ParseHeader](#) (struct [mtp\\_http\\_s](#) \*http, const char \*string)
- [int mtp\\_http\\_Parse](#) (struct [mtp\\_http\\_s](#) \*http, const char \*string)
- const char \* [http\\_ParseRequest](#) ([mtp\\_http\\_p](#) http, const char \*string)
- const char \* [http\\_GetToken](#) (const char \*string, char \*\*token)
- [int mtp\\_http\\_ParseResponse](#) (struct [mtp\\_http\\_s](#) \*http, const char \*string)
- [int mtp\\_http\\_ComposeMessage](#) ([message\\_p](#) message)
- struct [message\\_s](#) \* [mtp\\_http\\_CreateMessage](#) ([mtp\\_http\\_t](#) \*mtp\_http, char \*hostname, [int](#) port)

## 5.58.1 Define Documentation

### 5.58.1.1 #define HTTP\_PARSE\_EXPR(parse\_name, struct\_name)

**Value:**

```
if ( !strcmp(name, parse_name) ) { \
    http->struct_name = (char*)malloc \
    ( \
        sizeof(char) * \
        (strlen(value)+1) \
    ); \
    strcpy(http->struct_name, value); \
} else
```

Referenced by mtp\_http\_ParseHeader().

### 5.58.1.2 #define SAFE\_FREE(object)

**Value:**

```
if(object) free(object); \
    object = NULL
```

### 5.58.1.3 #define SAFE\_FREE(elem)

**Value:**

```
if(elem) \
    free(elem)
```

Referenced by mtp\_http\_Destroy(), and mtp\_http\_ParseHeader().

## 5.58.2 Function Documentation

### 5.58.2.1 const char\* http\_GetExpression (const char \* *string*, char \*\* *expr*)

Parse an html expression.

**Parameters**

*string* (input) The html block of text: Will parse the first expression pointed to by 'string'.

*expr* (output) The allocated expression

**Returns**

A pointer to the next expression segment of the string block, or NULL.

Definition at line 349 of file mtp\_http.c.

Referenced by mtp\_http\_Parse(), and mtp\_http\_ParseHeader().

**5.58.2.2 const char\* http\_GetToken (const char \* *string*, char \*\* *token*)**

Definition at line 790 of file mtp\_http.c.

References cur.

Referenced by http\_ParseRequest().

**5.58.2.3 int http\_ParseExpression (const char \* *expression\_string*, char \*\* *name*, char \*\* *value*)**

Parse an expression into its name and value.

**Parameters**

*expression\_string* (input) The expression

*name* (output) An allocated name string or NULL

*value* (output) An allocated value string or NULL

**Returns**

error\_code\_t type

**Note**

an http expression is something like 'Date: Mon, 23 May 2005 22:38:34 GMT  
' where 'Date' is the name and the remainder of the string is the value

Definition at line 408 of file mtp\_http.c.

References CHECK\_NULL, MC\_ERR\_PARSE, and MC\_SUCCESS.

Referenced by mtp\_http\_Parse(), and mtp\_http\_ParseHeader().

**5.58.2.4 const char\* http\_ParseRequest (mtp\_http\_p *http*, const char \* *string*)**

Definition at line 701 of file mtp\_http.c.

References cur, HTTP\_CONNECT, HTTP\_DELETE, HTTP\_GET, http\_GetToken(), HTTP\_HEAD, HTTP\_OPTIONS, mtp\_http\_s::http\_performative, HTTP\_PERFORMATIVE\_UNDEF, HTTP\_POST, HTTP\_PUT, HTTP\_RESPONSE, HTTP\_TRACE, mtp\_http\_s::response\_code, mtp\_http\_s::response\_string, and mtp\_http\_s::target.

Referenced by mtp\_http\_ParseHeader().

**5.58.2.5 int mtp\_http\_ComposeMessage (message\_p *message*)**

Definition at line 829 of file mtp\_http.c.

References buf, message\_s::isHTTP, MC\_SUCCESS, message\_s::message\_body, PACKAGE\_VERSION, message\_s::target, and message\_s::to\_address.

**5.58.2.6 struct message\_s\* mtp\_http\_CreateMessage (mtp\_http\_t \* *mtp\_http*, char \* *hostname*, int *port*) [read]**

Definition at line 875 of file mtp\_http.c.



References `buf`, `mtp_http_s::content`, `mtp_http_content_s::content_type`, `mtp_http_content_s::data`, `dynstring_Append()`, `dynstring_Destroy()`, `dynstring_New()`, `mtp_http_s::host`, `message_s::isHTTP`, `dynstring_s::len`, `dynstring_s::message`, `message_s::message_body`, `message_New()`, `mtp_http_s::message_parts`, `PACKAGE_VERSION`, `mtp_http_s::target`, and `message_s::to_address`.

Referenced by `MC_AclSend()`.

#### 5.58.2.7 `int mtp_http_Destroy (mtp_http_p http)`

Definition at line 54 of file `mtp_http.c`.

References `mtp_http_s::accept_ranges`, `mtp_http_s::boundary`, `mtp_http_s::cache_control`, `mtp_http_s::connection`, `mtp_http_s::content`, `mtp_http_s::content_length`, `mtp_http_content_s::content_type`, `mtp_http_s::content_type`, `mtp_http_content_s::data`, `mtp_http_s::date`, `mtp_http_s::host`, `mtp_http_s::http_version`, `mtp_http_s::message_parts`, `mtp_http_s::mime_version`, `mtp_http_s::response_string`, `mtp_http_s::return_code`, `SAFE_FREE`, `mtp_http_s::server`, `mtp_http_s::target`, and `mtp_http_s::user_agent`.

Referenced by `acc_connection_Thread()`, `MC_AclSend()`, `message_send_Thread()`, and `mtp_http_InitializeFromConnection()`.

#### 5.58.2.8 `int mtp_http_InitializeFromConnection (mtp_http_p http, connection_p connection, char * privatekey)`

Definition at line 180 of file `mtp_http.c`.

References `CHECK_NULL`, `connection_s::clientfd`, `mtp_http_s::content_length`, `dynstring_Append()`, `dynstring_Destroy()`, `dynstring_New()`, `ERR`, `mtp_http_s::header_length`, `HTTP_HEAD`, `mtp_http_s::http_performative`, `MC_ERR_CONNECT`, `mtp_http_Destroy()`, `mtp_http_New()`, `mtp_http_Parse()`, `mtp_http_ParseHeader()`, `PACKAGE_STRING`, `rece_de_msg()`, `send`, `SOCKET_ERROR`, `SOCKET_INPUT_SIZE`, and `strdup`.

#### 5.58.2.9 `mtp_http_p mtp_http_New (void)`

Definition at line 89 of file `mtp_http.c`.

References `CHECK_NULL`, and `mtp_http_s::content`.

Referenced by `acc_connection_Thread()`, `MC_AclSend()`, `message_send_Thread()`, and `mtp_http_InitializeFromConnection()`.

#### 5.58.2.10 `int mtp_http_Parse (struct mtp_http_s * http, const char * string)`

Definition at line 551 of file `mtp_http.c`.

References `mtp_http_s::boundary`, `mtp_http_s::content`, `mtp_http_s::content_length`, `mtp_http_content_s::content_type`, `mtp_http_s::content_type`, `mtp_http_content_s::data`, `http_GetExpression()`, `HTTP_HEAD`, `http_ParseExpression()`, `mtp_http_s::http_performative`, `HTTP_POST`, `HTTP_PUT`, `HTTP_RESPONSE`, `MC_SUCCESS`, `mtp_http_s::message_parts`, `mtp_http_ParseHeader()`, and `strdup`.

Referenced by `message_send_Thread()`, and `mtp_http_InitializeFromConnection()`.

#### 5.58.2.11 `const char* mtp_http_ParseHeader (struct mtp_http_s * http, const char * string)`

Definition at line 467 of file `mtp_http.c`.

References mtp\_http\_s::header\_length, http\_GetExpression(), HTTP\_PARSE\_EXPR, http\_ParseExpression(), http\_ParseRequest(), MC\_SUCCESS, and SAFE\_FREE.

Referenced by mtp\_http\_InitializeFromConnection(), and mtp\_http\_Parse().

#### **5.58.2.12 int mtp\_http\_ParseResponse (struct mtp\_http\_s \* *http*, const char \* *string*)**

Definition at line 823 of file mtp\_http.c.

#### **5.58.2.13 int rece\_de\_msg (char \* *buffer*, connection\_p *con*, char \* *privatekey*)**

Definition at line 100 of file mtp\_http.c.

References aes\_en\_de(), connection\_s::AES\_key, connection\_s::clientfd, int, connection\_s::nonce, and receive\_AES\_en\_MA().

Referenced by mtp\_http\_InitializeFromConnection().

## 5.59 /home/dko/Projects/mobilec/src/mxml-2.2.2/config.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdarg.h>
#include <ctype.h>
```

### Defines

- `#define MXML_VERSION "Mini-XML v2.2.2"`
- `#define HAVE_VSNPRINTF 1`
- `#define HAVE_STRDUP 1`

### Functions

- `char * mxml_strdup (const char *, va_list)`

#### 5.59.1 Define Documentation

##### 5.59.1.1 `#define HAVE_STRDUP 1`

Definition at line 49 of file config.h.

##### 5.59.1.2 `#define HAVE_VSNPRINTF 1`

Definition at line 42 of file config.h.

##### 5.59.1.3 `#define MXML_VERSION "Mini-XML v2.2.2"`

Definition at line 35 of file config.h.

Referenced by `write_documentation()`.

#### 5.59.2 Function Documentation

##### 5.59.2.1 `char* mxml_strdup (const char *, va_list)`

Definition at line 66 of file mxml-string.c.

Referenced by `mxml_error()`, `mxmlNewTextf()`, and `mxmlSetTextf()`.

## 5.60 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/config.h File Reference

### Defines

- `#define XYSSL_HAVE_ASM`
- `#define XYSSL_DEBUG_MSG`
- `#define XYSSL_SELF_TEST`
- `#define XYSSL_GENPRIME`
- `#define XYSSL_AES_C`
- `#define XYSSL_ARC4_C`
- `#define XYSSL_BASE64_C`
- `#define XYSSL_BIGNUM_C`
- `#define XYSSL_CERTS_C`
- `#define XYSSL_DEBUG_C`
- `#define XYSSL_DES_C`
- `#define XYSSL_DHM_C`
- `#define XYSSL_HAVEGE_C`
- `#define XYSSL_MD5_C`
- `#define XYSSL_NET_C`
- `#define XYSSL_PADLOCK_C`
- `#define XYSSL_RSA_C`
- `#define XYSSL_SHA1_C`
- `#define XYSSL_SHA2_C`
- `#define XYSSL_SHA4_C`
- `#define XYSSL_SSL_CLI_C`
- `#define XYSSL_SSL_SRV_C`
- `#define XYSSL_SSL_TLS_C`
- `#define XYSSL_TIMING_C`
- `#define XYSSL_X509_PARSE_C`
- `#define XYSSL_X509_WRITE_C`

### 5.60.1 Define Documentation

#### 5.60.1.1 `#define XYSSL_AES_C`

Definition at line 74 of file config.h.

#### 5.60.1.2 `#define XYSSL_ARC4_C`

Definition at line 84 of file config.h.

#### 5.60.1.3 `#define XYSSL_BASE64_C`

Definition at line 92 of file config.h.

#### 5.60.1.4 `#define XYSSL_BIGNUM_C`

Definition at line 103 of file config.h.

**5.60.1.5 #define XYSSL\_CERTS\_C**

Definition at line 111 of file config.h.

**5.60.1.6 #define XYSSL\_DEBUG\_C**

Definition at line 121 of file config.h.

**5.60.1.7 #define XYSSL\_DEBUG\_MSG**

Definition at line 47 of file config.h.

**5.60.1.8 #define XYSSL\_DES\_C**

Definition at line 131 of file config.h.

**5.60.1.9 #define XYSSL\_DHM\_C**

Definition at line 142 of file config.h.

**5.60.1.10 #define XYSSL\_GENPRIME**

Definition at line 57 of file config.h.

**5.60.1.11 #define XYSSL\_HAVE\_ASM**

Definition at line 36 of file config.h.

**5.60.1.12 #define XYSSL\_HAVEGE\_C**

Definition at line 150 of file config.h.

**5.60.1.13 #define XYSSL\_MD5\_C**

Definition at line 177 of file config.h.

**5.60.1.14 #define XYSSL\_NET\_C**

Definition at line 185 of file config.h.

**5.60.1.15 #define XYSSL\_PADLOCK\_C**

Definition at line 193 of file config.h.

**5.60.1.16 #define XYSSL\_RSA\_C**

Definition at line 204 of file config.h.

**5.60.1.17 #define XYSSL\_SELF\_TEST**

Definition at line 52 of file config.h.

**5.60.1.18 #define XYSSL\_SHA1\_C**

Definition at line 215 of file config.h.

**5.60.1.19 #define XYSSL\_SHA2\_C**

Definition at line 223 of file config.h.

**5.60.1.20 #define XYSSL\_SHA4\_C**

Definition at line 231 of file config.h.

**5.60.1.21 #define XYSSL\_SSL\_CLI\_C**

Definition at line 239 of file config.h.

**5.60.1.22 #define XYSSL\_SSL\_SRV\_C**

Definition at line 247 of file config.h.

**5.60.1.23 #define XYSSL\_SSL\_TLS\_C**

Definition at line 256 of file config.h.

**5.60.1.24 #define XYSSL\_TIMING\_C**

Definition at line 264 of file config.h.

**5.60.1.25 #define XYSSL\_X509\_PARSE\_C**

Definition at line 274 of file config.h.

**5.60.1.26 #define XYSSL\_X509\_WRITE\_C**

Definition at line 282 of file config.h.

## 5.61 /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-attr.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdarg.h>
#include <ctype.h>
#include <errno.h>
```

### Functions

- const char \* [mxmlElementGetAttr](#) (mxml\_node\_t \**node*, const char \**name*)
- void [mxmlElementSetAttr](#) (mxml\_node\_t \**node*, const char \**name*, const char \**value*)

#### 5.61.1 Function Documentation

##### 5.61.1.1 const char\* mxmlElementGetAttr (mxml\_node\_t \* *node*, const char \* *name*)

Definition at line 48 of file mxml-attr.c.

References [mxml\\_value\\_s::attrs](#), [mxml\\_value\\_u::element](#), [MXML\\_ELEMENT](#), [mxml\\_attr\\_s::name](#), [mxml\\_value\\_s::num\\_attrs](#), [mxml\\_node\\_s::type](#), [mxml\\_attr\\_s::value](#), and [mxml\\_node\\_s::value](#).

Referenced by [agent\\_xml\\_parse\\_\\_agent\\_code\(\)](#), [agent\\_xml\\_parse\\_\\_data\(\)](#), [agent\\_xml\\_parse\\_\\_task\(\)](#), [agent\\_xml\\_parse\\_\\_tasks\(\)](#), [fipa\\_envelope\\_HandleReceived\(\)](#), [index\\_compare\(\)](#), [index\\_find\(\)](#), [message\\_xml\\_parse\\_\\_message\(\)](#), [mxml\\_parse\\_element\(\)](#), [mxmlFindElement\(\)](#), [mxmlIndexNew\(\)](#), [scan\\_file\(\)](#), [sort\\_node\(\)](#), [type\\_cb\(\)](#), and [write\\_documentation\(\)](#).

##### 5.61.1.2 void mxmlElementSetAttr (mxml\_node\_t \* *node*, const char \* *name*, const char \* *value*)

Definition at line 95 of file mxml-attr.c.

References [mxml\\_value\\_s::attrs](#), [mxml\\_value\\_u::element](#), [MXML\\_ELEMENT](#), [mxml\\_error\(\)](#), [mxml\\_value\\_s::name](#), [mxml\\_attr\\_s::name](#), [mxml\\_value\\_s::num\\_attrs](#), [strdup](#), [mxml\\_node\\_s::type](#), [mxml\\_attr\\_s::value](#), and [mxml\\_node\\_s::value](#).

Referenced by [add\\_variable\(\)](#), [agent\\_xml\\_compose\\_\\_agent\\_code\(\)](#), [agent\\_xml\\_compose\\_\\_create\\_row\\_nodes\(\)](#), [agent\\_xml\\_compose\\_\\_data\(\)](#), [agent\\_xml\\_compose\\_\\_message\(\)](#), [agent\\_xml\\_compose\\_\\_task\(\)](#), [agent\\_xml\\_compose\\_\\_tasks\(\)](#), [fipa\\_envelope\\_Compose\\_params\(\)](#), [main\(\)](#), [mxml\\_parse\\_element\(\)](#), [scan\\_file\(\)](#), [sort\\_node\(\)](#), and [update\\_comment\(\)](#).

## 5.62 /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-entity.c File Reference

```
#include "config.h"
#include "mxml.h"
```

### Functions

- static [int default\\_callback](#) (const char \*name)
- [int mxmlEntityAddCallback](#) (int(\*cb)(const char \*name))
- const char \* [mxmlEntityGetName](#) (int val)
- [int mxmlEntityGetValue](#) (const char \*name)
- void [mxmlEntityRemoveCallback](#) (int(\*cb)(const char \*name))

### Variables

- static [int num\\_callbacks](#) = 1
- static [int\(\\* callbacks](#) [100])(const char \*name)

#### 5.62.1 Function Documentation

##### 5.62.1.1 static int default\_callback (const char \* *name*) [static]

Definition at line 173 of file mxml-entity.c.

References [int](#).

##### 5.62.1.2 int mxmlEntityAddCallback (int(\*) (const char \*name) *cb*)

Definition at line 69 of file mxml-entity.c.

References [callbacks](#), [mxml\\_error\(\)](#), and [num\\_callbacks](#).

##### 5.62.1.3 const char\* mxmlEntityGetName (int *val*)

Definition at line 95 of file mxml-entity.c.

Referenced by [mxml\\_write\\_name\(\)](#), and [mxml\\_write\\_string\(\)](#).

##### 5.62.1.4 int mxmlEntityGetValue (const char \* *name*)

Definition at line 125 of file mxml-entity.c.

References [callbacks](#), and [num\\_callbacks](#).

Referenced by [mxml\\_get\\_entity\(\)](#).



### 5.62.1.5 void mxmlEntityRemoveCallback (int(\*) (const char \*name) cb)

Definition at line 144 of file mxml-entity.c.

References `callbacks`, and `num_callbacks`.

## 5.62.2 Variable Documentation

### 5.62.2.1 int(\* callbacks[100])(const char \*name) [static]

Initial value:

```
{
    default_callback
}
```

Definition at line 58 of file mxml-entity.c.

Referenced by `mxmlEntityAddCallback()`, `mxmlEntityGetValue()`, and `mxmlEntityRemoveCallback()`.

### 5.62.2.2 int num\_callbacks = 1 [static]

Definition at line 57 of file mxml-entity.c.

Referenced by `mxmlEntityAddCallback()`, `mxmlEntityGetValue()`, and `mxmlEntityRemoveCallback()`.

## 5.63 /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-file.c File Reference

```
#include "mxml.h"
#include <unistd.h>
#include "config.h"
```

### Data Structures

- struct [mxml\\_fdbuf\\_s](#)

### Defines

- #define [ENCODE\\_UTF8](#) 0
- #define [ENCODE\\_UTF16BE](#) 1
- #define [ENCODE\\_UTF16LE](#) 2
- #define [mxml\\_bad\\_char](#)(ch) ((ch) < ' ' && (ch) != '\n' && (ch) != '\r' && (ch) != '\t')

### Typedefs

- typedef struct [mxml\\_fdbuf\\_s](#) [mxml\\_fdbuf\\_t](#)

### Functions

- static [int](#) [mxml\\_add\\_char](#) ([int](#) ch, [char](#) \*\*ptr, [char](#) \*\*buffer, [int](#) \*bufsize)
- static [int](#) [mxml\\_fd\\_getc](#) ([void](#) \*p, [int](#) \*encoding)
- static [int](#) [mxml\\_fd\\_putc](#) ([int](#) ch, [void](#) \*p)
- static [int](#) [mxml\\_fd\\_read](#) ([mxml\\_fdbuf\\_t](#) \*buf)
- static [int](#) [mxml\\_fd\\_write](#) ([mxml\\_fdbuf\\_t](#) \*buf)
- static [int](#) [mxml\\_file\\_getc](#) ([void](#) \*p, [int](#) \*encoding)
- static [int](#) [mxml\\_file\\_putc](#) ([int](#) ch, [void](#) \*p)
- static [int](#) [mxml\\_get\\_entity](#) ([mxml\\_node\\_t](#) \*parent, [void](#) \*p, [int](#) \*encoding, [int](#)(\*getc\_cb)([void](#) \*, [int](#) \*))
- static [mxml\\_node\\_t](#) \* [mxml\\_load\\_data](#) ([mxml\\_node\\_t](#) \*top, [void](#) \*p, [mxml\\_type\\_t](#)(\*cb)([mxml\\_node\\_t](#) \*), [int](#)(\*getc\_cb)([void](#) \*, [int](#) \*))
- static [int](#) [mxml\\_parse\\_element](#) ([mxml\\_node\\_t](#) \*node, [void](#) \*p, [int](#) \*encoding, [int](#)(\*getc\_cb)([void](#) \*, [int](#) \*))
- static [int](#) [mxml\\_string\\_getc](#) ([void](#) \*p, [int](#) \*encoding)
- static [int](#) [mxml\\_string\\_putc](#) ([int](#) ch, [void](#) \*p)
- static [int](#) [mxml\\_write\\_name](#) ([const](#) [char](#) \*s, [void](#) \*p, [int](#)(\*putc\_cb)([int](#), [void](#) \*))
- static [int](#) [mxml\\_write\\_node](#) ([mxml\\_node\\_t](#) \*node, [void](#) \*p, [const](#) [char](#) \*(\*cb)([mxml\\_node\\_t](#) \*, [int](#)), [int](#) col, [int](#)(\*putc\_cb)([int](#), [void](#) \*))
- static [int](#) [mxml\\_write\\_string](#) ([const](#) [char](#) \*s, [void](#) \*p, [int](#)(\*putc\_cb)([int](#), [void](#) \*))
- static [int](#) [mxml\\_write\\_ws](#) ([mxml\\_node\\_t](#) \*node, [void](#) \*p, [const](#) [char](#) \*(\*cb)([mxml\\_node\\_t](#) \*, [int](#)), [int](#) ws, [int](#) col, [int](#)(\*putc\_cb)([int](#), [void](#) \*))
- [mxml\\_node\\_t](#) \* [mxmlLoadFd](#) ([mxml\\_node\\_t](#) \*top, [int](#) fd, [mxml\\_type\\_t](#)(\*cb)([mxml\\_node\\_t](#) \*node))
- [mxml\\_node\\_t](#) \* [mxmlLoadFile](#) ([mxml\\_node\\_t](#) \*top, [FILE](#) \*fp, [mxml\\_type\\_t](#)(\*cb)([mxml\\_node\\_t](#) \*node))

- `mxml_node_t * mxmlLoadString (mxml_node_t *top, const char *s, mxml_type_t(*cb)(mxml_node_t *node))`
- `char * mxmlSaveAllocString (mxml_node_t *node, const char *(*cb)(mxml_node_t *node, int ws))`
- `int mxmlSaveFd (mxml_node_t *node, int fd, const char *(*cb)(mxml_node_t *node, int ws))`
- `int mxmlSaveFile (mxml_node_t *node, FILE *fp, const char *(*cb)(mxml_node_t *node, int ws))`
- `int mxmlSaveString (mxml_node_t *node, char *buffer, int bufsize, const char *(*cb)(mxml_node_t *node, int ws))`
- `void mxmlSetCustomHandlers (mxml_custom_load_cb_t load, mxml_custom_save_cb_t save)`
- `void mxmlSetErrorCallback (void(*cb)(const char *))`

## Variables

- `void(* mxml_error_cb )(const char *)`
- `static mxml_custom_load_cb_t mxml_custom_load_cb = NULL`
- `static mxml_custom_save_cb_t mxml_custom_save_cb = NULL`

### 5.63.1 Define Documentation

#### 5.63.1.1 `#define ENCODE_UTF16BE 1`

Definition at line 69 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, and `mxml_string_getc()`.

#### 5.63.1.2 `#define ENCODE_UTF16LE 2`

Definition at line 70 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, and `mxml_string_getc()`.

#### 5.63.1.3 `#define ENCODE_UTF8 0`

Definition at line 68 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, `mxml_load_data()`, and `mxml_string_getc()`.

#### 5.63.1.4 `#define mxml_bad_char(ch) ((ch) < ' ' && (ch) != '\n' && (ch) != '\r' && (ch) != '\t')`

Definition at line 77 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, `mxml_get_entity()`, and `mxml_string_getc()`.

### 5.63.2 Typedef Documentation

#### 5.63.2.1 `typedef struct mxml_fdbuf_s mxml_fdbuf_t`

### 5.63.3 Function Documentation

#### 5.63.3.1 `static int mxml_add_char (int ch, char **ptr, char **buffer, int *bufsize) [static]`

Definition at line 484 of file mxml-file.c.

References `mxml_error()`.

Referenced by `mxml_load_data()`, and `mxml_parse_element()`.

#### **5.63.3.2 static int mxml\_fd\_getc (void \* *p*, int \* *encoding*) [static]**

Definition at line 564 of file `mxml-file.c`.

References `buf`, `mxml_fdbuf_s::current`, `ENCODE_UTF16BE`, `ENCODE_UTF16LE`, `ENCODE_UTF8`, `mxml_fdbuf_s::end`, `mxml_bad_char`, `mxml_error()`, and `mxml_fd_read()`.

Referenced by `mxmlLoadFd()`.

#### **5.63.3.3 static int mxml\_fd\_putc (int *ch*, void \* *p*) [static]**

Definition at line 851 of file `mxml-file.c`.

References `buf`, `mxml_fdbuf_s::current`, `mxml_fdbuf_s::end`, and `mxml_fd_write()`.

Referenced by `mxmlSaveFd()`.

#### **5.63.3.4 static int mxml\_fd\_read (mxml\_fdbuf\_t \* *buf*) [static]**

Definition at line 920 of file `mxml-file.c`.

References `mxml_fdbuf_s::buffer`, `mxml_fdbuf_s::current`, `mxml_fdbuf_s::end`, and `mxml_fdbuf_s::fd`.

Referenced by `mxml_fd_getc()`.

#### **5.63.3.5 static int mxml\_fd\_write (mxml\_fdbuf\_t \* *buf*) [static]**

Definition at line 959 of file `mxml-file.c`.

References `mxml_fdbuf_s::buffer`, `mxml_fdbuf_s::current`, and `mxml_fdbuf_s::fd`.

Referenced by `mxml_fd_putc()`, and `mxmlSaveFd()`.

#### **5.63.3.6 static int mxml\_file\_getc (void \* *p*, int \* *encoding*) [static]**

Definition at line 1002 of file `mxml-file.c`.

References `ENCODE_UTF16BE`, `ENCODE_UTF16LE`, `ENCODE_UTF8`, `mxml_bad_char`, and `mxml_error()`.

Referenced by `mxmlLoadFile()`.

#### **5.63.3.7 static int mxml\_file\_putc (int *ch*, void \* *p*) [static]**

Definition at line 1201 of file `mxml-file.c`.

Referenced by `mxmlSaveFile()`.

**5.63.3.8** `static int mxml_get_entity (mxml_node_t * parent, void * p, int * encoding, int(*)(void *, int *) getc_cb) [static]`

Definition at line 1256 of file mxml-file.c.

References `mxml_value_u::element`, `mxml_bad_char`, `mxml_error()`, `mxmlEntityGetValue()`, `mxml_value_s::name`, and `mxml_node_s::value`.

Referenced by `mxml_load_data()`, and `mxml_parse_element()`.

**5.63.3.9** `static mxml_node_t * mxml_load_data (mxml_node_t * top, void * p, mxml_type_t(*)(mxml_node_t *) cb, int(*)(void *, int *) getc_cb) [static]`

Definition at line 1317 of file mxml-file.c.

References `mxml_value_u::element`, `ENCODE_UTF8`, `mxml_add_char()`, `MXML_CUSTOM`, `mxml_custom_load_cb`, `mxml_error()`, `mxml_get_entity()`, `MXML_INTEGER`, `MXML_OPAQUE`, `mxml_parse_element()`, `MXML_REAL`, `MXML_TEXT`, `mxmlDelete()`, `mxmlNewCustom()`, `mxmlNewElement()`, `mxmlNewInteger()`, `mxmlNewOpaque()`, `mxmlNewReal()`, `mxmlNewText()`, `mxml_value_s::name`, `mxml_node_s::parent`, and `mxml_node_s::value`.

Referenced by `mxmlLoadFd()`, `mxmlLoadFile()`, and `mxmlLoadString()`.

**5.63.3.10** `static int mxml_parse_element (mxml_node_t * node, void * p, int * encoding, int(*)(void *, int *) getc_cb) [static]`

Definition at line 1841 of file mxml-file.c.

References `mxml_value_u::element`, `mxml_add_char()`, `mxml_error()`, `mxml_get_entity()`, `mxmlElementGetAttr()`, `mxmlElementSetAttr()`, `mxml_value_s::name`, and `mxml_node_s::value`.

Referenced by `mxml_load_data()`.

**5.63.3.11** `static int mxml_string_getc (void * p, int * encoding) [static]`

Definition at line 2103 of file mxml-file.c.

References `ENCODE_UTF16BE`, `ENCODE_UTF16LE`, `ENCODE_UTF8`, `mxml_bad_char`, and `mxml_error()`.

Referenced by `mxmlLoadString()`.

**5.63.3.12** `static int mxml_string_putc (int ch, void * p) [static]`

Definition at line 2339 of file mxml-file.c.

Referenced by `mxmlSaveString()`.

**5.63.3.13** `static int mxml_write_name (const char * s, void * p, int(*)(int, void *) putc_cb) [static]`

Definition at line 2413 of file mxml-file.c.

References `mxmlEntityGetName()`.

Referenced by `mxml_write_node()`.

**5.63.3.14** `static int mxml_write_node (mxml_node_t * node, void * p, const char  
*(*)(mxml_node_t *, int) cb, int col, int(*) (int, void *) putc_cb) [static]`

Definition at line 2488 of file mxml-file.c.

References mxml\_value\_s::attrs, mxml\_node\_s::child, mxml\_value\_u::element, mxml\_value\_u::integer, MXML\_CUSTOM, mxml\_custom\_save\_cb, MXML\_ELEMENT, MXML\_INTEGER, MXML\_OPAQUE, MXML\_REAL, MXML\_TEXT, MXML\_WRAP, mxml\_write\_name(), mxml\_write\_string(), mxml\_write\_ws(), MXML\_WS\_AFTER\_CLOSE, MXML\_WS\_AFTER\_OPEN, MXML\_WS\_BEFORE\_CLOSE, MXML\_WS\_BEFORE\_OPEN, mxml\_attr\_s::name, mxml\_value\_s::name, mxml\_node\_s::next, mxml\_value\_s::num\_attrs, mxml\_value\_u::opaque, mxml\_node\_s::prev, mxml\_value\_u::real, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, mxml\_attr\_s::value, mxml\_node\_s::value, and mxml\_text\_s::whitespace.

Referenced by mxmlSaveFd(), mxmlSaveFile(), and mxmlSaveString().

**5.63.3.15** `static int mxml_write_string (const char * s, void * p, int(*) (int, void *) putc_cb)  
[static]`

Definition at line 2769 of file mxml-file.c.

References mxmlEntityGetName().

Referenced by mxml\_write\_node().

**5.63.3.16** `static int mxml_write_ws (mxml_node_t * node, void * p, const char *(*)(mxml_node_t  
*, int) cb, int ws, int col, int(*) (int, void *) putc_cb) [static]`

Definition at line 2809 of file mxml-file.c.

References MXML\_TAB.

Referenced by mxml\_write\_node().

**5.63.3.17** `mxml_node_t* mxmlLoadFd (mxml_node_t * top, int fd, mxml_type_t(*) (mxml_node_t  
*node) cb)`

Definition at line 160 of file mxml-file.c.

References buf, mxml\_fdbuf\_s::buffer, mxml\_fdbuf\_s::current, mxml\_fdbuf\_s::end, mxml\_fdbuf\_s::fd, mxml\_fd\_getc(), and mxml\_load\_data().

Referenced by main().

**5.63.3.18** `mxml_node_t* mxmlLoadFile (mxml_node_t * top, FILE * fp,  
mxml_type_t(*) (mxml_node_t *node) cb)`

Definition at line 200 of file mxml-file.c.

References mxml\_file\_getc(), and mxml\_load\_data().

Referenced by main().

### 5.63.3.19 **mxml\_node\_t\* mxmlLoadString (mxml\_node\_t \* *top*, const char \* *s*, mxml\_type\_t(\*) (mxml\_node\_t \**node*) *cb*)**

Definition at line 229 of file mxml-file.c.

References `mxml_load_data()`, and `mxml_string_getc()`.

Referenced by `acc_connection_Thread()`, `agent_xml_compose()`, `fipa_envelope_Compose()`, `fipa_envelope_Parse()`, `main()`, `MC_LoadAgentFromFile()`, `message_InitializeFromConnection()`, and `message_InitializeFromString()`.

### 5.63.3.20 **char\* mxmlSaveAllocString (mxml\_node\_t \* *node*, const char \*(\*) (mxml\_node\_t \**node*, int *ws*) *cb*)**

Definition at line 259 of file mxml-file.c.

References `mxmlSaveString()`, and `strdup`.

Referenced by `fipa_envelope_Compose()`, `MC_GetAgentXMLString()`, and `message_InitializeFromAgent()`.

### 5.63.3.21 **int mxmlSaveFd (mxml\_node\_t \* *node*, int *fd*, const char \*(\*) (mxml\_node\_t \**node*, int *ws*) *cb*)**

Definition at line 316 of file mxml-file.c.

References `buf`, `mxml_fdbuf_s::buffer`, `mxml_fdbuf_s::current`, `mxml_fdbuf_s::end`, `mxml_fdbuf_s::fd`, `mxml_fd_putc()`, `mxml_fd_write()`, and `mxml_write_node()`.

Referenced by `main()`.

### 5.63.3.22 **int mxmlSaveFile (mxml\_node\_t \* *node*, FILE \* *fp*, const char \*(\*) (mxml\_node\_t \**node*, int *ws*) *cb*)**

Definition at line 363 of file mxml-file.c.

References `mxml_file_putc()`, and `mxml_write_node()`.

Referenced by `main()`.

### 5.63.3.23 **int mxmlSaveString (mxml\_node\_t \* *node*, char \* *buffer*, int *bufsize*, const char \*(\*) (mxml\_node\_t \**node*, int *ws*) *cb*)**

Definition at line 405 of file mxml-file.c.

References `mxml_string_putc()`, and `mxml_write_node()`.

Referenced by `main()`, and `mxmlSaveAllocString()`.

### 5.63.3.24 **void mxmlSetCustomHandlers (mxml\_custom\_load\_cb\_t *load*, mxml\_custom\_save\_cb\_t *save*)**

Definition at line 457 of file mxml-file.c.

References `mxml_custom_load_cb`, and `mxml_custom_save_cb`.

**5.63.3.25 void mxmlSetErrorCallback (void(\*) (const char \*) cb)**

Definition at line 472 of file mxml-file.c.

References mxml\_error\_cb.

**5.63.4 Variable Documentation****5.63.4.1 mxml\_custom\_load\_cb\_t mxml\_custom\_load\_cb = NULL [static]**

Definition at line 104 of file mxml-file.c.

Referenced by mxml\_load\_data(), and mxmlSetCustomHandlers().

**5.63.4.2 mxml\_custom\_save\_cb\_t mxml\_custom\_save\_cb = NULL [static]**

Definition at line 105 of file mxml-file.c.

Referenced by mxml\_write\_node(), and mxmlSetCustomHandlers().

**5.63.4.3 void(\* mxml\_error\_cb)(const char \*)**

Definition at line 45 of file mxml-private.c.

Referenced by mxml\_error(), and mxmlSetErrorCallback().



## 5.64 /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-index.c File Reference

```
#include "config.h"
#include "mxml.h"
```

### Functions

- static [int](#) [index\\_compare](#) ([mxml\\_index\\_t](#) \*ind, [mxml\\_node\\_t](#) \*first, [mxml\\_node\\_t](#) \*second)
- static [int](#) [index\\_find](#) ([mxml\\_index\\_t](#) \*ind, const char \*element, const char \*value, [mxml\\_node\\_t](#) \*node)
- static void [index\\_sort](#) ([mxml\\_index\\_t](#) \*ind, [int](#) left, [int](#) right)
- void [mxmlIndexDelete](#) ([mxml\\_index\\_t](#) \*ind)
- [mxml\\_node\\_t](#) \* [mxmlIndexEnum](#) ([mxml\\_index\\_t](#) \*ind)
- [mxml\\_node\\_t](#) \* [mxmlIndexFind](#) ([mxml\\_index\\_t](#) \*ind, const char \*element, const char \*value)
- [mxml\\_index\\_t](#) \* [mxmlIndexNew](#) ([mxml\\_node\\_t](#) \*node, const char \*element, const char \*attr)
- [mxml\\_node\\_t](#) \* [mxmlIndexReset](#) ([mxml\\_index\\_t](#) \*ind)

### 5.64.1 Function Documentation

**5.64.1.1** static [int](#) [index\\_compare](#) ([mxml\\_index\\_t](#) \*ind, [mxml\\_node\\_t](#) \*first, [mxml\\_node\\_t](#) \*second) [**static**]

Definition at line 498 of file [mxml-index.c](#).

References [mxml\\_index\\_s::attr](#), [mxml\\_value\\_u::element](#), [mxmlElementGetAttr\(\)](#), [mxml\\_value\\_s::name](#), and [mxml\\_node\\_s::value](#).

Referenced by [index\\_sort\(\)](#).

**5.64.1.2** static [int](#) [index\\_find](#) ([mxml\\_index\\_t](#) \*ind, const char \*element, const char \*value, [mxml\\_node\\_t](#) \*node) [**static**]

Definition at line 537 of file [mxml-index.c](#).

References [mxml\\_index\\_s::attr](#), [mxml\\_value\\_u::element](#), [mxmlElementGetAttr\(\)](#), [mxml\\_value\\_s::name](#), and [mxml\\_node\\_s::value](#).

Referenced by [mxmlIndexFind\(\)](#).

**5.64.1.3** static void [index\\_sort](#) ([mxml\\_index\\_t](#) \*ind, [int](#) left, [int](#) right) [**static**]

Definition at line 580 of file [mxml-index.c](#).

References [index\\_compare\(\)](#), and [mxml\\_index\\_s::nodes](#).

Referenced by [mxmlIndexNew\(\)](#).

**5.64.1.4** void [mxmlIndexDelete](#) ([mxml\\_index\\_t](#) \*ind)

Definition at line 62 of file [mxml-index.c](#).

References `mxml_index_s::alloc_nodes`, `mxml_index_s::attr`, and `mxml_index_s::nodes`.

Referenced by `main()`, and `mxmlIndexNew()`.

#### 5.64.1.5 `mxml_node_t* mxmlIndexEnum (mxml_index_t * ind)`

Definition at line 92 of file `mxml-index.c`.

References `mxml_index_s::cur_node`, `mxml_index_s::nodes`, and `mxml_index_s::num_nodes`.

Referenced by `main()`, and `mxmlIndexFind()`.

#### 5.64.1.6 `mxml_node_t* mxmlIndexFind (mxml_index_t * ind, const char * element, const char * value)`

Definition at line 122 of file `mxml-index.c`.

References `mxml_index_s::attr`, `mxml_index_s::cur_node`, `index_find()`, `mxmlIndexEnum()`, `mxml_index_s::nodes`, and `mxml_index_s::num_nodes`.

Referenced by `main()`.

#### 5.64.1.7 `mxml_index_t* mxmlIndexNew (mxml_node_t * node, const char * element, const char * attr)`

Definition at line 305 of file `mxml-index.c`.

References `mxml_index_s::alloc_nodes`, `mxml_index_s::attr`, `mxml_value_u::element`, `index_sort()`, `MXML_DESCEND`, `mxml_error()`, `mxmlElementGetAttr()`, `mxmlFindElement()`, `mxmlIndexDelete()`, `mxml_value_s::name`, `mxml_index_s::nodes`, `mxml_index_s::num_nodes`, `strdup`, and `mxml_node_s::value`.

Referenced by `main()`.

#### 5.64.1.8 `mxml_node_t* mxmlIndexReset (mxml_index_t * ind)`

Definition at line 463 of file `mxml-index.c`.

References `mxml_index_s::cur_node`, `mxml_index_s::nodes`, and `mxml_index_s::num_nodes`.

Referenced by `main()`.

## 5.65 /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-node.c File Reference

```
#include "config.h"
#include "mxml.h"
```

### Functions

- static [mxml\\_node\\_t](#) \* [mxml\\_new](#) ([mxml\\_node\\_t](#) \*parent, [mxml\\_type\\_t](#) type)
- void [mxmlAdd](#) ([mxml\\_node\\_t](#) \*parent, int where, [mxml\\_node\\_t](#) \*child, [mxml\\_node\\_t](#) \*node)
- void [mxmlDelete](#) ([mxml\\_node\\_t](#) \*node)
- [mxml\\_node\\_t](#) \* [mxmlNewCustom](#) ([mxml\\_node\\_t](#) \*parent, void \*data, void(\*destroy)(void \*))
- [mxml\\_node\\_t](#) \* [mxmlNewElement](#) ([mxml\\_node\\_t](#) \*parent, const char \*name)
- [mxml\\_node\\_t](#) \* [mxmlNewInteger](#) ([mxml\\_node\\_t](#) \*parent, int integer)
- [mxml\\_node\\_t](#) \* [mxmlNewOpaque](#) ([mxml\\_node\\_t](#) \*parent, const char \*opaque)
- [mxml\\_node\\_t](#) \* [mxmlNewReal](#) ([mxml\\_node\\_t](#) \*parent, double real)
- [mxml\\_node\\_t](#) \* [mxmlNewText](#) ([mxml\\_node\\_t](#) \*parent, int whitespace, const char \*string)
- [mxml\\_node\\_t](#) \* [mxmlNewTextf](#) ([mxml\\_node\\_t](#) \*parent, int whitespace, const char \*format,...)
- void [mxmlRemove](#) ([mxml\\_node\\_t](#) \*node)

### 5.65.1 Function Documentation

#### 5.65.1.1 static [mxml\\_node\\_t](#) \* [mxml\\_new](#) ([mxml\\_node\\_t](#) \*parent, [mxml\\_type\\_t](#) type) [static]

Definition at line 621 of file [mxml-node.c](#).

References [MXML\\_ADD\\_AFTER](#), [MXML\\_ADD\\_TO\\_PARENT](#), [mxmlAdd\(\)](#), [node](#), and [mxml\\_node\\_s::type](#).

Referenced by [mxmlNewCustom\(\)](#), [mxmlNewElement\(\)](#), [mxmlNewInteger\(\)](#), [mxmlNewOpaque\(\)](#), [mxmlNewReal\(\)](#), [mxmlNewText\(\)](#), and [mxmlNewTextf\(\)](#).

#### 5.65.1.2 void [mxmlAdd](#) ([mxml\\_node\\_t](#) \*parent, int where, [mxml\\_node\\_t](#) \*child, [mxml\\_node\\_t](#) \*node)

Definition at line 66 of file [mxml-node.c](#).

References [mxml\\_node\\_s::child](#), [mxml\\_node\\_s::last\\_child](#), [MXML\\_ADD\\_AFTER](#), [MXML\\_ADD\\_BEFORE](#), [mxmlRemove\(\)](#), [mxml\\_node\\_s::next](#), [mxml\\_node\\_s::parent](#), and [mxml\\_node\\_s::prev](#).

Referenced by [add\\_variable\(\)](#), [agent\\_xml\\_compose\(\)](#), [agent\\_xml\\_compose\\_\\_agent\\_data\(\)](#), [agent\\_xml\\_compose\\_\\_create\\_row\\_nodes\(\)](#), [agent\\_xml\\_compose\\_\\_data\(\)](#), [agent\\_xml\\_compose\\_\\_gaf\\_message\(\)](#), [agent\\_xml\\_compose\\_\\_message\(\)](#), [agent\\_xml\\_compose\\_\\_mobile\\_agent\(\)](#), [agent\\_xml\\_compose\\_\\_task\(\)](#), [agent\\_xml\\_compose\\_\\_tasks\(\)](#), [fipa\\_envelope\\_Compose\(\)](#), [fipa\\_envelope\\_Compose\\_\\_envelope\(\)](#), [fipa\\_envelope\\_Compose\\_\\_params\(\)](#), [mxml\\_new\(\)](#), [scan\\_file\(\)](#), and [sort\\_node\(\)](#).

#### 5.65.1.3 void [mxmlDelete](#) ([mxml\\_node\\_t](#) \*node)

Definition at line 200 of file [mxml-node.c](#).

References `mxml_value_s::attrs`, `mxml_node_s::child`, `mxml_value_u::custom`, `mxml_custom_s::data`, `mxml_custom_s::destroy`, `mxml_value_u::element`, `MXML_CUSTOM`, `MXML_ELEMENT`, `MXML_INTEGER`, `MXML_OPAQUE`, `MXML_REAL`, `MXML_TEXT`, `mxmlDelete()`, `mxmlRemove()`, `mxml_attr_s::name`, `mxml_value_s::name`, `mxml_value_s::num_attrs`, `mxml_value_u::opaque`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::type`, `mxml_attr_s::value`, and `mxml_node_s::value`.

Referenced by `add_variable()`, `agent_datastate_Destroy()`, `fipa_envelope_Compose()`, `fipa_envelope_Parse()`, `main()`, `message_Destroy()`, `mxml_load_data()`, `mxmlDelete()`, `scan_file()`, and `sort_node()`.

#### 5.65.1.4 `mxml_node_t* mxmlNewCustom (mxml_node_t * parent, void * data, void(*) (void *) destroy)`

Definition at line 291 of file `mxml-node.c`.

References `mxml_value_u::custom`, `mxml_custom_s::data`, `mxml_custom_s::destroy`, `MXML_CUSTOM`, `mxml_new()`, `node`, and `mxml_node_s::value`.

Referenced by `mxml_load_data()`.

#### 5.65.1.5 `mxml_node_t* mxmlNewElement (mxml_node_t * parent, const char * name)`

Definition at line 327 of file `mxml-node.c`.

References `mxml_value_u::element`, `MXML_ELEMENT`, `mxml_new()`, `mxml_value_s::name`, `node`, `strdup`, and `mxml_node_s::value`.

Referenced by `add_variable()`, `agent_xml_compose_agent_code()`, `agent_xml_compose_agent_data()`, `agent_xml_compose_create_row_nodes()`, `agent_xml_compose_data()`, `agent_xml_compose_gaf_message()`, `agent_xml_compose_home()`, `agent_xml_compose_message()`, `agent_xml_compose_mobile_agent()`, `agent_xml_compose_name()`, `agent_xml_compose_owner()`, `agent_xml_compose_task()`, `agent_xml_compose_tasks()`, `agent_xml_compose_wg_code()`, `fipa_envelope_Compose_acl_representation()`, `fipa_envelope_Compose_date()`, `fipa_envelope_Compose_envelope()`, `fipa_envelope_Compose_from()`, `fipa_envelope_Compose_intended_receiver()`, `fipa_envelope_Compose_params()`, `fipa_envelope_Compose_payload_encoding()`, `fipa_envelope_Compose_to()`, `main()`, `mxml_load_data()`, `scan_file()`, and `xml_new_cdata()`.

#### 5.65.1.6 `mxml_node_t* mxmlNewInteger (mxml_node_t * parent, int integer)`

Definition at line 365 of file `mxml-node.c`.

References `mxml_value_u::integer`, `MXML_INTEGER`, `mxml_new()`, `node`, and `mxml_node_s::value`.

Referenced by `main()`, and `mxml_load_data()`.

#### 5.65.1.7 `mxml_node_t* mxmlNewOpaque (mxml_node_t * parent, const char * opaque)`

Definition at line 396 of file `mxml-node.c`.

References `mxml_new()`, `MXML_OPAQUE`, `node`, `mxml_value_u::opaque`, `strdup`, and `mxml_node_s::value`.

Referenced by `main()`, and `mxml_load_data()`.

**5.65.1.8 mxml\_node\_t\* mxmlNewReal (mxml\_node\_t \* *parent*, double *real*)**

Definition at line 434 of file mxml-node.c.

References mxml\_new(), MXML\_REAL, node, mxml\_value\_u::real, and mxml\_node\_s::value.

Referenced by main(), and mxml\_load\_data().

**5.65.1.9 mxml\_node\_t\* mxmlNewText (mxml\_node\_t \* *parent*, int *whitespace*, const char \* *string*)**

Definition at line 466 of file mxml-node.c.

References mxml\_new(), MXML\_TEXT, node, strdup, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::value, and mxml\_text\_s::whitespace.

Referenced by agent\_xml\_compose\_\_create\_row\_nodes(), agent\_xml\_compose\_\_home(), agent\_xml\_compose\_\_name(), agent\_xml\_compose\_\_owner(), agent\_xml\_compose\_\_wg\_code(), fipa\_envelope\_Compose\_\_acl\_representation(), fipa\_envelope\_Compose\_\_date(), fipa\_envelope\_Compose\_\_from(), fipa\_envelope\_Compose\_\_intended\_receiver(), fipa\_envelope\_Compose\_\_payload\_encoding(), fipa\_envelope\_Compose\_\_to(), main(), mxml\_load\_data(), and scan\_file().

**5.65.1.10 mxml\_node\_t\* mxmlNewTextf (mxml\_node\_t \* *parent*, int *whitespace*, const char \* *format*, ...)**

Definition at line 510 of file mxml-node.c.

References mxml\_new(), mxml\_strdupf(), MXML\_TEXT, node, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::value, and mxml\_text\_s::whitespace.

**5.65.1.11 void mxmlRemove (mxml\_node\_t \* *node*)**

Definition at line 557 of file mxml-node.c.

References mxml\_node\_s::child, mxml\_node\_s::last\_child, mxml\_node\_s::next, mxml\_node\_s::parent, and mxml\_node\_s::prev.

Referenced by mxmlAdd(), and mxmlDelete().

## 5.66 /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-private.c

### File Reference

```
#include "config.h"
#include "mxml.h"
```

### Functions

- void [mxml\\_error](#) (const char \*format,...)
- [mxml\\_type\\_t mxml\\_integer\\_cb](#) ([mxml\\_node\\_t](#) \*node)
- [mxml\\_type\\_t mxml\\_opaque\\_cb](#) ([mxml\\_node\\_t](#) \*node)
- [mxml\\_type\\_t mxml\\_real\\_cb](#) ([mxml\\_node\\_t](#) \*node)

### Variables

- void(\* [mxml\\_error\\_cb](#) )(const char \*) = NULL

### 5.66.1 Function Documentation

#### 5.66.1.1 void [mxml\\_error](#) (const char \**format*, ...)

Definition at line 53 of file [mxml-private.c](#).

References [mxml\\_error\\_cb](#), and [mxml\\_strdup\(\)](#).

Referenced by [mxml\\_add\\_char\(\)](#), [mxml\\_fd\\_getc\(\)](#), [mxml\\_file\\_getc\(\)](#), [mxml\\_get\\_entity\(\)](#), [mxml\\_load\\_data\(\)](#), [mxml\\_parse\\_element\(\)](#), [mxml\\_string\\_getc\(\)](#), [mxmlElementSetAttr\(\)](#), [mxmlEntityAddCallback\(\)](#), and [mxmlIndexNew\(\)](#).

#### 5.66.1.2 [mxml\\_type\\_t mxml\\_integer\\_cb](#) ([mxml\\_node\\_t](#) \**node*)

Definition at line 99 of file [mxml-private.c](#).

References [MXML\\_INTEGER](#).

#### 5.66.1.3 [mxml\\_type\\_t mxml\\_opaque\\_cb](#) ([mxml\\_node\\_t](#) \**node*)

Definition at line 112 of file [mxml-private.c](#).

References [MXML\\_OPAQUE](#).

#### 5.66.1.4 [mxml\\_type\\_t mxml\\_real\\_cb](#) ([mxml\\_node\\_t](#) \**node*)

Definition at line 125 of file [mxml-private.c](#).

References [MXML\\_REAL](#).

## 5.66.2 Variable Documentation

### 5.66.2.1 `void(* mxml_error_cb)(const char *) = NULL`

Definition at line 45 of file mxml-private.c.

Referenced by `mxml_error()`, and `mxmlSetErrorCallback()`.

## 5.67 /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-search.c

### File Reference

```
#include "config.h"
#include "mxml.h"
```

#### Functions

- [mxml\\_node\\_t \\* mxmlFindElement](#) ([mxml\\_node\\_t \\* node](#), [mxml\\_node\\_t \\* top](#), [const char \\* name](#), [const char \\* attr](#), [const char \\* value](#), [int descend](#))
- [mxml\\_node\\_t \\* mxmlWalkNext](#) ([mxml\\_node\\_t \\* node](#), [mxml\\_node\\_t \\* top](#), [int descend](#))
- [mxml\\_node\\_t \\* mxmlWalkPrev](#) ([mxml\\_node\\_t \\* node](#), [mxml\\_node\\_t \\* top](#), [int descend](#))

#### 5.67.1 Function Documentation

##### 5.67.1.1 [mxml\\_node\\_t \\* mxmlFindElement](#) ([mxml\\_node\\_t \\* node](#), [mxml\\_node\\_t \\* top](#), [const char \\* name](#), [const char \\* attr](#), [const char \\* value](#), [int descend](#))

Definition at line 55 of file mxml-search.c.

References [mxml\\_value\\_u::element](#), [MXML\\_DESCEND](#), [MXML\\_ELEMENT](#), [mxmlElementGetAttr\(\)](#), [mxmlWalkNext\(\)](#), [mxml\\_value\\_s::name](#), [mxml\\_node\\_s::next](#), [mxml\\_node\\_s::type](#), and [mxml\\_node\\_s::value](#).

Referenced by [agent\\_xml\\_parse\\_\\_fill\\_row\\_data\(\)](#), [agent\\_xml\\_parse\\_\\_task\(\)](#), [agent\\_xml\\_parse\\_\\_tasks\(\)](#), [fipa\\_envelope\\_HandleAclRepresentation\(\)](#), [fipa\\_envelope\\_HandleComments\(\)](#), [fipa\\_envelope\\_HandleDate\(\)](#), [fipa\\_envelope\\_HandleEnvelope\(\)](#), [fipa\\_envelope\\_HandleFrom\(\)](#), [fipa\\_envelope\\_HandleIntendedReceiver\(\)](#), [fipa\\_envelope\\_HandleParams\(\)](#), [fipa\\_envelope\\_HandlePayloadEncoding\(\)](#), [fipa\\_envelope\\_HandlePayloadLength\(\)](#), [fipa\\_envelope\\_HandleReceived\(\)](#), [fipa\\_envelope\\_HandleTo\(\)](#), [fipa\\_envelope\\_ParseAddresses\(\)](#), [fipa\\_envelope\\_ParseAgentIdentifier\(\)](#), [fipa\\_envelope\\_ParseResolvers\(\)](#), [main\(\)](#), [MC\\_LoadAgentFromFile\(\)](#), [message\\_xml\\_parse\(\)](#), [mxmlIndexNew\(\)](#), [scan\\_file\(\)](#), [sort\\_node\(\)](#), [write\\_documentation\(\)](#), [write\\_element\(\)](#), [xml\\_find\\_sibling\(\)](#), and [xml\\_get\\_child\(\)](#).

##### 5.67.1.2 [mxml\\_node\\_t \\* mxmlWalkNext](#) ([mxml\\_node\\_t \\* node](#), [mxml\\_node\\_t \\* top](#), [int descend](#))

Definition at line 137 of file mxml-search.c.

References [mxml\\_node\\_s::child](#), [mxml\\_node\\_s::next](#), and [mxml\\_node\\_s::parent](#).

Referenced by [mxmlFindElement\(\)](#), and [write\\_element\(\)](#).

##### 5.67.1.3 [mxml\\_node\\_t \\* mxmlWalkPrev](#) ([mxml\\_node\\_t \\* node](#), [mxml\\_node\\_t \\* top](#), [int descend](#))

Definition at line 173 of file mxml-search.c.

References [mxml\\_node\\_s::last\\_child](#), [mxml\\_node\\_s::parent](#), and [mxml\\_node\\_s::prev](#).



## 5.68 /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-set.c File Reference

```
#include "config.h"
#include "mxml.h"
```

### Functions

- [int mxmlSetCustom](#) ([mxml\\_node\\_t](#) \**node*, void \**data*, void(\**destroy*)(void \*))
- [int mxmlSetElement](#) ([mxml\\_node\\_t](#) \**node*, const char \**name*)
- [int mxmlSetInteger](#) ([mxml\\_node\\_t](#) \**node*, int *integer*)
- [int mxmlSetOpaque](#) ([mxml\\_node\\_t](#) \**node*, const char \**opaque*)
- [int mxmlSetReal](#) ([mxml\\_node\\_t](#) \**node*, double *real*)
- [int mxmlSetText](#) ([mxml\\_node\\_t](#) \**node*, int *whitespace*, const char \**string*)
- [int mxmlSetTextf](#) ([mxml\\_node\\_t](#) \**node*, int *whitespace*, const char \**format*,...)

### 5.68.1 Function Documentation

#### 5.68.1.1 int mxmlSetCustom (mxml\_node\_t \* *node*, void \* *data*, void(\*) (void \*) *destroy*)

Definition at line 50 of file mxml-set.c.

References [mxml\\_value\\_u::custom](#), [mxml\\_custom\\_s::data](#), [mxml\\_custom\\_s::destroy](#), [MXML\\_CUSTOM](#), [mxml\\_node\\_s::type](#), and [mxml\\_node\\_s::value](#).

#### 5.68.1.2 int mxmlSetElement (mxml\_node\_t \* *node*, const char \* *name*)

Definition at line 83 of file mxml-set.c.

References [mxml\\_value\\_u::element](#), [MXML\\_ELEMENT](#), [mxml\\_value\\_s::name](#), [strdup](#), [mxml\\_node\\_s::type](#), and [mxml\\_node\\_s::value](#).

#### 5.68.1.3 int mxmlSetInteger (mxml\_node\_t \* *node*, int *integer*)

Definition at line 113 of file mxml-set.c.

References [mxml\\_value\\_u::integer](#), [MXML\\_INTEGER](#), [mxml\\_node\\_s::type](#), and [mxml\\_node\\_s::value](#).

#### 5.68.1.4 int mxmlSetOpaque (mxml\_node\_t \* *node*, const char \* *opaque*)

Definition at line 140 of file mxml-set.c.

References [MXML\\_OPAQUE](#), [mxml\\_value\\_u::opaque](#), [strdup](#), [mxml\\_node\\_s::type](#), and [mxml\\_node\\_s::value](#).

#### 5.68.1.5 int mxmlSetReal (mxml\_node\_t \* *node*, double *real*)

Definition at line 170 of file mxml-set.c.

References [MXML\\_REAL](#), [mxml\\_value\\_u::real](#), [mxml\\_node\\_s::type](#), and [mxml\\_node\\_s::value](#).

**5.68.1.6 int mxmlSetText (mxml\_node\_t \* *node*, int *whitespace*, const char \* *string*)**

Definition at line 197 of file mxml-set.c.

References MXML\_TEXT, strdup, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, mxml\_node\_s::value, and mxml\_text\_s::whitespace.

**5.68.1.7 int mxmlSetTextf (mxml\_node\_t \* *node*, int *whitespace*, const char \* *format*, ...)**

Definition at line 229 of file mxml-set.c.

References mxml\_strdupf(), MXML\_TEXT, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, mxml\_node\_s::value, and mxml\_text\_s::whitespace.

## 5.69 /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-string.c File Reference

```
#include "config.h"
```

### Functions

- `char * mxml_strdup` (`const char *s`)
- `char * mxml_strdupf` (`const char *format, va_list ap`)
- `int mxml_vsnprintf` (`char *buffer, size_t bufsize, const char *format, va_list ap`)

### 5.69.1 Function Documentation

#### 5.69.1.1 `char* mxml_strdup (const char * s)`

Definition at line 45 of file `mxml-string.c`.

#### 5.69.1.2 `char* mxml_strdupf (const char * format, va_list ap)`

Definition at line 66 of file `mxml-string.c`.

References `mxml_vsnprintf()`, and `strdup`.

Referenced by `mxml_error()`, `mxmlNewTextf()`, and `mxmlSetTextf()`.

#### 5.69.1.3 `int mxml_vsnprintf (char * buffer, size_t bufsize, const char * format, va_list ap)`

Definition at line 111 of file `mxml-string.c`.

References `size`.

Referenced by `mxml_strdupf()`.

## 5.70 /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#include <errno.h>
```

### Data Structures

- struct [mxml\\_attr\\_s](#)
- struct [mxml\\_value\\_s](#)
- struct [mxml\\_text\\_s](#)
- struct [mxml\\_custom\\_s](#)
- union [mxml\\_value\\_u](#)
- struct [mxml\\_node\\_s](#)
- struct [mxml\\_index\\_s](#)

### Defines

- #define [MXML\\_WRAP](#) 72
- #define [MXML\\_TAB](#) 8
- #define [MXML\\_NO\\_CALLBACK](#) 0
- #define [MXML\\_INTEGER\\_CALLBACK](#) mxml\_integer\_cb
- #define [MXML\\_OPAQUE\\_CALLBACK](#) mxml\_opaque\_cb
- #define [MXML\\_REAL\\_CALLBACK](#) mxml\_real\_cb
- #define [MXML\\_TEXT\\_CALLBACK](#) 0
- #define [MXML\\_NO\\_PARENT](#) 0
- #define [MXML\\_DESCEND](#) 1
- #define [MXML\\_NO\\_DESCEND](#) 0
- #define [MXML\\_DESCEND\\_FIRST](#) -1
- #define [MXML\\_WS\\_BEFORE\\_OPEN](#) 0
- #define [MXML\\_WS\\_AFTER\\_OPEN](#) 1
- #define [MXML\\_WS\\_BEFORE\\_CLOSE](#) 2
- #define [MXML\\_WS\\_AFTER\\_CLOSE](#) 3
- #define [MXML\\_ADD\\_BEFORE](#) 0
- #define [MXML\\_ADD\\_AFTER](#) 1
- #define [MXML\\_ADD\\_TO\\_PARENT](#) NULL

### Typedefs

- typedef enum [mxml\\_type\\_e](#) mxml\_type\_t
- typedef struct [mxml\\_attr\\_s](#) mxml\_attr\_t
- typedef struct [mxml\\_value\\_s](#) mxml\_element\_t
- typedef struct [mxml\\_text\\_s](#) mxml\_text\_t
- typedef struct [mxml\\_custom\\_s](#) mxml\_custom\_t

- typedef union [mxml\\_value\\_u](#) [mxml\\_value\\_t](#)
- typedef struct [mxml\\_node\\_s](#) [mxml\\_node\\_t](#)
- typedef struct [mxml\\_index\\_s](#) [mxml\\_index\\_t](#)
- typedef [int](#)(\* [mxml\\_custom\\_load\\_cb\\_t](#))([mxml\\_node\\_t](#) \*, const char \*)
- typedef char \*(\* [mxml\\_custom\\_save\\_cb\\_t](#))([mxml\\_node\\_t](#) \*)

## Enumerations

- enum [mxml\\_type\\_e](#) {  
[MXML\\_ELEMENT](#), [MXML\\_INTEGER](#), [MXML\\_OPAQUE](#), [MXML\\_REAL](#),  
[MXML\\_TEXT](#), [MXML\\_CUSTOM](#) }

## Functions

- void [mxmlAdd](#) ([mxml\\_node\\_t](#) \*parent, [int](#) where, [mxml\\_node\\_t](#) \*child, [mxml\\_node\\_t](#) \*node)
- void [mxmlDelete](#) ([mxml\\_node\\_t](#) \*node)
- const char \* [mxmlElementGetAttr](#) ([mxml\\_node\\_t](#) \*node, const char \*name)
- void [mxmlElementSetAttr](#) ([mxml\\_node\\_t](#) \*node, const char \*name, const char \*value)
- [int](#) [mxmlEntityAddCallback](#) ([int](#)(\*cb)(const char \*name))
- const char \* [mxmlEntityGetName](#) ([int](#) val)
- [int](#) [mxmlEntityGetValue](#) (const char \*name)
- void [mxmlEntityRemoveCallback](#) ([int](#)(\*cb)(const char \*name))
- [mxml\\_node\\_t](#) \* [mxmlFindElement](#) ([mxml\\_node\\_t](#) \*node, [mxml\\_node\\_t](#) \*top, const char \*name, const char \*attr, const char \*value, [int](#) descend)
- void [mxmlIndexDelete](#) ([mxml\\_index\\_t](#) \*ind)
- [mxml\\_node\\_t](#) \* [mxmlIndexEnum](#) ([mxml\\_index\\_t](#) \*ind)
- [mxml\\_node\\_t](#) \* [mxmlIndexFind](#) ([mxml\\_index\\_t](#) \*ind, const char \*element, const char \*value)
- [mxml\\_index\\_t](#) \* [mxmlIndexNew](#) ([mxml\\_node\\_t](#) \*node, const char \*element, const char \*attr)
- [mxml\\_node\\_t](#) \* [mxmlIndexReset](#) ([mxml\\_index\\_t](#) \*ind)
- [mxml\\_node\\_t](#) \* [mxmlLoadFd](#) ([mxml\\_node\\_t](#) \*top, [int](#) fd, [mxml\\_type\\_t](#)(\*cb)([mxml\\_node\\_t](#) \*))
- [mxml\\_node\\_t](#) \* [mxmlLoadFile](#) ([mxml\\_node\\_t](#) \*top, FILE \*fp, [mxml\\_type\\_t](#)(\*cb)([mxml\\_node\\_t](#) \*))
- [mxml\\_node\\_t](#) \* [mxmlLoadString](#) ([mxml\\_node\\_t](#) \*top, const char \*s, [mxml\\_type\\_t](#)(\*cb)([mxml\\_node\\_t](#) \*))
- [mxml\\_node\\_t](#) \* [mxmlNewCustom](#) ([mxml\\_node\\_t](#) \*parent, void \*data, void(\*destroy)(void \*))
- [mxml\\_node\\_t](#) \* [mxmlNewElement](#) ([mxml\\_node\\_t](#) \*parent, const char \*name)
- [mxml\\_node\\_t](#) \* [mxmlNewInteger](#) ([mxml\\_node\\_t](#) \*parent, [int](#) integer)
- [mxml\\_node\\_t](#) \* [mxmlNewOpaque](#) ([mxml\\_node\\_t](#) \*parent, const char \*opaque)
- [mxml\\_node\\_t](#) \* [mxmlNewReal](#) ([mxml\\_node\\_t](#) \*parent, double real)
- [mxml\\_node\\_t](#) \* [mxmlNewText](#) ([mxml\\_node\\_t](#) \*parent, [int](#) whitespace, const char \*string)
- [mxml\\_node\\_t](#) \* [mxmlNewTextf](#) ([mxml\\_node\\_t](#) \*parent, [int](#) whitespace, const char \*format,...)
- void [mxmlRemove](#) ([mxml\\_node\\_t](#) \*node)
- char \* [mxmlSaveAllocString](#) ([mxml\\_node\\_t](#) \*node, const char \*(\*cb)([mxml\\_node\\_t](#) \*, [int](#)))
- [int](#) [mxmlSaveFd](#) ([mxml\\_node\\_t](#) \*node, [int](#) fd, const char \*(\*cb)([mxml\\_node\\_t](#) \*, [int](#)))
- [int](#) [mxmlSaveFile](#) ([mxml\\_node\\_t](#) \*node, FILE \*fp, const char \*(\*cb)([mxml\\_node\\_t](#) \*, [int](#)))
- [int](#) [mxmlSaveString](#) ([mxml\\_node\\_t](#) \*node, char \*buffer, [int](#) bufsize, const char \*(\*cb)([mxml\\_node\\_t](#) \*, [int](#)))
- [int](#) [mxmlSetCustom](#) ([mxml\\_node\\_t](#) \*node, void \*data, void(\*destroy)(void \*))
- void [mxmlSetCustomHandlers](#) ([mxml\\_custom\\_load\\_cb\\_t](#) load, [mxml\\_custom\\_save\\_cb\\_t](#) save)
- [int](#) [mxmlSetElement](#) ([mxml\\_node\\_t](#) \*node, const char \*name)

- void `mxmlSetErrorCallback` (void(\*cb)(const char \*))
- int `mxmlSetInteger` (mxml\_node\_t \*node, int integer)
- int `mxmlSetOpaque` (mxml\_node\_t \*node, const char \*opaque)
- int `mxmlSetReal` (mxml\_node\_t \*node, double real)
- int `mxmlSetText` (mxml\_node\_t \*node, int whitespace, const char \*string)
- int `mxmlSetTextf` (mxml\_node\_t \*node, int whitespace, const char \*format,...)
- mxml\_node\_t \* `mxmlWalkNext` (mxml\_node\_t \*node, mxml\_node\_t \*top, int descend)
- mxml\_node\_t \* `mxmlWalkPrev` (mxml\_node\_t \*node, mxml\_node\_t \*top, int descend)
- void `mxml_error` (const char \*format,...)
- mxml\_type\_t `mxml_integer_cb` (mxml\_node\_t \*node)
- mxml\_type\_t `mxml_opaque_cb` (mxml\_node\_t \*node)
- mxml\_type\_t `mxml_real_cb` (mxml\_node\_t \*node)

## 5.70.1 Define Documentation

### 5.70.1.1 #define MXML\_ADD\_AFTER 1

Definition at line 68 of file `mxml.h`.

Referenced by `add_variable()`, `agent_xml_compose()`, `agent_xml_compose__agent_data()`, `agent_xml_compose__create_row_nodes()`, `agent_xml_compose__data()`, `agent_xml_compose__gaf_message()`, `agent_xml_compose__message()`, `agent_xml_compose__mobile_agent()`, `agent_xml_compose__task()`, `agent_xml_compose__tasks()`, `fipa_envelope_Compose()`, `fipa_envelope_Compose__envelope()`, `fipa_envelope_Compose__params()`, `mxml_new()`, `mxmlAdd()`, `scan_file()`, and `sort_node()`.

### 5.70.1.2 #define MXML\_ADD\_BEFORE 0

Definition at line 67 of file `mxml.h`.

Referenced by `mxmlAdd()`, `scan_file()`, and `sort_node()`.

### 5.70.1.3 #define MXML\_ADD\_TO\_PARENT NULL

Definition at line 69 of file `mxml.h`.

Referenced by `add_variable()`, `agent_xml_compose()`, `agent_xml_compose__create_row_nodes()`, `fipa_envelope_Compose()`, `fipa_envelope_Compose__envelope()`, `fipa_envelope_Compose__params()`, `mxml_new()`, `scan_file()`, and `sort_node()`.

### 5.70.1.4 #define MXML\_DESCEND 1

Definition at line 58 of file `mxml.h`.

Referenced by `agent_xml_parse__tasks()`, `main()`, `MC_LoadAgentFromFile()`, `message_xml_parse()`, `mxmlFindElement()`, `mxmlIndexNew()`, and `write_element()`.

### 5.70.1.5 #define MXML\_DESCEND\_FIRST -1

Definition at line 60 of file `mxml.h`.

Referenced by `agent_xml_parse__fill_row_data()`, `agent_xml_parse__task()`, `agent_xml_parse__tasks()`, `fipa_envelope_HandleAclRepresentation()`, `fipa_envelope_HandleComments()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandleEnvelope()`, `fipa_envelope_HandleFrom()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleParams()`, `fipa_envelope_HandlePayloadEncoding()`, `fipa_envelope_HandlePayloadLength()`, `fipa_envelope_HandleReceived()`, `fipa_envelope_HandleTo()`, `fipa_envelope_ParseAddresses()`, `fipa_envelope_ParseAgentIdentifier()`, `fipa_envelope_ParseResolvers()`, `scan_file()`, `sort_node()`, and `write_documentation()`.

#### 5.70.1.6 `#define MXML_INTEGER_CALLBACK mxml_integer_cb`

Definition at line 48 of file `mxml.h`.

Referenced by `main()`.

#### 5.70.1.7 `#define MXML_NO_CALLBACK 0`

Definition at line 47 of file `mxml.h`.

Referenced by `acc_connection_Thread()`, `agent_xml_compose()`, `fipa_envelope_Compose()`, `fipa_envelope_Parse()`, `main()`, `message_InitializeFromAgent()`, `message_InitializeFromConnection()`, and `message_InitializeFromString()`.

#### 5.70.1.8 `#define MXML_NO_DESCEND 0`

Definition at line 59 of file `mxml.h`.

Referenced by `agent_xml_parse__task()`, `agent_xml_parse__tasks()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleTo()`, `fipa_envelope_ParseAddresses()`, `fipa_envelope_ParseResolvers()`, `main()`, `message_xml_parse()`, `write_documentation()`, `write_element()`, `xml_find_sibling()`, and `xml_get_deep_child()`.

#### 5.70.1.9 `#define MXML_NO_PARENT 0`

Definition at line 56 of file `mxml.h`.

Referenced by `agent_xml_compose__agent_code()`, `agent_xml_compose__create_row_nodes()`, `main()`, and `scan_file()`.

#### 5.70.1.10 `#define MXML_OPAQUE_CALLBACK mxml_opaque_cb`

Definition at line 50 of file `mxml.h`.

Referenced by `main()`.

#### 5.70.1.11 `#define MXML_REAL_CALLBACK mxml_real_cb`

Definition at line 52 of file `mxml.h`.

Referenced by `main()`.

**5.70.1.12 #define MXML\_TAB 8**

Definition at line 45 of file mxml.h.

Referenced by mxml\_write\_ws().

**5.70.1.13 #define MXML\_TEXT\_CALLBACK 0**

Definition at line 54 of file mxml.h.

**5.70.1.14 #define MXML\_WRAP 72**

Definition at line 44 of file mxml.h.

Referenced by mxml\_write\_node().

**5.70.1.15 #define MXML\_WS\_AFTER\_CLOSE 3**

Definition at line 65 of file mxml.h.

Referenced by mxml\_write\_node(), whitespace\_cb(), and ws\_cb().

**5.70.1.16 #define MXML\_WS\_AFTER\_OPEN 1**

Definition at line 63 of file mxml.h.

Referenced by mxml\_write\_node(), whitespace\_cb(), and ws\_cb().

**5.70.1.17 #define MXML\_WS\_BEFORE\_CLOSE 2**

Definition at line 64 of file mxml.h.

Referenced by mxml\_write\_node(), whitespace\_cb(), and ws\_cb().

**5.70.1.18 #define MXML\_WS\_BEFORE\_OPEN 0**

Definition at line 62 of file mxml.h.

Referenced by mxml\_write\_node(), whitespace\_cb(), and ws\_cb().

**5.70.2 Typedef Documentation****5.70.2.1 typedef struct mxml\_attr\_s mxml\_attr\_t****5.70.2.2 typedef int(\* mxml\_custom\_load\_cb\_t)(mxml\_node\_t \*, const char \*)**

Definition at line 142 of file mxml.h.

**5.70.2.3 typedef char\*( \* mxml\_custom\_save\_cb\_t)(mxml\_node\_t \*)**

Definition at line 145 of file mxml.h.



**5.70.2.4** typedef struct mxml\_custom\_s mxml\_custom\_t

**5.70.2.5** typedef struct mxml\_value\_s mxml\_element\_t

**5.70.2.6** typedef struct mxml\_index\_s mxml\_index\_t

**5.70.2.7** typedef struct mxml\_node\_s mxml\_node\_t

**5.70.2.8** typedef struct mxml\_text\_s mxml\_text\_t

**5.70.2.9** typedef enum mxml\_type\_e mxml\_type\_t

**5.70.2.10** typedef union mxml\_value\_u mxml\_value\_t

### 5.70.3 Enumeration Type Documentation

**5.70.3.1** enum mxml\_type\_e

Enumerator:

*MXML\_ELEMENT*

*MXML\_INTEGER*

*MXML\_OPAQUE*

*MXML\_REAL*

*MXML\_TEXT*

*MXML\_CUSTOM*

Definition at line 76 of file mxml.h.

### 5.70.4 Function Documentation

**5.70.4.1** void mxml\_error (const char \* *format*, ...)

Definition at line 53 of file mxml-private.c.

References mxml\_error\_cb, and mxml\_strdup().

Referenced by mxml\_add\_char(), mxml\_fd\_getc(), mxml\_file\_getc(), mxml\_get\_entity(), mxml\_load\_data(), mxml\_parse\_element(), mxml\_string\_getc(), mxmlElementSetAttr(), mxmlEntityAddCallback(), and mxmlIndexNew().

**5.70.4.2** mxml\_type\_t mxml\_integer\_cb (mxml\_node\_t \* *node*)

Definition at line 99 of file mxml-private.c.

References MXML\_INTEGER.

**5.70.4.3** mxml\_type\_t mxml\_opaque\_cb (mxml\_node\_t \* *node*)

Definition at line 112 of file mxml-private.c.

References MXML\_OPAQUE.

**5.70.4.4 mxmxml\_type\_t mxmxml\_real\_cb (mxml\_node\_t \* node)**

Definition at line 125 of file mxml-private.c.

References MXML\_REAL.

**5.70.4.5 void mxmlAdd (mxml\_node\_t \* parent, int where, mxml\_node\_t \* child, mxml\_node\_t \* node)**

Definition at line 66 of file mxml-node.c.

References mxml\_node\_s::child, mxml\_node\_s::last\_child, MXML\_ADD\_AFTER, MXML\_ADD\_BEFORE, mxmlRemove(), mxml\_node\_s::next, mxml\_node\_s::parent, and mxml\_node\_s::prev.

Referenced by add\_variable(), agent\_xml\_compose(), agent\_xml\_compose\_\_agent\_data(), agent\_xml\_compose\_\_create\_row\_nodes(), agent\_xml\_compose\_\_data(), agent\_xml\_compose\_\_gaf\_message(), agent\_xml\_compose\_\_message(), agent\_xml\_compose\_\_mobile\_agent(), agent\_xml\_compose\_\_task(), agent\_xml\_compose\_\_tasks(), fipa\_envelope\_Compose(), fipa\_envelope\_Compose\_\_envelope(), fipa\_envelope\_Compose\_\_params(), mxml\_new(), scan\_file(), and sort\_node().

**5.70.4.6 void mxmlDelete (mxml\_node\_t \* node)**

Definition at line 200 of file mxml-node.c.

References mxml\_value\_s::attrs, mxml\_node\_s::child, mxml\_value\_u::custom, mxml\_custom\_s::data, mxml\_custom\_s::destroy, mxml\_value\_u::element, MXML\_CUSTOM, MXML\_ELEMENT, MXML\_INTEGER, MXML\_OPAQUE, MXML\_REAL, MXML\_TEXT, mxmlDelete(), mxmlRemove(), mxml\_attr\_s::name, mxml\_value\_s::name, mxml\_value\_s::num\_attrs, mxml\_value\_u::opaque, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, mxml\_attr\_s::value, and mxml\_node\_s::value.

Referenced by add\_variable(), agent\_datastate\_Destroy(), fipa\_envelope\_Compose(), fipa\_envelope\_Parse(), main(), message\_Destroy(), mxml\_load\_data(), mxmlDelete(), scan\_file(), and sort\_node().

**5.70.4.7 const char\* mxmlElementGetAttr (mxml\_node\_t \* node, const char \* name)**

Definition at line 48 of file mxml-attr.c.

References mxml\_value\_s::attrs, mxml\_value\_u::element, MXML\_ELEMENT, mxml\_attr\_s::name, mxml\_value\_s::num\_attrs, mxml\_node\_s::type, mxml\_attr\_s::value, and mxml\_node\_s::value.

Referenced by agent\_xml\_parse\_\_agent\_code(), agent\_xml\_parse\_\_data(), agent\_xml\_parse\_\_task(), agent\_xml\_parse\_\_tasks(), fipa\_envelope\_HandleReceived(), index\_compare(), index\_find(), message\_xml\_parse\_\_message(), mxml\_parse\_element(), mxmlFindElement(), mxmlIndexNew(), scan\_file(), sort\_node(), type\_cb(), and write\_documentation().

**5.70.4.8 void mxmlElementSetAttr (mxml\_node\_t \* node, const char \* name, const char \* value)**

Definition at line 95 of file mxml-attr.c.

References mxml\_value\_s::attrs, mxml\_value\_u::element, MXML\_ELEMENT, mxml\_error(), mxml\_value\_s::name, mxml\_attr\_s::name, mxml\_value\_s::num\_attrs, strdup, mxml\_node\_s::type, mxml\_attr\_s::value, and mxml\_node\_s::value.

Referenced by add\_variable(), agent\_xml\_compose\_\_agent\_code(), agent\_xml\_compose\_\_create\_row\_nodes(), agent\_xml\_compose\_\_data(), agent\_xml\_compose\_\_message(), agent\_xml\_compose\_\_task(),

agent\_xml\_compose\_\_tasks(), fipa\_envelope\_Compose\_\_params(), main(), mxml\_parse\_element(), scan\_file(), sort\_node(), and update\_comment().

#### 5.70.4.9 int mxmlEntityAddCallback (int(\*) (const char \*name) cb)

Definition at line 69 of file mxml-entity.c.

References callbacks, mxml\_error(), and num\_callbacks.

#### 5.70.4.10 const char\* mxmlEntityGetName (int val)

Definition at line 95 of file mxml-entity.c.

Referenced by mxml\_write\_name(), and mxml\_write\_string().

#### 5.70.4.11 int mxmlEntityGetValue (const char \* name)

Definition at line 125 of file mxml-entity.c.

References callbacks, and num\_callbacks.

Referenced by mxml\_get\_entity().

#### 5.70.4.12 void mxmlEntityRemoveCallback (int(\*) (const char \*name) cb)

Definition at line 144 of file mxml-entity.c.

References callbacks, and num\_callbacks.

#### 5.70.4.13 mxml\_node\_t\* mxmlFindElement (mxml\_node\_t \* node, mxml\_node\_t \* top, const char \* name, const char \* attr, const char \* value, int descend)

Definition at line 55 of file mxml-search.c.

References mxml\_value\_u::element, MXML\_DESCEND, MXML\_ELEMENT, mxmlElementGetAttr(), mxmlWalkNext(), mxml\_value\_s::name, mxml\_node\_s::next, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by agent\_xml\_parse\_\_fill\_row\_data(), agent\_xml\_parse\_\_task(), agent\_xml\_parse\_\_tasks(), fipa\_envelope\_HandleAclRepresentation(), fipa\_envelope\_HandleComments(), fipa\_envelope\_HandleDate(), fipa\_envelope\_HandleEnvelope(), fipa\_envelope\_HandleFrom(), fipa\_envelope\_HandleIntendedReceiver(), fipa\_envelope\_HandleParams(), fipa\_envelope\_HandlePayloadEncoding(), fipa\_envelope\_HandlePayloadLength(), fipa\_envelope\_HandleReceived(), fipa\_envelope\_HandleTo(), fipa\_envelope\_ParseAddresses(), fipa\_envelope\_ParseAgentIdentifier(), fipa\_envelope\_ParseResolvers(), main(), MC\_LoadAgentFromFile(), message\_xml\_parse(), mxmlIndexNew(), scan\_file(), sort\_node(), write\_documentation(), write\_element(), xml\_find\_sibling(), and xml\_get\_child().

#### 5.70.4.14 void mxmlIndexDelete (mxml\_index\_t \* ind)

Definition at line 62 of file mxml-index.c.

References mxml\_index\_s::alloc\_nodes, mxml\_index\_s::attr, and mxml\_index\_s::nodes.

Referenced by main(), and mxmlIndexNew().

**5.70.4.15 mxml\_node\_t\* mxmlIndexEnum (mxml\_index\_t \* ind)**

Definition at line 92 of file mxml-index.c.

References mxml\_index\_s::cur\_node, mxml\_index\_s::nodes, and mxml\_index\_s::num\_nodes.

Referenced by main(), and mxmlIndexFind().

**5.70.4.16 mxml\_node\_t\* mxmlIndexFind (mxml\_index\_t \* ind, const char \* element, const char \* value)**

Definition at line 122 of file mxml-index.c.

References mxml\_index\_s::attr, mxml\_index\_s::cur\_node, index\_find(), mxmlIndexEnum(), mxml\_index\_s::nodes, and mxml\_index\_s::num\_nodes.

Referenced by main().

**5.70.4.17 mxml\_index\_t\* mxmlIndexNew (mxml\_node\_t \* node, const char \* element, const char \* attr)**

Definition at line 305 of file mxml-index.c.

References mxml\_index\_s::alloc\_nodes, mxml\_index\_s::attr, mxml\_value\_u::element, index\_sort(), MXML\_DESCEND, mxml\_error(), mxmlElementGetAttr(), mxmlFindElement(), mxmlIndexDelete(), mxml\_value\_s::name, mxml\_index\_s::nodes, mxml\_index\_s::num\_nodes, strdup, and mxml\_node\_s::value.

Referenced by main().

**5.70.4.18 mxml\_node\_t\* mxmlIndexReset (mxml\_index\_t \* ind)**

Definition at line 463 of file mxml-index.c.

References mxml\_index\_s::cur\_node, mxml\_index\_s::nodes, and mxml\_index\_s::num\_nodes.

Referenced by main().

**5.70.4.19 mxml\_node\_t\* mxmlLoadFd (mxml\_node\_t \* top, int fd, mxml\_type\_t(\*) (mxml\_node\_t \*) cb)****5.70.4.20 mxml\_node\_t\* mxmlLoadFile (mxml\_node\_t \* top, FILE \* fp, mxml\_type\_t(\*) (mxml\_node\_t \*) cb)****5.70.4.21 mxml\_node\_t\* mxmlLoadString (mxml\_node\_t \* top, const char \* s, mxml\_type\_t(\*) (mxml\_node\_t \*) cb)****5.70.4.22 mxml\_node\_t\* mxmlNewCustom (mxml\_node\_t \* parent, void \* data, void(\*) (void \*) destroy)**

Definition at line 291 of file mxml-node.c.

References mxml\_value\_u::custom, mxml\_custom\_s::data, mxml\_custom\_s::destroy, MXML\_CUSTOM, mxml\_new(), node, and mxml\_node\_s::value.

Referenced by mxml\_load\_data().

**5.70.4.23 mxml\_node\_t\* mxmlNewElement (mxml\_node\_t \* *parent*, const char \* *name*)**

Definition at line 327 of file mxml-node.c.

References mxml\_value\_u::element, MXML\_ELEMENT, mxml\_new(), mxml\_value\_s::name, node, strdup, and mxml\_node\_s::value.

Referenced by add\_variable(), agent\_xml\_compose\_\_agent\_code(), agent\_xml\_compose\_\_agent\_data(), agent\_xml\_compose\_\_create\_row\_nodes(), agent\_xml\_compose\_\_data(), agent\_xml\_compose\_\_gaf\_message(), agent\_xml\_compose\_\_home(), agent\_xml\_compose\_\_message(), agent\_xml\_compose\_\_mobile\_agent(), agent\_xml\_compose\_\_name(), agent\_xml\_compose\_\_owner(), agent\_xml\_compose\_\_task(), agent\_xml\_compose\_\_tasks(), agent\_xml\_compose\_\_wg\_code(), fipa\_envelope\_Compose\_\_acl\_representation(), fipa\_envelope\_Compose\_\_date(), fipa\_envelope\_Compose\_\_envelope(), fipa\_envelope\_Compose\_\_from(), fipa\_envelope\_Compose\_\_intended\_receiver(), fipa\_envelope\_Compose\_\_params(), fipa\_envelope\_Compose\_\_payload\_encoding(), fipa\_envelope\_Compose\_\_to(), main(), mxml\_load\_data(), scan\_file(), and xml\_new\_cdata().

**5.70.4.24 mxml\_node\_t\* mxmlNewInteger (mxml\_node\_t \* *parent*, int *integer*)**

Definition at line 365 of file mxml-node.c.

References mxml\_value\_u::integer, MXML\_INTEGER, mxml\_new(), node, and mxml\_node\_s::value.

Referenced by main(), and mxml\_load\_data().

**5.70.4.25 mxml\_node\_t\* mxmlNewOpaque (mxml\_node\_t \* *parent*, const char \* *opaque*)**

Definition at line 396 of file mxml-node.c.

References mxml\_new(), MXML\_OPAQUE, node, mxml\_value\_u::opaque, strdup, and mxml\_node\_s::value.

Referenced by main(), and mxml\_load\_data().

**5.70.4.26 mxml\_node\_t\* mxmlNewReal (mxml\_node\_t \* *parent*, double *real*)**

Definition at line 434 of file mxml-node.c.

References mxml\_new(), MXML\_REAL, node, mxml\_value\_u::real, and mxml\_node\_s::value.

Referenced by main(), and mxml\_load\_data().

**5.70.4.27 mxml\_node\_t\* mxmlNewText (mxml\_node\_t \* *parent*, int *whitespace*, const char \* *string*)**

Definition at line 466 of file mxml-node.c.

References mxml\_new(), MXML\_TEXT, node, strdup, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::value, and mxml\_text\_s::whitespace.

Referenced by agent\_xml\_compose\_\_create\_row\_nodes(), agent\_xml\_compose\_\_home(), agent\_xml\_compose\_\_name(), agent\_xml\_compose\_\_owner(), agent\_xml\_compose\_\_wg\_code(), fipa\_envelope\_Compose\_\_acl\_representation(), fipa\_envelope\_Compose\_\_date(), fipa\_envelope\_Compose\_\_from(), fipa\_envelope\_Compose\_\_intended\_receiver(), fipa\_envelope\_Compose\_\_payload\_encoding(), fipa\_envelope\_Compose\_\_to(), main(), mxml\_load\_data(), and scan\_file().

**5.70.4.28** `mxml_node_t* mxmlNewTextf (mxml_node_t * parent, int whitespace, const char * format, ...)`

Definition at line 510 of file `mxml-node.c`.

References `mxml_new()`, `mxml_strdupf()`, `MXML_TEXT`, `node`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::value`, and `mxml_text_s::whitespace`.

**5.70.4.29** `void mxmlRemove (mxml_node_t * node)`

Definition at line 557 of file `mxml-node.c`.

References `mxml_node_s::child`, `mxml_node_s::last_child`, `mxml_node_s::next`, `mxml_node_s::parent`, and `mxml_node_s::prev`.

Referenced by `mxmlAdd()`, and `mxmlDelete()`.

**5.70.4.30** `char* mxmlSaveAllocString (mxml_node_t * node, const char *(*)(mxml_node_t *, int) cb)`

**5.70.4.31** `int mxmlSaveFd (mxml_node_t * node, int fd, const char *(*)(mxml_node_t *, int) cb)`

**5.70.4.32** `int mxmlSaveFile (mxml_node_t * node, FILE * fp, const char *(*)(mxml_node_t *, int) cb)`

**5.70.4.33** `int mxmlSaveString (mxml_node_t * node, char * buffer, int bufsize, const char *(*)(mxml_node_t *, int) cb)`

**5.70.4.34** `int mxmlSetCustom (mxml_node_t * node, void * data, void (*)(void *) destroy)`

Definition at line 50 of file `mxml-set.c`.

References `mxml_value_u::custom`, `mxml_custom_s::data`, `mxml_custom_s::destroy`, `MXML_CUSTOM`, `mxml_node_s::type`, and `mxml_node_s::value`.

**5.70.4.35** `void mxmlSetCustomHandlers (mxml_custom_load_cb_t load, mxml_custom_save_cb_t save)`

Definition at line 457 of file `mxml-file.c`.

References `mxml_custom_load_cb`, and `mxml_custom_save_cb`.

**5.70.4.36** `int mxmlSetElement (mxml_node_t * node, const char * name)`

Definition at line 83 of file `mxml-set.c`.

References `mxml_value_u::element`, `MXML_ELEMENT`, `mxml_value_s::name`, `strdup`, `mxml_node_s::type`, and `mxml_node_s::value`.

**5.70.4.37** `void mxmlSetErrorCallback (void (*)(const char *) cb)`

Definition at line 472 of file `mxml-file.c`.

References `mxml_error_cb`.

**5.70.4.38 int mxmlSetInteger (mxml\_node\_t \* node, int integer)**

Definition at line 113 of file mxml-set.c.

References mxml\_value\_u::integer, MXML\_INTEGER, mxml\_node\_s::type, and mxml\_node\_s::value.

**5.70.4.39 int mxmlSetOpaque (mxml\_node\_t \* node, const char \* opaque)**

Definition at line 140 of file mxml-set.c.

References MXML\_OPAQUE, mxml\_value\_u::opaque, strdup, mxml\_node\_s::type, and mxml\_node\_s::value.

**5.70.4.40 int mxmlSetReal (mxml\_node\_t \* node, double real)**

Definition at line 170 of file mxml-set.c.

References MXML\_REAL, mxml\_value\_u::real, mxml\_node\_s::type, and mxml\_node\_s::value.

**5.70.4.41 int mxmlSetText (mxml\_node\_t \* node, int whitespace, const char \* string)**

Definition at line 197 of file mxml-set.c.

References MXML\_TEXT, strdup, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, mxml\_node\_s::value, and mxml\_text\_s::whitespace.

**5.70.4.42 int mxmlSetTextf (mxml\_node\_t \* node, int whitespace, const char \* format, ...)**

Definition at line 229 of file mxml-set.c.

References mxml\_strdupf(), MXML\_TEXT, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, mxml\_node\_s::value, and mxml\_text\_s::whitespace.

**5.70.4.43 mxml\_node\_t\* mxmlWalkNext (mxml\_node\_t \* node, mxml\_node\_t \* top, int descend)**

Definition at line 137 of file mxml-search.c.

References mxml\_node\_s::child, mxml\_node\_s::next, and mxml\_node\_s::parent.

Referenced by mxmlFindElement(), and write\_element().

**5.70.4.44 mxml\_node\_t\* mxmlWalkPrev (mxml\_node\_t \* node, mxml\_node\_t \* top, int descend)**

Definition at line 173 of file mxml-search.c.

References mxml\_node\_s::last\_child, mxml\_node\_s::parent, and mxml\_node\_s::prev.

## 5.71 /home/dko/Projects/mobilec/src/mxml-2.2.2/mxmldoc.c File Reference

```
#include "config.h"
#include "mxml.h"
```

### Defines

- `#define STATE_NONE` 0
- `#define STATE_PREPROCESSOR` 1
- `#define STATE_C_COMMENT` 2
- `#define STATE_CXX_COMMENT` 3
- `#define STATE_STRING` 4
- `#define STATE_CHARACTER` 5
- `#define STATE_IDENTIFIER` 6

### Functions

- static `mxml_node_t` \* `add_variable` (`mxml_node_t` \*parent, const char \*name, `mxml_node_t` \*type)
- static void `safe_strcpy` (char \*dst, const char \*src)
- static int `scan_file` (const char \*filename, FILE \*fp, `mxml_node_t` \*doc)
- static void `sort_node` (`mxml_node_t` \*tree, `mxml_node_t` \*func)
- static void `update_comment` (`mxml_node_t` \*parent, `mxml_node_t` \*comment)
- static void `write_documentation` (`mxml_node_t` \*doc)
- static void `write_element` (`mxml_node_t` \*doc, `mxml_node_t` \*element)
- static void `write_string` (const char \*s)
- static const char \* `ws_cb` (`mxml_node_t` \*node, int where)
- int `main` (int argc, char \*argv[ ])

#### 5.71.1 Define Documentation

##### 5.71.1.1 `#define STATE_C_COMMENT` 2

Definition at line 129 of file `mxmldoc.c`.

Referenced by `scan_file()`.

##### 5.71.1.2 `#define STATE_CHARACTER` 5

Definition at line 132 of file `mxmldoc.c`.

Referenced by `scan_file()`.

##### 5.71.1.3 `#define STATE_CXX_COMMENT` 3

Definition at line 130 of file `mxmldoc.c`.

Referenced by `scan_file()`.



#### 5.71.1.4 **#define STATE\_IDENTIFIER 6**

Definition at line 133 of file mxmldoc.c.

Referenced by scan\_file().

#### 5.71.1.5 **#define STATE\_NONE 0**

Definition at line 127 of file mxmldoc.c.

Referenced by scan\_file().

#### 5.71.1.6 **#define STATE\_PREPROCESSOR 1**

Definition at line 128 of file mxmldoc.c.

Referenced by scan\_file().

#### 5.71.1.7 **#define STATE\_STRING 4**

Definition at line 131 of file mxmldoc.c.

Referenced by scan\_file().

### 5.71.2 **Function Documentation**

#### 5.71.2.1 **static mxml\_node\_t \* add\_variable (mxml\_node\_t \* *parent*, const char \* *name*, mxml\_node\_t \* *type*) [static]**

Definition at line 313 of file mxmldoc.c.

References mxml\_node\_s::child, mxml\_node\_s::last\_child, MXML\_ADD\_AFTER, MXML\_ADD\_TO\_PARENT, mxmlAdd(), mxmlDelete(), mxmlElementSetAttr(), mxmlNewElement(), mxml\_node\_s::next, node, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::value, and mxml\_text\_s::whitespace.

Referenced by scan\_file().

#### 5.71.2.2 **int main (int *argc*, char \* *argv*[])**

Definition at line 159 of file mxmldoc.c.

References MXML\_DESCEND, MXML\_NO\_CALLBACK, mxmlDelete(), mxmlElementSetAttr(), mxmlFindElement(), mxmlLoadFile(), mxmlNewElement(), mxmlSaveFile(), scan\_file(), write\_documentation(), and ws\_cb().

#### 5.71.2.3 **static void safe\_strcpy (char \* *dst*, const char \* *src*) [static]**

Definition at line 427 of file mxmldoc.c.

Referenced by update\_comment().

**5.71.2.4 static int scan\_file (const char \*filename, FILE \*fp, mxml\_node\_t \*doc) [static]**

Definition at line 442 of file mxmldoc.c.

References add\_variable(), mxml\_node\_s::child, mxml\_value\_u::element, mxml\_node\_s::last\_child, MXML\_ADD\_AFTER, MXML\_ADD\_BEFORE, MXML\_ADD\_TO\_PARENT, MXML\_DESCEND\_FIRST, MXML\_NO\_PARENT, mxmlAdd(), mxmlDelete(), mxmlElementGetAttr(), mxmlElementSetAttr(), mxmlFindElement(), mxmlNewElement(), mxmlNewText(), mxml\_value\_s::name, mxml\_node\_s::next, node, sort\_node(), STATE\_C\_COMMENT, STATE\_CHARACTER, STATE\_CXX\_COMMENT, STATE\_IDENTIFIER, STATE\_NONE, STATE\_PREPROCESSOR, STATE\_STRING, mxml\_text\_s::string, mxml\_value\_u::text, update\_comment(), mxml\_node\_s::value, and mxml\_text\_s::whitespace.

Referenced by main().

**5.71.2.5 static void sort\_node (mxml\_node\_t \*tree, mxml\_node\_t \*func) [static]**

Definition at line 1617 of file mxmldoc.c.

References mxml\_node\_s::child, mxml\_value\_u::element, MXML\_ADD\_AFTER, MXML\_ADD\_BEFORE, MXML\_ADD\_TO\_PARENT, MXML\_DESCEND\_FIRST, mxmlAdd(), mxmlDelete(), mxmlElementGetAttr(), mxmlElementSetAttr(), mxmlFindElement(), mxml\_value\_s::name, mxml\_node\_s::next, mxml\_node\_s::parent, and mxml\_node\_s::value.

Referenced by scan\_file().

**5.71.2.6 static void update\_comment (mxml\_node\_t \*parent, mxml\_node\_t \*comment) [static]**

Definition at line 1705 of file mxmldoc.c.

References mxml\_value\_u::element, mxmlElementSetAttr(), mxml\_value\_s::name, safe\_strcpy(), mxml\_text\_s::string, mxml\_value\_u::text, and mxml\_node\_s::value.

Referenced by scan\_file().

**5.71.2.7 static void write\_documentation (mxml\_node\_t \*doc) [static]**

Definition at line 1805 of file mxmldoc.c.

References mxml\_node\_s::child, MXML\_DESCEND\_FIRST, MXML\_NO\_DESCEND, MXML\_VERSION, mxmlElementGetAttr(), mxmlFindElement(), and write\_element().

Referenced by main().

**5.71.2.8 static void write\_element (mxml\_node\_t \*doc, mxml\_node\_t \*element) [static]**

Definition at line 2615 of file mxmldoc.c.

References mxml\_node\_s::child, MXML\_DESCEND, MXML\_NO\_DESCEND, MXML\_TEXT, mxmlFindElement(), mxmlWalkNext(), node, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, mxml\_node\_s::value, mxml\_text\_s::whitespace, and write\_string().

Referenced by write\_documentation().

**5.71.2.9 static void write\_string (const char \* *s*) [static]**

Definition at line 2660 of file mxmldoc.c.

Referenced by write\_element().

**5.71.2.10 static const char \* ws\_cb (mxml\_node\_t \* *node*, int *where*) [static]**

Definition at line 2718 of file mxmldoc.c.

References mxml\_value\_u::element, MXML\_WS\_AFTER\_CLOSE, MXML\_WS\_AFTER\_OPEN, MXML\_WS\_BEFORE\_CLOSE, MXML\_WS\_BEFORE\_OPEN, mxml\_value\_s::name, mxml\_node\_s::parent, and mxml\_node\_s::value.

Referenced by main().

## 5.72 /home/dko/Projects/mobilec/src/mxml-2.2.2/test/class.cxx File Reference

### Data Structures

- class `foo_c`

### Variables

- `foo_c f`
- `foo_c foo`
- `bar = b`

### 5.72.1 Variable Documentation

#### 5.72.1.1 `bar = b`

Definition at line 77 of file class.cxx.

#### 5.72.1.2 `foo_c f`

Referenced by `foo_float_function()`, `main()`, `MC_Initialize()`, `md5_check()`, `md5_file()`, `read_encrypted_file()`, `sha1_check()`, `sha1_file()`, `sha2_check()`, `sha2_file()`, `sha4_file()`, `x509parse_crfile()`, and `x509parse_keyfile()`.

#### 5.72.1.3 `foo_c foo`

## 5.73 /home/dko/Projects/mobilec/src/mxml-2.2.2/test/enum.cxx File Reference

### Typedefs

- typedef enum [foo\\_enum\\_e](#) [foo\\_enum\\_t](#)

### Enumerations

- enum [foo\\_enum\\_e](#) { [FOO\\_ONE](#), [FOO\\_TWO](#), [FOO\\_RED](#), [FOO\\_BLUE](#) }

#### 5.73.1 Typedef Documentation

##### 5.73.1.1 typedef enum [foo\\_enum\\_e](#) [foo\\_enum\\_t](#)

#### 5.73.2 Enumeration Type Documentation

##### 5.73.2.1 enum [foo\\_enum\\_e](#)

Enumerator:

*[FOO\\_ONE](#)*  
*[FOO\\_TWO](#)*  
*[FOO\\_RED](#)*  
*[FOO\\_BLUE](#)*

Definition at line 1 of file enum.cxx.

## 5.74 /home/dko/Projects/mobilec/src/mxml-2.2.2/test/function.cxx

### File Reference

#### Functions

- void `foo_void_function` (`int` *one*, `float` \**two*, `const char` \**three*)
- float `foo_float_function` (`int` *one*, `const char` \**two*)
- int `foo_default_string` (`int` *one*, `const char` \**two*="2")
- int `foo_default_int` (`int` *one*, `int` *two*=2)

#### 5.74.1 Function Documentation

##### 5.74.1.1 int `foo_default_int` (`int` *one*, `int` *two* = 2)

Definition at line 65 of file `function.cxx`.

##### 5.74.1.2 int `foo_default_string` (`int` *one*, `const char` \* *two* = "2")

Definition at line 45 of file `function.cxx`.

##### 5.74.1.3 float `foo_float_function` (`int` *one*, `const char` \* *two*)

Definition at line 26 of file `function.cxx`.

References `f`.

##### 5.74.1.4 void `foo_void_function` (`int` *one*, `float` \* *two*, `const char` \* *three*)

Definition at line 6 of file `function.cxx`.

## 5.75 /home/dko/Projects/mobilec/src/mxml-2.2.2/test/struct.cxx File Reference

### Data Structures

- struct [foo\\_s](#)

### Typedefs

- typedef struct [foo\\_s](#) [foo\\_t](#)

### 5.75.1 Typedef Documentation

#### 5.75.1.1 typedef struct [foo\\_s](#) [foo\\_t](#)

## 5.76 /home/dko/Projects/mobilec/src/mxml-2.2.2/testmxml.c File Reference

```
#include "mxml.h"
#include <unistd.h>
#include "config.h"
#include <fcntl.h>
```

### Functions

- [mxml\\_type\\_t type\\_cb \(mxml\\_node\\_t \\*node\)](#)
- [const char \\* whitespace\\_cb \(mxml\\_node\\_t \\*node, int where\)](#)
- [int main \(int argc, char \\*argv\[ \]\)](#)

### 5.76.1 Function Documentation

#### 5.76.1.1 int main (int argc, char \* argv[ ])

Definition at line 60 of file testmxml.c.

References [mxml\\_node\\_s::child](#), [mxml\\_value\\_u::element](#), [f](#), [mxml\\_value\\_u::integer](#), [mxml\\_node\\_s::last\\_child](#), [MXML\\_DESCEND](#), [MXML\\_ELEMENT](#), [MXML\\_INTEGER](#), [MXML\\_INTEGER\\_CALLBACK](#), [MXML\\_NO\\_CALLBACK](#), [MXML\\_NO\\_DESCEND](#), [MXML\\_NO\\_PARENT](#), [MXML\\_OPAQUE](#), [MXML\\_OPAQUE\\_CALLBACK](#), [MXML\\_REAL](#), [MXML\\_REAL\\_CALLBACK](#), [MXML\\_TEXT](#), [mxmlDelete\(\)](#), [mxmlFindElement\(\)](#), [mxmlIndexDelete\(\)](#), [mxmlIndexEnum\(\)](#), [mxmlIndexFind\(\)](#), [mxmlIndexNew\(\)](#), [mxmlIndexReset\(\)](#), [mxmlLoadFd\(\)](#), [mxmlLoadFile\(\)](#), [mxmlLoadString\(\)](#), [mxmlNewElement\(\)](#), [mxmlNewInteger\(\)](#), [mxmlNewOpaque\(\)](#), [mxmlNewReal\(\)](#), [mxmlNewText\(\)](#), [mxmlSaveFd\(\)](#), [mxmlSaveFile\(\)](#), [mxmlSaveString\(\)](#), [mxml\\_value\\_s::name](#), [mxml\\_node\\_s::next](#), [node](#), [mxml\\_index\\_s::num\\_nodes](#), [mxml\\_value\\_u::opaque](#), [mxml\\_value\\_u::real](#), [mxml\\_text\\_s::string](#), [mxml\\_value\\_u::text](#), [mxml\\_node\\_s::type](#), [type\\_cb\(\)](#), [mxml\\_node\\_s::value](#), [mxml\\_text\\_s::whitespace](#), and [whitespace\\_cb\(\)](#).

#### 5.76.1.2 mxml\_type\_t type\_cb (mxml\_node\_t \* node)

Definition at line 543 of file testmxml.c.

References [mxml\\_value\\_u::element](#), [MXML\\_INTEGER](#), [MXML\\_OPAQUE](#), [MXML\\_REAL](#), [MXML\\_TEXT](#), [mxmlElementGetAttr\(\)](#), [mxml\\_value\\_s::name](#), and [mxml\\_node\\_s::value](#).

Referenced by [main\(\)](#).

#### 5.76.1.3 const char\* whitespace\_cb (mxml\_node\_t \* node, int where)

Definition at line 572 of file testmxml.c.

References [mxml\\_node\\_s::child](#), [mxml\\_value\\_u::element](#), [MXML\\_WS\\_AFTER\\_CLOSE](#), [MXML\\_WS\\_AFTER\\_OPEN](#), [MXML\\_WS\\_BEFORE\\_CLOSE](#), [MXML\\_WS\\_BEFORE\\_OPEN](#), [mxml\\_value\\_s::name](#), [mxml\\_node\\_s::parent](#), and [mxml\\_node\\_s::value](#).



## 5.77 /home/dko/Projects/mobilec/src/mxml-2.2.2/winconfig.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdarg.h>
#include <ctype.h>
```

### Defines

- #define [MXML\\_VERSION](#) "Mini-XML v2.2.2"
- #define [HAVE\\_VSNPRINTF](#) 1
- #define [HAVE\\_STRDUP](#) 1
- #define [strdup](#) mxml\_strdup
- #define [vsnprintf](#) mxml\_vsnprintf

### Functions

- char \* [mxml\\_strdup](#) (const char \*)
- char \* [mxml\\_strdupf](#) (const char \*, va\_list)
- int [mxml\\_vsnprintf](#) (char \*, size\_t, const char \*, va\_list)

#### 5.77.1 Define Documentation

##### 5.77.1.1 #define HAVE\_STRDUP 1

Definition at line 49 of file winconfig.h.

##### 5.77.1.2 #define HAVE\_VSNPRINTF 1

Definition at line 42 of file winconfig.h.

##### 5.77.1.3 #define MXML\_VERSION "Mini-XML v2.2.2"

Definition at line 35 of file winconfig.h.

##### 5.77.1.4 #define strdup mxml\_strdup

Definition at line 58 of file winconfig.h.

Referenced by [acc\\_connection\\_Thread\(\)](#), [agent\\_AddPersistentVariable\(\)](#), [agent\\_task\\_Copy\(\)](#), [agent\\_xml\\_parse\\_\\_task\(\)](#), [connection\\_Copy\(\)](#), [fipa\\_acl\\_envelope\\_Received\\_Copy\(\)](#), [fipa\\_acl\\_Param\\_Copy\(\)](#), [fipa\\_agent\\_identifier\\_Copy\(\)](#), [fipa\\_message\\_string\\_Copy\(\)](#), [fipa\\_number\\_Copy\(\)](#), [fipa\\_string\\_Copy\(\)](#), [fipa\\_url\\_Copy\(\)](#), [fipa\\_url\\_Parse\(\)](#), [fipa\\_word\\_Copy\(\)](#), [interpreter\\_variable\\_data\\_Copy\(\)](#), [interpreter\\_variable\\_data\\_Initialize\(\)](#), [interpreter\\_variable\\_data\\_InitializeFromAgent\(\)](#), [MC\\_AclAddReceiver\(\)](#), [MC\\_AclAddReplyTo\(\)](#), [MC\\_AclGetSender\(\)](#), [MC\\_AclSend\(\)](#), [MC\\_AclSetContent\(\)](#),

MC\_AclSetConversationID(), MC\_AclSetSender(), MC\_AddStationaryAgent(), MC\_AgentAddTask(), MC\_AgentVariableSave(), MC\_ComposeAgentWithWorkgroup(), MC\_Initialize(), MC\_SaveData\_chdl(), message\_InitializeFromAgent(), mtp\_http\_InitializeFromConnection(), mtp\_http\_Parse(), mxml\_strdupf(), mxmlElementSetAttr(), mxmlIndexNew(), mxmlNewElement(), mxmlNewOpaque(), mxmlNewText(), mxmlSaveAllocString(), mxmlSetElement(), mxmlSetOpaque(), and mxmlSetText().

#### **5.77.1.5 #define vsnprintf mxml\_vsnprintf**

Definition at line 65 of file winconfig.h.

Referenced by debug\_fmt().

### **5.77.2 Function Documentation**

#### **5.77.2.1 char\* mxml\_strdup (const char \*)**

Definition at line 45 of file mxml-string.c.

#### **5.77.2.2 char\* mxml\_strdupf (const char \*, va\_list)**

Definition at line 66 of file mxml-string.c.

References mxml\_vsnprintf(), and strdup.

#### **5.77.2.3 int mxml\_vsnprintf (char \*, size\_t, const char \*, va\_list)**

Definition at line 111 of file mxml-string.c.

References size.

Referenced by mxml\_strdupf().

## 5.78 /home/dko/Projects/mobilec/src/winconfig.h File Reference

### Defines

- #define [PACKAGE\\_STRING](#) "MobileC V2.0.0"
- #define [PACKAGE\\_VERSION](#) "2.0.0"
- #define [strtok\\_r](#)(buf, delim, save\_ptr) strtok( buf, delim )
- #define [strdup](#)(str) \_strdup( str )

### 5.78.1 Define Documentation

#### 5.78.1.1 #define PACKAGE\_STRING "MobileC V2.0.0"

Definition at line 4 of file winconfig.h.

Referenced by mtp\_http\_InitializeFromConnection().

#### 5.78.1.2 #define PACKAGE\_VERSION "2.0.0"

Definition at line 5 of file winconfig.h.

Referenced by mtp\_http\_ComposeMessage(), mtp\_http\_CreateMessage(), and udplisten\_Thread().

#### 5.78.1.3 #define strdup(str) \_strdup( str )

Definition at line 8 of file winconfig.h.

#### 5.78.1.4 #define strtok\_r(buf, delim, save\_ptr) strtok( buf, delim )

Definition at line 6 of file winconfig.h.

Referenced by agent\_xml\_parse\_\_fill\_row\_data(), message\_InitializeFromAgent(), message\_send\_Thread(), and message\_xml\_parse\_\_message().

## 5.79 /home/dko/Projects/mobilec/src/security/asm.c File Reference

```
#include "../include/mc_platform.h"
#include "../include/message.h"
#include "asm.h"
#include "asm_message_composer.h"
#include <mxml.h>
#include "config.h"
#include "bignum.h"
#include "asm_node.h"
```

## 5.80 /home/dko/Projects/mobilec/src/security/asm.h File Reference

```
#include "config.h"
#include "../include/ap_queue_template.h"
#include "../include/data_structures.h"
#include "../mc_list/list.h"
#include "asm_node.h"
```

## 5.81 /home/dko/Projects/mobilec/src/security/asm\_message\_composer.c File Reference

```
#include "asm_message_composer.h"  
#include "config.h"  
#include "../include/mc_platform.h"
```

## 5.82 /home/dko/Projects/mobilec/src/security/asm\_message\_composer.h File Reference

```
#include <mxml.h>
#include "asm.h"
#include "config.h"
```

### 5.83 /home/dko/Projects/mobilec/src/security/asm\_message\_parser.c File Reference

```
#include "asm_message_parser.h"
#include "../include/mc_error.h"
#include "../include/xml_parser.h"
#include "asm_node.h"
#include "config.h"
#include "../include/xml_helper.h"
```



## 5.84 /home/dko/Projects/mobilec/src/security/asm\_message\_parser.h File Reference

```
#include "../include/mc_error.h"
#include "../include/xml_parser.h"
#include "asm_node.h"
#include "config.h"
```

## 5.85 /home/dko/Projects/mobilec/src/security/asm\_node.c File Reference

```
#include <netdb.h>
#include "config.h"
#include "asm.h"
#include "asm_message_parser.h"
#include "asm_node.h"
#include "xyssl-0.9/include/xyssl/havege.h"
#include <stdio.h>
```

## 5.86 /home/dko/Projects/mobilec/src/security/asm\_node.h File Reference

```
#include <mxml.h>
#include <netinet/in.h>
#include "../include/macros.h"
#include "../include/message.h"
#include "xyssl-0.9/include/xyssl/dhm.h"
#include "bignum.h"
#include "xyssl-0.9/include/xyssl/rsa.h"
#include "config.h"
```

## 5.87 /home/dko/Projects/mobilec/src/security/interface.c File Reference

```
#include "interface.h"
#include <sys/socket.h>
```

### Functions

- static void [separate\\_key\\_parts](#) (char \*key, char \*N, char \*E, char \*D, char \*P, char \*Q, char \*DP, char \*DQ, char \*QP)
- [int rsa\\_encryption](#) (char \*publickey, char \*plaintext, char \*ciphertext)
- [int rsa\\_decryption](#) (char \*ciphertext, char \*plaintext, char \*privatekey)
- static [int append\\_nonce\\_to\\_MA](#) ([int](#) \*my\_nonce, char \*MA\_file)
- static [int remove\\_nonce\\_from\\_MA](#) (char \*MA\_file)
- static [int extract\\_nonce\\_from\\_MA](#) ([int](#) sockfd, [int](#) \*my\_nonce, char \*MA\_file)
- [int read\\_known\\_host\\_file](#) (char \*pubkey, char \*hname, char \*filename)
- [int read\\_encrypted\\_file](#) (char \*enfile, char \*string, unsigned char \*passphrase)
- [int initiate\\_migration\\_process](#) ([int](#) new\_fd, [int](#) \*my\_nonce, char \*pubkey, char \*privkey, unsigned char \*aes\_key)
- [int reply\\_migration\\_process](#) ([int](#) sockfd, [int](#) \*my\_nonce, char \*pubkey, char \*privkey, unsigned char \*aes\_key)
- void [generate\\_AES\\_key](#) (char \*key)
- [int aes\\_en\\_de](#) ([int](#) mode, char \*infile, char \*outfile, unsigned char \*AES\_key, [int](#) \*nonce, [int](#) new\_fd)
- [int send\\_AES\\_en\\_MA](#) ([int](#) sockfd, [int](#) \*my\_nonce, char \*outfile, char \*pubkey)
- [int receive\\_AES\\_en\\_MA](#) ([int](#) new\_fd, [int](#) \*nonce, char \*infile, char \*privkey)
- [int generate\\_RSA\\_keys\\_plaintext](#) (char \*pubkeyfile, char \*privkeyfile)
- [int generate\\_RSA\\_keys\\_ciphertext](#) (char \*pubkeyfile, char \*privkeyfile, unsigned char \*passphrase)

### 5.87.1 Function Documentation

#### 5.87.1.1 [int aes\\_en\\_de](#) ([int](#) mode, char \*infile, char \*outfile, unsigned char \*AES\_key, [int](#) \*nonce, [int](#) new\_fd)

Definition at line 1122 of file interface.c.

References [aes\\_crypt\\_ecb\(\)](#), [AES\\_DECRYPT](#), [AES\\_ENCRYPT](#), [aes\\_setkey\\_dec\(\)](#), [aes\\_setkey\\_enc\(\)](#), [append\\_nonce\\_to\\_MA\(\)](#), [extract\\_nonce\\_from\\_MA\(\)](#), [int](#), [MODE\\_DECRYPT](#), [MODE\\_ENCRYPT](#), [sha2\\_finish\(\)](#), [sha2\\_hmac\\_finish\(\)](#), [sha2\\_hmac\\_starts\(\)](#), [sha2\\_hmac\\_update\(\)](#), [sha2\\_starts\(\)](#), and [sha2\\_update\(\)](#).

Referenced by [auth\\_rece\\_send\\_msg\(\)](#), [generate\\_RSA\\_keys\\_ciphertext\(\)](#), [read\\_encrypted\\_file\(\)](#), and [rece\\_de\\_msg\(\)](#).

#### 5.87.1.2 [static int append\\_nonce\\_to\\_MA](#) ([int](#) \*my\_nonce, char \*MA\_file) **[static]**

Definition at line 187 of file interface.c.

Referenced by [aes\\_en\\_de\(\)](#).

**5.87.1.3 static int extract\_nonce\_from\_MA (int *sockfd*, int \* *my\_nonce*, char \* *MA\_file*)  
[static]**

Definition at line 264 of file interface.c.

References `remove_nonce_from_MA()`, and `send`.

Referenced by `aes_en_de()`.

**5.87.1.4 void generate\_AES\_key (char \* *key*)**

Definition at line 1099 of file interface.c.

References `havege_init()`, and `havege_rand()`.

Referenced by `initiate_migration_process()`, and `reply_migration_process()`.

**5.87.1.5 int generate\_RSA\_keys\_ciphertext (char \* *pubkeyfile*, char \* *privkeyfile*, unsigned char \* *passphrase*)**

Definition at line 1752 of file interface.c.

References `aes_en_de()`, `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `EXPONENT`, `havege_init()`, `havege_rand()`, `KEY_SIZE`, `mpi_write_file()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_gen_key()`, `rsa_init()`, and `RSA_PKCS_V15`.

Referenced by `main()`.

**5.87.1.6 int generate\_RSA\_keys\_plaintext (char \* *pubkeyfile*, char \* *privkeyfile*)**

Definition at line 1693 of file interface.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `EXPONENT`, `havege_init()`, `havege_rand()`, `KEY_SIZE`, `mpi_write_file()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_gen_key()`, `rsa_init()`, and `RSA_PKCS_V15`.

Referenced by `main()`.

**5.87.1.7 int initiate\_migration\_process (int *new\_fd*, int \* *my\_nonce*, char \* *pubkey*, char \* *privkey*, unsigned char \* *aes\_key*)**

Definition at line 432 of file interface.c.

References `generate_AES_key()`, `havege_init()`, `havege_rand()`, `md5()`, `rsa_decryption()`, `rsa_encryption()`, and `send`.

Referenced by `auth_rece_send_msg()`.

**5.87.1.8 int read\_encrypted\_file (char \* *enfile*, char \* *string*, unsigned char \* *passphrase*)**

Definition at line 398 of file interface.c.

References `aes_en_de()`, and `f`.

Referenced by `MC_Initialize()`.

**5.87.1.9 int read\_known\_host\_file (char \* *pubkey*, char \* *hname*, char \* *filename*)**

Definition at line 329 of file interface.c.

Referenced by auth\_conn\_rece\_key(), and auth\_rece\_send\_msg().

**5.87.1.10 int receive\_AES\_en\_MA (int *new\_fd*, int \* *nonce*, char \* *infile*, char \* *privkey*)**

Definition at line 1555 of file interface.c.

References rsa\_decryption(), and size.

Referenced by rece\_de\_msg().

**5.87.1.11 static int remove\_nonce\_from\_MA (char \* *MA\_file*) [static]**

Definition at line 220 of file interface.c.

References int.

Referenced by extract\_nonce\_from\_MA().

**5.87.1.12 int reply\_migration\_process (int *sockfd*, int \* *my\_nonce*, char \* *pubkey*, char \* *privkey*, unsigned char \* *aes\_key*)**

Definition at line 777 of file interface.c.

References generate\_AES\_key(), md5(), rsa\_decryption(), rsa\_encryption(), and send.

Referenced by auth\_conn\_rece\_key().

**5.87.1.13 int rsa\_decryption (char \* *ciphertext*, char \* *plaintext*, char \* *privatekey*)**

Definition at line 133 of file interface.c.

References rsa\_context::D, rsa\_context::DP, rsa\_context::DQ, rsa\_context::E, rsa\_context::len, mpi\_msb(), mpi\_read\_mystring(), rsa\_context::N, rsa\_context::P, P, rsa\_context::Q, rsa\_context::QP, rsa\_check\_privkey(), RSA\_DE, rsa\_init(), rsa\_pkcs1\_decrypt(), RSA\_PKCS\_V15, and separate\_key\_parts().

Referenced by initiate\_migration\_process(), receive\_AES\_en\_MA(), and reply\_migration\_process().

**5.87.1.14 int rsa\_encryption (char \* *publickey*, char \* *plaintext*, char \* *ciphertext*)**

Definition at line 91 of file interface.c.

References rsa\_context::E, rsa\_context::len, mpi\_msb(), mpi\_read\_mystring(), rsa\_context::N, rsa\_check\_pubkey(), RSA\_EN, rsa\_init(), rsa\_pkcs1\_encrypt(), RSA\_PKCS\_V15, and separate\_key\_parts().

Referenced by initiate\_migration\_process(), reply\_migration\_process(), and send\_AES\_en\_MA().

**5.87.1.15 int send\_AES\_en\_MA (int *sockfd*, int \* *my\_nonce*, char \* *outfile*, char \* *pubkey*)**

Definition at line 1419 of file interface.c.

References int, rsa\_encryption(), send, and size.

Referenced by `auth_rece_send_msg()`.

**5.87.1.16** `static void separate_key_parts (char * key, char * N, char * E, char * D, char * P,  
char * Q, char * DP, char * DQ, char * QP) [static]`

Definition at line 28 of file `interface.c`.

Referenced by `rsa_decryption()`, and `rsa_encryption()`.

## 5.88 /home/dko/Projects/mobilec/src/security/interface.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/stat.h>
#include "xyssl-0.9/include/xyssl/havege.h"
#include "xyssl-0.9/include/xyssl/bignum.h"
#include "xyssl-0.9/include/xyssl/rsa.h"
#include "xyssl-0.9/include/xyssl/sha2.h"
#include "xyssl-0.9/include/xyssl/aes.h"
```

### Defines

- #define [PATH](#) xyssl-0.9/include/xyssl
- #define [MODE\\_ENCRYPT](#) 0
- #define [MODE\\_DECRYPT](#) 1
- #define [MAXDATASIZE](#) 4096
- #define [KEY\\_SIZE](#) 1024
- #define [EXPONENT](#) 65537
- #define [RSA\\_EN](#) 0
- #define [RSA\\_DE](#) 1

### Functions

- [int read\\_known\\_host\\_file](#) (char \*pubkey, char \*hname, char \*filename)
- [int read\\_encrypted\\_file](#) (char \*enfile, char \*string, unsigned char \*passphrase)
- [int rsa\\_encryption](#) (char \*publickey, char \*plaintext, char \*ciphertext)
- [int rsa\\_decryption](#) (char \*ciphertext, char \*plaintext, char \*privatekeyfile)
- [int initiate\\_migration\\_process](#) (int sockfd, int \*nonce, char \*publickey, char \*privatekey, unsigned char \*aes\_key)
- [int reply\\_migration\\_process](#) (int new\_fd, int \*nonce, char \*publickey, char \*privatekey, unsigned char \*aes\_key)
- void [generate\\_AES\\_key](#) (char \*key)
- [int generate\\_encrypt\\_send\\_AES\\_key](#) (int sockfd, int \*nonce, unsigned char \*key, char \*publickey, char \*privkey)
- [int receive\\_decrypt\\_AES\\_key](#) (int new\_fd, int \*nonce, unsigned char \*key, char \*privkey, char \*publickey)
- [int aes\\_en\\_de](#) (int mode, char \*infile, char \*outfile, unsigned char \*AES\_key, int \*nonce, int sockfd)
- [int send\\_AES\\_en\\_MA](#) (int sockfd, int \*nonce, char \*outfile, char \*peer\_pubkey)
- [int receive\\_AES\\_en\\_MA](#) (int new\_fd, int \*nonce, char \*infile, char \*privatekey)
- [int receiving\\_verifying\\_MA\\_RSA](#) (int sockfd, char \*privkeyfile)
- [int receiving\\_decrypting\\_MA\\_RSA](#) (int new\_fd, char \*privkeyfile)
- [int generate\\_RSA\\_keys\\_plaintext](#) (char \*pubkeyfile, char \*privkeyfile)
- [int generate\\_RSA\\_keys\\_ciphertext](#) (char \*pubkeyfile, char \*privkeyfile, unsigned char \*passphrase)



## 5.88.1 Define Documentation

### 5.88.1.1 #define EXPONENT 65537

Definition at line 25 of file interface.h.

Referenced by generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), and main().

### 5.88.1.2 #define KEY\_SIZE 1024

Definition at line 24 of file interface.h.

Referenced by generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), and main().

### 5.88.1.3 #define MAXDATASIZE 4096

Definition at line 22 of file interface.h.

### 5.88.1.4 #define MODE\_DECRYPT 1

Definition at line 21 of file interface.h.

Referenced by aes\_en\_de(), and main().

### 5.88.1.5 #define MODE\_ENCRYPT 0

Definition at line 20 of file interface.h.

Referenced by aes\_en\_de(), and main().

### 5.88.1.6 #define PATH xyssl-0.9/include/xyssl

Definition at line 9 of file interface.h.

### 5.88.1.7 #define RSA\_DE 1

Definition at line 28 of file interface.h.

Referenced by rsa\_decryption().

### 5.88.1.8 #define RSA\_EN 0

Definition at line 27 of file interface.h.

Referenced by rsa\_encryption().

## 5.88.2 Function Documentation

### 5.88.2.1 **int aes\_en\_de** (int *mode*, char \* *infile*, char \* *outfile*, unsigned char \* *AES\_key*, int \* *nonce*, int *sockfd*)

Definition at line 1122 of file interface.c.

References aes\_crypt\_ecb(), AES\_DECRYPT, AES\_ENCRYPT, aes\_setkey\_dec(), aes\_setkey\_enc(), append\_nonce\_to\_MA(), extract\_nonce\_from\_MA(), int, MODE\_DECRYPT, MODE\_ENCRYPT, sha2\_finish(), sha2\_hmac\_finish(), sha2\_hmac\_starts(), sha2\_hmac\_update(), sha2\_starts(), and sha2\_update().

Referenced by auth\_rece\_send\_msg(), generate\_RSA\_keys\_ciphertext(), read\_encrypted\_file(), and rece\_de\_msg().

### 5.88.2.2 **void generate\_AES\_key** (char \* *key*)

Definition at line 1099 of file interface.c.

References havege\_init(), and havege\_rand().

Referenced by initiate\_migration\_process(), and reply\_migration\_process().

### 5.88.2.3 **int generate\_encrypt\_send\_AES\_key** (int *sockfd*, int \* *nonce*, unsigned char \* *key*, char \* *publickey*, char \* *privkey*)

### 5.88.2.4 **int generate\_RSA\_keys\_ciphertext** (char \* *pubkeyfile*, char \* *privkeyfile*, unsigned char \* *passphrase*)

Definition at line 1752 of file interface.c.

References aes\_en\_de(), rsa\_context::D, rsa\_context::DP, rsa\_context::DQ, rsa\_context::E, EXPONENT, havege\_init(), havege\_rand(), KEY\_SIZE, mpi\_write\_file(), rsa\_context::N, rsa\_context::P, rsa\_context::Q, rsa\_context::QP, rsa\_gen\_key(), rsa\_init(), and RSA\_PKCS\_V15.

Referenced by main().

### 5.88.2.5 **int generate\_RSA\_keys\_plaintext** (char \* *pubkeyfile*, char \* *privkeyfile*)

Definition at line 1693 of file interface.c.

References rsa\_context::D, rsa\_context::DP, rsa\_context::DQ, rsa\_context::E, EXPONENT, havege\_init(), havege\_rand(), KEY\_SIZE, mpi\_write\_file(), rsa\_context::N, rsa\_context::P, rsa\_context::Q, rsa\_context::QP, rsa\_gen\_key(), rsa\_init(), and RSA\_PKCS\_V15.

Referenced by main().

### 5.88.2.6 **int initiate\_migration\_process** (int *sockfd*, int \* *nonce*, char \* *publickey*, char \* *privatekey*, unsigned char \* *aes\_key*)

Definition at line 432 of file interface.c.

References generate\_AES\_key(), havege\_init(), havege\_rand(), md5(), rsa\_decryption(), rsa\_encryption(), and send.

Referenced by auth\_rece\_send\_msg().

**5.88.2.7 int read\_encrypted\_file (char \* *enfile*, char \* *string*, unsigned char \* *passphrase*)**

Definition at line 398 of file interface.c.

References aes\_en\_de(), and f.

Referenced by MC\_Initialize().

**5.88.2.8 int read\_known\_host\_file (char \* *pubkey*, char \* *hname*, char \* *filename*)**

Definition at line 329 of file interface.c.

Referenced by auth\_conn\_rece\_key(), and auth\_rece\_send\_msg().

**5.88.2.9 int receive\_AES\_en\_MA (int *new\_fd*, int \* *nonce*, char \* *infile*, char \* *privatekey*)**

Definition at line 1555 of file interface.c.

References rsa\_decryption(), and size.

Referenced by rece\_de\_msg().

**5.88.2.10 int receive\_decrypt\_AES\_key (int *new\_fd*, int \* *nonce*, unsigned char \* *key*, char \* *privkey*, char \* *publickey*)****5.88.2.11 int receiving\_decrypting\_MA\_RSA (int *new\_fd*, char \* *privkeyfile*)****5.88.2.12 int receiving\_verifying\_MA\_RSA (int *sockfd*, char \* *privkeyfile*)****5.88.2.13 int reply\_migration\_process (int *new\_fd*, int \* *nonce*, char \* *publickey*, char \* *privatekey*, unsigned char \* *aes\_key*)**

Definition at line 777 of file interface.c.

References generate\_AES\_key(), md5(), rsa\_decryption(), rsa\_encryption(), and send.

Referenced by auth\_conn\_rece\_key().

**5.88.2.14 int rsa\_decryption (char \* *ciphertext*, char \* *plaintext*, char \* *privatekeyfile*)**

Definition at line 133 of file interface.c.

References rsa\_context::D, rsa\_context::DP, rsa\_context::DQ, rsa\_context::E, rsa\_context::len, mpi\_msb(), mpi\_read\_mystring(), rsa\_context::N, rsa\_context::P, P, rsa\_context::Q, rsa\_context::QP, rsa\_check\_privkey(), RSA\_DE, rsa\_init(), rsa\_pkcs1\_decrypt(), RSA\_PKCS\_V15, and separate\_key\_parts().

Referenced by initiate\_migration\_process(), receive\_AES\_en\_MA(), and reply\_migration\_process().

**5.88.2.15 int rsa\_encryption (char \* *publickey*, char \* *plaintext*, char \* *ciphertext*)**

Definition at line 91 of file interface.c.

References rsa\_context::E, rsa\_context::len, mpi\_msb(), mpi\_read\_mystring(), rsa\_context::N, rsa\_check\_pubkey(), RSA\_EN, rsa\_init(), rsa\_pkcs1\_encrypt(), RSA\_PKCS\_V15, and separate\_key\_parts().

Referenced by initiate\_migration\_process(), reply\_migration\_process(), and send\_AES\_en\_MA().

**5.88.2.16** `int send_AES_en_MA (int sockfd, int * nonce, char * outfile, char * peer_pubkey)`

Definition at line 1419 of file `interface.c`.

References `int`, `rsa_encryption()`, `send`, and `size`.

Referenced by `auth_rece_send_msg()`.

## 5.89 /home/dko/Projects/mobilec/src/security/mc\_dh.c File Reference

```
#include <stdio.h>
#include "../include/mc_error.h"
#include "xyssl-0.9/include/xyssl/dhm.h"
#include "xyssl-0.9/include/xyssl/havege.h"
#include "mc_dh.h"
#include "asm_node.h"
#include "config.h"
```

## 5.90 /home/dko/Projects/mobilec/src/security/mc\_dh.h File Reference

```
#include "xyssl-0.9/include/xyssl/rsa.h"
#include "asm_node.h"
#include "config.h"
```

## 5.91 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/aes.h File Reference

### Data Structures

- struct [aes\\_context](#)  
*AES context structure.*

### Defines

- #define [AES\\_ENCRYPT](#) 1
- #define [AES\\_DECRYPT](#) 0

### Functions

- void [aes\\_setkey\\_enc](#) ([aes\\_context](#) \*ctx, unsigned char \*key, int keysize)  
*AES key schedule (encryption).*
- void [aes\\_setkey\\_dec](#) ([aes\\_context](#) \*ctx, unsigned char \*key, int keysize)  
*AES key schedule (decryption).*
- void [aes\\_crypt\\_ecb](#) ([aes\\_context](#) \*ctx, int mode, unsigned char input[16], unsigned char output[16])  
*AES-ECB block encryption/decryption.*
- void [aes\\_crypt\\_cbc](#) ([aes\\_context](#) \*ctx, int mode, int length, unsigned char iv[16], unsigned char \*input, unsigned char \*output)  
*AES-CBC buffer encryption/decryption.*
- void [aes\\_crypt\\_cfb](#) ([aes\\_context](#) \*ctx, int mode, int length, int \*iv\_off, unsigned char iv[16], unsigned char \*input, unsigned char \*output)  
*AES-CFB buffer encryption/decryption.*
- int [aes\\_self\\_test](#) (int verbose)  
*Checkup routine.*

### 5.91.1 Detailed Description

Definition in file [aes.h](#).

### 5.91.2 Define Documentation

#### 5.91.2.1 #define AES\_DECRYPT 0

Definition at line 8 of file [aes.h](#).

Referenced by [aes\\_crypt\\_cbc\(\)](#), [aes\\_crypt\\_cfb\(\)](#), [aes\\_crypt\\_ecb\(\)](#), [aes\\_en\\_de\(\)](#), [aes\\_self\\_test\(\)](#), [main\(\)](#), and [ssl\\_decrypt\\_buf\(\)](#).

### 5.91.2.2 #define AES\_ENCRYPT 1

Definition at line 7 of file aes.h.

Referenced by aes\_crypt\_cfb(), aes\_en\_de(), main(), and ssl\_encrypt\_buf().

## 5.91.3 Function Documentation

### 5.91.3.1 void aes\_crypt\_cbc (aes\_context \* ctx, int mode, int length, unsigned char iv[16], unsigned char \* input, unsigned char \* output)

AES-CBC buffer encryption/decryption.

#### Parameters

*ctx* AES context  
*mode* AES\_ENCRYPT or AES\_DECRYPT  
*length* length of the input data  
*iv* initialization vector (updated after use)  
*input* buffer holding the input data  
*output* buffer holding the output data

Definition at line 732 of file aes.c.

References aes\_crypt\_ecb(), and AES\_DECRYPT.

Referenced by aes\_self\_test(), main(), ssl\_decrypt\_buf(), and ssl\_encrypt\_buf().

### 5.91.3.2 void aes\_crypt\_cfb (aes\_context \* ctx, int mode, int length, int \* iv\_off, unsigned char iv[16], unsigned char \* input, unsigned char \* output)

AES-CFB buffer encryption/decryption.

#### Parameters

*ctx* AES context  
*mode* AES\_ENCRYPT or AES\_DECRYPT  
*length* length of the input data  
*iv\_off* offset in IV (updated after use)  
*iv* initialization vector (updated after use)  
*input* buffer holding the input data  
*output* buffer holding the output data

Definition at line 787 of file aes.c.

References aes\_crypt\_ecb(), AES\_DECRYPT, and AES\_ENCRYPT.

Referenced by aes\_self\_test().



### 5.91.3.3 void aes\_crypt\_ecb (aes\_context \* *ctx*, int *mode*, unsigned char *input*[16], unsigned char *output*[16])

AES-ECB block encryption/decryption.

#### Parameters

*ctx* AES context

*mode* AES\_ENCRYPT or AES\_DECRYPT

*input* 16-byte input block

*output* 16-byte output block

Definition at line 639 of file aes.c.

References AES\_DECRYPT, AES\_FROUND, AES\_RROUND, FSb, GET\_ULONG\_LE, aes\_context::nr, PUT\_ULONG\_LE, aes\_context::rk, and RSb.

Referenced by aes\_crypt\_cbc(), aes\_crypt\_cfb(), aes\_en\_de(), aes\_self\_test(), and main().

### 5.91.3.4 int aes\_self\_test (int *verbose*)

Checkup routine.

#### Returns

0 if successful, or 1 if the test failed

Definition at line 902 of file aes.c.

References aes\_crypt\_cbc(), aes\_crypt\_cfb(), aes\_crypt\_ecb(), AES\_DECRYPT, aes\_setkey\_dec(), aes\_setkey\_enc(), aes\_test\_cbc\_dec, aes\_test\_cbc\_enc, aes\_test\_cfb\_dec, aes\_test\_cfb\_enc, aes\_test\_ecb\_dec, aes\_test\_ecb\_enc, buf, and prv.

Referenced by main().

### 5.91.3.5 void aes\_setkey\_dec (aes\_context \* *ctx*, unsigned char \* *key*, int *keysize*)

AES key schedule (decryption).

#### Parameters

*ctx* AES context to be initialized

*key* decryption key

*keysize* must be 128, 192 or 256

Definition at line 542 of file aes.c.

References aes\_setkey\_enc(), aes\_context::buf, FSb, aes\_context::nr, aes\_context::rk, RT0, RT1, RT2, and RT3.

Referenced by aes\_en\_de(), aes\_self\_test(), main(), and ssl\_derive\_keys().

**5.91.3.6 void aes\_setkey\_enc (aes\_context \* *ctx*, unsigned char \* *key*, int *keysize*)**

AES key schedule (encryption).

**Parameters**

*ctx* AES context to be initialized

*key* encryption key

*keysize* must be 128, 192 or 256

Definition at line 439 of file aes.c.

References aes\_gen\_tables(), aes\_init\_done, aes\_context::buf, FSb, GET\_ULONG\_LE, aes\_context::nr, RCON, and aes\_context::rk.

Referenced by aes\_en\_de(), aes\_self\_test(), aes\_setkey\_dec(), main(), and ssl\_derive\_keys().

## 5.92 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/arc4.h File Reference

### Data Structures

- struct [arc4\\_context](#)  
*ARC4 context structure.*

### Functions

- void [arc4\\_setup](#) ([arc4\\_context](#) \*ctx, unsigned char \*key, [int](#) keylen)  
*ARC4 key schedule.*
- void [arc4\\_crypt](#) ([arc4\\_context](#) \*ctx, unsigned char \*buf, [int](#) buflen)  
*ARC4 cipher function.*
- [int](#) [arc4\\_self\\_test](#) ([int](#) verbose)

### 5.92.1 Detailed Description

Definition in file [arc4.h](#).

### 5.92.2 Function Documentation

#### 5.92.2.1 void [arc4\\_crypt](#) ([arc4\\_context](#) \* ctx, unsigned char \* buf, [int](#) buflen)

ARC4 cipher function.

#### Parameters

*ctx* ARC4 context  
*buf* buffer to be processed  
*buflen* amount of data in buf

Definition at line 63 of file [arc4.c](#).

References [arc4\\_context::m](#), [arc4\\_context::x](#), and [arc4\\_context::y](#).

Referenced by [arc4\\_self\\_test\(\)](#), [main\(\)](#), [ssl\\_decrypt\\_buf\(\)](#), and [ssl\\_encrypt\\_buf\(\)](#).

#### 5.92.2.2 [int](#) [arc4\\_self\\_test](#) ([int](#) verbose)

Definition at line 122 of file [arc4.c](#).

References [arc4\\_crypt\(\)](#), [arc4\\_setup\(\)](#), [arc4\\_test\\_ct](#), [arc4\\_test\\_key](#), [arc4\\_test\\_pt](#), and [buf](#).

Referenced by [main\(\)](#).

**5.92.2.3 void arc4\_setup (arc4\_context \* *ctx*, unsigned char \* *key*, int *keylen*)**

ARC4 key schedule.

**Parameters**

*ctx* ARC4 context to be initialized

*key* the secret key

*keylen* length of the key

Definition at line 35 of file arc4.c.

References arc4\_context::m, arc4\_context::x, and arc4\_context::y.

Referenced by arc4\_self\_test(), main(), and ssl\_derive\_keys().

## 5.93 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/base64.h File Reference

### Defines

- `#define XYSSL_ERR_BASE64_BUFFER_TOO_SMALL -0x0010`
- `#define XYSSL_ERR_BASE64_INVALID_CHARACTER -0x0012`

### Functions

- `int base64_encode` (unsigned char \*dst, int \*dlen, unsigned char \*src, int slen)  
*Encode a buffer into base64 format.*
- `int base64_decode` (unsigned char \*dst, int \*dlen, unsigned char \*src, int slen)  
*Decode a base64-formatted buffer.*
- `int base64_self_test` (int verbose)  
*Checkup routine.*

### 5.93.1 Detailed Description

Definition in file [base64.h](#).

### 5.93.2 Define Documentation

#### 5.93.2.1 `#define XYSSL_ERR_BASE64_BUFFER_TOO_SMALL -0x0010`

Definition at line 7 of file [base64.h](#).

Referenced by [base64\\_decode\(\)](#), and [base64\\_encode\(\)](#).

#### 5.93.2.2 `#define XYSSL_ERR_BASE64_INVALID_CHARACTER -0x0012`

Definition at line 8 of file [base64.h](#).

Referenced by [base64\\_decode\(\)](#), [x509parse\\_crt\(\)](#), and [x509parse\\_key\(\)](#).

### 5.93.3 Function Documentation

#### 5.93.3.1 `int base64_decode` (unsigned char \*dst, int \*dlen, unsigned char \*src, int slen)

Decode a base64-formatted buffer.

#### Parameters

- dst* destination buffer
- dlen* size of the buffer
- src* source buffer

*slen* amount of data to be decoded

### Returns

0 if successful, XYSSL\_ERR\_BASE64\_BUFFER\_TOO\_SMALL, or XYSSL\_ERR\_BASE64\_INVALID\_DATA if the input data is not correct. \*dlen is always updated to reflect the amount of data that has (or would have) been written.

### Note

Call this function with \*dlen = 0 to obtain the required buffer size in \*dlen

Definition at line 121 of file base64.c.

References base64\_dec\_map, XYSSL\_ERR\_BASE64\_BUFFER\_TOO\_SMALL, and XYSSL\_ERR\_BASE64\_INVALID\_CHARACTER.

Referenced by base64\_self\_test(), x509parse\_crt(), and x509parse\_key().

### 5.93.3.2 int base64\_encode (unsigned char \*dst, int \*dlen, unsigned char \*src, int slen)

Encode a buffer into base64 format.

### Parameters

*dst* destination buffer

*dlen* size of the buffer

*src* source buffer

*slen* amount of data to be encoded

### Returns

0 if successful, or XYSSL\_ERR\_BASE64\_BUFFER\_TOO\_SMALL. \*dlen is always updated to reflect the amount of data that has (or would have) been written.

### Note

Call this function with \*dlen = 0 to obtain the required buffer size in \*dlen

Definition at line 58 of file base64.c.

References base64\_enc\_map, and XYSSL\_ERR\_BASE64\_BUFFER\_TOO\_SMALL.

Referenced by base64\_self\_test().

### 5.93.3.3 int base64\_self\_test (int verbose)

Checkup routine.

### Returns

0 if successful, or 1 if the test failed

Definition at line 206 of file base64.c.

References base64\_decode(), base64\_encode(), base64\_test\_dec, and base64\_test\_enc.

Referenced by main().

## 5.94 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/bignum.h File Reference

```
#include <stdio.h>
```

### Data Structures

- struct [mpi](#)  
*MPI structure.*

### Defines

- #define [XYSSL\\_ERR\\_MPI\\_FILE\\_IO\\_ERROR](#) -0x0002
- #define [XYSSL\\_ERR\\_MPI\\_BAD\\_INPUT\\_DATA](#) -0x0004
- #define [XYSSL\\_ERR\\_MPI\\_INVALID\\_CHARACTER](#) -0x0006
- #define [XYSSL\\_ERR\\_MPI\\_BUFFER\\_TOO\\_SMALL](#) -0x0008
- #define [XYSSL\\_ERR\\_MPI\\_NEGATIVE\\_VALUE](#) -0x000A
- #define [XYSSL\\_ERR\\_MPI\\_DIVISION\\_BY\\_ZERO](#) -0x000C
- #define [XYSSL\\_ERR\\_MPI\\_NOT\\_ACCEPTABLE](#) -0x000E
- #define [MPI\\_CHK](#)(f) if( ( ret = f ) != 0 ) goto cleanup

### Typedefs

- typedef unsigned long [t\\_int](#)
- typedef unsigned long long [t\\_dbl](#)

### Functions

- void [mpi\\_init](#) ([mpi](#) \*X,...)  
*Initialize one or more mpi.*
- void [mpi\\_free](#) ([mpi](#) \*X,...)  
*Unallocate one or more mpi.*
- int [mpi\\_grow](#) ([mpi](#) \*X, int nlimbs)  
*Enlarge to the specified number of limbs.*
- int [mpi\\_copy](#) ([mpi](#) \*X, [mpi](#) \*Y)  
*Copy the contents of Y into X.*
- void [mpi\\_swap](#) ([mpi](#) \*X, [mpi](#) \*Y)  
*Swap the contents of X and Y.*
- int [mpi\\_lset](#) ([mpi](#) \*X, int z)  
*Set value from integer.*
- int [mpi\\_lsb](#) ([mpi](#) \*X)

*Return the number of least significant bits.*

- `int mpi_msb (mpi *X)`

*Return the number of most significant bits.*

- `int mpi_size (mpi *X)`

*Return the total size in bytes.*

- `int mpi_read_string (mpi *X, int radix, char *s)`

*Import from an ASCII string.*

- `int mpi_write_string (mpi *X, int radix, char *s, int *slen)`

*Export into an ASCII string.*

- `int mpi_read_file (mpi *X, int radix, FILE *fin)`

*Read X from an opened file.*

- `int mpi_read_mystring (mpi *X, int radix, char *s)`

- `int mpi_write_file (char *p, mpi *X, int radix, FILE *fout)`

*Write X into an opened file, or stdout.*

- `int mpi_read_binary (mpi *X, unsigned char *buf, int buflen)`

*Import X from unsigned binary data, big endian.*

- `int mpi_write_binary (mpi *X, unsigned char *buf, int buflen)`

*Export X into unsigned binary data, big endian.*

- `int mpi_shift_l (mpi *X, int count)`

*Left-shift:  $X \ll = \text{count}$ .*

- `int mpi_shift_r (mpi *X, int count)`

*Right-shift:  $X \gg = \text{count}$ .*

- `int mpi_cmp_abs (mpi *X, mpi *Y)`

*Compare unsigned values.*

- `int mpi_cmp_mpi (mpi *X, mpi *Y)`

*Compare signed values.*

- `int mpi_cmp_int (mpi *X, int z)`

*Compare signed values.*

- `int mpi_add_abs (mpi *X, mpi *A, mpi *B)`

*Unsigned addition:  $X = |A| + |B|$ .*

- `int mpi_sub_abs (mpi *X, mpi *A, mpi *B)`

*Unsigned subtraction:  $X = |A| - |B|$ .*

- `int mpi_add_mpi (mpi *X, mpi *A, mpi *B)`

*Signed addition:  $X = A + B$ .*



- `int mpi_sub_mpi (mpi *X, mpi *A, mpi *B)`  
*Signed subtraction:  $X = A - B$ .*
- `int mpi_add_int (mpi *X, mpi *A, int b)`  
*Signed addition:  $X = A + b$ .*
- `int mpi_sub_int (mpi *X, mpi *A, int b)`  
*Signed subtraction:  $X = A - b$ .*
- `int mpi_mul_mpi (mpi *X, mpi *A, mpi *B)`  
*Baseline multiplication:  $X = A * B$ .*
- `int mpi_mul_int (mpi *X, mpi *A, t_int b)`  
*Baseline multiplication:  $X = A * b$ .*
- `int mpi_div_mpi (mpi *Q, mpi *R, mpi *A, mpi *B)`  
*Division by mpi:  $A = Q * B + R$ .*
- `int mpi_div_int (mpi *Q, mpi *R, mpi *A, int b)`  
*Division by int:  $A = Q * b + R$ .*
- `int mpi_mod_mpi (mpi *R, mpi *A, mpi *B)`  
*Modulo:  $R = A \bmod B$ .*
- `int mpi_mod_int (t_int *r, mpi *A, int b)`  
*Modulo:  $r = A \bmod b$ .*
- `int mpi_exp_mod (mpi *X, mpi *A, mpi *E, mpi *N, mpi *_RR)`  
*Sliding-window exponentiation:  $X = A^E \bmod N$ .*
- `int mpi_gcd (mpi *G, mpi *A, mpi *B)`  
*Greatest common divisor:  $G = \gcd(A, B)$ .*
- `int mpi_inv_mod (mpi *X, mpi *A, mpi *N)`  
*Modular inverse:  $X = A^{-1} \bmod N$ .*
- `int mpi_is_prime (mpi *X, int(*f_rng)(void *), void *p_rng)`  
*Miller-Rabin primality test.*
- `int mpi_gen_prime (mpi *X, int nbits, int dh_flag, int(*f_rng)(void *), void *p_rng)`  
*Prime number generation.*
- `int mpi_self_test (int verbose)`  
*Checkup routine.*

### 5.94.1 Detailed Description

Definition in file [bignum.h](#).

## 5.94.2 Define Documentation

### 5.94.2.1 **#define MPI\_CHK(f) if( ( ret = f ) != 0 ) goto cleanup**

Definition at line 17 of file bignum.h.

Referenced by dhmm\_calc\_secret(), dhmm\_make\_params(), dhmm\_make\_public(), mpi\_add\_abs(), mpi\_add\_mpi(), mpi\_copy(), mpi\_div\_mpi(), mpi\_exp\_mod(), mpi\_gcd(), mpi\_gen\_prime(), mpi\_inv\_mod(), mpi\_is\_prime(), mpi\_lset(), mpi\_mod\_mpi(), mpi\_mul\_mpi(), mpi\_read\_binary(), mpi\_read\_string(), mpi\_self\_test(), mpi\_shift\_l(), mpi\_sub\_abs(), mpi\_sub\_mpi(), mpi\_write\_file(), mpi\_write\_hlp(), mpi\_write\_string(), rsa\_check\_privkey(), rsa\_gen\_key(), rsa\_private(), and rsa\_public().

### 5.94.2.2 **#define XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA -0x0004**

Definition at line 10 of file bignum.h.

Referenced by mpi\_exp\_mod(), mpi\_gen\_prime(), mpi\_inv\_mod(), mpi\_read\_string(), mpi\_write\_hlp(), and mpi\_write\_string().

### 5.94.2.3 **#define XYSSL\_ERR\_MPI\_BUFFER\_TOO\_SMALL -0x0008**

Definition at line 12 of file bignum.h.

Referenced by mpi\_write\_binary(), and mpi\_write\_string().

### 5.94.2.4 **#define XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO -0x000C**

Definition at line 14 of file bignum.h.

Referenced by mpi\_div\_mpi(), and mpi\_mod\_int().

### 5.94.2.5 **#define XYSSL\_ERR\_MPI\_FILE\_IO\_ERROR -0x0002**

Definition at line 9 of file bignum.h.

Referenced by mpi\_read\_file(), and mpi\_write\_file().

### 5.94.2.6 **#define XYSSL\_ERR\_MPI\_INVALID\_CHARACTER -0x0006**

Definition at line 11 of file bignum.h.

Referenced by mpi\_get\_digit().

### 5.94.2.7 **#define XYSSL\_ERR\_MPI\_NEGATIVE\_VALUE -0x000A**

Definition at line 13 of file bignum.h.

Referenced by mpi\_sub\_abs().

### 5.94.2.8 **#define XYSSL\_ERR\_MPI\_NOT\_ACCEPTABLE -0x000E**

Definition at line 15 of file bignum.h.

Referenced by `mpi_gen_prime()`, `mpi_inv_mod()`, and `mpi_is_prime()`.

### 5.94.3 Typedef Documentation

#### 5.94.3.1 `typedef unsigned long long t_dbl`

Definition at line 39 of file `bignum.h`.

#### 5.94.3.2 `typedef unsigned long t_int`

Definition at line 30 of file `bignum.h`.

### 5.94.4 Function Documentation

#### 5.94.4.1 `int mpi_add_abs (mpi * X, mpi * A, mpi * B)`

Unsigned addition:  $X = |A| + |B|$ .

##### Returns

0 if successful, 1 if memory allocation failed

Definition at line 694 of file `bignum.c`.

References `MPI_CHK`, `mpi_copy()`, `mpi_grow()`, `mpi::n`, and `mpi::p`.

Referenced by `mpi_add_mpi()`, and `mpi_sub_mpi()`.

#### 5.94.4.2 `int mpi_add_int (mpi * X, mpi * A, int b)`

Signed addition:  $X = A + b$ .

##### Returns

0 if successful, 1 if memory allocation failed

Definition at line 860 of file `bignum.c`.

References `mpi_add_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_gen_prime()`, and `mpi_read_string()`.

#### 5.94.4.3 `int mpi_add_mpi (mpi * X, mpi * A, mpi * B)`

Signed addition:  $X = A + B$ .

##### Returns

0 if successful, 1 if memory allocation failed

Definition at line 798 of file `bignum.c`.

References `mpi_add_abs()`, `MPI_CHK`, `mpi_cmp_abs()`, `mpi_sub_abs()`, and `mpi::s`.

Referenced by `mpi_add_int()`, `mpi_div_mpi()`, `mpi_inv_mod()`, `mpi_mod_mpi()`, and `rsa_private()`.

**5.94.4.4 int mpi\_cmp\_abs (mpi \* X, mpi \* Y)**

Compare unsigned values.

**Returns**

1 if  $|X|$  is greater than  $|Y|$ , -1 if  $|X|$  is lesser than  $|Y|$  or 0 if  $|X|$  is equal to  $|Y|$

Definition at line 615 of file bignum.c.

References `mpi::n`, and `mpi::p`.

Referenced by `mpi_add_mpi()`, `mpi_div_mpi()`, `mpi_montmul()`, `mpi_sub_abs()`, and `mpi_sub_mpi()`.

**5.94.4.5 int mpi\_cmp\_int (mpi \* X, int z)**

Compare signed values.

**Returns**

1 if X is greater than z, -1 if X is lesser than z or 0 if X is equal to z

Definition at line 678 of file bignum.c.

References `mpi_cmp_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mod_mpi()`, `mpi_write_hlp()`, `rsa_check_privkey()`, and `rsa_gen_key()`.

**5.94.4.6 int mpi\_cmp\_mpi (mpi \* X, mpi \* Y)**

Compare signed values.

**Returns**

1 if X is greater than Y, -1 if X is lesser than Y or 0 if X is equal to Y

Definition at line 645 of file bignum.c.

References `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `dhm_make_params()`, `dhm_make_public()`, `mpi_cmp_int()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mod_mpi()`, `mpi_self_test()`, `rsa_check_privkey()`, `rsa_gen_key()`, `rsa_private()`, and `rsa_public()`.

**5.94.4.7 int mpi\_copy (mpi \* X, mpi \* Y)**

Copy the contents of Y into X.

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 128 of file bignum.c.

References `ciL`, `MPI_CHK`, `mpi_grow()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_add_abs()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mul_mpi()`, `mpi_sub_abs()`, and `mpi_write_string()`.

**5.94.4.8 int mpi\_div\_int (mpi \* *Q*, mpi \* *R*, mpi \* *A*, int *b*)**

Division by int:  $A = Q * b + R$ .

**Returns**

0 if successful, 1 if memory allocation failed, XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO if  $b == 0$

**Note**

Either *Q* or *R* can be NULL.

Definition at line 1173 of file bignum.c.

References mpi\_div\_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by main(), and mpi\_write\_hlp().

**5.94.4.9 int mpi\_div\_mpi (mpi \* *Q*, mpi \* *R*, mpi \* *A*, mpi \* *B*)**

Division by mpi:  $A = Q * B + R$ .

**Returns**

0 if successful, 1 if memory allocation failed, XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO if  $B == 0$

**Note**

Either *Q* or *R* can be NULL.

Definition at line 1008 of file bignum.c.

References biH, biL, mpi\_add\_mpi(), MPI\_CHK, mpi\_cmp\_abs(), mpi\_cmp\_int(), mpi\_cmp\_mpi(), mpi\_copy(), mpi\_free(), mpi\_grow(), mpi\_init(), mpi\_lset(), mpi\_msb(), mpi\_mul\_int(), mpi\_shift\_l(), mpi\_shift\_r(), mpi\_sub\_mpi(), mpi::n, mpi::p, mpi::s, and XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO.

Referenced by mpi\_div\_int(), mpi\_mod\_mpi(), and mpi\_self\_test().

**5.94.4.10 int mpi\_exp\_mod (mpi \* *X*, mpi \* *A*, mpi \* *E*, mpi \* *N*, mpi \* *\_RR*)**

Sliding-window exponentiation:  $X = A^E \bmod N$ .

**Returns**

0 if successful, 1 if memory allocation failed, XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA if *N* is negative or even

**Note**

*\_RR* is used to avoid re-computing  $R * R \bmod N$  across multiple calls, which speeds up things a bit. It can be set to NULL if the extra performance is unneeded.

Definition at line 1328 of file bignum.c.

References biL, MPI\_CHK, mpi\_cmp\_int(), mpi\_cmp\_mpi(), mpi\_copy(), mpi\_free(), mpi\_grow(), mpi\_init(), mpi\_lset(), mpi\_mod\_mpi(), mpi\_montg\_init(), mpi\_montmul(), mpi\_montred(), mpi\_msb(), mpi\_shift\_l(), mpi::n, mpi::p, and XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA.

Referenced by dhm\_calc\_secret(), dhm\_make\_params(), dhm\_make\_public(), main(), mpi\_is\_prime(), mpi\_self\_test(), rsa\_private(), and rsa\_public().

**5.94.4.11 void mpi\_free (mpi \* *X*, ...)**

Unallocate one or more mpi.

Definition at line 73 of file bignum.c.

References ciL, mpi::n, mpi::p, and mpi::s.

Referenced by dhm\_free(), main(), mpi\_div\_mpi(), mpi\_exp\_mod(), mpi\_gcd(), mpi\_gen\_prime(), mpi\_inv\_mod(), mpi\_is\_prime(), mpi\_mul\_mpi(), mpi\_read\_string(), mpi\_self\_test(), mpi\_sub\_abs(), mpi\_write\_string(), rsa\_check\_privkey(), rsa\_free(), rsa\_gen\_key(), rsa\_private(), and rsa\_public().

**5.94.4.12 int mpi\_gcd (mpi \* *G*, mpi \* *A*, mpi \* *B*)**

Greatest common divisor:  $G = \gcd(A, B)$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 1507 of file bignum.c.

References MPI\_CHK, mpi\_cmp\_int(), mpi\_cmp\_mpi(), mpi\_copy(), mpi\_free(), mpi\_init(), mpi\_lset(), mpi\_mul\_mpi(), mpi\_shift\_r(), mpi\_sub\_abs(), mpi::p, and mpi::s.

Referenced by mpi\_inv\_mod(), rsa\_check\_privkey(), and rsa\_gen\_key().

**5.94.4.13 int mpi\_gen\_prime (mpi \* *X*, int *nbits*, int *dh\_flag*, int(\*) (void \*) *f\_rng*, void \* *p\_rng*)**

Prime number generation.

**Parameters**

*X* destination mpi

*nbits* required size of *X* in bits

*dh\_flag* if 1, then  $(X-1)/2$  will be prime too

*f\_rng* RNG function

*p\_rng* RNG parameter

**Returns**

0 if successful (probably prime), 1 if memory allocation failed, XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA if *nbits* is  $< 3$

Definition at line 1778 of file bignum.c.

References BITS\_TO\_LIMBS, ciL, mpi\_add\_int(), MPI\_CHK, mpi\_free(), mpi\_grow(), mpi\_init(), mpi\_is\_prime(), mpi\_lset(), mpi\_msb(), mpi\_shift\_l(), mpi\_shift\_r(), mpi\_sub\_int(), mpi::p, XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_MPI\_NOT\_ACCEPTABLE.

Referenced by main(), and rsa\_gen\_key().

**5.94.4.14 int mpi\_grow (mpi \* *X*, int *nlimbs*)**

Enlarge to the specified number of limbs.

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 100 of file bignum.c.

References `ciL`, `mpi::n`, and `mpi::p`.

Referenced by `dhm_make_params()`, `dhm_make_public()`, `mpi_add_abs()`, `mpi_copy()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gen_prime()`, `mpi_is_prime()`, `mpi_lset()`, `mpi_mul_mpi()`, `mpi_read_binary()`, `mpi_read_string()`, and `mpi_shift_l()`.

**5.94.4.15 void mpi\_init (mpi \* X, ...)**

Initialize one or more mpi.

Definition at line 52 of file bignum.c.

References `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `main()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mul_mpi()`, `mpi_read_string()`, `mpi_self_test()`, `mpi_sub_abs()`, `mpi_write_string()`, `rsa_check_privkey()`, `rsa_gen_key()`, `rsa_private()`, and `rsa_public()`.

**5.94.4.16 int mpi\_inv\_mod (mpi \* X, mpi \* A, mpi \* N)**

Modular inverse:  $X = A^{-1} \bmod N$ .

**Returns**

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_BAD_INPUT_DATA` if `N` is negative or nil `XYSSL_ERR_MPI_NOT_ACCEPTABLE` if `A` has no inverse mod `N`

Definition at line 1549 of file bignum.c.

References `mpi_add_mpi()`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_free()`, `mpi_gcd()`, `mpi_init()`, `mpi_lset()`, `mpi_mod_mpi()`, `mpi_shift_r()`, `mpi_sub_mpi()`, `mpi::p`, `XYSSL_ERR_MPI_BAD_INPUT_DATA`, and `XYSSL_ERR_MPI_NOT_ACCEPTABLE`.

Referenced by `main()`, `mpi_self_test()`, and `rsa_gen_key()`.

**5.94.4.17 int mpi\_is\_prime (mpi \* X, int(\*)(void \*)f\_rng, void \* p\_rng)**

Miller-Rabin primality test.

**Returns**

0 if successful (probably prime), 1 if memory allocation failed, `XYSSL_ERR_MPI_NOT_ACCEPTABLE` if `X` is not prime

Definition at line 1667 of file bignum.c.

References `ciL`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_grow()`, `mpi_init()`, `mpi_lsb()`, `mpi_mod_int()`, `mpi_mod_mpi()`, `mpi_msb()`, `mpi_mul_mpi()`, `mpi_shift_r()`, `mpi_sub_int()`, `mpi::n`, `mpi::p`, `R`, `mpi::s`, `small_prime`, and `XYSSL_ERR_MPI_NOT_ACCEPTABLE`.

Referenced by `main()`, and `mpi_gen_prime()`.

**5.94.4.18 int mpi\_lsb (mpi \* X)**

Return the number of least significant bits.

Definition at line 185 of file bignum.c.

References biL, int, mpi::n, and mpi::p.

Referenced by mpi\_is\_prime().

**5.94.4.19 int mpi\_lset (mpi \* X, int z)**

Set value from integer.

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 167 of file bignum.c.

References ciL, MPI\_CHK, mpi\_grow(), mpi::n, mpi::p, and mpi::s.

Referenced by dh\_m\_make\_params(), dh\_m\_make\_public(), mpi\_div\_mpi(), mpi\_exp\_mod(), mpi\_gcd(), mpi\_gen\_prime(), mpi\_inv\_mod(), mpi\_mul\_mpi(), mpi\_read\_binary(), mpi\_read\_string(), and rsa\_gen\_key().

**5.94.4.20 int mpi\_mod\_int (t\_int \* r, mpi \* A, int b)**

Modulo:  $r = A \bmod b$ .

**Returns**

0 if successful, 1 if memory allocation failed, XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO if  $b == 0$

Definition at line 1209 of file bignum.c.

References biH, mpi::n, mpi::p, and XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO.

Referenced by mpi\_is\_prime(), and mpi\_write\_hlp().

**5.94.4.21 int mpi\_mod\_mpi (mpi \* R, mpi \* A, mpi \* B)**

Modulo:  $R = A \bmod B$ .

**Returns**

0 if successful, 1 if memory allocation failed, XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO if  $B == 0$

Definition at line 1189 of file bignum.c.

References mpi\_add\_mpi(), MPI\_CHK, mpi\_cmp\_int(), mpi\_cmp\_mpi(), mpi\_div\_mpi(), and mpi\_sub\_mpi().

Referenced by mpi\_exp\_mod(), mpi\_inv\_mod(), mpi\_is\_prime(), rsa\_check\_privkey(), rsa\_gen\_key(), and rsa\_private().



**5.94.4.22 int mpi\_msb (mpi \* X)**

Return the number of most significant bits.

Definition at line 200 of file bignum.c.

References `biL`, `mpi::n`, and `mpi::p`.

Referenced by `d2i_RSA_PUBKEY()`, `main()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gen_prime()`, `mpi_is_prime()`, `mpi_shift_l()`, `mpi_size()`, `mpi_write_string()`, `rsa_check_pubkey()`, `rsa_decryption()`, `rsa_encryption()`, and `rsa_gen_key()`.

**5.94.4.23 int mpi\_mul\_int (mpi \* X, mpi \* A, t\_int b)**

Baseline multiplication:  $X = A * b$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 992 of file bignum.c.

References `mpi_mul_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_div_mpi()`, and `mpi_read_string()`.

**5.94.4.24 int mpi\_mul\_mpi (mpi \* X, mpi \* A, mpi \* B)**

Baseline multiplication:  $X = A * B$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 956 of file bignum.c.

References `MPI_CHK`, `mpi_copy()`, `mpi_free()`, `mpi_grow()`, `mpi_init()`, `mpi_lset()`, `mpi_mul_hlp()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `main()`, `mpi_gcd()`, `mpi_is_prime()`, `mpi_mul_int()`, `mpi_self_test()`, `rsa_check_privkey()`, `rsa_gen_key()`, and `rsa_private()`.

**5.94.4.25 int mpi\_read\_binary (mpi \* X, unsigned char \* buf, int buflen)**

Import X from unsigned binary data, big endian.

**Parameters**

*X* destination mpi

*buf* input buffer

*buflen* input buffer size

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 484 of file bignum.c.

References CHARS\_TO\_LIMBS, ciL, MPI\_CHK, mpi\_grow(), mpi\_lset(), and mpi::p.

Referenced by asn1\_get\_mpi(), d2i\_RSA\_PUBKEY(), dhm\_read\_bignum(), dhm\_read\_public(), rsa\_private(), and rsa\_public().

#### 5.94.4.26 int mpi\_read\_file (mpi \* *X*, int *radix*, FILE \* *fin*)

Read *X* from an opened file.

##### Parameters

*X* destination mpi  
*radix* input numeric base  
*fin* input file handle

##### Returns

0 if successful, or an XYSSL\_ERR\_MPI\_XXX error code

Definition at line 391 of file bignum.c.

References mpi\_get\_digit(), mpi\_read\_string(), and XYSSL\_ERR\_MPI\_FILE\_IO\_ERROR.

Referenced by main().

#### 5.94.4.27 int mpi\_read\_mystring (mpi \* *X*, int *radix*, char \* *s*)

Definition at line 418 of file bignum.c.

References mpi\_get\_digit(), and mpi\_read\_string().

Referenced by rsa\_decryption(), and rsa\_encryption().

#### 5.94.4.28 int mpi\_read\_string (mpi \* *X*, int *radix*, char \* *s*)

Import from an ASCII string.

##### Parameters

*X* destination mpi  
*radix* input numeric base  
*s* null-terminated string buffer

##### Returns

0 if successful, or an XYSSL\_ERR\_MPI\_XXX error code

Definition at line 243 of file bignum.c.

References BITS\_TO\_LIMBS, ciL, int, mpi\_add\_int(), MPI\_CHK, mpi\_free(), mpi\_get\_digit(), mpi\_grow(), mpi\_init(), mpi\_lset(), mpi\_mul\_int(), mpi::p, mpi::s, and XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA.

Referenced by main(), mpi\_read\_file(), mpi\_read\_mystring(), mpi\_self\_test(), rsa\_self\_test(), and ssl\_set\_dh\_param().

**5.94.4.29 int mpi\_self\_test (int *verbose*)**

Checkup routine.

**Returns**

0 if successful, or 1 if the test failed

Definition at line 1854 of file bignum.c.

References MPI\_CHK, mpi\_cmp\_mpi(), mpi\_div\_mpi(), mpi\_exp\_mod(), mpi\_free(), mpi\_init(), mpi\_inv\_mod(), mpi\_mul\_mpi(), and mpi\_read\_string().

Referenced by main().

**5.94.4.30 int mpi\_shift\_l (mpi \* *X*, int *count*)**

Left-shift:  $X \ll= \text{count}$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 526 of file bignum.c.

References biL, BITS\_TO\_LIMBS, MPI\_CHK, mpi\_grow(), mpi\_msb(), mpi::n, and mpi::p.

Referenced by mpi\_div\_mpi(), mpi\_exp\_mod(), and mpi\_gen\_prime().

**5.94.4.31 int mpi\_shift\_r (mpi \* *X*, int *count*)**

Right-shift:  $X \gg= \text{count}$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 575 of file bignum.c.

References biL, mpi::n, and mpi::p.

Referenced by dhm\_make\_params(), dhm\_make\_public(), mpi\_div\_mpi(), mpi\_gcd(), mpi\_gen\_prime(), mpi\_inv\_mod(), and mpi\_is\_prime().

**5.94.4.32 int mpi\_size (mpi \* *X*)**

Return the total size in bytes.

Definition at line 218 of file bignum.c.

References mpi\_msb().

Referenced by dhm\_calc\_secret(), dhm\_make\_params(), dhm\_read\_params(), mpi\_write\_binary(), x509parse\_crt(), and x509parse\_key().

**5.94.4.33 int mpi\_sub\_abs (mpi \* X, mpi \* A, mpi \* B)**

Unsigned subtraction:  $X = |A| - |B|$ .

**Returns**

0 if successful, XYSSL\_ERR\_MPI\_NEGATIVE\_VALUE if B is greater than A

Definition at line 761 of file bignum.c.

References MPI\_CHK, mpi\_cmp\_abs(), mpi\_copy(), mpi\_free(), mpi\_init(), mpi\_sub\_hlp(), mpi::n, mpi::p, and XYSSL\_ERR\_MPI\_NEGATIVE\_VALUE.

Referenced by mpi\_add\_mpi(), mpi\_gcd(), and mpi\_sub\_mpi().

**5.94.4.34 int mpi\_sub\_int (mpi \* X, mpi \* A, int b)**

Signed subtraction:  $X = A - b$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 876 of file bignum.c.

References mpi\_sub\_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by main(), mpi\_gen\_prime(), mpi\_is\_prime(), rsa\_check\_privkey(), and rsa\_gen\_key().

**5.94.4.35 int mpi\_sub\_mpi (mpi \* X, mpi \* A, mpi \* B)**

Signed subtraction:  $X = A - B$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 829 of file bignum.c.

References mpi\_add\_abs(), MPI\_CHK, mpi\_cmp\_abs(), mpi\_sub\_abs(), and mpi::s.

Referenced by mpi\_div\_mpi(), mpi\_inv\_mod(), mpi\_mod\_mpi(), mpi\_sub\_int(), and rsa\_private().

**5.94.4.36 void mpi\_swap (mpi \* X, mpi \* Y)**

Swap the contents of X and Y.

Definition at line 155 of file bignum.c.

Referenced by rsa\_gen\_key().

**5.94.4.37 int mpi\_write\_binary (mpi \* X, unsigned char \* buf, int buflen)**

Export X into unsigned binary data, big endian.

**Parameters**

*X* source mpi  
*buf* output buffer  
*buflen* output buffer size

**Returns**

0 if successful, XYSSL\_ERR\_MPI\_BUFFER\_TOO\_SMALL if buf isn't large enough

**Note**

Call this function with \*buflen = 0 to obtain the minimum required buffer size in \*buflen.

Definition at line 506 of file bignum.c.

References ciL, mpi\_size(), mpi::p, and XYSSL\_ERR\_MPI\_BUFFER\_TOO\_SMALL.

Referenced by dhm\_calc\_secret(), dhm\_make\_public(), rsa\_private(), and rsa\_public().

**5.94.4.38 int mpi\_write\_file (char \* *p*, mpi \* *X*, int *radix*, FILE \* *fout*)**

Write X into an opened file, or stdout.

**Parameters**

*p* prefix, can be NULL  
*X* source mpi  
*radix* output numeric base  
*fout* output file handle

**Returns**

0 if successful, or an XYSSL\_ERR\_MPI\_XXX error code

**Note**

Set fout == NULL to print X on the console.

Definition at line 447 of file bignum.c.

References MPI\_CHK, mpi\_write\_string(), and XYSSL\_ERR\_MPI\_FILE\_IO\_ERROR.

Referenced by generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), and main().

**5.94.4.39 int mpi\_write\_string (mpi \* *X*, int *radix*, char \* *s*, int \* *slen*)**

Export into an ASCII string.

**Parameters**

*X* source mpi  
*radix* output numeric base  
*s* string buffer  
*slen* string buffer size

**Returns**

0 if successful, or an XYSSL\_ERR\_MPI\_XXX error code

**Note**

Call this function with \*slen = 0 to obtain the minimum required buffer size in \*slen.

Definition at line 328 of file bignum.c.

References ciL, MPI\_CHK, mpi\_copy(), mpi\_free(), mpi\_init(), mpi\_msb(), mpi\_write\_hlp(), mpi::n, mpi::p, mpi::s, XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_MPI\_BUFFER\_TOO\_SMALL.

Referenced by mpi\_write\_file().

## 5.95 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/bn\_mul.h File Reference

```
#include "xyssl/config.h"
```

### Defines

- #define [MULADDC\\_INIT](#)
- #define [MULADDC\\_CORE](#)
- #define [MULADDC\\_STOP](#) }

### 5.95.1 Detailed Description

Definition in file [bn\\_mul.h](#).

### 5.95.2 Define Documentation

#### 5.95.2.1 #define MULADDC\_CORE

**Value:**

```
s0 = ( *s << biH ) >> biH;          \
s1 = ( *s >> biH ); s++;             \
rx = s0 * b1; r0 = s0 * b0;          \
ry = s1 * b0; r1 = s1 * b1;          \
r1 += ( rx >> biH );                 \
r1 += ( ry >> biH );                 \
rx <=< biH; ry <=< biH;               \
r0 += rx; r1 += (r0 < rx);            \
r0 += ry; r1 += (r0 < ry);            \
r0 += c; r1 += (r0 < c);              \
r0 += *d; r1 += (r0 < *d);            \
c = r1; *(d++) = r0;
```

Definition at line 662 of file [bn\\_mul.h](#).

Referenced by [mpi\\_mul\\_hlp\(\)](#).

#### 5.95.2.2 #define MULADDC\_INIT

**Value:**

```
{                                     \
    t_int s0, s1, b0, b1;           \
    t_int r0, r1, rx, ry;           \
    b0 = ( b << biH ) >> biH;       \
    b1 = ( b >> biH );
```

Definition at line 655 of file [bn\\_mul.h](#).

Referenced by [mpi\\_mul\\_hlp\(\)](#).

**5.95.2.3 #define MULADDC\_STOP }**

Definition at line 676 of file bn\_mul.h.

Referenced by mpi\_mul\_hlp().



## 5.96 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/certs.h File Reference

### Variables

- char [test\\_ca\\_cert](#) []
- char [test\\_ca\\_key](#) []
- char [test\\_ca\\_pwd](#) []
- char [test\\_srv\\_cert](#) []
- char [test\\_srv\\_key](#) []
- char [test\\_cli\\_cert](#) []
- char [test\\_cli\\_key](#) []
- char [xyssl\\_ca\\_cert](#) []

### 5.96.1 Detailed Description

Definition in file [certs.h](#).

### 5.96.2 Variable Documentation

#### 5.96.2.1 char test\_ca\_cert[]

Definition at line 25 of file [certs.c](#).

Referenced by [main\(\)](#), [ssl\\_test\(\)](#), and [x509\\_self\\_test\(\)](#).

#### 5.96.2.2 char test\_ca\_key[]

Definition at line 49 of file [certs.c](#).

Referenced by [x509\\_self\\_test\(\)](#).

#### 5.96.2.3 char test\_ca\_pwd[]

Definition at line 81 of file [certs.c](#).

Referenced by [x509\\_self\\_test\(\)](#).

#### 5.96.2.4 char test\_cli\_cert[]

Definition at line 134 of file [certs.c](#).

Referenced by [main\(\)](#), and [x509\\_self\\_test\(\)](#).

#### 5.96.2.5 char test\_cli\_key[]

Definition at line 156 of file [certs.c](#).

Referenced by [main\(\)](#).

**5.96.2.6 char test\_srv\_cert[ ]**

Definition at line 83 of file certs.c.

Referenced by main(), and ssl\_test().

**5.96.2.7 char test\_srv\_key[ ]**

Definition at line 105 of file certs.c.

Referenced by main(), and ssl\_test().

**5.96.2.8 char xyssl\_ca\_cert[ ]**

Definition at line 185 of file certs.c.

Referenced by main().

## 5.97 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/debug.h File Reference

```
#include "xyssl/config.h"
#include "xyssl/ssl.h"
```

### Defines

- #define [SSL\\_DEBUG\\_MSG](#)(level, args) debug\_print\_msg( ssl, level, \_\_FILE\_\_, \_\_LINE\_\_, debug\_fmt args );
- #define [SSL\\_DEBUG\\_RET](#)(level, text, ret) debug\_print\_ret( ssl, level, \_\_FILE\_\_, \_\_LINE\_\_, text, ret );
- #define [SSL\\_DEBUG\\_BUF](#)(level, text, buf, len) debug\_print\_buf( ssl, level, \_\_FILE\_\_, \_\_LINE\_\_, text, buf, len );
- #define [SSL\\_DEBUG\\_MPI](#)(level, text, X) debug\_print\_mpi( ssl, level, \_\_FILE\_\_, \_\_LINE\_\_, text, X );
- #define [SSL\\_DEBUG\\_CRT](#)(level, text, crt) debug\_print\_crt( ssl, level, \_\_FILE\_\_, \_\_LINE\_\_, text, crt );

### Functions

- char \* [debug\\_fmt](#) (const char \*format,...)
- void [debug\\_print\\_msg](#) (ssl\_context \*ssl, int level, char \*file, int line, char \*text)
- void [debug\\_print\\_ret](#) (ssl\_context \*ssl, int level, char \*file, int line, char \*text, int ret)
- void [debug\\_print\\_buf](#) (ssl\_context \*ssl, int level, char \*file, int line, char \*text, unsigned char \*buf, int len)
- void [debug\\_print\\_mpi](#) (ssl\_context \*ssl, int level, char \*file, int line, char \*text, mpi \*X)
- void [debug\\_print\\_crt](#) (ssl\_context \*ssl, int level, char \*file, int line, char \*text, x509\_cert \*crt)

### 5.97.1 Detailed Description

Definition in file [debug.h](#).

### 5.97.2 Define Documentation

**5.97.2.1** #define [SSL\\_DEBUG\\_BUF](#)(level, text, buf, len) debug\_print\_buf( ssl, level, \_\_FILE\_\_, \_\_LINE\_\_, text, buf, len );

Definition at line 18 of file [debug.h](#).

Referenced by [ssl\\_calc\\_finished\(\)](#), [ssl\\_calc\\_verify\(\)](#), [ssl\\_decrypt\\_buf\(\)](#), [ssl\\_derive\\_keys\(\)](#), [ssl\\_encrypt\\_buf\(\)](#), [ssl\\_parse\\_client\\_hello\(\)](#), [ssl\\_parse\\_server\\_hello\(\)](#), [ssl\\_parse\\_server\\_key\\_exchange\(\)](#), [ssl\\_read\\_record\(\)](#), [ssl\\_write\\_certificate\\_request\(\)](#), [ssl\\_write\\_client\\_hello\(\)](#), [ssl\\_write\\_record\(\)](#), [ssl\\_write\\_server\\_hello\(\)](#), and [ssl\\_write\\_server\\_key\\_exchange\(\)](#).

**5.97.2.2** `#define SSL_DEBUG_CRT(level, text, crt) debug_print_crt( ssl, level, __FILE__,  
__LINE__, text, crt );`

Definition at line 24 of file debug.h.

Referenced by `ssl_parse_certificate()`, and `ssl_write_certificate()`.

**5.97.2.3** `#define SSL_DEBUG_MPI(level, text, X) debug_print_mpi( ssl, level, __FILE__,  
__LINE__, text, X );`

Definition at line 21 of file debug.h.

Referenced by `ssl_parse_client_key_exchange()`, `ssl_parse_server_key_exchange()`, `ssl_write_client_key_exchange()`, and `ssl_write_server_key_exchange()`.

**5.97.2.4** `#define SSL_DEBUG_MSG(level, args) debug_print_msg( ssl, level, __FILE__,  
__LINE__, debug_fmt args );`

Definition at line 12 of file debug.h.

Referenced by `ssl_calc_finished()`, `ssl_calc_verify()`, `ssl_close_notify()`, `ssl_decrypt_buf()`, `ssl_derive_keys()`, `ssl_encrypt_buf()`, `ssl_fetch_input()`, `ssl_flush_output()`, `ssl_free()`, `ssl_handshake()`, `ssl_handshake_client()`, `ssl_handshake_server()`, `ssl_init()`, `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `ssl_read()`, `ssl_read_record()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

**5.97.2.5** `#define SSL_DEBUG_RET(level, text, ret) debug_print_ret( ssl, level, __FILE__,  
__LINE__, text, ret );`

Definition at line 15 of file debug.h.

Referenced by `ssl_close_notify()`, `ssl_fetch_input()`, `ssl_flush_output()`, `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `ssl_read()`, `ssl_read_record()`, `ssl_set_dh_param()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

## 5.97.3 Function Documentation

**5.97.3.1** `char* debug_fmt (const char *format, ...)`

Definition at line 38 of file debug.c.

References `vsnprintf`.

**5.97.3.2 void debug\_print\_buf (ssl\_context \* ssl, int level, char \* file, int line, char \* text, unsigned char \* buf, int len)**

Definition at line 82 of file debug.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

**5.97.3.3 void debug\_print\_cert (ssl\_context \* ssl, int level, char \* file, int line, char \* text, x509\_cert \* crt)**

Definition at line 169 of file debug.c.

References `debug_print_mpi()`, `rsa_context::E`, `_ssl_context::f_dbg`, `rsa_context::N`, `_x509_cert::next`, `_ssl_context::p_dbg`, `_x509_cert::rsa`, and `x509parse_cert_info()`.

**5.97.3.4 void debug\_print\_mpi (ssl\_context \* ssl, int level, char \* file, int line, char \* text, mpi \* X)**

Definition at line 124 of file debug.c.

References `_ssl_context::f_dbg`, `mpi::n`, `mpi::p`, and `_ssl_context::p_dbg`.

Referenced by `debug_print_cert()`.

**5.97.3.5 void debug\_print\_msg (ssl\_context \* ssl, int level, char \* file, int line, char \* text)**

Definition at line 52 of file debug.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

**5.97.3.6 void debug\_print\_ret (ssl\_context \* ssl, int level, char \* file, int line, char \* text, int ret)**

Definition at line 66 of file debug.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

## 5.98 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/des.h File Reference

### Data Structures

- struct [des\\_context](#)  
*DES context structure.*
- struct [des3\\_context](#)  
*Triple-DES context structure.*

### Defines

- #define [DES\\_ENCRYPT](#) 1
- #define [DES\\_DECRYPT](#) 0

### Functions

- void [des\\_setkey\\_enc](#) ([des\\_context](#) \*ctx, unsigned char key[8])  
*DES key schedule (56-bit, encryption).*
- void [des\\_setkey\\_dec](#) ([des\\_context](#) \*ctx, unsigned char key[8])  
*DES key schedule (56-bit, decryption).*
- void [des3\\_set2key\\_enc](#) ([des3\\_context](#) \*ctx, unsigned char key[16])  
*Triple-DES key schedule (112-bit, encryption).*
- void [des3\\_set2key\\_dec](#) ([des3\\_context](#) \*ctx, unsigned char key[16])  
*Triple-DES key schedule (112-bit, decryption).*
- void [des3\\_set3key\\_enc](#) ([des3\\_context](#) \*ctx, unsigned char key[24])  
*Triple-DES key schedule (168-bit, encryption).*
- void [des3\\_set3key\\_dec](#) ([des3\\_context](#) \*ctx, unsigned char key[24])  
*Triple-DES key schedule (168-bit, decryption).*
- void [des\\_crypt\\_ecb](#) ([des\\_context](#) \*ctx, unsigned char input[8], unsigned char output[8])  
*DES-ECB block encryption/decryption.*
- void [des\\_crypt\\_cbc](#) ([des\\_context](#) \*ctx, int mode, int length, unsigned char iv[8], unsigned char \*input, unsigned char \*output)  
*DES-CBC buffer encryption/decryption.*
- void [des3\\_crypt\\_ecb](#) ([des3\\_context](#) \*ctx, unsigned char input[8], unsigned char output[8])  
*3DES-ECB block encryption/decryption*
- void [des3\\_crypt\\_cbc](#) ([des3\\_context](#) \*ctx, int mode, int length, unsigned char iv[8], unsigned char \*input, unsigned char \*output)

*3DES-CBC buffer encryption/decryption*

- [int des\\_self\\_test](#) ([int](#) verbose)

## 5.98.1 Detailed Description

Definition in file [des.h](#).

## 5.98.2 Define Documentation

### 5.98.2.1 #define DES\_DECRYPT 0

Definition at line 8 of file [des.h](#).

Referenced by [des\\_self\\_test\(\)](#), [ssl\\_decrypt\\_buf\(\)](#), and [x509\\_des3\\_decrypt\(\)](#).

### 5.98.2.2 #define DES\_ENCRYPT 1

Definition at line 7 of file [des.h](#).

Referenced by [des3\\_crypt\\_cbc\(\)](#), [des\\_crypt\\_cbc\(\)](#), [main\(\)](#), and [ssl\\_encrypt\\_buf\(\)](#).

## 5.98.3 Function Documentation

### 5.98.3.1 void des3\_crypt\_cbc (des3\_context \* *ctx*, int *mode*, int *length*, unsigned char *iv*[8], unsigned char \* *input*, unsigned char \* *output*)

3DES-CBC buffer encryption/decryption

#### Parameters

- ctx* 3DES context
- mode* DES\_ENCRYPT or DES\_DECRYPT
- length* length of the input data
- iv* initialization vector (updated after use)
- input* buffer holding the input data
- output* buffer holding the output data

Definition at line 593 of file [des.c](#).

References [des3\\_crypt\\_ecb\(\)](#), and [DES\\_ENCRYPT](#).

Referenced by [des\\_self\\_test\(\)](#), [main\(\)](#), [ssl\\_decrypt\\_buf\(\)](#), [ssl\\_encrypt\\_buf\(\)](#), and [x509\\_des3\\_decrypt\(\)](#).

### 5.98.3.2 void des3\_crypt\_ecb (des3\_context \* *ctx*, unsigned char *input*[8], unsigned char *output*[8])

3DES-ECB block encryption/decryption

#### Parameters

- ctx* 3DES context

*input* 64-bit input block

*output* 64-bit output block

Definition at line 552 of file des.c.

References DES\_FP, DES\_IP, DES\_ROUND, GET\_ULONG\_BE, PUT\_ULONG\_BE, and des3\_context::sk.

Referenced by des3\_crypt\_cbc(), and des\_self\_test().

### 5.98.3.3 void des3\_set2key\_dec (des3\_context \* ctx, unsigned char key[16])

Triple-DES key schedule (112-bit, decryption).

#### Parameters

*ctx* 3DES context to be initialized

*key* 16-byte secret key

Definition at line 420 of file des.c.

References des3\_set2key(), and des3\_context::sk.

Referenced by des\_self\_test().

### 5.98.3.4 void des3\_set2key\_enc (des3\_context \* ctx, unsigned char key[16])

Triple-DES key schedule (112-bit, encryption).

#### Parameters

*ctx* 3DES context to be initialized

*key* 16-byte secret key

Definition at line 409 of file des.c.

References des3\_set2key(), and des3\_context::sk.

Referenced by des\_self\_test().

### 5.98.3.5 void des3\_set3key\_dec (des3\_context \* ctx, unsigned char key[24])

Triple-DES key schedule (168-bit, decryption).

#### Parameters

*ctx* 3DES context to be initialized

*key* 24-byte secret key

Definition at line 465 of file des.c.

References des3\_set3key(), and des3\_context::sk.

Referenced by des\_self\_test(), ssl\_derive\_keys(), and x509\_des3\_decrypt().



**5.98.3.6 void des3\_set3key\_enc (des3\_context \* ctx, unsigned char key[24])**

Triple-DES key schedule (168-bit, encryption).

**Parameters**

*ctx* 3DES context to be initialized

*key* 24-byte secret key

Definition at line 454 of file des.c.

References des3\_set3key(), and des3\_context::sk.

Referenced by des\_self\_test(), main(), and ssl\_derive\_keys().

**5.98.3.7 void des\_crypt\_cbc (des\_context \* ctx, int mode, int length, unsigned char iv[8], unsigned char \* input, unsigned char \* output)**

DES-CBC buffer encryption/decryption.

**Parameters**

*ctx* DES context

*mode* DES\_ENCRYPT or DES\_DECRYPT

*length* length of the input data

*iv* initialization vector (updated after use)

*input* buffer holding the input data

*output* buffer holding the output data

Definition at line 505 of file des.c.

References des\_crypt\_ecb(), and DES\_ENCRYPT.

Referenced by des\_self\_test(), and main().

**5.98.3.8 void des\_crypt\_ecb (des\_context \* ctx, unsigned char input[8], unsigned char output[8])**

DES-ECB block encryption/decryption.

**Parameters**

*ctx* DES context

*input* 64-bit input block

*output* 64-bit output block

Definition at line 476 of file des.c.

References DES\_FP, DES\_IP, DES\_ROUND, GET\_ULONG\_BE, PUT\_ULONG\_BE, and des\_context::sk.

Referenced by des\_crypt\_cbc(), and des\_self\_test().

### 5.98.3.9 `int des_self_test (int verbose)`

Definition at line 694 of file `des.c`.

References `buf`, `des3_crypt_cbc()`, `des3_crypt_ecb()`, `des3_set2key_dec()`, `des3_set2key_enc()`, `des3_set3key_dec()`, `des3_set3key_enc()`, `des3_test_buf`, `des3_test_cbc_dec`, `des3_test_cbc_enc`, `des3_test_ecb_dec`, `des3_test_ecb_enc`, `des3_test_iv`, `des3_test_keys`, `des_crypt_cbc()`, `des_crypt_ecb()`, `DES_DECRYPT`, `des_setkey_dec()`, `des_setkey_enc()`, and `prv`.

Referenced by `main()`.

### 5.98.3.10 `void des_setkey_dec (des_context * ctx, unsigned char key[8])`

DES key schedule (56-bit, decryption).

#### Parameters

*ctx* DES context to be initialized

*key* 8-byte secret key

Definition at line 368 of file `des.c`.

References `des_setkey()`, `des_context::sk`, and `SWAP`.

Referenced by `des_self_test()`.

### 5.98.3.11 `void des_setkey_enc (des_context * ctx, unsigned char key[8])`

DES key schedule (56-bit, encryption).

#### Parameters

*ctx* DES context to be initialized

*key* 8-byte secret key

Definition at line 360 of file `des.c`.

References `des_setkey()`, and `des_context::sk`.

Referenced by `des_self_test()`, and `main()`.

## 5.99 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/dhm.h File Reference

```
#include "bignum.h"
```

### Data Structures

- struct [dhm\\_context](#)

### Defines

- #define [XYSSL\\_ERR\\_DHM\\_BAD\\_INPUT\\_DATA](#) -0x0480
- #define [XYSSL\\_ERR\\_DHM\\_READ\\_PARAMS\\_FAILED](#) -0x0490
- #define [XYSSL\\_ERR\\_DHM\\_MAKE\\_PARAMS\\_FAILED](#) -0x04A0
- #define [XYSSL\\_ERR\\_DHM\\_READ\\_PUBLIC\\_FAILED](#) -0x04B0
- #define [XYSSL\\_ERR\\_DHM\\_MAKE\\_PUBLIC\\_FAILED](#) -0x04C0
- #define [XYSSL\\_ERR\\_DHM\\_CALC\\_SECRET\\_FAILED](#) -0x04D0

### Functions

- [int dhm\\_read\\_params](#) ([dhm\\_context](#) \*ctx, unsigned char \*\*p, unsigned char \*end)  
*Parse the ServerKeyExchange parameters.*
- [int dhm\\_make\\_params](#) ([dhm\\_context](#) \*ctx, [int](#) s\_size, unsigned char \*output, [int](#) \*olen, [int](#)(\*f\_rng)(void \*), void \*p\_rng)  
*Setup and write the ServerKeyExchange parameters.*
- [int dhm\\_read\\_public](#) ([dhm\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*Import the peer's public value  $G^Y$ .*
- [int dhm\\_make\\_public](#) ([dhm\\_context](#) \*ctx, [int](#) s\_size, unsigned char \*output, [int](#) olen, [int](#)(\*f\_rng)(void \*), void \*p\_rng)  
*Create own private value  $X$  and export  $G^X$ .*
- [int dhm\\_calc\\_secret](#) ([dhm\\_context](#) \*ctx, unsigned char \*output, [int](#) \*olen)  
*Derive and export the shared secret  $(G^Y)^X \bmod P$ .*
- void [dhm\\_free](#) ([dhm\\_context](#) \*ctx)
- [int dhm\\_self\\_test](#) ([int](#) verbose)  
*Checkup routine.*

#### 5.99.1 Detailed Description

Definition in file [dhm.h](#).

## 5.99.2 Define Documentation

### 5.99.2.1 `#define XYSSL_ERR_DHM_BAD_INPUT_DATA -0x0480`

Definition at line 9 of file dhm.h.

Referenced by `dhm_calc_secret()`, `dhm_make_public()`, `dhm_read_bignum()`, `dhm_read_params()`, and `dhm_read_public()`.

### 5.99.2.2 `#define XYSSL_ERR_DHM_CALC_SECRET_FAILED -0x04D0`

Definition at line 14 of file dhm.h.

Referenced by `dhm_calc_secret()`.

### 5.99.2.3 `#define XYSSL_ERR_DHM_MAKE_PARAMS_FAILED -0x04A0`

Definition at line 11 of file dhm.h.

Referenced by `dhm_make_params()`.

### 5.99.2.4 `#define XYSSL_ERR_DHM_MAKE_PUBLIC_FAILED -0x04C0`

Definition at line 13 of file dhm.h.

Referenced by `dhm_make_public()`.

### 5.99.2.5 `#define XYSSL_ERR_DHM_READ_PARAMS_FAILED -0x0490`

Definition at line 10 of file dhm.h.

Referenced by `dhm_read_bignum()`.

### 5.99.2.6 `#define XYSSL_ERR_DHM_READ_PUBLIC_FAILED -0x04B0`

Definition at line 12 of file dhm.h.

Referenced by `dhm_read_public()`.

## 5.99.3 Function Documentation

### 5.99.3.1 `int dhm_calc_secret (dhm_context * ctx, unsigned char * output, int * olen)`

Derive and export the shared secret  $(G^Y)^X \bmod P$ .

#### Parameters

*ctx* DHM context

*output* destination buffer

*olen* number of chars written

**Returns**

0 if successful, or an XYSSL\_ERR\_DHM\_XXX error code

Definition at line 208 of file dhm.c.

References dhm\_context::GY, dhm\_context::K, MPI\_CHK, mpi\_exp\_mod(), mpi\_size(), mpi\_write\_binary(), dhm\_context::P, dhm\_context::RP, dhm\_context::X, XYSSL\_ERR\_DHM\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_DHM\_CALC\_SECRET\_FAILED.

Referenced by main(), ssl\_parse\_client\_key\_exchange(), and ssl\_write\_client\_key\_exchange().

**5.99.3.2 void dhm\_free (dhm\_context \* ctx)**

Definition at line 234 of file dhm.c.

References dhm\_context::G, dhm\_context::GX, dhm\_context::GY, dhm\_context::K, mpi\_free(), dhm\_context::P, dhm\_context::RP, and dhm\_context::X.

Referenced by main(), and ssl\_free().

**5.99.3.3 int dhm\_make\_params (dhm\_context \* ctx, int s\_size, unsigned char \* output, int \* olen, int(\*) (void \*) f\_rng, void \* p\_rng)**

Setup and write the ServerKeyExchange parameters.

**Parameters**

*ctx* DHM context  
*s\_size* private value size in bits  
*output* destination buffer  
*olen* number of chars written  
*f\_rng* RNG function  
*p\_rng* RNG parameter

**Note**

This function assumes that ctx->P and ctx->G have already been properly set (for example using mpi\_read\_string or mpi\_read\_binary).

**Returns**

0 if successful, or an XYSSL\_ERR\_DHM\_XXX error code

Definition at line 93 of file dhm.c.

References DHM\_MPI\_EXPORT, dhm\_context::G, dhm\_context::GX, dhm\_context::len, MPI\_CHK, mpi\_cmp\_mpi(), mpi\_exp\_mod(), mpi\_grow(), mpi\_lset(), mpi\_shift\_r(), mpi\_size(), dhm\_context::P, mpi::p, dhm\_context::RP, dhm\_context::X, and XYSSL\_ERR\_DHM\_MAKE\_PARAMS\_FAILED.

Referenced by main(), and ssl\_write\_server\_key\_exchange().

**5.99.3.4 int dhm\_make\_public (dhm\_context \* ctx, int s\_size, unsigned char \* output, int olen, int(\*) (void \*) f\_rng, void \* p\_rng)**

Create own private value X and export  $G^X$ .

**Parameters**

*ctx* DHM context  
*x\_size* private value size in bits  
*output* destination buffer  
*olen* must be equal to *ctx->P.len*  
*f\_rng* RNG function  
*p\_rng* RNG parameter

**Returns**

0 if successful, or an XYSSL\_ERR\_DHM\_XXX error code

Definition at line 167 of file dhm.c.

References *dhm\_context::G*, *dhm\_context::GX*, *dhm\_context::len*, *MPI\_CHK*, *mpi\_cmp\_mpi()*, *mpi\_exp\_mod()*, *mpi\_grow()*, *mpi\_lset()*, *mpi\_shift\_r()*, *mpi\_write\_binary()*, *dhm\_context::P*, *mpi::p*, *dhm\_context::RP*, *dhm\_context::X*, *XYSSL\_ERR\_DHM\_BAD\_INPUT\_DATA*, and *XYSSL\_ERR\_DHM\_MAKE\_PUBLIC\_FAILED*.

Referenced by *main()*, and *ssl\_write\_client\_key\_exchange()*.

**5.99.3.5 int dhm\_read\_params (dhm\_context \* ctx, unsigned char \*\* p, unsigned char \* end)**

Parse the ServerKeyExchange parameters.

**Parameters**

*ctx* DHM context  
*p* &(start of input buffer)  
*end* end of buffer

**Returns**

0 if successful, or an XYSSL\_ERR\_DHM\_XXX error code

Definition at line 63 of file dhm.c.

References *dhm\_read\_bignum()*, *dhm\_context::G*, *dhm\_context::GY*, *dhm\_context::len*, *mpi\_size()*, *dhm\_context::P*, and *XYSSL\_ERR\_DHM\_BAD\_INPUT\_DATA*.

Referenced by *main()*, and *ssl\_parse\_server\_key\_exchange()*.

**5.99.3.6 int dhm\_read\_public (dhm\_context \* ctx, unsigned char \* input, int ilen)**

Import the peer's public value  $G^Y$ .

**Parameters**

*ctx* DHM context  
*input* input buffer  
*ilen* size of buffer

### Returns

0 if successful, or an XYSSL\_ERR\_DHM\_XXX error code

Definition at line 150 of file dhm.c.

References dhm\_context::GY, dhm\_context::len, mpi\_read\_binary(), XYSSL\_ERR\_DHM\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_DHM\_READ\_PUBLIC\_FAILED.

Referenced by main(), and ssl\_parse\_client\_key\_exchange().

### 5.99.3.7 int dhm\_self\_test (int *verbose*)

Checkup routine.

### Returns

0 if successful, or 1 if the test failed

Definition at line 246 of file dhm.c.

## 5.100 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/havege.h File Reference

### Data Structures

- struct [havege\\_state](#)  
*HAVEGE state structure.*

### Defines

- #define [COLLECT\\_SIZE](#) 1024

### Functions

- void [havege\\_init](#) ([havege\\_state](#) \*hs)  
*HAVEGE initialization.*
- int [havege\\_rand](#) (void \*p\_rng)  
*HAVEGE rand function.*

#### 5.100.1 Detailed Description

Definition in file [havege.h](#).

#### 5.100.2 Define Documentation

##### 5.100.2.1 #define COLLECT\_SIZE 1024

Definition at line 7 of file [havege.h](#).

Referenced by [havege\\_fill\(\)](#), and [havege\\_rand\(\)](#).

#### 5.100.3 Function Documentation

##### 5.100.3.1 void havege\_init (havege\_state \* hs)

HAVEGE initialization.

#### Parameters

*hs* HAVEGE state to be initialized

Definition at line 188 of file [havege.c](#).

References [havege\\_fill\(\)](#).

Referenced by [generate\\_AES\\_key\(\)](#), [generate\\_RSA\\_keys\\_ciphertext\(\)](#), [generate\\_RSA\\_keys\\_plaintext\(\)](#), [initiate\\_migration\\_process\(\)](#), [main\(\)](#), and [ssl\\_test\(\)](#).



### 5.100.3.2 int havege\_rand (void \* *p\_rng*)

HAVEGE rand function.

#### Parameters

*rng\_st* points to an HAVEGE state

#### Returns

A random int

Definition at line 198 of file havege.c.

References COLLECT\_SIZE, havege\_fill(), havege\_state::offset, and havege\_state::pool.

Referenced by generate\_AES\_key(), generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), initiate\_migration\_process(), main(), and ssl\_test().

## 5.101 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md2.h File Reference

### Data Structures

- struct [md2\\_context](#)  
*MD2 context structure.*

### Functions

- void [md2\\_starts](#) ([md2\\_context](#) \*ctx)  
*MD2 context setup.*
- void [md2\\_update](#) ([md2\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*MD2 process buffer.*
- void [md2\\_finish](#) ([md2\\_context](#) \*ctx, unsigned char output[16])  
*MD2 final digest.*
- void [md2](#) (unsigned char \*input, [int](#) ilen, unsigned char output[16])  
*Output = MD2( input buffer ).*
- [int](#) [md2\\_file](#) (char \*path, unsigned char output[16])  
*Output = MD2( file contents ).*
- void [md2\\_hmac\\_starts](#) ([md2\\_context](#) \*ctx, unsigned char \*key, [int](#) keylen)  
*MD2 HMAC context setup.*
- void [md2\\_hmac\\_update](#) ([md2\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*MD2 HMAC process buffer.*
- void [md2\\_hmac\\_finish](#) ([md2\\_context](#) \*ctx, unsigned char output[16])  
*MD2 HMAC final digest.*
- void [md2\\_hmac](#) (unsigned char \*key, [int](#) keylen, unsigned char \*input, [int](#) ilen, unsigned char output[16])  
*Output = HMAC-MD2( hmac key, input buffer ).*
- [int](#) [md2\\_self\\_test](#) ([int](#) verbose)  
*Checkup routine.*

### 5.101.1 Detailed Description

Definition in file [md2.h](#).

## 5.101.2 Function Documentation

### 5.101.2.1 void md2 (unsigned char \* *input*, int *ilen*, unsigned char *output*[16])

Output = MD2( input buffer ).

#### Parameters

*input* buffer holding the data  
*ilen* length of the input data  
*output* MD2 checksum result

Referenced by x509\_hash().

### 5.101.2.2 int md2\_file (char \* *path*, unsigned char *output*[16])

Output = MD2( file contents ).

#### Parameters

*path* input file name  
*output* MD2 checksum result

#### Returns

0 if successful, 1 if fopen failed, or 2 if fread failed

### 5.101.2.3 void md2\_finish (md2\_context \* *ctx*, unsigned char *output*[16])

MD2 final digest.

#### Parameters

*ctx* MD2 context  
*output* MD2 checksum result

### 5.101.2.4 void md2\_hmac (unsigned char \* *key*, int *keylen*, unsigned char \* *input*, int *ilen*, unsigned char *output*[16])

Output = HMAC-MD2( hmac key, input buffer ).

#### Parameters

*key* HMAC secret key  
*keylen* length of the HMAC key  
*input* buffer holding the data  
*ilen* length of the input data  
*output* HMAC-MD2 result

**5.101.2.5 void md2\_hmac\_finish (md2\_context \* *ctx*, unsigned char *output*[16])**

MD2 HMAC final digest.

**Parameters**

*ctx* HMAC context  
*output* MD2 HMAC checksum result

**5.101.2.6 void md2\_hmac\_starts (md2\_context \* *ctx*, unsigned char \* *key*, int *keylen*)**

MD2 HMAC context setup.

**Parameters**

*ctx* HMAC context to be initialized  
*key* HMAC secret key  
*keylen* length of the HMAC key

**5.101.2.7 void md2\_hmac\_update (md2\_context \* *ctx*, unsigned char \* *input*, int *ilen*)**

MD2 HMAC process buffer.

**Parameters**

*ctx* HMAC context  
*input* buffer holding the data  
*ilen* length of the input data

**5.101.2.8 int md2\_self\_test (int *verbose*)**

Checkup routine.

**Returns**

0 if successful, or 1 if the test failed

Referenced by main().

**5.101.2.9 void md2\_starts (md2\_context \* *ctx*)**

MD2 context setup.

**Parameters**

*ctx* context to be initialized

#### 5.101.2.10 void md2\_update (md2\_context \* *ctx*, unsigned char \* *input*, int *ilen*)

MD2 process buffer.

##### Parameters

*ctx* MD2 context

*input* buffer holding the data

*ilen* length of the input data

## 5.102 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md4.h File Reference

### Data Structures

- struct [md4\\_context](#)  
*MD4 context structure.*

### Functions

- void [md4\\_starts](#) ([md4\\_context](#) \*ctx)  
*MD4 context setup.*
- void [md4\\_update](#) ([md4\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*MD4 process buffer.*
- void [md4\\_finish](#) ([md4\\_context](#) \*ctx, unsigned char output[16])  
*MD4 final digest.*
- void [md4](#) (unsigned char \*input, [int](#) ilen, unsigned char output[16])  
*Output = MD4( input buffer ).*
- [int](#) [md4\\_file](#) (char \*path, unsigned char output[16])  
*Output = MD4( file contents ).*
- void [md4\\_hmac\\_starts](#) ([md4\\_context](#) \*ctx, unsigned char \*key, [int](#) keylen)  
*MD4 HMAC context setup.*
- void [md4\\_hmac\\_update](#) ([md4\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*MD4 HMAC process buffer.*
- void [md4\\_hmac\\_finish](#) ([md4\\_context](#) \*ctx, unsigned char output[16])  
*MD4 HMAC final digest.*
- void [md4\\_hmac](#) (unsigned char \*key, [int](#) keylen, unsigned char \*input, [int](#) ilen, unsigned char output[16])  
*Output = HMAC-MD4( hmac key, input buffer ).*
- [int](#) [md4\\_self\\_test](#) ([int](#) verbose)  
*Checkup routine.*

### 5.102.1 Detailed Description

Definition in file [md4.h](#).

## 5.102.2 Function Documentation

### 5.102.2.1 void md4 (unsigned char \* *input*, int *ilen*, unsigned char *output*[16])

Output = MD4( input buffer ).

#### Parameters

*input* buffer holding the data  
*ilen* length of the input data  
*output* MD4 checksum result

Referenced by main(), and x509\_hash().

### 5.102.2.2 int md4\_file (char \* *path*, unsigned char *output*[16])

Output = MD4( file contents ).

#### Parameters

*path* input file name  
*output* MD4 checksum result

#### Returns

0 if successful, 1 if fopen failed, or 2 if fread failed

### 5.102.2.3 void md4\_finish (md4\_context \* *ctx*, unsigned char *output*[16])

MD4 final digest.

#### Parameters

*ctx* MD4 context  
*output* MD4 checksum result

### 5.102.2.4 void md4\_hmac (unsigned char \* *key*, int *keylen*, unsigned char \* *input*, int *ilen*, unsigned char *output*[16])

Output = HMAC-MD4( hmac key, input buffer ).

#### Parameters

*key* HMAC secret key  
*keylen* length of the HMAC key  
*input* buffer holding the data  
*ilen* length of the input data  
*output* HMAC-MD4 result

**5.102.2.5 void md4\_hmac\_finish (md4\_context \* *ctx*, unsigned char *output*[16])**

MD4 HMAC final digest.

**Parameters**

*ctx* HMAC context  
*output* MD4 HMAC checksum result

**5.102.2.6 void md4\_hmac\_starts (md4\_context \* *ctx*, unsigned char \* *key*, int *keylen*)**

MD4 HMAC context setup.

**Parameters**

*ctx* HMAC context to be initialized  
*key* HMAC secret key  
*keylen* length of the HMAC key

**5.102.2.7 void md4\_hmac\_update (md4\_context \* *ctx*, unsigned char \* *input*, int *ilen*)**

MD4 HMAC process buffer.

**Parameters**

*ctx* HMAC context  
*input* buffer holding the data  
*ilen* length of the input data

**5.102.2.8 int md4\_self\_test (int *verbose*)**

Checkup routine.

**Returns**

0 if successful, or 1 if the test failed

Referenced by main().

**5.102.2.9 void md4\_starts (md4\_context \* *ctx*)**

MD4 context setup.

**Parameters**

*ctx* context to be initialized



#### 5.102.2.10 void md4\_update (md4\_context \* *ctx*, unsigned char \* *input*, int *ilen*)

MD4 process buffer.

##### Parameters

*ctx* MD4 context

*input* buffer holding the data

*ilen* length of the input data

## 5.103 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md5.h File Reference

### Data Structures

- struct [md5\\_context](#)  
*MD5 context structure.*

### Functions

- void [md5\\_starts](#) ([md5\\_context](#) \*ctx)  
*MD5 context setup.*
- void [md5\\_update](#) ([md5\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*MD5 process buffer.*
- void [md5\\_finish](#) ([md5\\_context](#) \*ctx, unsigned char output[16])  
*MD5 final digest.*
- void [md5](#) (unsigned char \*input, [int](#) ilen, unsigned char output[16])  
*Output = MD5( input buffer ).*
- [int](#) [md5\\_file](#) (char \*path, unsigned char output[16])  
*Output = MD5( file contents ).*
- void [md5\\_hmac\\_starts](#) ([md5\\_context](#) \*ctx, unsigned char \*key, [int](#) keylen)  
*MD5 HMAC context setup.*
- void [md5\\_hmac\\_update](#) ([md5\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*MD5 HMAC process buffer.*
- void [md5\\_hmac\\_finish](#) ([md5\\_context](#) \*ctx, unsigned char output[16])  
*MD5 HMAC final digest.*
- void [md5\\_hmac](#) (unsigned char \*key, [int](#) keylen, unsigned char \*input, [int](#) ilen, unsigned char output[16])  
*Output = HMAC-MD5( hmac key, input buffer ).*
- [int](#) [md5\\_self\\_test](#) ([int](#) verbose)  
*Checkup routine.*

### 5.103.1 Detailed Description

Definition in file [md5.h](#).

## 5.103.2 Function Documentation

### 5.103.2.1 void md5 (unsigned char \* *input*, int *ilen*, unsigned char *output*[16])

Output = MD5( *input* buffer ).

#### Parameters

*input* buffer holding the data  
*ilen* length of the input data  
*output* MD5 checksum result

Definition at line 278 of file md5.c.

References md5\_finish(), md5\_starts(), and md5\_update().

Referenced by initiate\_migration\_process(), main(), md5\_hmac\_starts(), md5\_self\_test(), reply\_migration\_process(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_md5(), ssl\_parse\_finished(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_finished(), ssl\_write\_server\_key\_exchange(), and x509\_hash().

### 5.103.2.2 int md5\_file (char \* *path*, unsigned char *output*[16])

Output = MD5( *file* contents ).

#### Parameters

*path* input file name  
*output* MD5 checksum result

#### Returns

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 292 of file md5.c.

References buf, f, md5\_finish(), md5\_starts(), and md5\_update().

Referenced by md5\_wrapper().

### 5.103.2.3 void md5\_finish (md5\_context \* *ctx*, unsigned char *output*[16])

MD5 final digest.

#### Parameters

*ctx* MD5 context  
*output* MD5 checksum result

Definition at line 250 of file md5.c.

References md5\_padding, md5\_update(), PUT\_ULONG\_LE, md5\_context::state, and md5\_context::total.

Referenced by md5(), md5\_file(), md5\_hmac\_finish(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_md5(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_server\_key\_exchange(), and x509\_des3\_decrypt().

#### 5.103.2.4 void md5\_hmac (unsigned char \* *key*, int *keylen*, unsigned char \* *input*, int *ilen*, unsigned char *output*[16])

Output = HMAC-MD5( hmac key, input buffer ).

##### Parameters

*key* HMAC secret key  
*keylen* length of the HMAC key  
*input* buffer holding the data  
*ilen* length of the input data  
*output* HMAC-MD5 result

Definition at line 378 of file md5.c.

References md5\_hmac\_finish(), md5\_hmac\_starts(), and md5\_hmac\_update().

Referenced by ssl\_decrypt\_buf(), ssl\_encrypt\_buf(), and tls1\_prf().

#### 5.103.2.5 void md5\_hmac\_finish (md5\_context \* *ctx*, unsigned char *output*[16])

MD5 HMAC final digest.

##### Parameters

*ctx* HMAC context  
*output* MD5 HMAC checksum result

Definition at line 362 of file md5.c.

References md5\_finish(), md5\_starts(), md5\_update(), and md5\_context::opad.

Referenced by md5\_hmac(), and md5\_self\_test().

#### 5.103.2.6 void md5\_hmac\_starts (md5\_context \* *ctx*, unsigned char \* *key*, int *keylen*)

MD5 HMAC context setup.

##### Parameters

*ctx* HMAC context to be initialized  
*key* HMAC secret key  
*keylen* length of the HMAC key

Definition at line 324 of file md5.c.

References md5\_context::ipad, md5(), md5\_starts(), md5\_update(), and md5\_context::opad.

Referenced by md5\_hmac(), and md5\_self\_test().

**5.103.2.7 void md5\_hmac\_update (md5\_context \* *ctx*, unsigned char \* *input*, int *ilen*)**

MD5 HMAC process buffer.

**Parameters**

*ctx* HMAC context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 354 of file md5.c.

References md5\_update().

Referenced by md5\_hmac(), and md5\_self\_test().

**5.103.2.8 int md5\_self\_test (int *verbose*)**

Checksum routine.

**Returns**

0 if successful, or 1 if the test failed

Definition at line 495 of file md5.c.

References buf, md5(), md5\_hmac\_finish(), md5\_hmac\_starts(), md5\_hmac\_test\_buf, md5\_hmac\_test\_buflen, md5\_hmac\_test\_key, md5\_hmac\_test\_keylen, md5\_hmac\_test\_sum, md5\_hmac\_update(), md5\_test\_buf, md5\_test\_buflen, and md5\_test\_sum.

Referenced by main().

**5.103.2.9 void md5\_starts (md5\_context \* *ctx*)**

MD5 context setup.

**Parameters**

*ctx* context to be initialized

Definition at line 61 of file md5.c.

References md5\_context::state, and md5\_context::total.

Referenced by md5(), md5\_file(), md5\_hmac\_finish(), md5\_hmac\_starts(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_init(), ssl\_mac\_md5(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_server\_key\_exchange(), and x509\_des3\_decrypt().

**5.103.2.10 void md5\_update (md5\_context \* *ctx*, unsigned char \* *input*, int *ilen*)**

MD5 process buffer.

**Parameters**

*ctx* MD5 context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 198 of file md5.c.

References md5\_context::buffer, md5\_process(), and md5\_context::total.

Referenced by md5(), md5\_file(), md5\_finish(), md5\_hmac\_finish(), md5\_hmac\_starts(), md5\_hmac\_update(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_md5(), ssl\_parse\_client\_hello(), ssl\_parse\_server\_key\_exchange(), ssl\_read\_record(), ssl\_write\_record(), ssl\_write\_server\_key\_exchange(), and x509\_des3\_decrypt().

## 5.104 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/net.h File Reference

### Defines

- `#define XYSSL_ERR_NET_UNKNOWN_HOST -0x0F00`
- `#define XYSSL_ERR_NET_SOCKET_FAILED -0x0F10`
- `#define XYSSL_ERR_NET_CONNECT_FAILED -0x0F20`
- `#define XYSSL_ERR_NET_BIND_FAILED -0x0F30`
- `#define XYSSL_ERR_NET_LISTEN_FAILED -0x0F40`
- `#define XYSSL_ERR_NET_ACCEPT_FAILED -0x0F50`
- `#define XYSSL_ERR_NET_RECV_FAILED -0x0F60`
- `#define XYSSL_ERR_NET_SEND_FAILED -0x0F70`
- `#define XYSSL_ERR_NET_CONN_RESET -0x0F80`
- `#define XYSSL_ERR_NET_TRY_AGAIN -0x0F90`

### Functions

- `int net_connect (int *fd, char *host, int port)`  
*Initiate a TCP connection with host:port.*
- `int net_bind (int *fd, char *bind_ip, int port)`  
*Create a listening socket on bind\_ip:port. If bind\_ip == NULL, all interfaces are binded.*
- `int net_accept (int bind_fd, int *client_fd, void *client_ip)`  
*Accept a connection from a remote client.*
- `int net_set_block (int fd)`  
*Set the socket blocking.*
- `int net_set_nonblock (int fd)`  
*Set the socket non-blocking.*
- `void net_usleep (unsigned long usec)`  
*Portable usleep helper.*
- `int net_recv (void *ctx, unsigned char *buf, int len)`  
*Read at most 'len' characters. len is updated to reflect the actual number of characters read.*
- `int net_send (void *ctx, unsigned char *buf, int len)`  
*Write at most 'len' characters. len is updated to reflect the number of characters \_not\_ written.*
- `void net_close (int fd)`  
*Gracefully shutdown the connection.*

### 5.104.1 Detailed Description

Definition in file [net.h](#).

### 5.104.2 Define Documentation

#### 5.104.2.1 **#define XYSSL\_ERR\_NET\_ACCEPT\_FAILED -0x0F50**

Definition at line 12 of file net.h.

Referenced by net\_accept().

#### 5.104.2.2 **#define XYSSL\_ERR\_NET\_BIND\_FAILED -0x0F30**

Definition at line 10 of file net.h.

Referenced by net\_bind().

#### 5.104.2.3 **#define XYSSL\_ERR\_NET\_CONN\_RESET -0x0F80**

Definition at line 15 of file net.h.

Referenced by main(), net\_recv(), net\_send(), and ssl\_test().

#### 5.104.2.4 **#define XYSSL\_ERR\_NET\_CONNECT\_FAILED -0x0F20**

Definition at line 9 of file net.h.

Referenced by net\_connect().

#### 5.104.2.5 **#define XYSSL\_ERR\_NET\_LISTEN\_FAILED -0x0F40**

Definition at line 11 of file net.h.

Referenced by net\_bind().

#### 5.104.2.6 **#define XYSSL\_ERR\_NET\_RECV\_FAILED -0x0F60**

Definition at line 13 of file net.h.

Referenced by net\_recv().

#### 5.104.2.7 **#define XYSSL\_ERR\_NET\_SEND\_FAILED -0x0F70**

Definition at line 14 of file net.h.

Referenced by net\_send().

#### 5.104.2.8 **#define XYSSL\_ERR\_NET\_SOCKET\_FAILED -0x0F10**

Definition at line 8 of file net.h.

Referenced by net\_bind(), and net\_connect().



#### 5.104.2.9 `#define XYSSL_ERR_NET_TRY_AGAIN -0x0F90`

Definition at line 16 of file net.h.

Referenced by `main()`, `net_accept()`, `net_recv()`, `net_send()`, and `ssl_test()`.

#### 5.104.2.10 `#define XYSSL_ERR_NET_UNKNOWN_HOST -0x0F00`

Definition at line 7 of file net.h.

Referenced by `net_connect()`.

### 5.104.3 Function Documentation

#### 5.104.3.1 `int net_accept (int bind_fd, int * client_fd, void * client_ip)`

Accept a connection from a remote client.

##### Returns

0 if successful, `XYSSL_ERR_NET_ACCEPT_FAILED`, or `XYSSL_ERR_NET_WOULD_BLOCK` if `bind_fd` was set to non-blocking and `accept()` is blocking.

Definition at line 212 of file net.c.

References `int`, `net_is_blocking()`, `XYSSL_ERR_NET_ACCEPT_FAILED`, and `XYSSL_ERR_NET_TRY_AGAIN`.

Referenced by `main()`, and `ssl_test()`.

#### 5.104.3.2 `int net_bind (int * fd, char * bind_ip, int port)`

Create a listening socket on `bind_ip:port`. If `bind_ip == NULL`, all interfaces are binded.

##### Returns

0 if successful, or one of: `XYSSL_ERR_NET_SOCKET_FAILED`, `XYSSL_ERR_NET_BIND_FAILED`, `XYSSL_ERR_NET_LISTEN_FAILED`

Definition at line 126 of file net.c.

References `net_htons()`, `SOCKET_ERROR`, `XYSSL_ERR_NET_BIND_FAILED`, `XYSSL_ERR_NET_LISTEN_FAILED`, and `XYSSL_ERR_NET_SOCKET_FAILED`.

Referenced by `main()`, and `ssl_test()`.

#### 5.104.3.3 `void net_close (int fd)`

Gracefully shutdown the connection.

Definition at line 338 of file net.c.

Referenced by `main()`, and `ssl_test()`.

#### 5.104.3.4 `int net_connect (int *fd, char *host, int port)`

Initiate a TCP connection with host:port.

##### Returns

0 if successful, or one of: `XYSSL_ERR_NET_SOCKET_FAILED`, `XYSSL_ERR_NET_UNKNOWN_HOST`, `XYSSL_ERR_NET_CONNECT_FAILED`

Definition at line 81 of file net.c.

References `net_htons()`, `SOCKET_ERROR`, `XYSSL_ERR_NET_CONNECT_FAILED`, `XYSSL_ERR_NET_SOCKET_FAILED`, and `XYSSL_ERR_NET_UNKNOWN_HOST`.

Referenced by `main()`, and `ssl_test()`.

#### 5.104.3.5 `int net_recv (void *ctx, unsigned char *buf, int len)`

Read at most 'len' characters. len is updated to reflect the actual number of characters read.

##### Returns

This function returns the number of bytes received, or a negative error code; `XYSSL_ERR_NET_TRY_AGAIN` indicates `read()` is blocking.

Definition at line 277 of file net.c.

References `net_is_blocking()`, `XYSSL_ERR_NET_CONN_RESET`, `XYSSL_ERR_NET_RECV_FAILED`, and `XYSSL_ERR_NET_TRY_AGAIN`.

Referenced by `main()`, and `ssl_test()`.

#### 5.104.3.6 `int net_send (void *ctx, unsigned char *buf, int len)`

Write at most 'len' characters. len is updated to reflect the number of characters `_not_` written.

##### Returns

This function returns the number of bytes sent, or a negative error code; `XYSSL_ERR_NET_TRY_AGAIN` indicates `write()` is blocking.

Definition at line 309 of file net.c.

References `net_is_blocking()`, `XYSSL_ERR_NET_CONN_RESET`, `XYSSL_ERR_NET_SEND_FAILED`, and `XYSSL_ERR_NET_TRY_AGAIN`.

Referenced by `main()`, and `ssl_test()`.

#### 5.104.3.7 `int net_set_block (int fd)`

Set the socket blocking.

##### Returns

0 if successful, or a non-zero error code

Definition at line 243 of file net.c.

#### 5.104.3.8 `int net_set_nonblock (int fd)`

Set the socket non-blocking.

##### Returns

0 if successful, or a non-zero error code

Definition at line 253 of file net.c.

Referenced by ssl\_test().

#### 5.104.3.9 `void net_usleep (unsigned long usec)`

Portable usleep helper.

##### Note

Real amount of time slept will not be less than select()'s timeout granularity (typically, 10ms).

Definition at line 266 of file net.c.

## 5.105 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/openssl.h File Reference

```
#include "xyssl/aes.h"
#include "xyssl/md5.h"
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
```

### Defines

- #define [AES\\_SIZE](#) 16
- #define [AES\\_BLOCK\\_SIZE](#) 16
- #define [AES\\_KEY](#) aes\_context
- #define [MD5\\_CTX](#) md5\_context
- #define [SHA\\_CTX](#) sha1\_context
- #define [SHA1\\_Init](#)(CTX) sha1\_starts( (CTX) )
- #define [SHA1\\_Update](#)(CTX, BUF, LEN) sha1\_update( (CTX), (unsigned char \*) (BUF), (LEN) )
- #define [SHA1\\_Final](#)(OUT, CTX) sha1\_finish( (CTX), (OUT) )
- #define [MD5\\_Init](#)(CTX) md5\_starts( (CTX) )
- #define [MD5\\_Update](#)(CTX, BUF, LEN) md5\_update( (CTX), (unsigned char \*) (BUF), (LEN) )
- #define [MD5\\_Final](#)(OUT, CTX) md5\_finish( (CTX), (OUT) )
- #define [AES\\_set\\_encrypt\\_key](#)(KEY, KEYSIZE, CTX) aes\_setkey\_enc( (CTX), (KEY), (KEYSIZE) )
- #define [AES\\_set\\_decrypt\\_key](#)(KEY, KEYSIZE, CTX) aes\_setkey\_dec( (CTX), (KEY), (KEYSIZE) )
- #define [AES\\_cbc\\_encrypt](#)(INPUT, OUTPUT, LEN, CTX, IV, MODE) aes\_crypt\_cbc( (CTX), (MODE), (LEN), (IV), (INPUT), (OUTPUT) )
- #define [RSA](#) rsa\_context
- #define [RSA\\_PKCS1\\_PADDING](#) 1
- #define [RSA\\_size](#)(CTX) (CTX)->len
- #define [RSA\\_free](#)(CTX) rsa\_free( CTX )
- #define [ERR\\_get\\_error](#)() "ERR\_get\_error() not supported"
- #define [RSA\\_blinding\\_off](#)(IGNORE)
- #define [d2i\\_RSAPrivateKey](#)(a, b, c) new rsa\_context

### Functions

- [int \\_\\_RSA\\_Passthrough](#) (void \*output, void \*input, [int size](#))
- [rsa\\_context \\* d2i\\_RSA\\_PUBKEY](#) (void \*ignore, unsigned char \*\*bufptr, [int len](#))
- [int RSA\\_public\\_decrypt](#) ([int size](#), unsigned char \*input, unsigned char \*output, RSA \*key, [int ignore](#))
- [int RSA\\_private\\_decrypt](#) ([int size](#), unsigned char \*input, unsigned char \*output, RSA \*key, [int ignore](#))
- [int RSA\\_public\\_encrypt](#) ([int size](#), unsigned char \*input, unsigned char \*output, RSA \*key, [int ignore](#))
- [int RSA\\_private\\_encrypt](#) ([int size](#), unsigned char \*input, unsigned char \*output, RSA \*key, [int ignore](#))

### 5.105.1 Detailed Description

Definition in file [openssl.h](#).

### 5.105.2 Define Documentation

#### 5.105.2.1 #define AES\_BLOCK\_SIZE 16

Definition at line 16 of file openssl.h.

#### 5.105.2.2 #define AES\_cbc\_encrypt(INPUT, OUTPUT, LEN, CTX, IV, MODE) aes\_crypt\_cbc( (CTX), (MODE), (LEN), (IV), (INPUT), (OUTPUT) )

Definition at line 39 of file openssl.h.

#### 5.105.2.3 #define AES\_KEY aes\_context

Definition at line 17 of file openssl.h.

#### 5.105.2.4 #define AES\_set\_decrypt\_key(KEY, KEYSIZE, CTX) aes\_setkey\_dec( (CTX), (KEY), (KEYSIZE) )

Definition at line 37 of file openssl.h.

#### 5.105.2.5 #define AES\_set\_encrypt\_key(KEY, KEYSIZE, CTX) aes\_setkey\_enc( (CTX), (KEY), (KEYSIZE) )

Definition at line 35 of file openssl.h.

#### 5.105.2.6 #define AES\_SIZE 16

Definition at line 15 of file openssl.h.

#### 5.105.2.7 #define d2i\_RSAPrivateKey(a, b, c) new rsa\_context

Definition at line 102 of file openssl.h.

#### 5.105.2.8 #define ERR\_get\_error() "ERR\_get\_error() not supported"

Definition at line 99 of file openssl.h.

#### 5.105.2.9 #define MD5\_CTX md5\_context

Definition at line 18 of file openssl.h.

**5.105.2.10 #define MD5\_Final(OUT, CTX) md5\_finish( (CTX), (OUT) )**

Definition at line 32 of file openssl.h.

**5.105.2.11 #define MD5\_Init(CTX) md5\_starts( (CTX) )**

Definition at line 28 of file openssl.h.

**5.105.2.12 #define MD5\_Update(CTX, BUF, LEN) md5\_update( (CTX), (unsigned char \*) (BUF), (LEN) )**

Definition at line 30 of file openssl.h.

**5.105.2.13 #define RSA\_rsa\_context**

Definition at line 95 of file openssl.h.

**5.105.2.14 #define RSA\_blinding\_off(IGNORE)**

Definition at line 100 of file openssl.h.

**5.105.2.15 #define RSA\_free(CTX) rsa\_free( CTX )**

Definition at line 98 of file openssl.h.

**5.105.2.16 #define RSA\_PKCS1\_PADDING 1**

Definition at line 96 of file openssl.h.

**5.105.2.17 #define RSA\_size(CTX) (CTX)->len**

Definition at line 97 of file openssl.h.

Referenced by RSA\_private\_encrypt(), and RSA\_public\_encrypt().

**5.105.2.18 #define SHA1\_Final(OUT, CTX) sha1\_finish( (CTX), (OUT) )**

Definition at line 25 of file openssl.h.

**5.105.2.19 #define SHA1\_Init(CTX) sha1\_starts( (CTX) )**

Definition at line 21 of file openssl.h.

**5.105.2.20 #define SHA1\_Update(CTX, BUF, LEN) sha1\_update( (CTX), (unsigned char \*) (BUF), (LEN) )**

Definition at line 23 of file openssl.h.

**5.105.2.21 #define SHA\_CTX sha1\_context**

Definition at line 19 of file openssl.h.

**5.105.3 Function Documentation****5.105.3.1 int \_\_RSA\_Passthrough (void \* *output*, void \* *input*, int *size*) [inline]**

Definition at line 45 of file openssl.h.

**5.105.3.2 rsa\_context\* d2i\_RSA\_PUBKEY (void \* *ignore*, unsigned char \*\* *bufptr*, int *len*) [inline]**

Definition at line 51 of file openssl.h.

References `rsa_context::E`, `rsa_context::len`, `mpi_msb()`, `mpi_read_binary()`, and `rsa_context::N`.

**5.105.3.3 int RSA\_private\_decrypt (int *size*, unsigned char \* *input*, unsigned char \* *output*, RSA \* *key*, int *ignore*) [inline]**

Definition at line 105 of file openssl.h.

References `rsa_pkcs1_decrypt()`, and `RSA_PRIVATE`.

**5.105.3.4 int RSA\_private\_encrypt (int *size*, unsigned char \* *input*, unsigned char \* *output*, RSA \* *key*, int *ignore*) [inline]**

Definition at line 107 of file openssl.h.

References `rsa_pkcs1_encrypt()`, `RSA_PRIVATE`, and `RSA_size`.

**5.105.3.5 int RSA\_public\_decrypt (int *size*, unsigned char \* *input*, unsigned char \* *output*, RSA \* *key*, int *ignore*) [inline]**

Definition at line 104 of file openssl.h.

References `rsa_pkcs1_decrypt()`, and `RSA_PUBLIC`.

**5.105.3.6 int RSA\_public\_encrypt (int *size*, unsigned char \* *input*, unsigned char \* *output*, RSA \* *key*, int *ignore*) [inline]**

Definition at line 106 of file openssl.h.

References `rsa_pkcs1_encrypt()`, `RSA_PUBLIC`, and `RSA_size`.

## 5.106 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/padlock.h File Reference

```
#include "xyssl/aes.h"
```

### 5.106.1 Detailed Description

Definition in file [padlock.h](#).



## 5.107 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/rsa.h File Reference

```
#include "bignum.h"
```

### Data Structures

- struct [rsa\\_context](#)  
*RSA context structure.*

### Defines

- #define [XYSSL\\_ERR\\_RSA\\_BAD\\_INPUT\\_DATA](#) -0x0400
- #define [XYSSL\\_ERR\\_RSA\\_INVALID\\_PADDING](#) -0x0410
- #define [XYSSL\\_ERR\\_RSA\\_KEY\\_GEN\\_FAILED](#) -0x0420
- #define [XYSSL\\_ERR\\_RSA\\_KEY\\_CHECK\\_FAILED](#) -0x0430
- #define [XYSSL\\_ERR\\_RSA\\_PUBLIC\\_FAILED](#) -0x0440
- #define [XYSSL\\_ERR\\_RSA\\_PRIVATE\\_FAILED](#) -0x0450
- #define [XYSSL\\_ERR\\_RSA\\_VERIFY\\_FAILED](#) -0x0460
- #define [RSA\\_RAW](#) 0
- #define [RSA\\_MD2](#) 2
- #define [RSA\\_MD4](#) 3
- #define [RSA\\_MD5](#) 4
- #define [RSA\\_SHA1](#) 5
- #define [RSA\\_SHA256](#) 6
- #define [RSA\\_PUBLIC](#) 0
- #define [RSA\\_PRIVATE](#) 1
- #define [RSA\\_PKCS\\_V15](#) 0
- #define [RSA\\_PKCS\\_V21](#) 1
- #define [RSA\\_SIGN](#) 1
- #define [RSA\\_CRYPT](#) 2
- #define [ASN1\\_HASH\\_MDX](#)
- #define [ASN1\\_HASH\\_SHA1](#)

### Functions

- void [rsa\\_init](#) ([rsa\\_context](#) \*ctx, int padding, int hash\_id, int(\*f\_rng)(void \*), void \*p\_rng)  
*Initialize an RSA context.*
- int [rsa\\_gen\\_key](#) ([rsa\\_context](#) \*ctx, int nbits, int exponent)  
*Generate an RSA keypair.*
- int [rsa\\_check\\_pubkey](#) ([rsa\\_context](#) \*ctx)  
*Check a public RSA key.*
- int [rsa\\_check\\_privkey](#) ([rsa\\_context](#) \*ctx)  
*Check a private RSA key.*

- `int rsa_public (rsa_context *ctx, unsigned char *input, unsigned char *output)`  
*Do an RSA public key operation.*
- `int rsa_private (rsa_context *ctx, unsigned char *input, unsigned char *output)`  
*Do an RSA private key operation.*
- `int rsa_pkcs1_encrypt (rsa_context *ctx, int mode, int ilen, unsigned char *input, unsigned char *output)`  
*Add the message padding, then do an RSA operation.*
- `int rsa_pkcs1_decrypt (rsa_context *ctx, int mode, int *olen, unsigned char *input, unsigned char *output)`  
*Do an RSA operation, then remove the message padding.*
- `int rsa_pkcs1_sign (rsa_context *ctx, int mode, int hash_id, int hashlen, unsigned char *hash, unsigned char *sig)`  
*Do a private RSA to sign a message digest.*
- `int rsa_pkcs1_verify (rsa_context *ctx, int mode, int hash_id, int hashlen, unsigned char *hash, unsigned char *sig)`  
*Do a public RSA and check the message digest.*
- `void rsa_free (rsa_context *ctx)`  
*Free the components of an RSA key.*
- `int rsa_self_test (int verbose)`  
*Checkup routine.*

### 5.107.1 Detailed Description

Definition in file [rsa.h](#).

### 5.107.2 Define Documentation

#### 5.107.2.1 #define ASN1\_HASH\_MD5

**Value:**

```
"\x30\x20\x30\x0C\x06\x08\x2A\x86\x48" \
"\x86\xF7\x0D\x02\x00\x05\x00\x04\x10"
```

Definition at line 45 of file [rsa.h](#).

Referenced by [rsa\\_pkcs1\\_sign\(\)](#), and [rsa\\_pkcs1\\_verify\(\)](#).

### 5.107.2.2 #define ASN1\_HASH\_SHA1

Value:

```
"\x30\x21\x30\x09\x06\x05\x2B\x0E\x03" \
"\x02\x1A\x05\x00\x04\x14"
```

Definition at line 49 of file rsa.h.

Referenced by rsa\_pkcs1\_sign(), and rsa\_pkcs1\_verify().

### 5.107.2.3 #define RSA\_CRYPT 2

Definition at line 34 of file rsa.h.

Referenced by rsa\_pkcs1\_decrypt(), and rsa\_pkcs1\_encrypt().

### 5.107.2.4 #define RSA\_MD2 2

Definition at line 21 of file rsa.h.

Referenced by rsa\_pkcs1\_sign(), rsa\_pkcs1\_verify(), x509\_hash(), and x509parse\_cert\_info().

### 5.107.2.5 #define RSA\_MD4 3

Definition at line 22 of file rsa.h.

Referenced by rsa\_pkcs1\_sign(), rsa\_pkcs1\_verify(), x509\_hash(), and x509parse\_cert\_info().

### 5.107.2.6 #define RSA\_MD5 4

Definition at line 23 of file rsa.h.

Referenced by rsa\_pkcs1\_sign(), rsa\_pkcs1\_verify(), x509\_hash(), and x509parse\_cert\_info().

### 5.107.2.7 #define RSA\_PKCS\_V15 0

Definition at line 30 of file rsa.h.

Referenced by generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), main(), rsa\_decryption(), rsa\_encryption(), rsa\_pkcs1\_decrypt(), rsa\_pkcs1\_encrypt(), rsa\_pkcs1\_sign(), and rsa\_pkcs1\_verify().

### 5.107.2.8 #define RSA\_PKCS\_V21 1

Definition at line 31 of file rsa.h.

### 5.107.2.9 #define RSA\_PRIVATE 1

Definition at line 28 of file rsa.h.

Referenced by main(), RSA\_private\_decrypt(), RSA\_private\_encrypt(), rsa\_self\_test(), ssl\_parse\_client\_key\_exchange(), ssl\_write\_certificate\_verify(), and ssl\_write\_server\_key\_exchange().

**5.107.2.10 #define RSA\_PUBLIC 0**

Definition at line 27 of file rsa.h.

Referenced by main(), rsa\_pkcs1\_decrypt(), rsa\_pkcs1\_encrypt(), rsa\_pkcs1\_sign(), rsa\_pkcs1\_verify(), RSA\_public\_decrypt(), RSA\_public\_encrypt(), rsa\_self\_test(), ssl\_parse\_certificate\_verify(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_client\_key\_exchange(), and x509parse\_verify().

**5.107.2.11 #define RSA\_RAW 0**

Definition at line 20 of file rsa.h.

Referenced by rsa\_pkcs1\_sign(), rsa\_pkcs1\_verify(), ssl\_parse\_certificate\_verify(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_certificate\_verify(), and ssl\_write\_server\_key\_exchange().

**5.107.2.12 #define RSA\_SHA1 5**

Definition at line 24 of file rsa.h.

Referenced by main(), rsa\_pkcs1\_sign(), rsa\_pkcs1\_verify(), rsa\_self\_test(), x509\_hash(), and x509parse\_cert\_info().

**5.107.2.13 #define RSA\_SHA256 6**

Definition at line 25 of file rsa.h.

**5.107.2.14 #define RSA\_SIGN 1**

Definition at line 33 of file rsa.h.

Referenced by rsa\_pkcs1\_sign(), and rsa\_pkcs1\_verify().

**5.107.2.15 #define XYSSL\_ERR\_RSA\_BAD\_INPUT\_DATA -0x0400**

Definition at line 9 of file rsa.h.

Referenced by rsa\_gen\_key(), rsa\_pkcs1\_decrypt(), rsa\_pkcs1\_encrypt(), rsa\_pkcs1\_sign(), rsa\_pkcs1\_verify(), rsa\_private(), and rsa\_public().

**5.107.2.16 #define XYSSL\_ERR\_RSA\_INVALID\_PADDING -0x0410**

Definition at line 10 of file rsa.h.

Referenced by rsa\_pkcs1\_decrypt(), rsa\_pkcs1\_encrypt(), rsa\_pkcs1\_sign(), and rsa\_pkcs1\_verify().

**5.107.2.17 #define XYSSL\_ERR\_RSA\_KEY\_CHECK\_FAILED -0x0430**

Definition at line 12 of file rsa.h.

Referenced by rsa\_check\_privkey(), and rsa\_check\_pubkey().

**5.107.2.18 #define XYSSL\_ERR\_RSA\_KEY\_GEN\_FAILED -0x0420**

Definition at line 11 of file rsa.h.

Referenced by rsa\_gen\_key().

**5.107.2.19 #define XYSSL\_ERR\_RSA\_PRIVATE\_FAILED -0x0450**

Definition at line 14 of file rsa.h.

Referenced by rsa\_private().

**5.107.2.20 #define XYSSL\_ERR\_RSA\_PUBLIC\_FAILED -0x0440**

Definition at line 13 of file rsa.h.

Referenced by rsa\_public().

**5.107.2.21 #define XYSSL\_ERR\_RSA\_VERIFY\_FAILED -0x0460**

Definition at line 15 of file rsa.h.

Referenced by rsa\_pkcs1\_verify().

**5.107.3 Function Documentation****5.107.3.1 int rsa\_check\_privkey (rsa\_context \* ctx)**

Check a private RSA key.

**Parameters**

*ctx* RSA context to be checked

**Returns**

0 if successful, or an XYSSL\_ERR\_RSA\_XXX error code

Definition at line 152 of file rsa.c.

References rsa\_context::D, rsa\_context::E, MPI\_CHK, mpi\_cmp\_int(), mpi\_cmp\_mpi(), mpi\_free(), mpi\_gcd(), mpi\_init(), mpi\_mod\_mpi(), mpi\_mul\_mpi(), mpi\_sub\_int(), rsa\_context::N, rsa\_context::P, rsa\_context::Q, rsa\_check\_pubkey(), and XYSSL\_ERR\_RSA\_KEY\_CHECK\_FAILED.

Referenced by rsa\_decryption(), rsa\_self\_test(), and x509parse\_key().

**5.107.3.2 int rsa\_check\_pubkey (rsa\_context \* ctx)**

Check a public RSA key.

**Parameters**

*ctx* RSA context to be checked

**Returns**

0 if successful, or an XYSSL\_ERR\_RSA\_XXX error code

Definition at line 132 of file rsa.c.

References `rsa_context::E`, `mpi_msb()`, `rsa_context::N`, `mpi::p`, and `XYSSL_ERR_RSA_KEY_CHECK_FAILED`.

Referenced by `rsa_check_privkey()`, `rsa_encryption()`, `rsa_self_test()`, and `x509parse_crt()`.

**5.107.3.3 void rsa\_free (rsa\_context \* ctx)**

Free the components of an RSA key.

Definition at line 558 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `mpi_free()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_context::RN`, `rsa_context::RP`, and `rsa_context::RQ`.

Referenced by `main()`, `rsa_gen_key()`, `rsa_self_test()`, `ssl_test()`, `x509_free()`, `x509_self_test()`, and `x509parse_key()`.

**5.107.3.4 int rsa\_gen\_key (rsa\_context \* ctx, int nbits, int exponent)**

Generate an RSA keypair.

**Parameters**

*ctx* RSA context that will hold the key

*nbits* size of the public key in bits

*exponent* public exponent (e.g., 65537)

**Note**

[rsa\\_init\(\)](#) must be called beforehand to setup the RSA context (especially `f_rng` and `p_rng`).

**Returns**

0 if successful, or an XYSSL\_ERR\_RSA\_XXX error code

Definition at line 60 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `rsa_context::f_rng`, `rsa_context::len`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_free()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_init()`, `mpi_inv_mod()`, `mpi_lset()`, `mpi_mod_mpi()`, `mpi_msb()`, `mpi_mul_mpi()`, `mpi_sub_int()`, `mpi_swap()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::p_rng`, `rsa_context::Q`, `rsa_context::QP`, `rsa_free()`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_KEY_GEN_FAILED`.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, and `main()`.

**5.107.3.5 void rsa\_init (rsa\_context \* ctx, int padding, int hash\_id, int(\*) (void \*) f\_rng, void \* p\_rng)**

Initialize an RSA context.

### Parameters

*ctx* RSA context to be initialized  
*padding* RSA\_PKCS\_V15 or RSA\_PKCS\_V21  
*hash\_id* RSA\_PKCS\_V21 hash identifier  
*f\_rng* RNG function  
*p\_rng* RNG parameter

### Note

The *hash\_id* parameter is actually ignored when using RSA\_PKCS\_V15 padding.  
Currently (xyssl-0.8), RSA\_PKCS\_V21 padding is not supported.

Definition at line 40 of file `rsa.c`.

References `rsa_context::f_rng`, `rsa_context::hash_id`, `rsa_context::p_rng`, and `rsa_context::padding`.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, `main()`, `rsa_decryption()`, and `rsa_encryption()`.

#### 5.107.3.6 `int rsa_pkcs1_decrypt(rsa_context * ctx, int mode, int * olen, unsigned char * input, unsigned char * output)`

Do an RSA operation, then remove the message padding.

### Parameters

*ctx* RSA context  
*mode* RSA\_PUBLIC or RSA\_PRIVATE  
*input* buffer holding the encrypted data  
*output* buffer that will hold the plaintext  
*olen* will contain the plaintext length

### Returns

0 if successful, or an XYSSL\_ERR\_RSA\_XXX error code

### Note

The output buffer must be as large as the size of `ctx->N` (eg. 128 bytes if RSA-1024 is used).

Definition at line 326 of file `rsa.c`.

References `buf`, `int`, `rsa_context::len`, `rsa_context::padding`, `RSA_CRYPT`, `RSA_PKCS_V15`, `rsa_private()`, `rsa_public()`, `RSA_PUBLIC`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_INVALID_PADDING`.

Referenced by `rsa_decryption()`, `RSA_private_decrypt()`, `RSA_public_decrypt()`, `rsa_self_test()`, and `ssl_parse_client_key_exchange()`.

#### 5.107.3.7 `int rsa_pkcs1_encrypt(rsa_context * ctx, int mode, int ilen, unsigned char * input, unsigned char * output)`

Add the message padding, then do an RSA operation.

**Parameters**

*ctx* RSA context  
*mode* RSA\_PUBLIC or RSA\_PRIVATE  
*ilen* contains the the plaintext length  
*input* buffer holding the data to be encrypted  
*output* buffer that will hold the ciphertext

**Returns**

0 if successful, or an XYSSL\_ERR\_RSA\_XXX error code

**Note**

The output buffer must be as large as the size of ctx->N (eg. 128 bytes if RSA-1024 is used).

Definition at line 280 of file rsa.c.

References `rsa_context::len`, `rsa_context::padding`, `RSA_CRYPT`, `RSA_PKCS_V15`, `rsa_private()`, `rsa_public()`, `RSA_PUBLIC`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_INVALID_PADDING`.

Referenced by `rsa_encryption()`, `RSA_private_encrypt()`, `RSA_public_encrypt()`, `rsa_self_test()`, and `ssl_write_client_key_exchange()`.

### 5.107.3.8 `int rsa_pkcs1_sign(rsa_context * ctx, int mode, int hash_id, int hashlen, unsigned char * hash, unsigned char * sig)`

Do a private RSA to sign a message digest.

**Parameters**

*ctx* RSA context  
*mode* RSA\_PUBLIC or RSA\_PRIVATE  
*hash\_id* RSA\_RAW, RSA\_MD{2,4,5} or RSA\_SHA{1,256}  
*hashlen* message digest length (for RSA\_RAW only)  
*hash* buffer holding the message digest  
*sig* buffer that will hold the ciphertext

**Returns**

0 if the signing operation was successful, or an XYSSL\_ERR\_RSA\_XXX error code

**Note**

The "sig" buffer must be as large as the size of ctx->N (eg. 128 bytes if RSA-1024 is used).

Definition at line 379 of file rsa.c.

References `ASN1_HASH_MD5`, `ASN1_HASH_SHA1`, `rsa_context::len`, `rsa_context::padding`, `RSA_MD2`, `RSA_MD4`, `RSA_MD5`, `RSA_PKCS_V15`, `rsa_private()`, `rsa_public()`, `RSA_PUBLIC`, `RSA_RAW`, `RSA_SHA1`, `RSA_SIGN`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_INVALID_PADDING`.

Referenced by `main()`, `rsa_self_test()`, `ssl_write_certificate_verify()`, and `ssl_write_server_key_exchange()`.



### 5.107.3.9 `int rsa_pkcs1_verify(rsa_context * ctx, int mode, int hash_id, int hashlen, unsigned char * hash, unsigned char * sig)`

Do a public RSA and check the message digest.

#### Parameters

*ctx* points to an RSA public key  
*mode* RSA\_PUBLIC or RSA\_PRIVATE  
*hash\_id* RSA\_RAW, RSA\_MD{2,4,5} or RSA\_SHA{1,256}  
*hashlen* message digest length (for RSA\_RAW only)  
*hash* buffer holding the message digest  
*sig* buffer holding the ciphertext

#### Returns

0 if the verify operation was successful, or an XYSSL\_ERR\_RSA\_XXX error code

#### Note

The "sig" buffer must be as large as the size of ctx->N (eg. 128 bytes if RSA-1024 is used).

Definition at line 468 of file rsa.c.

References ASN1\_HASH\_MDX, ASN1\_HASH\_SHA1, buf, int, rsa\_context::len, rsa\_context::padding, RSA\_MD2, RSA\_MD4, RSA\_MD5, RSA\_PKCS\_V15, rsa\_private(), rsa\_public(), RSA\_PUBLIC, RSA\_RAW, RSA\_SHA1, RSA\_SIGN, XYSSL\_ERR\_RSA\_BAD\_INPUT\_DATA, XYSSL\_ERR\_RSA\_INVALID\_PADDING, and XYSSL\_ERR\_RSA\_VERIFY\_FAILED.

Referenced by main(), rsa\_self\_test(), ssl\_parse\_certificate\_verify(), ssl\_parse\_server\_key\_exchange(), and x509parse\_verify().

### 5.107.3.10 `int rsa_private(rsa_context * ctx, unsigned char * input, unsigned char * output)`

Do an RSA private key operation.

#### Parameters

*ctx* RSA context  
*input* input buffer  
*output* output buffer

#### Returns

0 if successful, or an XYSSL\_ERR\_RSA\_XXX error code

#### Note

The input and output buffers must be large enough (eg. 128 bytes if RSA-1024 is used).

Definition at line 221 of file rsa.c.

References rsa\_context::D, rsa\_context::DP, rsa\_context::DQ, rsa\_context::len, mpi\_add\_mpi(), MPI\_CHK, mpi\_cmp\_mpi(), mpi\_exp\_mod(), mpi\_free(), mpi\_init(), mpi\_mod\_mpi(), mpi\_mul\_mpi(), mpi\_read\_binary(), mpi\_sub\_mpi(), mpi\_write\_binary(), rsa\_context::N, rsa\_context::P, rsa\_context::Q, rsa\_context::QP, rsa\_context::RN, rsa\_context::RP, rsa\_context::RQ, XYSSL\_ERR\_RSA\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_RSA\_PRIVATE\_FAILED.

Referenced by `main()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, and `rsa_pkcs1_verify()`.

### 5.107.3.11 `int rsa_public (rsa_context * ctx, unsigned char * input, unsigned char * output)`

Do an RSA public key operation.

#### Parameters

*ctx* RSA context

*input* input buffer

*output* output buffer

#### Returns

0 if successful, or an `XYSSL_ERR_RSA_XXX` error code

#### Note

This function does NOT take care of message padding. Also, be sure to set `input[0] = 0`.  
The input and output buffers must be large enough (eg. 128 bytes if RSA-1024 is used).

Definition at line 187 of file `rsa.c`.

References `rsa_context::E`, `rsa_context::len`, `MPI_CHK`, `mpi_cmp_mpi()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_init()`, `mpi_read_binary()`, `mpi_write_binary()`, `rsa_context::N`, `rsa_context::RN`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_PUBLIC_FAILED`.

Referenced by `main()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, and `rsa_pkcs1_verify()`.

### 5.107.3.12 `int rsa_self_test (int verbose)`

Checksum routine.

#### Returns

0 if successful, or 1 if the test failed

Definition at line 627 of file `rsa.c`.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `KEY_LEN`, `rsa_context::len`, `mpi_read_string()`, `rsa_context::N`, `rsa_context::P`, `PT_LEN`, `rsa_context::Q`, `rsa_context::QP`, `rsa_check_privkey()`, `rsa_check_pubkey()`, `RSA_D`, `RSA_DP`, `RSA_DQ`, `RSA_E`, `rsa_free()`, `RSA_N`, `RSA_P`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, `rsa_pkcs1_verify()`, `RSA_PRIVATE`, `RSA_PT`, `RSA_PUBLIC`, `RSA_Q`, `RSA_QP`, `RSA_SHA1`, and `sha1()`.

Referenced by `main()`.

## 5.108 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha1.h File Reference

### Data Structures

- struct [sha1\\_context](#)  
*SHA-1 context structure.*

### Functions

- void [sha1\\_starts](#) ([sha1\\_context](#) \*ctx)  
*SHA-1 context setup.*
- void [sha1\\_update](#) ([sha1\\_context](#) \*ctx, unsigned char \*input, int ilen)  
*SHA-1 process buffer.*
- void [sha1\\_finish](#) ([sha1\\_context](#) \*ctx, unsigned char output[20])  
*SHA-1 final digest.*
- void [sha1](#) (unsigned char \*input, int ilen, unsigned char output[20])  
*Output = SHA-1( input buffer ).*
- int [sha1\\_file](#) (char \*path, unsigned char output[20])  
*Output = SHA-1( file contents ).*
- void [sha1\\_hmac\\_starts](#) ([sha1\\_context](#) \*ctx, unsigned char \*key, int keylen)  
*SHA-1 HMAC context setup.*
- void [sha1\\_hmac\\_update](#) ([sha1\\_context](#) \*ctx, unsigned char \*input, int ilen)  
*SHA-1 HMAC process buffer.*
- void [sha1\\_hmac\\_finish](#) ([sha1\\_context](#) \*ctx, unsigned char output[20])  
*SHA-1 HMAC final digest.*
- void [sha1\\_hmac](#) (unsigned char \*key, int keylen, unsigned char \*input, int ilen, unsigned char output[20])  
*Output = HMAC-SHA-1( hmac key, input buffer ).*
- int [sha1\\_self\\_test](#) (int verbose)  
*Checkup routine.*

### 5.108.1 Detailed Description

Definition in file [sha1.h](#).

## 5.108.2 Function Documentation

### 5.108.2.1 void sha1 (unsigned char \* *input*, int *ilen*, unsigned char *output*[20])

Output = SHA-1( input buffer ).

#### Parameters

*input* buffer holding the data  
*ilen* length of the input data  
*output* SHA-1 checksum result

Definition at line 313 of file sha1.c.

References sha1\_finish(), sha1\_starts(), and sha1\_update().

Referenced by main(), rsa\_self\_test(), sha1\_hmac\_starts(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_sha1(), ssl\_parse\_finished(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_finished(), ssl\_write\_server\_key\_exchange(), and x509\_hash().

### 5.108.2.2 int sha1\_file (char \* *path*, unsigned char *output*[20])

Output = SHA-1( file contents ).

#### Parameters

*path* input file name  
*output* SHA-1 checksum result

#### Returns

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 327 of file sha1.c.

References buf, f, sha1\_finish(), sha1\_starts(), and sha1\_update().

Referenced by main(), and sha1\_wrapper().

### 5.108.2.3 void sha1\_finish (sha1\_context \* *ctx*, unsigned char *output*[20])

SHA-1 final digest.

#### Parameters

*ctx* SHA-1 context  
*output* SHA-1 checksum result

Definition at line 284 of file sha1.c.

References PUT\_ULONG\_BE, sha1\_padding, sha1\_update(), sha1\_context::state, and sha1\_context::total.

Referenced by sha1(), sha1\_file(), sha1\_hmac\_finish(), sha1\_self\_test(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_sha1(), ssl\_parse\_server\_key\_exchange(), and ssl\_write\_server\_key\_exchange().

**5.108.2.4 void sha1\_hmac (unsigned char \* *key*, int *keylen*, unsigned char \* *input*, int *ilen*, unsigned char *output*[20])**

Output = HMAC-SHA-1( hmac key, input buffer ).

**Parameters**

*key* HMAC secret key  
*keylen* length of the HMAC key  
*input* buffer holding the data  
*ilen* length of the input data  
*output* HMAC-SHA-1 result

Definition at line 413 of file sha1.c.

References sha1\_hmac\_finish(), sha1\_hmac\_starts(), and sha1\_hmac\_update().

Referenced by ssl\_decrypt\_buf(), ssl\_encrypt\_buf(), and tls1\_prf().

**5.108.2.5 void sha1\_hmac\_finish (sha1\_context \* *ctx*, unsigned char *output*[20])**

SHA-1 HMAC final digest.

**Parameters**

*ctx* HMAC context  
*output* SHA-1 HMAC checksum result

Definition at line 397 of file sha1.c.

References sha1\_context::opad, sha1\_finish(), sha1\_starts(), and sha1\_update().

Referenced by sha1\_hmac(), and sha1\_self\_test().

**5.108.2.6 void sha1\_hmac\_starts (sha1\_context \* *ctx*, unsigned char \* *key*, int *keylen*)**

SHA-1 HMAC context setup.

**Parameters**

*ctx* HMAC context to be initialized  
*key* HMAC secret key  
*keylen* length of the HMAC key

Definition at line 359 of file sha1.c.

References sha1\_context::ipad, sha1\_context::opad, sha1(), sha1\_starts(), and sha1\_update().

Referenced by sha1\_hmac(), and sha1\_self\_test().

**5.108.2.7 void sha1\_hmac\_update (sha1\_context \* *ctx*, unsigned char \* *input*, int *ilen*)**

SHA-1 HMAC process buffer.

**Parameters**

*ctx* HMAC context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 389 of file sha1.c.

References sha1\_update().

Referenced by sha1\_hmac(), and sha1\_self\_test().

**5.108.2.8 int sha1\_self\_test (int *verbose*)**

Checksum routine.

**Returns**

0 if successful, or 1 if the test failed

Definition at line 521 of file sha1.c.

References buf, sha1\_finish(), sha1\_hmac\_finish(), sha1\_hmac\_starts(), sha1\_hmac\_test\_buf, sha1\_hmac\_test\_buflen, sha1\_hmac\_test\_key, sha1\_hmac\_test\_keylen, sha1\_hmac\_test\_sum, sha1\_hmac\_update(), sha1\_starts(), sha1\_test\_buf, sha1\_test\_buflen, sha1\_test\_sum, and sha1\_update().

Referenced by main().

**5.108.2.9 void sha1\_starts (sha1\_context \* *ctx*)**

SHA-1 context setup.

**Parameters**

*ctx* context to be initialized

Definition at line 61 of file sha1.c.

References sha1\_context::state, and sha1\_context::total.

Referenced by sha1(), sha1\_file(), sha1\_hmac\_finish(), sha1\_hmac\_starts(), sha1\_self\_test(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_init(), ssl\_mac\_sha1(), ssl\_parse\_server\_key\_exchange(), and ssl\_write\_server\_key\_exchange().

**5.108.2.10 void sha1\_update (sha1\_context \* *ctx*, unsigned char \* *input*, int *ilen*)**

SHA-1 process buffer.

**Parameters**

*ctx* SHA-1 context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 232 of file sha1.c.

References sha1\_context::buffer, sha1\_process(), and sha1\_context::total.

Referenced by sha1(), sha1\_file(), sha1\_finish(), sha1\_hmac\_finish(), sha1\_hmac\_starts(), sha1\_hmac\_update(), sha1\_self\_test(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_sha1(), ssl\_parse\_client\_hello(), ssl\_parse\_server\_key\_exchange(), ssl\_read\_record(), ssl\_write\_record(), and ssl\_write\_server\_key\_exchange().

## 5.109 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha2.h File Reference

### Data Structures

- struct [sha2\\_context](#)  
*SHA-256 context structure.*

### Functions

- void [sha2\\_starts](#) ([sha2\\_context](#) \*ctx, [int](#) is224)  
*SHA-256 context setup.*
- void [sha2\\_update](#) ([sha2\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*SHA-256 process buffer.*
- void [sha2\\_finish](#) ([sha2\\_context](#) \*ctx, unsigned char output[32])  
*SHA-256 final digest.*
- void [sha2](#) (unsigned char \*input, [int](#) ilen, unsigned char output[32], [int](#) is224)  
*Output = SHA-256( input buffer ).*
- [int](#) [sha2\\_file](#) (char \*path, unsigned char output[32], [int](#) is224)  
*Output = SHA-256( file contents ).*
- void [sha2\\_hmac\\_starts](#) ([sha2\\_context](#) \*ctx, unsigned char \*key, [int](#) keylen, [int](#) is224)  
*SHA-256 HMAC context setup.*
- void [sha2\\_hmac\\_update](#) ([sha2\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*SHA-256 HMAC process buffer.*
- void [sha2\\_hmac\\_finish](#) ([sha2\\_context](#) \*ctx, unsigned char output[32])  
*SHA-256 HMAC final digest.*
- void [sha2\\_hmac](#) (unsigned char \*key, [int](#) keylen, unsigned char \*input, [int](#) ilen, unsigned char output[32], [int](#) is224)  
*Output = HMAC-SHA-256( hmac key, input buffer ).*
- [int](#) [sha2\\_self\\_test](#) ([int](#) verbose)  
*Checkup routine.*

### 5.109.1 Detailed Description

Definition in file [sha2.h](#).



## 5.109.2 Function Documentation

### 5.109.2.1 void sha2 (unsigned char \* *input*, int *ilen*, unsigned char *output*[32], int *is224*)

Output = SHA-256( input buffer ).

#### Parameters

*input* buffer holding the data  
*ilen* length of the input data  
*output* SHA-224/256 checksum result  
*is224* 0 = use SHA256, 1 = use SHA224

Definition at line 314 of file sha2.c.

References sha2\_finish(), sha2\_starts(), and sha2\_update().

Referenced by main(), and sha2\_hmac\_starts().

### 5.109.2.2 int sha2\_file (char \* *path*, unsigned char *output*[32], int *is224*)

Output = SHA-256( file contents ).

#### Parameters

*path* input file name  
*output* SHA-224/256 checksum result  
*is224* 0 = use SHA256, 1 = use SHA224

#### Returns

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 329 of file sha2.c.

References buf, f, sha2\_finish(), sha2\_starts(), and sha2\_update().

Referenced by sha2\_wrapper().

### 5.109.2.3 void sha2\_finish (sha2\_context \* *ctx*, unsigned char *output*[32])

SHA-256 final digest.

#### Parameters

*ctx* SHA-256 context  
*output* SHA-224/256 checksum result

Definition at line 280 of file sha2.c.

References sha2\_context::is224, PUT\_ULONG\_BE, sha2\_padding, sha2\_update(), sha2\_context::state, and sha2\_context::total.

Referenced by aes\_en\_de(), main(), sha2(), sha2\_file(), sha2\_hmac\_finish(), and sha2\_self\_test().

**5.109.2.4** `void sha2_hmac (unsigned char * key, int keylen, unsigned char * input, int ilen, unsigned char output[32], int is224)`

Output = HMAC-SHA-256( hmac key, input buffer ).

#### Parameters

*key* HMAC secret key  
*keylen* length of the HMAC key  
*input* buffer holding the data  
*ilen* length of the input data  
*output* HMAC-SHA-224/256 result  
*is224* 0 = use SHA256, 1 = use SHA224

Definition at line 420 of file sha2.c.

References sha2\_hmac\_finish(), sha2\_hmac\_starts(), and sha2\_hmac\_update().

**5.109.2.5** `void sha2_hmac_finish (sha2_context * ctx, unsigned char output[32])`

SHA-256 HMAC final digest.

#### Parameters

*ctx* HMAC context  
*output* SHA-224/256 HMAC checksum result

Definition at line 400 of file sha2.c.

References sha2\_context::is224, sha2\_context::opad, sha2\_finish(), sha2\_starts(), and sha2\_update().

Referenced by aes\_en\_de(), main(), sha2\_hmac(), and sha2\_self\_test().

**5.109.2.6** `void sha2_hmac_starts (sha2_context * ctx, unsigned char * key, int keylen, int is224)`

SHA-256 HMAC context setup.

#### Parameters

*ctx* HMAC context to be initialized  
*key* HMAC secret key  
*keylen* length of the HMAC key  
*is224* 0 = use SHA256, 1 = use SHA224

Definition at line 361 of file sha2.c.

References sha2\_context::ipad, sha2\_context::opad, sha2(), sha2\_starts(), and sha2\_update().

Referenced by aes\_en\_de(), main(), sha2\_hmac(), and sha2\_self\_test().

**5.109.2.7 void sha2\_hmac\_update (sha2\_context \* *ctx*, unsigned char \* *input*, int *ilen*)**

SHA-256 HMAC process buffer.

**Parameters**

*ctx* HMAC context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 392 of file sha2.c.

References sha2\_update().

Referenced by aes\_en\_de(), main(), sha2\_hmac(), and sha2\_self\_test().

**5.109.2.8 int sha2\_self\_test (int *verbose*)**

Checksum routine.

**Returns**

0 if successful, or 1 if the test failed

Definition at line 599 of file sha2.c.

References buf, sha2\_finish(), sha2\_hmac\_finish(), sha2\_hmac\_starts(), sha2\_hmac\_test\_buf, sha2\_hmac\_test\_buflen, sha2\_hmac\_test\_key, sha2\_hmac\_test\_keylen, sha2\_hmac\_test\_sum, sha2\_hmac\_update(), sha2\_starts(), sha2\_test\_buf, sha2\_test\_buflen, sha2\_test\_sum, and sha2\_update().

Referenced by main().

**5.109.2.9 void sha2\_starts (sha2\_context \* *ctx*, int *is224*)**

SHA-256 context setup.

**Parameters**

*ctx* context to be initialized

*is224* 0 = use SHA256, 1 = use SHA224

Definition at line 61 of file sha2.c.

References sha2\_context::is224, sha2\_context::state, and sha2\_context::total.

Referenced by aes\_en\_de(), main(), sha2(), sha2\_file(), sha2\_hmac\_finish(), sha2\_hmac\_starts(), and sha2\_self\_test().

**5.109.2.10 void sha2\_update (sha2\_context \* *ctx*, unsigned char \* *input*, int *ilen*)**

SHA-256 process buffer.

**Parameters**

*ctx* SHA-256 context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 228 of file sha2.c.

References sha2\_context::buffer, sha2\_process(), and sha2\_context::total.

Referenced by aes\_en\_de(), main(), sha2(), sha2\_file(), sha2\_finish(), sha2\_hmac\_finish(), sha2\_hmac\_starts(), sha2\_hmac\_update(), and sha2\_self\_test().

## 5.110 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha4.h File Reference

### Data Structures

- struct [sha4\\_context](#)  
*SHA-512 context structure.*

### Defines

- #define [UL64](#)(x) x##ULL
- #define [int64](#) long long

### Functions

- void [sha4\\_starts](#) ([sha4\\_context](#) \*ctx, [int](#) is384)  
*SHA-512 context setup.*
- void [sha4\\_update](#) ([sha4\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*SHA-512 process buffer.*
- void [sha4\\_finish](#) ([sha4\\_context](#) \*ctx, unsigned char output[64])  
*SHA-512 final digest.*
- void [sha4](#) (unsigned char \*input, [int](#) ilen, unsigned char output[64], [int](#) is384)  
*Output = SHA-512( input buffer ).*
- [int](#) [sha4\\_file](#) (char \*path, unsigned char output[64], [int](#) is384)  
*Output = SHA-512( file contents ).*
- void [sha4\\_hmac\\_starts](#) ([sha4\\_context](#) \*ctx, unsigned char \*key, [int](#) keylen, [int](#) is384)  
*SHA-512 HMAC context setup.*
- void [sha4\\_hmac\\_update](#) ([sha4\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*SHA-512 HMAC process buffer.*
- void [sha4\\_hmac\\_finish](#) ([sha4\\_context](#) \*ctx, unsigned char output[64])  
*SHA-512 HMAC final digest.*
- void [sha4\\_hmac](#) (unsigned char \*key, [int](#) keylen, unsigned char \*input, [int](#) ilen, unsigned char output[64], [int](#) is384)  
*Output = HMAC-SHA-512( hmac key, input buffer ).*
- [int](#) [sha4\\_self\\_test](#) ([int](#) verbose)  
*Checkup routine.*

### 5.110.1 Detailed Description

Definition in file [sha4.h](#).

### 5.110.2 Define Documentation

#### 5.110.2.1 `#define int64 long long`

Definition at line 12 of file sha4.h.

Referenced by sha4\_finish(), sha4\_process(), and sha4\_update().

#### 5.110.2.2 `#define UL64(x) x##ULL`

Definition at line 11 of file sha4.h.

Referenced by sha4\_starts().

### 5.110.3 Function Documentation

#### 5.110.3.1 `void sha4 (unsigned char * input, int ilen, unsigned char output[64], int is384)`

Output = SHA-512( input buffer ).

##### Parameters

*input* buffer holding the data  
*ilen* length of the input data  
*output* SHA-384/512 checksum result  
*is384* 0 = use SHA512, 1 = use SHA384

Definition at line 312 of file sha4.c.

References sha4\_finish(), sha4\_starts(), and sha4\_update().

Referenced by sha4\_hmac\_starts().

#### 5.110.3.2 `int sha4_file (char * path, unsigned char output[64], int is384)`

Output = SHA-512( file contents ).

##### Parameters

*path* input file name  
*output* SHA-384/512 checksum result  
*is384* 0 = use SHA512, 1 = use SHA384

##### Returns

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 327 of file sha4.c.

References buf, f, sha4\_finish(), sha4\_starts(), and sha4\_update().

**5.110.3.3 void sha4\_finish (sha4\_context \* ctx, unsigned char output[64])**

SHA-512 final digest.

**Parameters**

*ctx* SHA-512 context

*output* SHA-384/512 checksum result

Definition at line 276 of file sha4.c.

References int, int64, sha4\_context::is384, PUT\_UINT64\_BE, sha4\_padding, sha4\_update(), sha4\_context::state, and sha4\_context::total.

Referenced by sha4(), sha4\_file(), sha4\_hmac\_finish(), and sha4\_self\_test().

**5.110.3.4 void sha4\_hmac (unsigned char \* key, int keylen, unsigned char \* input, int ilen, unsigned char output[64], int is384)**

Output = HMAC-SHA-512( hmac key, input buffer ).

**Parameters**

*key* HMAC secret key

*keylen* length of the HMAC key

*input* buffer holding the data

*ilen* length of the input data

*output* HMAC-SHA-384/512 result

*is384* 0 = use SHA512, 1 = use SHA384

Definition at line 419 of file sha4.c.

References sha4\_hmac\_finish(), sha4\_hmac\_starts(), and sha4\_hmac\_update().

**5.110.3.5 void sha4\_hmac\_finish (sha4\_context \* ctx, unsigned char output[64])**

SHA-512 HMAC final digest.

**Parameters**

*ctx* HMAC context

*output* SHA-384/512 HMAC checksum result

Definition at line 399 of file sha4.c.

References sha4\_context::is384, sha4\_context::opad, sha4\_finish(), sha4\_starts(), and sha4\_update().

Referenced by sha4\_hmac(), and sha4\_self\_test().

**5.110.3.6 void sha4\_hmac\_starts (sha4\_context \* ctx, unsigned char \* key, int keylen, int is384)**

SHA-512 HMAC context setup.

**Parameters**

*ctx* HMAC context to be initialized  
*is384* 0 = use SHA512, 1 = use SHA384  
*key* HMAC secret key  
*keylen* length of the HMAC key

Definition at line 359 of file sha4.c.

References sha4\_context::ipad, sha4\_context::opad, sha4(), sha4\_starts(), and sha4\_update().

Referenced by sha4\_hmac(), and sha4\_self\_test().

**5.110.3.7 void sha4\_hmac\_update (sha4\_context \* ctx, unsigned char \* input, int ilen)**

SHA-512 HMAC process buffer.

**Parameters**

*ctx* HMAC context  
*input* buffer holding the data  
*ilen* length of the input data

Definition at line 390 of file sha4.c.

References sha4\_update().

Referenced by sha4\_hmac(), and sha4\_self\_test().

**5.110.3.8 int sha4\_self\_test (int verbose)**

Checksum routine.

**Returns**

0 if successful, or 1 if the test failed

Definition at line 654 of file sha4.c.

References buf, sha4\_finish(), sha4\_hmac\_finish(), sha4\_hmac\_starts(), sha4\_hmac\_test\_buf, sha4\_hmac\_test\_buflen, sha4\_hmac\_test\_key, sha4\_hmac\_test\_keylen, sha4\_hmac\_test\_sum, sha4\_hmac\_update(), sha4\_starts(), sha4\_test\_buf, sha4\_test\_buflen, sha4\_test\_sum, and sha4\_update().

Referenced by main().

**5.110.3.9 void sha4\_starts (sha4\_context \* ctx, int is384)**

SHA-512 context setup.



**Parameters**

*ctx* context to be initialized

*is384* 0 = use SHA512, 1 = use SHA384

Definition at line 116 of file sha4.c.

References sha4\_context::is384, sha4\_context::state, sha4\_context::total, and UL64.

Referenced by sha4(), sha4\_file(), sha4\_hmac\_finish(), sha4\_hmac\_starts(), and sha4\_self\_test().

**5.110.3.10 void sha4\_update (sha4\_context \* ctx, unsigned char \* input, int ilen)**

SHA-512 process buffer.

**Parameters**

*ctx* SHA-512 context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 221 of file sha4.c.

References sha4\_context::buffer, int, int64, sha4\_process(), and sha4\_context::total.

Referenced by sha4(), sha4\_file(), sha4\_finish(), sha4\_hmac\_finish(), sha4\_hmac\_starts(), sha4\_hmac\_update(), and sha4\_self\_test().

## 5.111 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/ssl.h File Reference

```
#include <time.h>
#include "xyssl/net.h"
#include "xyssl/dhm.h"
#include "xyssl/rsa.h"
#include "xyssl/md5.h"
#include "xyssl/sha1.h"
#include "xyssl/x509.h"
```

### Data Structures

- struct [\\_ssl\\_session](#)
- struct [\\_ssl\\_context](#)

### Defines

- #define [XYSSL\\_ERR\\_SSL\\_FEATURE\\_UNAVAILABLE](#) -0x1000
- #define [XYSSL\\_ERR\\_SSL\\_BAD\\_INPUT\\_DATA](#) -0x1800
- #define [XYSSL\\_ERR\\_SSL\\_INVALID\\_MAC](#) -0x2000
- #define [XYSSL\\_ERR\\_SSL\\_INVALID\\_RECORD](#) -0x2800
- #define [XYSSL\\_ERR\\_SSL\\_INVALID\\_MODULUS\\_SIZE](#) -0x3000
- #define [XYSSL\\_ERR\\_SSL\\_UNKNOWN\\_CIPHER](#) -0x3800
- #define [XYSSL\\_ERR\\_SSL\\_NO\\_CIPHER\\_CHOSEN](#) -0x4000
- #define [XYSSL\\_ERR\\_SSL\\_NO\\_SESSION\\_FOUND](#) -0x4800
- #define [XYSSL\\_ERR\\_SSL\\_NO\\_CLIENT\\_CERTIFICATE](#) -0x5000
- #define [XYSSL\\_ERR\\_SSL\\_CERTIFICATE\\_TOO\\_LARGE](#) -0x5800
- #define [XYSSL\\_ERR\\_SSL\\_CERTIFICATE\\_REQUIRED](#) -0x6000
- #define [XYSSL\\_ERR\\_SSL\\_PRIVATE\\_KEY\\_REQUIRED](#) -0x6800
- #define [XYSSL\\_ERR\\_SSL\\_CA\\_CHAIN\\_REQUIRED](#) -0x7000
- #define [XYSSL\\_ERR\\_SSL\\_UNEXPECTED\\_MESSAGE](#) -0x7800
- #define [XYSSL\\_ERR\\_SSL\\_FATAL\\_ALERT\\_MESSAGE](#) -0x8000
- #define [XYSSL\\_ERR\\_SSL\\_PEER\\_VERIFY\\_FAILED](#) -0x8800
- #define [XYSSL\\_ERR\\_SSL\\_PEER\\_CLOSE\\_NOTIFY](#) -0x9000
- #define [XYSSL\\_ERR\\_SSL\\_BAD\\_HS\\_CLIENT\\_HELLO](#) -0x9800
- #define [XYSSL\\_ERR\\_SSL\\_BAD\\_HS\\_SERVER\\_HELLO](#) -0xA000
- #define [XYSSL\\_ERR\\_SSL\\_BAD\\_HS\\_CERTIFICATE](#) -0xA800
- #define [XYSSL\\_ERR\\_SSL\\_BAD\\_HS\\_CERTIFICATE\\_REQUEST](#) -0xB000
- #define [XYSSL\\_ERR\\_SSL\\_BAD\\_HS\\_SERVER\\_KEY\\_EXCHANGE](#) -0xB800
- #define [XYSSL\\_ERR\\_SSL\\_BAD\\_HS\\_SERVER\\_HELLO\\_DONE](#) -0xC000
- #define [XYSSL\\_ERR\\_SSL\\_BAD\\_HS\\_CLIENT\\_KEY\\_EXCHANGE](#) -0xC800
- #define [XYSSL\\_ERR\\_SSL\\_BAD\\_HS\\_CERTIFICATE\\_VERIFY](#) -0xD000
- #define [XYSSL\\_ERR\\_SSL\\_BAD\\_HS\\_CHANGE\\_CIPHER\\_SPEC](#) -0xD800
- #define [XYSSL\\_ERR\\_SSL\\_BAD\\_HS\\_FINISHED](#) -0xE000
- #define [SSL\\_MAJOR\\_VERSION\\_3](#) 3

- #define [SSL\\_MINOR\\_VERSION\\_0](#) 0
- #define [SSL\\_MINOR\\_VERSION\\_1](#) 1
- #define [SSL\\_MINOR\\_VERSION\\_2](#) 2
- #define [SSL\\_IS\\_CLIENT](#) 0
- #define [SSL\\_IS\\_SERVER](#) 1
- #define [SSL\\_COMPRESS\\_NULL](#) 0
- #define [SSL\\_VERIFY\\_NONE](#) 0
- #define [SSL\\_VERIFY\\_OPTIONAL](#) 1
- #define [SSL\\_VERIFY\\_REQUIRED](#) 2
- #define [SSL\\_MAX\\_CONTENT\\_LEN](#) 16384
- #define [SSL\\_BUFFER\\_LEN](#) (SSL\_MAX\_CONTENT\_LEN + 512)
- #define [SSL\\_RSA\\_RC4\\_128\\_MD5](#) 4
- #define [SSL\\_RSA\\_RC4\\_128\\_SHA](#) 5
- #define [SSL\\_RSA\\_DES\\_168\\_SHA](#) 10
- #define [SSL\\_EDH\\_RSA\\_DES\\_168\\_SHA](#) 22
- #define [SSL\\_RSA\\_AES\\_128\\_SHA](#) 47
- #define [SSL\\_RSA\\_AES\\_256\\_SHA](#) 53
- #define [SSL\\_EDH\\_RSA\\_AES\\_256\\_SHA](#) 57
- #define [SSL\\_MSG\\_CHANGE\\_CIPHER\\_SPEC](#) 20
- #define [SSL\\_MSG\\_ALERT](#) 21
- #define [SSL\\_MSG\\_HANDSHAKE](#) 22
- #define [SSL\\_MSG\\_APPLICATION\\_DATA](#) 23
- #define [SSL\\_ALERT\\_CLOSE\\_NOTIFY](#) 0
- #define [SSL\\_ALERT\\_WARNING](#) 1
- #define [SSL\\_ALERT\\_FATAL](#) 2
- #define [SSL\\_ALERT\\_NO\\_CERTIFICATE](#) 41
- #define [SSL\\_HS\\_HELLO\\_REQUEST](#) 0
- #define [SSL\\_HS\\_CLIENT\\_HELLO](#) 1
- #define [SSL\\_HS\\_SERVER\\_HELLO](#) 2
- #define [SSL\\_HS\\_CERTIFICATE](#) 11
- #define [SSL\\_HS\\_SERVER\\_KEY\\_EXCHANGE](#) 12
- #define [SSL\\_HS\\_CERTIFICATE\\_REQUEST](#) 13
- #define [SSL\\_HS\\_SERVER\\_HELLO\\_DONE](#) 14
- #define [SSL\\_HS\\_CERTIFICATE\\_VERIFY](#) 15
- #define [SSL\\_HS\\_CLIENT\\_KEY\\_EXCHANGE](#) 16
- #define [SSL\\_HS\\_FINISHED](#) 20
- #define [TLS\\_EXT\\_SERVERNAME](#) 0
- #define [TLS\\_EXT\\_SERVERNAME\\_HOSTNAME](#) 0

## Typedefs

- typedef struct [\\_ssl\\_session](#) [ssl\\_session](#)
- typedef struct [\\_ssl\\_context](#) [ssl\\_context](#)

## Enumerations

- enum `ssl_states` {  
`SSL_HELLO_REQUEST`, `SSL_CLIENT_HELLO`, `SSL_SERVER_HELLO`, `SSL_SERVER_CERTIFICATE`,  
`SSL_SERVER_KEY_EXCHANGE`, `SSL_CERTIFICATE_REQUEST`, `SSL_SERVER_HELLO_DONE`, `SSL_CLIENT_CERTIFICATE`,  
`SSL_CLIENT_KEY_EXCHANGE`, `SSL_CERTIFICATE_VERIFY`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_CLIENT_FINISHED`,  
`SSL_SERVER_CHANGE_CIPHER_SPEC`, `SSL_SERVER_FINISHED`, `SSL_FLUSH_BUFFERS`, `SSL_HANDSHAKE_OVER` }

## Functions

- int `ssl_init` (`ssl_context` \*ssl)  
Initialize an SSL context.
- void `ssl_set_endpoint` (`ssl_context` \*ssl, int endpoint)  
Set the current endpoint type.
- void `ssl_set_authmode` (`ssl_context` \*ssl, int authmode)  
Set the certificate verification mode.
- void `ssl_set_rng` (`ssl_context` \*ssl, int(\*f\_rng)(void \*), void \*p\_rng)  
Set the random number generator callback.
- void `ssl_set_dbg` (`ssl_context` \*ssl, void(\*f\_dbg)(void \*, int, char \*), void \*p\_dbg)  
Set the debug callback.
- void `ssl_set_bio` (`ssl_context` \*ssl, int(\*f\_rcv)(void \*, unsigned char \*, int), void \*p\_rcv, int(\*f\_send)(void \*, unsigned char \*, int), void \*p\_send)  
Set the underlying BIO read and write callbacks.
- void `ssl_set_scb` (`ssl_context` \*ssl, int(\*s\_get)(`ssl_context` \*), int(\*s\_set)(`ssl_context` \*))  
Set the session callbacks (server-side only).
- void `ssl_set_session` (`ssl_context` \*ssl, int resume, int timeout, `ssl_session` \*session)  
Set the session resuming flag, timeout and data.
- void `ssl_set_ciphers` (`ssl_context` \*ssl, int \*ciphers)  
Set the list of allowed ciphersuites.
- void `ssl_set_ca_chain` (`ssl_context` \*ssl, `x509_cert` \*ca\_chain, char \*peer\_cn)  
Set the data required to verify peer certificate.
- void `ssl_set_own_cert` (`ssl_context` \*ssl, `x509_cert` \*own\_cert, `rsa_context` \*rsa\_key)  
Set own certificate and private key.
- int `ssl_set_dh_param` (`ssl_context` \*ssl, char \*dhm\_P, char \*dhm\_G)

*Set the Diffie-Hellman public  $P$  and  $G$  values, read as hexadecimal strings (server-side only).*

- `int ssl_set_hostname (ssl_context *ssl, char *hostname)`  
*Set hostname for ServerName TLS Extension.*
- `int ssl_get_bytes_avail (ssl_context *ssl)`  
*Return the number of data bytes available to read.*
- `int ssl_get_verify_result (ssl_context *ssl)`  
*Return the result of the certificate verification.*
- `char * ssl_get_cipher (ssl_context *ssl)`  
*Return the name of the current cipher.*
- `int ssl_handshake (ssl_context *ssl)`  
*Perform the SSL handshake.*
- `int ssl_read (ssl_context *ssl, unsigned char *buf, int len)`  
*Read at most 'len' application data bytes.*
- `int ssl_write (ssl_context *ssl, unsigned char *buf, int len)`  
*Write exactly 'len' application data bytes.*
- `int ssl_close_notify (ssl_context *ssl)`  
*Notify the peer that the connection is being closed.*
- `void ssl_free (ssl_context *ssl)`  
*Free an SSL context.*
- `int ssl_handshake_client (ssl_context *ssl)`
- `int ssl_handshake_server (ssl_context *ssl)`
- `int ssl_derive_keys (ssl_context *ssl)`
- `void ssl_calc_verify (ssl_context *ssl, unsigned char hash[36])`
- `int ssl_read_record (ssl_context *ssl)`
- `int ssl_fetch_input (ssl_context *ssl, int nb_want)`
- `int ssl_write_record (ssl_context *ssl)`
- `int ssl_flush_output (ssl_context *ssl)`
- `int ssl_parse_certificate (ssl_context *ssl)`
- `int ssl_write_certificate (ssl_context *ssl)`
- `int ssl_parse_change_cipher_spec (ssl_context *ssl)`
- `int ssl_write_change_cipher_spec (ssl_context *ssl)`
- `int ssl_parse_finished (ssl_context *ssl)`
- `int ssl_write_finished (ssl_context *ssl)`

## Variables

- `int ssl_default_ciphers []`

### 5.111.1 Detailed Description

Definition in file [ssl.h](#).

### 5.111.2 Define Documentation

#### 5.111.2.1 `#define SSL_ALERT_CLOSE_NOTIFY 0`

Definition at line 87 of file [ssl.h](#).

Referenced by [ssl\\_close\\_notify\(\)](#), and [ssl\\_read\\_record\(\)](#).

#### 5.111.2.2 `#define SSL_ALERT_FATAL 2`

Definition at line 89 of file [ssl.h](#).

Referenced by [ssl\\_read\\_record\(\)](#).

#### 5.111.2.3 `#define SSL_ALERT_NO_CERTIFICATE 41`

Definition at line 90 of file [ssl.h](#).

Referenced by [ssl\\_parse\\_certificate\(\)](#), and [ssl\\_write\\_certificate\(\)](#).

#### 5.111.2.4 `#define SSL_ALERT_WARNING 1`

Definition at line 88 of file [ssl.h](#).

Referenced by [ssl\\_close\\_notify\(\)](#), [ssl\\_parse\\_certificate\(\)](#), [ssl\\_read\\_record\(\)](#), and [ssl\\_write\\_certificate\(\)](#).

#### 5.111.2.5 `#define SSL_BUFFER_LEN (SSL_MAX_CONTENT_LEN + 512)`

Definition at line 66 of file [ssl.h](#).

Referenced by [ssl\\_free\(\)](#), and [ssl\\_init\(\)](#).

#### 5.111.2.6 `#define SSL_COMPRESS_NULL 0`

Definition at line 54 of file [ssl.h](#).

Referenced by [ssl\\_parse\\_server\\_hello\(\)](#), [ssl\\_write\\_client\\_hello\(\)](#), and [ssl\\_write\\_server\\_hello\(\)](#).

#### 5.111.2.7 `#define SSL_EDH_RSA_AES_256_SHA 57`

Definition at line 77 of file [ssl.h](#).

Referenced by [main\(\)](#), [ssl\\_derive\\_keys\(\)](#), [ssl\\_get\\_cipher\(\)](#), [ssl\\_parse\\_client\\_key\\_exchange\(\)](#), [ssl\\_parse\\_server\\_key\\_exchange\(\)](#), [ssl\\_write\\_client\\_key\\_exchange\(\)](#), and [ssl\\_write\\_server\\_key\\_exchange\(\)](#).

**5.111.2.8 #define SSL\_EDH\_RSA\_DES\_168\_SHA 22**

Definition at line 74 of file ssl.h.

Referenced by main(), ssl\_derive\_keys(), ssl\_get\_cipher(), ssl\_parse\_client\_key\_exchange(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_client\_key\_exchange(), and ssl\_write\_server\_key\_exchange().

**5.111.2.9 #define SSL\_HS\_CERTIFICATE 11**

Definition at line 95 of file ssl.h.

Referenced by ssl\_parse\_certificate(), and ssl\_write\_certificate().

**5.111.2.10 #define SSL\_HS\_CERTIFICATE\_REQUEST 13**

Definition at line 97 of file ssl.h.

Referenced by ssl\_parse\_certificate\_request(), and ssl\_write\_certificate\_request().

**5.111.2.11 #define SSL\_HS\_CERTIFICATE\_VERIFY 15**

Definition at line 99 of file ssl.h.

Referenced by ssl\_parse\_certificate\_verify(), and ssl\_write\_certificate\_verify().

**5.111.2.12 #define SSL\_HS\_CLIENT\_HELLO 1**

Definition at line 93 of file ssl.h.

Referenced by ssl\_parse\_client\_hello(), and ssl\_write\_client\_hello().

**5.111.2.13 #define SSL\_HS\_CLIENT\_KEY\_EXCHANGE 16**

Definition at line 100 of file ssl.h.

Referenced by ssl\_parse\_client\_key\_exchange(), and ssl\_write\_client\_key\_exchange().

**5.111.2.14 #define SSL\_HS\_FINISHED 20**

Definition at line 101 of file ssl.h.

Referenced by ssl\_parse\_finished(), and ssl\_write\_finished().

**5.111.2.15 #define SSL\_HS\_HELLO\_REQUEST 0**

Definition at line 92 of file ssl.h.

**5.111.2.16 #define SSL\_HS\_SERVER\_HELLO 2**

Definition at line 94 of file ssl.h.

Referenced by ssl\_parse\_server\_hello(), and ssl\_write\_server\_hello().

**5.111.2.17 #define SSL\_HS\_SERVER\_HELLO\_DONE 14**

Definition at line 98 of file ssl.h.

Referenced by ssl\_parse\_server\_hello\_done(), and ssl\_write\_server\_hello\_done().

**5.111.2.18 #define SSL\_HS\_SERVER\_KEY\_EXCHANGE 12**

Definition at line 96 of file ssl.h.

Referenced by ssl\_parse\_server\_key\_exchange(), and ssl\_write\_server\_key\_exchange().

**5.111.2.19 #define SSL\_IS\_CLIENT 0**

Definition at line 52 of file ssl.h.

Referenced by main(), ssl\_calc\_finished(), ssl\_derive\_keys(), ssl\_handshake(), ssl\_parse\_finished(), ssl\_test(), ssl\_write\_certificate(), and ssl\_write\_finished().

**5.111.2.20 #define SSL\_IS\_SERVER 1**

Definition at line 53 of file ssl.h.

Referenced by main(), ssl\_handshake(), ssl\_parse\_certificate(), ssl\_parse\_finished(), and ssl\_test().

**5.111.2.21 #define SSL\_MAJOR\_VERSION\_3 3**

Definition at line 47 of file ssl.h.

Referenced by ssl\_parse\_client\_hello(), ssl\_parse\_server\_hello(), and ssl\_write\_client\_hello().

**5.111.2.22 #define SSL\_MAX\_CONTENT\_LEN 16384**

Definition at line 60 of file ssl.h.

Referenced by ssl\_read\_record(), ssl\_write(), and ssl\_write\_certificate().

**5.111.2.23 #define SSL\_MINOR\_VERSION\_0 0**

SSL v3.0

Definition at line 48 of file ssl.h.

Referenced by ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_decrypt\_buf(), ssl\_derive\_keys(), ssl\_encrypt\_buf(), ssl\_parse\_certificate(), ssl\_parse\_client\_key\_exchange(), ssl\_parse\_finished(), ssl\_parse\_server\_hello(), ssl\_read\_record(), ssl\_write\_certificate(), ssl\_write\_client\_hello(), ssl\_write\_client\_key\_exchange(), and ssl\_write\_finished().

**5.111.2.24 #define SSL\_MINOR\_VERSION\_1 1**

TLS v1.0

Definition at line 49 of file ssl.h.



Referenced by `ssl_parse_client_hello()`, `ssl_parse_server_hello()`, `ssl_read_record()`, and `ssl_write_client_hello()`.

#### 5.111.2.25 `#define SSL_MINOR_VERSION_2 2`

TLS v1.1

Definition at line 50 of file `ssl.h`.

#### 5.111.2.26 `#define SSL_MSG_ALERT 21`

Definition at line 83 of file `ssl.h`.

Referenced by `ssl_close_notify()`, `ssl_parse_certificate()`, `ssl_read_record()`, and `ssl_write_certificate()`.

#### 5.111.2.27 `#define SSL_MSG_APPLICATION_DATA 23`

Definition at line 85 of file `ssl.h`.

Referenced by `ssl_read()`, and `ssl_write()`.

#### 5.111.2.28 `#define SSL_MSG_CHANGE_CIPHER_SPEC 20`

Definition at line 82 of file `ssl.h`.

Referenced by `ssl_parse_change_cipher_spec()`, and `ssl_write_change_cipher_spec()`.

#### 5.111.2.29 `#define SSL_MSG_HANDSHAKE 22`

Definition at line 84 of file `ssl.h`.

Referenced by `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `ssl_read_record()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

#### 5.111.2.30 `#define SSL_RSA_AES_128_SHA 47`

Definition at line 75 of file `ssl.h`.

Referenced by `main()`, `ssl_derive_keys()`, and `ssl_get_cipher()`.

#### 5.111.2.31 `#define SSL_RSA_AES_256_SHA 53`

Definition at line 76 of file `ssl.h`.

Referenced by `main()`, `ssl_derive_keys()`, and `ssl_get_cipher()`.

**5.111.2.32 #define SSL\_RSA\_DES\_168\_SHA 10**

Definition at line 73 of file ssl.h.

Referenced by main(), ssl\_derive\_keys(), and ssl\_get\_cipher().

**5.111.2.33 #define SSL\_RSA\_RC4\_128\_MD5 4**

Definition at line 71 of file ssl.h.

Referenced by main(), ssl\_derive\_keys(), and ssl\_get\_cipher().

**5.111.2.34 #define SSL\_RSA\_RC4\_128\_SHA 5**

Definition at line 72 of file ssl.h.

Referenced by main(), ssl\_derive\_keys(), and ssl\_get\_cipher().

**5.111.2.35 #define SSL\_VERIFY\_NONE 0**

Definition at line 56 of file ssl.h.

Referenced by main(), ssl\_parse\_certificate(), ssl\_test(), and ssl\_write\_certificate\_request().

**5.111.2.36 #define SSL\_VERIFY\_OPTIONAL 1**

Definition at line 57 of file ssl.h.

Referenced by main(), and ssl\_parse\_certificate().

**5.111.2.37 #define SSL\_VERIFY\_REQUIRED 2**

Definition at line 58 of file ssl.h.

Referenced by ssl\_parse\_certificate().

**5.111.2.38 #define TLS\_EXT\_SERVERNAME 0**

Definition at line 106 of file ssl.h.

Referenced by ssl\_write\_client\_hello().

**5.111.2.39 #define TLS\_EXT\_SERVERNAME\_HOSTNAME 0**

Definition at line 107 of file ssl.h.

Referenced by ssl\_write\_client\_hello().

**5.111.2.40 #define XYSSL\_ERR\_SSL\_BAD\_HS\_CERTIFICATE -0xA800**

Definition at line 35 of file ssl.h.

Referenced by ssl\_parse\_certificate().

**5.111.2.41 #define XYSSL\_ERR\_SSL\_BAD\_HS\_CERTIFICATE\_REQUEST -0xB000**

Definition at line 36 of file ssl.h.

**5.111.2.42 #define XYSSL\_ERR\_SSL\_BAD\_HS\_CERTIFICATE\_VERIFY -0xD000**

Definition at line 40 of file ssl.h.

Referenced by ssl\_parse\_certificate\_verify().

**5.111.2.43 #define XYSSL\_ERR\_SSL\_BAD\_HS\_CHANGE\_CIPHER\_SPEC -0xD800**

Definition at line 41 of file ssl.h.

Referenced by ssl\_parse\_change\_cipher\_spec().

**5.111.2.44 #define XYSSL\_ERR\_SSL\_BAD\_HS\_CLIENT\_HELLO -0x9800**

Definition at line 33 of file ssl.h.

Referenced by ssl\_parse\_client\_hello().

**5.111.2.45 #define XYSSL\_ERR\_SSL\_BAD\_HS\_CLIENT\_KEY\_EXCHANGE -0xC800**

Definition at line 39 of file ssl.h.

Referenced by ssl\_parse\_client\_key\_exchange().

**5.111.2.46 #define XYSSL\_ERR\_SSL\_BAD\_HS\_FINISHED -0xE000**

Definition at line 42 of file ssl.h.

Referenced by ssl\_parse\_finished().

**5.111.2.47 #define XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_HELLO -0xA000**

Definition at line 34 of file ssl.h.

Referenced by ssl\_parse\_server\_hello().

**5.111.2.48 #define XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_HELLO\_DONE -0xC000**

Definition at line 38 of file ssl.h.

Referenced by ssl\_parse\_server\_hello\_done().

**5.111.2.49 #define XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_KEY\_EXCHANGE -0xB800**

Definition at line 37 of file ssl.h.

Referenced by ssl\_parse\_server\_key\_exchange().

**5.111.2.50 #define XYSSL\_ERR\_SSL\_BAD\_INPUT\_DATA -0x1800**

Definition at line 17 of file ssl.h.

Referenced by ssl\_handshake\_client(), ssl\_handshake\_server(), ssl\_set\_hostname(), and tls1\_prf().

**5.111.2.51 #define XYSSL\_ERR\_SSL\_CA\_CHAIN\_REQUIRED -0x7000**

Definition at line 28 of file ssl.h.

Referenced by ssl\_parse\_certificate().

**5.111.2.52 #define XYSSL\_ERR\_SSL\_CERTIFICATE\_REQUIRED -0x6000**

Definition at line 26 of file ssl.h.

Referenced by ssl\_write\_certificate().

**5.111.2.53 #define XYSSL\_ERR\_SSL\_CERTIFICATE\_TOO\_LARGE -0x5800**

Definition at line 25 of file ssl.h.

Referenced by ssl\_write\_certificate().

**5.111.2.54 #define XYSSL\_ERR\_SSL\_FATAL\_ALERT\_MESSAGE -0x8000**

Definition at line 30 of file ssl.h.

Referenced by ssl\_read\_record().

**5.111.2.55 #define XYSSL\_ERR\_SSL\_FEATURE\_UNAVAILABLE -0x1000**

Definition at line 16 of file ssl.h.

Referenced by ssl\_decrypt\_buf(), ssl\_derive\_keys(), ssl\_encrypt\_buf(), ssl\_handshake(), ssl\_parse\_client\_key\_exchange(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_client\_key\_exchange(), and ssl\_write\_server\_key\_exchange().

**5.111.2.56 #define XYSSL\_ERR\_SSL\_INVALID\_MAC -0x2000**

Definition at line 18 of file ssl.h.

Referenced by ssl\_decrypt\_buf().

**5.111.2.57 #define XYSSL\_ERR\_SSL\_INVALID\_MODULUS\_SIZE -0x3000**

Definition at line 20 of file ssl.h.

**5.111.2.58 #define XYSSL\_ERR\_SSL\_INVALID\_RECORD -0x2800**

Definition at line 19 of file ssl.h.

Referenced by ssl\_read\_record().

#### **5.111.2.59 #define XYSSL\_ERR\_SSL\_NO\_CIPHER\_CHOSEN -0x4000**

Definition at line 22 of file ssl.h.

Referenced by ssl\_parse\_client\_hello().

#### **5.111.2.60 #define XYSSL\_ERR\_SSL\_NO\_CLIENT\_CERTIFICATE -0x5000**

Definition at line 24 of file ssl.h.

Referenced by ssl\_parse\_certificate().

#### **5.111.2.61 #define XYSSL\_ERR\_SSL\_NO\_SESSION\_FOUND -0x4800**

Definition at line 23 of file ssl.h.

#### **5.111.2.62 #define XYSSL\_ERR\_SSL\_PEER\_CLOSE\_NOTIFY -0x9000**

Definition at line 32 of file ssl.h.

Referenced by main(), ssl\_read\_record(), and ssl\_test().

#### **5.111.2.63 #define XYSSL\_ERR\_SSL\_PEER\_VERIFY\_FAILED -0x8800**

Definition at line 31 of file ssl.h.

#### **5.111.2.64 #define XYSSL\_ERR\_SSL\_PRIVATE\_KEY\_REQUIRED -0x6800**

Definition at line 27 of file ssl.h.

Referenced by ssl\_write\_certificate\_verify().

#### **5.111.2.65 #define XYSSL\_ERR\_SSL\_UNEXPECTED\_MESSAGE -0x7800**

Definition at line 29 of file ssl.h.

Referenced by ssl\_parse\_certificate(), ssl\_parse\_certificate\_request(), ssl\_parse\_change\_cipher\_spec(), ssl\_parse\_finished(), ssl\_parse\_server\_hello(), ssl\_parse\_server\_hello\_done(), ssl\_parse\_server\_key\_exchange(), and ssl\_read().

#### **5.111.2.66 #define XYSSL\_ERR\_SSL\_UNKNOWN\_CIPHER -0x3800**

Definition at line 21 of file ssl.h.

### 5.111.3 Typedef Documentation

#### 5.111.3.1 typedef struct \_ssl\_context ssl\_context

Definition at line 134 of file ssl.h.

#### 5.111.3.2 typedef struct \_ssl\_session ssl\_session

Definition at line 133 of file ssl.h.

### 5.111.4 Enumeration Type Documentation

#### 5.111.4.1 enum ssl\_states

Enumerator:

*SSL\_HELLO\_REQUEST*  
*SSL\_CLIENT\_HELLO*  
*SSL\_SERVER\_HELLO*  
*SSL\_SERVER\_CERTIFICATE*  
*SSL\_SERVER\_KEY\_EXCHANGE*  
*SSL\_CERTIFICATE\_REQUEST*  
*SSL\_SERVER\_HELLO\_DONE*  
*SSL\_CLIENT\_CERTIFICATE*  
*SSL\_CLIENT\_KEY\_EXCHANGE*  
*SSL\_CERTIFICATE\_VERIFY*  
*SSL\_CLIENT\_CHANGE\_CIPHER\_SPEC*  
*SSL\_CLIENT\_FINISHED*  
*SSL\_SERVER\_CHANGE\_CIPHER\_SPEC*  
*SSL\_SERVER\_FINISHED*  
*SSL\_FLUSH\_BUFFERS*  
*SSL\_HANDSHAKE\_OVER*

Definition at line 112 of file ssl.h.

### 5.111.5 Function Documentation

#### 5.111.5.1 void ssl\_calc\_verify (ssl\_context \* ssl, unsigned char hash[36])

Definition at line 335 of file ssl\_tls.c.

References `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_session::master`, `md5()`, `md5_finish()`, `md5_starts()`, `md5_update()`, `_ssl_context::minor_ver`, `_ssl_context::session`, `sha1()`, `sha1_finish()`, `sha1_starts()`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, and `SSL_MINOR_VERSION_0`.

Referenced by `ssl_parse_certificate_verify()`, and `ssl_write_certificate_verify()`.

### 5.111.5.2 int ssl\_close\_notify (ssl\_context \* ssl)

Notify the peer that the connection is being closed.

Definition at line 1904 of file ssl\_tls.c.

References `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_ALERT_CLOSE_NOTIFY`, `SSL_ALERT_WARNING`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_flush_output()`, `SSL_HANDSHAKE_OVER`, `SSL_MSG_ALERT`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `main()`, and `ssl_test()`.

### 5.111.5.3 int ssl\_derive\_keys (ssl\_context \* ssl)

Definition at line 106 of file ssl\_tls.c.

References `aes_setkey_dec()`, `aes_setkey_enc()`, `arc4_setup()`, `_ssl_session::cipher`, `_ssl_context::ctx_dec`, `_ssl_context::ctx_enc`, `des3_set3key_dec()`, `des3_set3key_enc()`, `_ssl_context::endpoint`, `_ssl_context::iv_dec`, `_ssl_context::iv_enc`, `_ssl_context::ivlen`, `_ssl_context::keylen`, `_ssl_context::mac_dec`, `_ssl_context::mac_enc`, `_ssl_context::maclen`, `_ssl_session::master`, `md5()`, `md5_finish()`, `md5_starts()`, `md5_update()`, `_ssl_context::minlen`, `_ssl_context::minor_ver`, `_ssl_context::pmslen`, `_ssl_context::premaster`, `_ssl_context::randbytes`, `_ssl_context::resume`, `_ssl_context::session`, `sha1()`, `sha1_finish()`, `sha1_starts()`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_EDH_RSA_AES_256_SHA`, `SSL_EDH_RSA_DES_168_SHA`, `ssl_get_cipher()`, `SSL_IS_CLIENT`, `SSL_MINOR_VERSION_0`, `SSL_RSA_AES_128_SHA`, `SSL_RSA_AES_256_SHA`, `SSL_RSA_DES_168_SHA`, `SSL_RSA_RC4_128_MD5`, `SSL_RSA_RC4_128_SHA`, `tls1_prf()`, and `XYSSL_ERR_SSL_FEATURE_UNAVAILABLE`.

Referenced by `ssl_parse_client_key_exchange()`, `ssl_parse_server_hello()`, `ssl_write_client_key_exchange()`, and `ssl_write_server_hello()`.

### 5.111.5.4 int ssl\_fetch\_input (ssl\_context \* ssl, int nb\_want)

Definition at line 727 of file ssl\_tls.c.

References `_ssl_context::f_recv`, `_ssl_context::in_hdr`, `_ssl_context::in_left`, `_ssl_context::p_recv`, `SSL_DEBUG_MSG`, and `SSL_DEBUG_RET`.

Referenced by `ssl_parse_client_hello()`, and `ssl_read_record()`.

### 5.111.5.5 int ssl\_flush\_output (ssl\_context \* ssl)

Definition at line 756 of file ssl\_tls.c.

References `buf`, `_ssl_context::f_send`, `_ssl_context::out_hdr`, `_ssl_context::out_left`, `_ssl_context::out_msglen`, `_ssl_context::p_send`, `SSL_DEBUG_MSG`, and `SSL_DEBUG_RET`.

Referenced by `ssl_close_notify()`, `ssl_handshake_client()`, `ssl_handshake_server()`, `ssl_write()`, and `ssl_write_record()`.

### 5.111.5.6 void ssl\_free (ssl\_context \* ssl)

Free an SSL context.

Definition at line 1938 of file ssl\_tls.c.

References `_ssl_context::dhm_ctx`, `dhm_free()`, `_ssl_context::hostname`, `_ssl_context::hostname_len`, `_ssl_context::in_ctr`, `_ssl_context::out_ctr`, `_ssl_context::peer_cert`, `SSL_BUFFER_LEN`, `SSL_DEBUG_MSG`, and `x509_free()`.

Referenced by `main()`, and `ssl_test()`.

#### 5.111.5.7 `int ssl_get_bytes_avail (ssl_context * ssl)`

Return the number of data bytes available to read.

##### Parameters

*ssl* SSL context

##### Returns

how many bytes are available in the read buffer

Definition at line 1691 of file `ssl_tls.c`.

References `_ssl_context::in_msglen`, and `_ssl_context::in_offt`.

#### 5.111.5.8 `char* ssl_get_cipher (ssl_context * ssl)`

Return the name of the current cipher.

##### Parameters

*ssl* SSL context

##### Returns

a string containing the cipher name

Definition at line 1701 of file `ssl_tls.c`.

References `_ssl_session::cipher`, `_ssl_context::session`, `SSL_EDH_RSA_AES_256_SHA`, `SSL_EDH_RSA_DES_168_SHA`, `SSL_RSA_AES_128_SHA`, `SSL_RSA_AES_256_SHA`, `SSL_RSA_DES_168_SHA`, `SSL_RSA_RC4_128_MD5`, and `SSL_RSA_RC4_128_SHA`.

Referenced by `main()`, and `ssl_derive_keys()`.

#### 5.111.5.9 `int ssl_get_verify_result (ssl_context * ssl)`

Return the result of the certificate verification.

##### Parameters

*ssl* SSL context

##### Returns

0 if successful, or a combination of: `BADCERT_EXPIRED` `BADCERT_REVOKED` `BADCERT_CN_MISMATCH` `BADCERT_NOT_TRUSTED`

Definition at line 1696 of file `ssl_tls.c`.

References `_ssl_context::verify_result`.

Referenced by `main()`.



#### 5.111.5.10 `int ssl_handshake(ssl_context *ssl)`

Perform the SSL handshake.

##### Parameters

*ssl* SSL context

##### Returns

0 if successful, XYSSL\_ERR\_NET\_TRY\_AGAIN, or a specific SSL error code.

Definition at line 1767 of file `ssl_tls.c`.

References `_ssl_context::endpoint`, `SSL_DEBUG_MSG`, `ssl_handshake_client()`, `ssl_handshake_server()`, `SSL_IS_CLIENT`, `SSL_IS_SERVER`, and `XYSSL_ERR_SSL_FEATURE_UNAVAILABLE`.

Referenced by `main()`, `ssl_read()`, and `ssl_write()`.

#### 5.111.5.11 `int ssl_handshake_client(ssl_context *ssl)`

Definition at line 660 of file `ssl_cli.c`.

References `SSL_CERTIFICATE_REQUEST`, `SSL_CERTIFICATE_VERIFY`, `SSL_CLIENT_CERTIFICATE`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_CLIENT_FINISHED`, `SSL_CLIENT_HELLO`, `SSL_CLIENT_KEY_EXCHANGE`, `SSL_DEBUG_MSG`, `SSL_FLUSH_BUFFERS`, `ssl_flush_output()`, `SSL_HANDSHAKE_OVER`, `SSL_HELLO_REQUEST`, `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `SSL_SERVER_CERTIFICATE`, `SSL_SERVER_CHANGE_CIPHER_SPEC`, `SSL_SERVER_FINISHED`, `SSL_SERVER_HELLO`, `SSL_SERVER_HELLO_DONE`, `SSL_SERVER_KEY_EXCHANGE`, `ssl_write_certificate()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_BAD_INPUT_DATA`.

Referenced by `ssl_handshake()`.

#### 5.111.5.12 `int ssl_handshake_server(ssl_context *ssl)`

Definition at line 819 of file `ssl_srv.c`.

References `SSL_CERTIFICATE_REQUEST`, `SSL_CERTIFICATE_VERIFY`, `SSL_CLIENT_CERTIFICATE`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_CLIENT_FINISHED`, `SSL_CLIENT_HELLO`, `SSL_CLIENT_KEY_EXCHANGE`, `SSL_DEBUG_MSG`, `SSL_FLUSH_BUFFERS`, `ssl_flush_output()`, `SSL_HANDSHAKE_OVER`, `SSL_HELLO_REQUEST`, `ssl_parse_certificate()`, `ssl_parse_certificate_verify()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `SSL_SERVER_CERTIFICATE`, `SSL_SERVER_CHANGE_CIPHER_SPEC`, `SSL_SERVER_FINISHED`, `SSL_SERVER_HELLO`, `SSL_SERVER_HELLO_DONE`, `SSL_SERVER_KEY_EXCHANGE`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_change_cipher_spec()`, `ssl_write_finished()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, `ssl_write_server_key_exchange()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_BAD_INPUT_DATA`.

Referenced by `ssl_handshake()`.

#### 5.111.5.13 `int ssl_init(ssl_context *ssl)`

Initialize an SSL context.

**Parameters**

*ssl* SSL context

**Returns**

0 if successful, or 1 if memory allocation failed

Definition at line 1542 of file `ssl_tls.c`.

References `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::hostname`, `_ssl_context::hostname_len`, `_ssl_context::in_ctr`, `_ssl_context::in_hdr`, `_ssl_context::in_msg`, `md5_starts()`, `_ssl_context::out_ctr`, `_ssl_context::out_hdr`, `_ssl_context::out_msg`, `sha1_starts()`, `SSL_BUFFER_LEN`, and `SSL_DEBUG_MSG`.

Referenced by `main()`, and `ssl_test()`.

**5.111.5.14 int ssl\_parse\_certificate (ssl\_context \* ssl)**

Definition at line 1140 of file `ssl_tls.c`.

References `_ssl_context::authmode`, `_ssl_context::ca_chain`, `_ssl_context::endpoint`, `_ssl_context::in_hhlen`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `int`, `_ssl_context::minor_ver`, `_ssl_context::peer_cert`, `_ssl_context::peer_cn`, `SSL_ALERT_NO_CERTIFICATE`, `SSL_ALERT_WARNING`, `SSL_DEBUG_CRT`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HS_CERTIFICATE`, `SSL_IS_SERVER`, `SSL_MINOR_VERSION_0`, `SSL_MSG_ALERT`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, `SSL_VERIFY_NONE`, `SSL_VERIFY_OPTIONAL`, `SSL_VERIFY_REQUIRED`, `_ssl_context::state`, `_ssl_context::verify_result`, `x509parse_cert()`, `x509parse_verify()`, `XYSSL_ERR_SSL_BAD_HS_CERTIFICATE`, `XYSSL_ERR_SSL_CA_CHAIN_REQUIRED`, `XYSSL_ERR_SSL_NO_CLIENT_CERTIFICATE`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

**5.111.5.15 int ssl\_parse\_change\_cipher\_spec (ssl\_context \* ssl)**

Definition at line 1311 of file `ssl_tls.c`.

References `_ssl_context::do_crypt`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_MSG_CHANGE_CIPHER_SPEC`, `ssl_read_record()`, `_ssl_context::state`, `XYSSL_ERR_SSL_BAD_HS_CHANGE_CIPHER_SPEC`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

**5.111.5.16 int ssl\_parse\_finished (ssl\_context \* ssl)**

Definition at line 1480 of file `ssl_tls.c`.

References `buf`, `_ssl_context::do_crypt`, `_ssl_context::endpoint`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::in_hhlen`, `_ssl_context::in_msg`, `_ssl_context::in_msgtype`, `md5()`, `_ssl_context::minor_ver`, `_ssl_context::resume`, `sha1()`, `ssl_calc_finished()`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HANDSHAKE_OVER`, `SSL_HS_FINISHED`, `SSL_IS_CLIENT`, `SSL_IS_SERVER`, `SSL_MINOR_VERSION_0`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, `_ssl_context::state`, `XYSSL_ERR_SSL_BAD_HS_FINISHED`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

**5.111.5.17 int ssl\_read (ssl\_context \* ssl, unsigned char \* buf, int len)**

Read at most 'len' application data bytes.

**Parameters**

*ssl* SSL context

*buf* buffer that will hold the data

*len* how many bytes must be read

**Returns**

This function returns the number of bytes read, or a negative error code.

Definition at line 1791 of file ssl\_tls.c.

References `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `_ssl_context::in_offt`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_handshake()`, `SSL_HANDSHAKE_OVER`, `SSL_MSG_APPLICATION_DATA`, `ssl_read_record()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `main()`, and `ssl_test()`.

**5.111.5.18 int ssl\_read\_record (ssl\_context \* ssl)**

Definition at line 842 of file ssl\_tls.c.

References `_ssl_context::do_crypt`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::in_hdr`, `_ssl_context::in_hhlen`, `_ssl_context::in_left`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `_ssl_context::major_ver`, `md5_update()`, `_ssl_context::minlen`, `_ssl_context::minor_ver`, `sha1_update()`, `SSL_ALERT_CLOSE_NOTIFY`, `SSL_ALERT_FATAL`, `SSL_ALERT_WARNING`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_decrypt_buf()`, `ssl_fetch_input()`, `SSL_MAX_CONTENT_LEN`, `SSL_MINOR_VERSION_0`, `SSL_MINOR_VERSION_1`, `SSL_MSG_ALERT`, `SSL_MSG_HANDSHAKE`, `XYSSL_ERR_SSL_FATAL_ALERT_MESSAGE`, `XYSSL_ERR_SSL_INVALID_RECORD`, and `XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY`.

Referenced by `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, and `ssl_read()`.

**5.111.5.19 void ssl\_set\_authmode (ssl\_context \* ssl, int authmode)**

Set the certificate verification mode.

**Parameters**

*ssl* SSL context

*mode* can be:

`SSL_VERIFY_NONE`: peer certificate is not checked (default), this is insecure and SHOULD be avoided.

`SSL_VERIFY_OPTIONAL`: peer certificate is checked, however the handshake continues even if verification failed; `ssl_get_verify_result()` can be called after the handshake is complete.

`SSL_VERIFY_REQUIRED`: peer *must* present a valid certificate, handshake is aborted if verification failed.

Definition at line 1589 of file ssl\_tls.c.

References `_ssl_context::authmode`.

Referenced by `main()`, and `ssl_test()`.

**5.111.5.20** `void ssl_set_bio (ssl_context * ssl, int(*) (void *, unsigned char *, int) f_recv, void * p_recv, int(*) (void *, unsigned char *, int) f_send, void * p_send)`

Set the underlying BIO read and write callbacks.

#### Parameters

*ssl* SSL context

*f\_recv* read callback

*p\_recv* read parameter

*f\_send* write callback

*p\_send* write parameter

Definition at line 1610 of file ssl\_tls.c.

References `_ssl_context::f_recv`, `_ssl_context::f_send`, `_ssl_context::p_recv`, and `_ssl_context::p_send`.

Referenced by `main()`, and `ssl_test()`.

**5.111.5.21** `void ssl_set_ca_chain (ssl_context * ssl, x509_cert * ca_chain, char * peer_cn)`

Set the data required to verify peer certificate.

#### Parameters

*ssl* SSL context

*ca\_chain* trusted CA chain

*peer\_cn* expected peer CommonName (or NULL)

#### Note

TODO: add two more parameters: depth and crl

Definition at line 1641 of file ssl\_tls.c.

References `_ssl_context::ca_chain`, and `_ssl_context::peer_cn`.

Referenced by `main()`, and `ssl_test()`.

**5.111.5.22** `void ssl_set_ciphers (ssl_context * ssl, int * ciphers)`

Set the list of allowed ciphersuites.

#### Parameters

*ssl* SSL context

*ciphers* 0-terminated list of allowed ciphers

Definition at line 1636 of file ssl\_tls.c.

References `_ssl_context::ciphers`.

Referenced by `main()`, and `ssl_test()`.

#### 5.111.5.23 `void ssl_set_dbg (ssl_context * ssl, void(*) (void *, int, char *) f_dbg, void * p_dbg)`

Set the debug callback.

##### Parameters

*ssl* SSL context

*f\_dbg* debug function

*p\_dbg* debug parameter

Definition at line 1602 of file ssl\_tls.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

Referenced by `main()`, and `ssl_test()`.

#### 5.111.5.24 `int ssl_set_dh_param (ssl_context * ssl, char * dhm_P, char * dhm_G)`

Set the Diffie-Hellman public P and G values, read as hexadecimal strings (server-side only).

##### Parameters

*ssl* SSL context

*dhm\_P* Diffie-Hellman-Merkle modulus

*dhm\_G* Diffie-Hellman-Merkle generator

##### Returns

0 if successful

Definition at line 1655 of file ssl\_tls.c.

References `_ssl_context::dhm_ctx`, `dhm_context::G`, `mpi_read_string()`, `dhm_context::P`, and `SSL_DEBUG_RET`.

Referenced by `main()`, and `ssl_test()`.

#### 5.111.5.25 `void ssl_set_endpoint (ssl_context * ssl, int endpoint)`

Set the current endpoint type.

##### Parameters

*ssl* SSL context

*endpoint* must be `SSL_IS_CLIENT` or `SSL_IS_SERVER`

Definition at line 1584 of file ssl\_tls.c.

References `_ssl_context::endpoint`.

Referenced by `main()`, and `ssl_test()`.

**5.111.5.26** `int ssl_set_hostname (ssl_context * ssl, char * hostname)`

Set hostname for ServerName TLS Extension.

**Parameters**

*ssl* SSL context

*hostname* the server hostname

**Returns**

0 if successful

Definition at line 1674 of file `ssl_tls.c`.

References `_ssl_context::hostname`, `_ssl_context::hostname_len`, and `XYSSL_ERR_SSL_BAD_INPUT_DATA`.

Referenced by `main()`.

**5.111.5.27** `void ssl_set_own_cert (ssl_context * ssl, x509_cert * own_cert, rsa_context * rsa_key)`

Set own certificate and private key.

**Parameters**

*ssl* SSL context

*own\_cert* own public certificate

*rsa\_key* own private RSA key

Definition at line 1648 of file `ssl_tls.c`.

References `_ssl_context::own_cert`, and `_ssl_context::rsa_key`.

Referenced by `main()`, and `ssl_test()`.

**5.111.5.28** `void ssl_set_rng (ssl_context * ssl, int(*) (void *) f_rng, void * p_rng)`

Set the random number generator callback.

**Parameters**

*ssl* SSL context

*f\_rng* RNG function

*p\_rng* RNG parameter

Definition at line 1594 of file `ssl_tls.c`.

References `_ssl_context::f_rng`, and `_ssl_context::p_rng`.

Referenced by `main()`, and `ssl_test()`.

**5.111.5.29 void ssl\_set\_scb (ssl\_context \* *ssl*, int(\*) (ssl\_context \*) *s\_get*, int(\*) (ssl\_context \*) *s\_set*)**

Set the session callbacks (server-side only).

**Parameters**

*ssl* SSL context  
*s\_get* session get callback  
*s\_set* session set callback

Definition at line 1620 of file ssl\_tls.c.

References \_ssl\_context::s\_get, and \_ssl\_context::s\_set.

Referenced by main().

**5.111.5.30 void ssl\_set\_session (ssl\_context \* *ssl*, int *resume*, int *timeout*, ssl\_session \* *session*)**

Set the session resuming flag, timeout and data.

**Parameters**

*ssl* SSL context  
*resume* if 0 (default), the session will not be resumed  
*timeout* session timeout in seconds, or 0 (no timeout)  
*session* session context

Definition at line 1628 of file ssl\_tls.c.

References \_ssl\_context::resume, \_ssl\_context::session, and \_ssl\_context::timeout.

Referenced by main(), and ssl\_test().

**5.111.5.31 int ssl\_write (ssl\_context \* *ssl*, unsigned char \* *buf*, int *len*)**

Write exactly 'len' application data bytes.

**Parameters**

*ssl* SSL context  
*buf* buffer holding the data  
*len* how many bytes must be written

**Returns**

This function returns the number of bytes written, or a negative error code.

**Note**

When this function returns XYSSL\_ERR\_NET\_TRY\_AGAIN, it must be called later with the *\*same\** arguments, until it returns a positive value.

Definition at line 1857 of file `ssl_tls.c`.

References `_ssl_context::out_left`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_flush_output()`, `ssl_handshake()`, `SSL_HANDSHAKE_OVER`, `SSL_MAX_CONTENT_LEN`, `SSL_MSG_APPLICATION_DATA`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `main()`, and `ssl_test()`.

#### 5.111.5.32 `int ssl_write_certificate(ssl_context *ssl)`

Definition at line 1044 of file `ssl_tls.c`.

References `_ssl_context::client_auth`, `_ssl_context::endpoint`, `_x509_buf::len`, `_ssl_context::minor_ver`, `_x509_cert::next`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `_ssl_context::own_cert`, `_x509_buf::p`, `_x509_cert::raw`, `SSL_ALERT_NO_CERTIFICATE`, `SSL_ALERT_WARNING`, `SSL_DEBUG_CRT`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HS_CERTIFICATE`, `SSL_IS_CLIENT`, `SSL_MAX_CONTENT_LEN`, `SSL_MINOR_VERSION_0`, `SSL_MSG_ALERT`, `SSL_MSG_HANDSHAKE`, `ssl_write_record()`, `_ssl_context::state`, `XYSSL_ERR_SSL_CERTIFICATE_REQUIRED`, and `XYSSL_ERR_SSL_CERTIFICATE_TOO_LARGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

#### 5.111.5.33 `int ssl_write_change_cipher_spec(ssl_context *ssl)`

Definition at line 1287 of file `ssl_tls.c`.

References `_ssl_context::do_crypt`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_MSG_CHANGE_CIPHER_SPEC`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

#### 5.111.5.34 `int ssl_write_finished(ssl_context *ssl)`

Definition at line 1433 of file `ssl_tls.c`.

References `_ssl_context::do_crypt`, `_ssl_context::endpoint`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `md5()`, `_ssl_context::minor_ver`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `_ssl_context::resume`, `sha1()`, `ssl_calc_finished()`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HANDSHAKE_OVER`, `SSL_HS_FINISHED`, `SSL_IS_CLIENT`, `SSL_MINOR_VERSION_0`, `SSL_MSG_HANDSHAKE`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

#### 5.111.5.35 `int ssl_write_record(ssl_context *ssl)`

Definition at line 786 of file `ssl_tls.c`.

References `_ssl_context::do_crypt`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::major_ver`, `md5_update()`, `_ssl_context::minor_ver`, `_ssl_context::out_hdr`, `_ssl_context::out_left`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_encrypt_buf()`, `ssl_flush_output()`, and `SSL_MSG_HANDSHAKE`.



Referenced by `ssl_close_notify()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

## 5.111.6 Variable Documentation

### 5.111.6.1 `int ssl_default_ciphers[ ]`

Definition at line 1739 of file `ssl_tls.c`.

Referenced by `main()`, and `ssl_test()`.

## 5.112 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/timing.h File Reference

### Data Structures

- struct [hr\\_time](#)  
*timer structure*

### Functions

- unsigned long [hardclock](#) (void)  
*Return the CPU cycle counter value.*
- unsigned long [get\\_timer](#) (struct [hr\\_time](#) \*val, int reset)  
*Return the elapsed time in milliseconds.*
- void [set\\_alarm](#) (int seconds)  
*Setup an alarm clock.*
- void [m\\_sleep](#) (int milliseconds)  
*Sleep for a certain amount of time.*

### Variables

- [int alarmed](#)

### 5.112.1 Detailed Description

Definition in file [timing.h](#).

### 5.112.2 Function Documentation

#### 5.112.2.1 unsigned long [get\\_timer](#) (struct [hr\\_time](#) \* val, int reset)

Return the elapsed time in milliseconds.

#### Parameters

*val* points to a timer structure  
*reset* if set to 1, the timer is restarted

Definition at line 225 of file [timing.c](#).

References [\\_hr\\_time::start](#).

Referenced by [ssl\\_test\(\)](#).

#### 5.112.2.2 unsigned long hardclock (void)

Return the CPU cycle counter value.

Definition at line 155 of file timing.c.

References tv\_init.

Referenced by main().

#### 5.112.2.3 void m\_sleep (int milliseconds)

Sleep for a certain amount of time.

Definition at line 258 of file timing.c.

#### 5.112.2.4 void set\_alarm (int seconds)

Setup an alarm clock.

##### Parameters

*seconds* delay before the "alarmed" flag is set

Definition at line 251 of file timing.c.

References sighandler().

Referenced by main().

### 5.112.3 Variable Documentation

#### 5.112.3.1 int alarmed

Definition at line 179 of file timing.c.

Referenced by main().

## 5.113 /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/x509.h File Reference

```
#include "xyssl/rsa.h"
```

### Data Structures

- struct [\\_x509\\_buf](#)
- struct [\\_x509\\_name](#)
- struct [\\_x509\\_time](#)
- struct [\\_x509\\_cert](#)
- struct [\\_x509\\_node](#)
- struct [\\_x509\\_raw](#)

### Defines

- #define [XYSSL\\_ERR\\_ASN1\\_OUT\\_OF\\_DATA](#) -0x0014
- #define [XYSSL\\_ERR\\_ASN1\\_UNEXPECTED\\_TAG](#) -0x0016
- #define [XYSSL\\_ERR\\_ASN1\\_INVALID\\_LENGTH](#) -0x0018
- #define [XYSSL\\_ERR\\_ASN1\\_LENGTH\\_MISMATCH](#) -0x001A
- #define [XYSSL\\_ERR\\_ASN1\\_INVALID\\_DATA](#) -0x001C
- #define [XYSSL\\_ERR\\_X509\\_FEATURE\\_UNAVAILABLE](#) -0x0020
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_INVALID\\_PEM](#) -0x0040
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_INVALID\\_FORMAT](#) -0x0060
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_INVALID\\_VERSION](#) -0x0080
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_INVALID\\_SERIAL](#) -0x00A0
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_INVALID\\_ALG](#) -0x00C0
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_INVALID\\_NAME](#) -0x00E0
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_INVALID\\_DATE](#) -0x0100
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_INVALID\\_PUBKEY](#) -0x0120
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_INVALID\\_SIGNATURE](#) -0x0140
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_INVALID\\_EXTENSIONS](#) -0x0160
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_UNKNOWN\\_VERSION](#) -0x0180
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_UNKNOWN\\_SIG\\_ALG](#) -0x01A0
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_UNKNOWN\\_PK\\_ALG](#) -0x01C0
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_SIG\\_MISMATCH](#) -0x01E0
- #define [XYSSL\\_ERR\\_X509\\_CERT\\_VERIFY\\_FAILED](#) -0x0200
- #define [XYSSL\\_ERR\\_X509\\_KEY\\_INVALID\\_PEM](#) -0x0220
- #define [XYSSL\\_ERR\\_X509\\_KEY\\_INVALID\\_VERSION](#) -0x0240
- #define [XYSSL\\_ERR\\_X509\\_KEY\\_INVALID\\_FORMAT](#) -0x0260
- #define [XYSSL\\_ERR\\_X509\\_KEY\\_INVALID\\_ENC\\_IV](#) -0x0280
- #define [XYSSL\\_ERR\\_X509\\_KEY\\_UNKNOWN\\_ENC\\_ALG](#) -0x02A0
- #define [XYSSL\\_ERR\\_X509\\_KEY\\_PASSWORD\\_REQUIRED](#) -0x02C0
- #define [XYSSL\\_ERR\\_X509\\_KEY\\_PASSWORD\\_MISMATCH](#) -0x02E0
- #define [XYSSL\\_ERR\\_X509\\_POINT\\_ERROR](#) -0x0300
- #define [XYSSL\\_ERR\\_X509\\_VALUE\\_TO\\_LENGTH](#) -0x0320
- #define [BADCERT\\_EXPIRED](#) 1
- #define [BADCERT\\_REVOKED](#) 2
- #define [BADCERT\\_CN\\_MISMATCH](#) 4

- #define [BADCERT\\_NOT\\_TRUSTED](#) 8
- #define [ASN1\\_BOOLEAN](#) 0x01
- #define [ASN1\\_INTEGER](#) 0x02
- #define [ASN1\\_BIT\\_STRING](#) 0x03
- #define [ASN1\\_OCTET\\_STRING](#) 0x04
- #define [ASN1\\_NULL](#) 0x05
- #define [ASN1\\_OID](#) 0x06
- #define [ASN1\\_UTF8\\_STRING](#) 0x0C
- #define [ASN1\\_SEQUENCE](#) 0x10
- #define [ASN1\\_SET](#) 0x11
- #define [ASN1\\_PRINTABLE\\_STRING](#) 0x13
- #define [ASN1\\_T61\\_STRING](#) 0x14
- #define [ASN1\\_IA5\\_STRING](#) 0x16
- #define [ASN1\\_UTC\\_TIME](#) 0x17
- #define [ASN1\\_UNIVERSAL\\_STRING](#) 0x1C
- #define [ASN1\\_BMP\\_STRING](#) 0x1E
- #define [ASN1\\_PRIMITIVE](#) 0x00
- #define [ASN1\\_CONSTRUCTED](#) 0x20
- #define [ASN1\\_CONTEXT\\_SPECIFIC](#) 0x80
- #define [X520\\_COMMON\\_NAME](#) 3
- #define [X520\\_COUNTRY](#) 6
- #define [X520\\_LOCALITY](#) 7
- #define [X520\\_STATE](#) 8
- #define [X520\\_ORGANIZATION](#) 10
- #define [X520\\_ORG\\_UNIT](#) 11
- #define [PKCS9\\_EMAIL](#) 1
- #define [X509\\_OUTPUT\\_DER](#) 0x01
- #define [X509\\_OUTPUT\\_PEM](#) 0x02
- #define [PEM\\_LINE\\_LENGTH](#) 72
- #define [X509\\_ISSUER](#) 0x01
- #define [X509\\_SUBJECT](#) 0x02
- #define [OID\\_X520](#) "\x55\x04"
- #define [OID\\_CN](#) "\x55\x04\x03"
- #define [OID\\_PKCS1](#) "\x2A\x86\x48\x86\xF7\x0D\x01\x01"
- #define [OID\\_PKCS1\\_RSA](#) "\x2A\x86\x48\x86\xF7\x0D\x01\x01\x01"
- #define [OID\\_PKCS1\\_RSA\\_SHA](#) "\x2A\x86\x48\x86\xF7\x0D\x01\x01\x05"
- #define [OID\\_PKCS9](#) "\x2A\x86\x48\x86\xF7\x0D\x01\x09"
- #define [OID\\_PKCS9\\_EMAIL](#) "\x2A\x86\x48\x86\xF7\x0D\x01\x09\x01"

## Typedefs

- typedef struct [\\_x509\\_buf](#) [x509\\_buf](#)
- typedef struct [\\_x509\\_name](#) [x509\\_name](#)
- typedef struct [\\_x509\\_time](#) [x509\\_time](#)
- typedef struct [\\_x509\\_cert](#) [x509\\_cert](#)
- typedef struct [\\_x509\\_node](#) [x509\\_node](#)
- typedef struct [\\_x509\\_raw](#) [x509\\_raw](#)

## Functions

- `int x509parse_cert (x509_cert *crt, unsigned char *buf, int buflen)`  
*Parse one or more certificates and add them to the chained list.*
- `int x509parse_crtfile (x509_cert *crt, char *path)`  
*Load one or more certificates and add them to the chained list.*
- `int x509parse_key (rsa_context *rsa, unsigned char *buf, int buflen, unsigned char *pwd, int pwrlen)`  
*Parse a private RSA key.*
- `int x509parse_keyfile (rsa_context *rsa, char *path, char *password)`  
*Load and parse a private RSA key.*
- `int x509parse_dn_gets (char *buf, char *end, x509_name *dn)`  
*Store the certificate DN in printable form into buf; no more than (end - buf) characters will be written.*
- `char * x509parse_cert_info (char *prefix, x509_cert *crt)`  
*Returns an informational string about the certificate.*
- `int x509parse_expired (x509_cert *crt)`  
*Return 0 if the certificate is still valid, or BADCERT\_EXPIRED.*
- `int x509parse_verify (x509_cert *crt, x509_cert *trust_ca, char *cn, int *flags)`  
*Verify the certificate signature.*
- `void x509_free (x509_cert *crt)`  
*Unallocate all certificate data.*
- `int x509_self_test (int verbose)`  
*Checkup routine.*

### 5.113.1 Detailed Description

Definition in file [x509.h](#).

### 5.113.2 Define Documentation

#### 5.113.2.1 `#define ASN1_BIT_STRING 0x03`

Definition at line 51 of file [x509.h](#).

Referenced by [x509\\_get\\_pubkey\(\)](#), and [x509\\_get\\_sig\(\)](#).

#### 5.113.2.2 `#define ASN1_BMP_STRING 0x1E`

Definition at line 63 of file [x509.h](#).

Referenced by [x509\\_get\\_name\(\)](#).

**5.113.2.3 #define ASN1\_BOOLEAN 0x01**

Definition at line 49 of file x509.h.

Referenced by `asn1_get_bool()`.

**5.113.2.4 #define ASN1\_CONSTRUCTED 0x20**

Definition at line 65 of file x509.h.

Referenced by `x509_get_alg()`, `x509_get_dates()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_uid()`, `x509_get_version()`, `x509parse_crt()`, and `x509parse_key()`.

**5.113.2.5 #define ASN1\_CONTEXT\_SPECIFIC 0x80**

Definition at line 66 of file x509.h.

Referenced by `x509_get_ext()`, `x509_get_serial()`, `x509_get_uid()`, and `x509_get_version()`.

**5.113.2.6 #define ASN1\_IA5\_STRING 0x16**

Definition at line 60 of file x509.h.

Referenced by `x509_get_name()`.

**5.113.2.7 #define ASN1\_INTEGER 0x02**

Definition at line 50 of file x509.h.

Referenced by `asn1_get_int()`, `asn1_get_mpi()`, and `x509_get_serial()`.

**5.113.2.8 #define ASN1\_NULL 0x05**

Definition at line 53 of file x509.h.

Referenced by `x509_get_alg()`.

**5.113.2.9 #define ASN1\_OCTET\_STRING 0x04**

Definition at line 52 of file x509.h.

Referenced by `x509_get_ext()`.

**5.113.2.10 #define ASN1\_OID 0x06**

Definition at line 54 of file x509.h.

Referenced by `x509_get_alg()`, and `x509_get_name()`.

**5.113.2.11 #define ASN1\_PRIMITIVE 0x00**

Definition at line 64 of file x509.h.

Referenced by x509\_get\_serial().

#### **5.113.2.12 #define ASN1\_PRINTABLE\_STRING 0x13**

Definition at line 58 of file x509.h.

Referenced by x509\_get\_name().

#### **5.113.2.13 #define ASN1\_SEQUENCE 0x10**

Definition at line 56 of file x509.h.

Referenced by x509\_get\_alg(), x509\_get\_dates(), x509\_get\_ext(), x509\_get\_name(), x509\_get\_pubkey(), x509parse\_crt(), and x509parse\_key().

#### **5.113.2.14 #define ASN1\_SET 0x11**

Definition at line 57 of file x509.h.

Referenced by x509\_get\_name().

#### **5.113.2.15 #define ASN1\_T61\_STRING 0x14**

Definition at line 59 of file x509.h.

Referenced by x509\_get\_name().

#### **5.113.2.16 #define ASN1\_UNIVERSAL\_STRING 0x1C**

Definition at line 62 of file x509.h.

Referenced by x509\_get\_name().

#### **5.113.2.17 #define ASN1.UTC\_TIME 0x17**

Definition at line 61 of file x509.h.

Referenced by x509\_get\_dates().

#### **5.113.2.18 #define ASN1\_UTF8\_STRING 0x0C**

Definition at line 55 of file x509.h.

Referenced by x509\_get\_name().

#### **5.113.2.19 #define BADCERT\_CN\_MISMATCH 4**

Definition at line 43 of file x509.h.

Referenced by main(), and x509parse\_verify().



**5.113.2.20 #define BADCERT\_EXPIRED 1**

Definition at line 41 of file x509.h.

Referenced by main(), and x509parse\_expired().

**5.113.2.21 #define BADCERT\_NOT\_TRUSTED 8**

Definition at line 44 of file x509.h.

Referenced by main(), and x509parse\_verify().

**5.113.2.22 #define BADCERT\_REVOKED 2**

Definition at line 42 of file x509.h.

Referenced by main().

**5.113.2.23 #define OID\_CN "\x55\x04\x03"**

Definition at line 86 of file x509.h.

Referenced by x509parse\_verify().

**5.113.2.24 #define OID\_PKCS1 "\x2A\x86\x48\x86\xF7\x0D\x01\x01"**

Definition at line 87 of file x509.h.

Referenced by x509parse\_crt().

**5.113.2.25 #define OID\_PKCS1\_RSA "\x2A\x86\x48\x86\xF7\x0D\x01\x01\x01"**

Definition at line 88 of file x509.h.

Referenced by x509\_get\_pubkey().

**5.113.2.26 #define OID\_PKCS1\_RSA\_SHA "\x2A\x86\x48\x86\xF7\x0D\x01\x01\x05"**

Definition at line 89 of file x509.h.

**5.113.2.27 #define OID\_PKCS9 "\x2A\x86\x48\x86\xF7\x0D\x01\x09"**

Definition at line 90 of file x509.h.

Referenced by x509parse\_dn\_gets().

**5.113.2.28 #define OID\_PKCS9\_EMAIL "\x2A\x86\x48\x86\xF7\x0D\x01\x09\x01"**

Definition at line 91 of file x509.h.

**5.113.2.29 #define OID\_X520 "\x55\x04"**

Definition at line 85 of file x509.h.

Referenced by x509parse\_dn\_gets().

**5.113.2.30 #define PEM\_LINE\_LENGTH 72**

Definition at line 81 of file x509.h.

**5.113.2.31 #define PKCS9\_EMAIL 1**

Definition at line 77 of file x509.h.

Referenced by x509parse\_dn\_gets().

**5.113.2.32 #define X509\_ISSUER 0x01**

Definition at line 82 of file x509.h.

**5.113.2.33 #define X509\_OUTPUT\_DER 0x01**

Definition at line 79 of file x509.h.

**5.113.2.34 #define X509\_OUTPUT\_PEM 0x02**

Definition at line 80 of file x509.h.

**5.113.2.35 #define X509\_SUBJECT 0x02**

Definition at line 83 of file x509.h.

**5.113.2.36 #define X520\_COMMON\_NAME 3**

Definition at line 71 of file x509.h.

Referenced by x509parse\_dn\_gets().

**5.113.2.37 #define X520\_COUNTRY 6**

Definition at line 72 of file x509.h.

Referenced by x509parse\_dn\_gets().

**5.113.2.38 #define X520\_LOCALITY 7**

Definition at line 73 of file x509.h.

Referenced by x509parse\_dn\_gets().

**5.113.2.39 #define X520\_ORG\_UNIT 11**

Definition at line 76 of file x509.h.

Referenced by x509parse\_dn\_gets().

**5.113.2.40 #define X520\_ORGANIZATION 10**

Definition at line 75 of file x509.h.

Referenced by x509parse\_dn\_gets().

**5.113.2.41 #define X520\_STATE 8**

Definition at line 74 of file x509.h.

Referenced by x509parse\_dn\_gets().

**5.113.2.42 #define XYSSL\_ERR\_ASN1\_INVALID\_DATA -0x001C**

Definition at line 13 of file x509.h.

**5.113.2.43 #define XYSSL\_ERR\_ASN1\_INVALID\_LENGTH -0x0018**

Definition at line 11 of file x509.h.

Referenced by asn1\_get\_bool(), asn1\_get\_int(), and asn1\_get\_len().

**5.113.2.44 #define XYSSL\_ERR\_ASN1\_LENGTH\_MISMATCH -0x001A**

Definition at line 12 of file x509.h.

Referenced by x509\_get\_alg(), x509\_get\_dates(), x509\_get\_ext(), x509\_get\_name(), x509\_get\_pubkey(), x509\_get\_version(), x509parse\_crt(), and x509parse\_key().

**5.113.2.45 #define XYSSL\_ERR\_ASN1\_OUT\_OF\_DATA -0x0014**

Definition at line 9 of file x509.h.

Referenced by asn1\_get\_len(), asn1\_get\_tag(), x509\_get\_name(), x509\_get\_pubkey(), and x509\_get\_serial().

**5.113.2.46 #define XYSSL\_ERR\_ASN1\_UNEXPECTED\_TAG -0x0016**

Definition at line 10 of file x509.h.

Referenced by asn1\_get\_tag(), x509\_get\_ext(), x509\_get\_name(), x509\_get\_serial(), x509\_get\_uid(), and x509\_get\_version().

**5.113.2.47 #define XYSSL\_ERR\_X509\_CERT\_INVALID\_ALG -0x00C0**

Definition at line 20 of file x509.h.

Referenced by x509\_get\_alg().

**5.113.2.48 #define XYSSL\_ERR\_X509\_CERT\_INVALID\_DATE -0x0100**

Definition at line 22 of file x509.h.

Referenced by x509\_get\_dates().

**5.113.2.49 #define XYSSL\_ERR\_X509\_CERT\_INVALID\_EXTENSIONS -0x0160**

Definition at line 25 of file x509.h.

Referenced by x509\_get\_ext().

**5.113.2.50 #define XYSSL\_ERR\_X509\_CERT\_INVALID\_FORMAT -0x0060**

Definition at line 17 of file x509.h.

Referenced by x509parse\_crt().

**5.113.2.51 #define XYSSL\_ERR\_X509\_CERT\_INVALID\_NAME -0x00E0**

Definition at line 21 of file x509.h.

Referenced by x509\_get\_name().

**5.113.2.52 #define XYSSL\_ERR\_X509\_CERT\_INVALID\_PEM -0x0040**

Definition at line 16 of file x509.h.

Referenced by x509parse\_crt().

**5.113.2.53 #define XYSSL\_ERR\_X509\_CERT\_INVALID\_PUBKEY -0x0120**

Definition at line 23 of file x509.h.

Referenced by x509\_get\_pubkey().

**5.113.2.54 #define XYSSL\_ERR\_X509\_CERT\_INVALID\_SERIAL -0x00A0**

Definition at line 19 of file x509.h.

Referenced by x509\_get\_serial().

**5.113.2.55 #define XYSSL\_ERR\_X509\_CERT\_INVALID\_SIGNATURE -0x0140**

Definition at line 24 of file x509.h.

Referenced by x509\_get\_sig().

**5.113.2.56 #define XYSSL\_ERR\_X509\_CERT\_INVALID\_VERSION -0x0080**

Definition at line 18 of file x509.h.

Referenced by x509\_get\_version().

**5.113.2.57 #define XYSSL\_ERR\_X509\_CERT\_SIG\_MISMATCH -0x01E0**

Definition at line 29 of file x509.h.

Referenced by x509parse\_crt().

**5.113.2.58 #define XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_PK\_ALG -0x01C0**

Definition at line 28 of file x509.h.

Referenced by x509\_get\_pubkey().

**5.113.2.59 #define XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_SIG\_ALG -0x01A0**

Definition at line 27 of file x509.h.

Referenced by x509parse\_crt().

**5.113.2.60 #define XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_VERSION -0x0180**

Definition at line 26 of file x509.h.

Referenced by x509parse\_crt().

**5.113.2.61 #define XYSSL\_ERR\_X509\_CERT\_VERIFY\_FAILED -0x0200**

Definition at line 30 of file x509.h.

Referenced by x509parse\_verify().

**5.113.2.62 #define XYSSL\_ERR\_X509\_FEATURE\_UNAVAILABLE -0x0020**

Definition at line 15 of file x509.h.

Referenced by x509parse\_key().

**5.113.2.63 #define XYSSL\_ERR\_X509\_KEY\_INVALID\_ENC\_IV -0x0280**

Definition at line 34 of file x509.h.

Referenced by x509\_get\_iv(), and x509parse\_key().

**5.113.2.64 #define XYSSL\_ERR\_X509\_KEY\_INVALID\_FORMAT -0x0260**

Definition at line 33 of file x509.h.

Referenced by x509parse\_key().

**5.113.2.65 #define XYSSL\_ERR\_X509\_KEY\_INVALID\_PEM -0x0220**

Definition at line 31 of file x509.h.

Referenced by x509parse\_key().

**5.113.2.66 #define XYSSL\_ERR\_X509\_KEY\_INVALID\_VERSION -0x0240**

Definition at line 32 of file x509.h.

Referenced by x509parse\_key().

**5.113.2.67 #define XYSSL\_ERR\_X509\_KEY\_PASSWORD\_MISMATCH -0x02E0**

Definition at line 37 of file x509.h.

Referenced by x509parse\_key().

**5.113.2.68 #define XYSSL\_ERR\_X509\_KEY\_PASSWORD\_REQUIRED -0x02C0**

Definition at line 36 of file x509.h.

Referenced by x509parse\_key().

**5.113.2.69 #define XYSSL\_ERR\_X509\_KEY\_UNKNOWN\_ENC\_ALG -0x02A0**

Definition at line 35 of file x509.h.

Referenced by x509parse\_key().

**5.113.2.70 #define XYSSL\_ERR\_X509\_POINT\_ERROR -0x0300**

Definition at line 38 of file x509.h.

**5.113.2.71 #define XYSSL\_ERR\_X509\_VALUE\_TO\_LENGTH -0x0320**

Definition at line 39 of file x509.h.

### 5.113.3 Typedef Documentation

**5.113.3.1** `typedef struct _x509_buf x509_buf`

**5.113.3.2** `typedef struct _x509_cert x509_cert`

**5.113.3.3** `typedef struct _x509_name x509_name`

**5.113.3.4** `typedef struct _x509_node x509_node`

**5.113.3.5** `typedef struct _x509_raw x509_raw`

**5.113.3.6** `typedef struct _x509_time x509_time`

### 5.113.4 Function Documentation

**5.113.4.1** `void x509_free (x509_cert * crt)`

Unallocate all certificate data.

Definition at line 1613 of file x509parse.c.

References `_x509_cert::issuer`, `_x509_buf::len`, `_x509_cert::next`, `_x509_name::next`, `_x509_buf::p`, `_x509_cert::raw`, `_x509_cert::rsa`, `rsa_free()`, and `_x509_cert::subject`.

Referenced by `main()`, `ssl_free()`, `ssl_test()`, `x509_self_test()`, and `x509parse_cert()`.

**5.113.4.2** `int x509_self_test (int verbose)`

Checkup routine.

#### Returns

0 if successful, or 1 if the test failed

Definition at line 1675 of file x509parse.c.

References `rsa_free()`, `test_ca_cert`, `test_ca_key`, `test_ca_pwd`, `test_cli_cert`, `x509_free()`, `x509parse_cert()`, `x509parse_key()`, and `x509parse_verify()`.

Referenced by `main()`.

**5.113.4.3** `char* x509parse_cert_info (char * prefix, x509_cert * crt)`

Returns an informational string about the certificate.

Definition at line 1399 of file x509parse.c.

References `buf`, `_x509_time::day`, `_x509_time::hour`, `_x509_cert::issuer`, `_x509_buf::len`, `_x509_time::min`, `_x509_time::mon`, `mpi::n`, `rsa_context::N`, `_x509_buf::p`, `_x509_cert::rsa`, `RSA_MD2`, `RSA_MD4`, `RSA_MD5`, `RSA_SHA1`, `_x509_time::sec`, `_x509_cert::serial`, `_x509_cert::sig_oid1`, `_x509_cert::subject`, `_x509_cert::valid_from`, `_x509_cert::valid_to`, `_x509_cert::version`, `x509parse_dn_gets()`, and `_x509_time::year`.

Referenced by `debug_print_cert()`, and `main()`.

#### 5.113.4.4 `int x509parse_crt(x509_cert *crt, unsigned char *buf, int buflen)`

Parse one or more certificates and add them to the chained list.

##### Parameters

*chain* points to the start of the chain  
*buf* buffer holding the certificate data  
*buflen* size of the buffer

##### Returns

0 if successful, or a specific X509 error code

Definition at line 647 of file x509parse.c.

References ASN1\_CONSTRUCTED, asn1\_get\_tag(), ASN1\_SEQUENCE, base64\_decode(), \_x509\_cert::ca\_istrue, rsa\_context::E, \_x509\_cert::issuer, \_x509\_cert::issuer\_id, \_x509\_cert::issuer\_raw, rsa\_context::len, \_x509\_buf::len, \_x509\_cert::max\_pathlen, mpi\_size(), rsa\_context::N, \_x509\_cert::next, OID\_PKCS1, \_x509\_buf::p, \_x509\_cert::pk\_oid, \_x509\_cert::raw, \_x509\_cert::rsa, rsa\_check\_pubkey(), \_x509\_cert::serial, \_x509\_cert::sig, \_x509\_cert::sig\_oid1, \_x509\_cert::sig\_oid2, \_x509\_cert::subject, \_x509\_cert::subject\_id, \_x509\_cert::subject\_raw, \_x509\_cert::tbs, \_x509\_cert::v3\_ext, \_x509\_cert::valid\_from, \_x509\_cert::valid\_to, \_x509\_cert::version, x509\_free(), x509\_get\_alg(), x509\_get\_dates(), x509\_get\_ext(), x509\_get\_name(), x509\_get\_pubkey(), x509\_get\_serial(), x509\_get\_sig(), x509\_get\_uid(), x509\_get\_version(), x509parse\_crt(), XYSSL\_ERR\_ASN1\_LENGTH\_MISMATCH, XYSSL\_ERR\_BASE64\_INVALID\_CHARACTER, XYSSL\_ERR\_X509\_CERT\_INVALID\_FORMAT, XYSSL\_ERR\_X509\_CERT\_INVALID\_PEM, XYSSL\_ERR\_X509\_CERT\_SIG\_MISMATCH, XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_SIG\_ALG, and XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_VERSION.

Referenced by main(), ssl\_parse\_certificate(), ssl\_test(), x509\_self\_test(), x509parse\_crt(), and x509parse\_crtfile().

#### 5.113.4.5 `int x509parse_crtfile(x509_cert *crt, char *path)`

Load one or more certificates and add them to the chained list.

##### Parameters

*chain* points to the start of the chain  
*path* filename to read the certificates from

##### Returns

0 if successful, or a specific X509 error code

Definition at line 979 of file x509parse.c.

References buf, f, and x509parse\_crt().

#### 5.113.4.6 `int x509parse_dn_gets(char *buf, char *end, x509_name *dn)`

Store the certificate DN in printable form into buf; no more than (end - buf) characters will be written.

Definition at line 1316 of file x509parse.c.



References `_x509_buf::len`, `_x509_name::next`, `_x509_name::oid`, `OID_PKCS9`, `OID_X520`, `_x509_buf::p`, `PKCS9_EMAIL`, `_x509_name::val`, `X520_COMMON_NAME`, `X520_COUNTRY`, `X520_LOCALITY`, `X520_ORG_UNIT`, `X520_ORGANIZATION`, and `X520_STATE`.

Referenced by `x509parse_cert_info()`.

#### 5.113.4.7 `int x509parse_expired(x509_cert * crt)`

Return 0 if the certificate is still valid, or `BADCERT_EXPIRED`.

Definition at line 1458 of file `x509parse.c`.

References `BADCERT_EXPIRED`, `_x509_time::day`, `_x509_time::mon`, `_x509_cert::valid_to`, and `_x509_time::year`.

Referenced by `x509parse_verify()`.

#### 5.113.4.8 `int x509parse_key(rsa_context * rsa, unsigned char * buf, int buflen, unsigned char * pwd, int pwrlen)`

Parse a private RSA key.

##### Parameters

*rsa* RSA context to be initialized  
*buf* input buffer  
*buflen* size of the buffer  
*pwd* password for decryption (optional)  
*pwrlen* size of the password

##### Returns

0 if successful, or a specific X509 error code

Definition at line 1082 of file `x509parse.c`.

References `ASN1_CONSTRUCTED`, `asn1_get_int()`, `asn1_get_mpi()`, `asn1_get_tag()`, `ASN1_SEQUENCE`, `base64_decode()`, `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `rsa_context::len`, `mpi_size()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_check_privkey()`, `rsa_free()`, `rsa_context::ver`, `x509_des3_decrypt()`, `x509_get_iv()`, `XYSSL_ERR_ASN1_LENGTH_MISMATCH`, `XYSSL_ERR_BASE64_INVALID_CHARACTER`, `XYSSL_ERR_X509_FEATURE_UNAVAILABLE`, `XYSSL_ERR_X509_KEY_INVALID_ENC_IV`, `XYSSL_ERR_X509_KEY_INVALID_FORMAT`, `XYSSL_ERR_X509_KEY_INVALID_PEM`, `XYSSL_ERR_X509_KEY_INVALID_VERSION`, `XYSSL_ERR_X509_KEY_PASSWORD_MISMATCH`, `XYSSL_ERR_X509_KEY_PASSWORD_REQUIRED`, and `XYSSL_ERR_X509_KEY_UNKNOWN_ENC_ALG`.

Referenced by `main()`, `ssl_test()`, `x509_self_test()`, and `x509parse_keyfile()`.

#### 5.113.4.9 `int x509parse_keyfile(rsa_context * rsa, char * path, char * password)`

Load and parse a private RSA key.

##### Parameters

*rsa* RSA context to be initialized

*path* filename to read the private key from  
*pwd* password to decrypt the file (can be NULL)

### Returns

0 if successful, or a specific X509 error code

Definition at line 1269 of file x509parse.c.

References buf, f, and x509parse\_key().

#### 5.113.4.10 int x509parse\_verify(x509\_cert \*crt, x509\_cert \*trust\_ca, char \*cn, int \*flags)

Verify the certificate signature.

### Parameters

*crt* a certificate to be verified  
*trust\_ca* the trusted CA chain  
*cn* expected Common Name (can be set to NULL if the CN must not be verified)  
*flags* result of the verification

### Returns

0 if successful or XYSSL\_ERR\_X509\_SIG\_VERIFY\_FAILED, in which case \*flags will have one or more of the following values set: BADCERT\_EXPIRED -- BADCERT\_REVOKED -- BADCERT\_CN\_MISMATCH -- BADCERT\_NOT\_TRUSTED

### Note

TODO: add two arguments, depth and crl

Definition at line 1503 of file x509parse.c.

References BADCERT\_CN\_MISMATCH, BADCERT\_NOT\_TRUSTED, \_x509\_cert::ca\_istrue, cur, \_x509\_cert::issuer\_raw, \_x509\_buf::len, \_x509\_cert::max\_pathlen, \_x509\_cert::next, \_x509\_name::next, \_x509\_name::oid, OID\_CN, \_x509\_buf::p, \_x509\_cert::rsa, rsa\_pkcs1\_verify(), RSA\_PUBLIC, \_x509\_cert::sig, \_x509\_cert::sig\_oid1, \_x509\_cert::subject, \_x509\_cert::subject\_raw, \_x509\_cert::tbs, \_x509\_name::val, \_x509\_cert::version, x509\_hash(), x509parse\_expired(), and XYSSL\_ERR\_X509\_CERT\_VERIFY\_FAILED.

Referenced by ssl\_parse\_certificate(), and x509\_self\_test().

## 5.114 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/aes.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/aes.h"
#include "xyssl/padlock.h"
#include <string.h>
#include <stdio.h>
```

### Defines

- #define [GET\\_ULONG\\_LE](#)(n, b, i)
- #define [PUT\\_ULONG\\_LE](#)(n, b, i)
- #define [ROTL8](#)(x) ( ( x << 8 ) & 0xFFFFFFFF ) | ( x >> 24 )
- #define [XTIME](#)(x) ( ( x << 1 ) ^ ( ( x & 0x80 ) ? 0x1B : 0x00 ) )
- #define [MUL](#)(x, y) ( ( x && y ) ? pow[(log[x]+log[y]) % 255] : 0 )
- #define [AES\\_FROUND](#)(X0, X1, X2, X3, Y0, Y1, Y2, Y3)
- #define [AES\\_RROUND](#)(X0, X1, X2, X3, Y0, Y1, Y2, Y3)

### Functions

- static void [aes\\_gen\\_tables](#) (void)
- void [aes\\_setkey\\_enc](#) ([aes\\_context](#) \*ctx, unsigned char \*key, [int](#) keysize)  
*AES key schedule (encryption).*
- void [aes\\_setkey\\_dec](#) ([aes\\_context](#) \*ctx, unsigned char \*key, [int](#) keysize)  
*AES key schedule (decryption).*
- void [aes\\_crypt\\_ecb](#) ([aes\\_context](#) \*ctx, [int](#) mode, unsigned char input[16], unsigned char output[16])  
*AES-ECB block encryption/decryption.*
- void [aes\\_crypt\\_cbc](#) ([aes\\_context](#) \*ctx, [int](#) mode, [int](#) length, unsigned char iv[16], unsigned char \*input, unsigned char \*output)  
*AES-CBC buffer encryption/decryption.*
- void [aes\\_crypt\\_cfb](#) ([aes\\_context](#) \*ctx, [int](#) mode, [int](#) length, [int](#) \*iv\_off, unsigned char iv[16], unsigned char \*input, unsigned char \*output)  
*AES-CFB buffer encryption/decryption.*
- [int](#) [aes\\_self\\_test](#) ([int](#) verbose)  
*Checkup routine.*

## Variables

- static unsigned char [FSb](#) [256]
- static unsigned long [FT0](#) [256]
- static unsigned long [FT1](#) [256]
- static unsigned long [FT2](#) [256]
- static unsigned long [FT3](#) [256]
- static unsigned char [RSb](#) [256]
- static unsigned long [RT0](#) [256]
- static unsigned long [RT1](#) [256]
- static unsigned long [RT2](#) [256]
- static unsigned long [RT3](#) [256]
- static unsigned long [RCON](#) [10]
- static `int` [aes\\_init\\_done](#) = 0
- static const unsigned char [aes\\_test\\_ecb\\_dec](#) [3][16]
- static const unsigned char [aes\\_test\\_ecb\\_enc](#) [3][16]
- static const unsigned char [aes\\_test\\_cbc\\_dec](#) [3][16]
- static const unsigned char [aes\\_test\\_cbc\\_enc](#) [3][16]
- static const unsigned char [aes\\_test\\_cfb\\_dec](#) [3][16]
- static const unsigned char [aes\\_test\\_cfb\\_enc](#) [3][16]

### 5.114.1 Define Documentation

#### 5.114.1.1 #define AES\_FROUND(X0, X1, X2, X3, Y0, Y1, Y2, Y3)

**Value:**

```
{
    X0 = *RK++ ^ FT0[ ( Y0          ) & 0xFF ] ^ \
              FT1[ ( Y1 >>  8 ) & 0xFF ] ^ \
              FT2[ ( Y2 >> 16 ) & 0xFF ] ^ \
              FT3[ ( Y3 >> 24 ) & 0xFF ]; \
    X1 = *RK++ ^ FT0[ ( Y1          ) & 0xFF ] ^ \
              FT1[ ( Y2 >>  8 ) & 0xFF ] ^ \
              FT2[ ( Y3 >> 16 ) & 0xFF ] ^ \
              FT3[ ( Y0 >> 24 ) & 0xFF ]; \
    X2 = *RK++ ^ FT0[ ( Y2          ) & 0xFF ] ^ \
              FT1[ ( Y3 >>  8 ) & 0xFF ] ^ \
              FT2[ ( Y0 >> 16 ) & 0xFF ] ^ \
              FT3[ ( Y1 >> 24 ) & 0xFF ]; \
    X3 = *RK++ ^ FT0[ ( Y3          ) & 0xFF ] ^ \
              FT1[ ( Y0 >>  8 ) & 0xFF ] ^ \
              FT2[ ( Y1 >> 16 ) & 0xFF ] ^ \
              FT3[ ( Y2 >> 24 ) & 0xFF ]; \
}
```

Definition at line 590 of file `aes.c`.

Referenced by `aes_crypt_ecb()`.

**5.114.1.2 #define AES\_RROUND(X0, X1, X2, X3, Y0, Y1, Y2, Y3)****Value:**

```

{
    X0 = *RK++ ^ RT0[ ( Y0          ) & 0xFF ] ^ \
              RT1[ ( Y3 >> 8 ) & 0xFF ] ^ \
              RT2[ ( Y2 >> 16 ) & 0xFF ] ^ \
              RT3[ ( Y1 >> 24 ) & 0xFF ]; \
    X1 = *RK++ ^ RT0[ ( Y1          ) & 0xFF ] ^ \
              RT1[ ( Y0 >> 8 ) & 0xFF ] ^ \
              RT2[ ( Y3 >> 16 ) & 0xFF ] ^ \
              RT3[ ( Y2 >> 24 ) & 0xFF ]; \
    X2 = *RK++ ^ RT0[ ( Y2          ) & 0xFF ] ^ \
              RT1[ ( Y1 >> 8 ) & 0xFF ] ^ \
              RT2[ ( Y0 >> 16 ) & 0xFF ] ^ \
              RT3[ ( Y3 >> 24 ) & 0xFF ]; \
    X3 = *RK++ ^ RT0[ ( Y3          ) & 0xFF ] ^ \
              RT1[ ( Y2 >> 8 ) & 0xFF ] ^ \
              RT2[ ( Y1 >> 16 ) & 0xFF ] ^ \
              RT3[ ( Y0 >> 24 ) & 0xFF ]; \
}

```

Definition at line 613 of file aes.c.

Referenced by aes\_crypt\_ecb().

**5.114.1.3 #define GET\_ULONG\_LE(n, b, i)****Value:**

```

{
    (n) = ( (unsigned long) (b)[(i)          ] ) \
    | ( (unsigned long) (b)[(i) + 1] << 8 ) \
    | ( (unsigned long) (b)[(i) + 2] << 16 ) \
    | ( (unsigned long) (b)[(i) + 3] << 24 ); \
}

```

Definition at line 40 of file aes.c.

Referenced by aes\_crypt\_ecb(), aes\_setkey\_enc(), and md5\_process().

**5.114.1.4 #define MUL(x, y) ((x && y) ? pow[(log[x]+log[y]) % 255] : 0)**

Definition at line 354 of file aes.c.

Referenced by aes\_gen\_tables().

**5.114.1.5 #define PUT\_ULONG\_LE(n, b, i)****Value:**

```

{
    (b)[(i)          ] = (unsigned char) ( (n)          ); \
    (b)[(i) + 1] = (unsigned char) ( (n) >> 8 ); \
    (b)[(i) + 2] = (unsigned char) ( (n) >> 16 ); \
    (b)[(i) + 3] = (unsigned char) ( (n) >> 24 ); \
}

```

Definition at line 50 of file aes.c.

Referenced by aes\_crypt\_ecb(), and md5\_finish().

#### 5.114.1.6 **#define ROTL8(x) ( ( x << 8 ) & 0xFFFFFFFF ) | ( x >> 24 )**

Definition at line 352 of file aes.c.

Referenced by aes\_gen\_tables().

#### 5.114.1.7 **#define XTIME(x) ( ( x << 1 ) ^ ( ( x & 0x80 ) ? 0x1B : 0x00 ) )**

Definition at line 353 of file aes.c.

Referenced by aes\_gen\_tables().

### 5.114.2 Function Documentation

#### 5.114.2.1 **void aes\_crypt\_cbc (aes\_context \* ctx, int mode, int length, unsigned char iv[16], unsigned char \* input, unsigned char \* output)**

AES-CBC buffer encryption/decryption.

##### Parameters

*ctx* AES context  
*mode* AES\_ENCRYPT or AES\_DECRYPT  
*length* length of the input data  
*iv* initialization vector (updated after use)  
*input* buffer holding the input data  
*output* buffer holding the output data

Definition at line 732 of file aes.c.

References aes\_crypt\_ecb(), and AES\_DECRYPT.

Referenced by aes\_self\_test(), main(), ssl\_decrypt\_buf(), and ssl\_encrypt\_buf().

#### 5.114.2.2 **void aes\_crypt\_cfb (aes\_context \* ctx, int mode, int length, int \* iv\_off, unsigned char iv[16], unsigned char \* input, unsigned char \* output)**

AES-CFB buffer encryption/decryption.

##### Parameters

*ctx* AES context  
*mode* AES\_ENCRYPT or AES\_DECRYPT  
*length* length of the input data  
*iv\_off* offset in IV (updated after use)  
*iv* initialization vector (updated after use)  
*input* buffer holding the input data

*output* buffer holding the output data

Definition at line 787 of file aes.c.

References `aes_crypt_ecb()`, `AES_DECRYPT`, and `AES_ENCRYPT`.

Referenced by `aes_self_test()`.

#### 5.114.2.3 void aes\_crypt\_ecb (aes\_context \* ctx, int mode, unsigned char input[16], unsigned char output[16])

AES-ECB block encryption/decryption.

##### Parameters

*ctx* AES context

*mode* `AES_ENCRYPT` or `AES_DECRYPT`

*input* 16-byte input block

*output* 16-byte output block

Definition at line 639 of file aes.c.

References `AES_DECRYPT`, `AES_FROUND`, `AES_RROUND`, `FSb`, `GET_ULONG_LE`, `aes_context::nr`, `PUT_ULONG_LE`, `aes_context::rk`, and `RSb`.

Referenced by `aes_crypt_cbc()`, `aes_crypt_cfb()`, `aes_en_de()`, `aes_self_test()`, and `main()`.

#### 5.114.2.4 static void aes\_gen\_tables (void) [static]

Definition at line 358 of file aes.c.

References `FSb`, `FT0`, `FT1`, `FT2`, `FT3`, `MUL`, `RCON`, `ROTL8`, `RSb`, `RT0`, `RT1`, `RT2`, `RT3`, and `XTIME`.

Referenced by `aes_setkey_enc()`.

#### 5.114.2.5 int aes\_self\_test (int verbose)

Checkup routine.

##### Returns

0 if successful, or 1 if the test failed

Definition at line 902 of file aes.c.

References `aes_crypt_cbc()`, `aes_crypt_cfb()`, `aes_crypt_ecb()`, `AES_DECRYPT`, `aes_setkey_dec()`, `aes_setkey_enc()`, `aes_test_cbc_dec`, `aes_test_cbc_enc`, `aes_test_cfb_dec`, `aes_test_cfb_enc`, `aes_test_ecb_dec`, `aes_test_ecb_enc`, `buf`, and `prv`.

Referenced by `main()`.

#### 5.114.2.6 void aes\_setkey\_dec (aes\_context \* ctx, unsigned char \* key, int keysize)

AES key schedule (decryption).

**Parameters***ctx* AES context to be initialized*key* decryption key*keysize* must be 128, 192 or 256

Definition at line 542 of file aes.c.

References aes\_setkey\_enc(), aes\_context::buf, FSb, aes\_context::nr, aes\_context::rk, RT0, RT1, RT2, and RT3.

Referenced by aes\_en\_de(), aes\_self\_test(), main(), and ssl\_derive\_keys().

**5.114.2.7 void aes\_setkey\_enc (aes\_context \* ctx, unsigned char \* key, int keysize)**

AES key schedule (encryption).

**Parameters***ctx* AES context to be initialized*key* encryption key*keysize* must be 128, 192 or 256

Definition at line 439 of file aes.c.

References aes\_gen\_tables(), aes\_init\_done, aes\_context::buf, FSb, GET\_ULONG\_LE, aes\_context::nr, RCON, and aes\_context::rk.

Referenced by aes\_en\_de(), aes\_self\_test(), aes\_setkey\_dec(), main(), and ssl\_derive\_keys().

**5.114.3 Variable Documentation****5.114.3.1 int aes\_init\_done = 0 [static]**

Definition at line 356 of file aes.c.

Referenced by aes\_setkey\_enc().

**5.114.3.2 const unsigned char aes\_test\_cbc\_dec[3][16] [static]****Initial value:**

```
{
    { 0xFA, 0xCA, 0x37, 0xE0, 0xB0, 0xC8, 0x53, 0x73,
      0xDF, 0x70, 0x6E, 0x73, 0xF7, 0xC9, 0xAF, 0x86 },
    { 0x5D, 0xF6, 0x78, 0xDD, 0x17, 0xBA, 0x4E, 0x75,
      0xB6, 0x17, 0x68, 0xC6, 0xAD, 0xEF, 0x7C, 0x7B },
    { 0x48, 0x04, 0xE1, 0x81, 0x8F, 0xE6, 0x29, 0x75,
      0x19, 0xA3, 0xE8, 0x8C, 0x57, 0x31, 0x04, 0x13 }
}
```

Definition at line 856 of file aes.c.

Referenced by aes\_self\_test().



**5.114.3.3 const unsigned char aes\_test\_cbc\_enc[3][16] [static]****Initial value:**

```
{
    { 0x8A, 0x05, 0xFC, 0x5E, 0x09, 0x5A, 0xF4, 0x84,
      0x8A, 0x08, 0xD3, 0x28, 0xD3, 0x68, 0x8E, 0x3D },
    { 0x7B, 0xD9, 0x66, 0xD5, 0x3A, 0xD8, 0xC1, 0xBB,
      0x85, 0xD2, 0xAD, 0xFA, 0xE8, 0x7B, 0xB1, 0x04 },
    { 0xFE, 0x3C, 0x53, 0x65, 0x3E, 0x2F, 0x45, 0xB5,
      0x6F, 0xCD, 0x88, 0xB2, 0xCC, 0x89, 0x8F, 0xF0 }
}
```

Definition at line 866 of file aes.c.

Referenced by aes\_self\_test().

**5.114.3.4 const unsigned char aes\_test\_cfb\_dec[3][16] [static]****Initial value:**

```
{
    { 0xBA, 0x75, 0x0C, 0xC9, 0x77, 0xF8, 0xD4, 0xE1,
      0x3E, 0x0F, 0xB5, 0x46, 0x2E, 0xA6, 0x33, 0xF6 },
    { 0xDB, 0x40, 0x4A, 0x98, 0x7B, 0xAA, 0xA3, 0xF3,
      0x92, 0x35, 0xAD, 0x58, 0x09, 0x9B, 0xFF, 0x6E },
    { 0xA8, 0x17, 0x41, 0x0E, 0x76, 0x71, 0x60, 0xE5,
      0xFD, 0x37, 0xC5, 0x43, 0xCC, 0xC8, 0xD6, 0xDA }
}
```

Definition at line 879 of file aes.c.

Referenced by aes\_self\_test().

**5.114.3.5 const unsigned char aes\_test\_cfb\_enc[3][16] [static]****Initial value:**

```
{
    { 0x45, 0x62, 0xC5, 0xA1, 0xF9, 0x10, 0x8F, 0xE0,
      0x87, 0x24, 0x25, 0x68, 0xB5, 0x12, 0xF3, 0x8B },
    { 0xB8, 0xD4, 0xD5, 0x09, 0xF5, 0xEE, 0x08, 0x38,
      0x48, 0x9B, 0x9D, 0xAD, 0x11, 0xB4, 0x2E, 0xD2 },
    { 0xE9, 0x10, 0x80, 0xDA, 0xEE, 0x2D, 0x81, 0xD9,
      0x41, 0x78, 0x91, 0xD5, 0x98, 0x78, 0xE1, 0xFA }
}
```

Definition at line 889 of file aes.c.

Referenced by aes\_self\_test().

**5.114.3.6 const unsigned char aes\_test\_ecb\_dec[3][16] [static]****Initial value:**

```
{
    { 0x44, 0x41, 0x6A, 0xC2, 0xD1, 0xF5, 0x3C, 0x58,
      0x33, 0x03, 0x91, 0x7E, 0x6B, 0xE9, 0xEB, 0xE0 },
    { 0x48, 0xE3, 0x1E, 0x9E, 0x25, 0x67, 0x18, 0xF2,
      0x92, 0x29, 0x31, 0x9C, 0x19, 0xF1, 0x5B, 0xA4 },
    { 0x05, 0x8C, 0xCF, 0xFD, 0xBB, 0xCB, 0x38, 0x2D,
      0x1F, 0x6F, 0x56, 0x58, 0x5D, 0x8A, 0x4A, 0xDE }
}
```

Definition at line 836 of file aes.c.

Referenced by aes\_self\_test().

#### 5.114.3.7 `const unsigned char aes_test_ecb_enc[3][16] [static]`

Initial value:

```
{
    { 0xC3, 0x4C, 0x05, 0x2C, 0xC0, 0xDA, 0x8D, 0x73,
      0x45, 0x1A, 0xFE, 0x5F, 0x03, 0xBE, 0x29, 0x7F },
    { 0xF3, 0xF6, 0x75, 0x2A, 0xE8, 0xD7, 0x83, 0x11,
      0x38, 0xF0, 0x41, 0x56, 0x06, 0x31, 0xB1, 0x14 },
    { 0x8B, 0x79, 0xEE, 0xCC, 0x93, 0xA0, 0xEE, 0x5D,
      0xFF, 0x30, 0xB4, 0xEA, 0x21, 0x63, 0x6D, 0xA4 }
}
```

Definition at line 846 of file aes.c.

Referenced by aes\_self\_test().

#### 5.114.3.8 `unsigned char FSb[256] [static]`

Definition at line 329 of file aes.c.

Referenced by aes\_crypt\_ecb(), aes\_gen\_tables(), aes\_setkey\_dec(), and aes\_setkey\_enc().

#### 5.114.3.9 `unsigned long FT0[256] [static]`

Definition at line 330 of file aes.c.

Referenced by aes\_gen\_tables().

#### 5.114.3.10 `unsigned long FT1[256] [static]`

Definition at line 331 of file aes.c.

Referenced by aes\_gen\_tables().

#### 5.114.3.11 `unsigned long FT2[256] [static]`

Definition at line 332 of file aes.c.

Referenced by aes\_gen\_tables().

**5.114.3.12 unsigned long FT3[256] [static]**

Definition at line 333 of file aes.c.

Referenced by aes\_gen\_tables().

**5.114.3.13 unsigned long RCON[10] [static]**

Definition at line 347 of file aes.c.

Referenced by aes\_gen\_tables(), and aes\_setkey\_enc().

**5.114.3.14 unsigned char RSb[256] [static]**

Definition at line 338 of file aes.c.

Referenced by aes\_crypt\_ecb(), and aes\_gen\_tables().

**5.114.3.15 unsigned long RT0[256] [static]**

Definition at line 339 of file aes.c.

Referenced by aes\_gen\_tables(), and aes\_setkey\_dec().

**5.114.3.16 unsigned long RT1[256] [static]**

Definition at line 340 of file aes.c.

Referenced by aes\_gen\_tables(), and aes\_setkey\_dec().

**5.114.3.17 unsigned long RT2[256] [static]**

Definition at line 341 of file aes.c.

Referenced by aes\_gen\_tables(), and aes\_setkey\_dec().

**5.114.3.18 unsigned long RT3[256] [static]**

Definition at line 342 of file aes.c.

Referenced by aes\_gen\_tables(), and aes\_setkey\_dec().

## 5.115 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/arc4.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/arc4.h"
#include <string.h>
#include <stdio.h>
```

### Functions

- void [arc4\\_setup](#) ([arc4\\_context](#) \*ctx, unsigned char \*key, [int](#) keylen)  
*ARC4 key schedule.*
- void [arc4\\_crypt](#) ([arc4\\_context](#) \*ctx, unsigned char \*buf, [int](#) buflen)  
*ARC4 cipher function.*
- [int](#) [arc4\\_self\\_test](#) ([int](#) verbose)

### Variables

- static const unsigned char [arc4\\_test\\_key](#) [3][8]
- static const unsigned char [arc4\\_test\\_pt](#) [3][8]
- static const unsigned char [arc4\\_test\\_ct](#) [3][8]

### 5.115.1 Function Documentation

#### 5.115.1.1 void [arc4\\_crypt](#) ([arc4\\_context](#) \* ctx, unsigned char \* buf, [int](#) buflen)

ARC4 cipher function.

#### Parameters

*ctx* ARC4 context

*buf* buffer to be processed

*buflen* amount of data in buf

Definition at line 63 of file arc4.c.

References [arc4\\_context::m](#), [arc4\\_context::x](#), and [arc4\\_context::y](#).

Referenced by [arc4\\_self\\_test\(\)](#), [main\(\)](#), [ssl\\_decrypt\\_buf\(\)](#), and [ssl\\_encrypt\\_buf\(\)](#).

#### 5.115.1.2 [int](#) [arc4\\_self\\_test](#) ([int](#) verbose)

Definition at line 122 of file arc4.c.

References [arc4\\_crypt\(\)](#), [arc4\\_setup\(\)](#), [arc4\\_test\\_ct](#), [arc4\\_test\\_key](#), [arc4\\_test\\_pt](#), and [buf](#).

Referenced by [main\(\)](#).

**5.115.1.3 void arc4\_setup (arc4\_context \* *ctx*, unsigned char \* *key*, int *keylen*)**

ARC4 key schedule.

**Parameters**

*ctx* ARC4 context to be initialized

*key* the secret key

*keylen* length of the key

Definition at line 35 of file arc4.c.

References arc4\_context::m, arc4\_context::x, and arc4\_context::y.

Referenced by arc4\_self\_test(), main(), and ssl\_derive\_keys().

**5.115.2 Variable Documentation****5.115.2.1 const unsigned char arc4\_test\_ct[3][8] [static]****Initial value:**

```
{
    { 0x75, 0xB7, 0x87, 0x80, 0x99, 0xE0, 0xC5, 0x96 },
    { 0x74, 0x94, 0xC2, 0xE7, 0x10, 0x4B, 0x08, 0x79 },
    { 0xDE, 0x18, 0x89, 0x41, 0xA3, 0x37, 0x5D, 0x3A }
}
```

Definition at line 112 of file arc4.c.

Referenced by arc4\_self\_test().

**5.115.2.2 const unsigned char arc4\_test\_key[3][8] [static]****Initial value:**

```
{
    { 0x01, 0x23, 0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF },
    { 0x01, 0x23, 0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF },
    { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 }
}
```

Definition at line 98 of file arc4.c.

Referenced by arc4\_self\_test().

**5.115.2.3 const unsigned char arc4\_test\_pt[3][8] [static]****Initial value:**

```
{
    { 0x01, 0x23, 0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF },
    { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 },
    { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 }
}
```

Definition at line 105 of file arc4.c.

Referenced by arc4\_self\_test().

## 5.116 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/base64.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/base64.h"
#include <string.h>
#include <stdio.h>
```

### Functions

- `int base64_encode` (unsigned char \*dst, int \*dlen, unsigned char \*src, int slen)  
*Encode a buffer into base64 format.*
- `int base64_decode` (unsigned char \*dst, int \*dlen, unsigned char \*src, int slen)  
*Decode a base64-formatted buffer.*
- `int base64_self_test` (int verbose)  
*Checkup routine.*

### Variables

- static const unsigned char `base64_enc_map` [64]
- static const unsigned char `base64_dec_map` [128]
- static const unsigned char `base64_test_dec` [64]
- static const unsigned char `base64_test_enc` []

### 5.116.1 Function Documentation

#### 5.116.1.1 int base64\_decode (unsigned char \*dst, int \*dlen, unsigned char \*src, int slen)

Decode a base64-formatted buffer.

##### Parameters

*dst* destination buffer  
*dlen* size of the buffer  
*src* source buffer  
*slen* amount of data to be decoded

##### Returns

0 if successful, XYSSL\_ERR\_BASE64\_BUFFER\_TOO\_SMALL, or XYSSL\_ERR\_BASE64\_INVALID\_DATA if the input data is not correct. \*dlen is always updated to reflect the amount of data that has (or would have) been written.

##### Note

Call this function with \*dlen = 0 to obtain the required buffer size in \*dlen

Definition at line 121 of file base64.c.

References base64\_dec\_map, XYSSL\_ERR\_BASE64\_BUFFER\_TOO\_SMALL, and XYSSL\_ERR\_BASE64\_INVALID\_CHARACTER.

Referenced by base64\_self\_test(), x509parse\_crt(), and x509parse\_key().

#### 5.116.1.2 int base64\_encode (unsigned char \*dst, int \*dlen, unsigned char \*src, int slen)

Encode a buffer into base64 format.

##### Parameters

*dst* destination buffer  
*dlen* size of the buffer  
*src* source buffer  
*slen* amount of data to be encoded

##### Returns

0 if successful, or XYSSL\_ERR\_BASE64\_BUFFER\_TOO\_SMALL. \*dlen is always updated to reflect the amount of data that has (or would have) been written.

##### Note

Call this function with \*dlen = 0 to obtain the required buffer size in \*dlen

Definition at line 58 of file base64.c.

References base64\_enc\_map, and XYSSL\_ERR\_BASE64\_BUFFER\_TOO\_SMALL.

Referenced by base64\_self\_test().

#### 5.116.1.3 int base64\_self\_test (int verbose)

Checkup routine.

##### Returns

0 if successful, or 1 if the test failed

Definition at line 206 of file base64.c.

References base64\_decode(), base64\_encode(), base64\_test\_dec, and base64\_test\_enc.

Referenced by main().

## 5.116.2 Variable Documentation

#### 5.116.2.1 const unsigned char base64\_dec\_map[128] [static]

##### Initial value:

```
{
    127, 127, 127, 127, 127, 127, 127, 127, 127, 127, 127,
```



```

127, 127, 127, 127, 127, 127, 127, 127, 127, 127,
127, 127, 127, 127, 127, 127, 127, 127, 127, 127,
127, 127, 127, 127, 127, 127, 127, 127, 127, 127,
127, 127, 127, 62, 127, 127, 127, 63, 52, 53,
54, 55, 56, 57, 58, 59, 60, 61, 127, 127,
127, 64, 127, 127, 127, 0, 1, 2, 3, 4,
5, 6, 7, 8, 9, 10, 11, 12, 13, 14,
15, 16, 17, 18, 19, 20, 21, 22, 23, 24,
25, 127, 127, 127, 127, 127, 127, 26, 27, 28,
29, 30, 31, 32, 33, 34, 35, 36, 37, 38,
39, 40, 41, 42, 43, 44, 45, 46, 47, 48,
49, 50, 51, 127, 127, 127, 127, 127, 127
}

```

Definition at line 38 of file base64.c.

Referenced by base64\_decode().

### 5.116.2.2 const unsigned char base64\_enc\_map[64] [static]

Initial value:

```

{
    'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J',
    'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T',
    'U', 'V', 'W', 'X', 'Y', 'Z', 'a', 'b', 'c', 'd',
    'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n',
    'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x',
    'y', 'z', '0', '1', '2', '3', '4', '5', '6', '7',
    '8', '9', '+', '/'
}

```

Definition at line 27 of file base64.c.

Referenced by base64\_encode().

### 5.116.2.3 const unsigned char base64\_test\_dec[64] [static]

Initial value:

```

{
    0x24, 0x48, 0x6E, 0x56, 0x87, 0x62, 0x5A, 0xBD,
    0xBF, 0x17, 0xD9, 0xA2, 0xC4, 0x17, 0x1A, 0x01,
    0x94, 0xED, 0x8F, 0x1E, 0x11, 0xB3, 0xD7, 0x09,
    0x0C, 0xB6, 0xE9, 0x10, 0x6F, 0x22, 0xEE, 0x13,
    0xCA, 0xB3, 0x07, 0x05, 0x76, 0xC9, 0xFA, 0x31,
    0x6C, 0x08, 0x34, 0xFF, 0x8D, 0xC2, 0x6C, 0x38,
    0x00, 0x43, 0xE9, 0x54, 0x97, 0xAF, 0x50, 0x4B,
    0xD1, 0x41, 0xBA, 0x95, 0x31, 0x5A, 0x0B, 0x97
}

```

Definition at line 187 of file base64.c.

Referenced by base64\_self\_test().

### 5.116.2.4 const unsigned char base64\_test\_enc[] [static]

Initial value:

```
"JEhuVodiWr2/F9mixBcaAZTtjx4Rs9cJDLbpEG8i7hPK"  
"swcFdsn6MWwINP+Nwmw4AEPpVJevUEvRQbqVMVoLlw=="
```

Definition at line 199 of file base64.c.

Referenced by base64\_self\_test().

## 5.117 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/bignum.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/bignum.h"
#include "xyssl/bn_mul.h"
#include <string.h>
#include <stdlib.h>
#include <stdarg.h>
```

### Defines

- `#define ciL ((int) sizeof(t_int))`
- `#define biL (ciL << 3)`
- `#define biH (ciL << 2)`
- `#define BITS_TO_LIMBS(i) (((i) + biL - 1) / biL)`
- `#define CHARS_TO_LIMBS(i) (((i) + ciL - 1) / ciL)`

### Functions

- `void mpi_init (mpi *X,...)`  
*Initialize one or more mpi.*
- `void mpi_free (mpi *X,...)`  
*Unallocate one or more mpi.*
- `int mpi_grow (mpi *X, int nlimbs)`  
*Enlarge to the specified number of limbs.*
- `int mpi_copy (mpi *X, mpi *Y)`  
*Copy the contents of Y into X.*
- `void mpi_swap (mpi *X, mpi *Y)`  
*Swap the contents of X and Y.*
- `int mpi_lset (mpi *X, int z)`  
*Set value from integer.*
- `int mpi_lsb (mpi *X)`  
*Return the number of least significant bits.*
- `int mpi_msb (mpi *X)`  
*Return the number of most significant bits.*
- `int mpi_size (mpi *X)`  
*Return the total size in bytes.*

- static `int mpi_get_digit (t_int *d, int radix, char c)`
- `int mpi_read_string (mpi *X, int radix, char *s)`  
*Import from an ASCII string.*
- static `int mpi_write_hlp (mpi *X, int radix, char **p)`
- `int mpi_write_string (mpi *X, int radix, char *s, int *slen)`  
*Export into an ASCII string.*
- `int mpi_read_file (mpi *X, int radix, FILE *fin)`  
*Read X from an opened file.*
- `int mpi_read_mystring (mpi *X, int radix, char *s)`
- `int mpi_write_file (char *p, mpi *X, int radix, FILE *fout)`  
*Write X into an opened file, or stdout.*
- `int mpi_read_binary (mpi *X, unsigned char *buf, int buflen)`  
*Import X from unsigned binary data, big endian.*
- `int mpi_write_binary (mpi *X, unsigned char *buf, int buflen)`  
*Export X into unsigned binary data, big endian.*
- `int mpi_shift_l (mpi *X, int count)`  
*Left-shift:  $X \ll = \text{count}$ .*
- `int mpi_shift_r (mpi *X, int count)`  
*Right-shift:  $X \gg = \text{count}$ .*
- `int mpi_cmp_abs (mpi *X, mpi *Y)`  
*Compare unsigned values.*
- `int mpi_cmp_mpi (mpi *X, mpi *Y)`  
*Compare signed values.*
- `int mpi_cmp_int (mpi *X, int z)`  
*Compare signed values.*
- `int mpi_add_abs (mpi *X, mpi *A, mpi *B)`  
*Unsigned addition:  $X = |A| + |B|$ .*
- static void `mpi_sub_hlp (int n, t_int *s, t_int *d)`
- `int mpi_sub_abs (mpi *X, mpi *A, mpi *B)`  
*Unsigned subtraction:  $X = |A| - |B|$ .*
- `int mpi_add_mpi (mpi *X, mpi *A, mpi *B)`  
*Signed addition:  $X = A + B$ .*
- `int mpi_sub_mpi (mpi *X, mpi *A, mpi *B)`  
*Signed subtraction:  $X = A - B$ .*
- `int mpi_add_int (mpi *X, mpi *A, int b)`

*Signed addition:  $X = A + b$ .*

- `int mpi_sub_int (mpi *X, mpi *A, int b)`

*Signed subtraction:  $X = A - b$ .*

- `static void mpi_mul_hlp (int i, t_int *s, t_int *d, t_int b)`
- `int mpi_mul_mpi (mpi *X, mpi *A, mpi *B)`

*Baseline multiplication:  $X = A * B$ .*

- `int mpi_mul_int (mpi *X, mpi *A, t_int b)`

*Baseline multiplication:  $X = A * b$ .*

- `int mpi_div_mpi (mpi *Q, mpi *R, mpi *A, mpi *B)`

*Division by mpi:  $A = Q * B + R$ .*

- `int mpi_div_int (mpi *Q, mpi *R, mpi *A, int b)`

*Division by int:  $A = Q * b + R$ .*

- `int mpi_mod_mpi (mpi *R, mpi *A, mpi *B)`

*Modulo:  $R = A \bmod B$ .*

- `int mpi_mod_int (t_int *r, mpi *A, int b)`

*Modulo:  $r = A \bmod b$ .*

- `static void mpi_montg_init (t_int *mm, mpi *N)`
- `static void mpi_montmul (mpi *A, mpi *B, mpi *N, t_int mm, mpi *T)`
- `static void mpi_montred (mpi *A, mpi *N, t_int mm, mpi *T)`
- `int mpi_exp_mod (mpi *X, mpi *A, mpi *E, mpi *N, mpi *_RR)`

*Sliding-window exponentiation:  $X = A^E \bmod N$ .*

- `int mpi_gcd (mpi *G, mpi *A, mpi *B)`

*Greatest common divisor:  $G = \gcd(A, B)$ .*

- `int mpi_inv_mod (mpi *X, mpi *A, mpi *N)`

*Modular inverse:  $X = A^{-1} \bmod N$ .*

- `int mpi_is_prime (mpi *X, int(*f_rng)(void *), void *p_rng)`

*Miller-Rabin primality test.*

- `int mpi_gen_prime (mpi *X, int nbits, int dh_flag, int(*f_rng)(void *), void *p_rng)`

*Prime number generation.*

- `int mpi_self_test (int verbose)`

*Checkup routine.*

## Variables

- `static const int small_prime [ ]`

### 5.117.1 Define Documentation

#### 5.117.1.1 `#define biH (ciL << 2)`

Definition at line 41 of file `bignum.c`.

Referenced by `mpi_div_mpi()`, and `mpi_mod_int()`.

#### 5.117.1.2 `#define biL (ciL << 3)`

Definition at line 40 of file `bignum.c`.

Referenced by `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_lsb()`, `mpi_montg_init()`, `mpi_msb()`, `mpi_shift_l()`, and `mpi_shift_r()`.

#### 5.117.1.3 `#define BITS_TO_LIMBS(i) (((i) + biL - 1) / biL)`

Definition at line 46 of file `bignum.c`.

Referenced by `mpi_gen_prime()`, `mpi_read_string()`, and `mpi_shift_l()`.

#### 5.117.1.4 `#define CHARS_TO_LIMBS(i) (((i) + ciL - 1) / ciL)`

Definition at line 47 of file `bignum.c`.

Referenced by `mpi_read_binary()`.

#### 5.117.1.5 `#define ciL ((int) sizeof(t_int))`

Definition at line 39 of file `bignum.c`.

Referenced by `mpi_copy()`, `mpi_free()`, `mpi_gen_prime()`, `mpi_grow()`, `mpi_is_prime()`, `mpi_lset()`, `mpi_montmul()`, `mpi_read_binary()`, `mpi_read_string()`, `mpi_write_binary()`, and `mpi_write_string()`.

### 5.117.2 Function Documentation

#### 5.117.2.1 `int mpi_add_abs (mpi * X, mpi * A, mpi * B)`

Unsigned addition:  $X = |A| + |B|$ .

##### Returns

0 if successful, 1 if memory allocation failed

Definition at line 694 of file `bignum.c`.

References `MPI_CHK`, `mpi_copy()`, `mpi_grow()`, `mpi::n`, and `mpi::p`.

Referenced by `mpi_add_mpi()`, and `mpi_sub_mpi()`.

#### 5.117.2.2 `int mpi_add_int (mpi * X, mpi * A, int b)`

Signed addition:  $X = A + b$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 860 of file bignum.c.

References `mpi_add_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_gen_prime()`, and `mpi_read_string()`.

**5.117.2.3 int mpi\_add\_mpi (mpi \* X, mpi \* A, mpi \* B)**

Signed addition:  $X = A + B$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 798 of file bignum.c.

References `mpi_add_abs()`, `MPI_CHK`, `mpi_cmp_abs()`, `mpi_sub_abs()`, and `mpi::s`.

Referenced by `mpi_add_int()`, `mpi_div_mpi()`, `mpi_inv_mod()`, `mpi_mod_mpi()`, and `rsa_private()`.

**5.117.2.4 int mpi\_cmp\_abs (mpi \* X, mpi \* Y)**

Compare unsigned values.

**Returns**

1 if  $|X|$  is greater than  $|Y|$ , -1 if  $|X|$  is lesser than  $|Y|$  or 0 if  $|X|$  is equal to  $|Y|$

Definition at line 615 of file bignum.c.

References `mpi::n`, and `mpi::p`.

Referenced by `mpi_add_mpi()`, `mpi_div_mpi()`, `mpi_montmul()`, `mpi_sub_abs()`, and `mpi_sub_mpi()`.

**5.117.2.5 int mpi\_cmp\_int (mpi \* X, int z)**

Compare signed values.

**Returns**

1 if X is greater than z, -1 if X is lesser than z or 0 if X is equal to z

Definition at line 678 of file bignum.c.

References `mpi_cmp_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mod_mpi()`, `mpi_write_hlp()`, `rsa_check_privkey()`, and `rsa_gen_key()`.

**5.117.2.6 int mpi\_cmp\_mpi (mpi \* X, mpi \* Y)**

Compare signed values.

**Returns**

1 if X is greater than Y, -1 if X is lesser than Y or 0 if X is equal to Y

Definition at line 645 of file bignum.c.

References `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `dhm_make_params()`, `dhm_make_public()`, `mpi_cmp_int()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mod_mpi()`, `mpi_self_test()`, `rsa_check_privkey()`, `rsa_gen_key()`, `rsa_private()`, and `rsa_public()`.

**5.117.2.7 int mpi\_copy (mpi \* X, mpi \* Y)**

Copy the contents of Y into X.

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 128 of file bignum.c.

References `ciL`, `MPI_CHK`, `mpi_grow()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_add_abs()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mul_mpi()`, `mpi_sub_abs()`, and `mpi_write_string()`.

**5.117.2.8 int mpi\_div\_int (mpi \* Q, mpi \* R, mpi \* A, int b)**

Division by int:  $A = Q * b + R$ .

**Returns**

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_DIVISION_BY_ZERO` if  $b == 0$

**Note**

Either Q or R can be NULL.

Definition at line 1173 of file bignum.c.

References `mpi_div_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `main()`, and `mpi_write_hlp()`.

**5.117.2.9 int mpi\_div\_mpi (mpi \* Q, mpi \* R, mpi \* A, mpi \* B)**

Division by mpi:  $A = Q * B + R$ .

**Returns**

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_DIVISION_BY_ZERO` if  $B == 0$

**Note**

Either Q or R can be NULL.



Definition at line 1008 of file bignum.c.

References biH, biL, mpi\_add\_mpi(), MPI\_CHK, mpi\_cmp\_abs(), mpi\_cmp\_int(), mpi\_cmp\_mpi(), mpi\_copy(), mpi\_free(), mpi\_grow(), mpi\_init(), mpi\_lset(), mpi\_msb(), mpi\_mul\_int(), mpi\_shift\_l(), mpi\_shift\_r(), mpi\_sub\_mpi(), mpi::n, mpi::p, mpi::s, and XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO.

Referenced by mpi\_div\_int(), mpi\_mod\_mpi(), and mpi\_self\_test().

#### 5.117.2.10 int mpi\_exp\_mod (mpi \* X, mpi \* A, mpi \* E, mpi \* N, mpi \* \_RR)

Sliding-window exponentiation:  $X = A^E \bmod N$ .

##### Returns

0 if successful, 1 if memory allocation failed, XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA if N is negative or even

##### Note

\_RR is used to avoid re-computing  $R \cdot R \bmod N$  across multiple calls, which speeds up things a bit. It can be set to NULL if the extra performance is unneeded.

Definition at line 1328 of file bignum.c.

References biL, MPI\_CHK, mpi\_cmp\_int(), mpi\_cmp\_mpi(), mpi\_copy(), mpi\_free(), mpi\_grow(), mpi\_init(), mpi\_lset(), mpi\_mod\_mpi(), mpi\_montg\_init(), mpi\_montmul(), mpi\_montred(), mpi\_msb(), mpi\_shift\_l(), mpi::n, mpi::p, and XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA.

Referenced by dhm\_calc\_secret(), dhm\_make\_params(), dhm\_make\_public(), main(), mpi\_is\_prime(), mpi\_self\_test(), rsa\_private(), and rsa\_public().

#### 5.117.2.11 void mpi\_free (mpi \* X, ...)

Unallocate one or more mpi.

Definition at line 73 of file bignum.c.

References ciL, mpi::n, mpi::p, and mpi::s.

Referenced by dhm\_free(), main(), mpi\_div\_mpi(), mpi\_exp\_mod(), mpi\_gcd(), mpi\_gen\_prime(), mpi\_inv\_mod(), mpi\_is\_prime(), mpi\_mul\_mpi(), mpi\_read\_string(), mpi\_self\_test(), mpi\_sub\_abs(), mpi\_write\_string(), rsa\_check\_privkey(), rsa\_free(), rsa\_gen\_key(), rsa\_private(), and rsa\_public().

#### 5.117.2.12 int mpi\_gcd (mpi \* G, mpi \* A, mpi \* B)

Greatest common divisor:  $G = \gcd(A, B)$ .

##### Returns

0 if successful, 1 if memory allocation failed

Definition at line 1507 of file bignum.c.

References MPI\_CHK, mpi\_cmp\_int(), mpi\_cmp\_mpi(), mpi\_copy(), mpi\_free(), mpi\_init(), mpi\_lset(), mpi\_mul\_mpi(), mpi\_shift\_r(), mpi\_sub\_abs(), mpi::p, and mpi::s.

Referenced by mpi\_inv\_mod(), rsa\_check\_privkey(), and rsa\_gen\_key().

**5.117.2.13 int mpi\_gen\_prime (mpi \* X, int nbits, int dh\_flag, int(\*) (void \*) f\_rng, void \* p\_rng)**

Prime number generation.

**Parameters**

*X* destination mpi  
*nbits* required size of X in bits  
*dh\_flag* if 1, then (X-1)/2 will be prime too  
*f\_rng* RNG function  
*p\_rng* RNG parameter

**Returns**

0 if successful (probably prime), 1 if memory allocation failed, XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA if nbits is < 3

Definition at line 1778 of file bignum.c.

References BITS\_TO\_LIMBS, ciL, mpi\_add\_int(), MPI\_CHK, mpi\_free(), mpi\_grow(), mpi\_init(), mpi\_is\_prime(), mpi\_lset(), mpi\_msb(), mpi\_shift\_l(), mpi\_shift\_r(), mpi\_sub\_int(), mpi::p, XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_MPI\_NOT\_ACCEPTABLE.

Referenced by main(), and rsa\_gen\_key().

**5.117.2.14 static int mpi\_get\_digit (t\_int \* d, int radix, char c) [static]**

Definition at line 226 of file bignum.c.

References XYSSL\_ERR\_MPI\_INVALID\_CHARACTER.

Referenced by mpi\_read\_file(), mpi\_read\_mystring(), and mpi\_read\_string().

**5.117.2.15 int mpi\_grow (mpi \* X, int nblimbs)**

Enlarge to the specified number of limbs.

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 100 of file bignum.c.

References ciL, mpi::n, and mpi::p.

Referenced by dhm\_make\_params(), dhm\_make\_public(), mpi\_add\_abs(), mpi\_copy(), mpi\_div\_mpi(), mpi\_exp\_mod(), mpi\_gen\_prime(), mpi\_is\_prime(), mpi\_lset(), mpi\_mul\_mpi(), mpi\_read\_binary(), mpi\_read\_string(), and mpi\_shift\_l().

**5.117.2.16 void mpi\_init (mpi \* X, ...)**

Initialize one or more mpi.

Definition at line 52 of file bignum.c.

References `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `main()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mul_mpi()`, `mpi_read_string()`, `mpi_self_test()`, `mpi_sub_abs()`, `mpi_write_string()`, `rsa_check_privkey()`, `rsa_gen_key()`, `rsa_private()`, and `rsa_public()`.

#### 5.117.2.17 `int mpi_inv_mod (mpi * X, mpi * A, mpi * N)`

Modular inverse:  $X = A^{-1} \bmod N$ .

##### Returns

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_BAD_INPUT_DATA` if `N` is negative or nil `XYSSL_ERR_MPI_NOT_ACCEPTABLE` if `A` has no inverse mod `N`

Definition at line 1549 of file `bignum.c`.

References `mpi_add_mpi()`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_free()`, `mpi_gcd()`, `mpi_init()`, `mpi_lset()`, `mpi_mod_mpi()`, `mpi_shift_r()`, `mpi_sub_mpi()`, `mpi::p`, `XYSSL_ERR_MPI_BAD_INPUT_DATA`, and `XYSSL_ERR_MPI_NOT_ACCEPTABLE`.

Referenced by `main()`, `mpi_self_test()`, and `rsa_gen_key()`.

#### 5.117.2.18 `int mpi_is_prime (mpi * X, int(*) (void *) f_rng, void * p_rng)`

Miller-Rabin primality test.

##### Returns

0 if successful (probably prime), 1 if memory allocation failed, `XYSSL_ERR_MPI_NOT_ACCEPTABLE` if `X` is not prime

Definition at line 1667 of file `bignum.c`.

References `ciL`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_grow()`, `mpi_init()`, `mpi_lsb()`, `mpi_mod_int()`, `mpi_mod_mpi()`, `mpi_msb()`, `mpi_mul_mpi()`, `mpi_shift_r()`, `mpi_sub_int()`, `mpi::n`, `mpi::p`, `R`, `mpi::s`, `small_prime`, and `XYSSL_ERR_MPI_NOT_ACCEPTABLE`.

Referenced by `main()`, and `mpi_gen_prime()`.

#### 5.117.2.19 `int mpi_lsb (mpi * X)`

Return the number of least significant bits.

Definition at line 185 of file `bignum.c`.

References `biL`, `int`, `mpi::n`, and `mpi::p`.

Referenced by `mpi_is_prime()`.

#### 5.117.2.20 `int mpi_lset (mpi * X, int z)`

Set value from integer.

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 167 of file bignum.c.

References ciL, MPI\_CHK, mpi\_grow(), mpi::n, mpi::p, and mpi::s.

Referenced by dhmake\_params(), dhmake\_public(), mpi\_div\_mpi(), mpi\_exp\_mod(), mpi\_gcd(), mpi\_gen\_prime(), mpi\_inv\_mod(), mpi\_mul\_mpi(), mpi\_read\_binary(), mpi\_read\_string(), and rsa\_gen\_key().

**5.117.2.21 int mpi\_mod\_int (t\_int \*r, mpi \*A, int b)**

Modulo:  $r = A \bmod b$ .

**Returns**

0 if successful, 1 if memory allocation failed, XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO if  $b == 0$

Definition at line 1209 of file bignum.c.

References biH, mpi::n, mpi::p, and XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO.

Referenced by mpi\_is\_prime(), and mpi\_write\_hlp().

**5.117.2.22 int mpi\_mod\_mpi (mpi \*R, mpi \*A, mpi \*B)**

Modulo:  $R = A \bmod B$ .

**Returns**

0 if successful, 1 if memory allocation failed, XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO if  $B == 0$

Definition at line 1189 of file bignum.c.

References mpi\_add\_mpi(), MPI\_CHK, mpi\_cmp\_int(), mpi\_cmp\_mpi(), mpi\_div\_mpi(), and mpi\_sub\_mpi().

Referenced by mpi\_exp\_mod(), mpi\_inv\_mod(), mpi\_is\_prime(), rsa\_check\_privkey(), rsa\_gen\_key(), and rsa\_private().

**5.117.2.23 static void mpi\_montg\_init (t\_int \*mm, mpi \*N) [static]**

Definition at line 1259 of file bignum.c.

References biL, and mpi::p.

Referenced by mpi\_exp\_mod().

**5.117.2.24 static void mpi\_montmul (mpi \*A, mpi \*B, mpi \*N, t\_int mm, mpi \*T) [static]**

Definition at line 1277 of file bignum.c.

References ciL, mpi\_cmp\_abs(), mpi\_mul\_hlp(), mpi\_sub\_hlp(), mpi::n, and mpi::p.

Referenced by mpi\_exp\_mod(), and mpi\_montred().

**5.117.2.25 static void mpi\_montred (mpi \* A, mpi \* N, t\_int mm, mpi \* T) [static]**

Definition at line 1314 of file bignum.c.

References mpi\_montmul(), mpi::n, mpi::p, and mpi::s.

Referenced by mpi\_exp\_mod().

**5.117.2.26 int mpi\_msb (mpi \* X)**

Return the number of most significant bits.

Definition at line 200 of file bignum.c.

References biL, mpi::n, and mpi::p.

Referenced by d2i\_RSA\_PUBKEY(), main(), mpi\_div\_mpi(), mpi\_exp\_mod(), mpi\_gen\_prime(), mpi\_is\_prime(), mpi\_shift\_l(), mpi\_size(), mpi\_write\_string(), rsa\_check\_pubkey(), rsa\_decryption(), rsa\_encryption(), and rsa\_gen\_key().

**5.117.2.27 static void mpi\_mul\_hlp (int i, t\_int \* s, t\_int \* d, t\_int b) [static]**

Definition at line 892 of file bignum.c.

References MULADDC\_CORE, MULADDC\_INIT, and MULADDC\_STOP.

Referenced by mpi\_montmul(), and mpi\_mul\_mpi().

**5.117.2.28 int mpi\_mul\_int (mpi \* X, mpi \* A, t\_int b)**

Baseline multiplication:  $X = A * b$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 992 of file bignum.c.

References mpi\_mul\_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by mpi\_div\_mpi(), and mpi\_read\_string().

**5.117.2.29 int mpi\_mul\_mpi (mpi \* X, mpi \* A, mpi \* B)**

Baseline multiplication:  $X = A * B$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 956 of file bignum.c.

References MPI\_CHK, mpi\_copy(), mpi\_free(), mpi\_grow(), mpi\_init(), mpi\_lset(), mpi\_mul\_hlp(), mpi::n, mpi::p, and mpi::s.

Referenced by main(), mpi\_gcd(), mpi\_is\_prime(), mpi\_mul\_int(), mpi\_self\_test(), rsa\_check\_privkey(), rsa\_gen\_key(), and rsa\_private().

**5.117.2.30 int mpi\_read\_binary (mpi \* *X*, unsigned char \* *buf*, int *buflen*)**

Import *X* from unsigned binary data, big endian.

**Parameters**

*X* destination mpi  
*buf* input buffer  
*buflen* input buffer size

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 484 of file bignum.c.

References CHARS\_TO\_LIMBS, ciL, MPI\_CHK, mpi\_grow(), mpi\_lset(), and mpi::p.

Referenced by asn1\_get\_mpi(), d2i\_RSA\_PUBKEY(), dhm\_read\_bignum(), dhm\_read\_public(), rsa\_private(), and rsa\_public().

**5.117.2.31 int mpi\_read\_file (mpi \* *X*, int *radix*, FILE \* *fin*)**

Read *X* from an opened file.

**Parameters**

*X* destination mpi  
*radix* input numeric base  
*fin* input file handle

**Returns**

0 if successful, or an XYSSL\_ERR\_MPI\_XXX error code

Definition at line 391 of file bignum.c.

References mpi\_get\_digit(), mpi\_read\_string(), and XYSSL\_ERR\_MPI\_FILE\_IO\_ERROR.

Referenced by main().

**5.117.2.32 int mpi\_read\_mystring (mpi \* *X*, int *radix*, char \* *s*)**

Definition at line 418 of file bignum.c.

References mpi\_get\_digit(), and mpi\_read\_string().

Referenced by rsa\_decryption(), and rsa\_encryption().

**5.117.2.33 int mpi\_read\_string (mpi \* *X*, int *radix*, char \* *s*)**

Import from an ASCII string.

**Parameters**

*X* destination mpi

*radix* input numeric base  
*s* null-terminated string buffer

### Returns

0 if successful, or an XYSSL\_ERR\_MPI\_XXX error code

Definition at line 243 of file bignum.c.

References BITS\_TO\_LIMBS, ciL, int, mpi\_add\_int(), MPI\_CHK, mpi\_free(), mpi\_get\_digit(), mpi\_grow(), mpi\_init(), mpi\_lset(), mpi\_mul\_int(), mpi::p, mpi::s, and XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA.

Referenced by main(), mpi\_read\_file(), mpi\_read\_mystring(), mpi\_self\_test(), rsa\_self\_test(), and ssl\_set\_dh\_param().

#### 5.117.2.34 int mpi\_self\_test (int *verbose*)

Checkup routine.

### Returns

0 if successful, or 1 if the test failed

Definition at line 1854 of file bignum.c.

References MPI\_CHK, mpi\_cmp\_mpi(), mpi\_div\_mpi(), mpi\_exp\_mod(), mpi\_free(), mpi\_init(), mpi\_inv\_mod(), mpi\_mul\_mpi(), and mpi\_read\_string().

Referenced by main().

#### 5.117.2.35 int mpi\_shift\_l (mpi \* *X*, int *count*)

Left-shift:  $X \ll= \text{count}$ .

### Returns

0 if successful, 1 if memory allocation failed

Definition at line 526 of file bignum.c.

References biL, BITS\_TO\_LIMBS, MPI\_CHK, mpi\_grow(), mpi\_msb(), mpi::n, and mpi::p.

Referenced by mpi\_div\_mpi(), mpi\_exp\_mod(), and mpi\_gen\_prime().

#### 5.117.2.36 int mpi\_shift\_r (mpi \* *X*, int *count*)

Right-shift:  $X \gg= \text{count}$ .

### Returns

0 if successful, 1 if memory allocation failed

Definition at line 575 of file bignum.c.

References biL, mpi::n, and mpi::p.

Referenced by dhm\_make\_params(), dhm\_make\_public(), mpi\_div\_mpi(), mpi\_gcd(), mpi\_gen\_prime(), mpi\_inv\_mod(), and mpi\_is\_prime().

**5.117.2.37 int mpi\_size (mpi \* X)**

Return the total size in bytes.

Definition at line 218 of file bignum.c.

References mpi\_msb().

Referenced by dhm\_calc\_secret(), dhm\_make\_params(), dhm\_read\_params(), mpi\_write\_binary(), x509parse\_crt(), and x509parse\_key().

**5.117.2.38 int mpi\_sub\_abs (mpi \* X, mpi \* A, mpi \* B)**

Unsigned subtraction:  $X = |A| - |B|$ .

**Returns**

0 if successful, XYSSL\_ERR\_MPI\_NEGATIVE\_VALUE if B is greater than A

Definition at line 761 of file bignum.c.

References MPI\_CHK, mpi\_cmp\_abs(), mpi\_copy(), mpi\_free(), mpi\_init(), mpi\_sub\_hlp(), mpi::n, mpi::p, and XYSSL\_ERR\_MPI\_NEGATIVE\_VALUE.

Referenced by mpi\_add\_mpi(), mpi\_gcd(), and mpi\_sub\_mpi().

**5.117.2.39 static void mpi\_sub\_hlp (int n, t\_int \* s, t\_int \* d) [static]**

Definition at line 740 of file bignum.c.

Referenced by mpi\_montmul(), and mpi\_sub\_abs().

**5.117.2.40 int mpi\_sub\_int (mpi \* X, mpi \* A, int b)**

Signed subtraction:  $X = A - b$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 876 of file bignum.c.

References mpi\_sub\_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by main(), mpi\_gen\_prime(), mpi\_is\_prime(), rsa\_check\_privkey(), and rsa\_gen\_key().

**5.117.2.41 int mpi\_sub\_mpi (mpi \* X, mpi \* A, mpi \* B)**

Signed subtraction:  $X = A - B$ .

**Returns**

0 if successful, 1 if memory allocation failed

Definition at line 829 of file bignum.c.

References mpi\_add\_abs(), MPI\_CHK, mpi\_cmp\_abs(), mpi\_sub\_abs(), and mpi::s.

Referenced by mpi\_div\_mpi(), mpi\_inv\_mod(), mpi\_mod\_mpi(), mpi\_sub\_int(), and rsa\_private().



**5.117.2.42 void mpi\_swap (mpi \* *X*, mpi \* *Y*)**

Swap the contents of *X* and *Y*.

Definition at line 155 of file bignum.c.

Referenced by rsa\_gen\_key().

**5.117.2.43 int mpi\_write\_binary (mpi \* *X*, unsigned char \* *buf*, int *buflen*)**

Export *X* into unsigned binary data, big endian.

**Parameters**

*X* source mpi

*buf* output buffer

*buflen* output buffer size

**Returns**

0 if successful, XYSSL\_ERR\_MPI\_BUFFER\_TOO\_SMALL if *buf* isn't large enough

**Note**

Call this function with *\*buflen* = 0 to obtain the minimum required buffer size in *\*buflen*.

Definition at line 506 of file bignum.c.

References ciL, mpi\_size(), mpi::p, and XYSSL\_ERR\_MPI\_BUFFER\_TOO\_SMALL.

Referenced by dhm\_calc\_secret(), dhm\_make\_public(), rsa\_private(), and rsa\_public().

**5.117.2.44 int mpi\_write\_file (char \* *p*, mpi \* *X*, int *radix*, FILE \* *fout*)**

Write *X* into an opened file, or stdout.

**Parameters**

*p* prefix, can be NULL

*X* source mpi

*radix* output numeric base

*fout* output file handle

**Returns**

0 if successful, or an XYSSL\_ERR\_MPI\_XXX error code

**Note**

Set *fout* == NULL to print *X* on the console.

Definition at line 447 of file bignum.c.

References MPI\_CHK, mpi\_write\_string(), and XYSSL\_ERR\_MPI\_FILE\_IO\_ERROR.

Referenced by generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), and main().

**5.117.2.45 static int mpi\_write\_hlp (mpi \* *X*, int *radix*, char \*\* *p*) [static]**

Definition at line 301 of file bignum.c.

References MPI\_CHK, mpi\_cmp\_int(), mpi\_div\_int(), mpi\_mod\_int(), and XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA.

Referenced by mpi\_write\_string().

**5.117.2.46 int mpi\_write\_string (mpi \* *X*, int *radix*, char \* *s*, int \* *slen*)**

Export into an ASCII string.

**Parameters**

*X* source mpi  
*radix* output numeric base  
*s* string buffer  
*slen* string buffer size

**Returns**

0 if successful, or an XYSSL\_ERR\_MPI\_XXX error code

**Note**

Call this function with \*slen = 0 to obtain the minimum required buffer size in \*slen.

Definition at line 328 of file bignum.c.

References ciL, MPI\_CHK, mpi\_copy(), mpi\_free(), mpi\_init(), mpi\_msb(), mpi\_write\_hlp(), mpi::n, mpi::p, mpi::s, XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_MPI\_BUFFER\_TOO\_SMALL.

Referenced by mpi\_write\_file().

**5.117.3 Variable Documentation****5.117.3.1 const int small\_prime[] [static]****Initial value:**

```
{
    3,    5,    7,   11,   13,   17,   19,   23,
   29,   31,   37,   41,   43,   47,   53,   59,
   61,   67,   71,   73,   79,   83,   89,   97,
  101,  103,  107,  109,  113,  127,  131,  137,
  139,  149,  151,  157,  163,  167,  173,  179,
  181,  191,  193,  197,  199,  211,  223,  227,
  229,  233,  239,  241,  251,  257,  263,  269,
  271,  277,  281,  283,  293,  307,  311,  313,
  317,  331,  337,  347,  349,  353,  359,  367,
  373,  379,  383,  389,  397,  401,  409,  419,
  421,  431,  433,  439,  443,  449,  457,  461,
  463,  467,  479,  487,  491,  499,  503,  509,
  521,  523,  541,  547,  557,  563,  569,  571,
  577,  587,  593,  599,  601,  607,  613,  617,
```

```
        619,  631,  641,  643,  647,  653,  659,  661,  
        673,  677,  683,  691,  701,  709,  719,  727,  
        733,  739,  743,  751,  757,  761,  769,  773,  
        787,  797,  809,  811,  821,  823,  827,  829,  
        839,  853,  857,  859,  863,  877,  881,  883,  
        887,  907,  911,  919,  929,  937,  941,  947,  
        953,  967,  971,  977,  983,  991,  997, -103  
    }
```

Definition at line 1639 of file bignum.c.

Referenced by mpi\_is\_prime().

## 5.118 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/certs.c File Reference

```
#include "xyssl/config.h"
```

### Variables

- char [test\\_ca\\_cert](#) []
- char [test\\_ca\\_key](#) []
- char [test\\_ca\\_pwd](#) [] = "test"
- char [test\\_srv\\_cert](#) []
- char [test\\_srv\\_key](#) []
- char [test\\_cli\\_cert](#) []
- char [test\\_cli\\_key](#) []
- char [xyssl\\_ca\\_cert](#) []

### 5.118.1 Variable Documentation

#### 5.118.1.1 char test\_ca\_cert[]

**Initial value:**

```
"-----BEGIN CERTIFICATE-----\r\n"
"MIIDpTCCAo2gAwIBAgIBADANBgkqhkiG9w0BAQUFADBFMQswCQYDVQQGEwJGUjEO\r\n"
"MAwGA1UEBxMFUGFyaXMxDjAMBgNVBAoTBVh5U1NMMRYwFAYDVQQDEw1YeVNTTCBU\r\n"
"ZXN0IENBMB4XDTA3MDcwNzA1MDAxOFoXDTE3MDcwNzA1MDAxOFowRTElMAkGA1UE\r\n"
"BhMCRlIxZjAMBgNVBAcTBVBhcm1zMQ4wDAYDVQQKEwVYeVNTTDEWMBQGA1UEAxMN\r\n"
"WH1TU0wgVGZvdCBBDQTCASiWdQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAM+k\r\n"
"gt70fIPiYqmvXr+9uPmW0N405eoSzxdiRLLCHqL4V/Ts0E/H+JNfHS4DHLAgxrJu\r\n"
"+ZiadvSJuhKI6eliMkAh5SU1DqaF3jrrFdJCooM6a077M4CRKE1tdAeZdf+BYp0q\r\n"
"BeMU9Y2+j7ibsQaPaizunbXLf4QdteExCwhYRJ8OVXSEasNt339gJzTD6k0hES3b\r\n"
"lEN3qbx6lqFaJ5MLHTix5uNVc2rvbOizV5oLhJNqm52AKOp1ltv6WTiI8loagvAc\r\n"
"jlhEZRNB9el5SL6Jai/uFcqXKzfXNKW3FYpQHfGobmiMfGt1lUSBj3F2mrqEC7gC\r\n"
"whY/FDvAI64/k5LZAFkCAwEAAOBnzCBnDAMBgNVHRMEBTADAQH/MB0GA1UdDgQW\r\n"
"BBS87h+Y6Porg+SkfV7DdXKTMdkyZzBtBgNVHSMEZjBkgBS87h+Y6Porg+SkfV7D\r\n"
"dXKTMdkyZ6FJpEcwRTElMAkGA1UEBhMCRlIxZjAMBgNVBAcTBVBhcm1zMQ4wDAYD\r\n"
"VQOKEwVYeVNTTDEWMBQGA1UEAxMNWH1TU0wgVGZvdCBBDQYIBADANBgkqhkiG9w0B\r\n"
"AQUFAAOCAQEAIHdohONCg6KAAhWDSmfEgSbKUI8/Zr/d56uw42H00sj/uKPQzUco\r\n"
"3Mx2BYElm1itg7q5OhrkB7J4ZB78EtNZM84nV+y6od3YpR0Z9VUxCx7948MozYRy\r\n"
"TKF5x/lKHXr1PJKfEO4clKdWTFAtWtGhewXrHJQ8C+ENh2Up2wTVh3Z+pEzuZNv3\r\n"
"u/JYu1H+vkt3l1WCy/9mxUnu+anW1DzxPwnjy4lX6Mi0BD2qfKBWLjVS+7v6ALCj\r\n"
"S2oRWWr4LUvXT7z9BBAvw2eJQD+a4uAya6EURG7AsAvr5MnWn/r0wLWmBJ6fBlYp\r\n"
"FlkOmamOFvstLMf74rLX+LGKeJ/nwui5FQ==\r\n"
"-----END CERTIFICATE-----\r\n"
```

Definition at line 25 of file certs.c.

Referenced by [main\(\)](#), [ssl\\_test\(\)](#), and [x509\\_self\\_test\(\)](#).

#### 5.118.1.2 char test\_ca\_key[]

Definition at line 49 of file certs.c.

Referenced by [x509\\_self\\_test\(\)](#).

### 5.118.1.3 char test\_ca\_pwd[] = "test"

Definition at line 81 of file certs.c.

Referenced by x509\_self\_test().

### 5.118.1.4 char test\_cli\_crt[]

**Initial value:**

```
"-----BEGIN CERTIFICATE-----\r\n"
"MIIDPTCCAiwGAwIBAgIBATANBgkqhkiG9w0BAQUFADBQMswCQQYDVQQGEwJGUjE0\r\n"
"MAwGA1UEBxMFUGFyaXMxdjAMBGNVBAoTBVh5U1NMMRYwFAYDVQQDEw1YeVNTTCBU\r\n"
"ZXN0IENBMB4XDTA3MDcwNzAlMDEyMFoXDTA4MDcwNjAlMDEyMFowMDELMAkGA1UE\r\n"
"BhMCRlIxdjAMBGNVBAoTBVh5U1NMMREwDwYDVQQDEwhKb2UgVXNlcjCCASIdQYJ\r\n"
"KoZlIhvcNAQEBBQADgEPADCCAQoCggEBAKZkg8ANl6kGmLGqKc6KUHb9IsZwb2+K\r\n"
"jBw83Qb0KuvPVnu3MzEcXFvOZ83g0PL/z8ob5PKr8HP6bVYzhsD65imcCDEEVPk2\r\n"
"9r0XGTggGjB601Fd8aTShUWE4NLrKw6YNXTXgTndvhHNxXwqmdNVLkmZjj3ZwYUc\r\n"
"eE8eE5jHs8cMDXJLMCwgKIM7Sax220hSHQHKwifVO4/Fdw5G+Suys8PhMX2jDXM\r\n"
"ICFwq8ld+bZGoNUtgp48FWhAMfJyTEaHh9LC46KkqGSDRIzx7/4cPB6QqrpzJN0o\r\n"
"Kr8KH7vdRDTFDm023D4C5l0Bw/2aC76DhEJpB2bGA4iIszJs+F/PIL8CAwEAAaNN\r\n"
"MEswCQYDVROTBAlwADAdBgNVHQ4EFgQUiWX1IvjrDYgt0zz5Sq16x01k0o4wHwYD\r\n"
"VR0jBBBgwFoAUvO4fm0j6K4PkpH1ew3VyKzHZMmcwDQYJKoZIhvcNAQEFBQADggEB\r\n"
"AGdQD7VthJmC+oeemuHk2TQX2wZNU+GsC+RLjtlencnky95KnljGvMtCznyLkS5D\r\n"
"faJLKfRlNo8pk5gRdscqgyIuQx5WnHNv4QBZmMsmvDICxzRQaxuPFHbS4aLXldeL\r\n"
"yOWm5Z4qkMHPCKvA86blYsEkksGDV47fF9ZkOQ8nkh7Z4eY4/5TwqTY72ww5g4NL\r\n"
"6DZtWpcpGbX99NRANvzcq9D+ElxkgHnH4YWafOKBclSgqrutbRLi2uZx/QpvuF+i\r\n"
"sUbe+HFPMWwU5lBv/oOhQkz0VD+HusYtXWS2lG88ct40aNLy2CkYUugdTR/b9Uea\r\n"
"p/i862sL/lo40qlQ0xv5N7U=\r\n"
"-----END CERTIFICATE-----\r\n"
```

Definition at line 134 of file certs.c.

Referenced by main(), and x509\_self\_test().

### 5.118.1.5 char test\_cli\_key[]

**Initial value:**

```
"-----BEGIN RSA PRIVATE KEY-----\r\n"
"MIIEowIBAAKCAQEApmSDwa2XqQaYsaopzopQdv0ixnBvb4qMHDzdBvQq689We7cz\r\n"
"MRxcW85nzeDQ8v/Pyhvk8qvw/ptVjOGwPrmKZwIMQRU+Tb2vRcZOCaAMHrTUV3x\r\n"
"pnKFRYtG0usrDpgldNeBM12+Ec3FfCqZ0lUuSZmOPdnBhRxwQTx4TmMezxwNcks\r\n"
"wLCAogztJrHbY6FIdAcrCJ9U7j8V3Dkb5K7Kzw+ExfaMNCwgIXCryV35tkag1S2C\r\n"
"njwVaEAX8nJMRoeH0sLjoqSoZINEjPHv/hw8HpCqunMk3SgqvYQfu91ENMUOY7bc\r\n"
"PgLMxQHD/ZoLvoOEQmKHzSYDiIizMmz4X88gvwIDAQABAoIBAEOBnBKAjDVN+j4ax\r\n"
"lDjEwZKqxVkmAUXDBDyDrCjxRoWY2gz7YWlALUMUbeV0f05v1zVrwbkUKKZeVBxI\r\n"
"QA9zRw28H8A6tfvolHgrIcx4dixMh3ePC+DVDJ6zglvKV2ipAwBufKYIrX0r4Io2\r\n"
"ZqUrNg9CeEYNlkHWceaN12rhYwO82pgHxnB1p5pI42pY7lzyLgSddf5n+M5UBOJI\r\n"
"gsNckvbGdv7WQPVFTRDiRgEnCJ3rI8oPSK6MOUWJw3rh2hbKx+ex8NPvEKbzEXiU\r\n"
"p5j1AlbHIWP5sYBbAlYviFtryAV4fyfLcWpfoqa33Oozofjlwoj0Aixz+6rerLjZ\r\n"
"cpTSrAECgYEA2oCcfUo6HH3Lq9oeWhFCOyG3YjZmFrJaJwJHnvroX9/pXHqYKog\r\n"
"TeHcjUJBtFzW0klcetYbZCFqT8v9nf0uPlgaiVGctXf1MSbFXDUFKkYBiFwzdWMT\r\n"
"Ysmvfhf82jMWZ8ecsXTyDRl858R5WPZ52qEsCc5X2un7QENm6FtVT8CgYEAwwKS\r\n"
"zQNzuoJETqZX7AalMk3JM8Fdam+Qm5LNMcbvbkKI8HKMS1VMuqaR0XdAX/imXAx\r\n"
"PlVhSsmoSDbsMpxBEZlptpCen/GcqtITxANTakrBHxqb2aQ5EEu7SgzfHZWse3\r\n"
"vQEYfctFB1PldcZUDzk4/w7WmyivpYtCWoaAh1IECgYEA0UYZ+1UJfVpapRj+swMP\r\n"
"DrQbo7i7t7lUaFYLNpFX2OPLTWC5txqn1OruTu5VHDqE+5hneDNUUTT3uOg4B2q\r\n"
"mdmmaNjh2M6wz0e0BVfexhNQynqMaqTe32IOM8DFs3L0xacgg7JfVn6P7CeQGOVe\r\n"
"wc96kICw6ZxhtJSqpOGipt8CgYBI/OPw+IXxJK4nNSpe+u4vCYP5mUI9hKEFYCbt\r\n"
"qKwvyAUknn/zgiIQ+z/iSErFMPmlwXjvWi0gL/qPb+Fp4hCLX8u2zNhY08P4Gin\r\n"
```

```
"Ej+pAntWxq+kHyfKEI5dyRwV/snfv1qwJy404JsSF3VMhIMdYDPzbb72Qnni5w5l\r\n"
"j00eAQKBgBqt9jJMd1JdpemC2dm0BuuDIz2h3/MH+CMjfaDLenVpKykn17B6N92h\r\n"
"k1MesqK3RQzDGwauDw431LQw0R69onn9fCM3wJw2yEC6wC9sF8I8hsNZbt64yZhZ\r\n"
"4Bi2YRTiHhpEuBqKlHHLDFHneo3SMYh8PU/PDQQcyWGHHUi9z1RE\r\n"
"-----END RSA PRIVATE KEY-----\r\n"
```

Definition at line 156 of file certs.c.

Referenced by main().

### 5.118.1.6 char test\_srv\_crt[]

**Initial value:**

```
"-----BEGIN CERTIFICATE-----\r\n"
"MIIDPjCCAiaGAWIBAgIBAJANBgkqhkiG9w0BAQUFADBFMQswCQYDVQQGEwJGUjE0\r\n"
"MAwGA1UEBxMFUGFyaXMxMDjAMBGNVBAoTBVh5U1NMMRYwFAyDVQQDEw1YeVNTTCBU\r\n"
"ZXN0IENBMB4XDTA3MDcwNzA1MDEyOVVoXDTA4MDcwNjA1MDEyOVowMTELMakGA1UE\r\n"
"BhMCRlIxZjAMBGNVBAoTBVh5U1NMMRIwEAYDVQQDEw1sb2NhbGhvc3QwgGEiMA0G\r\n"
"CSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQc40PDcGTgmHkt6noXDfkjVuymjiNYB\r\n"
"gtjtI7uAlKe3tXStacEecQek/OJxYqYr7ffcWals29LL6HbKpi0xLZKBbD9ACkDh\r\n"
"1Z/SvHlyQPILJdYb9DMw+kzZds5myXUjzn7Aem1YjoxMZUAMyc34i2900X2pL0v2\r\n"
"SFCEJ9Ym4MOnZxYl217+dX9ZbkGIgrT6uY2IYK4boDwxbTcyT8i/NPsvsiMwtWPM\r\n"
"rnQMr+XbgS98sUzcZE70Pe1TlV9Iy8j/8d2OiFo+qTyMu/6UpM2s3gdkQkMzx+Sm\r\n"
"4QitRUjzmEXeUePRUjEgHiv7vz069xuVBzrks36w5BXiVAhLke/OTKVPAGMBAAGj\r\n"
"TTBLMAkGA1UdEwQCAAAwHQYDVROBBYEFNkOyCTx64SDdPySGW1/tzD7/WMSMB8G\r\n"
"AlUdIwQYMBaAFZluH5jo+iuD5KR9XsN1cpMx2TJnMA0GCSqGSIb3DQEBBQUAA4IB\r\n"
"AQBelJv5t+suayq5Lo5bJneHjNZfgg8EigDQ7Nqaosv1QZAsh2N34Gg5YdkGyVdg\r\n"
"s32I/K5aaywyUbG9qVXQxCM2T95qBqyK56h9yJoZKwQD9H//+zB8kCK/16WvRfv3\r\n"
"V7eSR19qOFWlHe+1qGh2YhxedUfyi+fm4D36dGxqC2A34tzjo0QPHKtIeqM0kJy\r\n"
"zzL65T1bJQKkyTurHofFv0jW9ZFG2wkGysVgCY5fjuLi1do/sWUaXd2987iNfa+K\r\n"
"FrHsTi6urSfZuG1ZNxDXDHEE7Q2snAvvev+KR7DD9X4DJGcPX9gA4CGJj+9ZzyAA\r\n"
"ZTGpOzk1hIH44RFs2lJMZR1E\r\n"
"-----END CERTIFICATE-----\r\n"
```

Definition at line 83 of file certs.c.

Referenced by main(), and ssl\_test().

### 5.118.1.7 char test\_srv\_key[]

**Initial value:**

```
"-----BEGIN RSA PRIVATE KEY-----\r\n"
"MIIEowIBAAKCAQEAuNDw3Bk4Jh5Lep6Fw35I1bsp04jWAYI7Yi+7gNSnt7V0rWnB\r\n"
"HnEHpPzicWKmK+333FmpUtvSy+h2yqYtMS2SgWw/QApA4dWf0rx5ckDyCyXWG/Qz\r\n"
"MPpM2XbOZs11I85+wHptWI6MTGVADMnN+ItvdNF9qS9L9knwni fWJuDDp2cWJdte\r\n"
"/nV/WW5ICIK0+rmNiGCuG6A8MW03Mk/IvzT7FbIjMLVjzK50DK/124EvfLFM3GRO\r\n"
"9D3tU5VfSMvI/HdjohaPqk8jLv+1KTNrN4HZEJDM8fkpuEIrUVI85hf31Hj0Vix\r\n"
"IBYL+7890vcblQc65LN+sOQV4lQIS5HvzkylTwIDAQABAoIBABeah8h0aBlmMRmd\r\n"
"+VN4Y3D4kf7UcRCMQ21MzlOq1Si/QgGLyiBLK0DFE16LzNE7eTZpNRjh/lAQhmtn\r\n"
"QcpQGa/xlTomlRbCo8DUVWZkKQWHdYroa0lMDliPtDimzhEepE2M1T5EJmLzY3S+\r\n"
"qVGe7UMsJjJfWgJAezyXteANQK+2YSt+CjPIqIHchlKexUnvdN9++1oEx6AbuZ8T\r\n"
"4avhFYZQP15tZNGsk2LfQlYS/NfbowkCsd0/TVubJBmDGUML/E5MbxjxLz1aNB2M\r\n"
"V59cBNngsgA35CODAUf4xOyoSfZGqG1Rb9qQrv1E6Jz56dG8SsKF3HqnDjxiPOVBN\r\n"
"FBnVJ+ECgYEA29MhAsKMM4XqBUKp6pIMFTgm/s1E5vxiG70vqiIL+guvBhhQ7zs1\r\n"
"8UMTNXxOME1NoB/ev9fN0Cjc1Vr46b/x/yDw7wMb96i+vzEN0zu4RHWi3OWpCPbp\r\n"
"qBKEi3hzN8M+BulPX8CDQx3aLRrfxw51J5EuA0NeybngbItgxTi0u6kCgYEA1zr0\r\n"
"6P5YdOhYHTSWD1ked49MApcVuzaHnsHZVAhUqu3Rwiy9LRaJLZfr7fQDb9DYJbZp\r\n"
"sxTRLG6LSAcSR7mw+m+GvNqGt/9pSqbtW+L/VwVWSyF+YYklxZUD3UAAyrDVcDEC\r\n"
"a5S+jad4Csi/1Vht5ulWlckWl1fJvadn5ubKNDcCgYA+71xVGPP+1sFgTiytfrC8\r\n"
```

```
"5n2rl4MxinJ9+w0I+EbZCKNMYGvTgiU4dJasSMEdiBKslFMGo7dF8F0BLHF1IsIa\r\n"
"5Ah2tXItXn9l54o90iTQXMmK6qmRaneM6fhOoeaCwYAhpGxYIpqx/Xr4TOhiag46\r\n"
"jMMaphAeOvw4t1K2RDziOQKBgQCYPCCU0gxuw/o1jda2CxbZy9EmU/erEX09+0n+\r\n"
"TOFQpSEPq/z9WaxAFY9LfsdZ0ZktoeHmalbNdL3i6A3DWAM3YSQzQMRPmzOWnqXx\r\n"
"cgoCBmlvzkzaeLjO5phMoLQHJmmafvuCG6uxov3F8Hi3LyHUF2c8k0nL6ucmJ3vj\r\n"
"uzu4AQKBgBSASMAJS63M9UJB1Eazy2v2Nww04CmzNxUfWrHuKpd/C2ik4QKu0sRO\r\n"
"r9KnkDgxxEhjDm7lXhlW12PU42yORst5I3EaalCfi4KPFn/ozt+iNBYrzd8Tyvnb\r\n"
"qkdEC10+G2Fo/ER4NRCv7a24WNEsOMGzGRqw5cnSJrjzbZLYMaIyK\r\n"
"-----END RSA PRIVATE KEY-----\r\n"
```

Definition at line 105 of file certs.c.

Referenced by main(), and ssl\_test().

### 5.118.1.8 char xyssl\_ca\_crt[]

**Initial value:**

```
"-----BEGIN CERTIFICATE-----\r\n"
"MIID4DCCAsigAwIBAgIJAOLw9BMV1jxMMA0GCSqGSIb3DQEBBQUAMFMxCzAJBgNV\r\n"
"BAYTAKZSMQ4wDAYDVQQIEwVQYXJpczEOMAwGA1UEChMFWH1TU0wxJDAiBgNVBAMT\r\n"
"G1h5U1NMIENlcnRpZmljYXRlIEF1dGhvcml0eTAeFw0wNjEwMzEyMjU5MjRaFw0x\r\n"
"NjEwMzEyMjU5MjRaMFmxCzAJBgNVBAYTAKZSMQ4wDAYDVQQIEwVQYXJpczEOMAwG\r\n"
"A1UEChMFWH1TU0wxJDAiBgNVBAMTG1h5U1NMIENlcnRpZmljYXRlIEF1dGhvcml0\r\n"
"eTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAKnprf1RQ7IYPI3FmI/h\r\n"
"e2EJGfaiP+Jt551VZWFrs3A56Nn4KS57zTAKZUA7YGBLwLAznfaphJ7SvENALeZR\r\n"
"/J0c/n9jwMpfXReQL7RVpgg/zlR+t2DUI3DAwigPZiHHCSJSBC73vpMc6uH0eV2d\r\n"
"itqjjUnJG5F8Zg9/gX4UMRADlwGWqxvs+jc9i0XFKOEzga8+rOn6WvKyBM4e20I\r\n"
"Hc04BPF92d6sm4qLgyR4oXUkBz6NfDWX8ZdTvXuRaK9qMy1327cCT48sis9F6/eK\r\n"
"QAx0VHlqGTtxDYjHJPsmLNejuyUDvsaC8TMCSpFTXpMTUvJdmxsF1LZWmIo5lY5\r\n"
"zh0CAwEAAaOBtjCBszAdBgNVHQ4EFgQUBlVzoNjrkkgPkj8xjuFvVP2E4GswgYMG\r\n"
"AlUdIwR8MHqAFaZVc6DSa5JID5CFMY7hb1T9hOBroVekVTBTMQswCQYDVQQGEwJG\r\n"
"UjEOMAwGA1UECBMFUGFyaXMxDjAMBGNVBAoTBVh5U1NMMSQwIgYDVQQDExtYeVNT\r\n"
"TCBDZXJ0aWZpY2F0ZSBBDXR0b3JpdHmCCQDi8PQTFdY8TDAMBGNVHRMEBTADAQH\r\n"
"MA0GCSqGSIb3DQEBBQUAA4IBAQCgD65b2l5BASFSpVlrcRnLzu/99eWTVAJwJbbD\r\n"
"VhPAQiET0W4U/85EDK7uoFo/SEjyMB/m4T20A8FIDaK7jBPo/1gtbuQjGMR17h+z\r\n"
"F2iGuNh26Td26Uzqclt3oiFtSvDRoZ/9kqkEy7Lrs7FBzOmvfTvrqvADf7cLMA2D\r\n"
"ri/otDpzPr4XoDnwd4C+4bQC/Gr3Uder4VAeTOJtKdGqfYLVpWPSPVBDuVLUybKi\r\n"
"8cMAT6p9IG1e12u6vFqcBT/I67Q0bGU6gzVVz9ZVULXOYZMjjLafVXC1gesUH2WT\r\n"
"gTEAnEBkSRrkfAi+RezoEFAbmEl3fPt09dwSPku3x7cB3zaJ\r\n"
"-----END CERTIFICATE-----\r\n"
```

Definition at line 185 of file certs.c.

Referenced by main().

## 5.119 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/debug.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/debug.h"
#include <stdarg.h>
#include <stdlib.h>
```

### Functions

- char \* [debug\\_fmt](#) (const char \*format,...)
- void [debug\\_print\\_msg](#) (ssl\_context \*ssl, int level, char \*file, int line, char \*text)
- void [debug\\_print\\_ret](#) (ssl\_context \*ssl, int level, char \*file, int line, char \*text, int ret)
- void [debug\\_print\\_buf](#) (ssl\_context \*ssl, int level, char \*file, int line, char \*text, unsigned char \*buf, int len)
- void [debug\\_print\\_mpi](#) (ssl\_context \*ssl, int level, char \*file, int line, char \*text, mpi \*X)
- void [debug\\_print\\_cert](#) (ssl\_context \*ssl, int level, char \*file, int line, char \*text, x509\_cert \*crt)

### 5.119.1 Function Documentation

#### 5.119.1.1 char\* debug\_fmt (const char \*format, ...)

Definition at line 38 of file debug.c.

References vsnprintf.

#### 5.119.1.2 void debug\_print\_buf (ssl\_context \*ssl, int level, char \*file, int line, char \*text, unsigned char \*buf, int len)

Definition at line 82 of file debug.c.

References \_ssl\_context::f\_dbg, and \_ssl\_context::p\_dbg.

#### 5.119.1.3 void debug\_print\_cert (ssl\_context \*ssl, int level, char \*file, int line, char \*text, x509\_cert \*crt)

Definition at line 169 of file debug.c.

References debug\_print\_mpi(), rsa\_context::E, \_ssl\_context::f\_dbg, rsa\_context::N, \_x509\_cert::next, \_ssl\_context::p\_dbg, \_x509\_cert::rsa, and x509parse\_cert\_info().

#### 5.119.1.4 void debug\_print\_mpi (ssl\_context \*ssl, int level, char \*file, int line, char \*text, mpi \*X)

Definition at line 124 of file debug.c.

References \_ssl\_context::f\_dbg, mpi::n, mpi::p, and \_ssl\_context::p\_dbg.

Referenced by debug\_print\_cert().



**5.119.1.5 void debug\_print\_msg (ssl\_context \* *ssl*, int *level*, char \* *file*, int *line*, char \* *text*)**

Definition at line 52 of file debug.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

**5.119.1.6 void debug\_print\_ret (ssl\_context \* *ssl*, int *level*, char \* *file*, int *line*, char \* *text*, int *ret*)**

Definition at line 66 of file debug.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

## 5.120 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/des.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/des.h"
#include <string.h>
#include <stdio.h>
```

### Defines

- #define [GET\\_ULONG\\_BE](#)(n, b, i)
- #define [PUT\\_ULONG\\_BE](#)(n, b, i)
- #define [DES\\_IP](#)(X, Y)
- #define [DES\\_FP](#)(X, Y)
- #define [DES\\_ROUND](#)(X, Y)
- #define [SWAP](#)(a, b) { unsigned long t = a; a = b; b = t; t = 0; }

### Functions

- static void [des\\_setkey](#) (unsigned long SK[32], unsigned char key[8])
- void [des\\_setkey\\_enc](#) ([des\\_context](#) \*ctx, unsigned char key[8])  
*DES key schedule (56-bit, encryption).*
- void [des\\_setkey\\_dec](#) ([des\\_context](#) \*ctx, unsigned char key[8])  
*DES key schedule (56-bit, decryption).*
- static void [des3\\_set2key](#) (unsigned long esk[96], unsigned long dsk[96], unsigned char key[16])
- void [des3\\_set2key\\_enc](#) ([des3\\_context](#) \*ctx, unsigned char key[16])  
*Triple-DES key schedule (112-bit, encryption).*
- void [des3\\_set2key\\_dec](#) ([des3\\_context](#) \*ctx, unsigned char key[16])  
*Triple-DES key schedule (112-bit, decryption).*
- static void [des3\\_set3key](#) (unsigned long esk[96], unsigned long dsk[96], unsigned char key[24])
- void [des3\\_set3key\\_enc](#) ([des3\\_context](#) \*ctx, unsigned char key[24])  
*Triple-DES key schedule (168-bit, encryption).*
- void [des3\\_set3key\\_dec](#) ([des3\\_context](#) \*ctx, unsigned char key[24])  
*Triple-DES key schedule (168-bit, decryption).*
- void [des\\_crypt\\_ecb](#) ([des\\_context](#) \*ctx, unsigned char input[8], unsigned char output[8])  
*DES-ECB block encryption/decryption.*
- void [des\\_crypt\\_cbc](#) ([des\\_context](#) \*ctx, int mode, int length, unsigned char iv[8], unsigned char \*input, unsigned char \*output)  
*DES-CBC buffer encryption/decryption.*

- void [des3\\_crypt\\_ecb](#) ([des3\\_context](#) \*ctx, unsigned char input[8], unsigned char output[8])  
*3DES-ECB block encryption/decryption*
- void [des3\\_crypt\\_cbc](#) ([des3\\_context](#) \*ctx, [int](#) mode, [int](#) length, unsigned char iv[8], unsigned char \*input, unsigned char \*output)  
*3DES-CBC buffer encryption/decryption*
- [int](#) [des\\_self\\_test](#) ([int](#) verbose)

## Variables

- static const unsigned long [SB1](#) [64]
- static const unsigned long [SB2](#) [64]
- static const unsigned long [SB3](#) [64]
- static const unsigned long [SB4](#) [64]
- static const unsigned long [SB5](#) [64]
- static const unsigned long [SB6](#) [64]
- static const unsigned long [SB7](#) [64]
- static const unsigned long [SB8](#) [64]
- static const unsigned long [LHs](#) [16]
- static const unsigned long [RHs](#) [16]
- static const unsigned char [des3\\_test\\_keys](#) [24]
- static const unsigned char [des3\\_test\\_iv](#) [8]
- static const unsigned char [des3\\_test\\_buf](#) [8]
- static const unsigned char [des3\\_test\\_ecb\\_dec](#) [3][8]
- static const unsigned char [des3\\_test\\_ecb\\_enc](#) [3][8]
- static const unsigned char [des3\\_test\\_cbc\\_dec](#) [3][8]
- static const unsigned char [des3\\_test\\_cbc\\_enc](#) [3][8]

### 5.120.1 Define Documentation

#### 5.120.1.1 #define DES\_FP(X, Y)

Value:

```
{
    X = ((X << 31) | (X >> 1)) & 0xFFFFFFFF; \
    T = (X ^ Y) & 0xAAAAAAAA; X ^= T; Y ^= T; \
    Y = ((Y << 31) | (Y >> 1)) & 0xFFFFFFFF; \
    T = ((Y >> 8) ^ X) & 0x00FF00FF; X ^= T; Y ^= (T << 8); \
    T = ((Y >> 2) ^ X) & 0x33333333; X ^= T; Y ^= (T << 2); \
    T = ((X >> 16) ^ Y) & 0x0000FFFF; Y ^= T; X ^= (T << 16); \
    T = ((X >> 4) ^ Y) & 0x0F0F0F0F; Y ^= T; X ^= (T << 4); \
}
```

Definition at line 257 of file des.c.

Referenced by [des3\\_crypt\\_ecb\(\)](#), and [des\\_crypt\\_ecb\(\)](#).

**5.120.1.2 #define DES\_IP(X, Y)****Value:**

```

{
    T = ((X >> 4) ^ Y) & 0x0F0F0F0F; Y ^= T; X ^= (T << 4); \
    T = ((X >> 16) ^ Y) & 0x0000FFFF; Y ^= T; X ^= (T << 16); \
    T = ((Y >> 2) ^ X) & 0x33333333; X ^= T; Y ^= (T << 2); \
    T = ((Y >> 8) ^ X) & 0x00FF00FF; X ^= T; Y ^= (T << 8); \
    Y = ((Y << 1) | (Y >> 31)) & 0xFFFFFFFF; \
    T = (X ^ Y) & 0xAAAAAAAA; Y ^= T; X ^= T; \
    X = ((X << 1) | (X >> 31)) & 0xFFFFFFFF; \
}

```

Definition at line 243 of file des.c.

Referenced by des3\_crypt\_ecb(), and des\_crypt\_ecb().

**5.120.1.3 #define DES\_ROUND(X, Y)****Value:**

```

{
    T = *SK++ ^ X; \
    Y ^= SB8[ (T & 0x3F) ^ \
    SB6[ (T >> 8) & 0x3F ] ^ \
    SB4[ (T >> 16) & 0x3F ] ^ \
    SB2[ (T >> 24) & 0x3F ]; \
    T = *SK++ ^ ((X << 28) | (X >> 4)); \
    Y ^= SB7[ (T & 0x3F) ^ \
    SB5[ (T >> 8) & 0x3F ] ^ \
    SB3[ (T >> 16) & 0x3F ] ^ \
    SB1[ (T >> 24) & 0x3F ]; \
}

```

Definition at line 271 of file des.c.

Referenced by des3\_crypt\_ecb(), and des\_crypt\_ecb().

**5.120.1.4 #define GET\_ULONG\_BE(n, b, i)****Value:**

```

{
    (n) = ( (unsigned long) (b)[(i) << 24] \
    | ( (unsigned long) (b)[(i) + 1] << 16) \
    | ( (unsigned long) (b)[(i) + 2] << 8) \
    | ( (unsigned long) (b)[(i) + 3] ); \
}

```

Definition at line 39 of file des.c.

Referenced by des3\_crypt\_ecb(), des\_crypt\_ecb(), des\_setkey(), sha1\_process(), and sha2\_process().

**5.120.1.5 #define PUT\_ULONG\_BE(n, b, i)****Value:**

```

{
    (b) [(i)      ] = (unsigned char) ( (n) >> 24 ); \
    (b) [(i) + 1] = (unsigned char) ( (n) >> 16 ); \
    (b) [(i) + 2] = (unsigned char) ( (n) >> 8 );  \
    (b) [(i) + 3] = (unsigned char) ( (n)          ); \
}

```

Definition at line 49 of file des.c.

Referenced by des3\_crypt\_ecb(), des\_crypt\_ecb(), sha1\_finish(), and sha2\_finish().

#### 5.120.1.6 #define SWAP(a, b) { unsigned long t = a; a = b; b = t; t = 0; }

Definition at line 286 of file des.c.

Referenced by des\_setkey\_dec().

### 5.120.2 Function Documentation

#### 5.120.2.1 void des3\_crypt\_cbc (des3\_context \* ctx, int mode, int length, unsigned char iv[8], unsigned char \* input, unsigned char \* output)

3DES-CBC buffer encryption/decryption

##### Parameters

*ctx* 3DES context  
*mode* DES\_ENCRYPT or DES\_DECRYPT  
*length* length of the input data  
*iv* initialization vector (updated after use)  
*input* buffer holding the input data  
*output* buffer holding the output data

Definition at line 593 of file des.c.

References des3\_crypt\_ecb(), and DES\_ENCRYPT.

Referenced by des\_self\_test(), main(), ssl\_decrypt\_buf(), ssl\_encrypt\_buf(), and x509\_des3\_decrypt().

#### 5.120.2.2 void des3\_crypt\_ecb (des3\_context \* ctx, unsigned char input[8], unsigned char output[8])

3DES-ECB block encryption/decryption

##### Parameters

*ctx* 3DES context  
*input* 64-bit input block  
*output* 64-bit output block

Definition at line 552 of file des.c.

References DES\_FP, DES\_IP, DES\_ROUND, GET\_ULONG\_BE, PUT\_ULONG\_BE, and des3\_context::sk.

Referenced by des3\_crypt\_cbc(), and des\_self\_test().

### 5.120.2.3 `static void des3_set2key (unsigned long esk[96], unsigned long dsk[96], unsigned char key[16]) [static]`

Definition at line 381 of file `des.c`.

References `des_setkey()`.

Referenced by `des3_set2key_dec()`, and `des3_set2key_enc()`.

### 5.120.2.4 `void des3_set2key_dec (des3_context * ctx, unsigned char key[16])`

Triple-DES key schedule (112-bit, decryption).

#### Parameters

*ctx* 3DES context to be initialized

*key* 16-byte secret key

Definition at line 420 of file `des.c`.

References `des3_set2key()`, and `des3_context::sk`.

Referenced by `des_self_test()`.

### 5.120.2.5 `void des3_set2key_enc (des3_context * ctx, unsigned char key[16])`

Triple-DES key schedule (112-bit, encryption).

#### Parameters

*ctx* 3DES context to be initialized

*key* 16-byte secret key

Definition at line 409 of file `des.c`.

References `des3_set2key()`, and `des3_context::sk`.

Referenced by `des_self_test()`.

### 5.120.2.6 `static void des3_set3key (unsigned long esk[96], unsigned long dsk[96], unsigned char key[24]) [static]`

Definition at line 428 of file `des.c`.

References `des_setkey()`.

Referenced by `des3_set3key_dec()`, and `des3_set3key_enc()`.

### 5.120.2.7 `void des3_set3key_dec (des3_context * ctx, unsigned char key[24])`

Triple-DES key schedule (168-bit, decryption).

#### Parameters

*ctx* 3DES context to be initialized

*key* 24-byte secret key

Definition at line 465 of file des.c.

References `des3_set3key()`, and `des3_context::sk`.

Referenced by `des_self_test()`, `ssl_derive_keys()`, and `x509_des3_decrypt()`.

#### 5.120.2.8 `void des3_set3key_enc (des3_context * ctx, unsigned char key[24])`

Triple-DES key schedule (168-bit, encryption).

##### Parameters

*ctx* 3DES context to be initialized

*key* 24-byte secret key

Definition at line 454 of file des.c.

References `des3_set3key()`, and `des3_context::sk`.

Referenced by `des_self_test()`, `main()`, and `ssl_derive_keys()`.

#### 5.120.2.9 `void des_crypt_cbc (des_context * ctx, int mode, int length, unsigned char iv[8], unsigned char * input, unsigned char * output)`

DES-CBC buffer encryption/decryption.

##### Parameters

*ctx* DES context

*mode* DES\_ENCRYPT or DES\_DECRYPT

*length* length of the input data

*iv* initialization vector (updated after use)

*input* buffer holding the input data

*output* buffer holding the output data

Definition at line 505 of file des.c.

References `des_crypt_ecb()`, and `DES_ENCRYPT`.

Referenced by `des_self_test()`, and `main()`.

#### 5.120.2.10 `void des_crypt_ecb (des_context * ctx, unsigned char input[8], unsigned char output[8])`

DES-ECB block encryption/decryption.

##### Parameters

*ctx* DES context

*input* 64-bit input block

*output* 64-bit output block

Definition at line 476 of file des.c.

References DES\_FP, DES\_IP, DES\_ROUND, GET\_ULONG\_BE, PUT\_ULONG\_BE, and des\_context::sk.

Referenced by des\_crypt\_cbc(), and des\_self\_test().

#### 5.120.2.11 int des\_self\_test (int *verbose*)

Definition at line 694 of file des.c.

References buf, des3\_crypt\_cbc(), des3\_crypt\_ecb(), des3\_set2key\_dec(), des3\_set2key\_enc(), des3\_set3key\_dec(), des3\_set3key\_enc(), des3\_test\_buf, des3\_test\_cbc\_dec, des3\_test\_cbc\_enc, des3\_test\_ecb\_dec, des3\_test\_ecb\_enc, des3\_test\_iv, des3\_test\_keys, des\_crypt\_cbc(), des\_crypt\_ecb(), DES\_DECRYPT, des\_setkey\_dec(), des\_setkey\_enc(), and prv.

Referenced by main().

#### 5.120.2.12 static void des\_setkey (unsigned long *SK*[32], unsigned char *key*[8]) [static]

Definition at line 288 of file des.c.

References GET\_ULONG\_BE, LHs, and RHs.

Referenced by des3\_set2key(), des3\_set3key(), des\_setkey\_dec(), and des\_setkey\_enc().

#### 5.120.2.13 void des\_setkey\_dec (des\_context \* *ctx*, unsigned char *key*[8])

DES key schedule (56-bit, decryption).

##### Parameters

*ctx* DES context to be initialized

*key* 8-byte secret key

Definition at line 368 of file des.c.

References des\_setkey(), des\_context::sk, and SWAP.

Referenced by des\_self\_test().

#### 5.120.2.14 void des\_setkey\_enc (des\_context \* *ctx*, unsigned char *key*[8])

DES key schedule (56-bit, encryption).

##### Parameters

*ctx* DES context to be initialized

*key* 8-byte secret key

Definition at line 360 of file des.c.

References des\_setkey(), and des\_context::sk.

Referenced by des\_self\_test(), and main().



### 5.120.3 Variable Documentation

#### 5.120.3.1 `const unsigned char des3_test_buf[8] [static]`

**Initial value:**

```
{  
    0x4E, 0x6F, 0x77, 0x20, 0x69, 0x73, 0x20, 0x74  
}
```

Definition at line 658 of file des.c.

Referenced by des\_self\_test().

#### 5.120.3.2 `const unsigned char des3_test_cbc_dec[3][8] [static]`

**Initial value:**

```
{  
    { 0x12, 0x9F, 0x40, 0xB9, 0xD2, 0x00, 0x56, 0xB3 },  
    { 0x47, 0x0E, 0xFC, 0x9A, 0x6B, 0x8E, 0xE3, 0x93 },  
    { 0xC5, 0xCE, 0xCF, 0x63, 0xEC, 0xEC, 0x51, 0x4C }  
}
```

Definition at line 677 of file des.c.

Referenced by des\_self\_test().

#### 5.120.3.3 `const unsigned char des3_test_cbc_enc[3][8] [static]`

**Initial value:**

```
{  
    { 0x54, 0xF1, 0x5A, 0xF6, 0xEB, 0xE3, 0xA4, 0xB4 },  
    { 0x35, 0x76, 0x11, 0x56, 0x5F, 0xA1, 0x8E, 0x4D },  
    { 0xCB, 0x19, 0x1F, 0x85, 0xD1, 0xED, 0x84, 0x39 }  
}
```

Definition at line 684 of file des.c.

Referenced by des\_self\_test().

#### 5.120.3.4 `const unsigned char des3_test_ecb_dec[3][8] [static]`

**Initial value:**

```
{  
    { 0xCD, 0xD6, 0x4F, 0x2F, 0x94, 0x27, 0xC1, 0x5D },  
    { 0x69, 0x96, 0xC8, 0xFA, 0x47, 0xA2, 0xAB, 0xEB },  
    { 0x83, 0x25, 0x39, 0x76, 0x44, 0x09, 0x1A, 0x0A }  
}
```

Definition at line 663 of file des.c.

Referenced by des\_self\_test().

**5.120.3.5   const unsigned char des3\_test\_ecb\_enc[3][8]   [static]****Initial value:**

```
{
    { 0x6A, 0x2A, 0x19, 0xF4, 0x1E, 0xCA, 0x85, 0x4B },
    { 0x03, 0xE6, 0x9F, 0x5B, 0xFA, 0x58, 0xEB, 0x42 },
    { 0xDD, 0x17, 0xE8, 0xB8, 0xB4, 0x37, 0xD2, 0x32 }
}
```

Definition at line 670 of file des.c.

Referenced by des\_self\_test().

**5.120.3.6   const unsigned char des3\_test\_iv[8]   [static]****Initial value:**

```
{
    0x12, 0x34, 0x56, 0x78, 0x90, 0xAB, 0xCD, 0xEF,
}
```

Definition at line 653 of file des.c.

Referenced by des\_self\_test().

**5.120.3.7   const unsigned char des3\_test\_keys[24]   [static]****Initial value:**

```
{
    0x01, 0x23, 0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF,
    0x23, 0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF, 0x01,
    0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF, 0x01, 0x23
}
```

Definition at line 646 of file des.c.

Referenced by des\_self\_test().

**5.120.3.8   const unsigned long LHs[16]   [static]****Initial value:**

```
{
    0x00000000, 0x00000001, 0x00000100, 0x00000101,
    0x00010000, 0x00010001, 0x00010100, 0x00010101,
    0x01000000, 0x01000001, 0x01000100, 0x01000101,
    0x01010000, 0x01010001, 0x01010100, 0x01010101
}
```

Definition at line 224 of file des.c.

Referenced by des\_setkey().

**5.120.3.9 const unsigned long RHs[16] [static]****Initial value:**

```
{
    0x00000000, 0x01000000, 0x00010000, 0x01010000,
    0x00000100, 0x01000100, 0x00010100, 0x01010100,
    0x00000001, 0x01000001, 0x00010001, 0x01010001,
    0x00000010, 0x01000010, 0x00010010, 0x01010010,
}
```

Definition at line 232 of file des.c.

Referenced by des\_setkey().

**5.120.3.10 const unsigned long SB1[64] [static]****Initial value:**

```
{
    0x01010400, 0x00000000, 0x00010000, 0x01010404,
    0x01010004, 0x00010404, 0x00000004, 0x00010000,
    0x00000400, 0x01010400, 0x01010404, 0x00000400,
    0x01000404, 0x01010004, 0x01000000, 0x00000004,
    0x00000404, 0x01000400, 0x01000400, 0x00010400,
    0x00010400, 0x01010000, 0x01010000, 0x01000404,
    0x00010004, 0x01000004, 0x01000004, 0x00010004,
    0x00000000, 0x00000404, 0x00010404, 0x01000000,
    0x00010000, 0x01010404, 0x00000004, 0x01010000,
    0x01010400, 0x01000000, 0x01000000, 0x00000400,
    0x01010004, 0x00010000, 0x00010400, 0x01000004,
    0x00000400, 0x00000004, 0x01000404, 0x00010404,
    0x01010404, 0x00010004, 0x01010000, 0x01000404,
    0x01000004, 0x00000404, 0x00010404, 0x01010400,
    0x00000404, 0x01000400, 0x01000400, 0x00000000,
    0x00010004, 0x00010400, 0x00000000, 0x01010004
}
```

Definition at line 61 of file des.c.

**5.120.3.11 const unsigned long SB2[64] [static]****Initial value:**

```
{
    0x80108020, 0x80008000, 0x00008000, 0x00108020,
    0x00100000, 0x00000020, 0x80100020, 0x80008020,
    0x80000020, 0x80108020, 0x80108000, 0x80000000,
    0x80008000, 0x00100000, 0x00000020, 0x80100020,
    0x00108000, 0x00100020, 0x80008020, 0x00000000,
    0x80000000, 0x00008000, 0x00108020, 0x80100000,
    0x00100020, 0x80000020, 0x00000000, 0x00108000,
    0x00008020, 0x80108000, 0x80100000, 0x00008020,
    0x00000000, 0x00108020, 0x80100020, 0x00100000,
    0x80008020, 0x80100000, 0x80108000, 0x00008000,
    0x80100000, 0x80008000, 0x00000020, 0x80108020,
    0x00108020, 0x00000020, 0x00008000, 0x80000000,
    0x00008020, 0x80108000, 0x00100000, 0x80000020,
    0x00100020, 0x80008020, 0x80000020, 0x00100020,
}
```

```

    0x00108000, 0x00000000, 0x80008000, 0x00008020,
    0x80000000, 0x80100020, 0x80108020, 0x00108000
}

```

Definition at line 81 of file des.c.

#### 5.120.3.12 const unsigned long SB3[64] [static]

**Initial value:**

```

{
    0x00000208, 0x08020200, 0x00000000, 0x08020008,
    0x08000200, 0x00000000, 0x00020208, 0x08000200,
    0x00020008, 0x08000008, 0x08000008, 0x00020000,
    0x08020208, 0x00020008, 0x08020000, 0x00000208,
    0x08000000, 0x00000008, 0x08020200, 0x00000200,
    0x00020200, 0x08020000, 0x08020008, 0x00020208,
    0x08000208, 0x00020200, 0x00020000, 0x08000208,
    0x00000008, 0x08020208, 0x00000200, 0x08000000,
    0x08020200, 0x08000000, 0x00020008, 0x00000208,
    0x00020000, 0x08020200, 0x08000200, 0x00000000,
    0x00000200, 0x00020008, 0x08020208, 0x08000200,
    0x08000008, 0x00000200, 0x00000000, 0x08020008,
    0x08000208, 0x00020000, 0x08000000, 0x08020208,
    0x00000008, 0x00020208, 0x00020200, 0x08000008,
    0x08020000, 0x08000208, 0x00000208, 0x08020000,
    0x00020208, 0x00000008, 0x08020008, 0x00020200
}

```

Definition at line 101 of file des.c.

#### 5.120.3.13 const unsigned long SB4[64] [static]

**Initial value:**

```

{
    0x00802001, 0x00002081, 0x00002081, 0x00000080,
    0x00802080, 0x00800081, 0x00800001, 0x00002001,
    0x00000000, 0x00802000, 0x00802000, 0x00802081,
    0x00000081, 0x00000000, 0x00800080, 0x00800001,
    0x00000001, 0x00002000, 0x00800000, 0x00802001,
    0x00000080, 0x00800000, 0x00002001, 0x00002080,
    0x00800081, 0x00000001, 0x00002080, 0x00800080,
    0x00002000, 0x00802080, 0x00802081, 0x00000081,
    0x00800080, 0x00800001, 0x00802000, 0x00802081,
    0x00000081, 0x00000000, 0x00000000, 0x00802000,
    0x00002080, 0x00800080, 0x00800081, 0x00000001,
    0x00802001, 0x00002081, 0x00002081, 0x00000080,
    0x00802081, 0x00000081, 0x00000001, 0x00002000,
    0x00800001, 0x00002001, 0x00802080, 0x00800081,
    0x00002001, 0x00002080, 0x00800000, 0x00802001,
    0x00000080, 0x00800000, 0x00002000, 0x00802080
}

```

Definition at line 121 of file des.c.

#### 5.120.3.14 const unsigned long SB5[64] [static]

**Initial value:**

```

{
    0x00000100, 0x02080100, 0x02080000, 0x42000100,
    0x00080000, 0x00000100, 0x40000000, 0x02080000,
    0x40080100, 0x00080000, 0x02000100, 0x40080100,
    0x42000100, 0x42080000, 0x00080100, 0x40000000,
    0x02000000, 0x40080000, 0x40080000, 0x00000000,
    0x40000100, 0x42080100, 0x42080100, 0x02000100,
    0x42080000, 0x40000100, 0x00000000, 0x42000000,
    0x02080100, 0x02000000, 0x42000000, 0x00080100,
    0x00080000, 0x42000100, 0x00000100, 0x02000000,
    0x40000000, 0x02080000, 0x42000100, 0x40080100,
    0x02000100, 0x40000000, 0x42080000, 0x02080100,
    0x40080100, 0x00000100, 0x02000000, 0x42080000,
    0x42080100, 0x00080100, 0x42000000, 0x42080100,
    0x02080000, 0x00000000, 0x40080000, 0x42000000,
    0x00080100, 0x02000100, 0x40000100, 0x00080000,
    0x00000000, 0x40080000, 0x02080100, 0x40000100
}

```

Definition at line 141 of file des.c.

### 5.120.3.15 const unsigned long SB6[64] [static]

Initial value:

```

{
    0x20000010, 0x20400000, 0x00004000, 0x20404010,
    0x20400000, 0x00000010, 0x20404010, 0x00400000,
    0x20004000, 0x00404010, 0x00400000, 0x20000010,
    0x00400010, 0x20004000, 0x20000000, 0x00004010,
    0x00000000, 0x00400010, 0x20004010, 0x00004000,
    0x00404000, 0x20004010, 0x00000010, 0x20400010,
    0x20400010, 0x00000000, 0x00404010, 0x20404000,
    0x00004010, 0x00404000, 0x20404000, 0x20000000,
    0x20004000, 0x00000010, 0x20400010, 0x00404000,
    0x20404010, 0x00400000, 0x00004010, 0x20000010,
    0x00400000, 0x20004000, 0x20000000, 0x00004010,
    0x20000010, 0x20404010, 0x00404000, 0x20400000,
    0x00404010, 0x20404000, 0x00000000, 0x20400010,
    0x00000010, 0x00004000, 0x20400000, 0x00404010,
    0x00004000, 0x00400010, 0x20004010, 0x00000000,
    0x20404000, 0x20000000, 0x00400010, 0x20004010
}

```

Definition at line 161 of file des.c.

### 5.120.3.16 const unsigned long SB7[64] [static]

Initial value:

```

{
    0x00200000, 0x04200002, 0x04000802, 0x00000000,
    0x00000800, 0x04000802, 0x00200802, 0x04200800,
    0x04200802, 0x00200000, 0x00000000, 0x04000002,
    0x00000002, 0x04000000, 0x04200002, 0x00000802,
    0x04000800, 0x00200802, 0x00200002, 0x04000800,
    0x04000002, 0x04200000, 0x04200800, 0x00200002,
    0x04200000, 0x00000800, 0x00000802, 0x04200802,
    0x00200800, 0x00000002, 0x04000000, 0x00200800,

```

```

    0x04000000, 0x00200800, 0x00200000, 0x04000802,
    0x04000802, 0x04200002, 0x04200002, 0x00000002,
    0x00200002, 0x04000000, 0x04000800, 0x00200000,
    0x04200800, 0x00000802, 0x00200802, 0x04200800,
    0x00000802, 0x04000002, 0x04200802, 0x04200000,
    0x00200800, 0x00000000, 0x00000002, 0x04200802,
    0x00000000, 0x00200802, 0x04200000, 0x00000800,
    0x04000002, 0x04000800, 0x00000800, 0x00200002
}

```

Definition at line 181 of file des.c.

### 5.120.3.17 const unsigned long SB8[64] [static]

**Initial value:**

```

{
    0x10001040, 0x00001000, 0x00040000, 0x10041040,
    0x10000000, 0x10001040, 0x00000040, 0x10000000,
    0x00040040, 0x10040000, 0x10041040, 0x00041000,
    0x10041000, 0x00041040, 0x00001000, 0x00000040,
    0x10040000, 0x10000040, 0x10001000, 0x00001040,
    0x00041000, 0x00040040, 0x10040040, 0x10041000,
    0x00001040, 0x00000000, 0x00000000, 0x10040040,
    0x10000040, 0x10001000, 0x00041040, 0x00040000,
    0x00041040, 0x00040000, 0x10041000, 0x00001000,
    0x00000040, 0x10040040, 0x00001000, 0x00041040,
    0x10001000, 0x00000040, 0x10000040, 0x10040000,
    0x10040040, 0x10000000, 0x00040000, 0x10001040,
    0x00000000, 0x10041040, 0x00040040, 0x10000040,
    0x10040000, 0x10001000, 0x10001040, 0x00000000,
    0x10041040, 0x00041000, 0x00041000, 0x00001040,
    0x00001040, 0x00040040, 0x10000000, 0x10041000
}

```

Definition at line 201 of file des.c.

## 5.121 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/dhm.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/dhm.h"
#include <string.h>
```

### Defines

- #define [DHM\\_MPI\\_EXPORT](#)(X, n)

### Functions

- static [int dhm\\_read\\_bignum](#) ([mpi](#) \*X, unsigned char \*\*p, unsigned char \*end)
- [int dhm\\_read\\_params](#) ([dhm\\_context](#) \*ctx, unsigned char \*\*p, unsigned char \*end)  
*Parse the ServerKeyExchange parameters.*
- [int dhm\\_make\\_params](#) ([dhm\\_context](#) \*ctx, [int](#) x\_size, unsigned char \*output, [int](#) \*olen, [int](#)(\*f\_rng)(void \*), void \*p\_rng)  
*Setup and write the ServerKeyExchange parameters.*
- [int dhm\\_read\\_public](#) ([dhm\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*Import the peer's public value  $G^Y$ .*
- [int dhm\\_make\\_public](#) ([dhm\\_context](#) \*ctx, [int](#) x\_size, unsigned char \*output, [int](#) olen, [int](#)(\*f\_rng)(void \*), void \*p\_rng)  
*Create own private value X and export  $G^X$ .*
- [int dhm\\_calc\\_secret](#) ([dhm\\_context](#) \*ctx, unsigned char \*output, [int](#) \*olen)  
*Derive and export the shared secret  $(G^Y)^X \bmod P$ .*
- void [dhm\\_free](#) ([dhm\\_context](#) \*ctx)
- [int dhm\\_self\\_test](#) ([int](#) verbose)  
*Checkup routine.*

### 5.121.1 Define Documentation

#### 5.121.1.1 #define DHM\_MPI\_EXPORT(X, n)

##### Value:

```
MPI_CHK( mpi_write_binary( X, p + 2, n ) ); \
    *p++ = (unsigned char)( n >> 8 ); \
    *p++ = (unsigned char)( n ); p += n;
```

Referenced by [dhm\\_make\\_params](#)( ).

## 5.121.2 Function Documentation

### 5.121.2.1 `int dhm_calc_secret (dhm_context * ctx, unsigned char * output, int * olen)`

Derive and export the shared secret  $(G^Y)^X \bmod P$ .

#### Parameters

*ctx* DHM context  
*output* destination buffer  
*olen* number of chars written

#### Returns

0 if successful, or an XYSSL\_ERR\_DHM\_XXX error code

Definition at line 208 of file dhm.c.

References dhm\_context::GY, dhm\_context::K, MPI\_CHK, mpi\_exp\_mod(), mpi\_size(), mpi\_write\_binary(), dhm\_context::P, dhm\_context::RP, dhm\_context::X, XYSSL\_ERR\_DHM\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_DHM\_CALC\_SECRET\_FAILED.

Referenced by main(), ssl\_parse\_client\_key\_exchange(), and ssl\_write\_client\_key\_exchange().

### 5.121.2.2 `void dhm_free (dhm_context * ctx)`

Definition at line 234 of file dhm.c.

References dhm\_context::G, dhm\_context::GX, dhm\_context::GY, dhm\_context::K, mpi\_free(), dhm\_context::P, dhm\_context::RP, and dhm\_context::X.

Referenced by main(), and ssl\_free().

### 5.121.2.3 `int dhm_make_params (dhm_context * ctx, int s_size, unsigned char * output, int * olen, int(*) (void *) f_rng, void * p_rng)`

Setup and write the ServerKeyExchange parameters.

#### Parameters

*ctx* DHM context  
*s\_size* private value size in bits  
*output* destination buffer  
*olen* number of chars written  
*f\_rng* RNG function  
*p\_rng* RNG parameter

#### Note

This function assumes that *ctx*->P and *ctx*->G have already been properly set (for example using mpi\_read\_string or mpi\_read\_binary).

#### Returns

0 if successful, or an XYSSL\_ERR\_DHM\_XXX error code



Definition at line 93 of file dhm.c.

References DHM\_MPI\_EXPORT, dhm\_context::G, dhm\_context::GX, dhm\_context::len, MPI\_CHK, mpi\_cmp\_mpi(), mpi\_exp\_mod(), mpi\_grow(), mpi\_lset(), mpi\_shift\_r(), mpi\_size(), dhm\_context::P, mpi::p, dhm\_context::RP, dhm\_context::X, and XYSSL\_ERR\_DHM\_MAKE\_PARAMS\_FAILED.

Referenced by main(), and ssl\_write\_server\_key\_exchange().

#### 5.121.2.4 int dhm\_make\_public(dhm\_context \* ctx, int s\_size, unsigned char \* output, int olen, int(\*) (void \*) f\_rng, void \* p\_rng)

Create own private value X and export  $G^X$ .

##### Parameters

*ctx* DHM context  
*s\_size* private value size in bits  
*output* destination buffer  
*olen* must be equal to ctx->P.len  
*f\_rng* RNG function  
*p\_rng* RNG parameter

##### Returns

0 if successful, or an XYSSL\_ERR\_DHM\_XXX error code

Definition at line 167 of file dhm.c.

References dhm\_context::G, dhm\_context::GX, dhm\_context::len, MPI\_CHK, mpi\_cmp\_mpi(), mpi\_exp\_mod(), mpi\_grow(), mpi\_lset(), mpi\_shift\_r(), mpi\_write\_binary(), dhm\_context::P, mpi::p, dhm\_context::RP, dhm\_context::X, XYSSL\_ERR\_DHM\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_DHM\_MAKE\_PUBLIC\_FAILED.

Referenced by main(), and ssl\_write\_client\_key\_exchange().

#### 5.121.2.5 static int dhm\_read\_bignum(mpi \* X, unsigned char \*\* p, unsigned char \* end) [static]

Definition at line 37 of file dhm.c.

References mpi\_read\_binary(), XYSSL\_ERR\_DHM\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_DHM\_READ\_PARAMS\_FAILED.

Referenced by dhm\_read\_params().

#### 5.121.2.6 int dhm\_read\_params(dhm\_context \* ctx, unsigned char \*\* p, unsigned char \* end)

Parse the ServerKeyExchange parameters.

##### Parameters

*ctx* DHM context  
*p* &(start of input buffer)  
*end* end of buffer

**Returns**

0 if successful, or an XYSSL\_ERR\_DHM\_XXX error code

Definition at line 63 of file dhm.c.

References dhm\_read\_bignum(), dhm\_context::G, dhm\_context::GY, dhm\_context::len, mpi\_size(), dhm\_context::P, and XYSSL\_ERR\_DHM\_BAD\_INPUT\_DATA.

Referenced by main(), and ssl\_parse\_server\_key\_exchange().

**5.121.2.7 int dhm\_read\_public (dhm\_context \* *ctx*, unsigned char \* *input*, int *ilen*)**

Import the peer's public value  $G^Y$ .

**Parameters**

*ctx* DHM context

*input* input buffer

*ilen* size of buffer

**Returns**

0 if successful, or an XYSSL\_ERR\_DHM\_XXX error code

Definition at line 150 of file dhm.c.

References dhm\_context::GY, dhm\_context::len, mpi\_read\_binary(), XYSSL\_ERR\_DHM\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_DHM\_READ\_PUBLIC\_FAILED.

Referenced by main(), and ssl\_parse\_client\_key\_exchange().

**5.121.2.8 int dhm\_self\_test (int *verbose*)**

Checkup routine.

**Returns**

0 if successful, or 1 if the test failed

Definition at line 246 of file dhm.c.

## 5.122 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/havege.c File Reference

```
#include <string.h>
#include <time.h>
#include "xyssl/config.h"
#include "xyssl/havege.h"
#include "xyssl/timing.h"
```

### Defines

- `#define SWAP(X, Y) { int *T = X; X = Y; Y = T; }`
- `#define TST1_ENTER if( PTEST & 1 ) { PTEST ^= 3; PTEST >>= 1;`
- `#define TST2_ENTER if( PTEST & 1 ) { PTEST ^= 3; PTEST >>= 1;`
- `#define TST1_LEAVE U1++; }`
- `#define TST2_LEAVE U2++; }`
- `#define ONE_ITERATION`

### Functions

- `static void havege_fill (havege_state *hs)`
- `void havege_init (havege_state *hs)`  
*HAVEGE initialization.*
- `int havege_rand (void *p_rng)`  
*HAVEGE rand function.*

### 5.122.1 Define Documentation

#### 5.122.1.1 `#define ONE_ITERATION`

Definition at line 60 of file havege.c.

Referenced by `havege_fill()`.

#### 5.122.1.2 `#define SWAP(X, Y) { int *T = X; X = Y; Y = T; }`

Definition at line 52 of file havege.c.

#### 5.122.1.3 `#define TST1_ENTER if( PTEST & 1 ) { PTEST ^= 3; PTEST >>= 1;`

Definition at line 54 of file havege.c.

#### 5.122.1.4 `#define TST1_LEAVE U1++; }`

Definition at line 57 of file havege.c.

**5.122.1.5 #define TST2\_ENTER if( PTEST & 1 ) { PTEST ^= 3; PTEST >>= 1;**

Definition at line 55 of file havege.c.

**5.122.1.6 #define TST2\_LEAVE U2++; }**

Definition at line 58 of file havege.c.

**5.122.2 Function Documentation****5.122.2.1 static void havege\_fill (havege\_state \* *hs*) [static]**

Definition at line 154 of file havege.c.

References COLLECT\_SIZE, havege\_state::offset, ONE\_ITERATION, havege\_state::PT1, havege\_state::PT2, and havege\_state::WALK.

Referenced by havege\_init(), and havege\_rand().

**5.122.2.2 void havege\_init (havege\_state \* *hs*)**

HAVEGE initialization.

**Parameters**

*hs* HAVEGE state to be initialized

Definition at line 188 of file havege.c.

References havege\_fill().

Referenced by generate\_AES\_key(), generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), initiate\_migration\_process(), main(), and ssl\_test().

**5.122.2.3 int havege\_rand (void \* *p\_rng*)**

HAVEGE rand function.

**Parameters**

*rng\_st* points to an HAVEGE state

**Returns**

A random int

Definition at line 198 of file havege.c.

References COLLECT\_SIZE, havege\_fill(), havege\_state::offset, and havege\_state::pool.

Referenced by generate\_AES\_key(), generate\_RSA\_keys\_ciphertext(), generate\_RSA\_keys\_plaintext(), initiate\_migration\_process(), main(), and ssl\_test().

## **5.123   /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md2.c File Reference**

```
#include "xyssl/config.h"
```

## **5.124    /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md4.c File Reference**

```
#include "xyssl/config.h"
```

## 5.125 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md5.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/md5.h"
#include <string.h>
#include <stdio.h>
```

### Defines

- #define [GET\\_ULONG\\_LE](#)(n, b, i)
- #define [PUT\\_ULONG\\_LE](#)(n, b, i)
- #define [S](#)(x, n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))
- #define [P](#)(a, b, c, d, k, s, t)
- #define [F](#)(x, y, z) (z ^ (x & (y ^ z)))
- #define [F](#)(x, y, z) (y ^ (z & (x ^ y)))
- #define [F](#)(x, y, z) (x ^ y ^ z)
- #define [F](#)(x, y, z) (y ^ (x | ~z))

### Functions

- void [md5\\_starts](#) ([md5\\_context](#) \*ctx)  
*MD5 context setup.*
- static void [md5\\_process](#) ([md5\\_context](#) \*ctx, unsigned char data[64])
- void [md5\\_update](#) ([md5\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*MD5 process buffer.*
- void [md5\\_finish](#) ([md5\\_context](#) \*ctx, unsigned char output[16])  
*MD5 final digest.*
- void [md5](#) (unsigned char \*input, [int](#) ilen, unsigned char output[16])  
*Output = MD5( input buffer ).*
- [int](#) [md5\\_file](#) (char \*path, unsigned char output[16])  
*Output = MD5( file contents ).*
- void [md5\\_hmac\\_starts](#) ([md5\\_context](#) \*ctx, unsigned char \*key, [int](#) keylen)  
*MD5 HMAC context setup.*
- void [md5\\_hmac\\_update](#) ([md5\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*MD5 HMAC process buffer.*
- void [md5\\_hmac\\_finish](#) ([md5\\_context](#) \*ctx, unsigned char output[16])  
*MD5 HMAC final digest.*
- void [md5\\_hmac](#) (unsigned char \*key, [int](#) keylen, unsigned char \*input, [int](#) ilen, unsigned char output[16])

*Output = HMAC-MD5( hmac key, input buffer ).*

- `int md5_self_test` (`int verbose`)

*Checkup routine.*

## Variables

- static const unsigned char `md5_padding` [64]
- static unsigned char `md5_test_buf` [7][81]
- static const `int md5_test_buflen` [7]
- static const unsigned char `md5_test_sum` [7][16]
- static unsigned char `md5_hmac_test_key` [7][26]
- static const `int md5_hmac_test_keylen` [7]
- static unsigned char `md5_hmac_test_buf` [7][74]
- static const `int md5_hmac_test_buflen` [7]
- static const unsigned char `md5_hmac_test_sum` [7][16]

## 5.125.1 Define Documentation

**5.125.1.1** `#define F(x, y, z) (y ^ (x | ~z))`

**5.125.1.2** `#define F(x, y, z) (x ^ y ^ z)`

**5.125.1.3** `#define F(x, y, z) (y ^ (z & (x ^ y)))`

**5.125.1.4** `#define F(x, y, z) (z ^ (x & (y ^ z)))`

Referenced by `sha2_process()`, and `sha4_process()`.

**5.125.1.5** `#define GET_ULONG_LE(n, b, i)`

**Value:**

```
{
    (n) = ( (unsigned long) (b)[(i)      ]          ) \
    | ( (unsigned long) (b)[(i) + 1] << 8 )          \
    | ( (unsigned long) (b)[(i) + 2] << 16 )          \
    | ( (unsigned long) (b)[(i) + 3] << 24 );        \
}
```

Definition at line 39 of file `md5.c`.

**5.125.1.6** `#define P(a, b, c, d, k, s, t)`

**Value:**

```
{
    a += F(b, c, d) + X[k] + t; a = S(a, s) + b; \
}
```

Referenced by `main()`, `md5_process()`, `rsa_decryption()`, `sha1_process()`, `sha2_process()`, and `sha4_process()`.



**5.125.1.7 #define PUT\_ULONG\_LE(n, b, i)****Value:**

```

{
    (b)[(i)] = (unsigned char) ((n) >> 0);
    (b)[(i) + 1] = (unsigned char) ((n) >> 8);
    (b)[(i) + 2] = (unsigned char) ((n) >> 16);
    (b)[(i) + 3] = (unsigned char) ((n) >> 24);
}

```

Definition at line 49 of file md5.c.

**5.125.1.8 #define S(x, n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))****5.125.2 Function Documentation****5.125.2.1 void md5 (unsigned char \* *input*, int *ilen*, unsigned char *output*[16])**Output = MD5( *input* buffer ).**Parameters**

*input* buffer holding the data  
*ilen* length of the input data  
*output* MD5 checksum result

Definition at line 278 of file md5.c.

References md5\_finish(), md5\_starts(), and md5\_update().

Referenced by initiate\_migration\_process(), main(), md5\_hmac\_starts(), md5\_self\_test(), reply\_migration\_process(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_md5(), ssl\_parse\_finished(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_finished(), ssl\_write\_server\_key\_exchange(), and x509\_hash().

**5.125.2.2 int md5\_file (char \* *path*, unsigned char *output*[16])**Output = MD5( *file* contents ).**Parameters**

*path* input file name  
*output* MD5 checksum result

**Returns**

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 292 of file md5.c.

References buf, f, md5\_finish(), md5\_starts(), and md5\_update().

Referenced by md5\_wrapper().

**5.125.2.3 void md5\_finish (md5\_context \* *ctx*, unsigned char *output*[16])**

MD5 final digest.

**Parameters**

*ctx* MD5 context

*output* MD5 checksum result

Definition at line 250 of file md5.c.

References md5\_padding, md5\_update(), PUT\_ULONG\_LE, md5\_context::state, and md5\_context::total.

Referenced by md5(), md5\_file(), md5\_hmac\_finish(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_md5(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_server\_key\_exchange(), and x509\_des3\_decrypt().

**5.125.2.4 void md5\_hmac (unsigned char \* *key*, int *keylen*, unsigned char \* *input*, int *ilen*, unsigned char *output*[16])**

Output = HMAC-MD5( hmac key, input buffer ).

**Parameters**

*key* HMAC secret key

*keylen* length of the HMAC key

*input* buffer holding the data

*ilen* length of the input data

*output* HMAC-MD5 result

Definition at line 378 of file md5.c.

References md5\_hmac\_finish(), md5\_hmac\_starts(), and md5\_hmac\_update().

Referenced by ssl\_decrypt\_buf(), ssl\_encrypt\_buf(), and tls1\_prf().

**5.125.2.5 void md5\_hmac\_finish (md5\_context \* *ctx*, unsigned char *output*[16])**

MD5 HMAC final digest.

**Parameters**

*ctx* HMAC context

*output* MD5 HMAC checksum result

Definition at line 362 of file md5.c.

References md5\_finish(), md5\_starts(), md5\_update(), and md5\_context::opad.

Referenced by md5\_hmac(), and md5\_self\_test().

**5.125.2.6 void md5\_hmac\_starts (md5\_context \* *ctx*, unsigned char \* *key*, int *keylen*)**

MD5 HMAC context setup.

**Parameters**

*ctx* HMAC context to be initialized

*key* HMAC secret key

*keylen* length of the HMAC key

Definition at line 324 of file md5.c.

References md5\_context::ipad, md5(), md5\_starts(), md5\_update(), and md5\_context::opad.

Referenced by md5\_hmac(), and md5\_self\_test().

**5.125.2.7 void md5\_hmac\_update (md5\_context \* *ctx*, unsigned char \* *input*, int *ilen*)**

MD5 HMAC process buffer.

**Parameters**

*ctx* HMAC context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 354 of file md5.c.

References md5\_update().

Referenced by md5\_hmac(), and md5\_self\_test().

**5.125.2.8 static void md5\_process (md5\_context \* *ctx*, unsigned char *data*[64]) [static]**

Definition at line 72 of file md5.c.

References GET\_ULONGLONG\_LE, P, and md5\_context::state.

Referenced by md5\_update().

**5.125.2.9 int md5\_self\_test (int *verbose*)**

Checksum routine.

**Returns**

0 if successful, or 1 if the test failed

Definition at line 495 of file md5.c.

References buf, md5(), md5\_hmac\_finish(), md5\_hmac\_starts(), md5\_hmac\_test\_buf, md5\_hmac\_test\_buflen, md5\_hmac\_test\_key, md5\_hmac\_test\_keylen, md5\_hmac\_test\_sum, md5\_hmac\_update(), md5\_test\_buf, md5\_test\_buflen, and md5\_test\_sum.

Referenced by main().

### 5.125.2.10 void md5\_starts (md5\_context \* ctx)

MD5 context setup.

#### Parameters

*ctx* context to be initialized

Definition at line 61 of file md5.c.

References md5\_context::state, and md5\_context::total.

Referenced by md5(), md5\_file(), md5\_hmac\_finish(), md5\_hmac\_starts(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_init(), ssl\_mac\_md5(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_server\_key\_exchange(), and x509\_des3\_decrypt().

### 5.125.2.11 void md5\_update (md5\_context \* ctx, unsigned char \* input, int ilen)

MD5 process buffer.

#### Parameters

*ctx* MD5 context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 198 of file md5.c.

References md5\_context::buffer, md5\_process(), and md5\_context::total.

Referenced by md5(), md5\_file(), md5\_finish(), md5\_hmac\_finish(), md5\_hmac\_starts(), md5\_hmac\_update(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_md5(), ssl\_parse\_client\_hello(), ssl\_parse\_server\_key\_exchange(), ssl\_read\_record(), ssl\_write\_record(), ssl\_write\_server\_key\_exchange(), and x509\_des3\_decrypt().

## 5.125.3 Variable Documentation

### 5.125.3.1 unsigned char md5\_hmac\_test\_buf[7][74] [static]

Initial value:

```
{
    { "Hi There" },
    { "what do ya want for nothing?" },
    { "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD" },
    { "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD" },
    { "Test With Truncation" },
    { "Test Using Larger Than Block-Size Key - Hash Key First" },
```

```

    { "Test Using Larger Than Block-Size Key and Larger"
      " Than One Block-Size Data" }
}

```

Definition at line 449 of file md5.c.

Referenced by md5\_self\_test().

### 5.125.3.2 `const int md5_hmac_test_buflen[7] [static]`

**Initial value:**

```
{
    8, 28, 50, 50, 20, 54, 73
}
```

Definition at line 469 of file md5.c.

Referenced by md5\_self\_test().

### 5.125.3.3 unsigned char md5\_hmac\_test\_key[7][26] [static]

**Initial value:**

[illegible]

Definition at line 432 of file md5.c.

Referenced by md5\_self\_test().

#### 5.125.3.4 `const int md5_hmac_test_keylen[7] [static]`

**Initial value:**

$$\{16, 4, 16, 25, 16, 80, 80\}$$

Definition at line 444 of file md5.c.

Referenced by md5\_self\_test().

### 5.125.3.5 `const unsigned char md5_hmac_test_sum[7][16] [static]`

**Initial value:**

```
{
    { 0x92, 0x94, 0x72, 0x7A, 0x36, 0x38, 0xBB, 0x1C,
      0x13, 0xF4, 0x8E, 0xF8, 0x15, 0x8B, 0xFC, 0x9D },
    { 0x75, 0x0C, 0x78, 0x3E, 0x6A, 0xB0, 0xB5, 0x03,
      0xEA, 0xA8, 0x6E, 0x31, 0x0A, 0x5D, 0xB7, 0x38 },
    { 0x56, 0xBE, 0x34, 0x52, 0x1D, 0x14, 0x4C, 0x88,
      0xDB, 0xB8, 0xC7, 0x33, 0xF0, 0xE8, 0xB3, 0xF6 },
    { 0x69, 0x7E, 0xAF, 0x0A, 0xCA, 0x3A, 0x3A, 0xEA,
      0x3A, 0x75, 0x16, 0x47, 0x46, 0xFF, 0xAA, 0x79 },
    { 0x56, 0x46, 0x1E, 0xF2, 0x34, 0x2E, 0xDC, 0x00,
      0xF9, 0xBA, 0xB9, 0x95 },
    { 0x6B, 0x1A, 0xB7, 0xFE, 0x4B, 0xD7, 0xBF, 0x8F,
      0x0B, 0x62, 0xE6, 0xCE, 0x61, 0xB9, 0xD0, 0xCD },
    { 0x6F, 0x63, 0x0F, 0xAD, 0x67, 0xCD, 0xA0, 0xEE,
      0x1F, 0xB1, 0xF5, 0x62, 0xDB, 0x3A, 0xA5, 0x3E }
}
```

Definition at line 474 of file md5.c.

Referenced by md5\_self\_test().

### 5.125.3.6 `const unsigned char md5_padding[64] [static]`

**Initial value:**

```
{
    0x80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
}
```

Definition at line 239 of file md5.c.

Referenced by md5\_finish().

### 5.125.3.7 `unsigned char md5_test_buf[7][81] [static]`

**Initial value:**

```
{
    { "" },
    { "a" },
    { "abc" },
    { "message digest" },
    { "abcdefghijklmnopqrstuvwxyz" },
    { "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789" },
    { "12345678901234567890123456789012345678901234567890123456789012"
      "345678901234567890" }
}
```

Definition at line 394 of file md5.c.

Referenced by md5\_self\_test().

### 5.125.3.8 `const int md5_test_bufllen[7] [static]`

**Initial value:**

```
{
    0, 1, 3, 14, 26, 62, 80
}
```

Definition at line 406 of file md5.c.

Referenced by md5\_self\_test().

### 5.125.3.9 const unsigned char md5\_test\_sum[7][16] [static]

**Initial value:**

```
{
    { 0xD4, 0x1D, 0x8C, 0xD9, 0x8F, 0x00, 0xB2, 0x04,
      0xE9, 0x80, 0x09, 0x98, 0xEC, 0xF8, 0x42, 0x7E },
    { 0x0C, 0xC1, 0x75, 0xB9, 0xC0, 0xF1, 0xB6, 0xA8,
      0x31, 0xC3, 0x99, 0xE2, 0x69, 0x77, 0x26, 0x61 },
    { 0x90, 0x01, 0x50, 0x98, 0x3C, 0xD2, 0x4F, 0xB0,
      0xD6, 0x96, 0x3F, 0x7D, 0x28, 0xE1, 0x7F, 0x72 },
    { 0xF9, 0x6B, 0x69, 0x7D, 0x7C, 0xB7, 0x93, 0x8D,
      0x52, 0x5A, 0x2F, 0x31, 0xAA, 0xF1, 0x61, 0xD0 },
    { 0xC3, 0xFC, 0xD3, 0xD7, 0x61, 0x92, 0xE4, 0x00,
      0x7D, 0xFB, 0x49, 0x6C, 0xCA, 0x67, 0xE1, 0x3B },
    { 0xD1, 0x74, 0xAB, 0x98, 0xD2, 0x77, 0xD9, 0xF5,
      0xA5, 0x61, 0x1C, 0x2C, 0x9F, 0x41, 0x9D, 0x9F },
    { 0x57, 0xED, 0xF4, 0xA2, 0x2B, 0xE3, 0xC9, 0x55,
      0xAC, 0x49, 0xDA, 0x2E, 0x21, 0x07, 0xB6, 0x7A }
}
```

Definition at line 411 of file md5.c.

Referenced by md5\_self\_test().

## 5.126 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/net.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/net.h"
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <sys/time.h>
#include <unistd.h>
#include <signal.h>
#include <fcntl.h>
#include <netdb.h>
#include <errno.h>
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
```

### Functions

- static unsigned short [net\\_htons](#) (int port)
- [int net\\_connect](#) (int \*fd, char \*host, int port)  
*Initiate a TCP connection with host:port.*
- [int net\\_bind](#) (int \*fd, char \*bind\_ip, int port)  
*Create a listening socket on bind\_ip:port. If bind\_ip == NULL, all interfaces are binded.*
- static [int net\\_is\\_blocking](#) (void)
- [int net\\_accept](#) (int bind\_fd, int \*client\_fd, void \*client\_ip)  
*Accept a connection from a remote client.*
- [int net\\_set\\_block](#) (int fd)  
*Set the socket blocking.*
- [int net\\_set\\_nonblock](#) (int fd)  
*Set the socket non-blocking.*
- void [net\\_usleep](#) (unsigned long usec)  
*Portable usleep helper.*
- [int net\\_recv](#) (void \*ctx, unsigned char \*buf, int len)



*Read at most 'len' characters. len is updated to reflect the actual number of characters read.*

- `int net_send` (`void *ctx`, `unsigned char *buf`, `int len`)

*Write at most 'len' characters. len is updated to reflect the number of characters `_not_` written.*

- `void net_close` (`int fd`)

*Gracefully shutdown the connection.*

## 5.126.1 Function Documentation

### 5.126.1.1 `int net_accept` (`int bind_fd`, `int * client_fd`, `void * client_ip`)

Accept a connection from a remote client.

#### Returns

0 if successful, `XYSSL_ERR_NET_ACCEPT_FAILED`, or `XYSSL_ERR_NET_WOULD_BLOCK` if `bind_fd` was set to non-blocking and `accept()` is blocking.

Definition at line 212 of file `net.c`.

References `int`, `net_is_blocking()`, `XYSSL_ERR_NET_ACCEPT_FAILED`, and `XYSSL_ERR_NET_TRY_AGAIN`.

Referenced by `main()`, and `ssl_test()`.

### 5.126.1.2 `int net_bind` (`int * fd`, `char * bind_ip`, `int port`)

Create a listening socket on `bind_ip:port`. If `bind_ip == NULL`, all interfaces are binded.

#### Returns

0 if successful, or one of: `XYSSL_ERR_NET_SOCKET_FAILED`, `XYSSL_ERR_NET_BIND_FAILED`, `XYSSL_ERR_NET_LISTEN_FAILED`

Definition at line 126 of file `net.c`.

References `net_htons()`, `SOCKET_ERROR`, `XYSSL_ERR_NET_BIND_FAILED`, `XYSSL_ERR_NET_LISTEN_FAILED`, and `XYSSL_ERR_NET_SOCKET_FAILED`.

Referenced by `main()`, and `ssl_test()`.

### 5.126.1.3 `void net_close` (`int fd`)

Gracefully shutdown the connection.

Definition at line 338 of file `net.c`.

Referenced by `main()`, and `ssl_test()`.

### 5.126.1.4 `int net_connect` (`int * fd`, `char * host`, `int port`)

Initiate a TCP connection with `host:port`.

**Returns**

0 if successful, or one of: XYSSL\_ERR\_NET\_SOCKET\_FAILED, XYSSL\_ERR\_NET\_UNKNOWN\_HOST, XYSSL\_ERR\_NET\_CONNECT\_FAILED

Definition at line 81 of file net.c.

References net\_htons(), SOCKET\_ERROR, XYSSL\_ERR\_NET\_CONNECT\_FAILED, XYSSL\_ERR\_NET\_SOCKET\_FAILED, and XYSSL\_ERR\_NET\_UNKNOWN\_HOST.

Referenced by main(), and ssl\_test().

**5.126.1.5 static unsigned short net\_htons (int *port*) [static]**

Definition at line 67 of file net.c.

References buf.

Referenced by net\_bind(), and net\_connect().

**5.126.1.6 static int net\_is\_blocking (void) [static]**

Definition at line 190 of file net.c.

Referenced by net\_accept(), net\_recv(), and net\_send().

**5.126.1.7 int net\_recv (void \* *ctx*, unsigned char \* *buf*, int *len*)**

Read at most 'len' characters. len is updated to reflect the actual number of characters read.

**Returns**

This function returns the number of bytes received, or a negative error code; XYSSL\_ERR\_NET\_TRY\_AGAIN indicates read() is blocking.

Definition at line 277 of file net.c.

References net\_is\_blocking(), XYSSL\_ERR\_NET\_CONN\_RESET, XYSSL\_ERR\_NET\_RECV\_FAILED, and XYSSL\_ERR\_NET\_TRY\_AGAIN.

Referenced by main(), and ssl\_test().

**5.126.1.8 int net\_send (void \* *ctx*, unsigned char \* *buf*, int *len*)**

Write at most 'len' characters. len is updated to reflect the number of characters \_not\_ written.

**Returns**

This function returns the number of bytes sent, or a negative error code; XYSSL\_ERR\_NET\_TRY\_AGAIN indicates write() is blocking.

Definition at line 309 of file net.c.

References net\_is\_blocking(), XYSSL\_ERR\_NET\_CONN\_RESET, XYSSL\_ERR\_NET\_SEND\_FAILED, and XYSSL\_ERR\_NET\_TRY\_AGAIN.

Referenced by main(), and ssl\_test().

**5.126.1.9 int net\_set\_block (int *fd*)**

Set the socket blocking.

**Returns**

0 if successful, or a non-zero error code

Definition at line 243 of file net.c.

**5.126.1.10 int net\_set\_nonblock (int *fd*)**

Set the socket non-blocking.

**Returns**

0 if successful, or a non-zero error code

Definition at line 253 of file net.c.

Referenced by ssl\_test().

**5.126.1.11 void net\_usleep (unsigned long *usec*)**

Portable usleep helper.

**Note**

Real amount of time slept will not be less than select()'s timeout granularity (typically, 10ms).

Definition at line 266 of file net.c.

## **5.127    /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/padlock.c File Reference**

```
#include "xyssl/config.h"
#include "xyssl/aes.h"
#include "xyssl/padlock.h"
```

## 5.128 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/rsa.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/rsa.h"
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
#include "xyssl/sha1.h"
```

### Defines

- #define [KEY\\_LEN](#) 128
- #define [RSA\\_N](#)
- #define [RSA\\_E](#) "10001"
- #define [RSA\\_D](#)
- #define [RSA\\_P](#)
- #define [RSA\\_Q](#)
- #define [RSA\\_DP](#)
- #define [RSA\\_DQ](#)
- #define [RSA\\_QP](#)
- #define [PT\\_LEN](#) 24
- #define [RSA\\_PT](#)

### Functions

- void [rsa\\_init](#) ([rsa\\_context](#) \*ctx, int padding, int hash\_id, int(\*f\_rng)(void \*), void \*p\_rng)  
*Initialize an RSA context.*
- int [rsa\\_gen\\_key](#) ([rsa\\_context](#) \*ctx, int nbits, int exponent)  
*Generate an RSA keypair.*
- int [rsa\\_check\\_pubkey](#) ([rsa\\_context](#) \*ctx)  
*Check a public RSA key.*
- int [rsa\\_check\\_privkey](#) ([rsa\\_context](#) \*ctx)  
*Check a private RSA key.*
- int [rsa\\_public](#) ([rsa\\_context](#) \*ctx, unsigned char \*input, unsigned char \*output)  
*Do an RSA public key operation.*
- int [rsa\\_private](#) ([rsa\\_context](#) \*ctx, unsigned char \*input, unsigned char \*output)  
*Do an RSA private key operation.*
- int [rsa\\_pkcs1\\_encrypt](#) ([rsa\\_context](#) \*ctx, int mode, int ilen, unsigned char \*input, unsigned char \*output)  
*Add the message padding, then do an RSA operation.*

- `int rsa_pkcs1_decrypt (rsa_context *ctx, int mode, int *olen, unsigned char *input, unsigned char *output)`

*Do an RSA operation, then remove the message padding.*

- `int rsa_pkcs1_sign (rsa_context *ctx, int mode, int hash_id, int hashlen, unsigned char *hash, unsigned char *sig)`

*Do a private RSA to sign a message digest.*

- `int rsa_pkcs1_verify (rsa_context *ctx, int mode, int hash_id, int hashlen, unsigned char *hash, unsigned char *sig)`

*Do a public RSA and check the message digest.*

- `void rsa_free (rsa_context *ctx)`

*Free the components of an RSA key.*

- `int rsa_self_test (int verbose)`

*Checkup routine.*

## 5.128.1 Define Documentation

### 5.128.1.1 #define KEY\_LEN 128

Definition at line 573 of file rsa.c.

Referenced by `rsa_self_test()`.

### 5.128.1.2 #define PT\_LEN 24

Definition at line 620 of file rsa.c.

Referenced by `rsa_self_test()`.

### 5.128.1.3 #define RSA\_D

**Value:**

```
"24BF6185468786FDD303083D25E64EFC" \
    "66CA472BC44D253102F8B4A9D3BFA750" \
    "91386C0077937FE33FA325D28855837" \
    "AE1B484A8A9A45F7EE8C0C634F99E8CD" \
    "DF79C5CE07EE72C7F123142198164234" \
    "CABB724CF78B8173B9F880FC86322407" \
    "AF1FEDFDDE2BEB674CA15F3E81A1521E" \
    "071513A1E85B5DFA031F21ECAE91A34D"
```

Definition at line 586 of file rsa.c.

Referenced by `rsa_self_test()`.

#### 5.128.1.4 #define RSA\_DP

**Value:**

```
"C1ACF567564274FB07A0BBAD5D26E298" \
    "3C94D22288ACD763FD8E5600ED4A702D" \
    "F84198A5F06C2E72236AE490C93F07F8" \
    "3CC559CD27BC2D1CA488811730BB5725"
```

Definition at line 605 of file rsa.c.

Referenced by rsa\_self\_test().

#### 5.128.1.5 #define RSA\_DQ

**Value:**

```
"4959CBF6F8FEF750AEE6977C155579C7" \
    "D8AAEA56749EA28623272E4F7D0592AF" \
    "7C1F1313CAC9471B5C523BFE592F517B" \
    "407A1BD76C164B93DA2D32A383E58357"
```

Definition at line 610 of file rsa.c.

Referenced by rsa\_self\_test().

#### 5.128.1.6 #define RSA\_E "10001"

Definition at line 584 of file rsa.c.

Referenced by rsa\_self\_test().

#### 5.128.1.7 #define RSA\_N

**Value:**

```
"9292758453063D803DD603D5E777D788" \
    "8ED1D5BF35786190FA2F23EBC0848AEA" \
    "DDA92CA6C3D80B32C4D109BE0F36D6AE" \
    "7130B9CED7ACDF54CFC7555AC14EEBAB" \
    "93A89813FBF3C4F8066D2D800F7C38A8" \
    "1AE31942917403FF4946B0A83D3D3E05" \
    "EE57C6F5F5606FB5D4BC6CD34EE0801A" \
    "5E94BB77B07507233A0BC7BAC8F90F79"
```

Definition at line 575 of file rsa.c.

Referenced by rsa\_self\_test().

#### 5.128.1.8 #define RSA\_P

**Value:**

```
"C36D0EB7FCD285223CFB5AABA5BDA3D8" \
    "2C01CAD19EA484A87EA4377637E75500" \
    "FCB2005C5C7DD6EC4AC023CDA285D796" \
    "C3D9E75E1EFC42488BB4F1D13AC30A57"
```

Definition at line 595 of file rsa.c.

Referenced by rsa\_self\_test().

#### 5.128.1.9 #define RSA\_PT

**Value:**

```
"\xAA\xBB\xCC\x03\x02\x01\x00\xff\xff\xff\xff" \
"\x11\x22\x33\x0A\x0B\x0C\xCC\xDD\xDD\xDD\xDD"
```

Definition at line 621 of file rsa.c.

Referenced by rsa\_self\_test().

#### 5.128.1.10 #define RSA\_Q

**Value:**

```
"C000DF51A7C77AE8D7C7370C1FF55B69" \
"E211C2B9E5DB1ED0BF61D0D9899620F4" \
"910E4168387E3C30AA1E00C339A79508" \
"8452DD96A9A5EA5D9DCA68DA636032AF"
```

Definition at line 600 of file rsa.c.

Referenced by rsa\_self\_test().

#### 5.128.1.11 #define RSA\_QP

**Value:**

```
"9AE7FBC99546432DF71896FC239EADAE" \
"F38D18D2B2F0E2DD275AA977E2BF4411" \
"F5A3B2A5D33605AEBBCCBA7FEB9F2D2F" \
"A74206CEC169D74BF5A8C50D6F48EA08"
```

Definition at line 615 of file rsa.c.

Referenced by rsa\_self\_test().

### 5.128.2 Function Documentation

#### 5.128.2.1 int rsa\_check\_privkey (rsa\_context \* ctx)

Check a private RSA key.

##### Parameters

*ctx* RSA context to be checked

##### Returns

0 if successful, or an XYSSL\_ERR\_RSA\_XXX error code



Definition at line 152 of file rsa.c.

References `rsa_context::D`, `rsa_context::E`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_free()`, `mpi_gcd()`, `mpi_init()`, `mpi_mod_mpi()`, `mpi_mul_mpi()`, `mpi_sub_int()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_check_pubkey()`, and `XYSSL_ERR_RSA_KEY_CHECK_FAILED`.

Referenced by `rsa_decryption()`, `rsa_self_test()`, and `x509parse_key()`.

#### 5.128.2.2 `int rsa_check_pubkey (rsa_context * ctx)`

Check a public RSA key.

##### Parameters

*ctx* RSA context to be checked

##### Returns

0 if successful, or an `XYSSL_ERR_RSA_XXX` error code

Definition at line 132 of file rsa.c.

References `rsa_context::E`, `mpi_msb()`, `rsa_context::N`, `mpi::p`, and `XYSSL_ERR_RSA_KEY_CHECK_FAILED`.

Referenced by `rsa_check_privkey()`, `rsa_encryption()`, `rsa_self_test()`, and `x509parse_crt()`.

#### 5.128.2.3 `void rsa_free (rsa_context * ctx)`

Free the components of an RSA key.

Definition at line 558 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `mpi_free()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_context::RN`, `rsa_context::RP`, and `rsa_context::RQ`.

Referenced by `main()`, `rsa_gen_key()`, `rsa_self_test()`, `ssl_test()`, `x509_free()`, `x509_self_test()`, and `x509parse_key()`.

#### 5.128.2.4 `int rsa_gen_key (rsa_context * ctx, int nbits, int exponent)`

Generate an RSA keypair.

##### Parameters

*ctx* RSA context that will hold the key

*nbits* size of the public key in bits

*exponent* public exponent (e.g., 65537)

##### Note

[rsa\\_init\(\)](#) must be called beforehand to setup the RSA context (especially `f_rng` and `p_rng`).

##### Returns

0 if successful, or an `XYSSL_ERR_RSA_XXX` error code

Definition at line 60 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `rsa_context::f_rng`, `rsa_context::len`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_free()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_init()`, `mpi_inv_mod()`, `mpi_lset()`, `mpi_mod_mpi()`, `mpi_msb()`, `mpi_mul_mpi()`, `mpi_sub_int()`, `mpi_swap()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::p_rng`, `rsa_context::Q`, `rsa_context::QP`, `rsa_free()`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_KEY_GEN_FAILED`.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, and `main()`.

#### 5.128.2.5 `void rsa_init (rsa_context * ctx, int padding, int hash_id, int(*) (void *) f_rng, void * p_rng)`

Initialize an RSA context.

##### Parameters

*ctx* RSA context to be initialized  
*padding* RSA\_PKCS\_V15 or RSA\_PKCS\_V21  
*hash\_id* RSA\_PKCS\_V21 hash identifier  
*f\_rng* RNG function  
*p\_rng* RNG parameter

##### Note

The *hash\_id* parameter is actually ignored when using RSA\_PKCS\_V15 padding.  
 Currently (xyssl-0.8), RSA\_PKCS\_V21 padding is not supported.

Definition at line 40 of file rsa.c.

References `rsa_context::f_rng`, `rsa_context::hash_id`, `rsa_context::p_rng`, and `rsa_context::padding`.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, `main()`, `rsa_decryption()`, and `rsa_encryption()`.

#### 5.128.2.6 `int rsa_pkcs1_decrypt (rsa_context * ctx, int mode, int * olen, unsigned char * input, unsigned char * output)`

Do an RSA operation, then remove the message padding.

##### Parameters

*ctx* RSA context  
*mode* RSA\_PUBLIC or RSA\_PRIVATE  
*input* buffer holding the encrypted data  
*output* buffer that will hold the plaintext  
*olen* will contain the plaintext length

##### Returns

0 if successful, or an `XYSSL_ERR_RSA_XXX` error code

##### Note

The output buffer must be as large as the size of `ctx->N` (eg. 128 bytes if RSA-1024 is used).

Definition at line 326 of file rsa.c.

References `buf`, `int`, `rsa_context::len`, `rsa_context::padding`, `RSA_CRYPT`, `RSA_PKCS_V15`, `rsa_private()`, `rsa_public()`, `RSA_PUBLIC`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_INVALID_PADDING`.

Referenced by `rsa_decryption()`, `RSA_private_decrypt()`, `RSA_public_decrypt()`, `rsa_self_test()`, and `ssl_parse_client_key_exchange()`.

#### 5.128.2.7 `int rsa_pkcs1_encrypt(rsa_context * ctx, int mode, int ilen, unsigned char * input, unsigned char * output)`

Add the message padding, then do an RSA operation.

##### Parameters

*ctx* RSA context  
*mode* `RSA_PUBLIC` or `RSA_PRIVATE`  
*ilen* contains the the plaintext length  
*input* buffer holding the data to be encrypted  
*output* buffer that will hold the ciphertext

##### Returns

0 if successful, or an `XYSSL_ERR_RSA_XXX` error code

##### Note

The output buffer must be as large as the size of `ctx->N` (eg. 128 bytes if RSA-1024 is used).

Definition at line 280 of file rsa.c.

References `rsa_context::len`, `rsa_context::padding`, `RSA_CRYPT`, `RSA_PKCS_V15`, `rsa_private()`, `rsa_public()`, `RSA_PUBLIC`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_INVALID_PADDING`.

Referenced by `rsa_encryption()`, `RSA_private_encrypt()`, `RSA_public_encrypt()`, `rsa_self_test()`, and `ssl_write_client_key_exchange()`.

#### 5.128.2.8 `int rsa_pkcs1_sign(rsa_context * ctx, int mode, int hash_id, int hashlen, unsigned char * hash, unsigned char * sig)`

Do a private RSA to sign a message digest.

##### Parameters

*ctx* RSA context  
*mode* `RSA_PUBLIC` or `RSA_PRIVATE`  
*hash\_id* `RSA_RAW`, `RSA_MD{2,4,5}` or `RSA_SHA{1,256}`  
*hashlen* message digest length (for `RSA_RAW` only)  
*hash* buffer holding the message digest  
*sig* buffer that will hold the ciphertext

**Returns**

0 if the signing operation was successful, or an XYSSL\_ERR\_RSA\_XXX error code

**Note**

The "sig" buffer must be as large as the size of ctx->N (eg. 128 bytes if RSA-1024 is used).

Definition at line 379 of file rsa.c.

References ASN1\_HASH\_MDX, ASN1\_HASH\_SHA1, rsa\_context::len, rsa\_context::padding, RSA\_MD2, RSA\_MD4, RSA\_MD5, RSA\_PKCS\_V15, rsa\_private(), rsa\_public(), RSA\_PUBLIC, RSA\_RAW, RSA\_SHA1, RSA\_SIGN, XYSSL\_ERR\_RSA\_BAD\_INPUT\_DATA, and XYSSL\_ERR\_RSA\_INVALID\_PADDING.

Referenced by main(), rsa\_self\_test(), ssl\_write\_certificate\_verify(), and ssl\_write\_server\_key\_exchange().

**5.128.2.9 int rsa\_pkcs1\_verify (rsa\_context \* ctx, int mode, int hash\_id, int hashlen, unsigned char \* hash, unsigned char \* sig)**

Do a public RSA and check the message digest.

**Parameters**

*ctx* points to an RSA public key

*mode* RSA\_PUBLIC or RSA\_PRIVATE

*hash\_id* RSA\_RAW, RSA\_MD{2,4,5} or RSA\_SHA{1,256}

*hashlen* message digest length (for RSA\_RAW only)

*hash* buffer holding the message digest

*sig* buffer holding the ciphertext

**Returns**

0 if the verify operation was successful, or an XYSSL\_ERR\_RSA\_XXX error code

**Note**

The "sig" buffer must be as large as the size of ctx->N (eg. 128 bytes if RSA-1024 is used).

Definition at line 468 of file rsa.c.

References ASN1\_HASH\_MDX, ASN1\_HASH\_SHA1, buf, int, rsa\_context::len, rsa\_context::padding, RSA\_MD2, RSA\_MD4, RSA\_MD5, RSA\_PKCS\_V15, rsa\_private(), rsa\_public(), RSA\_PUBLIC, RSA\_RAW, RSA\_SHA1, RSA\_SIGN, XYSSL\_ERR\_RSA\_BAD\_INPUT\_DATA, XYSSL\_ERR\_RSA\_INVALID\_PADDING, and XYSSL\_ERR\_RSA\_VERIFY\_FAILED.

Referenced by main(), rsa\_self\_test(), ssl\_parse\_certificate\_verify(), ssl\_parse\_server\_key\_exchange(), and x509parse\_verify().

**5.128.2.10 int rsa\_private (rsa\_context \* ctx, unsigned char \* input, unsigned char \* output)**

Do an RSA private key operation.

**Parameters**

*ctx* RSA context  
*input* input buffer  
*output* output buffer

**Returns**

0 if successful, or an XYSSL\_ERR\_RSA\_XXX error code

**Note**

The input and output buffers must be large enough (eg. 128 bytes if RSA-1024 is used).

Definition at line 221 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::len`, `mpi_add_mpi()`, `MPI_CHK`, `mpi_cmp_mpi()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_init()`, `mpi_mod_mpi()`, `mpi_mul_mpi()`, `mpi_read_binary()`, `mpi_sub_mpi()`, `mpi_write_binary()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_context::RN`, `rsa_context::RP`, `rsa_context::RQ`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_PRIVATE_FAILED`.

Referenced by `main()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, and `rsa_pkcs1_verify()`.

**5.128.2.11 int rsa\_public (rsa\_context \* ctx, unsigned char \* input, unsigned char \* output)**

Do an RSA public key operation.

**Parameters**

*ctx* RSA context  
*input* input buffer  
*output* output buffer

**Returns**

0 if successful, or an XYSSL\_ERR\_RSA\_XXX error code

**Note**

This function does NOT take care of message padding. Also, be sure to set `input[0] = 0`.  
 The input and output buffers must be large enough (eg. 128 bytes if RSA-1024 is used).

Definition at line 187 of file rsa.c.

References `rsa_context::E`, `rsa_context::len`, `MPI_CHK`, `mpi_cmp_mpi()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_init()`, `mpi_read_binary()`, `mpi_write_binary()`, `rsa_context::N`, `rsa_context::RN`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_PUBLIC_FAILED`.

Referenced by `main()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, and `rsa_pkcs1_verify()`.

**5.128.2.12 int rsa\_self\_test (int verbose)**

Checkup routine.

**Returns**

0 if successful, or 1 if the test failed

Definition at line 627 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `KEY_LEN`, `rsa_context::len`, `mpi_read_string()`, `rsa_context::N`, `rsa_context::P`, `PT_LEN`, `rsa_context::Q`, `rsa_context::QP`, `rsa_check_privkey()`, `rsa_check_pubkey()`, `RSA_D`, `RSA_DP`, `RSA_DQ`, `RSA_E`, `rsa_free()`, `RSA_N`, `RSA_P`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, `rsa_pkcs1_verify()`, `RSA_PRIVATE`, `RSA_PT`, `RSA_PUBLIC`, `RSA_Q`, `RSA_QP`, `RSA_SHA1`, and `sha1()`.

Referenced by `main()`.

## 5.129 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha1.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/sha1.h"
#include <string.h>
#include <stdio.h>
```

### Defines

- #define [GET\\_ULONG\\_BE](#)(n, b, i)
- #define [PUT\\_ULONG\\_BE](#)(n, b, i)
- #define [S](#)(x, n) (((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))
- #define [R](#)(t)
- #define [P](#)(a, b, c, d, e, x)
- #define [F](#)(x, y, z) (z ^ (x & (y ^ z)))
- #define [K](#) 0x5A827999
- #define [F](#)(x, y, z) (x ^ y ^ z)
- #define [K](#) 0x6ED9EBA1
- #define [F](#)(x, y, z) ((x & y) | (z & (x | y)))
- #define [K](#) 0x8F1BBCDC
- #define [F](#)(x, y, z) (x ^ y ^ z)
- #define [K](#) 0xCA62C1D6

### Functions

- void [sha1\\_starts](#) ([sha1\\_context](#) \*ctx)  
*SHA-1 context setup.*
- static void [sha1\\_process](#) ([sha1\\_context](#) \*ctx, unsigned char data[64])
- void [sha1\\_update](#) ([sha1\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*SHA-1 process buffer.*
- void [sha1\\_finish](#) ([sha1\\_context](#) \*ctx, unsigned char output[20])  
*SHA-1 final digest.*
- void [sha1](#) (unsigned char \*input, [int](#) ilen, unsigned char output[20])  
*Output = SHA-1( input buffer ).*
- [int](#) [sha1\\_file](#) (char \*path, unsigned char output[20])  
*Output = SHA-1( file contents ).*
- void [sha1\\_hmac\\_starts](#) ([sha1\\_context](#) \*ctx, unsigned char \*key, [int](#) keylen)  
*SHA-1 HMAC context setup.*
- void [sha1\\_hmac\\_update](#) ([sha1\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*SHA-1 HMAC process buffer.*

- void `sha1_hmac_finish` (`sha1_context` \*ctx, unsigned char output[20])  
*SHA-1 HMAC final digest.*
- void `sha1_hmac` (unsigned char \*key, int keylen, unsigned char \*input, int ilen, unsigned char output[20])  
*Output = HMAC-SHA-1( hmac key, input buffer ).*
- int `sha1_self_test` (int verbose)  
*Checkup routine.*

## Variables

- static const unsigned char `sha1_padding` [64]
- static unsigned char `sha1_test_buf` [3][57]
- static const int `sha1_test_buflen` [3]
- static const unsigned char `sha1_test_sum` [3][20]
- static unsigned char `sha1_hmac_test_key` [7][26]
- static const int `sha1_hmac_test_keylen` [7]
- static unsigned char `sha1_hmac_test_buf` [7][74]
- static const int `sha1_hmac_test_buflen` [7]
- static const unsigned char `sha1_hmac_test_sum` [7][20]

## 5.129.1 Define Documentation

**5.129.1.1** `#define F(x, y, z) (x ^ y ^ z)`

**5.129.1.2** `#define F(x, y, z) ((x & y) | (z & (x | y)))`

**5.129.1.3** `#define F(x, y, z) (x ^ y ^ z)`

**5.129.1.4** `#define F(x, y, z) (z ^ (x & (y ^ z)))`

**5.129.1.5** `#define GET_ULONG_BE(n, b, i)`

**Value:**

```
{
    (n) = ( (unsigned long) (b)[(i)      ] << 24 ) \
    | ( (unsigned long) (b)[(i) + 1] << 16 ) \
    | ( (unsigned long) (b)[(i) + 2] <<  8 ) \
    | ( (unsigned long) (b)[(i) + 3]      ); \
}
```

Definition at line 39 of file sha1.c.



**5.129.1.6** `#define K 0xCA62C1D6`

**5.129.1.7** `#define K 0x8F1BBCDC`

**5.129.1.8** `#define K 0x6ED9EBA1`

**5.129.1.9** `#define K 0x5A827999`

Referenced by sha4\_process().

**5.129.1.10** `#define P(a, b, c, d, e, x)`

**Value:**

```
{
    e += S(a,5) + F(b,c,d) + K + x; b = S(b,30);
}
```

**5.129.1.11** `#define PUT_ULONG_BE(n, b, i)`

**Value:**

```
{
    (b)[(i)    ] = (unsigned char) ( (n) >> 24 );
    (b)[(i) + 1] = (unsigned char) ( (n) >> 16 );
    (b)[(i) + 2] = (unsigned char) ( (n) >>  8 );
    (b)[(i) + 3] = (unsigned char) ( (n)          );
}
```

Definition at line 49 of file sha1.c.

**5.129.1.12** `#define R(t)`

**Value:**

```
(
    temp = W[(t - 3) & 0x0F] ^ W[(t - 8) & 0x0F] ^
           W[(t - 14) & 0x0F] ^ W[t & 0x0F],
    ( W[t & 0x0F] = S(temp,1) )
)
```

Referenced by mpi\_is\_prime(), sha1\_process(), and sha2\_process().

**5.129.1.13** `#define S(x, n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))`

## 5.129.2 Function Documentation

**5.129.2.1** `void sha1 (unsigned char * input, int ilen, unsigned char output[20])`

Output = SHA-1( *input* buffer ).

**Parameters**

*input* buffer holding the data

*ilen* length of the input data  
*output* SHA-1 checksum result

Definition at line 313 of file sha1.c.

References sha1\_finish(), sha1\_starts(), and sha1\_update().

Referenced by main(), rsa\_self\_test(), sha1\_hmac\_starts(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_sha1(), ssl\_parse\_finished(), ssl\_parse\_server\_key\_exchange(), ssl\_write\_finished(), ssl\_write\_server\_key\_exchange(), and x509\_hash().

### 5.129.2.2 int sha1\_file (char \* *path*, unsigned char *output*[20])

Output = SHA-1( file contents ).

#### Parameters

*path* input file name  
*output* SHA-1 checksum result

#### Returns

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 327 of file sha1.c.

References buf, f, sha1\_finish(), sha1\_starts(), and sha1\_update().

Referenced by main(), and sha1\_wrapper().

### 5.129.2.3 void sha1\_finish (sha1\_context \* *ctx*, unsigned char *output*[20])

SHA-1 final digest.

#### Parameters

*ctx* SHA-1 context  
*output* SHA-1 checksum result

Definition at line 284 of file sha1.c.

References PUT\_ULONG\_BE, sha1\_padding, sha1\_update(), sha1\_context::state, and sha1\_context::total.

Referenced by sha1(), sha1\_file(), sha1\_hmac\_finish(), sha1\_self\_test(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_sha1(), ssl\_parse\_server\_key\_exchange(), and ssl\_write\_server\_key\_exchange().

### 5.129.2.4 void sha1\_hmac (unsigned char \* *key*, int *keylen*, unsigned char \* *input*, int *ilen*, unsigned char *output*[20])

Output = HMAC-SHA-1( hmac key, input buffer ).

#### Parameters

*key* HMAC secret key

*keylen* length of the HMAC key  
*input* buffer holding the data  
*ilen* length of the input data  
*output* HMAC-SHA-1 result

Definition at line 413 of file sha1.c.

References sha1\_hmac\_finish(), sha1\_hmac\_starts(), and sha1\_hmac\_update().

Referenced by ssl\_decrypt\_buf(), ssl\_encrypt\_buf(), and tls1\_prf().

#### 5.129.2.5 void sha1\_hmac\_finish (sha1\_context \* ctx, unsigned char output[20])

SHA-1 HMAC final digest.

##### Parameters

*ctx* HMAC context  
*output* SHA-1 HMAC checksum result

Definition at line 397 of file sha1.c.

References sha1\_context::opad, sha1\_finish(), sha1\_starts(), and sha1\_update().

Referenced by sha1\_hmac(), and sha1\_self\_test().

#### 5.129.2.6 void sha1\_hmac\_starts (sha1\_context \* ctx, unsigned char \* key, int keylen)

SHA-1 HMAC context setup.

##### Parameters

*ctx* HMAC context to be initialized  
*key* HMAC secret key  
*keylen* length of the HMAC key

Definition at line 359 of file sha1.c.

References sha1\_context::ipad, sha1\_context::opad, sha1(), sha1\_starts(), and sha1\_update().

Referenced by sha1\_hmac(), and sha1\_self\_test().

#### 5.129.2.7 void sha1\_hmac\_update (sha1\_context \* ctx, unsigned char \* input, int ilen)

SHA-1 HMAC process buffer.

##### Parameters

*ctx* HMAC context  
*input* buffer holding the data  
*ilen* length of the input data

Definition at line 389 of file sha1.c.

References sha1\_update().

Referenced by sha1\_hmac(), and sha1\_self\_test().

**5.129.2.8 static void sha1\_process (sha1\_context \* ctx, unsigned char data[64]) [static]**

Definition at line 73 of file sha1.c.

References GET\_ULONG\_BE, P, R, and sha1\_context::state.

Referenced by sha1\_update().

**5.129.2.9 int sha1\_self\_test (int verbose)**

Checksum routine.

**Returns**

0 if successful, or 1 if the test failed

Definition at line 521 of file sha1.c.

References buf, sha1\_finish(), sha1\_hmac\_finish(), sha1\_hmac\_starts(), sha1\_hmac\_test\_buf, sha1\_hmac\_test\_buflen, sha1\_hmac\_test\_key, sha1\_hmac\_test\_keylen, sha1\_hmac\_test\_sum, sha1\_hmac\_update(), sha1\_starts(), sha1\_test\_buf, sha1\_test\_buflen, sha1\_test\_sum, and sha1\_update().

Referenced by main().

**5.129.2.10 void sha1\_starts (sha1\_context \* ctx)**

SHA-1 context setup.

**Parameters**

*ctx* context to be initialized

Definition at line 61 of file sha1.c.

References sha1\_context::state, and sha1\_context::total.

Referenced by sha1(), sha1\_file(), sha1\_hmac\_finish(), sha1\_hmac\_starts(), sha1\_self\_test(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_init(), ssl\_mac\_sha1(), ssl\_parse\_server\_key\_exchange(), and ssl\_write\_server\_key\_exchange().

**5.129.2.11 void sha1\_update (sha1\_context \* ctx, unsigned char \* input, int ilen)**

SHA-1 process buffer.

**Parameters**

*ctx* SHA-1 context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 232 of file sha1.c.

References sha1\_context::buffer, sha1\_process(), and sha1\_context::total.

Referenced by sha1(), sha1\_file(), sha1\_finish(), sha1\_hmac\_finish(), sha1\_hmac\_starts(), sha1\_hmac\_update(), sha1\_self\_test(), ssl\_calc\_finished(), ssl\_calc\_verify(), ssl\_derive\_keys(), ssl\_mac\_sha1(), ssl\_parse\_client\_hello(), ssl\_parse\_server\_key\_exchange(), ssl\_read\_record(), ssl\_write\_record(), and ssl\_write\_server\_key\_exchange().

### 5.129.3 Variable Documentation

#### 5.129.3.1 unsigned char sha1\_hmac\_test\_buf[7][74] [static]

Initial value:

```
{
    { "Hi There" },
    { "what do ya want for nothing?" },
    { "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD" },
    { "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD" },
    { "Test With Truncation" },
    { "Test Using Larger Than Block-Size Key - Hash Key First" },
    { "Test Using Larger Than Block-Size Key and Larger"
      " Than One Block-Size Data" }
}
```

Definition at line 475 of file sha1.c.

Referenced by sha1\_self\_test().

#### 5.129.3.2 const int sha1\_hmac\_test\_buflen[7] [static]

Initial value:

```
{
    8, 28, 50, 50, 20, 54, 73
}
```

Definition at line 495 of file sha1.c.

Referenced by sha1\_self\_test().

#### 5.129.3.3 unsigned char sha1\_hmac\_test\_key[7][26] [static]

Initial value:

```
{
    { "\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B"
      "\x0B\x0B\x0B\x0B" },
    { "Jefe" },
    { "\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA"
      "\xAA\xAA\xAA\xAA" },
    { "\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0A\x0B\x0C\x0D\x0E\x0F\x10"
      "\x11\x12\x13\x14\x15\x16\x17\x18\x19" },
    { "\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C"
      "\x0C\x0C\x0C\x0C" },
    { "" },
    { "" }
}
```

Definition at line 455 of file sha1.c.

Referenced by sha1\_self\_test().

#### 5.129.3.4 `const int sha1_hmac_test_keylen[7] [static]`

**Initial value:**

```
{
    20, 4, 20, 25, 20, 80, 80
}
```

Definition at line 470 of file sha1.c.

Referenced by sha1\_self\_test().

#### 5.129.3.5 `const unsigned char sha1_hmac_test_sum[7][20] [static]`

**Initial value:**

```
{
    { 0xB6, 0x17, 0x31, 0x86, 0x55, 0x05, 0x72, 0x64, 0xE2, 0x8B,
      0xC0, 0xB6, 0xFB, 0x37, 0x8C, 0x8E, 0xF1, 0x46, 0xBE, 0x00 },
    { 0xEF, 0xFC, 0xDF, 0x6A, 0xE5, 0xEB, 0x2F, 0xA2, 0xD2, 0x74,
      0x16, 0xD5, 0xF1, 0x84, 0xDF, 0x9C, 0x25, 0x9A, 0x7C, 0x79 },
    { 0x12, 0x5D, 0x73, 0x42, 0xB9, 0xAC, 0x11, 0xCD, 0x91, 0xA3,
      0x9A, 0xF4, 0x8A, 0xA1, 0x7B, 0x4F, 0x63, 0xF1, 0x75, 0xD3 },
    { 0x4C, 0x90, 0x07, 0xF4, 0x02, 0x62, 0x50, 0xC6, 0xBC, 0x84,
      0x14, 0xF9, 0xBF, 0x50, 0xC8, 0x6C, 0x2D, 0x72, 0x35, 0xDA },
    { 0x4C, 0x1A, 0x03, 0x42, 0x4B, 0x55, 0xE0, 0x7F, 0xE7, 0xF2,
      0x7B, 0xE1 },
    { 0xAA, 0x4A, 0xE5, 0xE1, 0x52, 0x72, 0xD0, 0x0E, 0x95, 0x70,
      0x56, 0x37, 0xCE, 0x8A, 0x3B, 0x55, 0xED, 0x40, 0x21, 0x12 },
    { 0xE8, 0xE9, 0x9D, 0x0F, 0x45, 0x23, 0x7D, 0x78, 0x6D, 0x6B,
      0xBA, 0xA7, 0x96, 0x5C, 0x78, 0x08, 0xBB, 0xFF, 0x1A, 0x91 }
}
```

Definition at line 500 of file sha1.c.

Referenced by sha1\_self\_test().

#### 5.129.3.6 `const unsigned char sha1_padding[64] [static]`

**Initial value:**

```
{
    0x80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
}
```

Definition at line 273 of file sha1.c.

Referenced by sha1\_finish().

**5.129.3.7 unsigned char sha1\_test\_buf[3][57] [static]****Initial value:**

```
{
    { "abc" },
    { "abcdbcdecdefdefgfegefghfghighijhi jkijkljklmklmnlmnomnopnopq" },
    { "" }
}
```

Definition at line 430 of file sha1.c.

Referenced by sha1\_self\_test().

**5.129.3.8 const int sha1\_test\_buflen[3] [static]****Initial value:**

```
{
    3, 56, 1000
}
```

Definition at line 437 of file sha1.c.

Referenced by sha1\_self\_test().

**5.129.3.9 const unsigned char sha1\_test\_sum[3][20] [static]****Initial value:**

```
{
    { 0xA9, 0x99, 0x3E, 0x36, 0x47, 0x06, 0x81, 0x6A, 0xBA, 0x3E,
      0x25, 0x71, 0x78, 0x50, 0xC2, 0x6C, 0x9C, 0xD0, 0xD8, 0x9D },
    { 0x84, 0x98, 0x3E, 0x44, 0x1C, 0x3B, 0xD2, 0x6E, 0xBA, 0xAE,
      0x4A, 0xA1, 0xF9, 0x51, 0x29, 0xE5, 0xE5, 0x46, 0x70, 0xF1 },
    { 0x34, 0xAA, 0x97, 0x3C, 0xD4, 0xC4, 0xDA, 0xA4, 0xF6, 0x1E,
      0xEB, 0x2B, 0xDB, 0xAD, 0x27, 0x31, 0x65, 0x34, 0x01, 0x6F }
}
```

Definition at line 442 of file sha1.c.

Referenced by sha1\_self\_test().

## 5.130 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha2.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/sha2.h"
#include <string.h>
#include <stdio.h>
```

### Defines

- #define [GET\\_ULONG\\_BE](#)(n, b, i)
- #define [PUT\\_ULONG\\_BE](#)(n, b, i)
- #define [SHR](#)(x, n) ((x & 0xFFFFFFFF) >> n)
- #define [ROTR](#)(x, n) (SHR(x,n) | (x << (32 - n)))
- #define [S0](#)(x) (ROTR(x, 7) ^ ROTR(x,18) ^ SHR(x, 3))
- #define [S1](#)(x) (ROTR(x,17) ^ ROTR(x,19) ^ SHR(x,10))
- #define [S2](#)(x) (ROTR(x, 2) ^ ROTR(x,13) ^ ROTR(x,22))
- #define [S3](#)(x) (ROTR(x, 6) ^ ROTR(x,11) ^ ROTR(x,25))
- #define [F0](#)(x, y, z) ((x & y) | (z & (x | y)))
- #define [F1](#)(x, y, z) (z ^ (x & (y ^ z)))
- #define [R](#)(t)
- #define [P](#)(a, b, c, d, e, [f](#), g, h, x, [K](#))

### Functions

- void [sha2\\_starts](#) ([sha2\\_context](#) \*ctx, [int](#) is224)  
*SHA-256 context setup.*
- static void [sha2\\_process](#) ([sha2\\_context](#) \*ctx, unsigned char data[64])
- void [sha2\\_update](#) ([sha2\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*SHA-256 process buffer.*
- void [sha2\\_finish](#) ([sha2\\_context](#) \*ctx, unsigned char output[32])  
*SHA-256 final digest.*
- void [sha2](#) (unsigned char \*input, [int](#) ilen, unsigned char output[32], [int](#) is224)  
*Output = SHA-256( input buffer ).*
- [int](#) [sha2\\_file](#) (char \*path, unsigned char output[32], [int](#) is224)  
*Output = SHA-256( file contents ).*
- void [sha2\\_hmac\\_starts](#) ([sha2\\_context](#) \*ctx, unsigned char \*key, [int](#) keylen, [int](#) is224)  
*SHA-256 HMAC context setup.*
- void [sha2\\_hmac\\_update](#) ([sha2\\_context](#) \*ctx, unsigned char \*input, [int](#) ilen)  
*SHA-256 HMAC process buffer.*



- void `sha2_hmac_finish` (`sha2_context` \*ctx, unsigned char output[32])  
*SHA-256 HMAC final digest.*
- void `sha2_hmac` (unsigned char \*key, int keylen, unsigned char \*input, int ilen, unsigned char output[32], int is224)  
*Output = HMAC-SHA-256( hmac key, input buffer ).*
- int `sha2_self_test` (int verbose)  
*Checkup routine.*

## Variables

- static const unsigned char `sha2_padding` [64]
- static unsigned char `sha2_test_buf` [3][57]
- static const int `sha2_test_buflen` [3]
- static const unsigned char `sha2_test_sum` [6][32]
- static unsigned char `sha2_hmac_test_key` [7][26]
- static const int `sha2_hmac_test_keylen` [7]
- static unsigned char `sha2_hmac_test_buf` [7][153]
- static const int `sha2_hmac_test_buflen` [7]
- static const unsigned char `sha2_hmac_test_sum` [14][32]

### 5.130.1 Define Documentation

**5.130.1.1** `#define F0(x, y, z) ((x & y) | (z & (x | y)))`

**5.130.1.2** `#define F1(x, y, z) (z ^ (x & (y ^ z)))`

**5.130.1.3** `#define GET_ULONG_BE(n, b, i)`

**Value:**

```
{
    (n) = ( (unsigned long) (b)[(i)      ] << 24 ) \
    | ( (unsigned long) (b)[(i) + 1] << 16 ) \
    | ( (unsigned long) (b)[(i) + 2] <<  8 ) \
    | ( (unsigned long) (b)[(i) + 3]      ); \
}
```

Definition at line 39 of file sha2.c.

**5.130.1.4** `#define P(a, b, c, d, e, f, g, h, x, K)`

**Value:**

```
{
    temp1 = h + S3(e) + F1(e, f, g) + K + x; \
    temp2 = S2(a) + F0(a, b, c); \
    d += temp1; h = temp1 + temp2; \
}
```

**5.130.1.5 #define PUT\_ULONG\_BE(n, b, i)****Value:**

```

{
    (b)[(i)      ] = (unsigned char) ( (n) >> 24 ); \
    (b)[(i) + 1] = (unsigned char) ( (n) >> 16 ); \
    (b)[(i) + 2] = (unsigned char) ( (n) >>  8 ); \
    (b)[(i) + 3] = (unsigned char) ( (n)          ); \
}

```

Definition at line 49 of file sha2.c.

**5.130.1.6 #define R(t)****Value:**

```

(
    W[t] = S1(W[t - 2]) + W[t - 7] + \
          S0(W[t - 15]) + W[t - 16] \
)

```

**5.130.1.7 #define ROTR(x, n) (SHR(x,n) | (x << (32 - n)))****5.130.1.8 #define S0(x) (ROTR(x, 7) ^ ROTR(x,18) ^ SHR(x, 3))**

Referenced by sha4\_process().

**5.130.1.9 #define S1(x) (ROTR(x,17) ^ ROTR(x,19) ^ SHR(x,10))**

Referenced by sha4\_process(), and tls1\_prf().

**5.130.1.10 #define S2(x) (ROTR(x, 2) ^ ROTR(x,13) ^ ROTR(x,22))**

Referenced by tls1\_prf().

**5.130.1.11 #define S3(x) (ROTR(x, 6) ^ ROTR(x,11) ^ ROTR(x,25))****5.130.1.12 #define SHR(x, n) ((x & 0xFFFFFFFF) >> n)****5.130.2 Function Documentation****5.130.2.1 void sha2 (unsigned char \* *input*, int *ilen*, unsigned char *output*[32], int *is224*)**Output = SHA-256( *input* buffer ).**Parameters***input* buffer holding the data*ilen* length of the input data*output* SHA-224/256 checksum result

*is224* 0 = use SHA256, 1 = use SHA224

Definition at line 314 of file sha2.c.

References sha2\_finish(), sha2\_starts(), and sha2\_update().

Referenced by main(), and sha2\_hmac\_starts().

#### 5.130.2.2 int sha2\_file (char \* *path*, unsigned char *output*[32], int *is224*)

Output = SHA-256( file contents ).

##### Parameters

*path* input file name

*output* SHA-224/256 checksum result

*is224* 0 = use SHA256, 1 = use SHA224

##### Returns

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 329 of file sha2.c.

References buf, f, sha2\_finish(), sha2\_starts(), and sha2\_update().

Referenced by sha2\_wrapper().

#### 5.130.2.3 void sha2\_finish (sha2\_context \* *ctx*, unsigned char *output*[32])

SHA-256 final digest.

##### Parameters

*ctx* SHA-256 context

*output* SHA-224/256 checksum result

Definition at line 280 of file sha2.c.

References sha2\_context::is224, PUT\_ULONG\_BE, sha2\_padding, sha2\_update(), sha2\_context::state, and sha2\_context::total.

Referenced by aes\_en\_de(), main(), sha2(), sha2\_file(), sha2\_hmac\_finish(), and sha2\_self\_test().

#### 5.130.2.4 void sha2\_hmac (unsigned char \* *key*, int *keylen*, unsigned char \* *input*, int *ilen*, unsigned char *output*[32], int *is224*)

Output = HMAC-SHA-256( hmac key, input buffer ).

##### Parameters

*key* HMAC secret key

*keylen* length of the HMAC key

*input* buffer holding the data

*ilen* length of the input data  
*output* HMAC-SHA-224/256 result  
*is224* 0 = use SHA256, 1 = use SHA224

Definition at line 420 of file sha2.c.

References sha2\_hmac\_finish(), sha2\_hmac\_starts(), and sha2\_hmac\_update().

#### 5.130.2.5 void sha2\_hmac\_finish (sha2\_context \* ctx, unsigned char output[32])

SHA-256 HMAC final digest.

##### Parameters

*ctx* HMAC context  
*output* SHA-224/256 HMAC checksum result

Definition at line 400 of file sha2.c.

References sha2\_context::is224, sha2\_context::opad, sha2\_finish(), sha2\_starts(), and sha2\_update().

Referenced by aes\_en\_de(), main(), sha2\_hmac(), and sha2\_self\_test().

#### 5.130.2.6 void sha2\_hmac\_starts (sha2\_context \* ctx, unsigned char \* key, int keylen, int is224)

SHA-256 HMAC context setup.

##### Parameters

*ctx* HMAC context to be initialized  
*key* HMAC secret key  
*keylen* length of the HMAC key  
*is224* 0 = use SHA256, 1 = use SHA224

Definition at line 361 of file sha2.c.

References sha2\_context::ipad, sha2\_context::opad, sha2(), sha2\_starts(), and sha2\_update().

Referenced by aes\_en\_de(), main(), sha2\_hmac(), and sha2\_self\_test().

#### 5.130.2.7 void sha2\_hmac\_update (sha2\_context \* ctx, unsigned char \* input, int ilen)

SHA-256 HMAC process buffer.

##### Parameters

*ctx* HMAC context  
*input* buffer holding the data  
*ilen* length of the input data

Definition at line 392 of file sha2.c.

References sha2\_update().

Referenced by aes\_en\_de(), main(), sha2\_hmac(), and sha2\_self\_test().

**5.130.2.8 static void sha2\_process (sha2\_context \* ctx, unsigned char data[64]) [static]**

Definition at line 94 of file sha2.c.

References F, GET\_ULONG\_BE, P, R, and sha2\_context::state.

Referenced by sha2\_update().

**5.130.2.9 int sha2\_self\_test (int verbose)**

Checksum routine.

**Returns**

0 if successful, or 1 if the test failed

Definition at line 599 of file sha2.c.

References buf, sha2\_finish(), sha2\_hmac\_finish(), sha2\_hmac\_starts(), sha2\_hmac\_test\_buf, sha2\_hmac\_test\_buflen, sha2\_hmac\_test\_key, sha2\_hmac\_test\_keylen, sha2\_hmac\_test\_sum, sha2\_hmac\_update(), sha2\_starts(), sha2\_test\_buf, sha2\_test\_buflen, sha2\_test\_sum, and sha2\_update().

Referenced by main().

**5.130.2.10 void sha2\_starts (sha2\_context \* ctx, int is224)**

SHA-256 context setup.

**Parameters**

*ctx* context to be initialized

*is224* 0 = use SHA256, 1 = use SHA224

Definition at line 61 of file sha2.c.

References sha2\_context::is224, sha2\_context::state, and sha2\_context::total.

Referenced by aes\_en\_de(), main(), sha2(), sha2\_file(), sha2\_hmac\_finish(), sha2\_hmac\_starts(), and sha2\_self\_test().

**5.130.2.11 void sha2\_update (sha2\_context \* ctx, unsigned char \* input, int ilen)**

SHA-256 process buffer.

**Parameters**

*ctx* SHA-256 context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 228 of file sha2.c.

References sha2\_context::buffer, sha2\_process(), and sha2\_context::total.

Referenced by aes\_en\_de(), main(), sha2(), sha2\_file(), sha2\_finish(), sha2\_hmac\_finish(), sha2\_hmac\_starts(), sha2\_hmac\_update(), and sha2\_self\_test().

### 5.130.3 Variable Documentation

#### 5.130.3.1 unsigned char sha2\_hmac\_test\_buf[7][153] [static]

Initial value:

```
{
    { "Hi There" },
    { "what do ya want for nothing?" },
    { "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD" },
    { "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD" },
    { "Test With Truncation" },
    { "Test Using Larger Than Block-Size Key - Hash Key First" },
    { "This is a test using a larger than block-size key "
      "and a larger than block-size data. The key needs to "
      "be hashed before being used by the HMAC algorithm." }
}
```

Definition at line 507 of file sha2.c.

Referenced by sha2\_self\_test().

#### 5.130.3.2 const int sha2\_hmac\_test\_buflen[7] [static]

Initial value:

```
{
    8, 28, 50, 50, 20, 54, 152
}
```

Definition at line 528 of file sha2.c.

Referenced by sha2\_self\_test().

#### 5.130.3.3 unsigned char sha2\_hmac\_test\_key[7][26] [static]

Initial value:

```
{
    { "\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B"
      "\x0B\x0B\x0B\x0B" },
    { "Jefe" },
    { "\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA"
      "\xAA\xAA\xAA\xAA" },
    { "\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0A\x0B\x0C\x0D\x0E\x0F\x10"
      "\x11\x12\x13\x14\x15\x16\x17\x18\x19" },
    { "\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C"
      "\x0C\x0C\x0C\x0C" },
    { "" },
    { "" }
}
```

Definition at line 487 of file sha2.c.

Referenced by sha2\_self\_test().

#### 5.130.3.4 `const int sha2_hmac_test_keylen[7] [static]`

**Initial value:**

```
{
    20, 4, 20, 25, 20, 131, 131
}
```

Definition at line 502 of file sha2.c.

Referenced by sha2\_self\_test().

#### 5.130.3.5 `const unsigned char sha2_hmac_test_sum[14][32] [static]`

Definition at line 533 of file sha2.c.

Referenced by sha2\_self\_test().

#### 5.130.3.6 `const unsigned char sha2_padding[64] [static]`

**Initial value:**

```
{
    0x80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
}
```

Definition at line 269 of file sha2.c.

Referenced by sha2\_finish().

#### 5.130.3.7 `unsigned char sha2_test_buf[3][57] [static]`

**Initial value:**

```
{
    { "abc" },
    { "abcdbcdecdefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq" },
    { "" }
}
```

Definition at line 437 of file sha2.c.

Referenced by sha2\_self\_test().

#### 5.130.3.8 `const int sha2_test_buflen[3] [static]`

**Initial value:**

```
{
    3, 56, 1000
}
```

Definition at line 444 of file sha2.c.

Referenced by sha2\_self\_test().

### 5.130.3.9 const unsigned char sha2\_test\_sum[6][32] [static]

**Initial value:**

```
{
    { 0x23, 0x09, 0x7D, 0x22, 0x34, 0x05, 0xD8, 0x22,
      0x86, 0x42, 0xA4, 0x77, 0xBD, 0xA2, 0x55, 0xB3,
      0x2A, 0xAD, 0xBC, 0xE4, 0xBD, 0xA0, 0xB3, 0xF7,
      0xE3, 0x6C, 0x9D, 0xA7 },
    { 0x75, 0x38, 0x8B, 0x16, 0x51, 0x27, 0x76, 0xCC,
      0x5D, 0xBA, 0x5D, 0xA1, 0xFD, 0x89, 0x01, 0x50,
      0xB0, 0xC6, 0x45, 0x5C, 0xB4, 0xF5, 0x8B, 0x19,
      0x52, 0x52, 0x25, 0x25 },
    { 0x20, 0x79, 0x46, 0x55, 0x98, 0x0C, 0x91, 0xD8,
      0xBB, 0xB4, 0xC1, 0xEA, 0x97, 0x61, 0x8A, 0x4B,
      0xF0, 0x3F, 0x42, 0x58, 0x19, 0x48, 0xB2, 0xEE,
      0x4E, 0xE7, 0xAD, 0x67 },

    { 0xBA, 0x78, 0x16, 0xBF, 0x8F, 0x01, 0xCF, 0xEA,
      0x41, 0x41, 0x40, 0xDE, 0x5D, 0xAE, 0x22, 0x23,
      0xB0, 0x03, 0x61, 0xA3, 0x96, 0x17, 0x7A, 0x9C,
      0xB4, 0x10, 0xFF, 0x61, 0xF2, 0x00, 0x15, 0xAD },
    { 0x24, 0x8D, 0x6A, 0x61, 0xD2, 0x06, 0x38, 0xB8,
      0xE5, 0xC0, 0x26, 0x93, 0x0C, 0x3E, 0x60, 0x39,
      0xA3, 0x3C, 0xE4, 0x59, 0x64, 0xFF, 0x21, 0x67,
      0xF6, 0xEC, 0xED, 0xD4, 0x19, 0xDB, 0x06, 0xC1 },
    { 0xCD, 0xC7, 0x6E, 0x5C, 0x99, 0x14, 0xFB, 0x92,
      0x81, 0xA1, 0xC7, 0xE2, 0x84, 0xD7, 0x3E, 0x67,
      0xF1, 0x80, 0x9A, 0x48, 0xA4, 0x97, 0x20, 0x0E,
      0x04, 0x6D, 0x39, 0xCC, 0xC7, 0x11, 0x2C, 0xD0 }
}
```

Definition at line 449 of file sha2.c.

Referenced by sha2\_self\_test().



## 5.131 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha4.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/sha4.h"
#include <string.h>
#include <stdio.h>
```

### Defines

- #define [GET\\_UINT64\\_BE](#)(n, b, i)
- #define [PUT\\_UINT64\\_BE](#)(n, b, i)
- #define [SHR](#)(x, n) (x >> n)
- #define [ROTR](#)(x, n) (SHR(x,n) | (x << (64 - n)))
- #define [S0](#)(x) (ROTR(x, 1) ^ ROTR(x, 8) ^ SHR(x, 7))
- #define [S1](#)(x) (ROTR(x,19) ^ ROTR(x,61) ^ SHR(x, 6))
- #define [S2](#)(x) (ROTR(x,28) ^ ROTR(x,34) ^ ROTR(x,39))
- #define [S3](#)(x) (ROTR(x,14) ^ ROTR(x,18) ^ ROTR(x,41))
- #define [F0](#)(x, y, z) ((x & y) | (z & (x | y)))
- #define [F1](#)(x, y, z) (z ^ (x & (y ^ z)))
- #define [P](#)(a, b, c, d, e, f, g, h, x, K)

### Functions

- void [sha4\\_starts](#) ([sha4\\_context](#) \*ctx, int is384)  
*SHA-512 context setup.*
- static void [sha4\\_process](#) ([sha4\\_context](#) \*ctx, unsigned char data[128])
- void [sha4\\_update](#) ([sha4\\_context](#) \*ctx, unsigned char \*input, int ilen)  
*SHA-512 process buffer.*
- void [sha4\\_finish](#) ([sha4\\_context](#) \*ctx, unsigned char output[64])  
*SHA-512 final digest.*
- void [sha4](#) (unsigned char \*input, int ilen, unsigned char output[64], int is384)  
*Output = SHA-512( input buffer ).*
- int [sha4\\_file](#) (char \*path, unsigned char output[64], int is384)  
*Output = SHA-512( file contents ).*
- void [sha4\\_hmac\\_starts](#) ([sha4\\_context](#) \*ctx, unsigned char \*key, int keylen, int is384)  
*SHA-512 HMAC context setup.*
- void [sha4\\_hmac\\_update](#) ([sha4\\_context](#) \*ctx, unsigned char \*input, int ilen)  
*SHA-512 HMAC process buffer.*
- void [sha4\\_hmac\\_finish](#) ([sha4\\_context](#) \*ctx, unsigned char output[64])

*SHA-512 HMAC final digest.*

- void `sha4_hmac` (unsigned char \*key, int keylen, unsigned char \*input, int ilen, unsigned char output[64], int is384)

*Output = HMAC-SHA-512( hmac key, input buffer ).*

- int `sha4_self_test` (int verbose)

*Checkup routine.*

## Variables

- static const unsigned int64 `K` [80]
- static const unsigned char `sha4_padding` [128]
- static unsigned char `sha4_test_buf` [3][113]
- static const int `sha4_test_buflen` [3]
- static const unsigned char `sha4_test_sum` [6][64]
- static unsigned char `sha4_hmac_test_key` [7][26]
- static const int `sha4_hmac_test_keylen` [7]
- static unsigned char `sha4_hmac_test_buf` [7][153]
- static const int `sha4_hmac_test_buflen` [7]
- static const unsigned char `sha4_hmac_test_sum` [14][64]

### 5.131.1 Define Documentation

**5.131.1.1** `#define F0(x, y, z) ((x & y) | (z & (x | y)))`

**5.131.1.2** `#define F1(x, y, z) (z ^ (x & (y ^ z)))`

**5.131.1.3** `#define GET_UINT64_BE(n, b, i)`

**Value:**

```
{
    (n) = ( (unsigned int64) (b)[(i)      ] << 56 ) \
    | ( (unsigned int64) (b)[(i) + 1] << 48 ) \
    | ( (unsigned int64) (b)[(i) + 2] << 40 ) \
    | ( (unsigned int64) (b)[(i) + 3] << 32 ) \
    | ( (unsigned int64) (b)[(i) + 4] << 24 ) \
    | ( (unsigned int64) (b)[(i) + 5] << 16 ) \
    | ( (unsigned int64) (b)[(i) + 6] <<  8 ) \
    | ( (unsigned int64) (b)[(i) + 7]      ); \
}
```

Definition at line 39 of file sha4.c.

Referenced by `sha4_process()`.

**5.131.1.4** `#define P(a, b, c, d, e, f, g, h, x, K)`

**Value:**

```

{
    temp1 = h + S3(e) + F1(e,f,g) + K + x;
    temp2 = S2(a) + F0(a,b,c);
    d += temp1; h = temp1 + temp2;
}

```

#### 5.131.1.5 #define PUT\_UINT64\_BE(n, b, i)

**Value:**

```

{
    (b)[(i)    ] = (unsigned char) ( (n) >> 56 );
    (b)[(i) + 1] = (unsigned char) ( (n) >> 48 );
    (b)[(i) + 2] = (unsigned char) ( (n) >> 40 );
    (b)[(i) + 3] = (unsigned char) ( (n) >> 32 );
    (b)[(i) + 4] = (unsigned char) ( (n) >> 24 );
    (b)[(i) + 5] = (unsigned char) ( (n) >> 16 );
    (b)[(i) + 6] = (unsigned char) ( (n) >> 8 );
    (b)[(i) + 7] = (unsigned char) ( (n) );
}

```

Definition at line 53 of file sha4.c.

Referenced by sha4\_finish().

#### 5.131.1.6 #define ROTR(x, n) (SHR(x,n) | (x << (64 - n)))

#### 5.131.1.7 #define S0(x) (ROTR(x, 1) ^ ROTR(x, 8) ^ SHR(x, 7))

#### 5.131.1.8 #define S1(x) (ROTR(x,19) ^ ROTR(x,61) ^ SHR(x, 6))

#### 5.131.1.9 #define S2(x) (ROTR(x,28) ^ ROTR(x,34) ^ ROTR(x,39))

#### 5.131.1.10 #define S3(x) (ROTR(x,14) ^ ROTR(x,18) ^ ROTR(x,41))

#### 5.131.1.11 #define SHR(x, n) (x >> n)

### 5.131.2 Function Documentation

#### 5.131.2.1 void sha4 (unsigned char \* *input*, int *ilen*, unsigned char *output*[64], int *is384*)

Output = SHA-512( input buffer ).

##### Parameters

*input* buffer holding the data

*ilen* length of the input data

*output* SHA-384/512 checksum result

*is384* 0 = use SHA512, 1 = use SHA384

Definition at line 312 of file sha4.c.

References sha4\_finish(), sha4\_starts(), and sha4\_update().

Referenced by sha4\_hmac\_starts().

**5.131.2.2 int sha4\_file (char \* *path*, unsigned char *output*[64], int *is384*)**

Output = SHA-512( file contents ).

**Parameters**

*path* input file name

*output* SHA-384/512 checksum result

*is384* 0 = use SHA512, 1 = use SHA384

**Returns**

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 327 of file sha4.c.

References buf, f, sha4\_finish(), sha4\_starts(), and sha4\_update().

**5.131.2.3 void sha4\_finish (sha4\_context \* *ctx*, unsigned char *output*[64])**

SHA-512 final digest.

**Parameters**

*ctx* SHA-512 context

*output* SHA-384/512 checksum result

Definition at line 276 of file sha4.c.

References int, int64, sha4\_context::is384, PUT\_UINT64\_BE, sha4\_padding, sha4\_update(), sha4\_context::state, and sha4\_context::total.

Referenced by sha4(), sha4\_file(), sha4\_hmac\_finish(), and sha4\_self\_test().

**5.131.2.4 void sha4\_hmac (unsigned char \* *key*, int *keylen*, unsigned char \* *input*, int *ilen*, unsigned char *output*[64], int *is384*)**

Output = HMAC-SHA-512( hmac key, input buffer ).

**Parameters**

*key* HMAC secret key

*keylen* length of the HMAC key

*input* buffer holding the data

*ilen* length of the input data

*output* HMAC-SHA-384/512 result

*is384* 0 = use SHA512, 1 = use SHA384

Definition at line 419 of file sha4.c.

References sha4\_hmac\_finish(), sha4\_hmac\_starts(), and sha4\_hmac\_update().

**5.131.2.5 void sha4\_hmac\_finish (sha4\_context \* *ctx*, unsigned char *output*[64])**

SHA-512 HMAC final digest.

**Parameters**

*ctx* HMAC context

*output* SHA-384/512 HMAC checksum result

Definition at line 399 of file sha4.c.

References sha4\_context::is384, sha4\_context::opad, sha4\_finish(), sha4\_starts(), and sha4\_update().

Referenced by sha4\_hmac(), and sha4\_self\_test().

**5.131.2.6 void sha4\_hmac\_starts (sha4\_context \* *ctx*, unsigned char \* *key*, int *keylen*, int *is384*)**

SHA-512 HMAC context setup.

**Parameters**

*ctx* HMAC context to be initialized

*is384* 0 = use SHA512, 1 = use SHA384

*key* HMAC secret key

*keylen* length of the HMAC key

Definition at line 359 of file sha4.c.

References sha4\_context::ipad, sha4\_context::opad, sha4(), sha4\_starts(), and sha4\_update().

Referenced by sha4\_hmac(), and sha4\_self\_test().

**5.131.2.7 void sha4\_hmac\_update (sha4\_context \* *ctx*, unsigned char \* *input*, int *ilen*)**

SHA-512 HMAC process buffer.

**Parameters**

*ctx* HMAC context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 390 of file sha4.c.

References sha4\_update().

Referenced by sha4\_hmac(), and sha4\_self\_test().

**5.131.2.8 static void sha4\_process (sha4\_context \* *ctx*, unsigned char *data*[128]) [static]**

Definition at line 149 of file sha4.c.

References F, GET\_UINT64\_BE, int64, K, P, S0, S1, and sha4\_context::state.

Referenced by sha4\_update().

### 5.131.2.9 `int sha4_self_test (int verbose)`

Checkup routine.

#### Returns

0 if successful, or 1 if the test failed

Definition at line 654 of file sha4.c.

References `buf`, `sha4_finish()`, `sha4_hmac_finish()`, `sha4_hmac_starts()`, `sha4_hmac_test_buf`, `sha4_hmac_test_buflen`, `sha4_hmac_test_key`, `sha4_hmac_test_keylen`, `sha4_hmac_test_sum`, `sha4_hmac_update()`, `sha4_starts()`, `sha4_test_buf`, `sha4_test_buflen`, `sha4_test_sum`, and `sha4_update()`.

Referenced by `main()`.

### 5.131.2.10 `void sha4_starts (sha4_context * ctx, int is384)`

SHA-512 context setup.

#### Parameters

*ctx* context to be initialized

*is384* 0 = use SHA512, 1 = use SHA384

Definition at line 116 of file sha4.c.

References `sha4_context::is384`, `sha4_context::state`, `sha4_context::total`, and `UL64`.

Referenced by `sha4()`, `sha4_file()`, `sha4_hmac_finish()`, `sha4_hmac_starts()`, and `sha4_self_test()`.

### 5.131.2.11 `void sha4_update (sha4_context * ctx, unsigned char * input, int ilen)`

SHA-512 process buffer.

#### Parameters

*ctx* SHA-512 context

*input* buffer holding the data

*ilen* length of the input data

Definition at line 221 of file sha4.c.

References `sha4_context::buffer`, `int`, `int64`, `sha4_process()`, and `sha4_context::total`.

Referenced by `sha4()`, `sha4_file()`, `sha4_finish()`, `sha4_hmac_finish()`, `sha4_hmac_starts()`, `sha4_hmac_update()`, and `sha4_self_test()`.

## 5.131.3 Variable Documentation

### 5.131.3.1 `const unsigned int64 K[80] [static]`

Definition at line 69 of file sha4.c.



Definition at line 506 of file sha4.c.

Referenced by sha4\_self\_test().

### 5.131.3.5 `const int sha4_hmac_test_keylen[7] [static]`

**Initial value:**

```
{
    20, 4, 20, 25, 20, 131, 131
}
```

Definition at line 521 of file sha4.c.

Referenced by sha4\_self\_test().

### 5.131.3.6 `const unsigned char sha4_hmac_test_sum[14][64] [static]`

Definition at line 552 of file sha4.c.

Referenced by sha4\_self\_test().

### 5.131.3.7 `const unsigned char sha4_padding[128] [static]`

**Initial value:**

```
{
    0x80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
}
```

Definition at line 261 of file sha4.c.

Referenced by sha4\_finish().

### 5.131.3.8 `unsigned char sha4_test_buf[3][113] [static]`

**Initial value:**

```
{
    { "abc" },
    { "abcdefghijklghicdefghijdefghijkefghijklfghijklmghijklmn"
      "hijklmnoijklmnopjklmnopqklmnopqrlmnopqrsmnopqrstnopqrstu" },
    { "" }
}
```

Definition at line 437 of file sha4.c.

Referenced by sha4\_self\_test().



**5.131.3.9   const int sha4\_test\_buflen[3]   [static]**

**Initial value:**

```
{  
    3, 112, 1000  
}
```

Definition at line 445 of file sha4.c.

Referenced by sha4\_self\_test().

**5.131.3.10   const unsigned char sha4\_test\_sum[6][64]   [static]**

Definition at line 450 of file sha4.c.

Referenced by sha4\_self\_test().

## 5.132 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl\_cli.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/debug.h"
#include "xyssl/ssl.h"
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
```

### Functions

- static [int ssl\\_write\\_client\\_hello](#) (ssl\_context \*ssl)
- static [int ssl\\_parse\\_server\\_hello](#) (ssl\_context \*ssl)
- static [int ssl\\_parse\\_server\\_key\\_exchange](#) (ssl\_context \*ssl)
- static [int ssl\\_parse\\_certificate\\_request](#) (ssl\_context \*ssl)
- static [int ssl\\_parse\\_server\\_hello\\_done](#) (ssl\_context \*ssl)
- static [int ssl\\_write\\_client\\_key\\_exchange](#) (ssl\_context \*ssl)
- static [int ssl\\_write\\_certificate\\_verify](#) (ssl\_context \*ssl)
- [int ssl\\_handshake\\_client](#) (ssl\_context \*ssl)

### 5.132.1 Function Documentation

#### 5.132.1.1 [int ssl\\_handshake\\_client](#) (ssl\_context \*ssl)

Definition at line 660 of file `ssl_cli.c`.

References `SSL_CERTIFICATE_REQUEST`, `SSL_CERTIFICATE_VERIFY`, `SSL_CLIENT_CERTIFICATE`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_CLIENT_FINISHED`, `SSL_CLIENT_HELLO`, `SSL_CLIENT_KEY_EXCHANGE`, `SSL_DEBUG_MSG`, `SSL_FLUSH_BUFFERS`, `ssl_flush_output()`, `SSL_HANDSHAKE_OVER`, `SSL_HELLO_REQUEST`, `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `SSL_SERVER_CERTIFICATE`, `SSL_SERVER_CHANGE_CIPHER_SPEC`, `SSL_SERVER_FINISHED`, `SSL_SERVER_HELLO`, `SSL_SERVER_HELLO_DONE`, `SSL_SERVER_KEY_EXCHANGE`, `ssl_write_certificate()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_BAD_INPUT_DATA`.

Referenced by `ssl_handshake()`.

#### 5.132.1.2 [static int ssl\\_parse\\_certificate\\_request](#) (ssl\_context \*ssl) **[static]**

Definition at line 428 of file `ssl_cli.c`.

References `_ssl_context::client_auth`, `_ssl_context::in_msg`, `_ssl_context::in_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HS_CERTIFICATE_REQUEST`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `ssl_handshake_client()`.

**5.132.1.3 static int ssl\_parse\_server\_hello (ssl\_context \* ssl) [static]**

Definition at line 168 of file ssl\_cli.c.

References buf, \_ssl\_session::cipher, \_ssl\_context::ciphers, \_ssl\_session::id, \_ssl\_context::in\_hslen, \_ssl\_context::in\_msg, \_ssl\_context::in\_msgtype, \_ssl\_session::length, \_ssl\_context::minor\_ver, \_ssl\_context::randbytes, \_ssl\_context::resume, \_ssl\_context::session, SSL\_COMPRESS\_NULL, SSL\_DEBUG\_BUF, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, ssl\_derive\_keys(), SSL\_HS\_SERVER\_HELLO, SSL\_MAJOR\_VERSION\_3, SSL\_MINOR\_VERSION\_0, SSL\_MINOR\_VERSION\_1, SSL\_MSG\_HANDSHAKE, ssl\_read\_record(), SSL\_SERVER\_CHANGE\_CIPHER\_SPEC, \_ssl\_session::start, \_ssl\_context::state, XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_HELLO, and XYSSL\_ERR\_SSL\_UNEXPECTED\_MESSAGE.

Referenced by ssl\_handshake\_client().

**5.132.1.4 static int ssl\_parse\_server\_hello\_done (ssl\_context \* ssl) [static]**

Definition at line 471 of file ssl\_cli.c.

References \_ssl\_context::client\_auth, \_ssl\_context::in\_hslen, \_ssl\_context::in\_msg, \_ssl\_context::in\_msgtype, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, SSL\_HS\_SERVER\_HELLO\_DONE, SSL\_MSG\_HANDSHAKE, ssl\_read\_record(), \_ssl\_context::state, XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_HELLO\_DONE, and XYSSL\_ERR\_SSL\_UNEXPECTED\_MESSAGE.

Referenced by ssl\_handshake\_client().

**5.132.1.5 static int ssl\_parse\_server\_key\_exchange (ssl\_context \* ssl) [static]**

Definition at line 312 of file ssl\_cli.c.

References \_ssl\_session::cipher, \_ssl\_context::dhm\_ctx, dhm\_read\_params(), dhm\_context::G, dhm\_context::GY, \_ssl\_context::in\_hslen, \_ssl\_context::in\_msg, \_ssl\_context::in\_msgtype, dhm\_context::len, rsa\_context::len, md5(), md5\_finish(), md5\_starts(), md5\_update(), dhm\_context::P, \_ssl\_context::peer\_cert, \_ssl\_context::randbytes, x509\_cert::rsa, rsa\_pkcs1\_verify(), RSA\_PUBLIC, RSA\_RAW, \_ssl\_context::session, sha1(), sha1\_finish(), sha1\_starts(), sha1\_update(), SSL\_DEBUG\_BUF, SSL\_DEBUG\_MPI, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, SSL\_EDH\_RSA\_AES\_256\_SHA, SSL\_EDH\_RSA\_DES\_168\_SHA, SSL\_HS\_SERVER\_KEY\_EXCHANGE, SSL\_MSG\_HANDSHAKE, ssl\_read\_record(), \_ssl\_context::state, XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_KEY\_EXCHANGE, XYSSL\_ERR\_SSL\_FEATURE\_UNAVAILABLE, and XYSSL\_ERR\_SSL\_UNEXPECTED\_MESSAGE.

Referenced by ssl\_handshake\_client().

**5.132.1.6 static int ssl\_write\_certificate\_verify (ssl\_context \* ssl) [static]**

Definition at line 604 of file ssl\_cli.c.

References \_ssl\_context::client\_auth, rsa\_context::len, \_ssl\_context::out\_msg, \_ssl\_context::out\_msglen, \_ssl\_context::out\_msgtype, \_ssl\_context::rsa\_key, rsa\_pkcs1\_sign(), RSA\_PRIVATE, RSA\_RAW, ssl\_calc\_verify(), SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, SSL\_HS\_CERTIFICATE\_VERIFY, SSL\_MSG\_HANDSHAKE, ssl\_write\_record(), \_ssl\_context::state, and XYSSL\_ERR\_SSL\_PRIVATE\_KEY\_REQUIRED.

Referenced by ssl\_handshake\_client().

**5.132.1.7 static int ssl\_write\_client\_hello (ssl\_context \* ssl) [static]**

Definition at line 33 of file ssl\_cli.c.

References buf, \_ssl\_context::ciphers, \_ssl\_context::f\_rng, \_ssl\_context::hostname, \_ssl\_context::hostname\_len, \_ssl\_session::id, \_ssl\_session::length, \_ssl\_context::major\_ver, \_ssl\_context::max\_major\_ver, \_ssl\_context::max\_minor\_ver, \_ssl\_context::minor\_ver, \_ssl\_context::out\_msg, \_ssl\_context::out\_msglen, \_ssl\_context::out\_msgtype, \_ssl\_context::p\_rng, \_ssl\_context::randbytes, \_ssl\_context::resume, \_ssl\_context::session, SSL\_COMPRESS\_NULL, SSL\_DEBUG\_BUF, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, SSL\_HS\_CLIENT\_HELLO, SSL\_MAJOR\_VERSION\_3, SSL\_MINOR\_VERSION\_0, SSL\_MINOR\_VERSION\_1, SSL\_MSG\_HANDSHAKE, ssl\_write\_record(), \_ssl\_session::start, \_ssl\_context::state, \_ssl\_context::timeout, TLS\_EXT\_SERVERNAME, and TLS\_EXT\_SERVERNAME\_HOSTNAME.

Referenced by ssl\_handshake\_client().

**5.132.1.8 static int ssl\_write\_client\_key\_exchange (ssl\_context \* ssl) [static]**

Definition at line 506 of file ssl\_cli.c.

References \_ssl\_session::cipher, dhm\_calc\_secret(), \_ssl\_context::dhm\_ctx, dhm\_make\_public(), \_ssl\_context::f\_rng, dhm\_context::GX, dhm\_context::K, rsa\_context::len, dhm\_context::len, \_ssl\_context::max\_major\_ver, \_ssl\_context::max\_minor\_ver, \_ssl\_context::minor\_ver, \_ssl\_context::out\_msg, \_ssl\_context::out\_msglen, \_ssl\_context::out\_msgtype, \_ssl\_context::p\_rng, \_ssl\_context::peer\_cert, \_ssl\_context::pmslen, \_ssl\_context::premaster, x509\_cert::rsa, rsa\_pkcs1\_encrypt(), RSA\_PUBLIC, \_ssl\_context::session, SSL\_DEBUG\_MPI, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, ssl\_derive\_keys(), SSL\_EDH\_RSA\_AES\_256\_SHA, SSL\_EDH\_RSA\_DES\_168\_SHA, SSL\_HS\_CLIENT\_KEY\_EXCHANGE, SSL\_MINOR\_VERSION\_0, SSL\_MSG\_HANDSHAKE, ssl\_write\_record(), \_ssl\_context::state, dhm\_context::X, and XYSSL\_ERR\_SSL\_FEATURE\_UNAVAILABLE.

Referenced by ssl\_handshake\_client().

## 5.133 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl\_srv.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/debug.h"
#include "xyssl/ssl.h"
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
```

### Functions

- static [int ssl\\_parse\\_client\\_hello](#) (ssl\_context \*ssl)
- static [int ssl\\_write\\_server\\_hello](#) (ssl\_context \*ssl)
- static [int ssl\\_write\\_certificate\\_request](#) (ssl\_context \*ssl)
- static [int ssl\\_write\\_server\\_key\\_exchange](#) (ssl\_context \*ssl)
- static [int ssl\\_write\\_server\\_hello\\_done](#) (ssl\_context \*ssl)
- static [int ssl\\_parse\\_client\\_key\\_exchange](#) (ssl\_context \*ssl)
- static [int ssl\\_parse\\_certificate\\_verify](#) (ssl\_context \*ssl)
- [int ssl\\_handshake\\_server](#) (ssl\_context \*ssl)

### 5.133.1 Function Documentation

#### 5.133.1.1 [int ssl\\_handshake\\_server](#) (ssl\_context \* ssl)

Definition at line 819 of file ssl\_srv.c.

References [SSL\\_CERTIFICATE\\_REQUEST](#), [SSL\\_CERTIFICATE\\_VERIFY](#), [SSL\\_CLIENT\\_CERTIFICATE](#), [SSL\\_CLIENT\\_CHANGE\\_CIPHER\\_SPEC](#), [SSL\\_CLIENT\\_FINISHED](#), [SSL\\_CLIENT\\_HELLO](#), [SSL\\_CLIENT\\_KEY\\_EXCHANGE](#), [SSL\\_DEBUG\\_MSG](#), [SSL\\_FLUSH\\_BUFFERS](#), [ssl\\_flush\\_output\(\)](#), [SSL\\_HANDSHAKE\\_OVER](#), [SSL\\_HELLO\\_REQUEST](#), [ssl\\_parse\\_certificate\(\)](#), [ssl\\_parse\\_certificate\\_verify\(\)](#), [ssl\\_parse\\_change\\_cipher\\_spec\(\)](#), [ssl\\_parse\\_client\\_hello\(\)](#), [ssl\\_parse\\_client\\_key\\_exchange\(\)](#), [ssl\\_parse\\_finished\(\)](#), [SSL\\_SERVER\\_CERTIFICATE](#), [SSL\\_SERVER\\_CHANGE\\_CIPHER\\_SPEC](#), [SSL\\_SERVER\\_FINISHED](#), [SSL\\_SERVER\\_HELLO](#), [SSL\\_SERVER\\_HELLO\\_DONE](#), [SSL\\_SERVER\\_KEY\\_EXCHANGE](#), [ssl\\_write\\_certificate\(\)](#), [ssl\\_write\\_certificate\\_request\(\)](#), [ssl\\_write\\_change\\_cipher\\_spec\(\)](#), [ssl\\_write\\_finished\(\)](#), [ssl\\_write\\_server\\_hello\(\)](#), [ssl\\_write\\_server\\_hello\\_done\(\)](#), [ssl\\_write\\_server\\_key\\_exchange\(\)](#), [\\_ssl\\_context::state](#), and [XYSSL\\_ERR\\_SSL\\_BAD\\_INPUT\\_DATA](#).

Referenced by [ssl\\_handshake\(\)](#).

#### 5.133.1.2 [static int ssl\\_parse\\_certificate\\_verify](#) (ssl\_context \* ssl) **[static]**

Definition at line 758 of file ssl\_srv.c.

References [\\_ssl\\_context::in\\_hrlen](#), [\\_ssl\\_context::in\\_msg](#), [\\_ssl\\_context::in\\_msgtype](#), [rsa\\_context::len](#), [\\_ssl\\_context::peer\\_cert](#), [\\_x509\\_cert::rsa](#), [rsa\\_pkcs1\\_verify\(\)](#), [RSA\\_PUBLIC](#), [RSA\\_RAW](#), [ssl\\_calc\\_verify\(\)](#), [SSL\\_DEBUG\\_MSG](#), [SSL\\_DEBUG\\_RET](#), [SSL\\_HS\\_CERTIFICATE\\_VERIFY](#), [SSL\\_](#)

MSG\_HANDSHAKE, ssl\_read\_record(), \_ssl\_context::state, and XYSSL\_ERR\_SSL\_BAD\_HS\_CERTIFICATE\_VERIFY.

Referenced by ssl\_handshake\_server().

#### 5.133.1.3 static int ssl\_parse\_client\_hello (ssl\_context \* ssl) [static]

Definition at line 33 of file ssl\_srv.c.

References buf, \_ssl\_session::cipher, \_ssl\_context::ciphers, \_ssl\_context::fin\_md5, \_ssl\_context::fin\_sha1, \_ssl\_session::id, \_ssl\_context::in\_hdr, \_ssl\_context::in\_left, \_ssl\_context::in\_msg, \_ssl\_session::length, \_ssl\_context::major\_ver, \_ssl\_context::max\_major\_ver, \_ssl\_context::max\_minor\_ver, md5\_update(), \_ssl\_context::minor\_ver, \_ssl\_context::randbytes, \_ssl\_context::session, sha1\_update(), SSL\_DEBUG\_BUF, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, ssl\_fetch\_input(), SSL\_HS\_CLIENT\_HELLO, SSL\_MAJOR\_VERSION\_3, SSL\_MINOR\_VERSION\_1, SSL\_MSG\_HANDSHAKE, \_ssl\_context::state, XYSSL\_ERR\_SSL\_BAD\_HS\_CLIENT\_HELLO, and XYSSL\_ERR\_SSL\_NO\_CIPHER\_CHOSEN.

Referenced by ssl\_handshake\_server().

#### 5.133.1.4 static int ssl\_parse\_client\_key\_exchange (ssl\_context \* ssl) [static]

Definition at line 634 of file ssl\_srv.c.

References \_ssl\_session::cipher, dhm\_calc\_secret(), \_ssl\_context::dhm\_ctx, dhm\_read\_public(), \_ssl\_context::f\_rng, dhm\_context::GY, \_ssl\_context::in\_hrlen, \_ssl\_context::in\_msg, \_ssl\_context::in\_msgtype, dhm\_context::K, rsa\_context::len, dhm\_context::len, \_ssl\_context::max\_major\_ver, \_ssl\_context::max\_minor\_ver, \_ssl\_context::minor\_ver, \_ssl\_context::p\_rng, \_ssl\_context::pmslen, \_ssl\_context::premaster, \_ssl\_context::rsa\_key, rsa\_pkcs1\_decrypt(), RSA\_PRIVATE, \_ssl\_context::s\_set, \_ssl\_context::session, SSL\_DEBUG\_MPI, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, ssl\_derive\_keys(), SSL\_EDH\_RSA\_AES\_256\_SHA, SSL\_EDH\_RSA\_DES\_168\_SHA, SSL\_HS\_CLIENT\_KEY\_EXCHANGE, SSL\_MINOR\_VERSION\_0, SSL\_MSG\_HANDSHAKE, ssl\_read\_record(), \_ssl\_context::state, XYSSL\_ERR\_SSL\_BAD\_HS\_CLIENT\_KEY\_EXCHANGE, and XYSSL\_ERR\_SSL\_FEATURE\_UNAVAILABLE.

Referenced by ssl\_handshake\_server().

#### 5.133.1.5 static int ssl\_write\_certificate\_request (ssl\_context \* ssl) [static]

Definition at line 447 of file ssl\_srv.c.

References \_ssl\_context::authmode, buf, \_ssl\_context::ca\_chain, \_x509\_buf::len, \_x509\_cert::next, \_ssl\_context::out\_msg, \_ssl\_context::out\_msglen, \_ssl\_context::out\_msgtype, \_x509\_buf::p, SSL\_DEBUG\_BUF, SSL\_DEBUG\_MSG, SSL\_HS\_CERTIFICATE\_REQUEST, SSL\_MSG\_HANDSHAKE, SSL\_VERIFY\_NONE, ssl\_write\_record(), \_ssl\_context::state, and \_x509\_cert::subject\_raw.

Referenced by ssl\_handshake\_server().

#### 5.133.1.6 static int ssl\_write\_server\_hello (ssl\_context \* ssl) [static]

Definition at line 349 of file ssl\_srv.c.

References buf, \_ssl\_session::cipher, \_ssl\_context::f\_rng, \_ssl\_session::id, \_ssl\_session::length, \_ssl\_context::major\_ver, \_ssl\_context::minor\_ver, \_ssl\_context::out\_msg, \_ssl\_context::out\_msglen, \_ssl\_context::out\_msgtype, \_ssl\_context::p\_rng, \_ssl\_context::randbytes, \_ssl\_context::resume, \_ssl\_context::s\_get, \_ssl\_context::session, SSL\_COMPRESS\_NULL, SSL\_DEBUG\_BUF, SSL\_DEBUG\_

MSG, ssl\_derive\_keys(), SSL\_HS\_SERVER\_HELLO, SSL\_MSG\_HANDSHAKE, SSL\_SERVER\_CHANGE\_CIPHER\_SPEC, ssl\_write\_record(), and \_ssl\_context::state.

Referenced by ssl\_handshake\_server().

#### 5.133.1.7 static int ssl\_write\_server\_hello\_done (ssl\_context \* ssl) [static]

Definition at line 611 of file ssl\_srv.c.

References \_ssl\_context::out\_msg, \_ssl\_context::out\_msglen, \_ssl\_context::out\_msgtype, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, SSL\_HS\_SERVER\_HELLO\_DONE, SSL\_MSG\_HANDSHAKE, ssl\_write\_record(), and \_ssl\_context::state.

Referenced by ssl\_handshake\_server().

#### 5.133.1.8 static int ssl\_write\_server\_key\_exchange (ssl\_context \* ssl) [static]

Definition at line 512 of file ssl\_srv.c.

References \_ssl\_session::cipher, \_ssl\_context::dhm\_ctx, dhm\_make\_params(), \_ssl\_context::f\_rng, dhm\_context::G, dhm\_context::GX, rsa\_context::len, md5(), md5\_finish(), md5\_starts(), md5\_update(), \_ssl\_context::out\_msg, \_ssl\_context::out\_msglen, \_ssl\_context::out\_msgtype, dhm\_context::P, \_ssl\_context::p\_rng, \_ssl\_context::randbytes, \_ssl\_context::rsa\_key, rsa\_pkcs1\_sign(), RSA\_PRIVATE, RSA\_RAW, \_ssl\_context::session, sha1(), sha1\_finish(), sha1\_starts(), sha1\_update(), SSL\_DEBUG\_BUF, SSL\_DEBUG\_MPI, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, SSL\_EDH\_RSA\_AES\_256\_SHA, SSL\_EDH\_RSA\_DES\_168\_SHA, SSL\_HS\_SERVER\_KEY\_EXCHANGE, SSL\_MSG\_HANDSHAKE, ssl\_write\_record(), \_ssl\_context::state, dhm\_context::X, and XYSSL\_ERR\_SSL\_FEATURE\_UNAVAILABLE.

Referenced by ssl\_handshake\_server().

## 5.134 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl\_tls.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/aes.h"
#include "xyssl/arc4.h"
#include "xyssl/des.h"
#include "xyssl/debug.h"
#include "xyssl/ssl.h"
#include <string.h>
#include <stdlib.h>
#include <time.h>
```

### Functions

- static [int](#) [tls1\\_prf](#) (unsigned char \*secret, [int](#) slen, char \*label, unsigned char \*random, [int](#) rlen, unsigned char \*dstbuf, [int](#) dlen)
- [int](#) [ssl\\_derive\\_keys](#) ([ssl\\_context](#) \*ssl)
- void [ssl\\_calc\\_verify](#) ([ssl\\_context](#) \*ssl, unsigned char hash[36])
- static void [ssl\\_mac\\_md5](#) (unsigned char \*secret, unsigned char \*buf, [int](#) len, unsigned char \*ctr, [int](#) type)
- static void [ssl\\_mac\\_sha1](#) (unsigned char \*secret, unsigned char \*buf, [int](#) len, unsigned char \*ctr, [int](#) type)
- static [int](#) [ssl\\_encrypt\\_buf](#) ([ssl\\_context](#) \*ssl)
- static [int](#) [ssl\\_decrypt\\_buf](#) ([ssl\\_context](#) \*ssl)
- [int](#) [ssl\\_fetch\\_input](#) ([ssl\\_context](#) \*ssl, [int](#) nb\_want)
- [int](#) [ssl\\_flush\\_output](#) ([ssl\\_context](#) \*ssl)
- [int](#) [ssl\\_write\\_record](#) ([ssl\\_context](#) \*ssl)
- [int](#) [ssl\\_read\\_record](#) ([ssl\\_context](#) \*ssl)
- [int](#) [ssl\\_write\\_certificate](#) ([ssl\\_context](#) \*ssl)
- [int](#) [ssl\\_parse\\_certificate](#) ([ssl\\_context](#) \*ssl)
- [int](#) [ssl\\_write\\_change\\_cipher\\_spec](#) ([ssl\\_context](#) \*ssl)
- [int](#) [ssl\\_parse\\_change\\_cipher\\_spec](#) ([ssl\\_context](#) \*ssl)
- static void [ssl\\_calc\\_finished](#) ([ssl\\_context](#) \*ssl, unsigned char \*buf, [int](#) from, [md5\\_context](#) \*md5, [sha1\\_context](#) \*sha1)
- [int](#) [ssl\\_write\\_finished](#) ([ssl\\_context](#) \*ssl)
- [int](#) [ssl\\_parse\\_finished](#) ([ssl\\_context](#) \*ssl)
- [int](#) [ssl\\_init](#) ([ssl\\_context](#) \*ssl)

*Initialize an SSL context.*

- void [ssl\\_set\\_endpoint](#) ([ssl\\_context](#) \*ssl, [int](#) endpoint)

*Set the current endpoint type.*

- void [ssl\\_set\\_authmode](#) ([ssl\\_context](#) \*ssl, [int](#) authmode)

*Set the certificate verification mode.*



- void `ssl_set_rng` (`ssl_context` \*ssl, `int`(\*f\_rng)(void \*), void \*p\_rng)  
*Set the random number generator callback.*
- void `ssl_set_dbg` (`ssl_context` \*ssl, void(\*f\_dbg)(void \*, `int`, char \*), void \*p\_dbg)  
*Set the debug callback.*
- void `ssl_set_bio` (`ssl_context` \*ssl, `int`(\*f\_recv)(void \*, unsigned char \*, `int`), void \*p\_recv, `int`(\*f\_send)(void \*, unsigned char \*, `int`), void \*p\_send)  
*Set the underlying BIO read and write callbacks.*
- void `ssl_set_scb` (`ssl_context` \*ssl, `int`(\*s\_get)(`ssl_context` \*), `int`(\*s\_set)(`ssl_context` \*))  
*Set the session callbacks (server-side only).*
- void `ssl_set_session` (`ssl_context` \*ssl, `int` resume, `int` timeout, `ssl_session` \*session)  
*Set the session resuming flag, timeout and data.*
- void `ssl_set_ciphers` (`ssl_context` \*ssl, `int` \*ciphers)  
*Set the list of allowed ciphersuites.*
- void `ssl_set_ca_chain` (`ssl_context` \*ssl, `x509_cert` \*ca\_chain, char \*peer\_cn)  
*Set the data required to verify peer certificate.*
- void `ssl_set_own_cert` (`ssl_context` \*ssl, `x509_cert` \*own\_cert, `rsa_context` \*rsa\_key)  
*Set own certificate and private key.*
- `int` `ssl_set_dh_param` (`ssl_context` \*ssl, char \*dhm\_P, char \*dhm\_G)  
*Set the Diffie-Hellman public P and G values, read as hexadecimal strings (server-side only).*
- `int` `ssl_set_hostname` (`ssl_context` \*ssl, char \*hostname)  
*Set hostname for ServerName TLS Extension.*
- `int` `ssl_get_bytes_avail` (`ssl_context` \*ssl)  
*Return the number of data bytes available to read.*
- `int` `ssl_get_verify_result` (`ssl_context` \*ssl)  
*Return the result of the certificate verification.*
- char \* `ssl_get_cipher` (`ssl_context` \*ssl)  
*Return the name of the current cipher.*
- `int` `ssl_handshake` (`ssl_context` \*ssl)  
*Perform the SSL handshake.*
- `int` `ssl_read` (`ssl_context` \*ssl, unsigned char \*buf, `int` len)  
*Read at most 'len' application data bytes.*
- `int` `ssl_write` (`ssl_context` \*ssl, unsigned char \*buf, `int` len)  
*Write exactly 'len' application data bytes.*
- `int` `ssl_close_notify` (`ssl_context` \*ssl)

*Notify the peer that the connection is being closed.*

- void `ssl_free` (`ssl_context` \*`ssl`)

*Free an SSL context.*

## Variables

- int `ssl_default_ciphers` []

### 5.134.1 Function Documentation

#### 5.134.1.1 static void `ssl_calc_finished` (`ssl_context` \*`ssl`, unsigned char \*`buf`, int `from`, `md5_context` \*`md5`, `sha1_context` \*`sha1`) [static]

Definition at line 1344 of file `ssl_tls.c`.

References `_ssl_session::master`, `md5_finish()`, `md5_starts()`, `md5_update()`, `_ssl_context::minor_ver`, `_ssl_context::session`, `sha1_finish()`, `sha1_starts()`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_IS_CLIENT`, `SSL_MINOR_VERSION_0`, `sha1_context::state`, `md5_context::state`, and `tls1_prf()`.

Referenced by `ssl_parse_finished()`, and `ssl_write_finished()`.

#### 5.134.1.2 void `ssl_calc_verify` (`ssl_context` \*`ssl`, unsigned char `hash`[36])

Definition at line 335 of file `ssl_tls.c`.

References `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_session::master`, `md5()`, `md5_finish()`, `md5_starts()`, `md5_update()`, `_ssl_context::minor_ver`, `_ssl_context::session`, `sha1()`, `sha1_finish()`, `sha1_starts()`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, and `SSL_MINOR_VERSION_0`.

Referenced by `ssl_parse_certificate_verify()`, and `ssl_write_certificate_verify()`.

#### 5.134.1.3 int `ssl_close_notify` (`ssl_context` \*`ssl`)

*Notify the peer that the connection is being closed.*

Definition at line 1904 of file `ssl_tls.c`.

References `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_ALERT_CLOSE_NOTIFY`, `SSL_ALERT_WARNING`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_flush_output()`, `SSL_HANDSHAKE_OVER`, `SSL_MSG_ALERT`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `main()`, and `ssl_test()`.

#### 5.134.1.4 static int `ssl_decrypt_buf` (`ssl_context` \*`ssl`) [static]

Definition at line 555 of file `ssl_tls.c`.

References `aes_crypt_cbc()`, `AES_DECRYPT`, `arc4_crypt()`, `_ssl_context::ctx_dec`, `des3_crypt_cbc()`, `DES_DECRYPT`, `_ssl_context::in_ctr`, `_ssl_context::in_hdr`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `_ssl_context::iv_dec`, `_ssl_context::ivlen`, `_ssl_context::mac_dec`, `_ssl_context::maclen`, `md5_hmac()`, `_ssl_context::minlen`, `_ssl_context::minor_ver`, `_ssl_context::nb_`

zero, sha1\_hmac(), SSL\_DEBUG\_BUF, SSL\_DEBUG\_MSG, ssl\_mac\_md5(), ssl\_mac\_sha1(), SSL\_MINOR\_VERSION\_0, XYSSL\_ERR\_SSL\_FEATURE\_UNAVAILABLE, and XYSSL\_ERR\_SSL\_INVALID\_MAC.

Referenced by ssl\_read\_record().

#### 5.134.1.5 int ssl\_derive\_keys (ssl\_context \* ssl)

Definition at line 106 of file ssl\_tls.c.

References aes\_setkey\_dec(), aes\_setkey\_enc(), arc4\_setup(), \_ssl\_session::cipher, \_ssl\_context::ctx\_dec, \_ssl\_context::ctx\_enc, des3\_set3key\_dec(), des3\_set3key\_enc(), \_ssl\_context::endpoint, \_ssl\_context::iv\_dec, \_ssl\_context::iv\_enc, \_ssl\_context::ivlen, \_ssl\_context::keylen, \_ssl\_context::mac\_dec, \_ssl\_context::mac\_enc, \_ssl\_context::maclen, \_ssl\_session::master, md5(), md5\_finish(), md5\_starts(), md5\_update(), \_ssl\_context::minlen, \_ssl\_context::minor\_ver, \_ssl\_context::pmslen, \_ssl\_context::premaster, \_ssl\_context::randbytes, \_ssl\_context::resume, \_ssl\_context::session, sha1(), sha1\_finish(), sha1\_starts(), sha1\_update(), SSL\_DEBUG\_BUF, SSL\_DEBUG\_MSG, SSL\_EDH\_RSA\_AES\_256\_SHA, SSL\_EDH\_RSA\_DES\_168\_SHA, ssl\_get\_cipher(), SSL\_IS\_CLIENT, SSL\_MINOR\_VERSION\_0, SSL\_RSA\_AES\_128\_SHA, SSL\_RSA\_AES\_256\_SHA, SSL\_RSA\_DES\_168\_SHA, SSL\_RSA\_RC4\_128\_MD5, SSL\_RSA\_RC4\_128\_SHA, tls1\_prf(), and XYSSL\_ERR\_SSL\_FEATURE\_UNAVAILABLE.

Referenced by ssl\_parse\_client\_key\_exchange(), ssl\_parse\_server\_hello(), ssl\_write\_client\_key\_exchange(), and ssl\_write\_server\_hello().

#### 5.134.1.6 static int ssl\_encrypt\_buf (ssl\_context \* ssl) [static]

Definition at line 448 of file ssl\_tls.c.

References aes\_crypt\_cbc(), AES\_ENCRYPT, arc4\_crypt(), \_ssl\_context::ctx\_enc, des3\_crypt\_cbc(), DES\_ENCRYPT, \_ssl\_context::iv\_enc, \_ssl\_context::ivlen, \_ssl\_context::mac\_enc, \_ssl\_context::maclen, md5\_hmac(), \_ssl\_context::minor\_ver, \_ssl\_context::out\_ctr, \_ssl\_context::out\_msg, \_ssl\_context::out\_msglen, \_ssl\_context::out\_msgtype, sha1\_hmac(), SSL\_DEBUG\_BUF, SSL\_DEBUG\_MSG, ssl\_mac\_md5(), ssl\_mac\_sha1(), SSL\_MINOR\_VERSION\_0, and XYSSL\_ERR\_SSL\_FEATURE\_UNAVAILABLE.

Referenced by ssl\_write\_record().

#### 5.134.1.7 int ssl\_fetch\_input (ssl\_context \* ssl, int nb\_want)

Definition at line 727 of file ssl\_tls.c.

References \_ssl\_context::f\_recv, \_ssl\_context::in\_hdr, \_ssl\_context::in\_left, \_ssl\_context::p\_recv, SSL\_DEBUG\_MSG, and SSL\_DEBUG\_RET.

Referenced by ssl\_parse\_client\_hello(), and ssl\_read\_record().

#### 5.134.1.8 int ssl\_flush\_output (ssl\_context \* ssl)

Definition at line 756 of file ssl\_tls.c.

References buf, \_ssl\_context::f\_send, \_ssl\_context::out\_hdr, \_ssl\_context::out\_left, \_ssl\_context::out\_msglen, \_ssl\_context::p\_send, SSL\_DEBUG\_MSG, and SSL\_DEBUG\_RET.

Referenced by ssl\_close\_notify(), ssl\_handshake\_client(), ssl\_handshake\_server(), ssl\_write(), and ssl\_write\_record().

**5.134.1.9 void ssl\_free (ssl\_context \* ssl)**

Free an SSL context.

Definition at line 1938 of file ssl\_tls.c.

References `_ssl_context::dhm_ctx`, `dhm_free()`, `_ssl_context::hostname`, `_ssl_context::hostname_len`, `_ssl_context::in_ctr`, `_ssl_context::out_ctr`, `_ssl_context::peer_cert`, `SSL_BUFFER_LEN`, `SSL_DEBUG_MSG`, and `x509_free()`.

Referenced by `main()`, and `ssl_test()`.

**5.134.1.10 int ssl\_get\_bytes\_avail (ssl\_context \* ssl)**

Return the number of data bytes available to read.

**Parameters**

*ssl* SSL context

**Returns**

how many bytes are available in the read buffer

Definition at line 1691 of file ssl\_tls.c.

References `_ssl_context::in_msglen`, and `_ssl_context::in_offt`.

**5.134.1.11 char\* ssl\_get\_cipher (ssl\_context \* ssl)**

Return the name of the current cipher.

**Parameters**

*ssl* SSL context

**Returns**

a string containing the cipher name

Definition at line 1701 of file ssl\_tls.c.

References `_ssl_session::cipher`, `_ssl_context::session`, `SSL_EDH_RSA_AES_256_SHA`, `SSL_EDH_RSA_DES_168_SHA`, `SSL_RSA_AES_128_SHA`, `SSL_RSA_AES_256_SHA`, `SSL_RSA_DES_168_SHA`, `SSL_RSA_RC4_128_MD5`, and `SSL_RSA_RC4_128_SHA`.

Referenced by `main()`, and `ssl_derive_keys()`.

**5.134.1.12 int ssl\_get\_verify\_result (ssl\_context \* ssl)**

Return the result of the certificate verification.

**Parameters**

*ssl* SSL context

**Returns**

0 if successful, or a combination of: BADCERT\_EXPIRED BADCERT\_REVOKED BADCERT\_CN\_MISMATCH BADCERT\_NOT\_TRUSTED

Definition at line 1696 of file ssl\_tls.c.

References `_ssl_context::verify_result`.

Referenced by `main()`.

**5.134.1.13 int ssl\_handshake (ssl\_context \* ssl)**

Perform the SSL handshake.

**Parameters**

*ssl* SSL context

**Returns**

0 if successful, XYSSL\_ERR\_NET\_TRY\_AGAIN, or a specific SSL error code.

Definition at line 1767 of file ssl\_tls.c.

References `_ssl_context::endpoint`, `SSL_DEBUG_MSG`, `ssl_handshake_client()`, `ssl_handshake_server()`, `SSL_IS_CLIENT`, `SSL_IS_SERVER`, and `XYSSL_ERR_SSL_FEATURE_UNAVAILABLE`.

Referenced by `main()`, `ssl_read()`, and `ssl_write()`.

**5.134.1.14 int ssl\_init (ssl\_context \* ssl)**

Initialize an SSL context.

**Parameters**

*ssl* SSL context

**Returns**

0 if successful, or 1 if memory allocation failed

Definition at line 1542 of file ssl\_tls.c.

References `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::hostname`, `_ssl_context::hostname_len`, `_ssl_context::in_ctr`, `_ssl_context::in_hdr`, `_ssl_context::in_msg`, `md5_starts()`, `_ssl_context::out_ctr`, `_ssl_context::out_hdr`, `_ssl_context::out_msg`, `sha1_starts()`, `SSL_BUFFER_LEN`, and `SSL_DEBUG_MSG`.

Referenced by `main()`, and `ssl_test()`.

**5.134.1.15 static void ssl\_mac\_md5 (unsigned char \* secret, unsigned char \* buf, int len, unsigned char \* ctr, int type) [static]**

Definition at line 387 of file ssl\_tls.c.

References `md5()`, `md5_finish()`, `md5_starts()`, and `md5_update()`.

Referenced by `ssl_decrypt_buf()`, and `ssl_encrypt_buf()`.

#### 5.134.1.16 **static void ssl\_mac\_sha1 (unsigned char \* *secret*, unsigned char \* *buf*, int *len*, unsigned char \* *ctr*, int *type*) [static]**

Definition at line 416 of file ssl\_tls.c.

References sha1(), sha1\_finish(), sha1\_starts(), and sha1\_update().

Referenced by ssl\_decrypt\_buf(), and ssl\_encrypt\_buf().

#### 5.134.1.17 **int ssl\_parse\_certificate (ssl\_context \* *ssl*)**

Definition at line 1140 of file ssl\_tls.c.

References \_ssl\_context::authmode, \_ssl\_context::ca\_chain, \_ssl\_context::endpoint, \_ssl\_context::in\_hslen, \_ssl\_context::in\_msg, \_ssl\_context::in\_msglen, \_ssl\_context::in\_msgtype, int, \_ssl\_context::minor\_ver, \_ssl\_context::peer\_cert, \_ssl\_context::peer\_cn, SSL\_ALERT\_NO\_CERTIFICATE, SSL\_ALERT\_WARNING, SSL\_DEBUG\_CRT, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, SSL\_HS\_CERTIFICATE, SSL\_IS\_SERVER, SSL\_MINOR\_VERSION\_0, SSL\_MSG\_ALERT, SSL\_MSG\_HANDSHAKE, ssl\_read\_record(), SSL\_VERIFY\_NONE, SSL\_VERIFY\_OPTIONAL, SSL\_VERIFY\_REQUIRED, \_ssl\_context::state, \_ssl\_context::verify\_result, x509parse\_cert(), x509parse\_verify(), XYSSL\_ERR\_SSL\_BAD\_HS\_CERTIFICATE, XYSSL\_ERR\_SSL\_CA\_CHAIN\_REQUIRED, XYSSL\_ERR\_SSL\_NO\_CLIENT\_CERTIFICATE, and XYSSL\_ERR\_SSL\_UNEXPECTED\_MESSAGE.

Referenced by ssl\_handshake\_client(), and ssl\_handshake\_server().

#### 5.134.1.18 **int ssl\_parse\_change\_cipher\_spec (ssl\_context \* *ssl*)**

Definition at line 1311 of file ssl\_tls.c.

References \_ssl\_context::do\_crypt, \_ssl\_context::in\_msg, \_ssl\_context::in\_msglen, \_ssl\_context::in\_msgtype, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, SSL\_MSG\_CHANGE\_CIPHER\_SPEC, ssl\_read\_record(), \_ssl\_context::state, XYSSL\_ERR\_SSL\_BAD\_HS\_CHANGE\_CIPHER\_SPEC, and XYSSL\_ERR\_SSL\_UNEXPECTED\_MESSAGE.

Referenced by ssl\_handshake\_client(), and ssl\_handshake\_server().

#### 5.134.1.19 **int ssl\_parse\_finished (ssl\_context \* *ssl*)**

Definition at line 1480 of file ssl\_tls.c.

References buf, \_ssl\_context::do\_crypt, \_ssl\_context::endpoint, \_ssl\_context::fin\_md5, \_ssl\_context::fin\_sha1, \_ssl\_context::in\_hslen, \_ssl\_context::in\_msg, \_ssl\_context::in\_msgtype, md5(), \_ssl\_context::minor\_ver, \_ssl\_context::resume, sha1(), ssl\_calc\_finished(), SSL\_CLIENT\_CHANGE\_CIPHER\_SPEC, SSL\_DEBUG\_MSG, SSL\_DEBUG\_RET, SSL\_HANDSHAKE\_OVER, SSL\_HS\_FINISHED, SSL\_IS\_CLIENT, SSL\_IS\_SERVER, SSL\_MINOR\_VERSION\_0, SSL\_MSG\_HANDSHAKE, ssl\_read\_record(), \_ssl\_context::state, XYSSL\_ERR\_SSL\_BAD\_HS\_FINISHED, and XYSSL\_ERR\_SSL\_UNEXPECTED\_MESSAGE.

Referenced by ssl\_handshake\_client(), and ssl\_handshake\_server().

#### 5.134.1.20 **int ssl\_read (ssl\_context \* *ssl*, unsigned char \* *buf*, int *len*)**

Read at most 'len' application data bytes.

**Parameters**

*ssl* SSL context  
*buf* buffer that will hold the data  
*len* how many bytes must be read

**Returns**

This function returns the number of bytes read, or a negative error code.

Definition at line 1791 of file ssl\_tls.c.

References `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `_ssl_context::in_offt`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_handshake()`, `SSL_HANDSHAKE_OVER`, `SSL_MSG_APPLICATION_DATA`, `ssl_read_record()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `main()`, and `ssl_test()`.

**5.134.1.21 int ssl\_read\_record (ssl\_context \* ssl)**

Definition at line 842 of file ssl\_tls.c.

References `_ssl_context::do_crypt`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::in_hdr`, `_ssl_context::in_hslen`, `_ssl_context::in_left`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `_ssl_context::major_ver`, `md5_update()`, `_ssl_context::minlen`, `_ssl_context::minor_ver`, `sha1_update()`, `SSL_ALERT_CLOSE_NOTIFY`, `SSL_ALERT_FATAL`, `SSL_ALERT_WARNING`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_decrypt_buf()`, `ssl_fetch_input()`, `SSL_MAX_CONTENT_LEN`, `SSL_MINOR_VERSION_0`, `SSL_MINOR_VERSION_1`, `SSL_MSG_ALERT`, `SSL_MSG_HANDSHAKE`, `XYSSL_ERR_SSL_FATAL_ALERT_MESSAGE`, `XYSSL_ERR_SSL_INVALID_RECORD`, and `XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY`.

Referenced by `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, and `ssl_read()`.

**5.134.1.22 void ssl\_set\_authmode (ssl\_context \* ssl, int authmode)**

Set the certificate verification mode.

**Parameters**

*ssl* SSL context  
*mode* can be:

`SSL_VERIFY_NONE`: peer certificate is not checked (default), this is insecure and SHOULD be avoided.

`SSL_VERIFY_OPTIONAL`: peer certificate is checked, however the handshake continues even if verification failed; [ssl\\_get\\_verify\\_result\(\)](#) can be called after the handshake is complete.

`SSL_VERIFY_REQUIRED`: peer *must* present a valid certificate, handshake is aborted if verification failed.

Definition at line 1589 of file ssl\_tls.c.

References `_ssl_context::authmode`.

Referenced by `main()`, and `ssl_test()`.

**5.134.1.23** `void ssl_set_bio (ssl_context * ssl, int(*)(void *, unsigned char *, int) f_recv, void * p_recv, int(*)(void *, unsigned char *, int) f_send, void * p_send)`

Set the underlying BIO read and write callbacks.

#### Parameters

*ssl* SSL context  
*f\_recv* read callback  
*p\_recv* read parameter  
*f\_send* write callback  
*p\_send* write parameter

Definition at line 1610 of file `ssl_tls.c`.

References `_ssl_context::f_recv`, `_ssl_context::f_send`, `_ssl_context::p_recv`, and `_ssl_context::p_send`.

Referenced by `main()`, and `ssl_test()`.

**5.134.1.24** `void ssl_set_ca_chain (ssl_context * ssl, x509_cert * ca_chain, char * peer_cn)`

Set the data required to verify peer certificate.

#### Parameters

*ssl* SSL context  
*ca\_chain* trusted CA chain  
*peer\_cn* expected peer CommonName (or NULL)

#### Note

TODO: add two more parameters: depth and `crl`

Definition at line 1641 of file `ssl_tls.c`.

References `_ssl_context::ca_chain`, and `_ssl_context::peer_cn`.

Referenced by `main()`, and `ssl_test()`.

**5.134.1.25** `void ssl_set_ciphers (ssl_context * ssl, int * ciphers)`

Set the list of allowed ciphersuites.

#### Parameters

*ssl* SSL context  
*ciphers* 0-terminated list of allowed ciphers

Definition at line 1636 of file `ssl_tls.c`.

References `_ssl_context::ciphers`.

Referenced by `main()`, and `ssl_test()`.



**5.134.1.26 void ssl\_set\_dbg (ssl\_context \* ssl, void(\*) (void \*, int, char \*) f\_dbg, void \* p\_dbg)**

Set the debug callback.

**Parameters**

*ssl* SSL context

*f\_dbg* debug function

*p\_dbg* debug parameter

Definition at line 1602 of file ssl\_tls.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

Referenced by `main()`, and `ssl_test()`.

**5.134.1.27 int ssl\_set\_dh\_param (ssl\_context \* ssl, char \* dhm\_P, char \* dhm\_G)**

Set the Diffie-Hellman public P and G values, read as hexadecimal strings (server-side only).

**Parameters**

*ssl* SSL context

*dhm\_P* Diffie-Hellman-Merkle modulus

*dhm\_G* Diffie-Hellman-Merkle generator

**Returns**

0 if successful

Definition at line 1655 of file ssl\_tls.c.

References `_ssl_context::dhm_ctx`, `dhm_context::G`, `mpi_read_string()`, `dhm_context::P`, and `SSL_DEBUG_RET`.

Referenced by `main()`, and `ssl_test()`.

**5.134.1.28 void ssl\_set\_endpoint (ssl\_context \* ssl, int endpoint)**

Set the current endpoint type.

**Parameters**

*ssl* SSL context

*endpoint* must be `SSL_IS_CLIENT` or `SSL_IS_SERVER`

Definition at line 1584 of file ssl\_tls.c.

References `_ssl_context::endpoint`.

Referenced by `main()`, and `ssl_test()`.

**5.134.1.29** `int ssl_set_hostname (ssl_context * ssl, char * hostname)`

Set hostname for ServerName TLS Extension.

**Parameters**

*ssl* SSL context

*hostname* the server hostname

**Returns**

0 if successful

Definition at line 1674 of file `ssl_tls.c`.

References `_ssl_context::hostname`, `_ssl_context::hostname_len`, and `XYSSL_ERR_SSL_BAD_INPUT_DATA`.

Referenced by `main()`.

**5.134.1.30** `void ssl_set_own_cert (ssl_context * ssl, x509_cert * own_cert, rsa_context * rsa_key)`

Set own certificate and private key.

**Parameters**

*ssl* SSL context

*own\_cert* own public certificate

*rsa\_key* own private RSA key

Definition at line 1648 of file `ssl_tls.c`.

References `_ssl_context::own_cert`, and `_ssl_context::rsa_key`.

Referenced by `main()`, and `ssl_test()`.

**5.134.1.31** `void ssl_set_rng (ssl_context * ssl, int(*) (void *) f_rng, void * p_rng)`

Set the random number generator callback.

**Parameters**

*ssl* SSL context

*f\_rng* RNG function

*p\_rng* RNG parameter

Definition at line 1594 of file `ssl_tls.c`.

References `_ssl_context::f_rng`, and `_ssl_context::p_rng`.

Referenced by `main()`, and `ssl_test()`.

**5.134.1.32 void ssl\_set\_scb (ssl\_context \* *ssl*, int(\*)(ssl\_context \*) *s\_get*, int(\*)(ssl\_context \*) *s\_set*)**

Set the session callbacks (server-side only).

**Parameters**

*ssl* SSL context  
*s\_get* session get callback  
*s\_set* session set callback

Definition at line 1620 of file ssl\_tls.c.

References \_ssl\_context::s\_get, and \_ssl\_context::s\_set.

Referenced by main().

**5.134.1.33 void ssl\_set\_session (ssl\_context \* *ssl*, int *resume*, int *timeout*, ssl\_session \* *session*)**

Set the session resuming flag, timeout and data.

**Parameters**

*ssl* SSL context  
*resume* if 0 (default), the session will not be resumed  
*timeout* session timeout in seconds, or 0 (no timeout)  
*session* session context

Definition at line 1628 of file ssl\_tls.c.

References \_ssl\_context::resume, \_ssl\_context::session, and \_ssl\_context::timeout.

Referenced by main(), and ssl\_test().

**5.134.1.34 int ssl\_write (ssl\_context \* *ssl*, unsigned char \* *buf*, int *len*)**

Write exactly 'len' application data bytes.

**Parameters**

*ssl* SSL context  
*buf* buffer holding the data  
*len* how many bytes must be written

**Returns**

This function returns the number of bytes written, or a negative error code.

**Note**

When this function returns XYSSL\_ERR\_NET\_TRY\_AGAIN, it must be called later with the *\*same\** arguments, until it returns a positive value.

Definition at line 1857 of file `ssl_tls.c`.

References `_ssl_context::out_left`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_flush_output()`, `ssl_handshake()`, `SSL_HANDSHAKE_OVER`, `SSL_MAX_CONTENT_LEN`, `SSL_MSG_APPLICATION_DATA`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `main()`, and `ssl_test()`.

#### 5.134.1.35 `int ssl_write_certificate(ssl_context *ssl)`

Definition at line 1044 of file `ssl_tls.c`.

References `_ssl_context::client_auth`, `_ssl_context::endpoint`, `_x509_buf::len`, `_ssl_context::minor_ver`, `_x509_cert::next`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `_ssl_context::own_cert`, `_x509_buf::p`, `_x509_cert::raw`, `SSL_ALERT_NO_CERTIFICATE`, `SSL_ALERT_WARNING`, `SSL_DEBUG_CRT`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HS_CERTIFICATE`, `SSL_IS_CLIENT`, `SSL_MAX_CONTENT_LEN`, `SSL_MINOR_VERSION_0`, `SSL_MSG_ALERT`, `SSL_MSG_HANDSHAKE`, `ssl_write_record()`, `_ssl_context::state`, `XYSSL_ERR_SSL_CERTIFICATE_REQUIRED`, and `XYSSL_ERR_SSL_CERTIFICATE_TOO_LARGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

#### 5.134.1.36 `int ssl_write_change_cipher_spec(ssl_context *ssl)`

Definition at line 1287 of file `ssl_tls.c`.

References `_ssl_context::do_crypt`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_MSG_CHANGE_CIPHER_SPEC`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

#### 5.134.1.37 `int ssl_write_finished(ssl_context *ssl)`

Definition at line 1433 of file `ssl_tls.c`.

References `_ssl_context::do_crypt`, `_ssl_context::endpoint`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `md5()`, `_ssl_context::minor_ver`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `_ssl_context::resume`, `sha1()`, `ssl_calc_finished()`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HANDSHAKE_OVER`, `SSL_HS_FINISHED`, `SSL_IS_CLIENT`, `SSL_MINOR_VERSION_0`, `SSL_MSG_HANDSHAKE`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

#### 5.134.1.38 `int ssl_write_record(ssl_context *ssl)`

Definition at line 786 of file `ssl_tls.c`.

References `_ssl_context::do_crypt`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::major_ver`, `md5_update()`, `_ssl_context::minor_ver`, `_ssl_context::out_hdr`, `_ssl_context::out_left`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_encrypt_buf()`, `ssl_flush_output()`, and `SSL_MSG_HANDSHAKE`.

Referenced by `ssl_close_notify()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

**5.134.1.39** `static int tls1_prf (unsigned char * secret, int slen, char * label, unsigned char * random, int rlen, unsigned char * dstbuf, int dlen) [static]`

Definition at line 46 of file `ssl_tls.c`.

References `md5_hmac()`, `S1`, `S2`, `sha1_hmac()`, and `XYSSL_ERR_SSL_BAD_INPUT_DATA`.

Referenced by `ssl_calc_finished()`, and `ssl_derive_keys()`.

## 5.134.2 Variable Documentation

**5.134.2.1** `int ssl_default_ciphers[ ]`

**Initial value:**

```
{  
  
    SSL_EDH_RSA_AES_256_SHA,  
  
    SSL_EDH_RSA_DES_168_SHA,  
  
    SSL_RSA_AES_128_SHA,  
    SSL_RSA_AES_256_SHA,  
  
    SSL_RSA_DES_168_SHA,  
  
    SSL_RSA_RC4_128_SHA,  
    SSL_RSA_RC4_128_MD5,  
  
    0  
}
```

Definition at line 1739 of file `ssl_tls.c`.

Referenced by `main()`, and `ssl_test()`.

## 5.135 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/timing.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/timing.h"
#include <unistd.h>
#include <sys/types.h>
#include <sys/time.h>
#include <signal.h>
#include <time.h>
```

### Data Structures

- struct [\\_hr\\_time](#)

### Functions

- unsigned long [hardclock](#) (void)  
*Return the CPU cycle counter value.*
- unsigned long [get\\_timer](#) (struct [hr\\_time](#) \*val, int reset)  
*Return the elapsed time in milliseconds.*
- static void [sighandler](#) (int signum)
- void [set\\_alarm](#) (int seconds)  
*Setup an alarm clock.*
- void [m\\_sleep](#) (int milliseconds)  
*Sleep for a certain amount of time.*

### Variables

- static int [hardclock\\_init](#) = 0
- static struct timeval [tv\\_init](#)
- int [alarmed](#) = 0

### 5.135.1 Function Documentation

#### 5.135.1.1 unsigned long [get\\_timer](#) (struct [hr\\_time](#) \* val, int reset)

Return the elapsed time in milliseconds.

#### Parameters

*val* points to a timer structure

*reset* if set to 1, the timer is restarted

Definition at line 225 of file timing.c.

References `_hr_time::start`.

Referenced by `ssl_test()`.

#### 5.135.1.2 unsigned long hardclock (void)

Return the CPU cycle counter value.

Definition at line 155 of file timing.c.

References `tv_init`.

Referenced by `main()`.

#### 5.135.1.3 void m\_sleep (int milliseconds)

Sleep for a certain amount of time.

Definition at line 258 of file timing.c.

#### 5.135.1.4 void set\_alarm (int seconds)

Setup an alarm clock.

##### Parameters

*seconds* delay before the "alarmed" flag is set

Definition at line 251 of file timing.c.

References `sighandler()`.

Referenced by `main()`.

#### 5.135.1.5 static void sighandler (int signum) [static]

Definition at line 245 of file timing.c.

Referenced by `set_alarm()`.

### 5.135.2 Variable Documentation

#### 5.135.2.1 int alarmed = 0

Definition at line 179 of file timing.c.

Referenced by `main()`.

#### 5.135.2.2 int hardclock\_init = 0 [static]

Definition at line 152 of file timing.c.

**5.135.2.3 struct timeval tv\_init [static]**

Definition at line 153 of file timing.c.

Referenced by hardclock().



## 5.136 /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/x509parse.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/x509.h"
#include "xyssl/base64.h"
#include "xyssl/des.h"
#include "xyssl/md2.h"
#include "xyssl/md4.h"
#include "xyssl/md5.h"
#include "xyssl/sha1.h"
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include "xyssl/certs.h"
```

### Functions

- static [int](#) [asn1\\_get\\_len](#) (unsigned char \*\*p, unsigned char \*end, [int](#) \*len)
- static [int](#) [asn1\\_get\\_tag](#) (unsigned char \*\*p, unsigned char \*end, [int](#) \*len, [int](#) tag)
- static [int](#) [asn1\\_get\\_bool](#) (unsigned char \*\*p, unsigned char \*end, [int](#) \*val)
- static [int](#) [asn1\\_get\\_int](#) (unsigned char \*\*p, unsigned char \*end, [int](#) \*val)
- static [int](#) [asn1\\_get\\_mpi](#) (unsigned char \*\*p, unsigned char \*end, [mpi](#) \*X)
- static [int](#) [x509\\_get\\_version](#) (unsigned char \*\*p, unsigned char \*end, [int](#) \*ver)
- static [int](#) [x509\\_get\\_serial](#) (unsigned char \*\*p, unsigned char \*end, [x509\\_buf](#) \*serial)
- static [int](#) [x509\\_get\\_alg](#) (unsigned char \*\*p, unsigned char \*end, [x509\\_buf](#) \*alg)
- static [int](#) [x509\\_get\\_name](#) (unsigned char \*\*p, unsigned char \*end, [x509\\_name](#) \*cur)
- static [int](#) [x509\\_get\\_dates](#) (unsigned char \*\*p, unsigned char \*end, [x509\\_time](#) \*from, [x509\\_time](#) \*to)
- static [int](#) [x509\\_get\\_pubkey](#) (unsigned char \*\*p, unsigned char \*end, [x509\\_buf](#) \*pk\_alg\_oid, [mpi](#) \*N, [mpi](#) \*E)
- static [int](#) [x509\\_get\\_sig](#) (unsigned char \*\*p, unsigned char \*end, [x509\\_buf](#) \*sig)
- static [int](#) [x509\\_get\\_uid](#) (unsigned char \*\*p, unsigned char \*end, [x509\\_buf](#) \*uid, [int](#) n)
- static [int](#) [x509\\_get\\_ext](#) (unsigned char \*\*p, unsigned char \*end, [x509\\_buf](#) \*ext, [int](#) \*ca\_istrue, [int](#) \*max\_pathlen)
- [int](#) [x509parse\\_crt](#) ([x509\\_cert](#) \*chain, unsigned char \*buf, [int](#) buflen)
 

*Parse one or more certificates and add them to the chained list.*
- [int](#) [x509parse\\_crtfile](#) ([x509\\_cert](#) \*chain, char \*path)
 

*Load one or more certificates and add them to the chained list.*
- static [int](#) [x509\\_get\\_iv](#) (unsigned char \*s, unsigned char iv[8])
- static void [x509\\_des3\\_decrypt](#) (unsigned char des3\_iv[8], unsigned char \*buf, [int](#) buflen, unsigned char \*pwd, [int](#) pwdden)

- `int x509parse_key` (`rsa_context` \*rsa, unsigned char \*buf, int buflen, unsigned char \*pwd, int pwrlen)  
*Parse a private RSA key.*
- `int x509parse_keyfile` (`rsa_context` \*rsa, char \*path, char \*pwd)  
*Load and parse a private RSA key.*
- `int x509parse_dn_gets` (char \*buf, char \*end, `x509_name` \*dn)  
*Store the certificate DN in printable form into buf; no more than (end - buf) characters will be written.*
- char \* `x509parse_cert_info` (char \*prefix, `x509_cert` \*crt)  
*Returns an informational string about the certificate.*
- `int x509parse_expired` (`x509_cert` \*crt)  
*Return 0 if the certificate is still valid, or BADCERT\_EXPIRED.*
- static void `x509_hash` (unsigned char \*in, int len, int alg, unsigned char \*out)
- `int x509parse_verify` (`x509_cert` \*crt, `x509_cert` \*trust\_ca, char \*cn, int \*flags)  
*Verify the certificate signature.*
- void `x509_free` (`x509_cert` \*crt)  
*Unallocate all certificate data.*
- `int x509_self_test` (int verbose)  
*Checkup routine.*

### 5.136.1 Function Documentation

#### 5.136.1.1 static int asn1\_get\_bool (unsigned char \*\*p, unsigned char \*end, int \*val) [static]

Definition at line 108 of file x509parse.c.

References ASN1\_BOOLEAN, asn1\_get\_tag(), and XYSSL\_ERR\_ASN1\_INVALID\_LENGTH.

Referenced by x509\_get\_ext().

#### 5.136.1.2 static int asn1\_get\_int (unsigned char \*\*p, unsigned char \*end, int \*val) [static]

Definition at line 126 of file x509parse.c.

References asn1\_get\_tag(), ASN1\_INTEGER, and XYSSL\_ERR\_ASN1\_INVALID\_LENGTH.

Referenced by x509\_get\_ext(), x509\_get\_version(), and x509parse\_key().

#### 5.136.1.3 static int asn1\_get\_len (unsigned char \*\*p, unsigned char \*end, int \*len) [static]

Definition at line 52 of file x509parse.c.

References XYSSL\_ERR\_ASN1\_INVALID\_LENGTH, and XYSSL\_ERR\_ASN1\_OUT\_OF\_DATA.

Referenced by asn1\_get\_tag(), x509\_get\_name(), and x509\_get\_serial().

**5.136.1.4 static int asn1\_get\_mpi (unsigned char \*\**p*, unsigned char \**end*, mpi \**X*) [static]**

Definition at line 149 of file x509parse.c.

References `asn1_get_tag()`, `ASN1_INTEGER`, and `mpi_read_binary()`.

Referenced by `x509_get_pubkey()`, and `x509parse_key()`.

**5.136.1.5 static int asn1\_get\_tag (unsigned char \*\**p*, unsigned char \**end*, int \**len*, int *tag*) [static]**

Definition at line 93 of file x509parse.c.

References `asn1_get_len()`, `XYSSL_ERR_ASN1_OUT_OF_DATA`, and `XYSSL_ERR_ASN1_UNEXPECTED_TAG`.

Referenced by `asn1_get_bool()`, `asn1_get_int()`, `asn1_get_mpi()`, `x509_get_alg()`, `x509_get_dates()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_sig()`, `x509_get_uid()`, `x509_get_version()`, `x509parse_crt()`, and `x509parse_key()`.

**5.136.1.6 static void x509\_des3\_decrypt (unsigned char *des3\_iv*[8], unsigned char \**buf*, int *buflen*, unsigned char \**pwd*, int *pwdlen*) [static]**

Definition at line 1042 of file x509parse.c.

References `des3_crypt_cbc()`, `des3_set3key_dec()`, `DES_DECRYPT`, `md5_finish()`, `md5_starts()`, and `md5_update()`.

Referenced by `x509parse_key()`.

**5.136.1.7 void x509\_free (x509\_cert \**crt*)**

Unallocate all certificate data.

Definition at line 1613 of file x509parse.c.

References `_x509_cert::issuer`, `_x509_buf::len`, `_x509_cert::next`, `_x509_name::next`, `_x509_buf::p`, `_x509_cert::raw`, `_x509_cert::rsa`, `rsa_free()`, and `_x509_cert::subject`.

Referenced by `main()`, `ssl_free()`, `ssl_test()`, `x509_self_test()`, and `x509parse_crt()`.

**5.136.1.8 static int x509\_get\_alg (unsigned char \*\**p*, unsigned char \**end*, x509\_buf \**alg*) [static]**

Definition at line 229 of file x509parse.c.

References `ASN1_CONSTRUCTED`, `asn1_get_tag()`, `ASN1_NULL`, `ASN1_OID`, `ASN1_SEQUENCE`, `_x509_buf::len`, `_x509_buf::p`, `_x509_buf::tag`, `XYSSL_ERR_ASN1_LENGTH_MISMATCH`, and `XYSSL_ERR_X509_CERT_INVALID_ALG`.

Referenced by `x509_get_pubkey()`, and `x509parse_crt()`.

**5.136.1.9 static int x509\_get\_dates (unsigned char \*\**p*, unsigned char \**end*, x509\_time \**from*, x509\_time \**to*) [static]**

Definition at line 358 of file x509parse.c.

References ASN1\_CONSTRUCTED, asn1\_get\_tag(), ASN1\_SEQUENCE, ASN1\_UTC\_TIME, \_x509\_time::day, \_x509\_time::hour, \_x509\_time::min, \_x509\_time::mon, \_x509\_time::sec, XYSSL\_ERR\_ASN1\_LENGTH\_MISMATCH, XYSSL\_ERR\_X509\_CERT\_INVALID\_DATE, and \_x509\_time::year.

Referenced by x509parse\_crt().

**5.136.1.10** `static int x509_get_ext (unsigned char **p, unsigned char *end, x509_buf *ext, int *ca_istrue, int *max_pathlen) [static]`

Definition at line 530 of file x509parse.c.

References ASN1\_CONSTRUCTED, ASN1\_CONTEXT\_SPECIFIC, asn1\_get\_bool(), asn1\_get\_int(), asn1\_get\_tag(), ASN1\_OCTET\_STRING, ASN1\_SEQUENCE, \_x509\_buf::len, \_x509\_buf::p, \_x509\_buf::tag, XYSSL\_ERR\_ASN1\_LENGTH\_MISMATCH, XYSSL\_ERR\_ASN1\_UNEXPECTED\_TAG, and XYSSL\_ERR\_X509\_CERT\_INVALID\_EXTENSIONS.

Referenced by x509parse\_crt().

**5.136.1.11** `static int x509_get_iv (unsigned char *s, unsigned char iv[8]) [static]`

Definition at line 1018 of file x509parse.c.

References XYSSL\_ERR\_X509\_KEY\_INVALID\_ENC\_IV.

Referenced by x509parse\_key().

**5.136.1.12** `static int x509_get_name (unsigned char **p, unsigned char *end, x509_name *cur) [static]`

Definition at line 276 of file x509parse.c.

References ASN1\_BMP\_STRING, ASN1\_CONSTRUCTED, asn1\_get\_len(), asn1\_get\_tag(), ASN1\_IA5\_STRING, ASN1\_OID, ASN1\_PRINTABLE\_STRING, ASN1\_SEQUENCE, ASN1\_SET, ASN1\_T61\_STRING, ASN1\_UNIVERSAL\_STRING, ASN1\_UTF8\_STRING, \_x509\_buf::len, \_x509\_name::next, \_x509\_name::oid, \_x509\_buf::p, \_x509\_buf::tag, \_x509\_name::val, XYSSL\_ERR\_ASN1\_LENGTH\_MISMATCH, XYSSL\_ERR\_ASN1\_OUT\_OF\_DATA, XYSSL\_ERR\_ASN1\_UNEXPECTED\_TAG, and XYSSL\_ERR\_X509\_CERT\_INVALID\_NAME.

Referenced by x509parse\_crt().

**5.136.1.13** `static int x509_get_pubkey (unsigned char **p, unsigned char *end, x509_buf *pk_alg_oid, mpi *N, mpi *E) [static]`

Definition at line 421 of file x509parse.c.

References ASN1\_BIT\_STRING, ASN1\_CONSTRUCTED, asn1\_get\_mpi(), asn1\_get\_tag(), ASN1\_SEQUENCE, \_x509\_buf::len, OID\_PKCS1\_RSA, \_x509\_buf::p, x509\_get\_alg(), XYSSL\_ERR\_ASN1\_LENGTH\_MISMATCH, XYSSL\_ERR\_ASN1\_OUT\_OF\_DATA, XYSSL\_ERR\_X509\_CERT\_INVALID\_PUBKEY, and XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_PK\_ALG.

Referenced by x509parse\_crt().

**5.136.1.14 static int x509\_get\_serial (unsigned char \*\* *p*, unsigned char \* *end*, x509\_buf \* *serial*) [static]**

Definition at line 198 of file x509parse.c.

References ASN1\_CONTEXT\_SPECIFIC, asn1\_get\_len(), ASN1\_INTEGER, ASN1\_PRIMITIVE, \_x509\_buf::len, \_x509\_buf::p, \_x509\_buf::tag, XYSSL\_ERR\_ASN1\_OUT\_OF\_DATA, XYSSL\_ERR\_ASN1\_UNEXPECTED\_TAG, and XYSSL\_ERR\_X509\_CERT\_INVALID\_SERIAL.

Referenced by x509parse\_crt().

**5.136.1.15 static int x509\_get\_sig (unsigned char \*\* *p*, unsigned char \* *end*, x509\_buf \* *sig*) [static]**

Definition at line 476 of file x509parse.c.

References ASN1\_BIT\_STRING, asn1\_get\_tag(), \_x509\_buf::len, \_x509\_buf::p, \_x509\_buf::tag, and XYSSL\_ERR\_X509\_CERT\_INVALID\_SIGNATURE.

Referenced by x509parse\_crt().

**5.136.1.16 static int x509\_get\_uid (unsigned char \*\* *p*, unsigned char \* *end*, x509\_buf \* *uid*, int *n*) [static]**

Definition at line 501 of file x509parse.c.

References ASN1\_CONSTRUCTED, ASN1\_CONTEXT\_SPECIFIC, asn1\_get\_tag(), \_x509\_buf::len, \_x509\_buf::p, \_x509\_buf::tag, and XYSSL\_ERR\_ASN1\_UNEXPECTED\_TAG.

Referenced by x509parse\_crt().

**5.136.1.17 static int x509\_get\_version (unsigned char \*\* *p*, unsigned char \* *end*, int \* *ver*) [static]**

Definition at line 168 of file x509parse.c.

References ASN1\_CONSTRUCTED, ASN1\_CONTEXT\_SPECIFIC, asn1\_get\_int(), asn1\_get\_tag(), XYSSL\_ERR\_ASN1\_LENGTH\_MISMATCH, XYSSL\_ERR\_ASN1\_UNEXPECTED\_TAG, and XYSSL\_ERR\_X509\_CERT\_INVALID\_VERSION.

Referenced by x509parse\_crt().

**5.136.1.18 static void x509\_hash (unsigned char \* *in*, int *len*, int *alg*, unsigned char \* *out*) [static]**

Definition at line 1481 of file x509parse.c.

References md2(), md4(), md5(), RSA\_MD2, RSA\_MD4, RSA\_MD5, RSA\_SHA1, and sha1().

Referenced by x509parse\_verify().

**5.136.1.19 int x509\_self\_test (int *verbose*)**

Checkup routine.

**Returns**

0 if successful, or 1 if the test failed

Definition at line 1675 of file x509parse.c.

References `rsa_free()`, `test_ca_cert`, `test_ca_key`, `test_ca_pwd`, `test_cli_cert`, `x509_free()`, `x509parse_cert()`, `x509parse_key()`, and `x509parse_verify()`.

Referenced by `main()`.

**5.136.1.20 char\* x509parse\_cert\_info (char \* prefix, x509\_cert \* crt)**

Returns an informational string about the certificate.

Definition at line 1399 of file x509parse.c.

References `buf`, `_x509_time::day`, `_x509_time::hour`, `_x509_cert::issuer`, `_x509_buf::len`, `_x509_time::min`, `_x509_time::mon`, `mpi::n`, `rsa_context::N`, `_x509_buf::p`, `_x509_cert::rsa`, `RSA_MD2`, `RSA_MD4`, `RSA_MD5`, `RSA_SHA1`, `_x509_time::sec`, `_x509_cert::serial`, `_x509_cert::sig_oid1`, `_x509_cert::subject`, `_x509_cert::valid_from`, `_x509_cert::valid_to`, `_x509_cert::version`, `x509parse_dn_gets()`, and `_x509_time::year`.

Referenced by `debug_print_cert()`, and `main()`.

**5.136.1.21 int x509parse\_cert (x509\_cert \* crt, unsigned char \* buf, int buflen)**

Parse one or more certificates and add them to the chained list.

**Parameters**

*chain* points to the start of the chain  
*buf* buffer holding the certificate data  
*buflen* size of the buffer

**Returns**

0 if successful, or a specific X509 error code

Definition at line 647 of file x509parse.c.

References `ASN1_CONSTRUCTED`, `asn1_get_tag()`, `ASN1_SEQUENCE`, `base64_decode()`, `_x509_cert::ca_istrue`, `rsa_context::E`, `_x509_cert::issuer`, `_x509_cert::issuer_id`, `_x509_cert::issuer_raw`, `rsa_context::len`, `_x509_buf::len`, `_x509_cert::max_pathlen`, `mpi_size()`, `rsa_context::N`, `_x509_cert::next`, `OID_PKCS1`, `_x509_buf::p`, `_x509_cert::pk_oid`, `_x509_cert::raw`, `_x509_cert::rsa`, `rsa_check_pubkey()`, `_x509_cert::serial`, `_x509_cert::sig`, `_x509_cert::sig_oid1`, `_x509_cert::sig_oid2`, `_x509_cert::subject`, `_x509_cert::subject_id`, `_x509_cert::subject_raw`, `_x509_cert::tbs`, `_x509_cert::v3_ext`, `_x509_cert::valid_from`, `_x509_cert::valid_to`, `_x509_cert::version`, `x509_free()`, `x509_get_alg()`, `x509_get_dates()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_serial()`, `x509_get_sig()`, `x509_get_uid()`, `x509_get_version()`, `x509parse_cert()`, `XYSSL_ERR_ASN1_LENGTH_MISMATCH`, `XYSSL_ERR_BASE64_INVALID_CHARACTER`, `XYSSL_ERR_X509_CERT_INVALID_FORMAT`, `XYSSL_ERR_X509_CERT_INVALID_PEM`, `XYSSL_ERR_X509_CERT_SIG_MISMATCH`, `XYSSL_ERR_X509_CERT_UNKNOWN_SIG_ALG`, and `XYSSL_ERR_X509_CERT_UNKNOWN_VERSION`.

Referenced by `main()`, `ssl_parse_certificate()`, `ssl_test()`, `x509_self_test()`, `x509parse_cert()`, and `x509parse_crtdfile()`.

**5.136.1.22 int x509parse\_crtfile (x509\_cert \* *crt*, char \* *path*)**

Load one or more certificates and add them to the chained list.

**Parameters**

*chain* points to the start of the chain  
*path* filename to read the certificates from

**Returns**

0 if successful, or a specific X509 error code

Definition at line 979 of file x509parse.c.

References buf, f, and x509parse\_crt().

**5.136.1.23 int x509parse\_dn\_gets (char \* *buf*, char \* *end*, x509\_name \* *dn*)**

Store the certificate DN in printable form into buf; no more than (end - buf) characters will be written.

Definition at line 1316 of file x509parse.c.

References \_x509\_buf::len, \_x509\_name::next, \_x509\_name::oid, OID\_PKCS9, OID\_X520, \_x509\_buf::p, PKCS9\_EMAIL, \_x509\_name::val, X520\_COMMON\_NAME, X520\_COUNTRY, X520\_LOCALITY, X520\_ORG\_UNIT, X520\_ORGANIZATION, and X520\_STATE.

Referenced by x509parse\_cert\_info().

**5.136.1.24 int x509parse\_expired (x509\_cert \* *crt*)**

Return 0 if the certificate is still valid, or BADCERT\_EXPIRED.

Definition at line 1458 of file x509parse.c.

References BADCERT\_EXPIRED, \_x509\_time::day, \_x509\_time::mon, \_x509\_cert::valid\_to, and \_x509\_time::year.

Referenced by x509parse\_verify().

**5.136.1.25 int x509parse\_key (rsa\_context \* *rsa*, unsigned char \* *buf*, int *buflen*, unsigned char \* *pwd*, int *pwdlen*)**

Parse a private RSA key.

**Parameters**

*rsa* RSA context to be initialized  
*buf* input buffer  
*buflen* size of the buffer  
*pwd* password for decryption (optional)  
*pwdlen* size of the password

**Returns**

0 if successful, or a specific X509 error code

Definition at line 1082 of file x509parse.c.

References ASN1\_CONSTRUCTED, asn1\_get\_int(), asn1\_get\_mpi(), asn1\_get\_tag(), ASN1\_SEQUENCE, base64\_decode(), rsa\_context::D, rsa\_context::DP, rsa\_context::DQ, rsa\_context::E, rsa\_context::len, mpi\_size(), rsa\_context::N, rsa\_context::P, rsa\_context::Q, rsa\_context::QP, rsa\_check\_privkey(), rsa\_free(), rsa\_context::ver, x509\_des3\_decrypt(), x509\_get\_iv(), XYSSL\_ERR\_ASN1\_LENGTH\_MISMATCH, XYSSL\_ERR\_BASE64\_INVALID\_CHARACTER, XYSSL\_ERR\_X509\_FEATURE\_UNAVAILABLE, XYSSL\_ERR\_X509\_KEY\_INVALID\_ENC\_IV, XYSSL\_ERR\_X509\_KEY\_INVALID\_FORMAT, XYSSL\_ERR\_X509\_KEY\_INVALID\_PEM, XYSSL\_ERR\_X509\_KEY\_INVALID\_VERSION, XYSSL\_ERR\_X509\_KEY\_PASSWORD\_MISMATCH, XYSSL\_ERR\_X509\_KEY\_PASSWORD\_REQUIRED, and XYSSL\_ERR\_X509\_KEY\_UNKNOWN\_ENC\_ALG.

Referenced by main(), ssl\_test(), x509\_self\_test(), and x509parse\_keyfile().

### 5.136.1.26 int x509parse\_keyfile (rsa\_context \*rsa, char \*path, char \*password)

Load and parse a private RSA key.

#### Parameters

- rsa* RSA context to be initialized
- path* filename to read the private key from
- pwd* password to decrypt the file (can be NULL)

#### Returns

0 if successful, or a specific X509 error code

Definition at line 1269 of file x509parse.c.

References buf, f, and x509parse\_key().

### 5.136.1.27 int x509parse\_verify (x509\_cert \*crt, x509\_cert \*trust\_ca, char \*cn, int \*flags)

Verify the certificate signature.

#### Parameters

- crt* a certificate to be verified
- trust\_ca* the trusted CA chain
- cn* expected Common Name (can be set to NULL if the CN must not be verified)
- flags* result of the verification

#### Returns

0 if successful or XYSSL\_ERR\_X509\_SIG\_VERIFY\_FAILED, in which case \*flags will have one or more of the following values set: BADCERT\_EXPIRED -- BADCERT\_REVOKED -- BADCERT\_CN\_MISMATCH -- BADCERT\_NOT\_TRUSTED

#### Note

TODO: add two arguments, depth and crl



Definition at line 1503 of file x509parse.c.

References BADCERT\_CN\_MISMATCH, BADCERT\_NOT\_TRUSTED, \_x509\_cert::ca\_istrue, cur, \_x509\_cert::issuer\_raw, \_x509\_buf::len, \_x509\_cert::max\_pathlen, \_x509\_cert::next, \_x509\_name::next, \_x509\_name::oid, OID\_CN, \_x509\_buf::p, \_x509\_cert::rsa, rsa\_pkcs1\_verify(), RSA\_PUBLIC, \_x509\_cert::sig, \_x509\_cert::sig\_oid1, \_x509\_cert::subject, \_x509\_cert::subject\_raw, \_x509\_cert::tbs, \_x509\_name::val, \_x509\_cert::version, x509\_hash(), x509parse\_expired(), and XYSSL\_ERR\_X509\_CERT\_VERIFY\_FAILED.

Referenced by ssl\_parse\_certificate(), and x509\_self\_test().

## 5.137 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/aes/aescript2.c File Reference

```
#include <sys/types.h>
#include <unistd.h>
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include "xyssl/aes.h"
#include "xyssl/sha2.h"
```

### Defines

- `#define` [MODE\\_ENCRYPT](#) 0
- `#define` [MODE\\_DECRYPT](#) 1
- `#define` [USAGE](#)

### Functions

- `int` [main](#) (`int` argc, `char` \*argv[])

#### 5.137.1 Define Documentation

##### 5.137.1.1 `#define` [MODE\\_DECRYPT](#) 1

Definition at line 42 of file `aescript2.c`.

##### 5.137.1.2 `#define` [MODE\\_ENCRYPT](#) 0

Definition at line 41 of file `aescript2.c`.

##### 5.137.1.3 `#define` [USAGE](#)

#### Value:

```
"\n  aescript2 <mode> <input filename> <output filename> <key>\n" \
"\n    <mode>: 0 = encrypt, 1 = decrypt\n" \
"\n  example: aescript2 0 file file.aes hex:E76B2413958B00E193\n" \
"\n"
```

Definition at line 44 of file `aescript2.c`.

Referenced by `main()`.

## **5.137.2 Function Documentation**

### **5.137.2.1 int main (int *argc*, char \* *argv*[])**

Definition at line 50 of file aesencrypt2.c.

References aes\_crypt\_ecb(), AES\_DECRYPT, AES\_ENCRYPT, aes\_setkey\_dec(), aes\_setkey\_enc(), int, MODE\_DECRYPT, MODE\_ENCRYPT, sha2\_finish(), sha2\_hmac\_finish(), sha2\_hmac\_starts(), sha2\_hmac\_update(), sha2\_starts(), sha2\_update(), and USAGE.

## 5.138 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/hello.c File Reference

```
#include <stdio.h>
#include "xyssl/md5.h"
```

### Functions

- [int main](#) (void)

### 5.138.1 Function Documentation

#### 5.138.1.1 int main (void)

Definition at line 29 of file hello.c.

References [md5\(\)](#).

## 5.139 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/md5sum.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/md5.h"
```

### Functions

- static [int md5\\_wrapper](#) (char \*filename, unsigned char \*sum)
- static [int md5\\_print](#) (char \*filename)
- static [int md5\\_check](#) (char \*filename)
- [int main](#) (int argc, char \*argv[ ])

### 5.139.1 Function Documentation

#### 5.139.1.1 [int main](#) (int *argc*, char \* *argv*[ ])

Definition at line 131 of file md5sum.c.

References [md5\\_check\(\)](#), and [md5\\_print\(\)](#).

#### 5.139.1.2 [static int md5\\_check](#) (char \* *filename*) [[static](#)]

Definition at line 58 of file md5sum.c.

References [buf](#), [f](#), and [md5\\_wrapper\(\)](#).

Referenced by [main\(\)](#).

#### 5.139.1.3 [static int md5\\_print](#) (char \* *filename*) [[static](#)]

Definition at line 43 of file md5sum.c.

References [md5\\_wrapper\(\)](#).

Referenced by [main\(\)](#).

#### 5.139.1.4 [static int md5\\_wrapper](#) (char \* *filename*, unsigned char \* *sum*) [[static](#)]

Definition at line 30 of file md5sum.c.

References [md5\\_file\(\)](#).

Referenced by [md5\\_check\(\)](#), and [md5\\_print\(\)](#).

## 5.140 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/sha1sum.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/sha1.h"
```

### Functions

- static [int sha1\\_wrapper](#) (char \*filename, unsigned char \*sum)
- static [int sha1\\_print](#) (char \*filename)
- static [int sha1\\_check](#) (char \*filename)
- [int main](#) (int argc, char \*argv[ ])

### 5.140.1 Function Documentation

#### 5.140.1.1 [int main](#) (int *argc*, char \* *argv*[ ])

Definition at line 131 of file sha1sum.c.

References [sha1\\_check\(\)](#), and [sha1\\_print\(\)](#).

#### 5.140.1.2 [static int sha1\\_check](#) (char \* *filename*) [[static](#)]

Definition at line 58 of file sha1sum.c.

References [buf](#), [f](#), and [sha1\\_wrapper\(\)](#).

Referenced by [main\(\)](#).

#### 5.140.1.3 [static int sha1\\_print](#) (char \* *filename*) [[static](#)]

Definition at line 43 of file sha1sum.c.

References [sha1\\_wrapper\(\)](#).

Referenced by [main\(\)](#).

#### 5.140.1.4 [static int sha1\\_wrapper](#) (char \* *filename*, unsigned char \* *sum*) [[static](#)]

Definition at line 30 of file sha1sum.c.

References [sha1\\_file\(\)](#).

Referenced by [sha1\\_check\(\)](#), and [sha1\\_print\(\)](#).

## 5.141 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/sha2sum.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/sha2.h"
```

### Functions

- static [int sha2\\_wrapper](#) (char \*filename, unsigned char \*sum)
- static [int sha2\\_print](#) (char \*filename)
- static [int sha2\\_check](#) (char \*filename)
- [int main](#) (int argc, char \*argv[ ])

### 5.141.1 Function Documentation

#### 5.141.1.1 [int main](#) (int *argc*, char \* *argv*[ ])

Definition at line 131 of file sha2sum.c.

References [sha2\\_check\(\)](#), and [sha2\\_print\(\)](#).

#### 5.141.1.2 [static int sha2\\_check](#) (char \* *filename*) [[static](#)]

Definition at line 58 of file sha2sum.c.

References [buf](#), [f](#), and [sha2\\_wrapper\(\)](#).

Referenced by [main\(\)](#).

#### 5.141.1.3 [static int sha2\\_print](#) (char \* *filename*) [[static](#)]

Definition at line 43 of file sha2sum.c.

References [sha2\\_wrapper\(\)](#).

Referenced by [main\(\)](#).

#### 5.141.1.4 [static int sha2\\_wrapper](#) (char \* *filename*, unsigned char \* *sum*) [[static](#)]

Definition at line 30 of file sha2sum.c.

References [sha2\\_file\(\)](#).

Referenced by [sha2\\_check\(\)](#), and [sha2\\_print\(\)](#).

## 5.142 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh\_client.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/net.h"
#include "xyssl/aes.h"
#include "xyssl/dhm.h"
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
#include "xyssl/havege.h"
```

### Defines

- `#define SERVER_NAME "localhost"`
- `#define SERVER_PORT 11999`

### Functions

- `int main (void)`

#### 5.142.1 Define Documentation

##### 5.142.1.1 `#define SERVER_NAME "localhost"`

Definition at line 35 of file `dh_client.c`.

Referenced by `main()`.

##### 5.142.1.2 `#define SERVER_PORT 11999`

Definition at line 36 of file `dh_client.c`.

Referenced by `main()`.

#### 5.142.2 Function Documentation

##### 5.142.2.1 `int main (void)`

Definition at line 38 of file `dh_client.c`.

References `aes_crypt_ecb()`, `AES_DECRYPT`, `aes_setkey_dec()`, `buf`, `dhm_calc_secret()`, `dhm_free()`, `dhm_make_public()`, `dhm_read_params()`, `rsa_context::E`, `f`, `havege_init()`, `havege_rand()`, `dhm_context::len`, `rsa_context::len`, `mpi_msb()`, `mpi_read_file()`, `rsa_context::N`, `net_close()`, `net_connect()`, `net_recv()`, `net_send()`, `rsa_free()`, `rsa_init()`, `rsa_pkcs1_verify()`, `RSA_PKCS_V15`, `RSA_PUBLIC`, `RSA_SHA1`, `server_fd`, `SERVER_NAME`, `SERVER_PORT`, and `sha1()`.



## 5.143 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh\_genprime.c File Reference

```
#include <stdio.h>
#include "xyssl/bignum.h"
#include "xyssl/config.h"
#include "xyssl/havege.h"
```

### Defines

- `#define DH_P_SIZE 1024`
- `#define GENERATOR "4"`

### Functions

- `int main (void)`

#### 5.143.1 Define Documentation

##### 5.143.1.1 `#define DH_P_SIZE 1024`

Definition at line 35 of file dh\_genprime.c.

Referenced by main().

##### 5.143.1.2 `#define GENERATOR "4"`

Definition at line 36 of file dh\_genprime.c.

Referenced by main().

#### 5.143.2 Function Documentation

##### 5.143.2.1 `int main (void)`

Definition at line 38 of file dh\_genprime.c.

References DH\_P\_SIZE, GENERATOR, havege\_init(), havege\_rand(), mpi\_div\_int(), mpi\_free(), mpi\_gen\_prime(), mpi\_init(), mpi\_is\_prime(), mpi\_read\_string(), mpi\_sub\_int(), mpi\_write\_file(), and P.

## 5.144 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh\_server.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/net.h"
#include "xyssl/aes.h"
#include "xyssl/dhm.h"
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
#include "xyssl/havege.h"
```

### Defines

- `#define SERVER_PORT 11999`
- `#define PLAINTEXT "==Hello there!=="`

### Functions

- `int main (void)`

#### 5.144.1 Define Documentation

##### 5.144.1.1 `#define PLAINTEXT "==Hello there!=="`

Definition at line 36 of file `dh_server.c`.

Referenced by `main()`.

##### 5.144.1.2 `#define SERVER_PORT 11999`

Definition at line 35 of file `dh_server.c`.

#### 5.144.2 Function Documentation

##### 5.144.2.1 `int main (void)`

Definition at line 38 of file `dh_server.c`.

References `aes_crypt_ecb()`, `AES_ENCRYPT`, `aes_setkey_enc()`, `buf`, `client_fd`, `rsa_context::D`, `dhm_calc_secret()`, `dhm_free()`, `dhm_make_params()`, `dhm_read_public()`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `f`, `dhm_context::G`, `havege_init()`, `havege_rand()`, `dhm_context::len`, `rsa_context::len`, `mpi_msb()`, `mpi_read_file()`, `rsa_context::N`, `net_accept()`, `net_bind()`, `net_close()`, `net_recv()`, `net_send()`, `dhm_context::P`, `rsa_context::P`, `PLAINTEXT`, `rsa_context::Q`, `rsa_context::QP`, `rsa_free()`, `rsa_init()`, `rsa_pkcs1_sign()`, `RSA_PKCS_V15`, `RSA_PRIVATE`, `RSA_SHA1`, `SERVER_PORT`, and `sha1()`.

## 5.145 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/mpi\_demo.c File Reference

```
#include <stdio.h>
#include "xyssl/bignum.h"
```

### Functions

- [int main](#) (void)

#### 5.145.1 Function Documentation

##### 5.145.1.1 int main (void)

Definition at line 29 of file mpi\_demo.c.

References [mpi\\_exp\\_mod\(\)](#), [mpi\\_free\(\)](#), [mpi\\_init\(\)](#), [mpi\\_inv\\_mod\(\)](#), [mpi\\_mul\\_mpi\(\)](#), [mpi\\_read\\_string\(\)](#), [mpi\\_sub\\_int\(\)](#), [mpi\\_write\\_file\(\)](#), and [P](#).

## 5.146 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa\_genkey.c File Reference

```
#include <stdio.h>
#include "xyssl/havege.h"
#include "xyssl/bignum.h"
#include "xyssl/x509.h"
#include "xyssl/rsa.h"
```

### Defines

- `#define KEY_SIZE 1024`
- `#define EXPONENT 65537`

### Functions

- `int main (void)`

#### 5.146.1 Define Documentation

##### 5.146.1.1 `#define EXPONENT 65537`

Definition at line 33 of file `rsa_genkey.c`.

##### 5.146.1.2 `#define KEY_SIZE 1024`

Definition at line 32 of file `rsa_genkey.c`.

#### 5.146.2 Function Documentation

##### 5.146.2.1 `int main (void)`

Definition at line 35 of file `rsa_genkey.c`.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `EXPONENT`, `havege_init()`, `havege_rand()`, `KEY_SIZE`, `mpi_write_file()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_free()`, `rsa_gen_key()`, `rsa_init()`, and `RSA_PKCS_V15`.

## 5.147 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa\_sign.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
```

### Functions

- `int main (int argc, char *argv[])`

#### 5.147.1 Function Documentation

##### 5.147.1.1 `int main (int argc, char *argv[])`

Definition at line 31 of file `rsa_sign.c`.

References `buf`, `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `f`, `rsa_context::len`, `mpi_msb()`, `mpi_read_file()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_init()`, `rsa_pkcs1_sign()`, `RSA_PKCS_V15`, `RSA_PRIVATE`, `RSA_SHA1`, and `sha1_file()`.

## 5.148 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa\_verify.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
```

### Functions

- `int main (int argc, char *argv[])`

#### 5.148.1 Function Documentation

##### 5.148.1.1 `int main (int argc, char * argv[])`

Definition at line 31 of file `rsa_verify.c`.

References `buf`, `rsa_context::E`, `f`, `rsa_context::len`, `mpi_msb()`, `mpi_read_file()`, `rsa_context::N`, `rsa_init()`, `rsa_pkcs1_verify()`, `RSA_PKCS_V15`, `RSA_PUBLIC`, `RSA_SHA1`, and `sha1_file()`.

## 5.149 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl\_client1.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/net.h"
#include "xyssl/ssl.h"
#include "xyssl/havege.h"
```

### Defines

- #define [SERVER\\_PORT](#) 443
- #define [SERVER\\_NAME](#) "xyssl.org"
- #define [GET\\_REQUEST](#)
- #define [DEBUG\\_LEVEL](#) 0

### Functions

- void [my\\_debug](#) (void \*ctx, [int](#) level, char \*str)
- [int](#) [main](#) (void)

#### 5.149.1 Define Documentation

##### 5.149.1.1 #define [DEBUG\\_LEVEL](#) 0

Definition at line 42 of file ssl\_client1.c.

Referenced by [my\\_debug\(\)](#).

##### 5.149.1.2 #define [GET\\_REQUEST](#)

###### Value:

```
"GET /hello/ HTTP/1.1\r\n" \
"Host: xyssl.org\r\n\r\n"
```

Definition at line 38 of file ssl\_client1.c.

Referenced by [main\(\)](#).

##### 5.149.1.3 #define [SERVER\\_NAME](#) "xyssl.org"

Definition at line 37 of file ssl\_client1.c.

##### 5.149.1.4 #define [SERVER\\_PORT](#) 443

Definition at line 32 of file ssl\_client1.c.

## 5.149.2 Function Documentation

### 5.149.2.1 `int main (void)`

Definition at line 53 of file `ssl_client1.c`.

References `buf`, `GET_REQUEST`, `havege_init()`, `havege_rand()`, `my_debug()`, `net_close()`, `net_connect()`, `net_recv()`, `net_send()`, `server_fd`, `SERVER_NAME`, `SERVER_PORT`, `ssl_close_notify()`, `ssl_default_ciphers`, `ssl_free()`, `ssl_init()`, `SSL_IS_CLIENT`, `ssl_read()`, `ssl_set_authmode()`, `ssl_set_bio()`, `ssl_set_ciphers()`, `ssl_set_dbg()`, `ssl_set_endpoint()`, `ssl_set_rng()`, `ssl_set_session()`, `SSL_VERIFY_NONE`, `ssl_write()`, `XYSSL_ERR_NET_TRY_AGAIN`, and `XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY`.

### 5.149.2.2 `void my_debug (void *ctx, int level, char *str)`

Definition at line 44 of file `ssl_client1.c`.

References `DEBUG_LEVEL`.

Referenced by `main()`, and `ssl_test()`.



## 5.150 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl\_client2.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/net.h"
#include "xyssl/ssl.h"
#include "xyssl/havege.h"
#include "xyssl/certs.h"
#include "xyssl/x509.h"
```

### Defines

- `#define SERVER_PORT 443`
- `#define SERVER_NAME "xyssl.org"`
- `#define GET_REQUEST`
- `#define DEBUG_LEVEL 0`

### Functions

- `void my_debug (void *ctx, int level, char *str)`
- `int main (void)`

#### 5.150.1 Define Documentation

##### 5.150.1.1 `#define DEBUG_LEVEL 0`

Definition at line 44 of file `ssl_client2.c`.

##### 5.150.1.2 `#define GET_REQUEST`

###### Value:

```
"GET /hello/ HTTP/1.1\r\n" \
"Host: xyssl.org\r\n\r\n"
```

Definition at line 40 of file `ssl_client2.c`.

##### 5.150.1.3 `#define SERVER_NAME "xyssl.org"`

Definition at line 39 of file `ssl_client2.c`.

##### 5.150.1.4 `#define SERVER_PORT 443`

Definition at line 34 of file `ssl_client2.c`.

## 5.150.2 Function Documentation

### 5.150.2.1 `int main (void)`

Definition at line 55 of file `ssl_client2.c`.

References `BADCERT_CN_MISMATCH`, `BADCERT_EXPIRED`, `BADCERT_NOT_TRUSTED`, `BADCERT_REVOKED`, `buf`, `GET_REQUEST`, `havege_init()`, `havege_rand()`, `net_close()`, `net_connect()`, `net_recv()`, `net_send()`, `_ssl_context::peer_cert`, `rsa_free()`, `server_fd`, `SERVER_NAME`, `SERVER_PORT`, `ssl_close_notify()`, `ssl_default_ciphers`, `ssl_free()`, `ssl_get_cipher()`, `ssl_get_verify_result()`, `ssl_handshake()`, `ssl_init()`, `SSL_IS_CLIENT`, `ssl_read()`, `ssl_set_authmode()`, `ssl_set_bio()`, `ssl_set_ca_chain()`, `ssl_set_ciphers()`, `ssl_set_endpoint()`, `ssl_set_hostname()`, `ssl_set_own_cert()`, `ssl_set_rng()`, `ssl_set_session()`, `SSL_VERIFY_OPTIONAL`, `ssl_write()`, `test_cli_cert`, `test_cli_key`, `x509_free()`, `x509parse_cert_info()`, `x509parse_cert()`, `x509parse_key()`, `xyssl_ca_cert`, `XYSSL_ERR_NET_TRY_AGAIN`, and `XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY`.

### 5.150.2.2 `void my_debug (void *ctx, int level, char *str)`

Definition at line 46 of file `ssl_client2.c`.

References `DEBUG_LEVEL`.

## 5.151 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl\_server.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include "xyssl/havege.h"
#include "xyssl/certs.h"
#include "xyssl/x509.h"
#include "xyssl/ssl.h"
#include "xyssl/net.h"
```

### Defines

- #define [HTTP\\_RESPONSE](#)
- #define [DEBUG\\_LEVEL](#) 0

### Functions

- void [my\\_debug](#) (void \*ctx, [int](#) level, char \*str)
- static [int](#) [my\\_get\\_session](#) ([ssl\\_context](#) \*ssl)
- static [int](#) [my\\_set\\_session](#) ([ssl\\_context](#) \*ssl)
- [int](#) [main](#) (void)

### Variables

- char \* [my\\_dhm\\_P](#)
- char \* [my\\_dhm\\_G](#) = "4"
- [int](#) [my\\_ciphers](#) []
- [ssl\\_session](#) \* [s\\_list\\_1st](#) = NULL
- [ssl\\_session](#) \* [cur](#)
- [ssl\\_session](#) \* [prv](#)

### 5.151.1 Define Documentation

#### 5.151.1.1 #define [DEBUG\\_LEVEL](#) 0

Definition at line 75 of file `ssl_server.c`.

#### 5.151.1.2 #define [HTTP\\_RESPONSE](#)

##### Value:

```
"HTTP/1.0 200 OK\r\nContent-Type: text/html\r\n\r\n" \
"<h2><p><center>Successful connection using: %s\r\n"
```

Definition at line 39 of file ssl\_server.c.

Referenced by http\_ParseRequest(), main(), and mtp\_http\_Parse().

## 5.151.2 Function Documentation

### 5.151.2.1 int main (void)

Definition at line 160 of file ssl\_server.c.

References buf, client\_fd, havege\_init(), havege\_rand(), HTTP\_RESPONSE, my\_ciphers, my\_debug(), my\_dhm\_G, my\_dhm\_P, my\_get\_session(), my\_set\_session(), net\_accept(), net\_bind(), net\_close(), net\_recv(), net\_send(), \_ssl\_session::next, x509\_cert::next, rsa\_free(), ssl\_close\_notify(), ssl\_free(), ssl\_get\_cipher(), ssl\_handshake(), ssl\_init(), SSL\_IS\_SERVER, ssl\_read(), ssl\_set\_authmode(), ssl\_set\_bio(), ssl\_set\_ca\_chain(), ssl\_set\_ciphers(), ssl\_set\_dbg(), ssl\_set\_dh\_param(), ssl\_set\_endpoint(), ssl\_set\_own\_cert(), ssl\_set\_rng(), ssl\_set\_scb(), ssl\_set\_session(), SSL\_VERIFY\_NONE, ssl\_write(), test\_ca\_cert, test\_srv\_cert, test\_srv\_key, x509\_free(), x509parse\_cert(), x509parse\_key(), XYSSL\_ERR\_NET\_CONN\_RESET, XYSSL\_ERR\_NET\_TRY\_AGAIN, and XYSSL\_ERR\_SSL\_PEER\_CLOSE\_NOTIFY.

### 5.151.2.2 void my\_debug (void \* ctx, int level, char \* str)

Definition at line 77 of file ssl\_server.c.

References DEBUG\_LEVEL.

### 5.151.2.3 static int my\_get\_session (ssl\_context \* ssl) [static]

Definition at line 93 of file ssl\_server.c.

References \_ssl\_session::cipher, \_ssl\_session::id, \_ssl\_session::length, \_ssl\_session::master, \_ssl\_session::next, \_ssl\_context::resume, \_ssl\_context::session, \_ssl\_session::start, and \_ssl\_context::timeout.

Referenced by main().

### 5.151.2.4 static int my\_set\_session (ssl\_context \* ssl) [static]

Definition at line 125 of file ssl\_server.c.

References \_ssl\_session::id, \_ssl\_session::length, \_ssl\_session::next, \_ssl\_context::session, \_ssl\_session::start, and \_ssl\_context::timeout.

Referenced by main().

## 5.151.3 Variable Documentation

### 5.151.3.1 ssl\_session\* cur

Definition at line 91 of file ssl\_server.c.

Referenced by http\_GetToken(), http\_ParseRequest(), and x509parse\_verify().

### 5.151.3.2 int my\_ciphers[ ]

Initial value:

```
{
    SSL_EDH_RSA_AES_256_SHA,
    SSL_EDH_RSA_DES_168_SHA,
    SSL_RSA_AES_256_SHA,
    SSL_RSA_AES_128_SHA,
    SSL_RSA_DES_168_SHA,
    SSL_RSA_RC4_128_SHA,
    SSL_RSA_RC4_128_MD5,
    0
}
```

Definition at line 63 of file ssl\_server.c.

Referenced by main().

### **5.151.3.3 char\* my\_dhm\_G = "4"**

Definition at line 58 of file ssl\_server.c.

Referenced by main().

### **5.151.3.4 char\* my\_dhm\_P**

**Initial value:**

```
"E4004C1F94182000103D883A448B3F80"
"2CE4B44A83301270002C20D0321CFD00"
"11CCEF784C26A400F43DFB901BCA7538"
"F2C6B176001CF5A0FD16D2C48B1D0C1C"
"F6AC8E1DA6BCC3B4E1F96B0564965300"
"FFA1D0B601EB2800F489AA512C4B248C"
"01F76949A60BB7F00A40B1EAB64BDD48"
"E8A700D60B7F1200FA8E77B0A979DABF"
```

Definition at line 48 of file ssl\_server.c.

Referenced by main().

### **5.151.3.5 ssl\_session \* prv**

Definition at line 91 of file ssl\_server.c.

Referenced by aes\_self\_test(), and des\_self\_test().

### **5.151.3.6 ssl\_session\* s\_list\_1st = NULL**

Definition at line 90 of file ssl\_server.c.

## 5.152 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/benchmark.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include "xyssl/config.h"
#include "xyssl/md4.h"
#include "xyssl/md5.h"
#include "xyssl/sha1.h"
#include "xyssl/sha2.h"
#include "xyssl/arc4.h"
#include "xyssl/des.h"
#include "xyssl/aes.h"
#include "xyssl/rsa.h"
#include "xyssl/timing.h"
```

### Defines

- #define [BUFSIZE](#) 1024

### Functions

- static [int myrand](#) (void \*rng\_state)
- [int main](#) (void)

### Variables

- unsigned char [buf](#) [BUFSIZE]

### 5.152.1 Define Documentation

#### 5.152.1.1 #define BUFSIZE 1024

Definition at line 41 of file benchmark.c.

Referenced by [main\(\)](#).

### 5.152.2 Function Documentation

#### 5.152.2.1 int main (void)

Definition at line 53 of file benchmark.c.

References `aes_crypt_cbc()`, `AES_ENCRYPT`, `aes_setkey_enc()`, `alarmed`, `arc4_crypt()`, `arc4_setup()`, `buf`, `BUFSIZE`, `des3_crypt_cbc()`, `des3_set3key_enc()`, `des_crypt_cbc()`, `DES_ENCRYPT`, `des_setkey_enc()`, `hardclock()`, `md4()`, `md5()`, `myrand()`, `rsa_free()`, `rsa_gen_key()`, `rsa_init()`, `RSA_PKCS_V15`, `rsa_private()`, `rsa_public()`, `set_alarm()`, `sha1()`, and `sha2()`.

#### **5.152.2.2 static int myrand (void \* rng\_state) [static]**

Definition at line 43 of file `benchmark.c`.

Referenced by `main()`.

### **5.152.3 Variable Documentation**

#### **5.152.3.1 unsigned char buf[BUFSIZE]**

Definition at line 51 of file `benchmark.c`.

Referenced by `aes_self_test()`, `agent_xml_compose__create_row_nodes()`, `agent_xml_compose__data()`, `agent_xml_compose__task()`, `agent_xml_compose__tasks()`, `agent_xml_parse__fill_row_data()`, `agent_xml_parse__tasks()`, `arc4_self_test()`, `cmd_prompt_Thread()`, `des_self_test()`, `fipa_DateTime_Compose()`, `fipa_datetime_Parse()`, `fipa_envelope_Compose__date()`, `fipa_envelope_Compose__from()`, `fipa_envelope_HandleParams()`, `main()`, `MC_End()`, `MC_Initialize()`, `MC_LoadAgentFromFile()`, `MC_RetrieveAgentCode()`, `MC_SendAgentFile()`, `MC_SendAgentMigrationMessageFile()`, `md5_check()`, `md5_file()`, `md5_self_test()`, `message_InitializeFromAgent()`, `message_send_Thread()`, `message_xml_parse__message()`, `mtp_http_ComposeMessage()`, `mtp_http_CreateMessage()`, `mxml_fd_getc()`, `mxml_fd_putc()`, `mxmlLoadFd()`, `mxmlSaveFd()`, `net_htons()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_verify()`, `sha1_check()`, `sha1_file()`, `sha1_self_test()`, `sha2_check()`, `sha2_file()`, `sha2_self_test()`, `sha4_file()`, `sha4_self_test()`, `ssl_flush_output()`, `ssl_parse_client_hello()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_write_certificate_request()`, `ssl_write_client_hello()`, `ssl_write_server_hello()`, `udplisten_Thread()`, `x509parse_cert_info()`, `x509parse_crtfile()`, `x509parse_keyfile()`, and `xml_get_cdata()`.

## 5.153 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/selftest.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/config.h"
#include "xyssl/md2.h"
#include "xyssl/md4.h"
#include "xyssl/md5.h"
#include "xyssl/sha1.h"
#include "xyssl/sha2.h"
#include "xyssl/sha4.h"
#include "xyssl/arc4.h"
#include "xyssl/des.h"
#include "xyssl/aes.h"
#include "xyssl/base64.h"
#include "xyssl/bignum.h"
#include "xyssl/rsa.h"
#include "xyssl/x509.h"
```

### Functions

- `int main (int argc, char *argv[ ])`

#### 5.153.1 Function Documentation

##### 5.153.1.1 `int main (int argc, char * argv[ ])`

Definition at line 44 of file selftest.c.

References `aes_self_test()`, `arc4_self_test()`, `base64_self_test()`, `des_self_test()`, `md2_self_test()`, `md4_self_test()`, `md5_self_test()`, `mpi_self_test()`, `rsa_self_test()`, `sha1_self_test()`, `sha2_self_test()`, `sha4_self_test()`, and `x509_self_test()`.



## 5.154 /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/ssl\_test.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include "xyssl/net.h"
#include "xyssl/ssl.h"
#include "xyssl/havege.h"
#include "xyssl/timing.h"
#include "xyssl/certs.h"
```

### Data Structures

- struct [options](#)

### Defines

- #define [OPMODE\\_NONE](#) 0
- #define [OPMODE\\_CLIENT](#) 1
- #define [OPMODE\\_SERVER](#) 2
- #define [IOMODE\\_BLOCK](#) 0
- #define [IOMODE\\_NONBLOCK](#) 1
- #define [COMMAND\\_READ](#) 1
- #define [COMMAND\\_WRITE](#) 2
- #define [COMMAND\\_BOTH](#) 3
- #define [DFL\\_OPMODE](#) OPMODE\_NONE
- #define [DFL\\_IOMODE](#) IOMODE\_BLOCK
- #define [DFL\\_SERVER\\_NAME](#) "localhost"
- #define [DFL\\_SERVER\\_PORT](#) 4433
- #define [DFL\\_COMMAND](#) COMMAND\_READ
- #define [DFL\\_BUFFER\\_SIZE](#) 1024
- #define [DFL\\_MAX\\_BYTES](#) 0
- #define [DFL\\_DEBUG\\_LEVEL](#) 0
- #define [DFL\\_CONN\\_TIMEOUT](#) 0
- #define [DFL\\_MAX\\_CONNECTIONS](#) 0
- #define [DFL\\_SESSION\\_REUSE](#) 1
- #define [DFL\\_SESSION\\_LIFETIME](#) 86400
- #define [DFL\\_FORCE\\_CIPHER](#) 0
- #define [USAGE](#)

### Functions

- unsigned long [lcppm5](#) (unsigned long [int](#) \*state)
- void [my\\_debug](#) (void \*ctx, [int](#) level, char \*str)
- static [int](#) [ssl\\_test](#) (struct [options](#) \*opt)
- [int](#) [main](#) ([int](#) argc, char \*argv[ ])

## Variables

- char \* [dhm\\_G](#) = "4"
- char \* [dhm\\_P](#)
- int [server\\_fd](#) = -1

### 5.154.1 Define Documentation

#### 5.154.1.1 #define COMMAND\_BOTH 3

Definition at line 44 of file `ssl_test.c`.

Referenced by `main()`.

#### 5.154.1.2 #define COMMAND\_READ 1

Definition at line 42 of file `ssl_test.c`.

Referenced by `main()`, and `ssl_test()`.

#### 5.154.1.3 #define COMMAND\_WRITE 2

Definition at line 43 of file `ssl_test.c`.

Referenced by `main()`, and `ssl_test()`.

#### 5.154.1.4 #define DFL\_BUFFER\_SIZE 1024

Definition at line 51 of file `ssl_test.c`.

Referenced by `main()`.

#### 5.154.1.5 #define DFL\_COMMAND COMMAND\_READ

Definition at line 50 of file `ssl_test.c`.

Referenced by `main()`.

#### 5.154.1.6 #define DFL\_CONN\_TIMEOUT 0

Definition at line 54 of file `ssl_test.c`.

Referenced by `main()`.

#### 5.154.1.7 #define DFL\_DEBUG\_LEVEL 0

Definition at line 53 of file `ssl_test.c`.

Referenced by `main()`.

**5.154.1.8 #define DFL\_FORCE\_CIPHER 0**

Definition at line 58 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

**5.154.1.9 #define DFL\_IOMODE IOMODE\_BLOCK**

Definition at line 47 of file ssl\_test.c.

Referenced by main().

**5.154.1.10 #define DFL\_MAX\_BYTES 0**

Definition at line 52 of file ssl\_test.c.

Referenced by main().

**5.154.1.11 #define DFL\_MAX\_CONNECTIONS 0**

Definition at line 55 of file ssl\_test.c.

Referenced by main().

**5.154.1.12 #define DFL\_OPMODE OPMODE\_NONE**

Definition at line 46 of file ssl\_test.c.

Referenced by main().

**5.154.1.13 #define DFL\_SERVER\_NAME "localhost"**

Definition at line 48 of file ssl\_test.c.

Referenced by main().

**5.154.1.14 #define DFL\_SERVER\_PORT 4433**

Definition at line 49 of file ssl\_test.c.

Referenced by main().

**5.154.1.15 #define DFL\_SESSION\_LIFETIME 86400**

Definition at line 57 of file ssl\_test.c.

Referenced by main().

**5.154.1.16 #define DFL\_SESSION\_REUSE 1**

Definition at line 56 of file ssl\_test.c.

Referenced by main().

**5.154.1.17 #define IOMODE\_BLOCK 0**

Definition at line 39 of file ssl\_test.c.

Referenced by main().

**5.154.1.18 #define IOMODE\_NONBLOCK 1**

Definition at line 40 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

**5.154.1.19 #define OPMODE\_CLIENT 1**

Definition at line 36 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

**5.154.1.20 #define OPMODE\_NONE 0**

Definition at line 35 of file ssl\_test.c.

**5.154.1.21 #define OPMODE\_SERVER 2**

Definition at line 37 of file ssl\_test.c.

Referenced by main(), and ssl\_test().

**5.154.1.22 #define USAGE****Value:**

```
"\n usage: ssl_test opmode=<> command=<>...\n"
"\n acceptable parameters:\n"
"  opmode=client/server      default: <none>\n"
"  iomode=block/nonblock    default: block\n"
"  server_name=%s           default: localhost\n"
"  server_port=%d           default: 4433\n"
"  command=read/write/both  default: read\n"
"  buffer_size=%d (bytes)   default: 1024\n"
"  max_bytes=%d (bytes)     default: 0 (no limit)\n"
"  debug_level=%d           default: 0 (disabled)\n"
"  conn_timeout=%d (ms)     default: 0 (no timeout)\n"
"  max_connections=%d       default: 0 (no limit)\n"
"  session_reuse=on/off     default: on (enabled)\n"
"  session_lifetime=%d (s)  default: 86400\n"
"  force_cipher=<name>      default: all enabled\n"
" acceptable cipher names:\n"
"  SSL_RSA_RC4_128_MD5      SSL_RSA_RC4_128_SHA\n"
"  SSL_RSA_DES_168_SHA      SSL_EDH_RSA_DES_168_SHA\n"
"  SSL_RSA_AES_128_SHA      SSL_EDH_RSA_AES_256_SHA\n"
"  SSL_RSA_AES_256_SHA\n"
```

Definition at line 372 of file ssl\_test.c.

## 5.154.2 Function Documentation

### 5.154.2.1 unsigned long int lcppm5 (unsigned long int \* *state*)

Definition at line 96 of file ssl\_test.c.

Referenced by ssl\_test().

### 5.154.2.2 int main (int *argc*, char \* *argv*[])

Definition at line 394 of file ssl\_test.c.

References options::buffer\_size, options::command, COMMAND\_BOTH, COMMAND\_READ, COMMAND\_WRITE, options::conn\_timeout, options::debug\_level, DFL\_BUFFER\_SIZE, DFL\_COMMAND, DFL\_CONN\_TIMEOUT, DFL\_DEBUG\_LEVEL, DFL\_FORCE\_CIPHER, DFL\_IOMODE, DFL\_MAX\_BYTES, DFL\_MAX\_CONNECTIONS, DFL\_OPMODE, DFL\_SERVER\_NAME, DFL\_SERVER\_PORT, DFL\_SESSION\_LIFETIME, DFL\_SESSION\_REUSE, options::force\_cipher, options::iomode, IOMODE\_BLOCK, IOMODE\_NONBLOCK, options::max\_bytes, options::max\_connections, options::opmode, OPMODE\_CLIENT, OPMODE\_SERVER, options::server\_name, options::server\_port, options::session\_lifetime, options::session\_reuse, SSL\_EDH\_RSA\_AES\_256\_SHA, SSL\_EDH\_RSA\_DES\_168\_SHA, SSL\_RSA\_AES\_128\_SHA, SSL\_RSA\_AES\_256\_SHA, SSL\_RSA\_DES\_168\_SHA, SSL\_RSA\_RC4\_128\_MD5, SSL\_RSA\_RC4\_128\_SHA, ssl\_test(), and USAGE.

### 5.154.2.3 void my\_debug (void \* *ctx*, int *level*, char \* *str*)

Definition at line 113 of file ssl\_test.c.

### 5.154.2.4 static int ssl\_test (struct options \* *opt*) [static]

Definition at line 122 of file ssl\_test.c.

References options::buffer\_size, client\_fd, options::command, COMMAND\_READ, COMMAND\_WRITE, options::conn\_timeout, DFL\_FORCE\_CIPHER, dhm\_G, dhm\_P, options::force\_cipher, get\_timer(), havege\_init(), havege\_rand(), options::iomode, IOMODE\_NONBLOCK, lcppm5(), options::max\_bytes, my\_debug(), net\_accept(), net\_bind(), net\_close(), net\_connect(), net\_recv(), net\_send(), net\_set\_nonblock(), x509\_cert::next, options::opmode, OPMODE\_CLIENT, OPMODE\_SERVER, rsa\_free(), server\_fd, options::server\_name, options::server\_port, options::session\_lifetime, options::session\_reuse, ssl\_close\_notify(), ssl\_default\_ciphers, ssl\_free(), ssl\_init(), SSL\_IS\_CLIENT, SSL\_IS\_SERVER, ssl\_read(), ssl\_set\_authmode(), ssl\_set\_bio(), ssl\_set\_ca\_chain(), ssl\_set\_ciphers(), ssl\_set\_dbg(), ssl\_set\_dh\_param(), ssl\_set\_endpoint(), ssl\_set\_own\_cert(), ssl\_set\_rng(), ssl\_set\_session(), SSL\_VERIFY\_NONE, ssl\_write(), test\_ca crt, test\_srv crt, test\_srv\_key, x509\_free(), x509parse\_crt(), x509parse\_key(), XYSSL\_ERR\_NET\_CONN\_RESET, XYSSL\_ERR\_NET\_TRY\_AGAIN, and XYSSL\_ERR\_SSL\_PEER\_CLOSE\_NOTIFY.

Referenced by main().

## 5.154.3 Variable Documentation

### 5.154.3.1 char\* dhm\_G = "4"

Definition at line 63 of file ssl\_test.c.

Referenced by ssl\_test().

### 5.154.3.2 char\* dhm\_P

**Initial value:**

```
"E4004C1F94182000103D883A448B3F802CE4B44A83301270002C20D0321CFD00"  
"11CCEF784C26A400F43DFB901BCA7538F2C6B176001CF5A0FD16D2C48B1D0C1C"  
"F6AC8E1DA6BCC3B4E1F96B0564965300FFA1D0B601EB2800F489AA512C4B248C"  
"01F76949A60BB7F00A40B1EAB64BDD48E8A700D60B7F1200FA8E77B0A979DABF"
```

Definition at line 64 of file ssl\_test.c.

Referenced by ssl\_test().

### 5.154.3.3 int server\_fd = -1

Definition at line 70 of file ssl\_test.c.

## 5.155 /home/dko/Projects/mobilec/src/util/mc\_genkey.c File Reference

```
#include "../security/interface.h"
```

### Functions

- void [print\\_usage](#) ()
- int [main](#) (int argc, char \*argv[])

#### 5.155.1 Function Documentation

##### 5.155.1.1 int main (int *argc*, char \* *argv*[])

Definition at line 19 of file mc\_genkey.c.

References [generate\\_RSA\\_keys\\_ciphertext\(\)](#), [generate\\_RSA\\_keys\\_plaintext\(\)](#), and [print\\_usage\(\)](#).

##### 5.155.1.2 void [print\\_usage](#) ()

Definition at line 11 of file mc\_genkey.c.

Referenced by [main\(\)](#).

## 5.156 /home/dko/Projects/mobilec/src/xml\_compose.c File Reference

```
#include "config.h"
#include <mxml.h>
#include "include/agent.h"
#include "include/xml_compose.h"
#include "include/xml_helper.h"
```

### Functions

- [mxml\\_node\\_t \\* agent\\_xml\\_compose \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_gaf\\_message \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_message \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_mobile\\_agent \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_agent\\_data \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_name \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_owner \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_home \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_wg\\_code \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_tasks \(agent\\_p agent\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_task \(agent\\_p agent, int index\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_data \(agent\\_p agent, int index, interpreter\\_variable\\_data\\_t \\*interp\\_variable\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_agent\\_code \(agent\\_p agent, int index\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_row \(interpreter\\_variable\\_data\\_t \\*interp\\_variable, int index\)](#)
- [mxml\\_node\\_t \\* agent\\_xml\\_compose\\_\\_create\\_row\\_nodes \(void \\*data, int index, int \\*extent, ChType\\_t type, int dim, int extent\\_index\)](#)

### 5.156.1 Function Documentation

#### 5.156.1.1 [mxml\\_node\\_t\\* agent\\_xml\\_compose \(agent\\_p agent\)](#)

Definition at line 46 of file `xml_compose.c`.

References `agent_xml_compose__gaf_message()`, `MXML_ADD_AFTER`, `MXML_ADD_TO_PARENT`, `MXML_NO_CALLBACK`, `mxmlAdd()`, `mxmlLoadString()`, and `node`.

Referenced by `message_InitializeFromAgent()`.

#### 5.156.1.2 [mxml\\_node\\_t\\* agent\\_xml\\_compose\\_\\_agent\\_code \(agent\\_p agent, int index\)](#)

Definition at line 521 of file `xml_compose.c`.

References `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_s::datastate`, `MXML_NO_PARENT`, `mxmlElementSetAttr()`, `mxmlNewElement()`, `node`, and `xml_new_cdata()`.

Referenced by `agent_xml_compose__tasks()`.



**5.156.1.3 mxml\_node\_t\* agent\_xml\_compose\_\_agent\_data (agent\_p agent)**

Definition at line 150 of file xml\_compose.c.

References agent\_xml\_compose\_\_home(), agent\_xml\_compose\_\_name(), agent\_xml\_compose\_\_owner(), agent\_xml\_compose\_\_tasks(), agent\_xml\_compose\_\_wg\_code(), MXML\_ADD\_AFTER, mxmlAdd(), mxmlNewElement(), and node.

Referenced by agent\_xml\_compose\_\_mobile\_agent().

**5.156.1.4 mxml\_node\_t\* agent\_xml\_compose\_\_create\_row\_nodes (void \* data, int index, int \* extent, CType\_t type, int dim, int extent\_index)**

Definition at line 572 of file xml\_compose.c.

References agent\_xml\_compose\_\_create\_row\_nodes(), buf, CH\_DATATYPE\_SIZE, CH\_DATATYPE\_VALUE\_STRING, MXML\_ADD\_AFTER, MXML\_ADD\_TO\_PARENT, MXML\_NO\_PARENT, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), mxmlNewText(), node, and size.

Referenced by agent\_xml\_compose\_\_create\_row\_nodes(), and agent\_xml\_compose\_\_row().

**5.156.1.5 mxml\_node\_t\* agent\_xml\_compose\_\_data (agent\_p agent, int index, interpreter\_variable\_data\_t \* interp\_variable)**

Definition at line 447 of file xml\_compose.c.

References agent\_xml\_compose\_\_row(), interpreter\_variable\_data\_s::array\_dim, buf, CH\_DATATYPE\_STRING, CH\_DATATYPE\_VALUE\_STRING, interpreter\_variable\_data\_s::data, interpreter\_variable\_data\_s::data\_type, MXML\_ADD\_AFTER, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), interpreter\_variable\_data\_s::name, and node.

Referenced by agent\_xml\_compose\_\_task().

**5.156.1.6 mxml\_node\_t\* agent\_xml\_compose\_\_gaf\_message (agent\_p agent)**

Definition at line 66 of file xml\_compose.c.

References agent\_xml\_compose\_\_message(), MXML\_ADD\_AFTER, mxmlAdd(), mxmlNewElement(), and node.

Referenced by agent\_xml\_compose().

**5.156.1.7 mxml\_node\_t\* agent\_xml\_compose\_\_home (agent\_p agent)**

Definition at line 253 of file xml\_compose.c.

References agent\_s::home, mxmlNewElement(), mxmlNewText(), and node.

Referenced by agent\_xml\_compose\_\_agent\_data().

**5.156.1.8 mxml\_node\_t\* agent\_xml\_compose\_\_message (agent\_p agent)**

Definition at line 85 of file xml\_compose.c.

References `agent_s::agent_type`, `agent_xml_compose__mobile_agent()`, `MC_LOCAL_AGENT`, `MC_REMOTE_AGENT`, `MC_RETURN_AGENT`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlElementSetAttr()`, `mxmlNewElement()`, and `node`.

Referenced by `agent_xml_compose__gaf_message()`.

#### **5.156.1.9 `mxml_node_t* agent_xml_compose__mobile_agent (agent_p agent)`**

Definition at line 129 of file `xml_compose.c`.

References `agent_xml_compose__agent_data()`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlNewElement()`, and `node`.

Referenced by `agent_xml_compose__message()`.

#### **5.156.1.10 `mxml_node_t* agent_xml_compose__name (agent_p agent)`**

Definition at line 221 of file `xml_compose.c`.

References `mxmlNewElement()`, `mxmlNewText()`, `agent_s::name`, and `node`.

Referenced by `agent_xml_compose__agent_data()`.

#### **5.156.1.11 `mxml_node_t* agent_xml_compose__owner (agent_p agent)`**

Definition at line 237 of file `xml_compose.c`.

References `mxmlNewElement()`, `mxmlNewText()`, `node`, and `agent_s::owner`.

Referenced by `agent_xml_compose__agent_data()`.

#### **5.156.1.12 `mxml_node_t* agent_xml_compose__row (interpreter_variable_data_t * interp_variable, int index)`**

Definition at line 549 of file `xml_compose.c`.

References `agent_xml_compose__create_row_nodes()`, `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, and `node`.

Referenced by `agent_xml_compose__data()`.

#### **5.156.1.13 `mxml_node_t* agent_xml_compose__task (agent_p agent, int index)`**

Definition at line 342 of file `xml_compose.c`.

References `agent_task_s::agent_return_data`, `agent_task_s::agent_variable_list`, `agent_xml_compose__data()`, `buf`, `agent_task_s::code_id`, `agent_s::datastate`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlElementSetAttr()`, `mxmlNewElement()`, `node`, `agent_task_s::persistent`, `agent_datastate_s::persistent`, `agent_task_s::server_name`, `agent_datastate_s::tasks`, and `agent_task_s::var_name`.

Referenced by `agent_xml_compose__tasks()`.

**5.156.1.14 mxml\_node\_t\* agent\_xml\_compose\_\_tasks (agent\_p *agent*)**

Definition at line 285 of file xml\_compose.c.

References agent\_xml\_compose\_\_agent\_code(), agent\_xml\_compose\_\_task(), buf, agent\_s::datastate, MXML\_ADD\_AFTER, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), node, agent\_datastate\_s::number\_of\_tasks, and agent\_datastate\_s::task\_progress.

Referenced by agent\_xml\_compose\_\_agent\_data().

**5.156.1.15 mxml\_node\_t\* agent\_xml\_compose\_\_wg\_code (agent\_p *agent*)**

Definition at line 269 of file xml\_compose.c.

References mxmlNewElement(), mxmlNewText(), node, and agent\_s::wg\_code.

Referenced by agent\_xml\_compose\_\_agent\_data().

## 5.157 /home/dko/Projects/mobilec/src/xml\_helper.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <mxml.h>
#include "include/xml_helper.h"
#include "include/macros.h"
```

### Functions

- [mxml\\_node\\_t \\* xml\\_find\\_sibling](#) (const [mxml\\_node\\_t \\*node](#), const char \*sibling\_name)
- char \* [xml\\_get\\_cdata](#) (const [mxml\\_node\\_t \\*node](#))
- [mxml\\_node\\_t \\* xml\\_get\\_child](#) (const [mxml\\_node\\_t \\*node](#), const char \*child\_name, [int](#) descend)
- [mxml\\_node\\_t \\* xml\\_get\\_deep\\_child](#) (const [mxml\\_node\\_t \\*parent](#), const char \*\*child\_path)
- [mxml\\_node\\_t \\* xml\\_get\\_next\\_element](#) (const [mxml\\_node\\_t \\*node](#))
- char \* [xml\\_get\\_text](#) (const [mxml\\_node\\_t \\*node](#))
- char \* [xml\\_get\\_element\\_name](#) (const [mxml\\_node\\_t \\*node](#))
- [mxml\\_node\\_t \\* xml\\_new\\_cdata](#) ([mxml\\_node\\_t \\*parent](#), const char \*text)
- const char \* [whitespace\\_cb](#) ([mxml\\_node\\_t \\*node](#), [int](#) where)

### 5.157.1 Function Documentation

#### 5.157.1.1 const char\* whitespace\_cb (mxml\_node\_t \* node, int where)

Definition at line 254 of file xml\_helper.c.

References `MXML_WS_AFTER_CLOSE`, and `MXML_WS_BEFORE_OPEN`.

Referenced by `main()`.

#### 5.157.1.2 mxml\_node\_t\* xml\_find\_sibling (const mxml\_node\_t \* node, const char \* sibling\_name)

Definition at line 54 of file xml\_helper.c.

References `MXML_NO_DESCEND`, `mxmlFindElement()`, `mxml_node_s::parent`, and `xml_get_element_name()`.

#### 5.157.1.3 char\* xml\_get\_cdata (const mxml\_node\_t \* node)

Definition at line 76 of file xml\_helper.c.

References `buf`, `CHECK_NULL`, `MXML_ELEMENT`, `mxml_node_s::type`, and `xml_get_element_name()`.

Referenced by `xml_get_text()`.

**5.157.1.4 mxml\_node\_t\* xml\_get\_child (const mxml\_node\_t \* *node*, const char \* *child\_name*, int *descend*)**

Definition at line 109 of file xml\_helper.c.

References mxmlFindElement().

Referenced by agent\_return\_xml\_parse(), agent\_xml\_parse\_\_agent\_data(), agent\_xml\_parse\_\_data(), agent\_xml\_parse\_\_mobile\_agent(), message\_xml\_parse(), message\_xml\_parse\_\_message(), and xml\_get\_deep\_child().

**5.157.1.5 mxml\_node\_t\* xml\_get\_deep\_child (const mxml\_node\_t \* *parent*, const char \*\* *child\_path*)**

Definition at line 128 of file xml\_helper.c.

References MXML\_NO\_DESCEND, node, and xml\_get\_child().

**5.157.1.6 char\* xml\_get\_element\_name (const mxml\_node\_t \* *node*)**

Definition at line 222 of file xml\_helper.c.

References mxml\_value\_u::element, MXML\_ELEMENT, mxml\_value\_s::name, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by agent\_xml\_parse\_\_data(), agent\_xml\_parse\_\_mobile\_agent(), agent\_xml\_parse\_\_row(), message\_xml\_parse(), xml\_find\_sibling(), and xml\_get\_cdata().

**5.157.1.7 mxml\_node\_t\* xml\_get\_next\_element (const mxml\_node\_t \* *node*)**

Definition at line 142 of file xml\_helper.c.

References MXML\_ELEMENT, mxml\_node\_s::next, and mxml\_node\_s::type.

**5.157.1.8 char\* xml\_get\_text (const mxml\_node\_t \* *node*)**

Definition at line 160 of file xml\_helper.c.

References CHECK\_NULL, mxml\_node\_s::child, mxml\_value\_u::element, MXML\_ELEMENT, MXML\_TEXT, mxml\_value\_s::name, mxml\_node\_s::next, mxml\_text\_s::string, mxml\_value\_u::text, mxml\_node\_s::type, mxml\_node\_s::value, and xml\_get\_cdata().

Referenced by agent\_xml\_parse\_\_agent\_code(), agent\_xml\_parse\_\_home(), agent\_xml\_parse\_\_name(), agent\_xml\_parse\_\_owner(), agent\_xml\_parse\_\_sender(), and agent\_xml\_parse\_\_wg\_code().

**5.157.1.9 mxml\_node\_t\* xml\_new\_cdata (mxml\_node\_t \* *parent*, const char \* *text*)**

Definition at line 235 of file xml\_helper.c.

References CHECK\_NULL, mxmlNewElement(), and node.

Referenced by agent\_xml\_compose\_\_agent\_code().

## 5.158 /home/dko/Projects/mobilec/src/xml\_parser.c File Reference

```
#include <mxml.h>
#include <string.h>
#include <stdlib.h>
#include "config.h"
#include "include/interpreter_variable_data.h"
#include "include/message.h"
#include "include/xml_parser.h"
#include "include/xml_helper.h"
```

### Defines

- `#define \_XOPEN\_SOURCE 600`

### Functions

- `error\_code\_t agent\_xml\_parse (agent\_p agent)`
- `error\_code\_t agent\_xml\_parse\_\_mobile\_agent (agent\_p agent, xml\_parser\_p xml_parser)`
- `error\_code\_t agent\_xml\_parse\_\_agent\_data (agent\_p agent, xml\_parser\_p xml_parser)`
- `error\_code\_t agent\_xml\_parse\_\_name (agent\_p agent, xml\_parser\_p xml_parser)`
- `error\_code\_t agent\_xml\_parse\_\_owner (agent\_p agent, xml\_parser\_p xml_parser)`
- `error\_code\_t agent\_xml\_parse\_\_home (agent\_p agent, xml\_parser\_p xml_parser)`
- `error\_code\_t agent\_xml\_parse\_\_sender (agent\_p agent, xml\_parser\_p xml_parser)`
- `error\_code\_t agent\_xml\_parse\_\_wg\_code (agent\_p agent, xml\_parser\_p xml_parser)`
- `error\_code\_t agent\_xml\_parse\_\_tasks (agent\_p agent, xml\_parser\_p xml_parser)`
- `error\_code\_t agent\_xml\_parse\_\_task (agent\_p agent, xml\_parser\_p xml_parser, int index)`
- `error\_code\_t agent\_xml\_parse\_\_data (agent\_p agent, xml\_parser\_p xml_parser, int index)`
- `error\_code\_t agent\_xml\_parse\_\_row (interpreter\_variable\_data\_t *interp_variable, xml\_parser\_p xml_parser, int index)`
- `void agent\_xml\_parse\_\_fill\_row\_data (void *data, ChType\_t type, int *extent, const mxml\_node\_t *node, int *index)`
- `error\_code\_t agent\_xml\_parse\_\_agent\_code (agent\_p agent, int index, xml\_parser\_p xml_parser)`
- `error\_code\_t agent\_return\_xml\_parse (agent\_p agent)`
- `error\_code\_t message\_xml\_parse (message\_p message)`
- `error\_code\_t message\_xml\_parse\_\_message (message\_p message, xml\_parser\_p xml_parser)`

### 5.158.1 Define Documentation

#### 5.158.1.1 `#define \_XOPEN\_SOURCE 600`

Definition at line 38 of file `xml_parser.c`.

## 5.158.2 Function Documentation

### 5.158.2.1 `error_code_t agent_return_xml_parse (agent_p agent)`

Definition at line 913 of file `xml_parser.c`.

References `agent_xml_parse__home()`, `agent_xml_parse__name()`, `agent_xml_parse__owner()`, `agent_xml_parse__tasks()`, `agent_s::datastate`, `MC_SUCCESS`, `xml_get_child()`, and `agent_datastate_s::xml_root`.

### 5.158.2.2 `error_code_t agent_xml_parse (agent_p agent)`

Definition at line 52 of file `xml_parser.c`.

References `agent_xml_parse__mobile_agent()`, `agent_s::datastate`, `MC_SUCCESS`, and `agent_datastate_s::xml_agent_root`.

Referenced by `agent_initialize()`.

### 5.158.2.3 `error_code_t agent_xml_parse__agent_code (agent_p agent, int index, xml_parser_p xml_parser)`

Definition at line 872 of file `xml_parser.c`.

References `agent_datastate_s::agent_code`, `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_task_s::code_id`, `agent_s::datastate`, `MC_SUCCESS`, `mxmlelementGetAttr()`, `agent_datastate_s::number_of_tasks`, `agent_datastate_s::task_progress`, `agent_datastate_s::tasks`, and `xml_get_text()`.

Referenced by `agent_xml_parse__tasks()`.

### 5.158.2.4 `error_code_t agent_xml_parse__agent_data (agent_p agent, xml_parser_p xml_parser)`

Definition at line 93 of file `xml_parser.c`.

References `agent_xml_parse__home()`, `agent_xml_parse__name()`, `agent_xml_parse__owner()`, `agent_xml_parse__sender()`, `agent_xml_parse__tasks()`, `agent_xml_parse__wg_code()`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `xml_get_child()`.

Referenced by `agent_xml_parse__mobile_agent()`.

### 5.158.2.5 `error_code_t agent_xml_parse__data (agent_p agent, xml_parser_p xml_parser, int index)`

Definition at line 538 of file `xml_parser.c`.

References `agent_task_s::agent_return_data`, `agent_task_s::agent_variable_list`, `agent_xml_parse__row()`, `interpreter_variable_data_s::array_dim`, `CH_DATATYPE_SIZE`, `CH_DATATYPE_STR_TO_VAL`, `CH_STRING_DATATYPE`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, `agent_s::datastate`, `interpreter_variable_data_New()`, `MC_ERR_PARSE`, `MC_SUCCESS`, `mxmlelementGetAttr()`, `interpreter_variable_data_s::name`, `mxmlelement_s::parent`, `agent_task_s::persistent`, `agent_datastate_s::tasks`, `xml_get_child()`, and `xml_get_element_name()`.

Referenced by `agent_xml_parse__task()`.

#### 5.158.2.6 **void agent\_xml\_parse\_\_fill\_row\_data (void \* *data*, ChType\_t *type*, int \* *extent*, const mxml\_node\_t \* *node*, int \* *index*)**

Definition at line 751 of file xml\_parser.c.

References agent\_xml\_parse\_\_fill\_row\_data(), buf, CH\_DATATYPE\_SIZE, mxml\_node\_s::child, MXML\_DESCEND\_FIRST, MXML\_ELEMENT, MXML\_TEXT, mxmlFindElement(), mxml\_text\_s::string, strtok\_r, mxml\_value\_u::text, mxml\_node\_s::type, and mxml\_node\_s::value.

Referenced by agent\_xml\_parse\_\_fill\_row\_data(), and agent\_xml\_parse\_\_row().

#### 5.158.2.7 **error\_code\_t agent\_xml\_parse\_\_home (agent\_p *agent*, xml\_parser\_p *xml\_parser*)**

Definition at line 219 of file xml\_parser.c.

References CHECK\_NULL, agent\_s::home, MC\_SUCCESS, and xml\_get\_text().

Referenced by agent\_return\_xml\_parse(), and agent\_xml\_parse\_\_agent\_data().

#### 5.158.2.8 **error\_code\_t agent\_xml\_parse\_\_mobile\_agent (agent\_p *agent*, xml\_parser\_p *xml\_parser*)**

Definition at line 65 of file xml\_parser.c.

References agent\_xml\_parse\_\_agent\_data(), MC\_ERR\_PARSE, xml\_get\_child(), and xml\_get\_element\_name().

Referenced by agent\_xml\_parse().

#### 5.158.2.9 **error\_code\_t agent\_xml\_parse\_\_name (agent\_p *agent*, xml\_parser\_p *xml\_parser*)**

Definition at line 166 of file xml\_parser.c.

References CHECK\_NULL, MC\_ERR\_PARSE, MC\_SUCCESS, agent\_s::name, and xml\_get\_text().

Referenced by agent\_return\_xml\_parse(), and agent\_xml\_parse\_\_agent\_data().

#### 5.158.2.10 **error\_code\_t agent\_xml\_parse\_\_owner (agent\_p *agent*, xml\_parser\_p *xml\_parser*)**

Definition at line 192 of file xml\_parser.c.

References CHECK\_NULL, MC\_SUCCESS, agent\_s::owner, and xml\_get\_text().

Referenced by agent\_return\_xml\_parse(), and agent\_xml\_parse\_\_agent\_data().

#### 5.158.2.11 **error\_code\_t agent\_xml\_parse\_\_row (interpreter\_variable\_data\_t \* *interp\_variable*, xml\_parser\_p *xml\_parser*, int *index*)**

Definition at line 684 of file xml\_parser.c.

References agent\_xml\_parse\_\_fill\_row\_data(), interpreter\_variable\_data\_s::array\_dim, interpreter\_variable\_data\_s::array\_extent, CH\_DATATYPE\_SIZE, interpreter\_variable\_data\_s::data, interpreter\_variable\_data\_s::data\_type, MC\_SUCCESS, and xml\_get\_element\_name().

Referenced by agent\_xml\_parse\_\_data().



**5.158.2.12 error\_code\_t agent\_xml\_parse\_\_sender (agent\_p agent, xml\_parser\_p xml\_parser)**

Definition at line 245 of file xml\_parser.c.

References CHECK\_NULL, MC\_SUCCESS, agent\_s::sender, and xml\_get\_text().

Referenced by agent\_xml\_parse\_\_agent\_data().

**5.158.2.13 error\_code\_t agent\_xml\_parse\_\_task (agent\_p agent, xml\_parser\_p xml\_parser, int index)**

Definition at line 442 of file xml\_parser.c.

References agent\_xml\_parse\_\_data(), CHECK\_NULL, agent\_task\_s::code\_id, agent\_s::datastate, MC\_ERR\_PARSE, MC\_SUCCESS, MXML\_DESCEND\_FIRST, MXML\_NO\_DESCEND, mxmlElementGetAttr(), mxmlFindElement(), agent\_task\_s::persistent, agent\_task\_s::server\_name, strdup, agent\_datastate\_s::tasks, and agent\_task\_s::var\_name.

Referenced by agent\_xml\_parse\_\_tasks().

**5.158.2.14 error\_code\_t agent\_xml\_parse\_\_tasks (agent\_p agent, xml\_parser\_p xml\_parser)**

Definition at line 304 of file xml\_parser.c.

References agent\_datastate\_s::agent\_code, agent\_datastate\_s::agent\_code\_ids, agent\_datastate\_s::agent\_codes, agent\_task\_New(), agent\_xml\_parse\_\_agent\_code(), agent\_xml\_parse\_\_task(), buf, agent\_s::datastate, MC\_ERR\_PARSE, MXML\_DESCEND, MXML\_DESCEND\_FIRST, MXML\_NO\_DESCEND, mxmlElementGetAttr(), mxmlFindElement(), agent\_datastate\_s::number\_of\_tasks, agent\_datastate\_s::task\_progress, and agent\_datastate\_s::tasks.

Referenced by agent\_return\_xml\_parse(), and agent\_xml\_parse\_\_agent\_data().

**5.158.2.15 error\_code\_t agent\_xml\_parse\_\_wg\_code (agent\_p agent, xml\_parser\_p xml\_parser)**

Definition at line 273 of file xml\_parser.c.

References MC\_SUCCESS, agent\_s::wg\_code, and xml\_get\_text().

Referenced by agent\_xml\_parse\_\_agent\_data().

**5.158.2.16 error\_code\_t message\_xml\_parse (message\_p message)**

Definition at line 948 of file xml\_parser.c.

References MC\_ERR\_PARSE, message\_xml\_parse\_\_message(), MXML\_DESCEND, MXML\_NO\_DESCEND, mxmlFindElement(), xml\_get\_child(), xml\_get\_element\_name(), and message\_s::xml\_root.

Referenced by acc\_connection\_Thread(), message\_InitializeFromConnection(), and message\_InitializeFromString().

**5.158.2.17 error\_code\_t message\_xml\_parse\_\_message (message\_p message, xml\_parser\_p xml\_parser)**

Definition at line 1003 of file xml\_parser.c.

References `message_s::addr`, `buf`, `CHECK_NULL`, `ENCRYPTED_DATA`, `ENCRYPTION_INITIALIZE`, `FIPA_ACL`, `message_s::from_address`, `MC_ERR_PARSE`, `MC_SUCCESS`, `message_s::message_type`, `MOBILE_AGENT`, `mxmlElementGetAttr()`, `port`, `REQUEST_ENCRYPTION_INITIALIZE`, `RETURN_MSG`, `strtok_r`, `xml_get_child()`, and `message_s::xml_payload`.

Referenced by `message_xml_parse()`.

# Index

~foo\_c  
  foo\_c, 97  
~foo\_s  
  foo\_s, 100  
/home/dko/Projects/mobilec/src/acc.c, 167  
/home/dko/Projects/mobilec/src/agent.c, 172  
/home/dko/Projects/mobilec/src/agent\_datastate.c, 176  
/home/dko/Projects/mobilec/src/agent\_mailbox.c, 177  
/home/dko/Projects/mobilec/src/agent\_return\_data.c, 179  
/home/dko/Projects/mobilec/src/agent\_task.c, 181  
/home/dko/Projects/mobilec/src/ams.c, 182  
/home/dko/Projects/mobilec/src/barrier.c, 184  
/home/dko/Projects/mobilec/src/cmd\_prompt.c, 186  
/home/dko/Projects/mobilec/src/connection.c, 189  
/home/dko/Projects/mobilec/src/data\_structures.c, 190  
/home/dko/Projects/mobilec/src/df.c, 192  
/home/dko/Projects/mobilec/src/dynstring.c, 197  
/home/dko/Projects/mobilec/src/fipa\_acl.c, 198  
/home/dko/Projects/mobilec/src/fipa\_envelope.c, 211  
/home/dko/Projects/mobilec/src/include/acc.h, 219  
/home/dko/Projects/mobilec/src/include/agent.h, 222  
/home/dko/Projects/mobilec/src/include/agent\_datastate.h, 226  
/home/dko/Projects/mobilec/src/include/agent\_lib.h, 228  
/home/dko/Projects/mobilec/src/include/agent\_mailbox.h, 240  
/home/dko/Projects/mobilec/src/include/agent\_task.h, 242  
/home/dko/Projects/mobilec/src/include/ams.h, 244  
/home/dko/Projects/mobilec/src/include/ap\_queue\_template.h, 246  
/home/dko/Projects/mobilec/src/include/barrier.h, 249  
/home/dko/Projects/mobilec/src/include/cmd\_prompt.h, 252  
/home/dko/Projects/mobilec/src/include/commands.h, 255  
/home/dko/Projects/mobilec/src/include/commands.x.h, 258  
/home/dko/Projects/mobilec/src/include/connection.h, 260  
/home/dko/Projects/mobilec/src/include/data\_structures.h, 262  
/home/dko/Projects/mobilec/src/include/df.h, 266  
/home/dko/Projects/mobilec/src/include/df\_request.x.h, 273  
/home/dko/Projects/mobilec/src/include/dynstring.h, 274  
/home/dko/Projects/mobilec/src/include/fipa\_acl.h, 276  
/home/dko/Projects/mobilec/src/include/fipa\_acl\_envelope.h, 293  
/home/dko/Projects/mobilec/src/include/fipa\_comm.h, 301  
/home/dko/Projects/mobilec/src/include/host\_id.h, 309  
/home/dko/Projects/mobilec/src/include/interpreter\_variable\_data.h, 310  
/home/dko/Projects/mobilec/src/include/libmc.h, 312  
/home/dko/Projects/mobilec/src/include/macros.h, 365  
/home/dko/Projects/mobilec/src/include/mc\_error.h, 377  
/home/dko/Projects/mobilec/src/include/mc\_platform.h, 379  
/home/dko/Projects/mobilec/src/include/mc\_rwlock.h, 381  
/home/dko/Projects/mobilec/src/include/message.h, 383  
/home/dko/Projects/mobilec/src/include/mobilec.h, 387  
/home/dko/Projects/mobilec/src/include/mtp\_http.h, 388  
/home/dko/Projects/mobilec/src/include/xml\_compose.h, 394  
/home/dko/Projects/mobilec/src/include/xml\_helper.h, 398  
/home/dko/Projects/mobilec/src/include/xml\_parser.h, 400  
/home/dko/Projects/mobilec/src/libmc.c, 404  
/home/dko/Projects/mobilec/src/mc\_list/list.c, 462

- /home/dko/Projects/mobilec/src/mc\_list/list.h, [464](#)
- /home/dko/Projects/mobilec/src/mc\_platform.c, [467](#)
- /home/dko/Projects/mobilec/src/mc\_rwlock.c, [469](#)
- /home/dko/Projects/mobilec/src/mc\_sync/sync\_list.c, [471](#)
- /home/dko/Projects/mobilec/src/mc\_sync/sync\_list.h, [474](#)
- /home/dko/Projects/mobilec/src/message.c, [477](#)
- /home/dko/Projects/mobilec/src/message\_queue.c, [481](#)
- /home/dko/Projects/mobilec/src/mtp\_http.c, [482](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/config.h, [487](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-attr.c, [491](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-entity.c, [492](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-file.c, [494](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-index.c, [501](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-node.c, [503](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-private.c, [506](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-search.c, [508](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-set.c, [509](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml-string.c, [511](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/mxml.h, [512](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/mxmldoc.c, [524](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/test/class.cxx, [528](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/test/enum.cxx, [529](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/test/function.cxx, [530](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/test/struct.cxx, [531](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/testmxml.c, [532](#)
- /home/dko/Projects/mobilec/src/mxml-2.2.2/winconfig.h, [533](#)
- /home/dko/Projects/mobilec/src/security/asm.c, [536](#)
- /home/dko/Projects/mobilec/src/security/asm.h, [537](#)
- /home/dko/Projects/mobilec/src/security/asm\_message\_composer.c, [538](#)
- /home/dko/Projects/mobilec/src/security/asm\_message\_composer.h, [539](#)
- /home/dko/Projects/mobilec/src/security/asm\_message\_parser.c, [540](#)
- /home/dko/Projects/mobilec/src/security/asm\_message\_parser.h, [541](#)
- /home/dko/Projects/mobilec/src/security/asm\_node.c, [542](#)
- /home/dko/Projects/mobilec/src/security/asm\_node.h, [543](#)
- /home/dko/Projects/mobilec/src/security/interface.c, [544](#)
- /home/dko/Projects/mobilec/src/security/interface.h, [548](#)
- /home/dko/Projects/mobilec/src/security/mc\_dh.c, [553](#)
- /home/dko/Projects/mobilec/src/security/mc\_dh.h, [554](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/aes.h, [555](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/arc4.h, [559](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/base64.h, [561](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/bignum.h, [563](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/bn\_mul.h, [579](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/certs.h, [581](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/config.h, [488](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/debug.h, [583](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/des.h, [586](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/dhm.h, [591](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/havege.h, [596](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md2.h, [598](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md4.h, [602](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/md5.h, [606](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/net.h, [611](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/openssl.h, [616](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/padlock.h, [620](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/rsa.h, [621](#)

- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha1.h, [631](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha2.h, [636](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/sha4.h, [641](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/ssl.h, [646](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/timing.h, [670](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/include/xyssl/x509.h, [672](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/aes.c, [687](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/arc4.c, [696](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/base64.c, [699](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/bignum.c, [703](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/certs.c, [720](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/debug.c, [724](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/des.c, [726](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/dhm.c, [739](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/havege.c, [743](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md2.c, [745](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md4.c, [746](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/md5.c, [747](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/net.c, [756](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/padlock.c, [760](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/rsa.c, [761](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha1.c, [771](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha2.c, [780](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/sha4.c, [789](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl\_cli.c, [798](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl\_srv.c, [801](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/ssl\_tls.c, [804](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/timing.c, [818](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/library/x509parse.c, [821](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/aes/aesencrypt2.c, [830](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/hello.c, [832](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/md5sum.c, [833](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/sha1sum.c, [834](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/hash/sha2sum.c, [835](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh\_client.c, [836](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh\_genprime.c, [837](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/dh\_server.c, [838](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/mpi\_demo.c, [839](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa\_genkey.c, [840](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa\_sign.c, [841](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/pkey/rsa\_verify.c, [842](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl\_client1.c, [843](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl\_client2.c, [845](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/ssl/ssl\_server.c, [847](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/benchmark.c, [850](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/selftest.c, [852](#)
- /home/dko/Projects/mobilec/src/security/xyssl-0.9/programs/test/ssl\_test.c, [853](#)
- /home/dko/Projects/mobilec/src/util/mc\_genkey.c, [859](#)
- /home/dko/Projects/mobilec/src/winconfig.h, [535](#)
- /home/dko/Projects/mobilec/src/xml\_compose.c, [860](#)
- /home/dko/Projects/mobilec/src/xml\_helper.c, [864](#)
- /home/dko/Projects/mobilec/src/xml\_parser.c, [866](#)
- \_\_ERROR\_CODE\_E\_\_  
libmc.h, [320](#)
- \_\_FIPA\_PERFORMATIVE\_E\_\_  
libmc.h, [320](#)
- \_\_XOPEN\_SOURCE  
xml\_parser.c, [866](#)
- \_\_RSA\_Passthrough

- openssl.h, 619
- \_hr\_time, 9
  - start, 9
- \_ssl\_context, 10
  - authmode, 11
  - ca\_chain, 11
  - ciphers, 11
  - client\_auth, 11
  - ctx\_dec, 12
  - ctx\_enc, 12
  - dhm\_ctx, 12
  - do\_crypt, 12
  - endpoint, 12
  - f\_dbg, 12
  - f\_recv, 13
  - f\_rng, 13
  - f\_send, 13
  - fin\_md5, 13
  - fin\_sha1, 13
  - hostname, 13
  - hostname\_len, 13
  - in\_ctr, 13
  - in\_hdr, 14
  - in\_hslen, 14
  - in\_left, 14
  - in\_msg, 14
  - in\_msglen, 14
  - in\_msgtype, 14
  - in\_offt, 15
  - iv\_dec, 15
  - iv\_enc, 15
  - ivlen, 15
  - keylen, 15
  - mac\_dec, 15
  - mac\_enc, 16
  - maclen, 16
  - major\_ver, 16
  - max\_major\_ver, 16
  - max\_minor\_ver, 16
  - minlen, 16
  - minor\_ver, 16
  - nb\_zero, 17
  - out\_ctr, 17
  - out\_hdr, 17
  - out\_left, 17
  - out\_msg, 17
  - out\_msglen, 17
  - out\_msgtype, 18
  - own\_cert, 18
  - p\_dbg, 18
  - p\_recv, 18
  - p\_rng, 18
  - p\_send, 18
  - peer\_cert, 19
  - peer\_cn, 19
  - pmslen, 19
  - premaster, 19
  - randbytes, 19
  - resume, 19
  - rsa\_key, 20
  - s\_get, 20
  - s\_set, 20
  - session, 20
  - state, 20
  - timeout, 20
  - verify\_result, 21
- \_ssl\_session, 22
  - cipher, 22
  - id, 22
  - length, 22
  - master, 22
  - next, 23
  - start, 23
- \_x509\_buf, 24
  - len, 24
  - p, 24
  - tag, 24
- \_x509\_cert, 25
  - ca\_istrue, 25
  - issuer, 25
  - issuer\_id, 25
  - issuer\_raw, 25
  - max\_pathlen, 26
  - next, 26
  - pk\_oid, 26
  - raw, 26
  - rsa, 26
  - serial, 26
  - sig, 26
  - sig\_oid1, 26
  - sig\_oid2, 27
  - subject, 27
  - subject\_id, 27
  - subject\_raw, 27
  - tbs, 27
  - v3\_ext, 27
  - valid\_from, 27
  - valid\_to, 27
  - version, 27
- \_x509\_name, 29
  - next, 29
  - oid, 29
  - val, 29
- \_x509\_node, 30
  - data, 30
  - end, 30
  - len, 30
  - p, 30

- [\\_x509\\_raw](#), [31](#)
  - [issuer](#), [31](#)
  - [raw](#), [31](#)
  - [serial](#), [31](#)
  - [sign](#), [31](#)
  - [signalg](#), [31](#)
  - [subject](#), [31](#)
  - [subpubkey](#), [31](#)
  - [tbs](#), [32](#)
  - [tbs\\_signalg](#), [32](#)
  - [validity](#), [32](#)
  - [version](#), [32](#)
- [\\_x509\\_time](#), [33](#)
  - [day](#), [33](#)
  - [hour](#), [33](#)
  - [min](#), [33](#)
  - [mon](#), [33](#)
  - [sec](#), [33](#)
  - [year](#), [33](#)
- [acc](#)
  - [mc\\_platform\\_s](#), [111](#)
- [acc.c](#)
  - [acc\\_connection\\_Thread](#), [169](#)
  - [acc\\_Destroy](#), [170](#)
  - [acc\\_Initialize](#), [170](#)
  - [acc\\_MessageHandlerThread](#), [170](#)
  - [acc\\_Start](#), [170](#)
  - [acc\\_Thread](#), [170](#)
  - [auth\\_conn\\_rece\\_key](#), [171](#)
  - [BACKLOG](#), [169](#)
  - [BUFLen](#), [169](#)
  - [CONN\\_THREADS](#), [169](#)
  - [CONNECT\\_THREAD\\_EXIT](#), [169](#)
  - [listen\\_Thread](#), [171](#)
  - [udplisten\\_Thread](#), [171](#)
  - [UDPPort](#), [169](#)
- [acc.h](#)
  - [acc\\_connection\\_Thread](#), [219](#)
  - [acc\\_Destroy](#), [220](#)
  - [acc\\_MessageHandlerThread](#), [220](#)
  - [acc\\_Start](#), [220](#)
  - [acc\\_Thread](#), [220](#)
  - [addr](#), [221](#)
  - [client\\_fd](#), [221](#)
  - [listen\\_Thread](#), [220](#)
  - [port](#), [221](#)
  - [server\\_fd](#), [221](#)
  - [STRUCT](#), [220](#), [221](#)
  - [udplisten\\_Thread](#), [221](#)
- [acc\\_connection\\_Thread](#)
  - [acc.c](#), [169](#)
  - [acc.h](#), [219](#)
- [acc\\_Destroy](#)
  - [acc.c](#), [170](#)
  - [acc.h](#), [220](#)
- [acc\\_Initialize](#)
  - [acc.c](#), [170](#)
- [acc\\_MessageHandlerThread](#)
  - [acc.c](#), [170](#)
  - [acc.h](#), [220](#)
- [acc\\_Start](#)
  - [acc.c](#), [170](#)
  - [acc.h](#), [220](#)
- [acc\\_Thread](#)
  - [acc.c](#), [170](#)
  - [acc.h](#), [220](#)
- [accept\\_ranges](#)
  - [mtp\\_http\\_s](#), [134](#)
- [ACCEPTED](#)
  - [mtp\\_http.h](#), [390](#)
- [acl\\_representation](#)
  - [fipa\\_acl\\_Param\\_s](#), [73](#)
- [action\\_list](#)
  - [fipa\\_comm\\_performative\\_t](#), [79](#)
- [add\\_variable](#)
  - [mxmldoc.c](#), [525](#)
- [addr](#)
  - [acc.h](#), [221](#)
  - [connection\\_s](#), [58](#)
  - [message\\_s](#), [127](#)
- [addresses](#)
  - [fipa\\_agent\\_identifier\\_s](#), [75](#)
- [aes.c](#)
  - [aes\\_crypt\\_cbc](#), [690](#)
  - [aes\\_crypt\\_cfb](#), [690](#)
  - [aes\\_crypt\\_ecb](#), [691](#)
  - [AES\\_FROUND](#), [688](#)
  - [aes\\_gen\\_tables](#), [691](#)
  - [aes\\_init\\_done](#), [692](#)
  - [AES\\_RROUND](#), [688](#)
  - [aes\\_self\\_test](#), [691](#)
  - [aes\\_setkey\\_dec](#), [691](#)
  - [aes\\_setkey\\_enc](#), [692](#)
  - [aes\\_test\\_cbc\\_dec](#), [692](#)
  - [aes\\_test\\_cbc\\_enc](#), [692](#)
  - [aes\\_test\\_cfb\\_dec](#), [693](#)
  - [aes\\_test\\_cfb\\_enc](#), [693](#)
  - [aes\\_test\\_ecb\\_dec](#), [693](#)
  - [aes\\_test\\_ecb\\_enc](#), [694](#)
  - [FSb](#), [694](#)
  - [FT0](#), [694](#)
  - [FT1](#), [694](#)
  - [FT2](#), [694](#)
  - [FT3](#), [694](#)
  - [GET\\_ULONG\\_LE](#), [689](#)
  - [MUL](#), [689](#)
  - [PUT\\_ULONG\\_LE](#), [689](#)

- RCON, 695
- ROTL8, 690
- RSb, 695
- RT0, 695
- RT1, 695
- RT2, 695
- RT3, 695
- XTIME, 690
- aes.h
  - aes\_crypt\_cbc, 556
  - aes\_crypt\_cfb, 556
  - aes\_crypt\_ecb, 556
  - AES\_DECRYPT, 555
  - AES\_ENCRYPT, 555
  - aes\_self\_test, 557
  - aes\_setkey\_dec, 557
  - aes\_setkey\_enc, 557
- AES\_BLOCK\_SIZE
  - openssl.h, 617
- AES\_cbc\_encrypt
  - openssl.h, 617
- aes\_context, 35
  - buf, 35
  - nr, 35
  - rk, 35
- aes\_crypt\_cbc
  - aes.c, 690
  - aes.h, 556
- aes\_crypt\_cfb
  - aes.c, 690
  - aes.h, 556
- aes\_crypt\_ecb
  - aes.c, 691
  - aes.h, 556
- AES\_DECRYPT
  - aes.h, 555
- aes\_en\_de
  - interface.c, 544
  - interface.h, 550
- AES\_ENCRYPT
  - aes.h, 555
- AES\_FROUND
  - aes.c, 688
- aes\_gen\_tables
  - aes.c, 691
- aes\_init\_done
  - aes.c, 692
- AES\_KEY
  - openssl.h, 617
- AES\_key
  - connection\_s, 58
- AES\_RROUND
  - aes.c, 688
- aes\_self\_test
  - aes.c, 691
  - aes.h, 557
- AES\_set\_decrypt\_key
  - openssl.h, 617
- AES\_set\_encrypt\_key
  - openssl.h, 617
- aes\_setkey\_dec
  - aes.c, 691
  - aes.h, 557
- aes\_setkey\_enc
  - aes.c, 692
  - aes.h, 557
- AES\_SIZE
  - openssl.h, 617
- aes\_test\_cbc\_dec
  - aes.c, 692
- aes\_test\_cbc\_enc
  - aes.c, 692
- aes\_test\_cfb\_dec
  - aes.c, 693
- aes\_test\_cfb\_enc
  - aes.c, 693
- aes\_test\_ecb\_dec
  - aes.c, 693
- aes\_test\_ecb\_enc
  - aes.c, 694
- aesencrypt2.c
  - main, 831
  - MODE\_DECRYPT, 830
  - MODE\_ENCRYPT, 830
  - USAGE, 830
- agency
  - mc\_platform\_s, 111
  - stationary\_agent\_info\_s, 162
- agency\_p
  - libmc.h, 321
- agency\_s, 36
  - agentInitCallback, 36
  - agentInitUserData, 36
  - bluetooth, 36
  - client, 36
  - default\_agentstatus, 37
  - enable\_security, 37
  - filename, 37
  - hostName, 37
  - initInterps, 37
  - known\_host\_filename, 37
  - last\_error, 37
  - mc\_platform, 37
  - portno, 38
  - portnoc, 38
  - priv\_key\_filename, 38
  - server, 38
  - stack\_size, 38



- threads, 38
- agency\_t
  - libmc.h, 321
- agent
  - fipa\_comm\_t, 85
  - stationary\_agent\_info\_s, 162
- agent.c
  - agent\_AddPersistentVariable, 172
  - agent\_ChScriptInitVar, 172
  - agent\_Copy, 173
  - agent\_Destroy, 173
  - agent\_Initialize, 173
  - agent\_New, 174
  - agent\_NewBinary, 174
  - agent\_RunChScript, 174
  - agent\_RunChScriptThread, 174
- agent.h
  - agent\_AddPersistentVariable, 223
  - agent\_ChScriptInitVar, 223
  - agent\_Copy, 224
  - agent\_Destroy, 224
  - agent\_Initialize, 224
  - agent\_New, 224
  - agent\_NewBinary, 224
  - agent\_p, 223
  - agent\_RunChScript, 225
  - agent\_RunChScriptThread, 225
  - AGENT\_T, 222
  - agent\_t, 223
  - MCAgent\_t, 223
- AGENT\_UPDATE
  - message.h, 384
- agent\_AddPersistentVariable
  - agent.c, 172
  - agent.h, 223
- agent\_address
  - agent\_s, 44
  - fipa\_comm\_t, 85
- agent\_ChScriptInitVar
  - agent.c, 172
  - agent.h, 223
- agent\_code
  - agent\_datastate\_s, 40
- agent\_code\_ids
  - agent\_datastate\_s, 40
- agent\_codes
  - agent\_datastate\_s, 40
- agent\_Copy
  - agent.c, 173
  - agent.h, 224
- agent\_datastate.c
  - agent\_datastate\_Copy, 176
  - agent\_datastate\_Destroy, 176
  - agent\_datastate\_New, 176
- agent\_datastate.h
  - agent\_datastate\_Copy, 226
  - agent\_datastate\_Destroy, 226
  - agent\_datastate\_New, 227
  - agent\_datastate\_p, 226
  - agent\_datastate\_t, 226
- agent\_datastate\_Copy
  - agent\_datastate.c, 176
  - agent\_datastate.h, 226
- agent\_datastate\_Destroy
  - agent\_datastate.c, 176
  - agent\_datastate.h, 226
- agent\_datastate\_New
  - agent\_datastate.c, 176
  - agent\_datastate.h, 227
- agent\_datastate\_p
  - agent\_datastate.h, 226
- agent\_datastate\_s, 40
  - agent\_code, 40
  - agent\_code\_ids, 40
  - agent\_codes, 40
  - init\_agent\_status, 40
  - number\_of\_tasks, 41
  - persistent, 41
  - progress\_modifier, 41
  - return\_data, 41
  - task\_progress, 41
  - tasks, 41
  - xml\_agent\_root, 42
  - xml\_root, 42
- agent\_datastate\_t
  - agent\_datastate.h, 226
- agent\_Destroy
  - agent.c, 173
  - agent.h, 224
- agent\_Initialize
  - agent.c, 173
  - agent.h, 224
- agent\_interp
  - agent\_s, 44
- agent\_lib.h
  - MC\_AclAddReceiver\_chdl, 229
  - MC\_AclAddReplyTo\_chdl, 229
  - MC\_AclDestroy\_chdl, 229
  - MC\_AclGetContent\_chdl, 229
  - MC\_AclGetConversationID\_chdl, 230
  - MC\_AclGetPerformative\_chdl, 230
  - MC\_AclGetProtocol\_chdl, 230
  - MC\_AclGetSender\_chdl, 230
  - MC\_AclNew\_chdl, 230
  - MC\_AclPost\_chdl, 230
  - MC\_AclReply\_chdl, 230
  - MC\_AclRetrieve\_chdl, 231
  - MC\_AclSend\_chdl, 231

- MC\_AclSetContent\_chdl, 231
- MC\_AclSetConversationID\_chdl, 231
- MC\_AclSetPerformative\_chdl, 231
- MC\_AclSetProtocol\_chdl, 231
- MC\_AclSetSender\_chdl, 231
- MC\_AclWaitRetrieve\_chdl, 232
- MC\_AddAgent\_chdl, 232
- MC\_AgentAddTask\_chdl, 232
- MC\_AgentVariableRetrieve\_chdl, 232
- MC\_AgentVariableSave\_chdl, 232
- MC\_Barrier\_chdl, 232
- MC\_BarrierDelete\_chdl, 232
- MC\_BarrierInit\_chdl, 233
- MC\_CallAgentFunc\_chdl, 233
- MC\_ComposeAgent\_chdl, 233
- MC\_ComposeAgentFromFile\_chdl, 233
- MC\_ComposeAgentFromFileWithWorkgroup\_chdl, 233
- MC\_ComposeAgentWithWorkgroup\_chdl, 233
- MC\_CondBroadcast\_chdl, 233
- MC\_CondReset\_chdl, 234
- MC\_CondSignal\_chdl, 234
- MC\_CondWait\_chdl, 234
- MC\_DeleteAgent\_chdl, 234
- MC\_DeleteAgentWG\_chdl, 234
- MC\_DeregisterService\_chdl, 234
- MC\_DestroyServiceSearchResult\_chdl, 234
- MC\_End\_chdl, 235
- MC\_FindAgentByID\_chdl, 235
- MC\_FindAgentByName\_chdl, 235
- MC\_GetAgentID\_chdl, 235
- MC\_GetAgentName\_chdl, 235
- MC\_GetAgentNumTasks\_chdl, 235
- MC\_GetAgentStatus\_chdl, 235
- MC\_GetAgentXMLString\_chdl, 236
- MC\_GetTimeOfDay\_chdl, 236
- MC\_HaltAgency\_chdl, 236
- MC\_MigrateAgent\_chdl, 236
- MC\_MutexLock\_chdl, 236
- MC\_MutexUnlock\_chdl, 236
- MC\_PrintAgentCode\_chdl, 236
- MC\_RegisterService\_chdl, 237
- MC\_ResumeAgency\_chdl, 237
- MC\_RetrieveAgent\_chdl, 237
- MC\_RetrieveAgentCode\_chdl, 237
- MC\_SaveData\_chdl, 237
- MC\_SearchForService\_chdl, 237
- MC\_SemaphorePost\_chdl, 238
- MC\_SemaphoreWait\_chdl, 238
- MC\_SendAgentMigrationMessage\_chdl, 238
- MC\_SendAgentMigrationMessageFile\_chdl, 238
- MC\_SendSteerCommand\_chdl, 238
- MC\_SetAgentStatus\_chdl, 238
- MC\_SetDefaultAgentStatus\_chdl, 238
- MC\_SyncDelete\_chdl, 239
- MC\_SyncInit\_chdl, 239
- MC\_TerminateAgent\_chdl, 239
- MC\_TerminateAgentWG\_chdl, 239
- agent\_mailbox.c
  - agent\_mailbox\_Copy, 177
  - agent\_mailbox\_Destroy, 177
  - agent\_mailbox\_New, 177
  - agent\_mailbox\_Post, 177
  - agent\_mailbox\_Retrieve, 177
  - agent\_mailbox\_WaitRetrieve, 177
- agent\_mailbox.h
  - agent\_mailbox\_Copy, 240
  - agent\_mailbox\_Destroy, 240
  - agent\_mailbox\_New, 240
  - agent\_mailbox\_p, 240
  - agent\_mailbox\_Post, 241
  - agent\_mailbox\_Retrieve, 241
  - agent\_mailbox\_t, 240
  - agent\_mailbox\_WaitRetrieve, 241
- agent\_mailbox\_Copy
  - agent\_mailbox.c, 177
  - agent\_mailbox.h, 240
- agent\_mailbox\_Destroy
  - agent\_mailbox.c, 177
  - agent\_mailbox.h, 240
- agent\_mailbox\_New
  - agent\_mailbox.c, 177
  - agent\_mailbox.h, 240
- agent\_mailbox\_p
  - agent\_mailbox.h, 240
  - data\_structures.h, 264
- agent\_mailbox\_Post
  - agent\_mailbox.c, 177
  - agent\_mailbox.h, 241
- agent\_mailbox\_Retrieve
  - agent\_mailbox.c, 177
  - agent\_mailbox.h, 241
- agent\_mailbox\_s, 43
  - mail\_queue, 43
- agent\_mailbox\_t
  - agent\_mailbox.h, 240
- agent\_mailbox\_WaitRetrieve
  - agent\_mailbox.c, 177
  - agent\_mailbox.h, 241
- agent\_name
  - fipa\_comm\_t, 85
- agent\_New
  - agent.c, 174
  - agent.h, 224
- agent\_NewBinary
  - agent.c, 174

- agent.h, 224
- agent\_p
  - agent.h, 223
  - libmc.h, 321
- agent\_persistent
  - agent\_s, 45
- agent\_pipe\_active
  - agent\_s, 45
- agent\_pipe\_ready\_to\_read
  - agent\_s, 45
- agent\_queue
  - mc\_platform\_s, 111
- agent\_queue\_Flush
  - data\_structures.c, 190
  - data\_structures.h, 263
- agent\_ready\_to\_send
  - agent\_s, 45
- agent\_return\_data
  - agent\_task\_s, 49
- agent\_return\_data.c
  - interpreter\_variable\_data\_Copy, 179
  - interpreter\_variable\_data\_Destroy, 179
  - interpreter\_variable\_data\_Initialize, 179
  - interpreter\_variable\_data\_InitializeFromAgent, 179
  - interpreter\_variable\_data\_New, 180
- agent\_return\_xml\_parse
  - xml\_parser.c, 867
- agent\_RunChScript
  - agent.c, 174
  - agent.h, 225
- agent\_RunChScriptThread
  - agent.c, 174
  - agent.h, 225
- agent\_s, 44
  - agent\_address, 44
  - agent\_interp, 44
  - agent\_persistent, 45
  - agent\_pipe\_active, 45
  - agent\_pipe\_ready\_to\_read, 45
  - agent\_ready\_to\_send, 45
  - agent\_script\_ready, 45
  - agent\_status, 45
  - agent\_thread, 45
  - agent\_thread\_id, 46
  - agent\_type, 46
  - arrival\_time, 46
  - binary, 46
  - connect\_id, 46
  - datastate, 46
  - home, 46
  - home\_port, 47
  - id, 47
  - lock, 47
  - mailbox, 47
  - mc\_platform, 47
  - name, 47
  - orphan, 47
  - owner, 48
  - return\_data, 48
  - run\_lock, 48
  - sender, 48
  - wg\_code, 48
- agent\_script\_ready
  - agent\_s, 45
- agent\_status
  - agent\_s, 45
- AGENT\_T
  - agent.h, 222
  - libmc.h, 320
- agent\_t
  - agent.h, 223
  - libmc.h, 321
- agent\_task.c
  - agent\_task\_Copy, 181
  - agent\_task\_Destroy, 181
  - agent\_task\_New, 181
- agent\_task.h
  - agent\_task\_Copy, 242
  - agent\_task\_Destroy, 242
  - agent\_task\_New, 243
  - agent\_task\_p, 242
  - agent\_task\_t, 242
- agent\_task\_Copy
  - agent\_task.c, 181
  - agent\_task.h, 242
- agent\_task\_Destroy
  - agent\_task.c, 181
  - agent\_task.h, 242
- agent\_task\_New
  - agent\_task.c, 181
  - agent\_task.h, 243
- agent\_task\_p
  - agent\_task.h, 242
- agent\_task\_s, 49
  - agent\_return\_data, 49
  - agent\_variable\_list, 49
  - code\_id, 49
  - init\_agent\_status, 49
  - num\_saved\_variables, 50
  - number\_of\_elements, 50
  - persistent, 50
  - saved\_variables, 50
  - server\_name, 50
  - size\_of\_element\_array, 50
  - var\_name, 50
- agent\_task\_t
  - agent\_task.h, 242

- agent\_thread
  - agent\_s, [45](#)
- agent\_thread\_id
  - agent\_s, [46](#)
- agent\_type
  - agent\_s, [46](#)
- agent\_variable\_list
  - agent\_task\_s, [49](#)
- agent\_xml\_compose
  - xml\_compose.c, [860](#)
  - xml\_compose.h, [394](#)
- agent\_xml\_compose\_\_agent\_code
  - xml\_compose.c, [860](#)
  - xml\_compose.h, [394](#)
- agent\_xml\_compose\_\_agent\_data
  - xml\_compose.c, [860](#)
  - xml\_compose.h, [394](#)
- agent\_xml\_compose\_\_create\_row\_nodes
  - xml\_compose.c, [861](#)
  - xml\_compose.h, [395](#)
- agent\_xml\_compose\_\_data
  - xml\_compose.c, [861](#)
  - xml\_compose.h, [395](#)
- agent\_xml\_compose\_\_gaf\_message
  - xml\_compose.c, [861](#)
  - xml\_compose.h, [395](#)
- agent\_xml\_compose\_\_home
  - xml\_compose.c, [861](#)
  - xml\_compose.h, [395](#)
- agent\_xml\_compose\_\_message
  - xml\_compose.c, [861](#)
  - xml\_compose.h, [395](#)
- agent\_xml\_compose\_\_mobile\_agent
  - xml\_compose.c, [862](#)
  - xml\_compose.h, [395](#)
- agent\_xml\_compose\_\_name
  - xml\_compose.c, [862](#)
  - xml\_compose.h, [396](#)
- agent\_xml\_compose\_\_owner
  - xml\_compose.c, [862](#)
  - xml\_compose.h, [396](#)
- agent\_xml\_compose\_\_row
  - xml\_compose.c, [862](#)
  - xml\_compose.h, [396](#)
- agent\_xml\_compose\_\_task
  - xml\_compose.c, [862](#)
  - xml\_compose.h, [396](#)
- agent\_xml\_compose\_\_tasks
  - xml\_compose.c, [862](#)
  - xml\_compose.h, [396](#)
- agent\_xml\_compose\_\_wg\_code
  - xml\_compose.c, [863](#)
  - xml\_compose.h, [396](#)
- agent\_xml\_flag
  - message\_s, [127](#)
- agent\_xml\_parse
  - xml\_parser.c, [867](#)
  - xml\_parser.h, [400](#)
- agent\_xml\_parse\_\_agent\_code
  - xml\_parser.c, [867](#)
  - xml\_parser.h, [400](#)
- agent\_xml\_parse\_\_agent\_data
  - xml\_parser.c, [867](#)
  - xml\_parser.h, [400](#)
- agent\_xml\_parse\_\_data
  - xml\_parser.c, [867](#)
  - xml\_parser.h, [401](#)
- agent\_xml\_parse\_\_fill\_row\_data
  - xml\_parser.c, [867](#)
  - xml\_parser.h, [401](#)
- agent\_xml\_parse\_\_home
  - xml\_parser.c, [868](#)
  - xml\_parser.h, [401](#)
- agent\_xml\_parse\_\_mobile\_agent
  - xml\_parser.c, [868](#)
  - xml\_parser.h, [401](#)
- agent\_xml\_parse\_\_name
  - xml\_parser.c, [868](#)
  - xml\_parser.h, [401](#)
- agent\_xml\_parse\_\_owner
  - xml\_parser.c, [868](#)
  - xml\_parser.h, [402](#)
- agent\_xml\_parse\_\_row
  - xml\_parser.c, [868](#)
  - xml\_parser.h, [402](#)
- agent\_xml\_parse\_\_sender
  - xml\_parser.c, [868](#)
  - xml\_parser.h, [402](#)
- agent\_xml\_parse\_\_task
  - xml\_parser.c, [869](#)
  - xml\_parser.h, [402](#)
- agent\_xml\_parse\_\_tasks
  - xml\_parser.c, [869](#)
  - xml\_parser.h, [402](#)
- agent\_xml\_parse\_\_wg\_code
  - xml\_parser.c, [869](#)
  - xml\_parser.h, [402](#)
- agentInitCallback
  - agency\_s, [36](#)
- agentInitUserData
  - agency\_s, [36](#)
- alarmed
  - timing.c, [819](#)
  - timing.h, [671](#)
- alloc\_nodes
  - mxml\_index\_s, [141](#)
- ams
  - mc\_platform\_s, [111](#)

- ams.c
  - ams\_Destroy, 182
  - ams\_Initialize, 182
  - ams\_ManageAgentList, 182
  - ams\_Print, 182
  - ams\_Start, 183
  - ams\_Thread, 183
- ams.h
  - ams\_Destroy, 244
  - ams\_ManageAgentList, 244
  - ams\_Print, 244
  - ams\_RemoveFinishedAgents, 244
  - ams\_Start, 244
  - ams\_Thread, 244
  - STRUCT, 245
- ams\_Destroy
  - ams.c, 182
  - ams.h, 244
- ams\_Initialize
  - ams.c, 182
- ams\_ManageAgentList
  - ams.c, 182
  - ams.h, 244
- ams\_Print
  - ams.c, 182
  - ams.h, 244
- ams\_RemoveFinishedAgents
  - ams.h, 244
- ams\_Start
  - ams.c, 183
  - ams.h, 244
- ams\_Thread
  - ams.c, 183
  - ams.h, 244
- AP\_GENERIC\_Copy
  - data\_structures.c, 190
- AP\_GENERIC\_Destroy
  - data\_structures.c, 190
  - data\_structures.h, 263
- AP\_GENERIC\_p
  - ap\_queue\_template.h, 248
- AP\_GENERIC\_s, 52
  - none, 52
- AP\_GENERIC\_t
  - ap\_queue\_template.h, 248
- AP\_QUEUE\_DECL\_TEMPLATE
  - ap\_queue\_template.h, 246
  - data\_structures.h, 263
- AP\_QUEUE\_GENERIC\_DECL\_TEMPLATE
  - ap\_queue\_template.h, 247
  - data\_structures.h, 263, 264
- AP\_QUEUE\_REMOVE\_TEMPLATE
  - ap\_queue\_template.h, 247
- AP\_QUEUE\_SEARCH\_TEMPLATE
  - ap\_queue\_template.h, 247
  - data\_structures.c, 190, 191
- AP\_QUEUE\_STD\_DEFN\_TEMPLATE
  - ap\_queue\_template.h, 248
  - data\_structures.c, 191
- ap\_queue\_template.h
  - AP\_GENERIC\_p, 248
  - AP\_GENERIC\_t, 248
  - AP\_QUEUE\_DECL\_TEMPLATE, 246
  - AP\_QUEUE\_GENERIC\_DECL\_TEMPLATE, 247
  - AP\_QUEUE\_REMOVE\_TEMPLATE, 247
  - AP\_QUEUE\_SEARCH\_TEMPLATE, 247
  - AP\_QUEUE\_STD\_DEFN\_TEMPLATE, 248
  - interpreter\_queue\_CreateRetrieve, 248
- append\_nonce\_to\_MA
  - interface.c, 544
- arc4.c
  - arc4\_crypt, 696
  - arc4\_self\_test, 696
  - arc4\_setup, 696
  - arc4\_test\_ct, 697
  - arc4\_test\_key, 697
  - arc4\_test\_pt, 697
- arc4.h
  - arc4\_crypt, 559
  - arc4\_self\_test, 559
  - arc4\_setup, 559
- arc4\_context, 53
  - m, 53
  - x, 53
  - y, 53
- arc4\_crypt
  - arc4.c, 696
  - arc4.h, 559
- arc4\_self\_test
  - arc4.c, 696
  - arc4.h, 559
- arc4\_setup
  - arc4.c, 696
  - arc4.h, 559
- arc4\_test\_ct
  - arc4.c, 697
- arc4\_test\_key
  - arc4.c, 697
- arc4\_test\_pt
  - arc4.c, 697
- args
  - command\_s, 57
  - stationary\_agent\_info\_s, 162
- array\_dim
  - interpreter\_variable\_data\_s, 106
- array\_extent
  - interpreter\_variable\_data\_s, 106

- arrival\_time
  - agent\_s, [46](#)
- asm\_message\_queue
  - mc\_platform\_s, [111](#)
- ASN1\_BIT\_STRING
  - x509.h, [674](#)
- ASN1\_BMP\_STRING
  - x509.h, [674](#)
- ASN1\_BOOLEAN
  - x509.h, [674](#)
- ASN1\_CONSTRUCTED
  - x509.h, [675](#)
- ASN1\_CONTEXT\_SPECIFIC
  - x509.h, [675](#)
- asn1\_get\_bool
  - x509parse.c, [822](#)
- asn1\_get\_int
  - x509parse.c, [822](#)
- asn1\_get\_len
  - x509parse.c, [822](#)
- asn1\_get\_mpi
  - x509parse.c, [822](#)
- asn1\_get\_tag
  - x509parse.c, [823](#)
- ASN1\_HASH\_MDX
  - rsa.h, [622](#)
- ASN1\_HASH\_SHA1
  - rsa.h, [622](#)
- ASN1\_IA5\_STRING
  - x509.h, [675](#)
- ASN1\_INTEGER
  - x509.h, [675](#)
- ASN1\_NULL
  - x509.h, [675](#)
- ASN1\_OCTET\_STRING
  - x509.h, [675](#)
- ASN1\_OID
  - x509.h, [675](#)
- ASN1\_PRIMITIVE
  - x509.h, [675](#)
- ASN1\_PRINTABLE\_STRING
  - x509.h, [676](#)
- ASN1\_SEQUENCE
  - x509.h, [676](#)
- ASN1\_SET
  - x509.h, [676](#)
- ASN1\_T61\_STRING
  - x509.h, [676](#)
- ASN1\_UNIVERSAL\_STRING
  - x509.h, [676](#)
- ASN1\_UTC\_TIME
  - x509.h, [676](#)
- ASN1\_UTF8\_STRING
  - x509.h, [676](#)
- attr
  - mxml\_index\_s, [141](#)
  - stationary\_agent\_info\_s, [162](#)
- attrs
  - mxml\_value\_s, [146](#)
- auth\_conn\_rece\_key
  - acc.c, [171](#)
- auth\_rece\_send\_msg
  - message.c, [478](#)
  - message.h, [384](#)
- authmode
  - \_ssl\_context, [11](#)
- BACKLOG
  - acc.c, [169](#)
- BAD\_REQUEST
  - mtp\_http.h, [390](#)
- BADCERT\_CN\_MISMATCH
  - x509.h, [676](#)
- BADCERT\_EXPIRED
  - x509.h, [676](#)
- BADCERT\_NOT\_TRUSTED
  - x509.h, [677](#)
- BADCERT\_REVOKED
  - x509.h, [677](#)
- bar
  - class.cxx, [528](#)
  - foo\_c, [98](#)
  - foo\_s, [101](#)
- barfoo
  - foo\_c, [98](#)
- barrier.c
  - barrier\_node\_Destroy, [184](#)
  - barrier\_node\_Initialize, [184](#)
  - barrier\_queue\_Add, [184](#)
  - barrier\_queue\_Delete, [184](#)
  - barrier\_queue\_Destroy, [185](#)
  - barrier\_queue\_Get, [185](#)
  - barrier\_queue\_New, [185](#)
  - barrier\_queue\_Pop, [185](#)
- barrier.h
  - barrier\_node\_Destroy, [250](#)
  - barrier\_node\_Initialize, [250](#)
  - barrier\_node\_p, [249](#)
  - barrier\_node\_t, [249](#)
  - barrier\_queue\_Add, [250](#)
  - barrier\_queue\_Delete, [250](#)
  - barrier\_queue\_Destroy, [250](#)
  - barrier\_queue\_Get, [250](#)
  - barrier\_queue\_New, [251](#)
  - barrier\_queue\_p, [249](#)
  - barrier\_queue\_Pop, [251](#)
  - barrier\_queue\_t, [249](#)
- barrier\_node\_Destroy

- barrier.c, 184
- barrier.h, 250
- barrier\_node\_Initialize
  - barrier.c, 184
  - barrier.h, 250
- barrier\_node\_p
  - barrier.h, 249
- barrier\_node\_s, 54
  - cond, 54
  - id, 54
  - lock, 54
  - num\_registered, 54
  - num\_waiting, 54
- barrier\_node\_t
  - barrier.h, 249
- barrier\_queue
  - mc\_platform\_s, 111
- barrier\_queue\_Add
  - barrier.c, 184
  - barrier.h, 250
- barrier\_queue\_Delete
  - barrier.c, 184
  - barrier.h, 250
- barrier\_queue\_Destroy
  - barrier.c, 185
  - barrier.h, 250
- barrier\_queue\_Get
  - barrier.c, 185
  - barrier.h, 250
- barrier\_queue\_New
  - barrier.c, 185
  - barrier.h, 251
- barrier\_queue\_p
  - barrier.h, 249
- barrier\_queue\_Pop
  - barrier.c, 185
  - barrier.h, 251
- barrier\_queue\_s, 55
  - list, 55
  - lock, 55
  - size, 55
- barrier\_queue\_t
  - barrier.h, 249
- base64.c
  - base64\_dec\_map, 700
  - base64\_decode, 699
  - base64\_enc\_map, 701
  - base64\_encode, 700
  - base64\_self\_test, 700
  - base64\_test\_dec, 701
  - base64\_test\_enc, 701
- base64.h
  - base64\_decode, 561
  - base64\_encode, 562
  - base64\_self\_test, 562
  - XYSSL\_ERR\_BASE64\_BUFFER\_TOO\_SMALL, 561
  - XYSSL\_ERR\_BASE64\_INVALID\_CHARACTER, 561
- base64\_dec\_map
  - base64.c, 700
- base64\_decode
  - base64.c, 699
  - base64.h, 561
- base64\_enc\_map
  - base64.c, 701
- base64\_encode
  - base64.c, 700
  - base64.h, 562
- base64\_self\_test
  - base64.c, 700
  - base64.h, 562
- base64\_test\_dec
  - base64.c, 701
- base64\_test\_enc
  - base64.c, 701
- benchmark.c
  - buf, 851
  - BUFSIZE, 850
  - main, 850
  - myrand, 851
- bignum.c
  - biH, 706
  - biL, 706
  - BITS\_TO\_LIMBS, 706
  - CHARS\_TO\_LIMBS, 706
  - ciL, 706
  - mpi\_add\_abs, 706
  - mpi\_add\_int, 706
  - mpi\_add\_mpi, 707
  - mpi\_cmp\_abs, 707
  - mpi\_cmp\_int, 707
  - mpi\_cmp\_mpi, 707
  - mpi\_copy, 708
  - mpi\_div\_int, 708
  - mpi\_div\_mpi, 708
  - mpi\_exp\_mod, 709
  - mpi\_free, 709
  - mpi\_gcd, 709
  - mpi\_gen\_prime, 709
  - mpi\_get\_digit, 710
  - mpi\_grow, 710
  - mpi\_init, 710
  - mpi\_inv\_mod, 711
  - mpi\_is\_prime, 711
  - mpi\_lsb, 711
  - mpi\_lset, 711
  - mpi\_mod\_int, 712

- mpi\_mod\_mpi, 712
- mpi\_montg\_init, 712
- mpi\_montmul, 712
- mpi\_montred, 712
- mpi\_msb, 713
- mpi\_mul\_hlp, 713
- mpi\_mul\_int, 713
- mpi\_mul\_mpi, 713
- mpi\_read\_binary, 713
- mpi\_read\_file, 714
- mpi\_read\_mystring, 714
- mpi\_read\_string, 714
- mpi\_self\_test, 715
- mpi\_shift\_l, 715
- mpi\_shift\_r, 715
- mpi\_size, 715
- mpi\_sub\_abs, 716
- mpi\_sub\_hlp, 716
- mpi\_sub\_int, 716
- mpi\_sub\_mpi, 716
- mpi\_swap, 716
- mpi\_write\_binary, 717
- mpi\_write\_file, 717
- mpi\_write\_hlp, 717
- mpi\_write\_string, 718
- small\_prime, 718
- bignum.h
  - mpi\_add\_abs, 567
  - mpi\_add\_int, 567
  - mpi\_add\_mpi, 567
  - MPI\_CHK, 566
  - mpi\_cmp\_abs, 567
  - mpi\_cmp\_int, 568
  - mpi\_cmp\_mpi, 568
  - mpi\_copy, 568
  - mpi\_div\_int, 568
  - mpi\_div\_mpi, 569
  - mpi\_exp\_mod, 569
  - mpi\_free, 569
  - mpi\_gcd, 570
  - mpi\_gen\_prime, 570
  - mpi\_grow, 570
  - mpi\_init, 571
  - mpi\_inv\_mod, 571
  - mpi\_is\_prime, 571
  - mpi\_lsb, 571
  - mpi\_lset, 572
  - mpi\_mod\_int, 572
  - mpi\_mod\_mpi, 572
  - mpi\_msb, 572
  - mpi\_mul\_int, 573
  - mpi\_mul\_mpi, 573
  - mpi\_read\_binary, 573
  - mpi\_read\_file, 574
  - mpi\_read\_mystring, 574
  - mpi\_read\_string, 574
  - mpi\_self\_test, 574
  - mpi\_shift\_l, 575
  - mpi\_shift\_r, 575
  - mpi\_size, 575
  - mpi\_sub\_abs, 575
  - mpi\_sub\_int, 576
  - mpi\_sub\_mpi, 576
  - mpi\_swap, 576
  - mpi\_write\_binary, 576
  - mpi\_write\_file, 577
  - mpi\_write\_string, 577
  - t\_dbl, 567
  - t\_int, 567
  - XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA, 566
  - XYSSL\_ERR\_MPI\_BUFFER\_TOO\_SMALL, 566
  - XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO, 566
  - XYSSL\_ERR\_MPI\_FILE\_IO\_ERROR, 566
  - XYSSL\_ERR\_MPI\_INVALID\_CHARACTER, 566
  - XYSSL\_ERR\_MPI\_NEGATIVE\_VALUE, 566
  - XYSSL\_ERR\_MPI\_NOT\_ACCEPTABLE, 566
- biH
  - bignum.c, 706
- biL
  - bignum.c, 706
- binary
  - agent\_s, 46
- BITS\_TO\_LIMBS
  - bignum.c, 706
- bluetooth
  - agency\_s, 36
  - mc\_platform\_s, 111
  - MCAgencyOptions\_s, 119
- bn\_mul.h
  - MULADDC\_CORE, 579
  - MULADDC\_INIT, 579
  - MULADDC\_STOP, 579
- boundary
  - mtp\_http\_s, 134
- buf
  - aes\_context, 35
  - benchmark.c, 851
- buffer
  - md2\_context, 121
  - md4\_context, 123
  - md5\_context, 125
  - mxmml\_fdbuf\_s, 140



- sha1\_context, [156](#)
- sha2\_context, [158](#)
- sha4\_context, [160](#)
- buffer\_size
  - options, [149](#)
- BUFLLEN
  - acc.c, [169](#)
- BUFSIZE
  - benchmark.c, [850](#)
- ca\_chain
  - \_ssl\_context, [11](#)
- ca\_istrue
  - \_x509\_cert, [25](#)
- cache\_control
  - mtp\_http\_s, [134](#)
- callback
  - fipa\_comm\_action\_t, [77](#)
- callbacks
  - mxml-entity.c, [493](#)
- CANCEL
  - message.h, [384](#)
- certs.c
  - test\_ca\_cert, [720](#)
  - test\_ca\_key, [720](#)
  - test\_ca\_pwd, [720](#)
  - test\_cli\_cert, [721](#)
  - test\_cli\_key, [721](#)
  - test\_srv\_cert, [722](#)
  - test\_srv\_key, [722](#)
  - xyssl\_ca\_cert, [723](#)
- certs.h
  - test\_ca\_cert, [581](#)
  - test\_ca\_key, [581](#)
  - test\_ca\_pwd, [581](#)
  - test\_cli\_cert, [581](#)
  - test\_cli\_key, [581](#)
  - test\_srv\_cert, [581](#)
  - test\_srv\_key, [582](#)
  - xyssl\_ca\_cert, [582](#)
- cfp\_request
  - fipa\_comm\_protocol\_cn\_t, [80](#)
- CH\_DATATYPE\_SIZE
  - macros.h, [366](#)
- CH\_DATATYPE\_STR\_TO\_VAL
  - macros.h, [366](#)
- CH\_DATATYPE\_STRING
  - macros.h, [367](#)
- CH\_DATATYPE\_VALUE\_STRING
  - macros.h, [367](#)
- ch\_options
  - MCAgencyOptions\_s, [119](#)
- CH\_STRING\_DATATYPE
  - macros.h, [368](#)
- CHARS\_TO\_LIMBS
  - bignum.c, [706](#)
- check\_list
  - fipa\_comm\_action\_t, [77](#)
- CHECK\_NULL
  - macros.h, [368](#)
- child
  - mxml\_node\_s, [143](#)
- ciL
  - bignum.c, [706](#)
- cipher
  - \_ssl\_session, [22](#)
- ciphers
  - \_ssl\_context, [11](#)
- cksum
  - md2\_context, [121](#)
- class.cxx
  - bar, [528](#)
  - f, [528](#)
  - foo, [528](#)
- client
  - agency\_s, [36](#)
- client\_auth
  - \_ssl\_context, [11](#)
- client\_fd
  - acc.h, [221](#)
- client\_thread\_arg
  - mc\_platform\_s, [111](#)
- clientfd
  - connection\_s, [58](#)
- cmd\_handler\_t
  - commands.h, [256](#)
- cmd\_handlers
  - commands.h, [257](#)
- cmd\_prompt
  - mc\_platform\_s, [112](#)
- cmd\_prompt.c
  - cmd\_prompt\_Destroy, [186](#)
  - cmd\_prompt\_Initialize, [186](#)
  - cmd\_prompt\_Start, [186](#)
  - cmd\_prompt\_Thread, [187](#)
  - dealloc\_command, [187](#)
  - exec\_command, [187](#)
  - handler\_COMPOSE\_SEND, [187](#)
  - handler\_FLUSH\_AGENTS, [187](#)
  - handler\_HELP, [187](#)
  - handler\_PRINT\_CONNECTLIST, [187](#)
  - handler\_PRINTLIST\_AGENTS, [188](#)
  - handler\_PRINTLIST\_MESSAGE, [188](#)
  - handler\_QUIT, [188](#)
  - handler\_SEND, [188](#)
  - process\_command, [188](#)
  - split\_string, [188](#)
- cmd\_prompt.h

- cmd\_prompt\_Destroy, 253
- cmd\_prompt\_Initialize, 253
- cmd\_prompt\_p, 252
- cmd\_prompt\_Start, 253
- cmd\_prompt\_t, 252
- cmd\_prompt\_Thread, 253
- CMDLINE\_SIZE, 252
- command\_p, 253
- command\_t, 253
- dealloc\_command, 253
- exec\_command, 253
- process\_command, 253
- split\_string, 253
- cmd\_prompt\_Destroy
  - cmd\_prompt.c, 186
  - cmd\_prompt.h, 253
- cmd\_prompt\_Initialize
  - cmd\_prompt.c, 186
  - cmd\_prompt.h, 253
- cmd\_prompt\_p
  - cmd\_prompt.h, 252
- cmd\_prompt\_s, 56
  - thread, 56
- cmd\_prompt\_Start
  - cmd\_prompt.c, 186
  - cmd\_prompt.h, 253
- cmd\_prompt\_t
  - cmd\_prompt.h, 252
- cmd\_prompt\_Thread
  - cmd\_prompt.c, 187
  - cmd\_prompt.h, 253
- CMDLINE\_SIZE
  - cmd\_prompt.h, 252
- code\_id
  - agent\_task\_s, 49
- COLLECT\_SIZE
  - havege.h, 596
- COMMAND
  - commands.h, 255–257
  - commands.x.h, 258
- command
  - options, 149
- COMMAND\_BOTH
  - ssl\_test.c, 854
- command\_cmds
  - commands.h, 257
- command\_descriptions
  - commands.h, 257
- command\_indices\_e
  - commands.h, 256
- command\_p
  - cmd\_prompt.h, 253
- COMMAND\_READ
  - ssl\_test.c, 854
- command\_s, 57
  - args, 57
  - index, 57
  - num\_args, 57
- command\_t
  - cmd\_prompt.h, 253
- COMMAND\_WRITE
  - ssl\_test.c, 854
- commands.h
  - cmd\_handler\_t, 256
  - cmd\_handlers, 257
  - COMMAND, 255–257
  - command\_cmds, 257
  - command\_descriptions, 257
  - command\_indices\_e, 256
- commands.x.h
  - COMMAND, 258
  - compose\_send, 259
  - help, 259
  - is, 259
  - print\_connectlist, 259
  - printlist\_agents, 259
  - send, 259
- comments
  - fipa\_acl\_Param\_s, 73
- COMPOSE\_BLOCKSIZE
  - dynstring.h, 274
- compose\_send
  - commands.x.h, 259
- cond
  - barrier\_node\_s, 54
  - df.h, 270
  - mc\_rwlock\_s, 117
  - syncListNode\_s, 164
- COND\_BROADCAST
  - macros.h, 369
- COND\_DESTROY
  - macros.h, 369
- COND\_INIT
  - macros.h, 369
- COND\_RESET
  - macros.h, 369
- COND\_SIGNAL
  - macros.h, 370
- COND\_SLEEP
  - macros.h, 370
- COND\_SLEEP\_ACTION
  - macros.h, 370
- COND\_T
  - macros.h, 370
- COND\_WAIT
  - macros.h, 370
- CONFLICT
  - mtp\_http.h, 390

- CONN\_THREADS
  - acc.c, 169
- conn\_timeout
  - options, 149
- connect\_id
  - agent\_s, 46
  - connection\_s, 58
  - message\_s, 127
- CONNECT\_THREAD\_EXIT
  - acc.c, 169
- connection
  - mtp\_http\_s, 134
- connection.c
  - connection\_Copy, 189
  - connection\_Destroy, 189
  - connection\_New, 189
- connection.h
  - connection\_Close, 260
  - connection\_Copy, 260
  - connection\_CreateMessage, 260
  - connection\_Destroy, 261
  - connection\_New, 261
  - connection\_p, 260
  - connection\_RecvFromConnection, 261
  - connection\_SendtoConnection, 261
  - connection\_SetHosts, 261
  - connection\_t, 260
- connection\_Close
  - connection.h, 260
- connection\_Copy
  - connection.c, 189
  - connection.h, 260
- connection\_CreateMessage
  - connection.h, 260
- connection\_Destroy
  - connection.c, 189
  - connection.h, 261
- connection\_New
  - connection.c, 189
  - connection.h, 261
- connection\_p
  - connection.h, 260
- connection\_queue
  - mc\_platform\_s, 112
- connection\_RecvFromConnection
  - connection.h, 261
- connection\_s, 58
  - addr, 58
  - AES\_key, 58
  - clientfd, 58
  - connect\_id, 58
  - nonce, 58
  - remote\_hostname, 58
  - serverfd, 59
- connection\_SendtoConnection
  - connection.h, 261
- connection\_SetHosts
  - connection.h, 261
- connection\_t
  - connection.h, 260
- content
  - fipa\_acl\_message\_s, 70
  - fipa\_comm\_reply\_t, 84
  - fipa\_expression\_s, 89
  - fipa\_string\_s, 93
  - fipa\_word\_s, 96
  - mtp\_http\_s, 135
- content\_length
  - mtp\_http\_s, 135
- content\_type
  - mtp\_http\_content\_s, 133
  - mtp\_http\_s, 135
- CONTINUE
  - mtp\_http.h, 390
- contract\_func
  - fipa\_comm\_protocol\_cn\_t, 80
- conversation\_id
  - fipa\_acl\_message\_s, 70
  - fipa\_comm\_protocol\_t, 82
- CREATED
  - mtp\_http.h, 390
- ctx\_dec
  - \_ssl\_context, 12
- ctx\_enc
  - \_ssl\_context, 12
- cur
  - ssl\_server.c, 848
- cur\_node
  - mxml\_index\_s, 141
- current
  - mxml\_fdbuf\_s, 140
- custom
  - mxml\_value\_u, 147
- D
  - rsa\_context, 152
- d2i\_RSA\_PUBKEY
  - openssl.h, 619
- d2i\_RSAPrivateKey
  - openssl.h, 617
- DATA
  - list.h, 464
- data
  - \_x509\_node, 30
  - fipa\_list\_t, 90
  - interpreter\_variable\_data\_s, 106
  - mtp\_http\_content\_s, 133
  - mxml\_custom\_s, 139

- data\_structures.c
  - agent\_queue\_Flush, 190
  - AP\_GENERIC\_Copy, 190
  - AP\_GENERIC\_Destroy, 190
  - AP\_QUEUE\_SEARCH\_TEMPLATE, 190, 191
  - AP\_QUEUE\_STD\_DEFN\_TEMPLATE, 191
- data\_structures.h
  - agent\_mailbox\_p, 264
  - agent\_queue\_Flush, 263
  - AP\_GENERIC\_Destroy, 263
  - AP\_QUEUE\_DECL\_TEMPLATE, 263
  - AP\_QUEUE\_GENERIC\_DECL\_TEMPLATE, 263, 264
  - int, 264
  - Remove, 264
  - RemoveID, 264
  - RemoveName, 265
  - Search, 265
  - SearchName, 265
- data\_type
  - interpreter\_variable\_data\_s, 106
- datastate
  - agent\_s, 46
- date
  - fipa\_acl\_Param\_s, 73
  - mtp\_http\_s, 135
- datetime
  - fipa\_expression\_s::content\_u, 60
- day
  - \_x509\_time, 33
  - fipa\_DateTime\_s, 87
- dealloc\_command
  - cmd\_prompt.c, 187
  - cmd\_prompt.h, 253
- debug
  - fipa\_comm\_t, 85
- debug.c
  - debug\_fmt, 724
  - debug\_print\_buf, 724
  - debug\_print\_crt, 724
  - debug\_print\_mpi, 724
  - debug\_print\_msg, 724
  - debug\_print\_ret, 725
- debug.h
  - debug\_fmt, 584
  - debug\_print\_buf, 584
  - debug\_print\_crt, 585
  - debug\_print\_mpi, 585
  - debug\_print\_msg, 585
  - debug\_print\_ret, 585
  - SSL\_DEBUG\_BUF, 583
  - SSL\_DEBUG\_CRT, 583
  - SSL\_DEBUG\_MPI, 584
  - SSL\_DEBUG\_MSG, 584
  - SSL\_DEBUG\_RET, 584
- debug\_fmt
  - debug.c, 724
  - debug.h, 584
- DEBUG\_LEVEL
  - ssl\_client1.c, 843
  - ssl\_client2.c, 845
  - ssl\_server.c, 847
- debug\_level
  - options, 149
- debug\_print\_buf
  - debug.c, 724
  - debug.h, 584
- debug\_print\_crt
  - debug.c, 724
  - debug.h, 585
- debug\_print\_mpi
  - debug.c, 724
  - debug.h, 585
- debug\_print\_msg
  - debug.c, 724
  - debug.h, 585
- debug\_print\_ret
  - debug.c, 725
  - debug.h, 585
- default\_agent\_status
  - MCAgencyOptions\_s, 119
- default\_agentstatus
  - agency\_s, 37
  - mc\_platform\_s, 112
- default\_callback
  - mxml-entity.c, 492
- DEFAULT\_HOSTNAME\_LENGTH
  - mc\_platform.c, 467
- default\_reply
  - fipa\_comm\_performative\_t, 79
  - fipa\_comm\_t, 85
- des.c
  - des3\_crypt\_cbc, 729
  - des3\_crypt\_ecb, 729
  - des3\_set2key, 729
  - des3\_set2key\_dec, 730
  - des3\_set2key\_enc, 730
  - des3\_set3key, 730
  - des3\_set3key\_dec, 730
  - des3\_set3key\_enc, 731
  - des3\_test\_buf, 733
  - des3\_test\_cbc\_dec, 733
  - des3\_test\_cbc\_enc, 733
  - des3\_test\_ecb\_dec, 733
  - des3\_test\_ecb\_enc, 733
  - des3\_test\_iv, 734
  - des3\_test\_keys, 734

- des\_crypt\_cbc, [731](#)
- des\_crypt\_ecb, [731](#)
- DES\_FP, [727](#)
- DES\_IP, [727](#)
- DES\_ROUND, [728](#)
- des\_self\_test, [732](#)
- des\_setkey, [732](#)
- des\_setkey\_dec, [732](#)
- des\_setkey\_enc, [732](#)
- GET\_ULONG\_BE, [728](#)
- LHs, [734](#)
- PUT\_ULONG\_BE, [728](#)
- RHs, [734](#)
- SB1, [735](#)
- SB2, [735](#)
- SB3, [736](#)
- SB4, [736](#)
- SB5, [736](#)
- SB6, [737](#)
- SB7, [737](#)
- SB8, [738](#)
- SWAP, [729](#)
- des.h
  - des3\_crypt\_cbc, [587](#)
  - des3\_crypt\_ecb, [587](#)
  - des3\_set2key\_dec, [588](#)
  - des3\_set2key\_enc, [588](#)
  - des3\_set3key\_dec, [588](#)
  - des3\_set3key\_enc, [588](#)
  - des\_crypt\_cbc, [589](#)
  - des\_crypt\_ecb, [589](#)
  - DES\_DECRYPT, [587](#)
  - DES\_ENCRYPT, [587](#)
  - des\_self\_test, [589](#)
  - des\_setkey\_dec, [590](#)
  - des\_setkey\_enc, [590](#)
- des3\_context, [62](#)
  - mode, [62](#)
  - sk, [62](#)
- des3\_crypt\_cbc
  - des.c, [729](#)
  - des.h, [587](#)
- des3\_crypt\_ecb
  - des.c, [729](#)
  - des.h, [587](#)
- des3\_set2key
  - des.c, [729](#)
- des3\_set2key\_dec
  - des.c, [730](#)
  - des.h, [588](#)
- des3\_set2key\_enc
  - des.c, [730](#)
  - des.h, [588](#)
- des3\_set3key
  - des.c, [730](#)
- des3\_set3key\_dec
  - des.c, [730](#)
  - des.h, [588](#)
- des3\_set3key\_enc
  - des.c, [731](#)
  - des.h, [588](#)
- des3\_test\_buf
  - des.c, [733](#)
- des3\_test\_cbc\_dec
  - des.c, [733](#)
- des3\_test\_cbc\_enc
  - des.c, [733](#)
- des3\_test\_ecb\_dec
  - des.c, [733](#)
- des3\_test\_ecb\_enc
  - des.c, [733](#)
- des3\_test\_iv
  - des.c, [734](#)
- des3\_test\_keys
  - des.c, [734](#)
- des\_context, [63](#)
  - mode, [63](#)
  - sk, [63](#)
- des\_crypt\_cbc
  - des.c, [731](#)
  - des.h, [589](#)
- des\_crypt\_ecb
  - des.c, [731](#)
  - des.h, [589](#)
- DES\_DECRYPT
  - des.h, [587](#)
- DES\_ENCRYPT
  - des.h, [587](#)
- DES\_FP
  - des.c, [727](#)
- DES\_IP
  - des.c, [727](#)
- DES\_ROUND
  - des.c, [728](#)
- des\_self\_test
  - des.c, [732](#)
  - des.h, [589](#)
- des\_setkey
  - des.c, [732](#)
- des\_setkey\_dec
  - des.c, [732](#)
  - des.h, [590](#)
- des\_setkey\_enc
  - des.c, [732](#)
  - des.h, [590](#)
- destroy
  - mxml\_custom\_s, [139](#)
- df

- mc\_platform\_s, 112
- df.c
  - df\_Add, 193
  - df\_AddRequest, 193
  - df\_Destroy, 193
  - df\_Initialize, 193
  - df\_node\_Destroy, 193
  - df\_ProcessRequest, 193
  - df\_request\_list\_Destroy, 194
  - df\_request\_list\_New, 194
  - df\_request\_list\_node\_Destroy, 194
  - df\_request\_list\_node\_New, 194
  - df\_request\_list\_Pop, 194
  - df\_request\_search\_Destroy, 194
  - df\_request\_search\_New, 194
  - df\_SearchForService, 195
  - df\_Start, 195
  - df\_Thread, 195
  - REQUEST, 192
  - request\_handler\_DEREGISTER, 195
  - request\_handler\_REGISTER, 195
  - request\_handler\_SEARCH, 195
  - request\_handler\_SUBSCRIBE, 195
- df.h
  - cond, 270
  - df\_AddRequest, 268
  - df\_Destroy, 268
  - df\_Initialize, 268
  - df\_node\_Destroy, 268
  - df\_ProcessRequest, 268
  - df\_request\_list\_Destroy, 268
  - df\_request\_list\_index\_e, 267
  - df\_request\_list\_New, 268
  - df\_request\_list\_node\_Destroy, 268
  - df\_request\_list\_node\_New, 269
  - df\_request\_list\_Pop, 269
  - df\_request\_search\_Destroy, 269
  - df\_request\_search\_New, 269
  - df\_SearchForService, 269
  - df\_Start, 269
  - df\_Thread, 269
  - HERE, 267
  - INSERT, 267
  - lock, 270
  - mc\_platform, 270
  - MISC, 267
  - node, 270
  - num\_entries, 271
  - REQUEST, 267, 270
  - request\_list, 271
  - search\_results, 271
  - search\_string, 271
  - service\_list, 271
  - service\_types\_e, 267
  - size, 271
  - SOME, 267
  - STRUCT, 270
  - thread, 271
  - TYPES, 267
  - waiting, 271
  - waiting\_cond, 272
  - waiting\_lock, 272
  - ZERO, 267
  - df\_Add
    - df.c, 193
  - df\_AddRequest
    - df.c, 193
    - df.h, 268
  - df\_Destroy
    - df.c, 193
    - df.h, 268
  - df\_Initialize
    - df.c, 193
    - df.h, 268
  - df\_node\_Destroy
    - df.c, 193
    - df.h, 268
  - df\_ProcessRequest
    - df.c, 193
    - df.h, 268
  - df\_request.x.h
    - register, 273
    - REQUEST, 273
    - search, 273
  - df\_request\_list\_Destroy
    - df.c, 194
    - df.h, 268
  - df\_request\_list\_index\_e
    - df.h, 267
  - df\_request\_list\_New
    - df.c, 194
    - df.h, 268
  - df\_request\_list\_node\_Destroy
    - df.c, 194
    - df.h, 268
  - df\_request\_list\_node\_New
    - df.c, 194
    - df.h, 269
  - df\_request\_list\_Pop
    - df.c, 194
    - df.h, 269
  - df\_request\_search\_Destroy
    - df.c, 194
    - df.h, 269
  - df\_request\_search\_New
    - df.c, 194
    - df.h, 269
  - df\_SearchForService

- df.c, [195](#)
- df.h, [269](#)
- df\_Start
  - df.c, [195](#)
  - df.h, [269](#)
- df\_Thread
  - df.c, [195](#)
  - df.h, [269](#)
- DFL\_BUFFER\_SIZE
  - ssl\_test.c, [854](#)
- DFL\_COMMAND
  - ssl\_test.c, [854](#)
- DFL\_CONN\_TIMEOUT
  - ssl\_test.c, [854](#)
- DFL\_DEBUG\_LEVEL
  - ssl\_test.c, [854](#)
- DFL\_FORCE\_CIPHER
  - ssl\_test.c, [854](#)
- DFL\_IOMODE
  - ssl\_test.c, [855](#)
- DFL\_MAX\_BYTES
  - ssl\_test.c, [855](#)
- DFL\_MAX\_CONNECTIONS
  - ssl\_test.c, [855](#)
- DFL\_OPMODE
  - ssl\_test.c, [855](#)
- DFL\_SERVER\_NAME
  - ssl\_test.c, [855](#)
- DFL\_SERVER\_PORT
  - ssl\_test.c, [855](#)
- DFL\_SESSION\_LIFETIME
  - ssl\_test.c, [855](#)
- DFL\_SESSION\_REUSE
  - ssl\_test.c, [855](#)
- dh\_client.c
  - main, [836](#)
  - SERVER\_NAME, [836](#)
  - SERVER\_PORT, [836](#)
- dh\_genprime.c
  - DH\_P\_SIZE, [837](#)
  - GENERATOR, [837](#)
  - main, [837](#)
- DH\_P\_SIZE
  - dh\_genprime.c, [837](#)
- dh\_server.c
  - main, [838](#)
  - PLAINTEXT, [838](#)
  - SERVER\_PORT, [838](#)
- dhm.c
  - dhm\_calc\_secret, [740](#)
  - dhm\_free, [740](#)
  - dhm\_make\_params, [740](#)
  - dhm\_make\_public, [741](#)
  - DHM\_MPI\_EXPORT, [739](#)
  - dhm\_read\_bignum, [741](#)
  - dhm\_read\_params, [741](#)
  - dhm\_read\_public, [742](#)
  - dhm\_self\_test, [742](#)
- dhm.h
  - dhm\_calc\_secret, [592](#)
  - dhm\_free, [593](#)
  - dhm\_make\_params, [593](#)
  - dhm\_make\_public, [593](#)
  - dhm\_read\_params, [594](#)
  - dhm\_read\_public, [594](#)
  - dhm\_self\_test, [595](#)
  - XYSSL\_ERR\_DHM\_BAD\_INPUT\_DATA, [592](#)
  - XYSSL\_ERR\_DHM\_CALC\_SECRET\_FAILED, [592](#)
  - XYSSL\_ERR\_DHM\_MAKE\_PARAMS\_FAILED, [592](#)
  - XYSSL\_ERR\_DHM\_MAKE\_PUBLIC\_FAILED, [592](#)
  - XYSSL\_ERR\_DHM\_READ\_PARAMS\_FAILED, [592](#)
  - XYSSL\_ERR\_DHM\_READ\_PUBLIC\_FAILED, [592](#)
- dhm\_calc\_secret
  - dhm.c, [740](#)
  - dhm.h, [592](#)
- dhm\_context, [64](#)
  - G, [64](#)
  - GX, [64](#)
  - GY, [64](#)
  - K, [64](#)
  - len, [65](#)
  - P, [65](#)
  - RP, [65](#)
  - X, [65](#)
- dhm\_ctx
  - \_ssl\_context, [12](#)
- dhm\_free
  - dhm.c, [740](#)
  - dhm.h, [593](#)
- dhm\_G
  - ssl\_test.c, [857](#)
- dhm\_make\_params
  - dhm.c, [740](#)
  - dhm.h, [593](#)
- dhm\_make\_public
  - dhm.c, [741](#)
  - dhm.h, [593](#)
- DHM\_MPI\_EXPORT
  - dhm.c, [739](#)
- dhm\_P
  - ssl\_test.c, [858](#)
- dhm\_read\_bignum

- dhm.c, 741
- dhm\_read\_params
  - dhm.c, 741
  - dhm.h, 594
- dhm\_read\_public
  - dhm.c, 742
  - dhm.h, 594
- dhm\_self\_test
  - dhm.c, 742
  - dhm.h, 595
- do\_crypt
  - \_ssl\_context, 12
- DP
  - rsa\_context, 152
- DQ
  - rsa\_context, 152
- dynstring.c
  - dynstring\_Append, 197
  - dynstring\_Destroy, 197
  - dynstring\_New, 197
- dynstring.h
  - COMPOSE\_BLOCKSIZE, 274
  - dynstring\_Append, 274
  - dynstring\_Destroy, 275
  - dynstring\_New, 275
  - dynstring\_p, 274
  - dynstring\_t, 274
- dynstring\_Append
  - dynstring.c, 197
  - dynstring.h, 274
- dynstring\_Destroy
  - dynstring.c, 197
  - dynstring.h, 275
- dynstring\_New
  - dynstring.c, 197
  - dynstring.h, 275
- dynstring\_p
  - dynstring.h, 274
- dynstring\_s, 66
  - len, 66
  - message, 66
  - size, 66
- dynstring\_t
  - dynstring.h, 274
- E
  - rsa\_context, 153
- element
  - mxml\_value\_u, 147
- enable\_security
  - agency\_s, 37
  - MCAgencyOptions\_s, 119
- ENCODE\_UTF16BE
  - mxml-file.c, 495
- ENCODE\_UTF16LE
  - mxml-file.c, 495
- ENCODE\_UTF8
  - mxml-file.c, 495
- encoding
  - fipa\_acl\_message\_s, 70
- ENCRYPTED\_DATA
  - message.h, 384
- ENCRYPTION\_INITIALIZE
  - message.h, 384
- end
  - \_x509\_node, 30
  - mxml\_fdbuf\_s, 140
- endpoint
  - \_ssl\_context, 12
- enum.cxx
  - FOO\_BLUE, 529
  - FOO\_ONE, 529
  - FOO\_RED, 529
  - FOO\_TWO, 529
  - foo\_enum\_e, 529
  - foo\_enum\_t, 529
- ERR
  - libmc.h, 322
  - mc\_error.h, 377
- err
  - mc\_platform\_s, 112
- ERR\_get\_error
  - openssl.h, 617
- error
  - fipa\_comm\_protocol\_t, 82
  - fipa\_comm\_t, 85
- error\_code\_e
  - libmc.h, 322
  - mc\_error.h, 377
- error\_code\_t
  - libmc.h, 321
  - mc\_error.h, 377
- exec\_command
  - cmd\_prompt.c, 187
  - cmd\_prompt.h, 253
- EXPECTATION\_FAILED
  - mtp\_http.h, 391
- EXPONENT
  - interface.h, 549
  - rsa\_genkey.c, 840
- EXPORTMC
  - libmc.h, 320
- expression
  - fipa\_expression\_s::content\_u, 60
- extract\_nonce\_from\_MA
  - interface.c, 544
- F



- md5.c, 748
- sha1.c, 772
- f
  - class.cxx, 528
- F0
  - sha2.c, 781
  - sha4.c, 790
- F1
  - sha2.c, 781
  - sha4.c, 790
- f\_dbg
  - \_ssl\_context, 12
- f\_recv
  - \_ssl\_context, 13
- f\_rng
  - \_ssl\_context, 13
  - rsa\_context, 153
- f\_send
  - \_ssl\_context, 13
- FAILED\_DEPENDANCY
  - mtp\_http.h, 391
- false
  - fipa\_comm.h, 303
- fd
  - mxmml\_fdbuf\_s, 140
- filename
  - agency\_s, 37
- fin\_md5
  - \_ssl\_context, 13
- fin\_shal
  - \_ssl\_context, 13
- FIPA\_ACCEPT\_PROPOSAL
  - fipa\_acl.h, 280, 281
  - libmc.h, 322, 323
- FIPA\_ACL
  - message.h, 384
- fipa\_acl.h
  - FIPA\_ACCEPT\_PROPOSAL, 280, 281
  - FIPA\_AGREE, 280, 281
  - FIPA\_CALL\_FOR\_PROPOSAL, 280, 281
  - FIPA\_CANCEL, 280, 281
  - FIPA\_CONFIRM, 280, 281
  - FIPA\_DISCONFIRM, 280, 281
  - FIPA\_ERROR, 280, 281
  - FIPA\_EXPR\_DATETIME, 280
  - FIPA\_EXPR\_EXPRESSION, 280
  - FIPA\_EXPR\_NUMBER, 280
  - FIPA\_EXPR\_STRING, 280
  - FIPA\_EXPR\_WORD, 280
  - FIPA\_EXPR\_ZERO, 280
  - FIPA\_FAILURE, 280, 281
  - FIPA\_INFORM, 281
  - FIPA\_INFORM\_IF, 281
  - FIPA\_INFORM\_REF, 281
  - FIPA\_NOT\_UNDERSTOOD, 281
  - FIPA\_PROPOGATE, 281
  - FIPA\_PROPOSE, 281
  - FIPA\_PROTOCOL\_BROKERING, 282
  - FIPA\_PROTOCOL\_CONTRACT\_NET, 282
  - FIPA\_PROTOCOL\_DUTCH\_AUCTION, 282
  - FIPA\_PROTOCOL\_END, 282
  - FIPA\_PROTOCOL\_ENGLISH\_AUCTION, 282
  - FIPA\_PROTOCOL\_ERROR, 282
  - FIPA\_PROTOCOL\_ITERATED\_CONTRACT\_NET, 282
  - FIPA\_PROTOCOL\_NONE, 282
  - FIPA\_PROTOCOL\_PROPOSE, 282
  - FIPA\_PROTOCOL\_QUERY, 282
  - FIPA\_PROTOCOL\_RECRUITING, 282
  - FIPA\_PROTOCOL\_REQUEST, 282
  - FIPA\_PROTOCOL\_REQUEST\_WHEN, 282
  - FIPA\_PROTOCOL\_SUBSCRIBE, 282
  - FIPA\_PROXY, 281
  - FIPA\_QUERY\_IF, 281
  - FIPA\_QUERY\_REF, 281
  - FIPA\_REFUSE, 281
  - FIPA\_REJECT\_PROPOSAL, 281
  - FIPA\_REQUEST, 281
  - FIPA\_REQUEST\_WHEN, 281
  - FIPA\_REQUEST\_WHENEVER, 281
  - FIPA\_SUBSCRIBE, 281
  - FIPA\_ZERO, 280, 281
- FIPA\_AGREE
  - fipa\_acl.h, 280, 281
  - libmc.h, 322, 323
- FIPA\_CALL\_FOR\_PROPOSAL
  - fipa\_acl.h, 280, 281
  - libmc.h, 322, 323
- FIPA\_CANCEL
  - fipa\_acl.h, 280, 281
  - libmc.h, 322, 323
- fipa\_comm.h
  - FIPA\_COMM\_ACTION, 306
  - FIPA\_COMM\_ACTION\_BOTH, 305
  - FIPA\_COMM\_ACTION\_CALLBACK, 305
  - FIPA\_COMM\_ACTION\_ERROR, 305
  - FIPA\_COMM\_ACTION\_NONE, 305
  - FIPA\_COMM\_ACTION\_REPLY, 305
  - FIPA\_COMM\_CALLBACK, 306
  - FIPA\_COMM\_CHECK, 306
  - FIPA\_COMM\_CHECK\_CONTENT, 305
  - FIPA\_COMM\_CHECK\_CONVERSATIONID, 305
  - FIPA\_COMM\_CHECK\_ENCODING, 305
  - FIPA\_COMM\_CHECK\_ERROR, 305
  - FIPA\_COMM\_CHECK\_LANGUAGE, 305
  - FIPA\_COMM\_CHECK\_ONTOLOGY, 305

- FIPA\_COMM\_CHECK\_SENDER\_-  
ADDRESS, 305
- FIPA\_COMM\_CHECK\_SENDER\_NAME,  
305
- FIPA\_COMM\_END, 306
- FIPA\_COMM\_LOGIC\_AND, 305
- FIPA\_COMM\_LOGIC\_ERROR, 305
- FIPA\_COMM\_LOGIC\_OR, 305
- FIPA\_COMM\_PROTOCOL\_CN\_BID, 306
- FIPA\_COMM\_PROTOCOL\_CN\_CFP, 306
- FIPA\_COMM\_PROTOCOL\_CN\_DONE, 306
- FIPA\_COMM\_PROTOCOL\_CN\_ERROR,  
306
- FIPA\_COMM\_PROTOCOL\_CN\_RESULT,  
306
- FIPA\_COMM\_PROTOCOL\_CN\_START,  
306
- FIPA\_COMM\_REPLY, 306
- FIPA\_COMM\_ACTION  
fipa\_comm.h, 306
- FIPA\_COMM\_ACTION\_BOTH  
fipa\_comm.h, 305
- FIPA\_COMM\_ACTION\_CALLBACK  
fipa\_comm.h, 305
- FIPA\_COMM\_ACTION\_ERROR  
fipa\_comm.h, 305
- FIPA\_COMM\_ACTION\_NONE  
fipa\_comm.h, 305
- FIPA\_COMM\_ACTION\_REPLY  
fipa\_comm.h, 305
- FIPA\_COMM\_CALLBACK  
fipa\_comm.h, 306
- FIPA\_COMM\_CHECK  
fipa\_comm.h, 306
- FIPA\_COMM\_CHECK\_CONTENT  
fipa\_comm.h, 305
- FIPA\_COMM\_CHECK\_CONVERSATIONID  
fipa\_comm.h, 305
- FIPA\_COMM\_CHECK\_ENCODING  
fipa\_comm.h, 305
- FIPA\_COMM\_CHECK\_ERROR  
fipa\_comm.h, 305
- FIPA\_COMM\_CHECK\_LANGUAGE  
fipa\_comm.h, 305
- FIPA\_COMM\_CHECK\_ONTOLOGY  
fipa\_comm.h, 305
- FIPA\_COMM\_CHECK\_SENDER\_ADDRESS  
fipa\_comm.h, 305
- FIPA\_COMM\_CHECK\_SENDER\_NAME  
fipa\_comm.h, 305
- FIPA\_COMM\_END  
fipa\_comm.h, 306
- FIPA\_COMM\_LOGIC\_AND  
fipa\_comm.h, 305
- FIPA\_COMM\_LOGIC\_ERROR  
fipa\_comm.h, 305
- FIPA\_COMM\_LOGIC\_OR  
fipa\_comm.h, 305
- FIPA\_COMM\_PROTOCOL\_CN\_BID  
fipa\_comm.h, 306
- FIPA\_COMM\_PROTOCOL\_CN\_CFP  
fipa\_comm.h, 306
- FIPA\_COMM\_PROTOCOL\_CN\_DONE  
fipa\_comm.h, 306
- FIPA\_COMM\_PROTOCOL\_CN\_ERROR  
fipa\_comm.h, 306
- FIPA\_COMM\_PROTOCOL\_CN\_RESULT  
fipa\_comm.h, 306
- FIPA\_COMM\_PROTOCOL\_CN\_START  
fipa\_comm.h, 306
- FIPA\_COMM\_REPLY  
fipa\_comm.h, 306
- FIPA\_CONFIRM  
fipa\_acl.h, 280, 281  
libmc.h, 322, 323
- FIPA\_DISCONFIRM  
fipa\_acl.h, 280, 281  
libmc.h, 322, 323
- FIPA\_ERROR  
fipa\_acl.h, 280, 281  
libmc.h, 322, 323
- FIPA\_EXPR\_DATETIME  
fipa\_acl.h, 280
- FIPA\_EXPR\_EXPRESSION  
fipa\_acl.h, 280
- FIPA\_EXPR\_NUMBER  
fipa\_acl.h, 280
- FIPA\_EXPR\_STRING  
fipa\_acl.h, 280
- FIPA\_EXPR\_WORD  
fipa\_acl.h, 280
- FIPA\_EXPR\_ZERO  
fipa\_acl.h, 280
- FIPA\_FAILURE  
fipa\_acl.h, 280, 281  
libmc.h, 322, 323
- FIPA\_INFORM  
fipa\_acl.h, 281  
libmc.h, 323
- FIPA\_INFORM\_IF  
fipa\_acl.h, 281  
libmc.h, 323
- FIPA\_INFORM\_REF  
fipa\_acl.h, 281  
libmc.h, 323
- FIPA\_NOT\_UNDERSTOOD  
fipa\_acl.h, 281  
libmc.h, 323

- FIPA\_PROPOGATE
  - [fipa\\_acl.h](#), 281
  - [libmc.h](#), 323
- FIPA\_PROPOSE
  - [fipa\\_acl.h](#), 281
  - [libmc.h](#), 323
- FIPA\_PROTOCOL\_BROKERING
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_CONTRACT\_NET
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_DUTCH\_AUCTION
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_END
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_ENGLISH\_AUCTION
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_ERROR
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_ITERATED\_CONTRACT\_-\_NET
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_NONE
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_PROPOSE
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_QUERY
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_RECRUITING
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_REQUEST
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_REQUEST\_WHEN
  - [fipa\\_acl.h](#), 282
- FIPA\_PROTOCOL\_SUBSCRIBE
  - [fipa\\_acl.h](#), 282
- FIPA\_PROXY
  - [fipa\\_acl.h](#), 281
  - [libmc.h](#), 323
- FIPA\_QUERY\_IF
  - [fipa\\_acl.h](#), 281
  - [libmc.h](#), 323
- FIPA\_QUERY\_REF
  - [fipa\\_acl.h](#), 281
  - [libmc.h](#), 323
- FIPA\_REFUSE
  - [fipa\\_acl.h](#), 281
  - [libmc.h](#), 323
- FIPA\_REJECT\_PROPOSAL
  - [fipa\\_acl.h](#), 281
  - [libmc.h](#), 323
- FIPA\_REQUEST
  - [fipa\\_acl.h](#), 281
  - [libmc.h](#), 323
- FIPA\_REQUEST\_WHEN
  - [fipa\\_acl.h](#), 281
  - [libmc.h](#), 323
- FIPA\_REQUEST\_WHENEVER
  - [fipa\\_acl.h](#), 281
  - [libmc.h](#), 323
- FIPA\_SUBSCRIBE
  - [fipa\\_acl.h](#), 281
  - [libmc.h](#), 323
- FIPA\_ZERO
  - [fipa\\_acl.h](#), 280, 281
  - [libmc.h](#), 322, 323
- [fipa\\_acl.c](#)
  - [fipa\\_acl\\_Compose](#), 200
  - [fipa\\_acl\\_message\\_Copy](#), 200
  - [fipa\\_acl\\_message\\_Destroy](#), 200
  - [fipa\\_acl\\_message\\_New](#), 200
  - [fipa\\_acl\\_Parse](#), 200
  - [fipa\\_agent\\_identifier\\_Compose](#), 201
  - [fipa\\_agent\\_identifier\\_Copy](#), 201
  - [fipa\\_agent\\_identifier\\_Destroy](#), 201
  - [fipa\\_agent\\_identifier\\_New](#), 201
  - [fipa\\_agent\\_identifier\\_Parse](#), 201
  - [fipa\\_agent\\_identifier\\_set\\_Compose](#), 201
  - [fipa\\_agent\\_identifier\\_set\\_Copy](#), 202
  - [fipa\\_agent\\_identifier\\_set\\_Destroy](#), 202
  - [fipa\\_agent\\_identifier\\_set\\_New](#), 202
  - [fipa\\_agent\\_identifier\\_set\\_Parse](#), 202
  - [fipa\\_CheckNextToken](#), 202
  - [fipa\\_DateTime\\_Compose](#), 202
  - [fipa\\_DateTime\\_Copy](#), 203
  - [fipa\\_DateTime\\_Destroy](#), 203
  - [fipa\\_DateTime\\_New](#), 203
  - [fipa\\_datetime\\_Parse](#), 203
  - [fipa\\_expression\\_Compose](#), 203
  - [fipa\\_expression\\_Copy](#), 203
  - [fipa\\_expression\\_Destroy](#), 204
  - [fipa\\_expression\\_New](#), 204
  - [fipa\\_expression\\_Parse](#), 204
  - [fipa\\_GetAtom](#), 204
  - [fipa\\_GetNextWord](#), 204
  - [fipa\\_GetWholeToken](#), 205
  - [fipa\\_message\\_parameter\\_Parse](#), 205
  - [fipa\\_message\\_string\\_Copy](#), 205
  - [fipa\\_message\\_string\\_Destroy](#), 205
  - [fipa\\_message\\_string\\_New](#), 205
  - [fipa\\_message\\_type\\_Parse](#), 205
  - [fipa\\_number\\_Compose](#), 206
  - [fipa\\_number\\_Copy](#), 206
  - [fipa\\_number\\_Destroy](#), 206
  - [fipa\\_number\\_New](#), 206
  - [fipa\\_performative\\_Compose](#), 206
  - [fipa\\_protocol\\_Compose](#), 206
  - [fipa\\_protocol\\_type\\_Parse](#), 207
  - [fipa\\_Reply](#), 207

- fipa\_string\_Compose, 207
- fipa\_string\_Copy, 207
- fipa\_string\_Destroy, 207
- fipa\_string\_New, 208
- fipa\_string\_Parse, 208
- fipa\_url\_Compose, 208
- fipa\_url\_Copy, 208
- fipa\_url\_Destroy, 208
- fipa\_url\_New, 208
- fipa\_url\_Parse, 208
- fipa\_url\_sequence\_Compose, 208
- fipa\_url\_sequence\_Copy, 209
- fipa\_url\_sequence\_Destroy, 209
- fipa\_url\_sequence\_New, 209
- fipa\_url\_sequence\_Parse, 209
- fipa\_word\_Compose, 209
- fipa\_word\_Copy, 209
- fipa\_word\_Destroy, 210
- fipa\_word\_New, 210
- fipa\_word\_Parse, 210
- FREEEMEM, 199
- fipa\_acl.h
  - fipa\_acl\_Compose, 282
  - fipa\_acl\_message\_Copy, 282
  - fipa\_acl\_message\_Destroy, 282
  - fipa\_acl\_message\_New, 283
  - fipa\_acl\_message\_p, 279
  - fipa\_acl\_message\_t, 279
  - fipa\_acl\_Parse, 283
  - fipa\_agent\_identifier\_Compose, 283
  - fipa\_agent\_identifier\_Copy, 283
  - fipa\_agent\_identifier\_Destroy, 283
  - fipa\_agent\_identifier\_New, 283
  - fipa\_agent\_identifier\_p, 279
  - fipa\_agent\_identifier\_Parse, 284
  - fipa\_agent\_identifier\_set\_Compose, 284
  - fipa\_agent\_identifier\_set\_Copy, 284
  - fipa\_agent\_identifier\_set\_Destroy, 284
  - fipa\_agent\_identifier\_set\_New, 284
  - fipa\_agent\_identifier\_set\_p, 279
  - fipa\_agent\_identifier\_set\_Parse, 284
  - fipa\_agent\_identifier\_set\_t, 279
  - fipa\_agent\_identifier\_t, 279
  - fipa\_CheckNextToken, 285
  - fipa\_DateTime\_Compose, 285
  - fipa\_DateTime\_Copy, 285
  - fipa\_DateTime\_Destroy, 285
  - fipa\_DateTime\_New, 285
  - fipa\_DateTime\_p, 279
  - fipa\_datetime\_Parse, 285
  - fipa\_DateTime\_t, 279
  - fipa\_expression\_Compose, 286
  - fipa\_expression\_Copy, 286
  - fipa\_expression\_Destroy, 286
  - fipa\_expression\_New, 286
  - fipa\_expression\_p, 279
  - fipa\_expression\_Parse, 286
  - fipa\_expression\_t, 279
  - fipa\_expression\_type\_e, 280
  - fipa\_expression\_type\_t, 279
  - fipa\_GetAtom, 287
  - fipa\_GetNextWord, 287
  - fipa\_GetWholeToken, 287
  - fipa\_message\_parameter\_Parse, 287
  - fipa\_message\_string\_Copy, 287
  - fipa\_message\_string\_Destroy, 287
  - fipa\_message\_string\_New, 287
  - fipa\_message\_string\_p, 279
  - fipa\_message\_string\_t, 279
  - fipa\_message\_type\_Parse, 288
  - fipa\_number\_Compose, 288
  - fipa\_number\_Copy, 288
  - fipa\_number\_Destroy, 288
  - fipa\_number\_New, 288
  - fipa\_number\_p, 279
  - fipa\_number\_t, 279
  - fipa\_performative\_Compose, 288
  - fipa\_performative\_e, 280
  - fipa\_protocol\_Compose, 289
  - fipa\_protocol\_e, 281
  - fipa\_protocol\_type\_Parse, 289
  - fipa\_Reply, 289
  - fipa\_string\_Compose, 289
  - fipa\_string\_Copy, 289
  - fipa\_string\_Destroy, 290
  - fipa\_string\_New, 290
  - fipa\_string\_p, 279
  - fipa\_string\_Parse, 290
  - fipa\_string\_t, 279
  - fipa\_url\_Compose, 290
  - fipa\_url\_Copy, 290
  - fipa\_url\_Destroy, 290
  - fipa\_url\_New, 290
  - fipa\_url\_p, 280
  - fipa\_url\_Parse, 291
  - fipa\_url\_sequence\_Compose, 291
  - fipa\_url\_sequence\_Copy, 291
  - fipa\_url\_sequence\_Destroy, 291
  - fipa\_url\_sequence\_New, 291
  - fipa\_url\_sequence\_p, 280
  - fipa\_url\_sequence\_Parse, 291
  - fipa\_url\_sequence\_t, 280
  - fipa\_url\_t, 280
  - fipa\_word\_Compose, 292
  - fipa\_word\_Copy, 292
  - fipa\_word\_Destroy, 292
  - fipa\_word\_New, 292
  - fipa\_word\_p, 280

- fipa\_word\_Parse, 292
- fipa\_word\_t, 280
- fipa\_acl\_Compose
  - fipa\_acl.c, 200
  - fipa\_acl.h, 282
- fipa\_acl\_envelope.h
  - fipa\_acl\_envelope\_Copy, 294
  - fipa\_acl\_envelope\_Destroy, 294
  - fipa\_acl\_envelope\_New, 294
  - fipa\_acl\_envelope\_p, 294
  - fipa\_acl\_envelope\_Received\_Copy, 295
  - fipa\_acl\_envelope\_Received\_Destroy, 295
  - fipa\_acl\_envelope\_Received\_New, 295
  - fipa\_acl\_envelope\_Received\_t, 294
  - fipa\_acl\_envelope\_t, 294
  - fipa\_acl\_Param\_Copy, 295
  - fipa\_acl\_Param\_Destroy, 295
  - fipa\_acl\_Param\_New, 296
  - fipa\_acl\_Param\_t, 294
  - fipa\_envelope\_Compose, 296
  - fipa\_envelope\_Compose\_\_acl\_representation, 296
  - fipa\_envelope\_Compose\_\_date, 296
  - fipa\_envelope\_Compose\_\_envelope, 296
  - fipa\_envelope\_Compose\_\_from, 296
  - fipa\_envelope\_Compose\_\_intended\_receiver, 296
  - fipa\_envelope\_Compose\_\_params, 297
  - fipa\_envelope\_Compose\_\_payload\_encoding, 297
  - fipa\_envelope\_Compose\_\_to, 297
  - fipa\_envelope\_HandleAclRepresentation, 297
  - fipa\_envelope\_HandleComments, 297
  - fipa\_envelope\_HandleDate, 298
  - fipa\_envelope\_HandleEnvelope, 298
  - fipa\_envelope\_HandleFrom, 298
  - fipa\_envelope\_HandleIntendedReceiver, 298
  - fipa\_envelope\_HandleParams, 298
  - fipa\_envelope\_HandlePayloadEncoding, 299
  - fipa\_envelope\_HandlePayloadLength, 299
  - fipa\_envelope\_HandleReceived, 299
  - fipa\_envelope\_HandleTo, 299
  - fipa\_envelope\_Parse, 300
  - fipa\_envelope\_ParseAddresses, 300
  - fipa\_envelope\_ParseAgentIdentifier, 300
  - fipa\_envelope\_ParseResolvers, 300
- fipa\_acl\_envelope\_Copy
  - fipa\_acl\_envelope.h, 294
  - fipa\_envelope.c, 212
- fipa\_acl\_envelope\_Destroy
  - fipa\_acl\_envelope.h, 294
  - fipa\_envelope.c, 212
- fipa\_acl\_envelope\_New
  - fipa\_acl\_envelope.h, 294
- fipa\_envelope.c, 212
- fipa\_acl\_envelope\_p
  - fipa\_acl\_envelope.h, 294
- fipa\_acl\_envelope\_Received\_Copy
  - fipa\_acl\_envelope.h, 295
  - fipa\_envelope.c, 212
- fipa\_acl\_envelope\_Received\_Destroy
  - fipa\_acl\_envelope.h, 295
  - fipa\_envelope.c, 212
- fipa\_acl\_envelope\_Received\_New
  - fipa\_acl\_envelope.h, 295
  - fipa\_envelope.c, 213
- fipa\_acl\_envelope\_Received\_s, 67
  - received\_by, 67
  - received\_date, 67
  - received\_from, 67
  - received\_id, 67
  - received\_via, 67
- fipa\_acl\_envelope\_Received\_t
  - fipa\_acl\_envelope.h, 294
- fipa\_acl\_envelope\_s, 69
  - num\_params, 69
  - params, 69
- fipa\_acl\_envelope\_t
  - fipa\_acl\_envelope.h, 294
- fipa\_acl\_message\_Copy
  - fipa\_acl.c, 200
  - fipa\_acl.h, 282
- fipa\_acl\_message\_Destroy
  - fipa\_acl.c, 200
  - fipa\_acl.h, 282
- fipa\_acl\_message\_New
  - fipa\_acl.c, 200
  - fipa\_acl.h, 283
- fipa\_acl\_message\_p
  - fipa\_acl.h, 279
- fipa\_acl\_message\_s, 70
  - content, 70
  - conversation\_id, 70
  - encoding, 70
  - in\_reply\_to, 70
  - language, 71
  - ontology, 71
  - performative, 71
  - protocol, 71
  - receiver, 71
  - receiver\_num, 71
  - reply\_by, 71
  - reply\_to, 72
  - reply\_with, 72
  - sender, 72
- fipa\_acl\_message\_t
  - fipa\_acl.h, 279
- fipa\_acl\_Param\_Copy

- fipa\_acl\_envelope.h, 295
- fipa\_envelope.c, 213
- fipa\_acl\_Param\_Destroy
  - fipa\_acl\_envelope.h, 295
  - fipa\_envelope.c, 213
- fipa\_acl\_Param\_New
  - fipa\_acl\_envelope.h, 296
  - fipa\_envelope.c, 213
- fipa\_acl\_Param\_s, 73
  - acl\_representation, 73
  - comments, 73
  - date, 73
  - from, 73
  - intended\_receiver, 73
  - payload\_encoding, 74
  - payload\_length, 74
  - received, 74
  - to, 74
- fipa\_acl\_Param\_t
  - fipa\_acl\_envelope.h, 294
- fipa\_acl\_Parse
  - fipa\_acl.c, 200
  - fipa\_acl.h, 283
- fipa\_agent\_identifier\_Compose
  - fipa\_acl.c, 201
  - fipa\_acl.h, 283
- fipa\_agent\_identifier\_Copy
  - fipa\_acl.c, 201
  - fipa\_acl.h, 283
- fipa\_agent\_identifier\_Destroy
  - fipa\_acl.c, 201
  - fipa\_acl.h, 283
- fipa\_agent\_identifier\_New
  - fipa\_acl.c, 201
  - fipa\_acl.h, 283
- fipa\_agent\_identifier\_p
  - fipa\_acl.h, 279
- fipa\_agent\_identifier\_Parse
  - fipa\_acl.c, 201
  - fipa\_acl.h, 284
- fipa\_agent\_identifier\_s, 75
  - addresses, 75
  - name, 75
  - resolvers, 75
- fipa\_agent\_identifier\_set\_Compose
  - fipa\_acl.c, 201
  - fipa\_acl.h, 284
- fipa\_agent\_identifier\_set\_Copy
  - fipa\_acl.c, 202
  - fipa\_acl.h, 284
- fipa\_agent\_identifier\_set\_Destroy
  - fipa\_acl.c, 202
  - fipa\_acl.h, 284
- fipa\_agent\_identifier\_set\_New
  - fipa\_acl.c, 202
  - fipa\_acl.h, 284
- fipa\_agent\_identifier\_set\_p
  - fipa\_acl.h, 279
- fipa\_agent\_identifier\_set\_Parse
  - fipa\_acl.c, 202
  - fipa\_acl.h, 284
- fipa\_agent\_identifier\_set\_s, 76
  - fipa\_agent\_identifiers, 76
  - num, 76
  - retain\_order, 76
- fipa\_agent\_identifier\_set\_t
  - fipa\_acl.h, 279
- fipa\_agent\_identifier\_t
  - fipa\_acl.h, 279
- fipa\_agent\_identifiers
  - fipa\_agent\_identifier\_set\_s, 76
- fipa\_CheckNextToken
  - fipa\_acl.c, 202
  - fipa\_acl.h, 285
- fipa\_comm.h
  - false, 303
  - fipa\_comm\_action\_p, 303
  - fipa\_comm\_action\_s, 303
  - fipa\_comm\_action\_type\_e, 305
  - fipa\_comm\_callback, 303
  - fipa\_comm\_check\_type\_e, 305
  - fipa\_comm\_logic\_type\_e, 305
  - fipa\_comm\_message\_check\_p, 303
  - fipa\_comm\_message\_check\_s, 303
  - fipa\_comm\_p, 303
  - fipa\_comm\_performative\_p, 303
  - fipa\_comm\_performative\_s, 303
  - fipa\_comm\_protocol\_cn\_p, 304
  - fipa\_comm\_protocol\_cn\_s, 304
  - fipa\_comm\_protocol\_cn\_state\_e, 305
  - fipa\_comm\_protocol\_info\_COPY, 308
  - fipa\_comm\_protocol\_p, 304
  - fipa\_comm\_protocol\_s, 304
  - fipa\_comm\_reply\_p, 304
  - fipa\_comm\_reply\_s, 304
  - fipa\_comm\_s, 304
  - fipa\_comm\_type\_e, 306
  - fipa\_list\_Add, 308
  - fipa\_list\_p, 304
  - fipa\_list\_s, 304
  - mc\_FIPAComm\_AddNewProtocol, 308
  - mc\_FIPAComm\_DeregisterPerformative, 308
  - mc\_FIPAComm\_Destroy, 308
  - mc\_FIPAComm\_FSM\_CN, 308
  - mc\_FIPAComm\_HandleMessage, 308
  - mc\_FIPAComm\_HandleMessageOnce, 308
  - mc\_FIPAComm\_HandleMessageWait, 308
  - mc\_FIPAComm\_HandleProtocol, 308

- mc\_FIPAComm\_New, 308
- mc\_FIPAComm\_NewProtocol, 308
- mc\_FIPAComm\_RegisterAction, 308
- mc\_FIPAComm\_RegisterActionS, 308
- mc\_FIPAComm\_RegisterPerformative, 308
- mc\_FIPAComm\_RegisterProtocol, 308
- mc\_FIPAComm\_RegisterProtocolCallback, 308
- mc\_FIPAComm\_SetAgent, 308
- mc\_FIPAComm\_SetDefaultReply, 308
- mc\_FIPAComm\_-
  - SetPerformativeDefaultReply, 308
- mc\_FIPAComm\_SetProtocolState, 308
- mc\_FIPAComm\_StartProtocol, 308
- mc\_FIPAProtocol\_New, 308
- MC\_NUM\_PERFORMATIVES, 303
- MCFIPAComm\_p, 304
- MCFIPAComm\_s, 304
- protocol\_contract\_net\_callback, 305
- fipa\_comm\_action\_p
  - fipa\_comm.h, 303
- fipa\_comm\_action\_s
  - fipa\_comm.h, 303
- fipa\_comm\_action\_t, 77
  - callback, 77
  - check\_list, 77
  - next, 77
  - reply, 77
  - type, 77
- fipa\_comm\_action\_type\_e
  - fipa\_comm.h, 305
- fipa\_comm\_callback
  - fipa\_comm.h, 303
- fipa\_comm\_check\_type\_e
  - fipa\_comm.h, 305
- fipa\_comm\_logic\_type\_e
  - fipa\_comm.h, 305
- fipa\_comm\_message\_check\_p
  - fipa\_comm.h, 303
- fipa\_comm\_message\_check\_s
  - fipa\_comm.h, 303
- fipa\_comm\_message\_check\_t, 78
  - macro, 78
  - next, 78
  - type, 78
- fipa\_comm\_p
  - fipa\_comm.h, 303
- fipa\_comm\_performative\_p
  - fipa\_comm.h, 303
- fipa\_comm\_performative\_s
  - fipa\_comm.h, 303
- fipa\_comm\_performative\_t, 79
  - action\_list, 79
  - default\_reply, 79
- fipa\_comm\_protocol\_cn\_p
  - fipa\_comm.h, 304
- fipa\_comm\_protocol\_cn\_s
  - fipa\_comm.h, 304
- fipa\_comm\_protocol\_cn\_state\_e
  - fipa\_comm.h, 305
- fipa\_comm\_protocol\_cn\_t, 80
  - cfp\_request, 80
  - contract\_func, 80
  - proposal\_func, 80
  - queue, 80
  - replies, 80
  - require\_inform, 80
  - time\_out, 80
- fipa\_comm\_protocol\_info\_COPY
  - fipa\_comm.h, 308
- fipa\_comm\_protocol\_p
  - fipa\_comm.h, 304
- fipa\_comm\_protocol\_s
  - fipa\_comm.h, 304
- fipa\_comm\_protocol\_t, 82
  - conversation\_id, 82
  - error, 82
  - info, 82
  - network, 82
  - next, 82
  - state, 82
  - type, 82
- fipa\_comm\_reply\_p
  - fipa\_comm.h, 304
- fipa\_comm\_reply\_s
  - fipa\_comm.h, 304
- fipa\_comm\_reply\_t, 84
  - content, 84
  - type, 84
- fipa\_comm\_s
  - fipa\_comm.h, 304
- fipa\_comm\_t, 85
  - agent, 85
  - agent\_address, 85
  - agent\_name, 85
  - debug, 85
  - default\_reply, 85
  - error, 85
  - pCallbacks, 85
  - performative, 86
  - protocol, 86
- fipa\_comm\_type\_e
  - fipa\_comm.h, 306
- fipa\_DateTime\_Compose
  - fipa\_acl.c, 202
  - fipa\_acl.h, 285
- fipa\_DateTime\_Copy
  - fipa\_acl.c, 203



- fipa\_acl.h, 285
- fipa\_DateTime\_Destroy
  - fipa\_acl.c, 203
  - fipa\_acl.h, 285
- fipa\_DateTime\_New
  - fipa\_acl.c, 203
  - fipa\_acl.h, 285
- fipa\_DateTime\_p
  - fipa\_acl.h, 279
- fipa\_datetime\_Parse
  - fipa\_acl.c, 203
  - fipa\_acl.h, 285
- fipa\_DateTime\_s, 87
  - day, 87
  - hour, 87
  - is\_utc, 87
  - millisecond, 87
  - minute, 87
  - month, 87
  - second, 88
  - sign, 88
  - year, 88
- fipa\_DateTime\_t
  - fipa\_acl.h, 279
- fipa\_envelope.c
  - fipa\_acl\_envelope\_Copy, 212
  - fipa\_acl\_envelope\_Destroy, 212
  - fipa\_acl\_envelope\_New, 212
  - fipa\_acl\_envelope\_Received\_Copy, 212
  - fipa\_acl\_envelope\_Received\_Destroy, 212
  - fipa\_acl\_envelope\_Received\_New, 213
  - fipa\_acl\_Param\_Copy, 213
  - fipa\_acl\_Param\_Destroy, 213
  - fipa\_acl\_Param\_New, 213
  - fipa\_envelope\_Compose, 213
  - fipa\_envelope\_Compose\_\_acl\_representation, 213
  - fipa\_envelope\_Compose\_\_date, 214
  - fipa\_envelope\_Compose\_\_envelope, 214
  - fipa\_envelope\_Compose\_\_from, 214
  - fipa\_envelope\_Compose\_\_intended\_receiver, 214
  - fipa\_envelope\_Compose\_\_params, 214
  - fipa\_envelope\_Compose\_\_payload\_encoding, 215
  - fipa\_envelope\_Compose\_\_to, 215
  - fipa\_envelope\_HandleAclRepresentation, 215
  - fipa\_envelope\_HandleComments, 215
  - fipa\_envelope\_HandleDate, 215
  - fipa\_envelope\_HandleEnvelope, 215
  - fipa\_envelope\_HandleFrom, 216
  - fipa\_envelope\_HandleIntendedReceiver, 216
  - fipa\_envelope\_HandleParams, 216
  - fipa\_envelope\_HandlePayloadEncoding, 216
  - fipa\_envelope\_HandlePayloadLength, 216
  - fipa\_envelope\_HandleReceived, 217
  - fipa\_envelope\_HandleTo, 217
  - fipa\_envelope\_Parse, 217
  - fipa\_envelope\_ParseAddresses, 217
  - fipa\_envelope\_ParseAgentIdentifier, 218
  - fipa\_envelope\_ParseResolvers, 218
- fipa\_envelope\_Compose
  - fipa\_acl\_envelope.h, 296
  - fipa\_envelope.c, 213
- fipa\_envelope\_Compose\_\_acl\_representation
  - fipa\_acl\_envelope.h, 296
  - fipa\_envelope.c, 213
- fipa\_envelope\_Compose\_\_date
  - fipa\_acl\_envelope.h, 296
  - fipa\_envelope.c, 214
- fipa\_envelope\_Compose\_\_envelope
  - fipa\_acl\_envelope.h, 296
  - fipa\_envelope.c, 214
- fipa\_envelope\_Compose\_\_from
  - fipa\_acl\_envelope.h, 296
  - fipa\_envelope.c, 214
- fipa\_envelope\_Compose\_\_intended\_receiver
  - fipa\_acl\_envelope.h, 296
  - fipa\_envelope.c, 214
- fipa\_envelope\_Compose\_\_params
  - fipa\_acl\_envelope.h, 297
  - fipa\_envelope.c, 214
- fipa\_envelope\_Compose\_\_payload\_encoding
  - fipa\_acl\_envelope.h, 297
  - fipa\_envelope.c, 215
- fipa\_envelope\_Compose\_\_to
  - fipa\_acl\_envelope.h, 297
  - fipa\_envelope.c, 215
- fipa\_envelope\_HandleAclRepresentation
  - fipa\_acl\_envelope.h, 297
  - fipa\_envelope.c, 215
- fipa\_envelope\_HandleComments
  - fipa\_acl\_envelope.h, 297
  - fipa\_envelope.c, 215
- fipa\_envelope\_HandleDate
  - fipa\_acl\_envelope.h, 298
  - fipa\_envelope.c, 215
- fipa\_envelope\_HandleEnvelope
  - fipa\_acl\_envelope.h, 298
  - fipa\_envelope.c, 215
- fipa\_envelope\_HandleFrom
  - fipa\_acl\_envelope.h, 298
  - fipa\_envelope.c, 216
- fipa\_envelope\_HandleIntendedReceiver
  - fipa\_acl\_envelope.h, 298
  - fipa\_envelope.c, 216
- fipa\_envelope\_HandleParams
  - fipa\_acl\_envelope.h, 298



- fipa\_envelope.c, 216
- fipa\_envelope\_HandlePayloadEncoding
  - fipa\_acl\_envelope.h, 299
  - fipa\_envelope.c, 216
- fipa\_envelope\_HandlePayloadLength
  - fipa\_acl\_envelope.h, 299
  - fipa\_envelope.c, 216
- fipa\_envelope\_HandleReceived
  - fipa\_acl\_envelope.h, 299
  - fipa\_envelope.c, 217
- fipa\_envelope\_HandleTo
  - fipa\_acl\_envelope.h, 299
  - fipa\_envelope.c, 217
- fipa\_envelope\_Parse
  - fipa\_acl\_envelope.h, 300
  - fipa\_envelope.c, 217
- fipa\_envelope\_ParseAddresses
  - fipa\_acl\_envelope.h, 300
  - fipa\_envelope.c, 217
- fipa\_envelope\_ParseAgentIdentifier
  - fipa\_acl\_envelope.h, 300
  - fipa\_envelope.c, 218
- fipa\_envelope\_ParseResolvers
  - fipa\_acl\_envelope.h, 300
  - fipa\_envelope.c, 218
- fipa\_expression\_Compose
  - fipa\_acl.c, 203
  - fipa\_acl.h, 286
- fipa\_expression\_Copy
  - fipa\_acl.c, 203
  - fipa\_acl.h, 286
- fipa\_expression\_Destroy
  - fipa\_acl.c, 204
  - fipa\_acl.h, 286
- fipa\_expression\_New
  - fipa\_acl.c, 204
  - fipa\_acl.h, 286
- fipa\_expression\_p
  - fipa\_acl.h, 279
- fipa\_expression\_Parse
  - fipa\_acl.c, 204
  - fipa\_acl.h, 286
- fipa\_expression\_s, 89
  - content, 89
  - type, 89
- fipa\_expression\_s::content\_u, 60
  - datetime, 60
  - expression, 60
  - number, 60
  - string, 60
  - word, 60
- fipa\_expression\_t
  - fipa\_acl.h, 279
- fipa\_expression\_type\_e
  - fipa\_acl.h, 280
- fipa\_expression\_type\_t
  - fipa\_acl.h, 279
- fipa\_GetAtom
  - fipa\_acl.c, 204
  - fipa\_acl.h, 287
- fipa\_GetNextWord
  - fipa\_acl.c, 204
  - fipa\_acl.h, 287
- fipa\_GetWholeToken
  - fipa\_acl.c, 205
  - fipa\_acl.h, 287
- fipa\_list\_Add
  - fipa\_comm.h, 308
- fipa\_list\_p
  - fipa\_comm.h, 304
- fipa\_list\_s
  - fipa\_comm.h, 304
- fipa\_list\_t, 90
  - data, 90
  - next, 90
- fipa\_message\_parameter\_Parse
  - fipa\_acl.c, 205
  - fipa\_acl.h, 287
- fipa\_message\_string\_Copy
  - fipa\_acl.c, 205
  - fipa\_acl.h, 287
- fipa\_message\_string\_Destroy
  - fipa\_acl.c, 205
  - fipa\_acl.h, 287
- fipa\_message\_string\_New
  - fipa\_acl.c, 205
  - fipa\_acl.h, 287
- fipa\_message\_string\_p
  - fipa\_acl.h, 279
- fipa\_message\_string\_s, 91
  - message, 91
  - parse, 91
- fipa\_message\_string\_t
  - fipa\_acl.h, 279
- fipa\_message\_type\_Parse
  - fipa\_acl.c, 205
  - fipa\_acl.h, 288
- fipa\_number\_Compose
  - fipa\_acl.c, 206
  - fipa\_acl.h, 288
- fipa\_number\_Copy
  - fipa\_acl.c, 206
  - fipa\_acl.h, 288
- fipa\_number\_Destroy
  - fipa\_acl.c, 206
  - fipa\_acl.h, 288
- fipa\_number\_New
  - fipa\_acl.c, 206

- fipa\_acl.h, 288
- fipa\_number\_p
  - fipa\_acl.h, 279
- fipa\_number\_s, 92
  - str, 92
- fipa\_number\_t
  - fipa\_acl.h, 279
- fipa\_performative\_Compose
  - fipa\_acl.c, 206
  - fipa\_acl.h, 288
- fipa\_performative\_e
  - fipa\_acl.h, 280
  - libmc.h, 322
- fipa\_protocol\_Compose
  - fipa\_acl.c, 206
  - fipa\_acl.h, 289
- fipa\_protocol\_e
  - fipa\_acl.h, 281
- fipa\_protocol\_type\_Parse
  - fipa\_acl.c, 207
  - fipa\_acl.h, 289
- fipa\_Reply
  - fipa\_acl.c, 207
  - fipa\_acl.h, 289
- fipa\_string\_Compose
  - fipa\_acl.c, 207
  - fipa\_acl.h, 289
- fipa\_string\_Copy
  - fipa\_acl.c, 207
  - fipa\_acl.h, 289
- fipa\_string\_Destroy
  - fipa\_acl.c, 207
  - fipa\_acl.h, 290
- fipa\_string\_New
  - fipa\_acl.c, 208
  - fipa\_acl.h, 290
- fipa\_string\_p
  - fipa\_acl.h, 279
- fipa\_string\_Parse
  - fipa\_acl.c, 208
  - fipa\_acl.h, 290
- fipa\_string\_s, 93
  - content, 93
- fipa\_string\_t
  - fipa\_acl.h, 279
- fipa\_url\_Compose
  - fipa\_acl.c, 208
  - fipa\_acl.h, 290
- fipa\_url\_Copy
  - fipa\_acl.c, 208
  - fipa\_acl.h, 290
- fipa\_url\_Destroy
  - fipa\_acl.c, 208
  - fipa\_acl.h, 290
- fipa\_url\_New
  - fipa\_acl.c, 208
  - fipa\_acl.h, 290
- fipa\_url\_p
  - fipa\_acl.h, 280
- fipa\_url\_Parse
  - fipa\_acl.c, 208
  - fipa\_acl.h, 291
- fipa\_url\_s, 94
  - str, 94
- fipa\_url\_sequence\_Compose
  - fipa\_acl.c, 208
  - fipa\_acl.h, 291
- fipa\_url\_sequence\_Copy
  - fipa\_acl.c, 209
  - fipa\_acl.h, 291
- fipa\_url\_sequence\_Destroy
  - fipa\_acl.c, 209
  - fipa\_acl.h, 291
- fipa\_url\_sequence\_New
  - fipa\_acl.c, 209
  - fipa\_acl.h, 291
- fipa\_url\_sequence\_p
  - fipa\_acl.h, 280
- fipa\_url\_sequence\_Parse
  - fipa\_acl.c, 209
  - fipa\_acl.h, 291
- fipa\_url\_sequence\_s, 95
  - num, 95
  - urls, 95
- fipa\_url\_sequence\_t
  - fipa\_acl.h, 280
- fipa\_url\_t
  - fipa\_acl.h, 280
- fipa\_word\_Compose
  - fipa\_acl.c, 209
  - fipa\_acl.h, 292
- fipa\_word\_Copy
  - fipa\_acl.c, 209
  - fipa\_acl.h, 292
- fipa\_word\_Destroy
  - fipa\_acl.c, 210
  - fipa\_acl.h, 292
- fipa\_word\_New
  - fipa\_acl.c, 210
  - fipa\_acl.h, 292
- fipa\_word\_p
  - fipa\_acl.h, 280
- fipa\_word\_Parse
  - fipa\_acl.c, 210
  - fipa\_acl.h, 292
- fipa\_word\_s, 96
  - content, 96
- fipa\_word\_t

- fipa\_acl.h, 280
- foo
  - class.cxx, 528
  - foo\_c, 98
  - foo\_s, 101
- FOO\_BLUE
  - enum.cxx, 529
- FOO\_ONE
  - enum.cxx, 529
- FOO\_RED
  - enum.cxx, 529
- FOO\_TWO
  - enum.cxx, 529
- foo\_c, 97
  - ~foo\_c, 97
  - bar, 98
  - barfoo, 98
  - foo, 98
  - foo\_c, 97
  - foo\_c, 97
  - get\_bar, 97
  - get\_barfoo, 97
  - get\_foo, 98
  - get\_global, 98
  - global, 99
  - set\_bar, 98
  - set\_foo, 98
  - set\_foobar, 98
- foo\_default\_int
  - function.cxx, 530
- foo\_default\_string
  - function.cxx, 530
- foo\_enum\_e
  - enum.cxx, 529
- foo\_enum\_t
  - enum.cxx, 529
- foo\_float\_function
  - function.cxx, 530
- foo\_s, 100
  - ~foo\_s, 100
  - bar, 101
  - foo, 101
  - foo\_s, 100
  - foo\_s, 100
  - get\_bar, 100
  - get\_foo, 100
  - set\_bar, 100
  - set\_foo, 101
- foo\_t
  - struct.cxx, 531
- foo\_void\_function
  - function.cxx, 530
- FORBIDDEN
  - mtp\_http.h, 390
- force\_cipher
  - options, 149
- FREEMEM
  - fipa\_acl.c, 199
- from
  - fipa\_acl\_Param\_s, 73
- from\_address
  - message\_s, 127
- FSb
  - aes.c, 694
- FT0
  - aes.c, 694
- FT1
  - aes.c, 694
- FT2
  - aes.c, 694
- FT3
  - aes.c, 694
- function.cxx
  - foo\_default\_int, 530
  - foo\_default\_string, 530
  - foo\_float\_function, 530
  - foo\_void\_function, 530
- G
  - dhm\_context, 64
- g\_mc\_platform
  - libmc.c, 461
- generate\_AES\_key
  - interface.c, 545
  - interface.h, 550
- generate\_encrypt\_send\_AES\_key
  - interface.h, 550
- generate\_RSA\_keys\_ciphertext
  - interface.c, 545
  - interface.h, 550
- generate\_RSA\_keys\_plaintext
  - interface.c, 545
  - interface.h, 550
- GENERATOR
  - dh\_genprime.c, 837
- get\_bar
  - foo\_c, 97
  - foo\_s, 100
- get\_barfoo
  - foo\_c, 97
- get\_foo
  - foo\_c, 98
  - foo\_s, 100
- get\_global
  - foo\_c, 98
- GET\_REQUEST
  - ssl\_client1.c, 843
  - ssl\_client2.c, 845

- GET\_THREAD\_MODE
  - macros.h, 370
- get\_timer
  - timing.c, 818
  - timing.h, 670
- GET\_UINT64\_BE
  - sha4.c, 790
- GET\_ULONG\_BE
  - des.c, 728
  - sha1.c, 772
  - sha2.c, 781
- GET\_ULONG\_LE
  - aes.c, 689
  - md5.c, 748
- giant
  - mc\_platform\_s, 112
- giant\_cond
  - mc\_platform\_s, 112
- giant\_lock
  - mc\_platform\_s, 112
  - syncList\_s, 163
- global
  - foo\_c, 99
- GONE
  - mtp\_http.h, 390
- GX
  - dhm\_context, 64
- GY
  - dhm\_context, 64
- handler\_COMPOSE\_SEND
  - cmd\_prompt.c, 187
- handler\_FLUSH\_AGENTS
  - cmd\_prompt.c, 187
- handler\_HELP
  - cmd\_prompt.c, 187
- handler\_PRINT\_CONNECTLIST
  - cmd\_prompt.c, 187
- handler\_PRINTLIST\_AGENTS
  - cmd\_prompt.c, 188
- handler\_PRINTLIST\_MESSAGE
  - cmd\_prompt.c, 188
- handler\_QUIT
  - cmd\_prompt.c, 188
- handler\_SEND
  - cmd\_prompt.c, 188
- hardclock
  - timing.c, 819
  - timing.h, 670
- hardclock\_init
  - timing.c, 819
- hash\_id
  - rsa\_context, 153
- HAVE\_STRDUP
  - mxml-2.2.2/config.h, 487
  - mxml-2.2.2/winconfig.h, 533
- HAVE\_VSNPRINTF
  - mxml-2.2.2/config.h, 487
  - mxml-2.2.2/winconfig.h, 533
- havege.c
  - havege\_fill, 744
  - havege\_init, 744
  - havege\_rand, 744
  - ONE\_ITERATION, 743
  - SWAP, 743
  - TST1\_ENTER, 743
  - TST1\_LEAVE, 743
  - TST2\_ENTER, 743
  - TST2\_LEAVE, 744
- havege.h
  - COLLECT\_SIZE, 596
  - havege\_init, 596
  - havege\_rand, 596
- havege\_fill
  - havege.c, 744
- havege\_init
  - havege.c, 744
  - havege.h, 596
- havege\_rand
  - havege.c, 744
  - havege.h, 596
- havege\_state, 102
  - offset, 102
  - pool, 102
  - PT1, 102
  - PT2, 102
  - WALK, 102
- header\_length
  - mtp\_http\_s, 135
- hello.c
  - main, 832
- help
  - commands.x.h, 259
- HERE
  - df.h, 267
- home
  - agent\_s, 46
- home\_port
  - agent\_s, 47
- host
  - mtp\_http\_s, 135
- host\_id.h
  - host\_id\_Compare, 309
  - host\_id\_Destroy, 309
  - host\_id\_Initialize, 309
  - host\_id\_p, 309
  - host\_id\_t, 309
- host\_id\_Compare

- host\_id.h, 309
- host\_id\_Destroy
  - host\_id.h, 309
- host\_id\_Initialize
  - host\_id.h, 309
- host\_id\_p
  - host\_id.h, 309
- host\_id\_s, 104
  - hostname, 104
  - port, 104
- host\_id\_t
  - host\_id.h, 309
- hostName
  - agency\_s, 37
- hostname
  - \_ssl\_context, 13
  - host\_id\_s, 104
  - mc\_platform\_s, 113
- hostname\_len
  - \_ssl\_context, 13
- hour
  - \_x509\_time, 33
  - fipa\_DateTime\_s, 87
- hr\_time, 105
  - opaque, 105
- HTTP\_CONNECT
  - mtp\_http.h, 390
- HTTP\_DELETE
  - mtp\_http.h, 390
- HTTP\_GET
  - mtp\_http.h, 389
- HTTP\_HEAD
  - mtp\_http.h, 389
- HTTP\_NUM\_PERFORMATIVES
  - mtp\_http.h, 390
- HTTP\_OPTIONS
  - mtp\_http.h, 390
- HTTP\_PERFORMATIVE\_UNDEF
  - mtp\_http.h, 389
- HTTP\_PERFORMATIVE\_ZERO
  - mtp\_http.h, 389
- HTTP\_POST
  - mtp\_http.h, 389
- HTTP\_PUT
  - mtp\_http.h, 390
- HTTP\_RESPONSE
  - mtp\_http.h, 390
- HTTP\_TRACE
  - mtp\_http.h, 390
- http\_GetExpression
  - mtp\_http.c, 483
  - mtp\_http.h, 391
- http\_GetToken
  - mtp\_http.c, 483
- mtp\_http.h, 391
- HTTP\_PARSE\_EXPR
  - mtp\_http.c, 483
- http\_ParseExpression
  - mtp\_http.c, 484
  - mtp\_http.h, 391
- http\_ParseRequest
  - mtp\_http.c, 484
  - mtp\_http.h, 392
- http\_performative
  - mtp\_http\_s, 135
- http\_performative\_e
  - mtp\_http.h, 389
- HTTP\_RESPONSE
  - ssl\_server.c, 847
- http\_status\_code
  - mtp\_http\_s, 135
- http\_status\_code\_e
  - mtp\_http.h, 390
- http\_to\_hostport
  - message.c, 478
  - message.h, 384
- http\_type
  - message\_s, 128
- http\_version
  - mtp\_http\_s, 136
- id
  - \_ssl\_session, 22
  - agent\_s, 47
  - barrier\_node\_s, 54
  - syncListNode\_s, 164
- in\_ctr
  - \_ssl\_context, 13
- in\_hdr
  - \_ssl\_context, 14
- in\_hslen
  - \_ssl\_context, 14
- in\_left
  - \_ssl\_context, 14
- in\_msg
  - \_ssl\_context, 14
- in\_msglen
  - \_ssl\_context, 14
- in\_msgtype
  - \_ssl\_context, 14
- in\_offt
  - \_ssl\_context, 15
- in\_reply\_to
  - fipa\_acl\_message\_s, 70
- index
  - command\_s, 57
- index\_compare
  - mxml-index.c, 501

- index\_find
  - mxml-index.c, 501
- index\_sort
  - mxml-index.c, 501
- info
  - fipa\_comm\_protocol\_t, 82
- init\_agent\_status
  - agent\_datastate\_s, 40
  - agent\_task\_s, 49
- initiate\_migration\_process
  - interface.c, 545
  - interface.h, 550
- initInterps
  - agency\_s, 37
  - mc\_platform\_s, 113
  - MCAgencyOptions\_s, 119
- INSERT
  - df.h, 267
- int
  - data\_structures.h, 264
- int64
  - sha4.h, 642
- integer
  - mxml\_value\_u, 147
- intended\_receiver
  - fipa\_acl\_Param\_s, 73
- interface.c
  - aes\_en\_de, 544
  - append\_nonce\_to\_MA, 544
  - extract\_nonce\_from\_MA, 544
  - generate\_AES\_key, 545
  - generate\_RSA\_keys\_ciphertext, 545
  - generate\_RSA\_keys\_plaintext, 545
  - initiate\_migration\_process, 545
  - read\_encrypted\_file, 545
  - read\_known\_host\_file, 545
  - receive\_AES\_en\_MA, 546
  - remove\_nonce\_from\_MA, 546
  - reply\_migration\_process, 546
  - rsa\_decryption, 546
  - rsa\_encryption, 546
  - send\_AES\_en\_MA, 546
  - separate\_key\_parts, 547
- interface.h
  - aes\_en\_de, 550
  - EXPONENT, 549
  - generate\_AES\_key, 550
  - generate\_encrypt\_send\_AES\_key, 550
  - generate\_RSA\_keys\_ciphertext, 550
  - generate\_RSA\_keys\_plaintext, 550
  - initiate\_migration\_process, 550
  - KEY\_SIZE, 549
  - MAXDATASIZE, 549
  - MODE\_DECRYPT, 549
  - MODE\_ENCRYPT, 549
  - PATH, 549
  - read\_encrypted\_file, 550
  - read\_known\_host\_file, 551
  - receive\_AES\_en\_MA, 551
  - receive\_decrypt\_AES\_key, 551
  - receiving\_decrypting\_MA\_RSA, 551
  - receiving\_verifying\_MA\_RSA, 551
  - reply\_migration\_process, 551
  - RSA\_DE, 549
  - rsa\_decryption, 551
  - RSA\_EN, 549
  - rsa\_encryption, 551
  - send\_AES\_en\_MA, 551
- interp\_options
  - mc\_platform\_s, 113
- interpreter\_queue
  - mc\_platform\_s, 113
- interpreter\_queue\_CreateRetrieve
  - ap\_queue\_template.h, 248
- interpreter\_variable\_data.h
  - interpreter\_variable\_data\_Copy, 310
  - interpreter\_variable\_data\_Destroy, 310
  - interpreter\_variable\_data\_Initialize, 311
  - interpreter\_variable\_data\_InitializeFromAgent, 311
  - interpreter\_variable\_data\_New, 311
  - interpreter\_variable\_data\_p, 310
  - interpreter\_variable\_data\_t, 310
- interpreter\_variable\_data\_Copy
  - agent\_return\_data.c, 179
  - interpreter\_variable\_data.h, 310
- interpreter\_variable\_data\_Destroy
  - agent\_return\_data.c, 179
  - interpreter\_variable\_data.h, 310
- interpreter\_variable\_data\_Initialize
  - agent\_return\_data.c, 179
  - interpreter\_variable\_data.h, 311
- interpreter\_variable\_data\_InitializeFromAgent
  - agent\_return\_data.c, 179
  - interpreter\_variable\_data.h, 311
- interpreter\_variable\_data\_New
  - agent\_return\_data.c, 180
  - interpreter\_variable\_data.h, 311
- interpreter\_variable\_data\_p
  - interpreter\_variable\_data.h, 310
- interpreter\_variable\_data\_s, 106
  - array\_dim, 106
  - array\_extent, 106
  - data, 106
  - data\_type, 106
  - name, 107
  - size, 107
- interpreter\_variable\_data\_t

- interpreter\_variable\_data.h, 310
- iomode
  - options, 150
- IOMODE\_BLOCK
  - ssl\_test.c, 855
- IOMODE\_NONBLOCK
  - ssl\_test.c, 856
- ipad
  - md2\_context, 121
  - md4\_context, 123
  - md5\_context, 125
  - sha1\_context, 156
  - sha2\_context, 158
  - sha4\_context, 160
- is
  - commands.x.h, 259
- is224
  - sha2\_context, 158
- is384
  - sha4\_context, 160
- is\_utc
  - fipa\_DateTime\_s, 87
- isHTTP
  - message\_s, 128
- issuer
  - \_x509\_cert, 25
  - \_x509\_raw, 31
- issuer\_id
  - \_x509\_cert, 25
- issuer\_raw
  - \_x509\_cert, 25
- iv\_dec
  - \_ssl\_context, 15
- iv\_enc
  - \_ssl\_context, 15
- ivlen
  - \_ssl\_context, 15
- K
  - dhm\_context, 64
  - sha1.c, 772, 773
  - sha4.c, 794
- KEY\_LEN
  - rsa.c, 762
- KEY\_SIZE
  - interface.h, 549
  - rsa\_genkey.c, 840
- keylen
  - \_ssl\_context, 15
- known\_host\_filename
  - agency\_s, 37
  - MCAgencyOptions\_s, 120
- language
  - fipa\_acl\_message\_s, 71
- last\_child
  - mxml\_node\_s, 143
- last\_error
  - agency\_s, 37
- lcppm5
  - ssl\_test.c, 857
- left
  - md2\_context, 121
- len
  - \_x509\_buf, 24
  - \_x509\_node, 30
  - dhm\_context, 65
  - dynstring\_s, 66
  - rsa\_context, 153
- length
  - \_ssl\_session, 22
- LENGTH\_REQUIRED
  - mtp\_http.h, 390
- LHs
  - des.c, 734
- libmc.c
  - g\_mc\_platform, 461
  - MC\_AclAddReceiver, 412
  - MC\_AclAddReceiver\_chdl, 412
  - MC\_AclAddReplyTo, 412
  - MC\_AclAddReplyTo\_chdl, 413
  - MC\_AclDestroy, 413
  - MC\_AclDestroy\_chdl, 413
  - MC\_AclGetContent, 413
  - MC\_AclGetContent\_chdl, 413
  - MC\_AclGetConversationID, 414
  - MC\_AclGetConversationID\_chdl, 414
  - MC\_AclGetPerformative, 414
  - MC\_AclGetPerformative\_chdl, 414
  - MC\_AclGetProtocol, 414
  - MC\_AclGetProtocol\_chdl, 414
  - MC\_AclGetSender, 415
  - MC\_AclGetSender\_chdl, 415
  - MC\_AclNew, 415
  - MC\_AclNew\_chdl, 415
  - MC\_AclPost, 415
  - MC\_AclPost\_chdl, 416
  - MC\_AclReply, 416
  - MC\_AclReply\_chdl, 416
  - MC\_AclRetrieve, 416
  - MC\_AclRetrieve\_chdl, 417
  - MC\_AclSend, 417
  - MC\_AclSend\_chdl, 417
  - MC\_AclSetContent, 417
  - MC\_AclSetContent\_chdl, 417
  - MC\_AclSetConversationID, 418
  - MC\_AclSetConversationID\_chdl, 418
  - MC\_AclSetPerformative, 418

- MC\_AclSetPerformative\_chdl, 418
- MC\_AclSetProtocol, 418
- MC\_AclSetProtocol\_chdl, 418
- MC\_AclSetSender, 419
- MC\_AclSetSender\_chdl, 419
- MC\_AclWaitRetrieve, 419
- MC\_AclWaitRetrieve\_chdl, 419
- MC\_AddAgent, 419
- MC\_AddAgent\_chdl, 420
- MC\_AddAgentInitCallback, 420
- MC\_AddStationaryAgent, 420
- MC\_AgentAddTask, 421
- MC\_AgentAddTask\_chdl, 421
- MC\_AgentAddTaskFromFile, 421
- MC\_AgentInfo\_GetAgency, 422
- MC\_AgentInfo\_GetAgent, 422
- MC\_AgentInfo\_GetAgentArgs, 422
- MC\_AgentReturnArrayDim, 423
- MC\_AgentReturnArrayExtent, 423
- MC\_AgentReturnArrayNum, 423
- MC\_AgentReturnDataGetSymbolAddr, 424
- MC\_AgentReturnDataSize, 424
- MC\_AgentReturnDataType, 424
- MC\_AgentReturnIsArray, 425
- MC\_AgentVariableRetrieve, 425
- MC\_AgentVariableRetrieve\_chdl, 426
- MC\_AgentVariableRetrieveInfo, 426
- MC\_AgentVariableSave, 426
- MC\_AgentVariableSave\_chdl, 427
- MC\_Barrier, 427
- MC\_Barrier\_chdl, 427
- MC\_BarrierDelete, 427
- MC\_BarrierDelete\_chdl, 428
- MC\_BarrierInit, 428
- MC\_BarrierInit\_chdl, 428
- MC\_CallAgentFunc, 428
- MC\_CallAgentFunc\_chdl, 429
- MC\_CallAgentFuncArg, 429
- MC\_CallAgentFuncV, 430
- MC\_CallAgentFuncVar, 430
- MC\_ComposeAgent, 430
- MC\_ComposeAgent\_chdl, 431
- MC\_ComposeAgentFromFile, 431
- MC\_ComposeAgentFromFile\_chdl, 431
- MC\_ComposeAgentFromFileS, 432
- MC\_ComposeAgentFromFileWithWorkgroup, 432
- MC\_ComposeAgentFromFileWithWorkgroup\_chdl, 433
- MC\_ComposeAgentS, 433
- MC\_ComposeAgentWithWorkgroup, 434
- MC\_ComposeAgentWithWorkgroup\_chdl, 434
- MC\_CondBroadcast, 435
- MC\_CondBroadcast\_chdl, 435
- MC\_CondReset, 435
- MC\_CondReset\_chdl, 436
- MC\_CondSignal, 436
- MC\_CondSignal\_chdl, 436
- MC\_CondWait, 436
- MC\_CondWait\_chdl, 437
- MC\_CopyAgent, 437
- MC\_DeleteAgent, 437
- MC\_DeleteAgent\_chdl, 438
- MC\_DeleteAgentWG, 438
- MC\_DeleteAgentWG\_chdl, 438
- MC\_DeregisterService, 438
- MC\_DeregisterService\_chdl, 439
- MC\_DestroyServiceSearchResult, 439
- MC\_DestroyServiceSearchResult\_chdl, 439
- MC\_End, 439
- MC\_End\_chdl, 440
- MC\_FindAgentByID, 440
- MC\_FindAgentByID\_chdl, 440
- MC\_FindAgentByName, 440
- MC\_FindAgentByName\_chdl, 441
- MC\_GetAgentArrivalTime, 441
- MC\_GetAgentArrivalTime\_chdl, 441
- MC\_GetAgentExecEngine, 441
- MC\_GetAgentID, 441
- MC\_GetAgentID\_chdl, 442
- MC\_GetAgentName, 442
- MC\_GetAgentName\_chdl, 442
- MC\_GetAgentNumTasks, 442
- MC\_GetAgentNumTasks\_chdl, 442
- MC\_GetAgentReturnData, 442
- MC\_GetAgentStatus, 443
- MC\_GetAgentStatus\_chdl, 443
- MC\_GetAgentType, 443
- MC\_GetAgentXMLString, 444
- MC\_GetAgentXMLString\_chdl, 444
- MC\_GetAllAgents, 444
- MC\_GetTimeOfDay\_chdl, 444
- MC\_HaltAgency, 444
- MC\_HaltAgency\_chdl, 445
- MC\_Initialize, 445
- MC\_InitializeAgencyOptions, 445
- MC\_LoadAgentFromFile, 446
- MC\_MainLoop, 446
- MC\_MigrateAgent, 446
- MC\_MigrateAgent\_chdl, 447
- MC\_MutexLock, 447
- MC\_MutexLock\_chdl, 447
- MC\_MutexUnlock, 448
- MC\_MutexUnlock\_chdl, 448
- MC\_PrintAgentCode, 448
- MC\_PrintAgentCode\_chdl, 448
- MC\_RegisterService, 448



- MC\_RegisterService\_chdl, 449
- MC\_ResetSignal, 449
- MC\_ResumeAgency, 449
- MC\_ResumeAgency\_chdl, 450
- MC\_RetrieveAgent, 450
- MC\_RetrieveAgent\_chdl, 450
- MC\_RetrieveAgentCode, 450
- MC\_RetrieveAgentCode\_chdl, 451
- MC\_SaveData\_chdl, 451
- MC\_SearchForService, 451
- MC\_SearchForService\_chdl, 451
- MC\_SemaphorePost, 452
- MC\_SemaphorePost\_chdl, 452
- MC\_SemaphoreWait, 452
- MC\_SemaphoreWait\_chdl, 453
- MC\_SendAgent, 453
- MC\_SendAgentFile, 453
- MC\_SendAgentMigrationMessage, 453
- MC\_SendAgentMigrationMessage\_chdl, 454
- MC\_SendAgentMigrationMessageFile, 454
- MC\_SendAgentMigrationMessageFile\_chdl, 454
- MC\_SendCh, 454
- MC\_SendSteerCommand, 454
- MC\_SendSteerCommand\_chdl, 455
- MC\_SetAgentStatus, 455
- MC\_SetAgentStatus\_chdl, 455
- MC\_SetDefaultAgentStatus, 455
- MC\_SetDefaultAgentStatus\_chdl, 456
- MC\_SetThreadOff, 456
- MC\_SetThreadOn, 456
- MC\_SetThreadsAllOff, 457
- MC\_SetThreadsAllOn, 457
- MC\_Steer, 457
- MC\_SteerControl, 458
- MC\_SyncDelete, 458
- MC\_SyncDelete\_chdl, 458
- MC\_SyncInit, 459
- MC\_SyncInit\_chdl, 459
- MC\_TerminateAgent, 459
- MC\_TerminateAgent\_chdl, 460
- MC\_TerminateAgentWG, 460
- MC\_TerminateAgentWG\_chdl, 460
- MC\_WaitAgent, 460
- MC\_WaitRetrieveAgent, 460
- MC\_WaitSignal, 461
- libmc.h
  - \_ERROR\_CODE\_E\_, 320
  - \_FIPA\_PERFORMATIVE\_E\_, 320
  - agency\_p, 321
  - agency\_t, 321
  - agent\_p, 321
  - AGENT\_T, 320
  - agent\_t, 321
  - ERR, 322
  - error\_code\_e, 322
  - error\_code\_t, 321
  - EXPORTMC, 320
  - FIPA\_ACCEPT\_PROPOSAL, 322, 323
  - FIPA\_AGREE, 322, 323
  - FIPA\_CALL\_FOR\_PROPOSAL, 322, 323
  - FIPA\_CANCEL, 322, 323
  - FIPA\_CONFIRM, 322, 323
  - FIPA\_DISCONFIRM, 322, 323
  - FIPA\_ERROR, 322, 323
  - FIPA\_FAILURE, 322, 323
  - FIPA\_INFORM, 323
  - FIPA\_INFORM\_IF, 323
  - FIPA\_INFORM\_REF, 323
  - FIPA\_NOT\_UNDERSTOOD, 323
  - FIPA\_PROPOGATE, 323
  - FIPA\_PROPOSE, 323
  - FIPA\_PROXY, 323
  - FIPA\_QUERY\_IF, 323
  - FIPA\_QUERY\_REF, 323
  - FIPA\_REFUSE, 323
  - FIPA\_REJECT\_PROPOSAL, 323
  - FIPA\_REQUEST, 323
  - FIPA\_REQUEST\_WHEN, 323
  - FIPA\_REQUEST\_WHENEVER, 323
  - FIPA\_SUBSCRIBE, 323
  - FIPA\_ZERO, 322, 323
  - fipa\_performative\_e, 322
  - MC\_AGENT\_ACTIVE, 324
  - MC\_AGENT\_NEUTRAL, 324
  - MC\_AGENT\_SUSPENDED, 324
  - MC\_ALL\_SIGNALS, 324
  - MC\_ERR, 322
  - MC\_ERR\_CONNECT, 322
  - MC\_ERR\_EMPTY, 322
  - MC\_ERR\_INVALID, 322
  - MC\_ERR\_INVALID\_ARGS, 322
  - MC\_ERR\_MEMORY, 322
  - MC\_ERR\_NOT\_FOUND, 322
  - MC\_ERR\_PARSE, 322
  - MC\_ERR\_SEND, 322
  - MC\_EXEC\_AGENT, 324
  - MC\_LOCAL\_AGENT, 324
  - MC\_NO\_SIGNAL, 324
  - MC\_NO\_STATUS, 324
  - MC\_NONE, 324
  - MC\_RECV\_AGENT, 324
  - MC\_RECV\_CONNECTION, 324
  - MC\_RECV\_MESSAGE, 324
  - MC\_RECV\_RETURN, 324
  - MC\_REMOTE\_AGENT, 324
  - MC\_RESTART, 325
  - MC\_RETURN\_AGENT, 324

- MC\_RUN, [325](#)
- MC\_STOP, [325](#)
- MC\_SUCCESS, [322](#)
- MC\_SUSPEND, [325](#)
- MC\_THREAD\_ACC, [325](#)
- MC\_THREAD\_AGENT, [325](#)
- MC\_THREAD\_ALL, [325](#)
- MC\_THREAD\_AMS, [325](#)
- MC\_THREAD\_CP, [325](#)
- MC\_THREAD\_DF, [325](#)
- MC\_WAIT\_CH, [324](#)
- MC\_WAIT\_FINISHED, [324](#)
- MC\_WAIT\_MESSGSEND, [324](#)
- MC\_WARN\_DUPLICATE, [322](#)
- MC\_AclAddReceiver, [325](#)
- MC\_AclAddReplyTo, [325](#)
- MC\_AclDestroy, [325](#)
- MC\_AclGetContent, [326](#)
- MC\_AclGetConversationID, [326](#)
- MC\_AclGetPerformative, [326](#)
- MC\_AclGetProtocol, [326](#)
- MC\_AclGetSender, [326](#)
- MC\_AclNew, [327](#)
- MC\_AclPost, [327](#)
- MC\_AclReply, [327](#)
- MC\_AclRetrieve, [328](#)
- MC\_AclSend, [328](#)
- MC\_AclSetContent, [328](#)
- MC\_AclSetConversationID, [329](#)
- MC\_AclSetPerformative, [329](#)
- MC\_AclSetProtocol, [329](#)
- MC\_AclSetSender, [329](#)
- MC\_AclWaitRetrieve, [329](#)
- MC\_AddAgent, [330](#)
- MC\_AddAgentInitCallback, [330](#)
- MC\_AddStationaryAgent, [330](#)
- MC\_AgentAddTask, [331](#)
- MC\_AgentAddTaskFromFile, [331](#)
- MC\_AgentInfo\_GetAgency, [332](#)
- MC\_AgentInfo\_GetAgent, [332](#)
- MC\_AgentInfo\_GetAgentArgs, [332](#)
- MC\_AgentInitCallbackFunc\_t, [321](#)
- MC\_AgentReturnArrayDim, [333](#)
- MC\_AgentReturnArrayExtent, [333](#)
- MC\_AgentReturnArrayNum, [333](#)
- MC\_AgentReturnDataGetSymbolAddr, [334](#)
- MC\_AgentReturnDataSize, [334](#)
- MC\_AgentReturnDataType, [334](#)
- MC\_AgentReturnIsArray, [335](#)
- MC\_AgentStatus\_e, [323](#)
- MC\_AgentType\_e, [324](#)
- MC\_AgentVariableRetrieve, [335](#)
- MC\_AgentVariableRetrieveInfo, [336](#)
- MC\_AgentVariableSave, [336](#)
- MC\_Barrier, [337](#)
- MC\_BarrierDelete, [337](#)
- MC\_BarrierInit, [337](#)
- MC\_CallAgentFunc, [338](#)
- MC\_CallAgentFuncArg, [338](#)
- MC\_CallAgentFuncV, [339](#)
- MC\_CallAgentFuncVar, [339](#)
- MC\_ComposeAgent, [339](#)
- MC\_ComposeAgentFromFile, [340](#)
- MC\_ComposeAgentFromFileS, [340](#)
- MC\_ComposeAgentFromFileWithWorkgroup, [341](#)
- MC\_ComposeAgentS, [342](#)
- MC\_ComposeAgentWithWorkgroup, [342](#)
- MC\_CondBroadcast, [343](#)
- MC\_CondReset, [344](#)
- MC\_CondSignal, [344](#)
- MC\_CondWait, [345](#)
- MC\_CopyAgent, [345](#)
- MC\_DeleteAgent, [346](#)
- MC\_DeleteAgentWG, [346](#)
- MC\_DestroyServiceSearchResult, [346](#)
- MC\_End, [347](#)
- MC\_FindAgentByID, [347](#)
- MC\_FindAgentByName, [348](#)
- MC\_GetAgentExecEngine, [348](#)
- MC\_GetAgentID, [348](#)
- MC\_GetAgentName, [348](#)
- MC\_GetAgentNumTasks, [349](#)
- MC\_GetAgentReturnData, [349](#)
- MC\_GetAgentStatus, [349](#)
- MC\_GetAgentType, [350](#)
- MC\_GetAgentXMLString, [350](#)
- MC\_HaltAgency, [350](#)
- MC\_Initialize, [350](#)
- MC\_InitializeAgencyOptions, [351](#)
- MC\_LoadAgentFromFile, [352](#)
- MC\_MainLoop, [352](#)
- MC\_MigrateAgent, [352](#)
- MC\_MutexLock, [352](#)
- MC\_MutexUnlock, [353](#)
- MC\_PrintAgentCode, [353](#)
- MC\_RegisterService, [353](#)
- MC\_ResetSignal, [354](#)
- MC\_ResumeAgency, [354](#)
- MC\_RetrieveAgent, [355](#)
- MC\_RetrieveAgentCode, [355](#)
- MC\_SearchForService, [355](#)
- MC\_SemaphorePost, [356](#)
- MC\_SemaphoreWait, [356](#)
- MC\_SendAgent, [357](#)
- MC\_SendAgentFile, [357](#)
- MC\_SendAgentMigrationMessage, [357](#)
- MC\_SendAgentMigrationMessageFile, [358](#)

- MC\_SetAgentStatus, 358
- MC\_SetDefaultAgentStatus, 359
- MC\_SetThreadOff, 359
- MC\_SetThreadOn, 359
- MC\_SetThreadsAllOff, 360
- MC\_SetThreadsAllOn, 360
- MC\_Signal\_e, 324
- MC\_Steer, 360
- MC\_SteerCommand\_e, 324
- MC\_SteerCommand\_t, 321
- MC\_SteerControl, 361
- MC\_SyncDelete, 361
- MC\_SyncInit, 362
- MC\_TerminateAgent, 362
- MC\_TerminateAgentWG, 362
- MC\_ThreadIndex\_e, 325
- MC\_Wait, 320
- MC\_WaitAgent, 363
- MC\_WaitRetrieveAgent, 363
- MC\_WaitSignal, 363
- MCAgency\_t, 321
- MCAgencyOptions\_t, 321
- MCAgent\_t, 321
- stationary\_agent\_info\_t, 321
- list
  - barrier\_queue\_s, 55
  - syncList\_s, 163
- list.c
  - list\_pGetSize, 462
  - ListAdd, 462
  - ListDelete, 462
  - ListGetHead, 462
  - ListInitialize, 462
  - ListInsert, 463
  - ListPop, 463
  - ListSearch, 463
  - ListTerminate, 463
- list.h
  - DATA, 464
  - list\_p, 465
  - list\_t, 465
  - ListAdd, 465
  - ListDelete, 465
  - ListGetHead, 465
  - ListGetSize, 465
  - ListInitialize, 465
  - ListInsert, 465
  - listNode\_p, 465
  - listNode\_t, 465
  - ListPop, 465
  - ListSearch, 466
  - ListTerminate, 466
  - QUEUE\_TEMPLATE, 464
- list\_p
  - list.h, 465
- list\_pGetSize
  - list.c, 462
- list\_s, 108
  - listhead, 108
  - size, 108
- list\_t
  - list.h, 465
- ListAdd
  - list.c, 462
  - list.h, 465
- ListDelete
  - list.c, 462
  - list.h, 465
- listen\_Thread
  - acc.c, 171
  - acc.h, 220
- listen\_thread\_arg
  - mc\_platform\_s, 113
- ListGetHead
  - list.c, 462
  - list.h, 465
- ListGetSize
  - list.h, 465
- listhead
  - list\_s, 108
- ListInitialize
  - list.c, 462
  - list.h, 465
- ListInsert
  - list.c, 463
  - list.h, 465
- listNode\_p
  - list.h, 465
- listNode\_s, 109
  - next, 109
  - node\_data, 109
- listNode\_t
  - list.h, 465
- ListPop
  - list.c, 463
  - list.h, 465
- ListSearch
  - list.c, 463
  - list.h, 466
- ListTerminate
  - list.c, 463
  - list.h, 466
- lock
  - agent\_s, 47
  - barrier\_node\_s, 54
  - barrier\_queue\_s, 55
  - df.h, 270
  - mc\_rwlock\_s, 117

- syncList\_s, [163](#)
  - syncListNode\_s, [164](#)
- LOCKED
  - mtp\_http.h, [391](#)
- m
  - arc4\_context, [53](#)
- m\_sleep
  - timing.c, [819](#)
  - timing.h, [671](#)
- mac\_dec
  - \_ssl\_context, [15](#)
- mac\_enc
  - \_ssl\_context, [16](#)
- maclen
  - \_ssl\_context, [16](#)
- macro
  - fipa\_comm\_message\_check\_t, [78](#)
- macros.h
  - CH\_DATATYPE\_SIZE, [366](#)
  - CH\_DATATYPE\_STR\_TO\_VAL, [366](#)
  - CH\_DATATYPE\_STRING, [367](#)
  - CH\_DATATYPE\_VALUE\_STRING, [367](#)
  - CH\_STRING\_DATATYPE, [368](#)
  - CHECK\_NULL, [368](#)
  - COND\_BROADCAST, [369](#)
  - COND\_DESTROY, [369](#)
  - COND\_INIT, [369](#)
  - COND\_RESET, [369](#)
  - COND\_SIGNAL, [370](#)
  - COND\_SLEEP, [370](#)
  - COND\_SLEEP\_ACTION, [370](#)
  - COND\_T, [370](#)
  - COND\_WAIT, [370](#)
  - GET\_THREAD\_MODE, [370](#)
  - MUTEX\_DESTROY, [371](#)
  - MUTEX\_INIT, [371](#)
  - MUTEX\_LOCK, [371](#)
  - MUTEX\_NEW, [371](#)
  - MUTEX\_T, [372](#)
  - MUTEX\_UNLOCK, [372](#)
  - PTHREAD\_STACK\_SIZE, [372](#)
  - RWLOCK\_DESTROY, [372](#)
  - RWLOCK\_INIT, [372](#)
  - RWLOCK\_RDLOCK, [372](#)
  - RWLOCK\_RDUNLOCK, [372](#)
  - RWLOCK\_T, [373](#)
  - RWLOCK\_WRLOCK, [373](#)
  - RWLOCK\_WRUNLOCK, [373](#)
  - SEMAPHORE\_DESTROY, [373](#)
  - SEMAPHORE\_INIT, [373](#)
  - SEMAPHORE\_POST, [373](#)
  - SEMAPHORE\_T, [373](#)
  - SEMAPHORE\_WAIT, [373](#)
  - SET\_THREAD\_OFF, [374](#)
  - SET\_THREAD\_ON, [374](#)
  - SIGNAL, [374](#)
  - SLEEP\_QUEUE, [374](#)
  - SLEEP\_RESET, [374](#)
  - SOCKET\_ERROR, [374](#)
  - STRUCT, [375](#)
  - THREAD\_CANCEL, [375](#)
  - THREAD\_CREATE, [375](#)
  - THREAD\_DETACH, [375](#)
  - THREAD\_EXIT, [375](#)
  - THREAD\_JOIN, [375](#)
  - WAKE\_QUEUE, [376](#)
  - WARN, [376](#)
- mail\_queue
  - agent\_mailbox\_s, [43](#)
- mailbox
  - agent\_s, [47](#)
- main
  - aesencrypt2.c, [831](#)
  - benchmark.c, [850](#)
  - dh\_client.c, [836](#)
  - dh\_genprime.c, [837](#)
  - dh\_server.c, [838](#)
  - hello.c, [832](#)
  - mc\_genkey.c, [859](#)
  - md5sum.c, [833](#)
  - mpi\_demo.c, [839](#)
  - mxmldoc.c, [525](#)
  - rsa\_genkey.c, [840](#)
  - rsa\_sign.c, [841](#)
  - rsa\_verify.c, [842](#)
  - selftest.c, [852](#)
  - sha1sum.c, [834](#)
  - sha2sum.c, [835](#)
  - ssl\_client1.c, [844](#)
  - ssl\_client2.c, [846](#)
  - ssl\_server.c, [848](#)
  - ssl\_test.c, [857](#)
  - testmxml.c, [532](#)
- major\_ver
  - \_ssl\_context, [16](#)
- master
  - \_ssl\_session, [22](#)
- max\_bytes
  - options, [150](#)
- max\_connections
  - options, [150](#)
- max\_major\_ver
  - \_ssl\_context, [16](#)
- max\_minor\_ver
  - \_ssl\_context, [16](#)
- max\_pathlen
  - \_x509\_cert, [26](#)

- MAXDATASIZE
  - interface.h, 549
- MC\_AGENT\_ACTIVE
  - libmc.h, 324
- MC\_AGENT\_NEUTRAL
  - libmc.h, 324
- MC\_AGENT\_SUSPENDED
  - libmc.h, 324
- MC\_ALL\_SIGNALS
  - libmc.h, 324
- MC\_ERR
  - libmc.h, 322
  - mc\_error.h, 377
- MC\_ERR\_CONNECT
  - libmc.h, 322
  - mc\_error.h, 377
- MC\_ERR\_EMPTY
  - libmc.h, 322
  - mc\_error.h, 377
- MC\_ERR\_INVALID
  - libmc.h, 322
  - mc\_error.h, 377
- MC\_ERR\_INVALID\_ARGS
  - libmc.h, 322
  - mc\_error.h, 377, 378
- MC\_ERR\_MEMORY
  - libmc.h, 322
  - mc\_error.h, 377, 378
- MC\_ERR\_NOT\_FOUND
  - libmc.h, 322
  - mc\_error.h, 377, 378
- MC\_ERR\_PARSE
  - libmc.h, 322
  - mc\_error.h, 377
- MC\_ERR\_SEND
  - libmc.h, 322
  - mc\_error.h, 377, 378
- mc\_error.h
  - ERR, 377
  - MC\_ERR, 377
  - MC\_ERR\_CONNECT, 377
  - MC\_ERR\_EMPTY, 377
  - MC\_ERR\_INVALID, 377
  - MC\_ERR\_INVALID\_ARGS, 377, 378
  - MC\_ERR\_MEMORY, 377, 378
  - MC\_ERR\_NOT\_FOUND, 377, 378
  - MC\_ERR\_PARSE, 377
  - MC\_ERR\_SEND, 377, 378
  - MC\_SUCCESS, 377
  - MC\_WARN\_DUPLICATE, 377, 378
- MC\_EXEC\_AGENT
  - libmc.h, 324
- MC\_LOCAL\_AGENT
  - libmc.h, 324
- MC\_NO\_SIGNAL
  - libmc.h, 324
- MC\_NO\_STATUS
  - libmc.h, 324
- MC\_NONE
  - libmc.h, 324
- MC\_RECV\_AGENT
  - libmc.h, 324
- MC\_RECV\_CONNECTION
  - libmc.h, 324
- MC\_RECV\_MESSAGE
  - libmc.h, 324
- MC\_RECV\_RETURN
  - libmc.h, 324
- MC\_REMOTE\_AGENT
  - libmc.h, 324
- MC\_RESTART
  - libmc.h, 325
- MC\_RETURN\_AGENT
  - libmc.h, 324
- MC\_RUN
  - libmc.h, 325
- MC\_STOP
  - libmc.h, 325
- MC\_SUCCESS
  - libmc.h, 322
  - mc\_error.h, 377
- MC\_SUSPEND
  - libmc.h, 325
- MC\_THREAD\_ACC
  - libmc.h, 325
- MC\_THREAD\_AGENT
  - libmc.h, 325
- MC\_THREAD\_ALL
  - libmc.h, 325
- MC\_THREAD\_AMS
  - libmc.h, 325
- MC\_THREAD\_CP
  - libmc.h, 325
- MC\_THREAD\_DF
  - libmc.h, 325
- MC\_WAIT\_CH
  - libmc.h, 324
- MC\_WAIT\_FINISHED
  - libmc.h, 324
- MC\_WAIT\_MESSGSEND
  - libmc.h, 324
- MC\_WARN\_DUPLICATE
  - libmc.h, 322
  - mc\_error.h, 377, 378
- MC\_AclAddReceiver
  - libmc.c, 412
  - libmc.h, 325
- MC\_AclAddReceiver\_chdl

- agent\_lib.h, [229](#)
- libmc.c, [412](#)
- MC\_AclAddReplyTo
  - libmc.c, [412](#)
  - libmc.h, [325](#)
- MC\_AclAddReplyTo\_chdl
  - agent\_lib.h, [229](#)
  - libmc.c, [413](#)
- MC\_AclDestroy
  - libmc.c, [413](#)
  - libmc.h, [325](#)
- MC\_AclDestroy\_chdl
  - agent\_lib.h, [229](#)
  - libmc.c, [413](#)
- MC\_AclGetContent
  - libmc.c, [413](#)
  - libmc.h, [326](#)
- MC\_AclGetContent\_chdl
  - agent\_lib.h, [229](#)
  - libmc.c, [413](#)
- MC\_AclGetConversationID
  - libmc.c, [414](#)
  - libmc.h, [326](#)
- MC\_AclGetConversationID\_chdl
  - agent\_lib.h, [230](#)
  - libmc.c, [414](#)
- MC\_AclGetPerformative
  - libmc.c, [414](#)
  - libmc.h, [326](#)
- MC\_AclGetPerformative\_chdl
  - agent\_lib.h, [230](#)
  - libmc.c, [414](#)
- MC\_AclGetProtocol
  - libmc.c, [414](#)
  - libmc.h, [326](#)
- MC\_AclGetProtocol\_chdl
  - agent\_lib.h, [230](#)
  - libmc.c, [414](#)
- MC\_AclGetSender
  - libmc.c, [415](#)
  - libmc.h, [326](#)
- MC\_AclGetSender\_chdl
  - agent\_lib.h, [230](#)
  - libmc.c, [415](#)
- MC\_AclNew
  - libmc.c, [415](#)
  - libmc.h, [327](#)
- MC\_AclNew\_chdl
  - agent\_lib.h, [230](#)
  - libmc.c, [415](#)
- MC\_AclPost
  - libmc.c, [415](#)
  - libmc.h, [327](#)
- MC\_AclPost\_chdl
  - agent\_lib.h, [230](#)
  - libmc.c, [416](#)
- MC\_AclReply
  - libmc.c, [416](#)
  - libmc.h, [327](#)
- MC\_AclReply\_chdl
  - agent\_lib.h, [230](#)
  - libmc.c, [416](#)
- MC\_AclRetrieve
  - libmc.c, [416](#)
  - libmc.h, [328](#)
- MC\_AclRetrieve\_chdl
  - agent\_lib.h, [231](#)
  - libmc.c, [417](#)
- MC\_AclSend
  - libmc.c, [417](#)
  - libmc.h, [328](#)
- MC\_AclSend\_chdl
  - agent\_lib.h, [231](#)
  - libmc.c, [417](#)
- MC\_AclSetContent
  - libmc.c, [417](#)
  - libmc.h, [328](#)
- MC\_AclSetContent\_chdl
  - agent\_lib.h, [231](#)
  - libmc.c, [417](#)
- MC\_AclSetConversationID
  - libmc.c, [418](#)
  - libmc.h, [329](#)
- MC\_AclSetConversationID\_chdl
  - agent\_lib.h, [231](#)
  - libmc.c, [418](#)
- MC\_AclSetPerformative
  - libmc.c, [418](#)
  - libmc.h, [329](#)
- MC\_AclSetPerformative\_chdl
  - agent\_lib.h, [231](#)
  - libmc.c, [418](#)
- MC\_AclSetProtocol
  - libmc.c, [418](#)
  - libmc.h, [329](#)
- MC\_AclSetProtocol\_chdl
  - agent\_lib.h, [231](#)
  - libmc.c, [418](#)
- MC\_AclSetSender
  - libmc.c, [419](#)
  - libmc.h, [329](#)
- MC\_AclSetSender\_chdl
  - agent\_lib.h, [231](#)
  - libmc.c, [419](#)
- MC\_AclWaitRetrieve
  - libmc.c, [419](#)
  - libmc.h, [329](#)
- MC\_AclWaitRetrieve\_chdl
  - agent\_lib.h, [231](#)
  - libmc.c, [419](#)

- agent\_lib.h, 232
- libmc.c, 419
- MC\_AddAgent
  - libmc.c, 419
  - libmc.h, 330
- MC\_AddAgent\_chdl
  - agent\_lib.h, 232
  - libmc.c, 420
- MC\_AddAgentInitCallback
  - libmc.c, 420
  - libmc.h, 330
- MC\_AddStationaryAgent
  - libmc.c, 420
  - libmc.h, 330
- MC\_AgentAddTask
  - libmc.c, 421
  - libmc.h, 331
- MC\_AgentAddTask\_chdl
  - agent\_lib.h, 232
  - libmc.c, 421
- MC\_AgentAddTaskFromFile
  - libmc.c, 421
  - libmc.h, 331
- MC\_AgentInfo\_GetAgency
  - libmc.c, 422
  - libmc.h, 332
- MC\_AgentInfo\_GetAgent
  - libmc.c, 422
  - libmc.h, 332
- MC\_AgentInfo\_GetAgentArgs
  - libmc.c, 422
  - libmc.h, 332
- MC\_AgentInitCallbackFunc\_t
  - libmc.h, 321
- MC\_AgentReturnArrayDim
  - libmc.c, 423
  - libmc.h, 333
- MC\_AgentReturnArrayExtent
  - libmc.c, 423
  - libmc.h, 333
- MC\_AgentReturnArrayNum
  - libmc.c, 423
  - libmc.h, 333
- MC\_AgentReturnDataGetSymbolAddr
  - libmc.c, 424
  - libmc.h, 334
- MC\_AgentReturnDataSize
  - libmc.c, 424
  - libmc.h, 334
- MC\_AgentReturnDataType
  - libmc.c, 424
  - libmc.h, 334
- MC\_AgentReturnIsArray
  - libmc.c, 425
  - libmc.h, 335
- MC\_AgentStatus\_e
  - libmc.h, 323
- MC\_AgentType\_e
  - libmc.h, 324
- MC\_AgentVariableRetrieve
  - libmc.c, 425
  - libmc.h, 335
- MC\_AgentVariableRetrieve\_chdl
  - agent\_lib.h, 232
  - libmc.c, 426
- MC\_AgentVariableRetrieveInfo
  - libmc.c, 426
  - libmc.h, 336
- MC\_AgentVariableSave
  - libmc.c, 426
  - libmc.h, 336
- MC\_AgentVariableSave\_chdl
  - agent\_lib.h, 232
  - libmc.c, 427
- MC\_Barrier
  - libmc.c, 427
  - libmc.h, 337
- MC\_Barrier\_chdl
  - agent\_lib.h, 232
  - libmc.c, 427
- MC\_BarrierDelete
  - libmc.c, 427
  - libmc.h, 337
- MC\_BarrierDelete\_chdl
  - agent\_lib.h, 232
  - libmc.c, 428
- MC\_BarrierInit
  - libmc.c, 428
  - libmc.h, 337
- MC\_BarrierInit\_chdl
  - agent\_lib.h, 233
  - libmc.c, 428
- MC\_CallAgentFunc
  - libmc.c, 428
  - libmc.h, 338
- MC\_CallAgentFunc\_chdl
  - agent\_lib.h, 233
  - libmc.c, 429
- MC\_CallAgentFuncArg
  - libmc.c, 429
  - libmc.h, 338
- MC\_CallAgentFuncV
  - libmc.c, 430
  - libmc.h, 339
- MC\_CallAgentFuncVar
  - libmc.c, 430
  - libmc.h, 339
- MC\_ComposeAgent

- libmc.c, [430](#)
- libmc.h, [339](#)
- MC\_ComposeAgent\_chdl
  - agent\_lib.h, [233](#)
  - libmc.c, [431](#)
- MC\_ComposeAgentFromFile
  - libmc.c, [431](#)
  - libmc.h, [340](#)
- MC\_ComposeAgentFromFile\_chdl
  - agent\_lib.h, [233](#)
  - libmc.c, [431](#)
- MC\_ComposeAgentFromFileS
  - libmc.c, [432](#)
  - libmc.h, [340](#)
- MC\_ComposeAgentFromFileWithWorkgroup
  - libmc.c, [432](#)
  - libmc.h, [341](#)
- MC\_ComposeAgentFromFileWithWorkgroup\_chdl
  - agent\_lib.h, [233](#)
  - libmc.c, [433](#)
- MC\_ComposeAgentS
  - libmc.c, [433](#)
  - libmc.h, [342](#)
- MC\_ComposeAgentWithWorkgroup
  - libmc.c, [434](#)
  - libmc.h, [342](#)
- MC\_ComposeAgentWithWorkgroup\_chdl
  - agent\_lib.h, [233](#)
  - libmc.c, [434](#)
- MC\_CondBroadcast
  - libmc.c, [435](#)
  - libmc.h, [343](#)
- MC\_CondBroadcast\_chdl
  - agent\_lib.h, [233](#)
  - libmc.c, [435](#)
- MC\_CondReset
  - libmc.c, [435](#)
  - libmc.h, [344](#)
- MC\_CondReset\_chdl
  - agent\_lib.h, [234](#)
  - libmc.c, [436](#)
- MC\_CondSignal
  - libmc.c, [436](#)
  - libmc.h, [344](#)
- MC\_CondSignal\_chdl
  - agent\_lib.h, [234](#)
  - libmc.c, [436](#)
- MC\_CondWait
  - libmc.c, [436](#)
  - libmc.h, [345](#)
- MC\_CondWait\_chdl
  - agent\_lib.h, [234](#)
  - libmc.c, [437](#)
- MC\_CopyAgent
  - libmc.c, [437](#)
  - libmc.h, [345](#)
- MC\_DeleteAgent
  - libmc.c, [437](#)
  - libmc.h, [346](#)
- MC\_DeleteAgent\_chdl
  - agent\_lib.h, [234](#)
  - libmc.c, [438](#)
- MC\_DeleteAgentWG
  - libmc.c, [438](#)
  - libmc.h, [346](#)
- MC\_DeleteAgentWG\_chdl
  - agent\_lib.h, [234](#)
  - libmc.c, [438](#)
- MC\_DeregisterService
  - libmc.c, [438](#)
- MC\_DeregisterService\_chdl
  - agent\_lib.h, [234](#)
  - libmc.c, [439](#)
- MC\_DestroyServiceSearchResult
  - libmc.c, [439](#)
  - libmc.h, [346](#)
- MC\_DestroyServiceSearchResult\_chdl
  - agent\_lib.h, [234](#)
  - libmc.c, [439](#)
- MC\_End
  - libmc.c, [439](#)
  - libmc.h, [347](#)
- MC\_End\_chdl
  - agent\_lib.h, [235](#)
  - libmc.c, [440](#)
- mc\_error.h
  - error\_code\_e, [377](#)
  - error\_code\_t, [377](#)
- MC\_FindAgentByID
  - libmc.c, [440](#)
  - libmc.h, [347](#)
- MC\_FindAgentByID\_chdl
  - agent\_lib.h, [235](#)
  - libmc.c, [440](#)
- MC\_FindAgentByName
  - libmc.c, [440](#)
  - libmc.h, [348](#)
- MC\_FindAgentByName\_chdl
  - agent\_lib.h, [235](#)
  - libmc.c, [441](#)
- mc\_FIPAComm\_AddNewProtocol
  - fipa\_comm.h, [308](#)
- mc\_FIPAComm\_DeregisterPerformative
  - fipa\_comm.h, [308](#)
- mc\_FIPAComm\_Destroy
  - fipa\_comm.h, [308](#)
- mc\_FIPAComm\_FSM\_CN
  - fipa\_comm.h, [308](#)



- mc\_FIPAComm\_HandleMessage
  - fipa\_comm.h, 308
- mc\_FIPAComm\_HandleMessageOnce
  - fipa\_comm.h, 308
- mc\_FIPAComm\_HandleMessageWait
  - fipa\_comm.h, 308
- mc\_FIPAComm\_HandleProtocol
  - fipa\_comm.h, 308
- mc\_FIPAComm\_New
  - fipa\_comm.h, 308
- mc\_FIPAComm\_NewProtocol
  - fipa\_comm.h, 308
- mc\_FIPAComm\_RegisterAction
  - fipa\_comm.h, 308
- mc\_FIPAComm\_RegisterActionS
  - fipa\_comm.h, 308
- mc\_FIPAComm\_RegisterPerformative
  - fipa\_comm.h, 308
- mc\_FIPAComm\_RegisterProtocol
  - fipa\_comm.h, 308
- mc\_FIPAComm\_RegisterProtocolCallback
  - fipa\_comm.h, 308
- mc\_FIPAComm\_SetAgent
  - fipa\_comm.h, 308
- mc\_FIPAComm\_SetDefaultReply
  - fipa\_comm.h, 308
- mc\_FIPAComm\_SetPerformativeDefaultReply
  - fipa\_comm.h, 308
- mc\_FIPAComm\_SetProtocolState
  - fipa\_comm.h, 308
- mc\_FIPAComm\_StartProtocol
  - fipa\_comm.h, 308
- mc\_FIPAProtocol\_New
  - fipa\_comm.h, 308
- mc\_genkey.c
  - main, 859
  - print\_usage, 859
- MC\_GetAgentArrivalTime
  - libmc.c, 441
- MC\_GetAgentArrivalTime\_chdl
  - libmc.c, 441
- MC\_GetAgentExecEngine
  - libmc.c, 441
  - libmc.h, 348
- MC\_GetAgentID
  - libmc.c, 441
  - libmc.h, 348
- MC\_GetAgentID\_chdl
  - agent\_lib.h, 235
  - libmc.c, 442
- MC\_GetAgentName
  - libmc.c, 442
  - libmc.h, 348
- MC\_GetAgentName\_chdl
  - agent\_lib.h, 235
  - libmc.c, 442
- MC\_GetAgentNumTasks
  - libmc.c, 442
  - libmc.h, 349
- MC\_GetAgentNumTasks\_chdl
  - agent\_lib.h, 235
  - libmc.c, 442
- MC\_GetAgentReturnData
  - libmc.c, 442
  - libmc.h, 349
- MC\_GetAgentStatus
  - libmc.c, 443
  - libmc.h, 349
- MC\_GetAgentStatus\_chdl
  - agent\_lib.h, 235
  - libmc.c, 443
- MC\_GetAgentType
  - libmc.c, 443
  - libmc.h, 350
- MC\_GetAgentXMLString
  - libmc.c, 444
  - libmc.h, 350
- MC\_GetAgentXMLString\_chdl
  - agent\_lib.h, 236
  - libmc.c, 444
- MC\_GetAllAgents
  - libmc.c, 444
- MC\_GetTimeOfDay\_chdl
  - agent\_lib.h, 236
  - libmc.c, 444
- MC\_HaltAgency
  - libmc.c, 444
  - libmc.h, 350
- MC\_HaltAgency\_chdl
  - agent\_lib.h, 236
  - libmc.c, 445
- MC\_Initialize
  - libmc.c, 445
  - libmc.h, 350
- MC\_InitializeAgencyOptions
  - libmc.c, 445
  - libmc.h, 351
- MC\_LoadAgentFromFile
  - libmc.c, 446
  - libmc.h, 352
- MC\_MainLoop
  - libmc.c, 446
  - libmc.h, 352
- MC\_MigrateAgent
  - libmc.c, 446
  - libmc.h, 352
- MC\_MigrateAgent\_chdl
  - agent\_lib.h, 236

- libmc.c, 447
- MC\_MutexLock
  - libmc.c, 447
  - libmc.h, 352
- MC\_MutexLock\_chdl
  - agent\_lib.h, 236
  - libmc.c, 447
- MC\_MutexUnlock
  - libmc.c, 448
  - libmc.h, 353
- MC\_MutexUnlock\_chdl
  - agent\_lib.h, 236
  - libmc.c, 448
- MC\_NUM\_PERFORMATIVES
  - fipa\_comm.h, 303
- mc\_platform
  - agency\_s, 37
  - agent\_s, 47
  - df.h, 270
  - message\_send\_arg\_s, 130
- mc\_platform.c
  - DEFAULT\_HOSTNAME\_LENGTH, 467
  - mc\_platform\_Destroy, 467
  - mc\_platform\_Initialize, 467
- mc\_platform.h
  - mc\_platform\_Destroy, 380
  - mc\_platform\_Initialize, 380
  - mc\_platform\_p, 379
  - mc\_platform\_t, 379
- mc\_platform\_Destroy
  - mc\_platform.c, 467
  - mc\_platform.h, 380
- mc\_platform\_Initialize
  - mc\_platform.c, 467
  - mc\_platform.h, 380
- mc\_platform\_p
  - mc\_platform.h, 379
- mc\_platform\_s, 110
  - acc, 111
  - agency, 111
  - agent\_queue, 111
  - ams, 111
  - asm\_message\_queue, 111
  - barrier\_queue, 111
  - bluetooth, 111
  - client\_thread\_arg, 111
  - cmd\_prompt, 112
  - connection\_queue, 112
  - default\_agentstatus, 112
  - df, 112
  - err, 112
  - giant, 112
  - giant\_cond, 112
  - giant\_lock, 112
  - hostname, 113
  - initInterps, 113
  - interp\_options, 113
  - interpreter\_queue, 113
  - listen\_thread\_arg, 113
  - MC\_signal, 113
  - MC\_signal\_cond, 113
  - MC\_signal\_lock, 113
  - MC\_steer\_command, 114
  - MC\_steer\_cond, 114
  - MC\_steer\_lock, 114
  - MC\_sync\_cond, 114
  - MC\_sync\_lock, 114
  - message\_queue, 114
  - port, 114
  - private\_key, 115
  - quit, 115
  - quit\_cond, 115
  - quit\_lock, 115
  - sockfd, 115
  - stack\_size, 115
  - syncList, 115
- mc\_platform\_t
  - mc\_platform.h, 379
- MC\_PrintAgentCode
  - libmc.c, 448
  - libmc.h, 353
- MC\_PrintAgentCode\_chdl
  - agent\_lib.h, 236
  - libmc.c, 448
- MC\_RegisterService
  - libmc.c, 448
  - libmc.h, 353
- MC\_RegisterService\_chdl
  - agent\_lib.h, 237
  - libmc.c, 449
- MC\_ResetSignal
  - libmc.c, 449
  - libmc.h, 354
- MC\_ResumeAgency
  - libmc.c, 449
  - libmc.h, 354
- MC\_ResumeAgency\_chdl
  - agent\_lib.h, 237
  - libmc.c, 450
- MC\_RetrieveAgent
  - libmc.c, 450
  - libmc.h, 355
- MC\_RetrieveAgent\_chdl
  - agent\_lib.h, 237
  - libmc.c, 450
- MC\_RetrieveAgentCode
  - libmc.c, 450
  - libmc.h, 355

- MC\_RetrieveAgentCode\_chdl
  - agent\_lib.h, [237](#)
  - libmc.c, [451](#)
- mc\_rwlock.c
  - mc\_rwlock\_destroy, [469](#)
  - mc\_rwlock\_init, [469](#)
  - mc\_rwlock\_rdlock, [469](#)
  - mc\_rwlock\_rdunlock, [469](#)
  - mc\_rwlock\_wrllock, [469](#)
  - mc\_rwlock\_wrunlock, [470](#)
- mc\_rwlock.h
  - mc\_rwlock\_destroy, [381](#)
  - mc\_rwlock\_init, [381](#)
  - mc\_rwlock\_p, [381](#)
  - mc\_rwlock\_rdlock, [381](#)
  - mc\_rwlock\_rdunlock, [382](#)
  - mc\_rwlock\_t, [381](#)
  - mc\_rwlock\_wrllock, [382](#)
  - mc\_rwlock\_wrunlock, [382](#)
- mc\_rwlock\_destroy
  - mc\_rwlock.c, [469](#)
  - mc\_rwlock.h, [381](#)
- mc\_rwlock\_init
  - mc\_rwlock.c, [469](#)
  - mc\_rwlock.h, [381](#)
- mc\_rwlock\_p
  - mc\_rwlock.h, [381](#)
- mc\_rwlock\_rdlock
  - mc\_rwlock.c, [469](#)
  - mc\_rwlock.h, [381](#)
- mc\_rwlock\_rdunlock
  - mc\_rwlock.c, [469](#)
  - mc\_rwlock.h, [382](#)
- mc\_rwlock\_s, [117](#)
  - cond, [117](#)
  - lock, [117](#)
  - num\_readers, [117](#)
  - write\_flag, [117](#)
  - write\_request, [117](#)
- mc\_rwlock\_t
  - mc\_rwlock.h, [381](#)
- mc\_rwlock\_wrllock
  - mc\_rwlock.c, [469](#)
  - mc\_rwlock.h, [382](#)
- mc\_rwlock\_wrunlock
  - mc\_rwlock.c, [470](#)
  - mc\_rwlock.h, [382](#)
- MC\_SaveData\_chdl
  - agent\_lib.h, [237](#)
  - libmc.c, [451](#)
- MC\_SearchForService
  - libmc.c, [451](#)
  - libmc.h, [355](#)
- MC\_SearchForService\_chdl
  - agent\_lib.h, [237](#)
  - libmc.c, [451](#)
- MC\_SemaphorePost
  - libmc.c, [452](#)
  - libmc.h, [356](#)
- MC\_SemaphorePost\_chdl
  - agent\_lib.h, [238](#)
  - libmc.c, [452](#)
- MC\_SemaphoreWait
  - libmc.c, [452](#)
  - libmc.h, [356](#)
- MC\_SemaphoreWait\_chdl
  - agent\_lib.h, [238](#)
  - libmc.c, [453](#)
- MC\_SendAgent
  - libmc.c, [453](#)
  - libmc.h, [357](#)
- MC\_SendAgentFile
  - libmc.c, [453](#)
  - libmc.h, [357](#)
- MC\_SendAgentMigrationMessage
  - libmc.c, [453](#)
  - libmc.h, [357](#)
- MC\_SendAgentMigrationMessage\_chdl
  - agent\_lib.h, [238](#)
  - libmc.c, [454](#)
- MC\_SendAgentMigrationMessageFile
  - libmc.c, [454](#)
  - libmc.h, [358](#)
- MC\_SendAgentMigrationMessageFile\_chdl
  - agent\_lib.h, [238](#)
  - libmc.c, [454](#)
- MC\_SendCh
  - libmc.c, [454](#)
- MC\_SendSteerCommand
  - libmc.c, [454](#)
- MC\_SendSteerCommand\_chdl
  - agent\_lib.h, [238](#)
  - libmc.c, [455](#)
- MC\_SetAgentStatus
  - libmc.c, [455](#)
  - libmc.h, [358](#)
- MC\_SetAgentStatus\_chdl
  - agent\_lib.h, [238](#)
  - libmc.c, [455](#)
- MC\_SetDefaultAgentStatus
  - libmc.c, [455](#)
  - libmc.h, [359](#)
- MC\_SetDefaultAgentStatus\_chdl
  - agent\_lib.h, [238](#)
  - libmc.c, [456](#)
- MC\_SetThreadOff
  - libmc.c, [456](#)
  - libmc.h, [359](#)

- MC\_SetThreadOn
  - libmc.c, [456](#)
  - libmc.h, [359](#)
- MC\_SetThreadsAllOff
  - libmc.c, [457](#)
  - libmc.h, [360](#)
- MC\_SetThreadsAllOn
  - libmc.c, [457](#)
  - libmc.h, [360](#)
- MC\_signal
  - mc\_platform\_s, [113](#)
- MC\_signal\_cond
  - mc\_platform\_s, [113](#)
- MC\_Signal\_e
  - libmc.h, [324](#)
- MC\_signal\_lock
  - mc\_platform\_s, [113](#)
- MC\_Steer
  - libmc.c, [457](#)
  - libmc.h, [360](#)
- MC\_steer\_command
  - mc\_platform\_s, [114](#)
- MC\_steer\_cond
  - mc\_platform\_s, [114](#)
- MC\_steer\_lock
  - mc\_platform\_s, [114](#)
- MC\_SteerCommand\_e
  - libmc.h, [324](#)
- MC\_SteerCommand\_t
  - libmc.h, [321](#)
- MC\_SteerControl
  - libmc.c, [458](#)
  - libmc.h, [361](#)
- MC\_sync\_cond
  - mc\_platform\_s, [114](#)
- MC\_sync\_lock
  - mc\_platform\_s, [114](#)
- MC\_SyncDelete
  - libmc.c, [458](#)
  - libmc.h, [361](#)
- MC\_SyncDelete\_chdl
  - agent\_lib.h, [239](#)
  - libmc.c, [458](#)
- MC\_SyncInit
  - libmc.c, [459](#)
  - libmc.h, [362](#)
- MC\_SyncInit\_chdl
  - agent\_lib.h, [239](#)
  - libmc.c, [459](#)
- MC\_TerminateAgent
  - libmc.c, [459](#)
  - libmc.h, [362](#)
- MC\_TerminateAgent\_chdl
  - agent\_lib.h, [239](#)
- libmc.c, [460](#)
- MC\_TerminateAgentWG
  - libmc.c, [460](#)
  - libmc.h, [362](#)
- MC\_TerminateAgentWG\_chdl
  - agent\_lib.h, [239](#)
  - libmc.c, [460](#)
- MC\_ThreadIndex\_e
  - libmc.h, [325](#)
- MC\_Wait
  - libmc.h, [320](#)
- MC\_WaitAgent
  - libmc.c, [460](#)
  - libmc.h, [363](#)
- MC\_WaitRetrieveAgent
  - libmc.c, [460](#)
  - libmc.h, [363](#)
- MC\_WaitSignal
  - libmc.c, [461](#)
  - libmc.h, [363](#)
- MCAgency\_t
  - libmc.h, [321](#)
- MCAgencyOptions\_s, [119](#)
  - bluetooth, [119](#)
  - ch\_options, [119](#)
  - default\_agent\_status, [119](#)
  - enable\_security, [119](#)
  - initInterps, [119](#)
  - known\_host\_filename, [120](#)
  - modified, [120](#)
  - passphrase, [120](#)
  - priv\_key\_filename, [120](#)
  - stack\_size, [120](#)
  - threads, [120](#)
- MCAgencyOptions\_t
  - libmc.h, [321](#)
- MCAgent\_t
  - agent.h, [223](#)
  - libmc.h, [321](#)
- MCFIPAComm\_p
  - fipa\_comm.h, [304](#)
- MCFIPAComm\_s
  - fipa\_comm.h, [304](#)
- md2
  - md2.h, [599](#)
- md2.h
  - md2, [599](#)
  - md2\_file, [599](#)
  - md2\_finish, [599](#)
  - md2\_hmac, [599](#)
  - md2\_hmac\_finish, [599](#)
  - md2\_hmac\_starts, [600](#)
  - md2\_hmac\_update, [600](#)
  - md2\_self\_test, [600](#)

- md2\_starts, [600](#)
- md2\_update, [600](#)
- md2\_context, [121](#)
  - buffer, [121](#)
  - cksum, [121](#)
  - ipad, [121](#)
  - left, [121](#)
  - opad, [121](#)
  - state, [121](#)
- md2\_file
  - md2.h, [599](#)
- md2\_finish
  - md2.h, [599](#)
- md2\_hmac
  - md2.h, [599](#)
- md2\_hmac\_finish
  - md2.h, [599](#)
- md2\_hmac\_starts
  - md2.h, [600](#)
- md2\_hmac\_update
  - md2.h, [600](#)
- md2\_self\_test
  - md2.h, [600](#)
- md2\_starts
  - md2.h, [600](#)
- md2\_update
  - md2.h, [600](#)
- md4
  - md4.h, [603](#)
- md4.h
  - md4, [603](#)
  - md4\_file, [603](#)
  - md4\_finish, [603](#)
  - md4\_hmac, [603](#)
  - md4\_hmac\_finish, [603](#)
  - md4\_hmac\_starts, [604](#)
  - md4\_hmac\_update, [604](#)
  - md4\_self\_test, [604](#)
  - md4\_starts, [604](#)
  - md4\_update, [604](#)
- md4\_context, [123](#)
  - buffer, [123](#)
  - ipad, [123](#)
  - opad, [123](#)
  - state, [123](#)
  - total, [123](#)
- md4\_file
  - md4.h, [603](#)
- md4\_finish
  - md4.h, [603](#)
- md4\_hmac
  - md4.h, [603](#)
- md4\_hmac\_finish
  - md4.h, [603](#)
- md4\_hmac\_starts
  - md4.h, [604](#)
- md4\_hmac\_update
  - md4.h, [604](#)
- md4\_hmac\_finish
  - md4.h, [603](#)
- md4\_hmac\_update
  - md4.h, [604](#)
- md4\_hmac\_test\_key, [753](#)
- md4\_hmac\_test\_keylen, [753](#)
- md4\_hmac\_test\_sum, [753](#)
- md4\_hmac\_update, [751](#)
- md4\_padding, [754](#)
- md4\_process, [751](#)
- md4\_self\_test, [751](#)
- md4\_starts, [751](#)
- md4\_test\_buf, [754](#)
- md4\_test\_buflen, [754](#)
- md4\_test\_sum, [755](#)
- md4\_update, [752](#)
- P, [748](#)
- PUT\_ULONG\_LE, [748](#)
- S, [749](#)
- md5.h
  - md5, [607](#)
  - md5\_file, [607](#)
  - md5\_finish, [607](#)
  - md5\_hmac, [607](#)
  - md5\_hmac\_finish, [608](#)
  - md5\_hmac\_starts, [608](#)
  - md5\_hmac\_update, [608](#)
  - md5\_self\_test, [609](#)
  - md5\_starts, [609](#)
  - md5\_update, [609](#)
- md5\_check
  - md5sum.c, [833](#)
- md5\_context, [125](#)
  - buffer, [125](#)
- md5\_hmac\_starts
  - md4.h, [604](#)
- md5\_hmac\_update
  - md4.h, [604](#)
- md5\_self\_test
  - md4.h, [604](#)
- md5\_starts
  - md4.h, [604](#)
- md5\_update
  - md4.h, [604](#)
- md5
  - md5.c, [749](#)
  - md5.h, [607](#)
- md5.c
  - F, [748](#)
  - GET\_ULONG\_LE, [748](#)
  - md5, [749](#)
  - md5\_file, [749](#)
  - md5\_finish, [749](#)
  - md5\_hmac, [750](#)
  - md5\_hmac\_finish, [750](#)
  - md5\_hmac\_starts, [750](#)
  - md5\_hmac\_test\_buf, [752](#)
  - md5\_hmac\_test\_buflen, [753](#)
  - md5\_hmac\_test\_key, [753](#)
  - md5\_hmac\_test\_keylen, [753](#)
  - md5\_hmac\_test\_sum, [753](#)
  - md5\_hmac\_update, [751](#)
  - md5\_padding, [754](#)
  - md5\_process, [751](#)
  - md5\_self\_test, [751](#)
  - md5\_starts, [751](#)
  - md5\_test\_buf, [754](#)
  - md5\_test\_buflen, [754](#)
  - md5\_test\_sum, [755](#)
  - md5\_update, [752](#)
  - P, [748](#)
  - PUT\_ULONG\_LE, [748](#)
  - S, [749](#)

- ipad, [125](#)
- opad, [125](#)
- state, [125](#)
- total, [125](#)
- MD5\_CTX
  - openssl.h, [617](#)
- md5\_file
  - md5.c, [749](#)
  - md5.h, [607](#)
- MD5\_Final
  - openssl.h, [617](#)
- md5\_finish
  - md5.c, [749](#)
  - md5.h, [607](#)
- md5\_hmac
  - md5.c, [750](#)
  - md5.h, [607](#)
- md5\_hmac\_finish
  - md5.c, [750](#)
  - md5.h, [608](#)
- md5\_hmac\_starts
  - md5.c, [750](#)
  - md5.h, [608](#)
- md5\_hmac\_test\_buf
  - md5.c, [752](#)
- md5\_hmac\_test\_buflen
  - md5.c, [753](#)
- md5\_hmac\_test\_key
  - md5.c, [753](#)
- md5\_hmac\_test\_keylen
  - md5.c, [753](#)
- md5\_hmac\_test\_sum
  - md5.c, [753](#)
- md5\_hmac\_update
  - md5.c, [751](#)
  - md5.h, [608](#)
- MD5\_Init
  - openssl.h, [618](#)
- md5\_padding
  - md5.c, [754](#)
- md5\_print
  - md5sum.c, [833](#)
- md5\_process
  - md5.c, [751](#)
- md5\_self\_test
  - md5.c, [751](#)
  - md5.h, [609](#)
- md5\_starts
  - md5.c, [751](#)
  - md5.h, [609](#)
- md5\_test\_buf
  - md5.c, [754](#)
- md5\_test\_buflen
  - md5.c, [754](#)
- md5\_test\_sum
  - md5.c, [755](#)
- MD5\_Update
  - openssl.h, [618](#)
- md5\_update
  - md5.c, [752](#)
  - md5.h, [609](#)
- md5\_wrapper
  - md5sum.c, [833](#)
- md5sum.c
  - main, [833](#)
  - md5\_check, [833](#)
  - md5\_print, [833](#)
  - md5\_wrapper, [833](#)
- message
  - dynstring\_s, [66](#)
  - fipa\_message\_string\_s, [91](#)
  - message\_send\_arg\_s, [130](#)
- message.c
  - auth\_rece\_send\_msg, [478](#)
  - http\_to\_hostport, [478](#)
  - message\_Copy, [478](#)
  - message\_Destroy, [479](#)
  - message\_InitializeFromAgent, [479](#)
  - message\_InitializeFromConnection, [479](#)
  - message\_InitializeFromString, [479](#)
  - message\_New, [479](#)
  - message\_Send, [480](#)
  - message\_send\_Thread, [480](#)
  - MSG\_THREAD\_EXIT, [478](#)
  - MSG\_THREADS, [478](#)
  - SOCKET\_INPUT\_SIZE, [478](#)
- message.h
  - AGENT\_UPDATE, [384](#)
  - auth\_rece\_send\_msg, [384](#)
  - CANCEL, [384](#)
  - ENCRYPTED\_DATA, [384](#)
  - ENCRYPTION\_INITIALIZE, [384](#)
  - FIPA\_ACL, [384](#)
  - http\_to\_hostport, [384](#)
  - message\_Copy, [385](#)
  - message\_Destroy, [385](#)
  - message\_InitializeFromAgent, [385](#)
  - message\_InitializeFromConnection, [385](#)
  - message\_InitializeFromString, [385](#)
  - message\_New, [385](#)
  - message\_p, [384](#)
  - message\_Send, [385](#)
  - message\_send\_arg\_t, [384](#)
  - message\_send\_Thread, [386](#)
  - message\_t, [384](#)
  - message\_type\_e, [384](#)
  - message\_type\_t, [384](#)
  - MOBILE\_AGENT, [384](#)

- MSG\_SEC, 384
- N\_UNDRSTD, 384
- NUM\_MESSAGE\_TYPE, 384
- QUER\_IF, 384
- QUER\_REF, 384
- RELAY, 384
- REQUEST, 384
- REQUEST\_ENCRYPTION\_INITIALIZE, 384
- RETURN\_MSG, 384
- SUBSCRIBE, 384
- message\_body
  - message\_s, 128
- message\_Copy
  - message.c, 478
  - message.h, 385
- message\_Destroy
  - message.c, 479
  - message.h, 385
- message\_id
  - message\_s, 128
- message\_InitializeFromAgent
  - message.c, 479
  - message.h, 385
- message\_InitializeFromConnection
  - message.c, 479
  - message.h, 385
- message\_InitializeFromString
  - message.c, 479
  - message.h, 385
- message\_New
  - message.c, 479
  - message.h, 385
- message\_p
  - message.h, 384
- message\_parts
  - mtp\_http\_s, 136
- message\_queue
  - mc\_platform\_s, 114
- message\_queue.c
  - message\_queue\_SendOutgoing, 481
- message\_queue\_SendOutgoing
  - message\_queue.c, 481
- message\_s, 127
  - addr, 127
  - agent\_xml\_flag, 127
  - connect\_id, 127
  - from\_address, 127
  - http\_type, 128
  - isHTTP, 128
  - message\_body, 128
  - message\_id, 128
  - message\_type, 128
  - sending\_agent\_name, 128
  - target, 128
  - to\_address, 129
  - update\_name, 129
  - update\_num, 129
  - xml\_payload, 129
  - xml\_root, 129
- message\_Send
  - message.c, 480
  - message.h, 385
- message\_send\_arg\_s, 130
  - mc\_platform, 130
  - message, 130
  - privatekey, 130
- message\_send\_arg\_t
  - message.h, 384
- message\_send\_Thread
  - message.c, 480
  - message.h, 386
- message\_t
  - message.h, 384
- message\_type
  - message\_s, 128
- message\_type\_e
  - message.h, 384
- message\_type\_t
  - message.h, 384
- message\_xml\_parse
  - xml\_parser.c, 869
  - xml\_parser.h, 403
- message\_xml\_parse\_\_message
  - xml\_parser.c, 869
  - xml\_parser.h, 403
- METHOD\_NOT\_ALLOWED
  - mtp\_http.h, 390
- millisecond
  - fipa\_DateTime\_s, 87
- mime\_version
  - mtp\_http\_s, 136
- min
  - \_x509\_time, 33
- minlen
  - \_ssl\_context, 16
- minor\_ver
  - \_ssl\_context, 16
- minute
  - fipa\_DateTime\_s, 87
- MISC
  - df.h, 267
- MOBILE\_AGENT
  - message.h, 384
- mode
  - des3\_context, 62
  - des\_context, 63
- MODE\_DECRYPT

- aesencrypt2.c, 830
  - interface.h, 549
- MODE\_ENCRYPT
  - aesencrypt2.c, 830
  - interface.h, 549
- modified
  - MCAgencyOptions\_s, 120
- mon
  - \_x509\_time, 33
- month
  - fipa\_DateTime\_s, 87
- mpi, 131
  - n, 131
  - p, 131
  - s, 131
- mpi\_add\_abs
  - bignum.c, 706
  - bignum.h, 567
- mpi\_add\_int
  - bignum.c, 706
  - bignum.h, 567
- mpi\_add\_mpi
  - bignum.c, 707
  - bignum.h, 567
- MPI\_CHK
  - bignum.h, 566
- mpi\_cmp\_abs
  - bignum.c, 707
  - bignum.h, 567
- mpi\_cmp\_int
  - bignum.c, 707
  - bignum.h, 568
- mpi\_cmp\_mpi
  - bignum.c, 707
  - bignum.h, 568
- mpi\_copy
  - bignum.c, 708
  - bignum.h, 568
- mpi\_demo.c
  - main, 839
- mpi\_div\_int
  - bignum.c, 708
  - bignum.h, 568
- mpi\_div\_mpi
  - bignum.c, 708
  - bignum.h, 569
- mpi\_exp\_mod
  - bignum.c, 709
  - bignum.h, 569
- mpi\_free
  - bignum.c, 709
  - bignum.h, 569
- mpi\_gcd
  - bignum.c, 709
- bignum.h, 570
- mpi\_gen\_prime
  - bignum.c, 709
  - bignum.h, 570
- mpi\_get\_digit
  - bignum.c, 710
- mpi\_grow
  - bignum.c, 710
  - bignum.h, 570
- mpi\_init
  - bignum.c, 710
  - bignum.h, 571
- mpi\_inv\_mod
  - bignum.c, 711
  - bignum.h, 571
- mpi\_is\_prime
  - bignum.c, 711
  - bignum.h, 571
- mpi\_lsb
  - bignum.c, 711
  - bignum.h, 571
- mpi\_lset
  - bignum.c, 711
  - bignum.h, 572
- mpi\_mod\_int
  - bignum.c, 712
  - bignum.h, 572
- mpi\_mod\_mpi
  - bignum.c, 712
  - bignum.h, 572
- mpi\_montg\_init
  - bignum.c, 712
- mpi\_montmul
  - bignum.c, 712
- mpi\_montred
  - bignum.c, 712
- mpi\_msb
  - bignum.c, 713
  - bignum.h, 572
- mpi\_mul\_hlp
  - bignum.c, 713
- mpi\_mul\_int
  - bignum.c, 713
  - bignum.h, 573
- mpi\_mul\_mpi
  - bignum.c, 713
  - bignum.h, 573
- mpi\_read\_binary
  - bignum.c, 713
  - bignum.h, 573
- mpi\_read\_file
  - bignum.c, 714
  - bignum.h, 574
- mpi\_read\_mystring



- bignum.c, [714](#)
- bignum.h, [574](#)
- mpi\_read\_string
  - bignum.c, [714](#)
  - bignum.h, [574](#)
- mpi\_self\_test
  - bignum.c, [715](#)
  - bignum.h, [574](#)
- mpi\_shift\_l
  - bignum.c, [715](#)
  - bignum.h, [575](#)
- mpi\_shift\_r
  - bignum.c, [715](#)
  - bignum.h, [575](#)
- mpi\_size
  - bignum.c, [715](#)
  - bignum.h, [575](#)
- mpi\_sub\_abs
  - bignum.c, [716](#)
  - bignum.h, [575](#)
- mpi\_sub\_hlp
  - bignum.c, [716](#)
- mpi\_sub\_int
  - bignum.c, [716](#)
  - bignum.h, [576](#)
- mpi\_sub\_mpi
  - bignum.c, [716](#)
  - bignum.h, [576](#)
- mpi\_swap
  - bignum.c, [716](#)
  - bignum.h, [576](#)
- mpi\_write\_binary
  - bignum.c, [717](#)
  - bignum.h, [576](#)
- mpi\_write\_file
  - bignum.c, [717](#)
  - bignum.h, [577](#)
- mpi\_write\_hlp
  - bignum.c, [717](#)
- mpi\_write\_string
  - bignum.c, [718](#)
  - bignum.h, [577](#)
- MSG\_SEC
  - message.h, [384](#)
- MSG\_THREAD\_EXIT
  - message.c, [478](#)
- MSG\_THREADS
  - message.c, [478](#)
- mtp\_http.h
  - ACCEPTED, [390](#)
  - BAD\_REQUEST, [390](#)
  - CONFLICT, [390](#)
  - CONTINUE, [390](#)
  - CREATED, [390](#)
  - EXPECTATION\_FAILED, [391](#)
  - FAILED\_DEPENDANCY, [391](#)
  - FORBIDDEN, [390](#)
  - GONE, [390](#)
  - HTTP\_CONNECT, [390](#)
  - HTTP\_DELETE, [390](#)
  - HTTP\_GET, [389](#)
  - HTTP\_HEAD, [389](#)
  - HTTP\_NUM\_PERFORMATIVES, [390](#)
  - HTTP\_OPTIONS, [390](#)
  - HTTP\_PERFORMATIVE\_UNDEF, [389](#)
  - HTTP\_PERFORMATIVE\_ZERO, [389](#)
  - HTTP\_POST, [389](#)
  - HTTP\_PUT, [390](#)
  - HTTP\_RESPONSE, [390](#)
  - HTTP\_TRACE, [390](#)
  - LENGTH\_REQUIRED, [390](#)
  - LOCKED, [391](#)
  - METHOD\_NOT\_ALLOWED, [390](#)
  - MULTI\_STATUS, [390](#)
  - NO\_CONTENT, [390](#)
  - NON\_AUTHORITATIVE\_INFORMATION, [390](#)
  - NOT\_ACCEPTABLE, [390](#)
  - NOT\_FOUND, [390](#)
  - OK, [390](#)
  - PARTIAL\_CONTENT, [390](#)
  - PAYMENT\_REQUIRED, [390](#)
  - PRECONDITION\_FAILED, [390](#)
  - PROCESSING, [390](#)
  - PROXY\_AUTHENTICATION\_REQUIRED, [390](#)
  - REQUEST\_TIMEOUT, [390](#)
  - REQUEST\_URI\_TOO\_LONG, [390](#)
  - REQUESTED\_RANGE\_NOT\_SATISFIABLE, [391](#)
  - REQUEST\_ENTITY\_TOO\_LARGE, [390](#)
  - RESET\_CONTENT, [390](#)
  - RETRY\_WITH, [391](#)
  - SWITCHING\_PROTOCOLS, [390](#)
  - UNAUTHORIZED, [390](#)
  - UNORDERED\_COLLECTION, [391](#)
  - UNPROCESSABLE\_ENTITY, [391](#)
  - UNSUPPORTED\_MEDIA\_TYPE, [390](#)
  - UPGRADE\_REQUIRED, [391](#)
- mtp\_http.c
  - http\_GetExpression, [483](#)
  - http\_GetToken, [483](#)
  - HTTP\_PARSE\_EXPR, [483](#)
  - http\_ParseExpression, [484](#)
  - http\_ParseRequest, [484](#)
  - mtp\_http\_ComposeMessage, [484](#)
  - mtp\_http\_CreateMessage, [484](#)
  - mtp\_http\_Destroy, [485](#)

- mtp\_http\_InitializeFromConnection, 485
- mtp\_http\_New, 485
- mtp\_http\_Parse, 485
- mtp\_http\_ParseHeader, 485
- mtp\_http\_ParseResponse, 486
- rece\_de\_msg, 486
- SAFE\_FREE, 483
- mtp\_http.h
  - http\_GetExpression, 391
  - http\_GetToken, 391
  - http\_ParseExpression, 391
  - http\_ParseRequest, 392
  - http\_performative\_e, 389
  - http\_status\_code\_e, 390
  - mtp\_http\_ComposeMessage, 392
  - mtp\_http\_content\_t, 389
  - mtp\_http\_CreateMessage, 392
  - mtp\_http\_Destroy, 392
  - mtp\_http\_InitializeFromConnection, 392
  - mtp\_http\_New, 392
  - mtp\_http\_p, 389
  - mtp\_http\_Parse, 393
  - mtp\_http\_ParseHeader, 393
  - mtp\_http\_t, 389
  - SOCKET\_INPUT\_SIZE, 389
- mtp\_http\_ComposeMessage
  - mtp\_http.c, 484
  - mtp\_http.h, 392
- mtp\_http\_content\_s, 133
  - content\_type, 133
  - data, 133
- mtp\_http\_content\_t
  - mtp\_http.h, 389
- mtp\_http\_CreateMessage
  - mtp\_http.c, 484
  - mtp\_http.h, 392
- mtp\_http\_Destroy
  - mtp\_http.c, 485
  - mtp\_http.h, 392
- mtp\_http\_InitializeFromConnection
  - mtp\_http.c, 485
  - mtp\_http.h, 392
- mtp\_http\_New
  - mtp\_http.c, 485
  - mtp\_http.h, 392
- mtp\_http\_p
  - mtp\_http.h, 389
- mtp\_http\_Parse
  - mtp\_http.c, 485
  - mtp\_http.h, 393
- mtp\_http\_ParseHeader
  - mtp\_http.c, 485
  - mtp\_http.h, 393
- mtp\_http\_ParseResponse
  - mtp\_http.c, 486
- mtp\_http\_s, 134
  - accept\_ranges, 134
  - boundary, 134
  - cache\_control, 134
  - connection, 134
  - content, 135
  - content\_length, 135
  - content\_type, 135
  - date, 135
  - header\_length, 135
  - host, 135
  - http\_performative, 135
  - http\_status\_code, 135
  - http\_version, 136
  - message\_parts, 136
  - mime\_version, 136
  - response\_code, 136
  - response\_string, 136
  - return\_code, 136
  - server, 136
  - target, 136
  - user\_agent, 137
- mtp\_http\_t
  - mtp\_http.h, 389
- MUL
  - aes.c, 689
- MULADDC\_CORE
  - bn\_mul.h, 579
- MULADDC\_INIT
  - bn\_mul.h, 579
- MULADDC\_STOP
  - bn\_mul.h, 579
- MULTI\_STATUS
  - mtp\_http.h, 390
- MUTEX\_DESTROY
  - macros.h, 371
- MUTEX\_INIT
  - macros.h, 371
- MUTEX\_LOCK
  - macros.h, 371
- MUTEX\_NEW
  - macros.h, 371
- MUTEX\_T
  - macros.h, 372
- MUTEX\_UNLOCK
  - macros.h, 372
- mxml-2.2.2/config.h
  - HAVE\_STRDUP, 487
  - HAVE\_VSNPRINTF, 487
  - mxml\_strdupf, 487
  - MXML\_VERSION, 487
- mxml-2.2.2/winconfig.h
  - HAVE\_STRDUP, 533

- HAVE\_VSNPRINTF, 533
- mxml\_strdup, 534
- mxml\_strdupf, 534
- MXML\_VERSION, 533
- mxml\_vsnprintf, 534
- strdup, 533
- vsnprintf, 534
- mxml-attr.c
  - mxmlElementGetAttr, 491
  - mxmlElementSetAttr, 491
- mxml-entity.c
  - callbacks, 493
  - default\_callback, 492
  - mxmlEntityAddCallback, 492
  - mxmlEntityGetName, 492
  - mxmlEntityGetValue, 492
  - mxmlEntityRemoveCallback, 492
  - num\_callbacks, 493
- mxml-file.c
  - ENCODE\_UTF16BE, 495
  - ENCODE\_UTF16LE, 495
  - ENCODE\_UTF8, 495
  - mxml\_add\_char, 495
  - mxml\_bad\_char, 495
  - mxml\_custom\_load\_cb, 500
  - mxml\_custom\_save\_cb, 500
  - mxml\_error\_cb, 500
  - mxml\_fd\_getc, 496
  - mxml\_fd\_putc, 496
  - mxml\_fd\_read, 496
  - mxml\_fd\_write, 496
  - mxml\_fdbuf\_t, 495
  - mxml\_file\_getc, 496
  - mxml\_file\_putc, 496
  - mxml\_get\_entity, 496
  - mxml\_load\_data, 497
  - mxml\_parse\_element, 497
  - mxml\_string\_getc, 497
  - mxml\_string\_putc, 497
  - mxml\_write\_name, 497
  - mxml\_write\_node, 497
  - mxml\_write\_string, 498
  - mxml\_write\_ws, 498
  - mxmlLoadFd, 498
  - mxmlLoadFile, 498
  - mxmlLoadString, 498
  - mxmlSaveAllocString, 499
  - mxmlSaveFd, 499
  - mxmlSaveFile, 499
  - mxmlSaveString, 499
  - mxmlSetCustomHandlers, 499
  - mxmlSetErrorCallback, 499
- mxml-index.c
  - index\_compare, 501
  - index\_find, 501
  - index\_sort, 501
  - mxmlIndexDelete, 501
  - mxmlIndexEnum, 502
  - mxmlIndexFind, 502
  - mxmlIndexNew, 502
  - mxmlIndexReset, 502
- mxml-node.c
  - mxml\_new, 503
  - mxmlAdd, 503
  - mxmlDelete, 503
  - mxmlNewCustom, 504
  - mxmlNewElement, 504
  - mxmlNewInteger, 504
  - mxmlNewOpaque, 504
  - mxmlNewReal, 504
  - mxmlNewText, 505
  - mxmlNewTextf, 505
  - mxmlRemove, 505
- mxml-private.c
  - mxml\_error, 506
  - mxml\_error\_cb, 507
  - mxml\_integer\_cb, 506
  - mxml\_opaque\_cb, 506
  - mxml\_real\_cb, 506
- mxml-search.c
  - mxmlFindElement, 508
  - mxmlWalkNext, 508
  - mxmlWalkPrev, 508
- mxml-set.c
  - mxmlSetCustom, 509
  - mxmlSetElement, 509
  - mxmlSetInteger, 509
  - mxmlSetOpaque, 509
  - mxmlSetReal, 509
  - mxmlSetText, 509
  - mxmlSetTextf, 510
- mxml-string.c
  - mxml\_strdup, 511
  - mxml\_strdupf, 511
  - mxml\_vsnprintf, 511
- mxml.h
  - MXML\_CUSTOM, 517
  - MXML\_ELEMENT, 517
  - MXML\_INTEGER, 517
  - MXML\_OPAQUE, 517
  - MXML\_REAL, 517
  - MXML\_TEXT, 517
  - MXML\_ADD\_AFTER, 514
  - MXML\_ADD\_BEFORE, 514
  - MXML\_ADD\_TO\_PARENT, 514
  - mxml\_attr\_t, 516
  - mxml\_custom\_load\_cb\_t, 516
  - mxml\_custom\_save\_cb\_t, 516

- mxml\_custom\_t, 516
- MXML\_DESCEND, 514
- MXML\_DESCEND\_FIRST, 514
- mxml\_element\_t, 517
- mxml\_error, 517
- mxml\_index\_t, 517
- MXML\_INTEGER\_CALLBACK, 515
- mxml\_integer\_cb, 517
- MXML\_NO\_CALLBACK, 515
- MXML\_NO\_DESCEND, 515
- MXML\_NO\_PARENT, 515
- mxml\_node\_t, 517
- MXML\_OPAQUE\_CALLBACK, 515
- mxml\_opaque\_cb, 517
- MXML\_REAL\_CALLBACK, 515
- mxml\_real\_cb, 517
- MXML\_TAB, 515
- MXML\_TEXT\_CALLBACK, 516
- mxml\_text\_t, 517
- mxml\_type\_e, 517
- mxml\_type\_t, 517
- mxml\_value\_t, 517
- MXML\_WRAP, 516
- MXML\_WS\_AFTER\_CLOSE, 516
- MXML\_WS\_AFTER\_OPEN, 516
- MXML\_WS\_BEFORE\_CLOSE, 516
- MXML\_WS\_BEFORE\_OPEN, 516
- mxmlAdd, 518
- mxmlDelete, 518
- mxmlElementGetAttr, 518
- mxmlElementSetAttr, 518
- mxmlEntityAddCallback, 519
- mxmlEntityGetName, 519
- mxmlEntityGetValue, 519
- mxmlEntityRemoveCallback, 519
- mxmlFindElement, 519
- mxmlIndexDelete, 519
- mxmlIndexEnum, 519
- mxmlIndexFind, 520
- mxmlIndexNew, 520
- mxmlIndexReset, 520
- mxmlLoadFd, 520
- mxmlLoadFile, 520
- mxmlLoadString, 520
- mxmlNewCustom, 520
- mxmlNewElement, 520
- mxmlNewInteger, 521
- mxmlNewOpaque, 521
- mxmlNewReal, 521
- mxmlNewText, 521
- mxmlNewTextf, 521
- mxmlRemove, 522
- mxmlSaveAllocString, 522
- mxmlSaveFd, 522
- mxmlSaveFile, 522
- mxmlSaveString, 522
- mxmlSetCustom, 522
- mxmlSetCustomHandlers, 522
- mxmlSetElement, 522
- mxmlSetErrorCallback, 522
- mxmlSetInteger, 522
- mxmlSetOpaque, 523
- mxmlSetReal, 523
- mxmlSetText, 523
- mxmlSetTextf, 523
- mxmlWalkNext, 523
- mxmlWalkPrev, 523
- MXML\_CUSTOM
  - mxml.h, 517
- MXML\_ELEMENT
  - mxml.h, 517
- MXML\_INTEGER
  - mxml.h, 517
- MXML\_OPAQUE
  - mxml.h, 517
- MXML\_REAL
  - mxml.h, 517
- MXML\_TEXT
  - mxml.h, 517
- MXML\_ADD\_AFTER
  - mxml.h, 514
- MXML\_ADD\_BEFORE
  - mxml.h, 514
- mxml\_add\_char
  - mxml-file.c, 495
- MXML\_ADD\_TO\_PARENT
  - mxml.h, 514
- mxml\_attr\_s, 138
  - name, 138
  - value, 138
- mxml\_attr\_t
  - mxml.h, 516
- mxml\_bad\_char
  - mxml-file.c, 495
- mxml\_custom\_load\_cb
  - mxml-file.c, 500
- mxml\_custom\_load\_cb\_t
  - mxml.h, 516
- mxml\_custom\_s, 139
  - data, 139
  - destroy, 139
- mxml\_custom\_save\_cb
  - mxml-file.c, 500
- mxml\_custom\_save\_cb\_t
  - mxml.h, 516
- mxml\_custom\_t
  - mxml.h, 516
- MXML\_DESCEND

- mxml.h, 514
- MXML\_DESCEND\_FIRST
  - mxml.h, 514
- mxml\_element\_t
  - mxml.h, 517
- mxml\_error
  - mxml-private.c, 506
  - mxml.h, 517
- mxml\_error\_cb
  - mxml-file.c, 500
  - mxml-private.c, 507
- mxml\_fd\_getc
  - mxml-file.c, 496
- mxml\_fd\_putc
  - mxml-file.c, 496
- mxml\_fd\_read
  - mxml-file.c, 496
- mxml\_fd\_write
  - mxml-file.c, 496
- mxml\_fdbuf\_s, 140
  - buffer, 140
  - current, 140
  - end, 140
  - fd, 140
- mxml\_fdbuf\_t
  - mxml-file.c, 495
- mxml\_file\_getc
  - mxml-file.c, 496
- mxml\_file\_putc
  - mxml-file.c, 496
- mxml\_get\_entity
  - mxml-file.c, 496
- mxml\_index\_s, 141
  - alloc\_nodes, 141
  - attr, 141
  - cur\_node, 141
  - nodes, 141
  - num\_nodes, 141
- mxml\_index\_t
  - mxml.h, 517
- MXML\_INTEGER\_CALLBACK
  - mxml.h, 515
- mxml\_integer\_cb
  - mxml-private.c, 506
  - mxml.h, 517
- mxml\_load\_data
  - mxml-file.c, 497
- mxml\_new
  - mxml-node.c, 503
- MXML\_NO\_CALLBACK
  - mxml.h, 515
- MXML\_NO\_DESCEND
  - mxml.h, 515
- MXML\_NO\_PARENT
  - mxml.h, 515
- mxml\_node\_s, 143
  - child, 143
  - last\_child, 143
  - next, 143
  - parent, 143
  - prev, 143
  - type, 144
  - value, 144
- mxml\_node\_t
  - mxml.h, 517
- MXML\_OPAQUE\_CALLBACK
  - mxml.h, 515
- mxml\_opaque\_cb
  - mxml-private.c, 506
  - mxml.h, 517
- mxml\_parse\_element
  - mxml-file.c, 497
- MXML\_REAL\_CALLBACK
  - mxml.h, 515
- mxml\_real\_cb
  - mxml-private.c, 506
  - mxml.h, 517
- mxml\_strdup
  - mxml-2.2.2/winconfig.h, 534
  - mxml-string.c, 511
- mxml\_strdupf
  - mxml-2.2.2/config.h, 487
  - mxml-2.2.2/winconfig.h, 534
  - mxml-string.c, 511
- mxml\_string\_getc
  - mxml-file.c, 497
- mxml\_string\_putc
  - mxml-file.c, 497
- MXML\_TAB
  - mxml.h, 515
- MXML\_TEXT\_CALLBACK
  - mxml.h, 516
- mxml\_text\_s, 145
  - string, 145
  - whitespace, 145
- mxml\_text\_t
  - mxml.h, 517
- mxml\_type\_e
  - mxml.h, 517
- mxml\_type\_t
  - mxml.h, 517
- mxml\_value\_s, 146
  - attrs, 146
  - name, 146
  - num\_attrs, 146
- mxml\_value\_t
  - mxml.h, 517
- mxml\_value\_u, 147

- custom, [147](#)
- element, [147](#)
- integer, [147](#)
- opaque, [147](#)
- real, [147](#)
- text, [147](#)
- MXML\_VERSION
  - mxml-2.2.2/config.h, [487](#)
  - mxml-2.2.2/winconfig.h, [533](#)
- mxml\_vsnprintf
  - mxml-2.2.2/winconfig.h, [534](#)
  - mxml-string.c, [511](#)
- MXML\_WRAP
  - mxml.h, [516](#)
- mxml\_write\_name
  - mxml-file.c, [497](#)
- mxml\_write\_node
  - mxml-file.c, [497](#)
- mxml\_write\_string
  - mxml-file.c, [498](#)
- mxml\_write\_ws
  - mxml-file.c, [498](#)
- MXML\_WS\_AFTER\_CLOSE
  - mxml.h, [516](#)
- MXML\_WS\_AFTER\_OPEN
  - mxml.h, [516](#)
- MXML\_WS\_BEFORE\_CLOSE
  - mxml.h, [516](#)
- MXML\_WS\_BEFORE\_OPEN
  - mxml.h, [516](#)
- mxmlAdd
  - mxml-node.c, [503](#)
  - mxml.h, [518](#)
- mxmlDelete
  - mxml-node.c, [503](#)
  - mxml.h, [518](#)
- mxmldoc.c
  - add\_variable, [525](#)
  - main, [525](#)
  - safe\_strcpy, [525](#)
  - scan\_file, [525](#)
  - sort\_node, [526](#)
  - STATE\_C\_COMMENT, [524](#)
  - STATE\_CHARACTER, [524](#)
  - STATE\_CXX\_COMMENT, [524](#)
  - STATE\_IDENTIFIER, [524](#)
  - STATE\_NONE, [525](#)
  - STATE\_PREPROCESSOR, [525](#)
  - STATE\_STRING, [525](#)
  - update\_comment, [526](#)
  - write\_documentation, [526](#)
  - write\_element, [526](#)
  - write\_string, [526](#)
  - ws\_cb, [527](#)
- mxmlElementGetAttr
  - mxml-attr.c, [491](#)
  - mxml.h, [518](#)
- mxmlElementSetAttr
  - mxml-attr.c, [491](#)
  - mxml.h, [518](#)
- mxmlEntityAddCallback
  - mxml-entity.c, [492](#)
  - mxml.h, [519](#)
- mxmlEntityGetName
  - mxml-entity.c, [492](#)
  - mxml.h, [519](#)
- mxmlEntityGetValue
  - mxml-entity.c, [492](#)
  - mxml.h, [519](#)
- mxmlEntityRemoveCallback
  - mxml-entity.c, [492](#)
  - mxml.h, [519](#)
- mxmlFindElement
  - mxml-search.c, [508](#)
  - mxml.h, [519](#)
- mxmlIndexDelete
  - mxml-index.c, [501](#)
  - mxml.h, [519](#)
- mxmlIndexEnum
  - mxml-index.c, [502](#)
  - mxml.h, [519](#)
- mxmlIndexFind
  - mxml-index.c, [502](#)
  - mxml.h, [520](#)
- mxmlIndexNew
  - mxml-index.c, [502](#)
  - mxml.h, [520](#)
- mxmlIndexReset
  - mxml-index.c, [502](#)
  - mxml.h, [520](#)
- mxmlLoadFd
  - mxml-file.c, [498](#)
  - mxml.h, [520](#)
- mxmlLoadFile
  - mxml-file.c, [498](#)
  - mxml.h, [520](#)
- mxmlLoadString
  - mxml-file.c, [498](#)
  - mxml.h, [520](#)
- mxmlNewCustom
  - mxml-node.c, [504](#)
  - mxml.h, [520](#)
- mxmlNewElement
  - mxml-node.c, [504](#)
  - mxml.h, [520](#)
- mxmlNewInteger
  - mxml-node.c, [504](#)
  - mxml.h, [521](#)

- mxmxmlNewOpaque
  - mxmxml-node.c, 504
  - mxmxml.h, 521
- mxmxmlNewReal
  - mxmxml-node.c, 504
  - mxmxml.h, 521
- mxmxmlNewText
  - mxmxml-node.c, 505
  - mxmxml.h, 521
- mxmxmlNewTextf
  - mxmxml-node.c, 505
  - mxmxml.h, 521
- mxmxmlRemove
  - mxmxml-node.c, 505
  - mxmxml.h, 522
- mxmxmlSaveAllocString
  - mxmxml-file.c, 499
  - mxmxml.h, 522
- mxmxmlSaveFd
  - mxmxml-file.c, 499
  - mxmxml.h, 522
- mxmxmlSaveFile
  - mxmxml-file.c, 499
  - mxmxml.h, 522
- mxmxmlSaveString
  - mxmxml-file.c, 499
  - mxmxml.h, 522
- mxmxmlSetCustom
  - mxmxml-set.c, 509
  - mxmxml.h, 522
- mxmxmlSetCustomHandlers
  - mxmxml-file.c, 499
  - mxmxml.h, 522
- mxmxmlSetElement
  - mxmxml-set.c, 509
  - mxmxml.h, 522
- mxmxmlSetErrorCallback
  - mxmxml-file.c, 499
  - mxmxml.h, 522
- mxmxmlSetInteger
  - mxmxml-set.c, 509
  - mxmxml.h, 522
- mxmxmlSetOpaque
  - mxmxml-set.c, 509
  - mxmxml.h, 523
- mxmxmlSetReal
  - mxmxml-set.c, 509
  - mxmxml.h, 523
- mxmxmlSetText
  - mxmxml-set.c, 509
  - mxmxml.h, 523
- mxmxmlSetTextf
  - mxmxml-set.c, 510
  - mxmxml.h, 523
- mxmxmlWalkNext
  - mxmxml-search.c, 508
  - mxmxml.h, 523
- mxmxmlWalkPrev
  - mxmxml-search.c, 508
  - mxmxml.h, 523
- my\_ciphers
  - ssl\_server.c, 848
- my\_debug
  - ssl\_client1.c, 844
  - ssl\_client2.c, 846
  - ssl\_server.c, 848
  - ssl\_test.c, 857
- my\_dhm\_G
  - ssl\_server.c, 849
- my\_dhm\_P
  - ssl\_server.c, 849
- my\_get\_session
  - ssl\_server.c, 848
- my\_set\_session
  - ssl\_server.c, 848
- myrand
  - benchmark.c, 851
- N
  - rsa\_context, 153
- n
  - mpi, 131
- N\_UNDRSTD
  - message.h, 384
- name
  - agent\_s, 47
  - fipa\_agent\_identifier\_s, 75
  - interpreter\_variable\_data\_s, 107
  - mxmxml\_attr\_s, 138
  - mxmxml\_value\_s, 146
- nb\_zero
  - \_ssl\_context, 17
- net.c
  - net\_accept, 757
  - net\_bind, 757
  - net\_close, 757
  - net\_connect, 757
  - net\_htons, 758
  - net\_is\_blocking, 758
  - net\_recv, 758
  - net\_send, 758
  - net\_set\_block, 758
  - net\_set\_nonblock, 759
  - net\_usleep, 759
- net.h
  - net\_accept, 613
  - net\_bind, 613
  - net\_close, 613

- net\_connect, 613
- net\_recv, 614
- net\_send, 614
- net\_set\_block, 614
- net\_set\_nonblock, 614
- net\_usleep, 615
- XYSSL\_ERR\_NET\_ACCEPT\_FAILED, 612
- XYSSL\_ERR\_NET\_BIND\_FAILED, 612
- XYSSL\_ERR\_NET\_CONN\_RESET, 612
- XYSSL\_ERR\_NET\_CONNECT\_FAILED, 612
- XYSSL\_ERR\_NET\_LISTEN\_FAILED, 612
- XYSSL\_ERR\_NET\_RECV\_FAILED, 612
- XYSSL\_ERR\_NET\_SEND\_FAILED, 612
- XYSSL\_ERR\_NET\_SOCKET\_FAILED, 612
- XYSSL\_ERR\_NET\_TRY\_AGAIN, 612
- XYSSL\_ERR\_NET\_UNKNOWN\_HOST, 613
- net\_accept
  - net.c, 757
  - net.h, 613
- net\_bind
  - net.c, 757
  - net.h, 613
- net\_close
  - net.c, 757
  - net.h, 613
- net\_connect
  - net.c, 757
  - net.h, 613
- net\_htons
  - net.c, 758
- net\_is\_blocking
  - net.c, 758
- net\_recv
  - net.c, 758
  - net.h, 614
- net\_send
  - net.c, 758
  - net.h, 614
- net\_set\_block
  - net.c, 758
  - net.h, 614
- net\_set\_nonblock
  - net.c, 759
  - net.h, 614
- net\_usleep
  - net.c, 759
  - net.h, 615
- network
  - fipa\_comm\_protocol\_t, 82
- next
  - \_ssl\_session, 23
  - \_x509\_cert, 26
  - \_x509\_name, 29
  - fipa\_comm\_action\_t, 77
  - fipa\_comm\_message\_check\_t, 78
  - fipa\_comm\_protocol\_t, 82
  - fipa\_list\_t, 90
  - listNode\_s, 109
  - mxml\_node\_s, 143
- NO\_CONTENT
  - mtp\_http.h, 390
- node
  - df.h, 270
- node\_data
  - listNode\_s, 109
- nodes
  - mxml\_index\_s, 141
- NON\_AUTHORITATIVE\_INFORMATION
  - mtp\_http.h, 390
- nonce
  - connection\_s, 58
- none
  - AP\_GENERIC\_s, 52
- NOT\_ACCEPTABLE
  - mtp\_http.h, 390
- NOT\_FOUND
  - mtp\_http.h, 390
- nr
  - aes\_context, 35
- num
  - fipa\_agent\_identifier\_set\_s, 76
  - fipa\_url\_sequence\_s, 95
- NUM\_MESSAGE\_TYPE
  - message.h, 384
- num\_args
  - command\_s, 57
- num\_attrs
  - mxml\_value\_s, 146
- num\_callbacks
  - mxml-entity.c, 493
- num\_entries
  - df.h, 271
- num\_nodes
  - mxml\_index\_s, 141
- num\_params
  - fipa\_acl\_envelope\_s, 69
- num\_readers
  - mc\_rwlock\_s, 117
- num\_registered
  - barrier\_node\_s, 54
- num\_saved\_variables
  - agent\_task\_s, 50
- num\_waiting
  - barrier\_node\_s, 54
- number
  - fipa\_expression\_s::content\_u, 60



- number\_of\_elements
  - agent\_task\_s, [50](#)
- number\_of\_tasks
  - agent\_datastate\_s, [41](#)
- offset
  - havege\_state, [102](#)
- oid
  - \_x509\_name, [29](#)
- OID\_CN
  - x509.h, [677](#)
- OID\_PKCS1
  - x509.h, [677](#)
- OID\_PKCS1\_RSA
  - x509.h, [677](#)
- OID\_PKCS1\_RSA\_SHA
  - x509.h, [677](#)
- OID\_PKCS9
  - x509.h, [677](#)
- OID\_PKCS9\_EMAIL
  - x509.h, [677](#)
- OID\_X520
  - x509.h, [677](#)
- OK
  - mtp\_http.h, [390](#)
- ONE\_ITERATION
  - havege.c, [743](#)
- ontology
  - fipa\_acl\_message\_s, [71](#)
- opad
  - md2\_context, [121](#)
  - md4\_context, [123](#)
  - md5\_context, [125](#)
  - sha1\_context, [156](#)
  - sha2\_context, [158](#)
  - sha4\_context, [160](#)
- opaque
  - hr\_time, [105](#)
  - mxml\_value\_u, [147](#)
- openssl.h
  - \_\_RSA\_Passthrough, [619](#)
  - AES\_BLOCK\_SIZE, [617](#)
  - AES\_cbc\_encrypt, [617](#)
  - AES\_KEY, [617](#)
  - AES\_set\_decrypt\_key, [617](#)
  - AES\_set\_encrypt\_key, [617](#)
  - AES\_SIZE, [617](#)
  - d2i\_RSA\_PUBKEY, [619](#)
  - d2i\_RSAPrivateKey, [617](#)
  - ERR\_get\_error, [617](#)
  - MD5\_CTX, [617](#)
  - MD5\_Final, [617](#)
  - MD5\_Init, [618](#)
  - MD5\_Update, [618](#)
  - RSA, [618](#)
  - RSA\_blinding\_off, [618](#)
  - RSA\_free, [618](#)
  - RSA\_PKCS1\_PADDING, [618](#)
  - RSA\_private\_decrypt, [619](#)
  - RSA\_private\_encrypt, [619](#)
  - RSA\_public\_decrypt, [619](#)
  - RSA\_public\_encrypt, [619](#)
  - RSA\_size, [618](#)
  - SHA1\_Final, [618](#)
  - SHA1\_Init, [618](#)
  - SHA1\_Update, [618](#)
  - SHA\_CTX, [618](#)
- opmode
  - options, [150](#)
- OPMODE\_CLIENT
  - ssl\_test.c, [856](#)
- OPMODE\_NONE
  - ssl\_test.c, [856](#)
- OPMODE\_SERVER
  - ssl\_test.c, [856](#)
- options, [149](#)
  - buffer\_size, [149](#)
  - command, [149](#)
  - conn\_timeout, [149](#)
  - debug\_level, [149](#)
  - force\_cipher, [149](#)
  - iomode, [150](#)
  - max\_bytes, [150](#)
  - max\_connections, [150](#)
  - opmode, [150](#)
  - server\_name, [150](#)
  - server\_port, [150](#)
  - session\_lifetime, [150](#)
  - session\_reuse, [150](#)
- orphan
  - agent\_s, [47](#)
- out\_ctr
  - \_ssl\_context, [17](#)
- out\_hdr
  - \_ssl\_context, [17](#)
- out\_left
  - \_ssl\_context, [17](#)
- out\_msg
  - \_ssl\_context, [17](#)
- out\_msglen
  - \_ssl\_context, [17](#)
- out\_msgtype
  - \_ssl\_context, [18](#)
- own\_cert
  - \_ssl\_context, [18](#)
- owner
  - agent\_s, [48](#)

- P
  - dhm\_context, 65
  - md5.c, 748
  - rsa\_context, 154
  - sha1.c, 773
  - sha2.c, 781
  - sha4.c, 790
- p
  - \_x509\_buf, 24
  - \_x509\_node, 30
  - mpi, 131
- p\_dbg
  - \_ssl\_context, 18
- p\_recv
  - \_ssl\_context, 18
- p\_rng
  - \_ssl\_context, 18
  - rsa\_context, 154
- p\_send
  - \_ssl\_context, 18
- PACKAGE\_STRING
  - winconfig.h, 535
- PACKAGE\_VERSION
  - winconfig.h, 535
- padding
  - rsa\_context, 154
- params
  - fipa\_acl\_envelope\_s, 69
- parent
  - mxml\_node\_s, 143
- parse
  - fipa\_message\_string\_s, 91
- PARTIAL\_CONTENT
  - mtp\_http.h, 390
- passphrase
  - MCAgencyOptions\_s, 120
- PATH
  - interface.h, 549
- payload\_encoding
  - fipa\_acl\_Param\_s, 74
- payload\_length
  - fipa\_acl\_Param\_s, 74
- PAYMENT\_REQUIRED
  - mtp\_http.h, 390
- pCallbacks
  - fipa\_comm\_t, 85
- peer\_cert
  - \_ssl\_context, 19
- peer\_cn
  - \_ssl\_context, 19
- PEM\_LINE\_LENGTH
  - x509.h, 678
- performative
  - fipa\_acl\_message\_s, 71
  - fipa\_comm\_t, 86
- persistent
  - agent\_datastate\_s, 41
  - agent\_task\_s, 50
- pk\_oid
  - \_x509\_cert, 26
- PKCS9\_EMAIL
  - x509.h, 678
- PLAINTEXT
  - dh\_server.c, 838
- pmslen
  - \_ssl\_context, 19
- pool
  - havege\_state, 102
- port
  - acc.h, 221
  - host\_id\_s, 104
  - mc\_platform\_s, 114
- portno
  - agency\_s, 38
- portnoc
  - agency\_s, 38
- PRECONDITION\_FAILED
  - mtp\_http.h, 390
- premaster
  - \_ssl\_context, 19
- prev
  - mxml\_node\_s, 143
- print\_connectlist
  - commands.x.h, 259
- print\_usage
  - mc\_genkey.c, 859
- printlist\_agents
  - commands.x.h, 259
- priv\_key\_filename
  - agency\_s, 38
  - MCAgencyOptions\_s, 120
- private\_key
  - mc\_platform\_s, 115
- privatekey
  - message\_send\_arg\_s, 130
- process\_command
  - cmd\_prompt.c, 188
  - cmd\_prompt.h, 253
- PROCESSING
  - mtp\_http.h, 390
- progress\_modifier
  - agent\_datastate\_s, 41
- proposal\_func
  - fipa\_comm\_protocol\_cn\_t, 80
- protocol
  - fipa\_acl\_message\_s, 71
  - fipa\_comm\_t, 86
- protocol\_contract\_net\_callback

- fipa\_comm.h, 305
- PROXY\_AUTHENTICATION\_REQUIRED
  - mtp\_http.h, 390
- prv
  - ssl\_server.c, 849
- PT1
  - havege\_state, 102
- PT2
  - havege\_state, 102
- PT\_LEN
  - rsa.c, 762
- PTHREAD\_STACK\_SIZE
  - macros.h, 372
- PUT\_UINT64\_BE
  - sha4.c, 791
- PUT\_ULONG\_BE
  - des.c, 728
  - sha1.c, 773
  - sha2.c, 781
- PUT\_ULONG\_LE
  - aes.c, 689
  - md5.c, 748
- Q
  - rsa\_context, 154
- QP
  - rsa\_context, 154
- QUER\_IF
  - message.h, 384
- QUER\_REF
  - message.h, 384
- queue
  - fipa\_comm\_protocol\_cn\_t, 80
- QUEUE\_TEMPLATE
  - list.h, 464
- quit
  - mc\_platform\_s, 115
- quit\_cond
  - mc\_platform\_s, 115
- quit\_lock
  - mc\_platform\_s, 115
- R
  - sha1.c, 773
  - sha2.c, 782
- randbytes
  - \_ssl\_context, 19
- raw
  - \_x509\_cert, 26
  - \_x509\_raw, 31
- RCON
  - aes.c, 695
- read\_encrypted\_file
  - interface.c, 545
- interface.h, 550
- read\_known\_host\_file
  - interface.c, 545
  - interface.h, 551
- real
  - mxml\_value\_u, 147
- rece\_de\_msg
  - mtp\_http.c, 486
- receive\_AES\_en\_MA
  - interface.c, 546
  - interface.h, 551
- receive\_decrypt\_AES\_key
  - interface.h, 551
- received
  - fipa\_acl\_Param\_s, 74
- received\_by
  - fipa\_acl\_envelope\_Received\_s, 67
- received\_date
  - fipa\_acl\_envelope\_Received\_s, 67
- received\_from
  - fipa\_acl\_envelope\_Received\_s, 67
- received\_id
  - fipa\_acl\_envelope\_Received\_s, 67
- received\_via
  - fipa\_acl\_envelope\_Received\_s, 67
- receiver
  - fipa\_acl\_message\_s, 71
- receiver\_num
  - fipa\_acl\_message\_s, 71
- receiving\_decrypting\_MA\_RSA
  - interface.h, 551
- receiving\_verifying\_MA\_RSA
  - interface.h, 551
- register
  - df\_request.x.h, 273
- RELAY
  - message.h, 384
- remote\_hostname
  - connection\_s, 58
- Remove
  - data\_structures.h, 264
- remove\_nonce\_from\_MA
  - interface.c, 546
- RemoveID
  - data\_structures.h, 264
- RemoveName
  - data\_structures.h, 265
- replies
  - fipa\_comm\_protocol\_cn\_t, 80
- reply
  - fipa\_comm\_action\_t, 77
- reply\_by
  - fipa\_acl\_message\_s, 71
- reply\_migration\_process

- interface.c, [546](#)
- interface.h, [551](#)
- reply\_to
  - fipa\_acl\_message\_s, [72](#)
- reply\_with
  - fipa\_acl\_message\_s, [72](#)
- REQUEST
  - df.c, [192](#)
  - df.h, [267](#), [270](#)
  - df\_request.x.h, [273](#)
  - message.h, [384](#)
- REQUEST\_ENCRYPTION\_INITIALIZE
  - message.h, [384](#)
- REQUEST\_TIMEOUT
  - mtp\_http.h, [390](#)
- REQUEST\_URI\_TOO\_LONG
  - mtp\_http.h, [390](#)
- request\_handler\_DEREGISTER
  - df.c, [195](#)
- request\_handler\_REGISTER
  - df.c, [195](#)
- request\_handler\_SEARCH
  - df.c, [195](#)
- request\_handler\_SUBSCRIBE
  - df.c, [195](#)
- request\_list
  - df.h, [271](#)
- REQUESTED\_RANGE\_NOT\_SATISFIABLE
  - mtp\_http.h, [391](#)
- require\_inform
  - fipa\_comm\_protocol\_cn\_t, [80](#)
- REQUEST\_ENTITY\_TOO\_LARGE
  - mtp\_http.h, [390](#)
- RESET\_CONTENT
  - mtp\_http.h, [390](#)
- resolvers
  - fipa\_agent\_identifier\_s, [75](#)
- response\_code
  - mtp\_http\_s, [136](#)
- response\_string
  - mtp\_http\_s, [136](#)
- resume
  - \_ssl\_context, [19](#)
- retain\_order
  - fipa\_agent\_identifier\_set\_s, [76](#)
- RETRY\_WITH
  - mtp\_http.h, [391](#)
- RETURN\_MSG
  - message.h, [384](#)
- return\_code
  - mtp\_http\_s, [136](#)
- return\_data
  - agent\_datastate\_s, [41](#)
  - agent\_s, [48](#)
- RHs
  - des.c, [734](#)
- rk
  - aes\_context, [35](#)
- RN
  - rsa\_context, [154](#)
- ROTL8
  - aes.c, [690](#)
- ROTR
  - sha2.c, [782](#)
  - sha4.c, [791](#)
- RP
  - dhm\_context, [65](#)
  - rsa\_context, [154](#)
- RQ
  - rsa\_context, [155](#)
- RSA
  - openssl.h, [618](#)
- rsa
  - \_x509\_cert, [26](#)
- rsa.c
  - KEY\_LEN, [762](#)
  - PT\_LEN, [762](#)
  - rsa\_check\_privkey, [764](#)
  - rsa\_check\_pubkey, [765](#)
  - RSA\_D, [762](#)
  - RSA\_DP, [762](#)
  - RSA\_DQ, [763](#)
  - RSA\_E, [763](#)
  - rsa\_free, [765](#)
  - rsa\_gen\_key, [765](#)
  - rsa\_init, [766](#)
  - RSA\_N, [763](#)
  - RSA\_P, [763](#)
  - rsa\_pkcs1\_decrypt, [766](#)
  - rsa\_pkcs1\_encrypt, [767](#)
  - rsa\_pkcs1\_sign, [767](#)
  - rsa\_pkcs1\_verify, [768](#)
  - rsa\_private, [768](#)
  - RSA\_PT, [764](#)
  - rsa\_public, [769](#)
  - RSA\_Q, [764](#)
  - RSA\_QP, [764](#)
  - rsa\_self\_test, [769](#)
- rsa.h
  - ASN1\_HASH\_MD5, [622](#)
  - ASN1\_HASH\_SHA1, [622](#)
  - rsa\_check\_privkey, [625](#)
  - rsa\_check\_pubkey, [625](#)
  - RSA\_CRYPT, [623](#)
  - rsa\_free, [626](#)
  - rsa\_gen\_key, [626](#)
  - rsa\_init, [626](#)
  - RSA\_MD2, [623](#)

- RSA\_MD4, 623
- RSA\_MD5, 623
- rsa\_pkcs1\_decrypt, 627
- rsa\_pkcs1\_encrypt, 627
- rsa\_pkcs1\_sign, 628
- rsa\_pkcs1\_verify, 628
- RSA\_PKCS\_V15, 623
- RSA\_PKCS\_V21, 623
- RSA\_PRIVATE, 623
- rsa\_private, 629
- RSA\_PUBLIC, 623
- rsa\_public, 630
- RSA\_RAW, 624
- rsa\_self\_test, 630
- RSA\_SHA1, 624
- RSA\_SHA256, 624
- RSA\_SIGN, 624
- XYSSL\_ERR\_RSA\_BAD\_INPUT\_DATA, 624
- XYSSL\_ERR\_RSA\_INVALID\_PADDING, 624
- XYSSL\_ERR\_RSA\_KEY\_CHECK\_FAILED, 624
- XYSSL\_ERR\_RSA\_KEY\_GEN\_FAILED, 624
- XYSSL\_ERR\_RSA\_PRIVATE\_FAILED, 625
- XYSSL\_ERR\_RSA\_PUBLIC\_FAILED, 625
- XYSSL\_ERR\_RSA\_VERIFY\_FAILED, 625
- RSA\_blinding\_off, openssl.h, 618
- rsa\_check\_privkey, rsa.c, 764, rsa.h, 625
- rsa\_check\_pubkey, rsa.c, 765, rsa.h, 625
- rsa\_context, 152
  - D, 152
  - DP, 152
  - DQ, 152
  - E, 153
  - f\_rng, 153
  - hash\_id, 153
  - len, 153
  - N, 153
  - P, 154
  - p\_rng, 154
  - padding, 154
  - Q, 154
  - QP, 154
  - RN, 154
  - RP, 154
  - RQ, 155
  - ver, 155
- RSA\_CRYPT, rsa.h, 623
- RSA\_D, rsa.c, 762
- RSA\_DE, interface.h, 549
- rsa\_decryption, interface.c, 546, interface.h, 551
- RSA\_DP, rsa.c, 762
- RSA\_DQ, rsa.c, 763
- RSA\_E, rsa.c, 763
- RSA\_EN, interface.h, 549
- rsa\_encryption, interface.c, 546, interface.h, 551
- RSA\_free, openssl.h, 618
- rsa\_free, rsa.c, 765, rsa.h, 626
- rsa\_gen\_key, rsa.c, 765, rsa.h, 626
- rsa\_genkey.c, EXPONENT, 840, KEY\_SIZE, 840, main, 840
- rsa\_init, rsa.c, 766, rsa.h, 626
- rsa\_key, \_ssl\_context, 20
- RSA\_MD2, rsa.h, 623
- RSA\_MD4, rsa.h, 623
- RSA\_MD5, rsa.h, 623
- RSA\_N, rsa.c, 763
- RSA\_P, rsa.c, 763
- rsa\_pkcs1\_decrypt, rsa.c, 766, rsa.h, 627
- rsa\_pkcs1\_encrypt, rsa.c, 767, rsa.h, 627
- RSA\_PKCS1\_PADDING

- openssl.h, 618
- rsa\_pkcs1\_sign
  - rsa.c, 767
  - rsa.h, 628
- rsa\_pkcs1\_verify
  - rsa.c, 768
  - rsa.h, 628
- RSA\_PKCS\_V15
  - rsa.h, 623
- RSA\_PKCS\_V21
  - rsa.h, 623
- RSA\_PRIVATE
  - rsa.h, 623
- rsa\_private
  - rsa.c, 768
  - rsa.h, 629
- RSA\_private\_decrypt
  - openssl.h, 619
- RSA\_private\_encrypt
  - openssl.h, 619
- RSA\_PT
  - rsa.c, 764
- RSA\_PUBLIC
  - rsa.h, 623
- rsa\_public
  - rsa.c, 769
  - rsa.h, 630
- RSA\_public\_decrypt
  - openssl.h, 619
- RSA\_public\_encrypt
  - openssl.h, 619
- RSA\_Q
  - rsa.c, 764
- RSA\_QP
  - rsa.c, 764
- RSA\_RAW
  - rsa.h, 624
- rsa\_self\_test
  - rsa.c, 769
  - rsa.h, 630
- RSA\_SHA1
  - rsa.h, 624
- RSA\_SHA256
  - rsa.h, 624
- RSA\_SIGN
  - rsa.h, 624
- rsa\_sign.c
  - main, 841
- RSA\_size
  - openssl.h, 618
- rsa\_verify.c
  - main, 842
- RSb
  - aes.c, 695
- RT0
  - aes.c, 695
- RT1
  - aes.c, 695
- RT2
  - aes.c, 695
- RT3
  - aes.c, 695
- run\_lock
  - agent\_s, 48
- RWLOCK\_DESTROY
  - macros.h, 372
- RWLOCK\_INIT
  - macros.h, 372
- RWLOCK\_RDLOCK
  - macros.h, 372
- RWLOCK\_RDUNLOCK
  - macros.h, 372
- RWLOCK\_T
  - macros.h, 373
- RWLOCK\_WRLOCK
  - macros.h, 373
- RWLOCK\_WRUNLOCK
  - macros.h, 373
- S
  - md5.c, 749
  - sha1.c, 773
- s
  - mpi, 131
- S0
  - sha2.c, 782
  - sha4.c, 791
- S1
  - sha2.c, 782
  - sha4.c, 791
- S2
  - sha2.c, 782
  - sha4.c, 791
- S3
  - sha2.c, 782
  - sha4.c, 791
- s\_get
  - \_ssl\_context, 20
- s\_list\_1st
  - ssl\_server.c, 849
- s\_set
  - \_ssl\_context, 20
- SAFE\_FREE
  - mtp\_http.c, 483
- safe\_strcpy
  - mxmldoc.c, 525
- saved\_variables
  - agent\_task\_s, 50

- SB1
  - des.c, [735](#)
- SB2
  - des.c, [735](#)
- SB3
  - des.c, [736](#)
- SB4
  - des.c, [736](#)
- SB5
  - des.c, [736](#)
- SB6
  - des.c, [737](#)
- SB7
  - des.c, [737](#)
- SB8
  - des.c, [738](#)
- scan\_file
  - mxmldoc.c, [525](#)
- Search
  - data\_structures.h, [265](#)
- search
  - df\_request.x.h, [273](#)
- search\_results
  - df.h, [271](#)
- search\_string
  - df.h, [271](#)
- SearchName
  - data\_structures.h, [265](#)
- sec
  - \_x509\_time, [33](#)
- second
  - fipa\_DateTime\_s, [88](#)
- security/xyssl-0.9/include/xyssl/config.h
  - XYSSL\_AES\_C, [488](#)
  - XYSSL\_ARC4\_C, [488](#)
  - XYSSL\_BASE64\_C, [488](#)
  - XYSSL\_BIGNUM\_C, [488](#)
  - XYSSL\_CERTS\_C, [488](#)
  - XYSSL\_DEBUG\_C, [489](#)
  - XYSSL\_DEBUG\_MSG, [489](#)
  - XYSSL\_DES\_C, [489](#)
  - XYSSL\_DHM\_C, [489](#)
  - XYSSL\_GENPRIME, [489](#)
  - XYSSL\_HAVE\_ASM, [489](#)
  - XYSSL\_HAVEGE\_C, [489](#)
  - XYSSL\_MD5\_C, [489](#)
  - XYSSL\_NET\_C, [489](#)
  - XYSSL\_PADLOCK\_C, [489](#)
  - XYSSL\_RSA\_C, [489](#)
  - XYSSL\_SELF\_TEST, [490](#)
  - XYSSL\_SHA1\_C, [490](#)
  - XYSSL\_SHA2\_C, [490](#)
  - XYSSL\_SHA4\_C, [490](#)
  - XYSSL\_SSL\_CLI\_C, [490](#)
  - XYSSL\_SSL\_SRV\_C, [490](#)
  - XYSSL\_SSL\_TLS\_C, [490](#)
  - XYSSL\_TIMING\_C, [490](#)
  - XYSSL\_X509\_PARSE\_C, [490](#)
  - XYSSL\_X509\_WRITE\_C, [490](#)
- selftest.c
  - main, [852](#)
- sem
  - syncListNode\_s, [164](#)
- SEMAPHORE\_DESTROY
  - macros.h, [373](#)
- SEMAPHORE\_INIT
  - macros.h, [373](#)
- SEMAPHORE\_POST
  - macros.h, [373](#)
- SEMAPHORE\_T
  - macros.h, [373](#)
- SEMAPHORE\_WAIT
  - macros.h, [373](#)
- send
  - commands.x.h, [259](#)
- send\_AES\_en\_MA
  - interface.c, [546](#)
  - interface.h, [551](#)
- sender
  - agent\_s, [48](#)
  - fipa\_acl\_message\_s, [72](#)
- sending\_agent\_name
  - message\_s, [128](#)
- separate\_key\_parts
  - interface.c, [547](#)
- serial
  - \_x509\_cert, [26](#)
  - \_x509\_raw, [31](#)
- server
  - agency\_s, [38](#)
  - mtp\_http\_s, [136](#)
- server\_fd
  - acc.h, [221](#)
  - ssl\_test.c, [858](#)
- SERVER\_NAME
  - dh\_client.c, [836](#)
  - ssl\_client1.c, [843](#)
  - ssl\_client2.c, [845](#)
- server\_name
  - agent\_task\_s, [50](#)
  - options, [150](#)
- SERVER\_PORT
  - dh\_client.c, [836](#)
  - dh\_server.c, [838](#)
  - ssl\_client1.c, [843](#)
  - ssl\_client2.c, [845](#)
- server\_port
  - options, [150](#)

- serverfd
  - connection\_s, 59
- service\_list
  - df.h, 271
- service\_types\_e
  - df.h, 267
- session
  - \_ssl\_context, 20
- session\_lifetime
  - options, 150
- session\_reuse
  - options, 150
- set\_alarm
  - timing.c, 819
  - timing.h, 671
- set\_bar
  - foo\_c, 98
  - foo\_s, 100
- set\_foo
  - foo\_c, 98
  - foo\_s, 101
- set\_foobar
  - foo\_c, 98
- SET\_THREAD\_OFF
  - macros.h, 374
- SET\_THREAD\_ON
  - macros.h, 374
- sha1
  - sha1.c, 773
  - sha1.h, 632
- sha1.c
  - F, 772
  - GET\_ULONG\_BE, 772
  - K, 772, 773
  - P, 773
  - PUT\_ULONG\_BE, 773
  - R, 773
  - S, 773
  - sha1, 773
  - sha1\_file, 774
  - sha1\_finish, 774
  - sha1\_hmac, 774
  - sha1\_hmac\_finish, 775
  - sha1\_hmac\_starts, 775
  - sha1\_hmac\_test\_buf, 777
  - sha1\_hmac\_test\_buflen, 777
  - sha1\_hmac\_test\_key, 777
  - sha1\_hmac\_test\_keylen, 778
  - sha1\_hmac\_test\_sum, 778
  - sha1\_hmac\_update, 775
  - sha1\_padding, 778
  - sha1\_process, 775
  - sha1\_self\_test, 776
  - sha1\_starts, 776
  - sha1\_test\_buf, 778
  - sha1\_test\_buflen, 779
  - sha1\_test\_sum, 779
  - sha1\_update, 776
- sha1.h
  - sha1, 632
  - sha1\_file, 632
  - sha1\_finish, 632
  - sha1\_hmac, 632
  - sha1\_hmac\_finish, 633
  - sha1\_hmac\_starts, 633
  - sha1\_hmac\_update, 633
  - sha1\_self\_test, 634
  - sha1\_starts, 634
  - sha1\_update, 634
- sha1\_check
  - sha1sum.c, 834
- sha1\_context, 156
  - buffer, 156
  - ipad, 156
  - opad, 156
  - state, 156
  - total, 156
- sha1\_file
  - sha1.c, 774
  - sha1.h, 632
- SHA1\_Final
  - openssl.h, 618
- sha1\_finish
  - sha1.c, 774
  - sha1.h, 632
- sha1\_hmac
  - sha1.c, 774
  - sha1.h, 632
- sha1\_hmac\_finish
  - sha1.c, 775
  - sha1.h, 633
- sha1\_hmac\_starts
  - sha1.c, 775
  - sha1.h, 633
- sha1\_hmac\_test\_buf
  - sha1.c, 777
- sha1\_hmac\_test\_buflen
  - sha1.c, 777
- sha1\_hmac\_test\_key
  - sha1.c, 777
- sha1\_hmac\_test\_keylen
  - sha1.c, 778
- sha1\_hmac\_test\_sum
  - sha1.c, 778
- sha1\_hmac\_update
  - sha1.c, 775
  - sha1.h, 633
- SHA1\_Init



- openssl.h, 618
- sha1\_padding
  - sha1.c, 778
- sha1\_print
  - sha1sum.c, 834
- sha1\_process
  - sha1.c, 775
- sha1\_self\_test
  - sha1.c, 776
  - sha1.h, 634
- sha1\_starts
  - sha1.c, 776
  - sha1.h, 634
- sha1\_test\_buf
  - sha1.c, 778
- sha1\_test\_buflen
  - sha1.c, 779
- sha1\_test\_sum
  - sha1.c, 779
- SHA1\_Update
  - openssl.h, 618
- sha1\_update
  - sha1.c, 776
  - sha1.h, 634
- sha1\_wrapper
  - sha1sum.c, 834
- sha1sum.c
  - main, 834
  - sha1\_check, 834
  - sha1\_print, 834
  - sha1\_wrapper, 834
- sha2
  - sha2.c, 782
  - sha2.h, 637
- sha2.c
  - F0, 781
  - F1, 781
  - GET\_ULONG\_BE, 781
  - P, 781
  - PUT\_ULONG\_BE, 781
  - R, 782
  - ROTR, 782
  - S0, 782
  - S1, 782
  - S2, 782
  - S3, 782
  - sha2, 782
  - sha2\_file, 783
  - sha2\_finish, 783
  - sha2\_hmac, 783
  - sha2\_hmac\_finish, 784
  - sha2\_hmac\_starts, 784
  - sha2\_hmac\_test\_buf, 786
  - sha2\_hmac\_test\_buflen, 786
  - sha2\_hmac\_test\_key, 786
  - sha2\_hmac\_test\_keylen, 787
  - sha2\_hmac\_test\_sum, 787
  - sha2\_hmac\_update, 784
  - sha2\_padding, 787
  - sha2\_process, 784
  - sha2\_self\_test, 785
  - sha2\_starts, 785
  - sha2\_test\_buf, 787
  - sha2\_test\_buflen, 787
  - sha2\_test\_sum, 788
  - sha2\_update, 785
  - SHR, 782
- sha2.h
  - sha2, 637
  - sha2\_file, 637
  - sha2\_finish, 637
  - sha2\_hmac, 637
  - sha2\_hmac\_finish, 638
  - sha2\_hmac\_starts, 638
  - sha2\_hmac\_update, 638
  - sha2\_self\_test, 639
  - sha2\_starts, 639
  - sha2\_update, 639
- sha2\_check
  - sha2sum.c, 835
- sha2\_context, 158
  - buffer, 158
  - ipad, 158
  - is224, 158
  - opad, 158
  - state, 158
  - total, 159
- sha2\_file
  - sha2.c, 783
  - sha2.h, 637
- sha2\_finish
  - sha2.c, 783
  - sha2.h, 637
- sha2\_hmac
  - sha2.c, 783
  - sha2.h, 637
- sha2\_hmac\_finish
  - sha2.c, 784
  - sha2.h, 638
- sha2\_hmac\_starts
  - sha2.c, 784
  - sha2.h, 638
- sha2\_hmac\_test\_buf
  - sha2.c, 786
- sha2\_hmac\_test\_buflen
  - sha2.c, 786
- sha2\_hmac\_test\_key
  - sha2.c, 786

- sha2\_hmac\_test\_keylen
  - sha2.c, [787](#)
- sha2\_hmac\_test\_sum
  - sha2.c, [787](#)
- sha2\_hmac\_update
  - sha2.c, [784](#)
  - sha2.h, [638](#)
- sha2\_padding
  - sha2.c, [787](#)
- sha2\_print
  - sha2sum.c, [835](#)
- sha2\_process
  - sha2.c, [784](#)
- sha2\_self\_test
  - sha2.c, [785](#)
  - sha2.h, [639](#)
- sha2\_starts
  - sha2.c, [785](#)
  - sha2.h, [639](#)
- sha2\_test\_buf
  - sha2.c, [787](#)
- sha2\_test\_buflen
  - sha2.c, [787](#)
- sha2\_test\_sum
  - sha2.c, [788](#)
- sha2\_update
  - sha2.c, [785](#)
  - sha2.h, [639](#)
- sha2\_wrapper
  - sha2sum.c, [835](#)
- sha2sum.c
  - main, [835](#)
  - sha2\_check, [835](#)
  - sha2\_print, [835](#)
  - sha2\_wrapper, [835](#)
- sha4
  - sha4.c, [791](#)
  - sha4.h, [642](#)
- sha4.c
  - F0, [790](#)
  - F1, [790](#)
  - GET\_UINT64\_BE, [790](#)
  - K, [794](#)
  - P, [790](#)
  - PUT\_UINT64\_BE, [791](#)
  - ROTR, [791](#)
  - S0, [791](#)
  - S1, [791](#)
  - S2, [791](#)
  - S3, [791](#)
  - sha4, [791](#)
  - sha4\_file, [791](#)
  - sha4\_finish, [792](#)
  - sha4\_hmac, [792](#)
- sha4\_hmac\_finish, [792](#)
- sha4\_hmac\_starts, [793](#)
- sha4\_hmac\_test\_buf, [794](#)
- sha4\_hmac\_test\_buflen, [795](#)
- sha4\_hmac\_test\_key, [795](#)
- sha4\_hmac\_test\_keylen, [796](#)
- sha4\_hmac\_test\_sum, [796](#)
- sha4\_hmac\_update, [793](#)
- sha4\_padding, [796](#)
- sha4\_process, [793](#)
- sha4\_self\_test, [793](#)
- sha4\_starts, [794](#)
- sha4\_test\_buf, [796](#)
- sha4\_test\_buflen, [796](#)
- sha4\_test\_sum, [797](#)
- sha4\_update, [794](#)
- SHR, [791](#)
- sha4.h
  - int64, [642](#)
  - sha4, [642](#)
  - sha4\_file, [642](#)
  - sha4\_finish, [642](#)
  - sha4\_hmac, [643](#)
  - sha4\_hmac\_finish, [643](#)
  - sha4\_hmac\_starts, [643](#)
  - sha4\_hmac\_update, [644](#)
  - sha4\_self\_test, [644](#)
  - sha4\_starts, [644](#)
  - sha4\_update, [645](#)
  - UL64, [642](#)
- sha4\_context, [160](#)
  - buffer, [160](#)
  - ipad, [160](#)
  - is384, [160](#)
  - opad, [160](#)
  - state, [160](#)
  - total, [161](#)
- sha4\_file
  - sha4.c, [791](#)
  - sha4.h, [642](#)
- sha4\_finish
  - sha4.c, [792](#)
  - sha4.h, [642](#)
- sha4\_hmac
  - sha4.c, [792](#)
  - sha4.h, [643](#)
- sha4\_hmac\_finish
  - sha4.c, [792](#)
  - sha4.h, [643](#)
- sha4\_hmac\_starts
  - sha4.c, [793](#)
  - sha4.h, [643](#)
- sha4\_hmac\_test\_buf
  - sha4.c, [794](#)

- sha4\_hmac\_test\_buflen
  - sha4.c, [795](#)
- sha4\_hmac\_test\_key
  - sha4.c, [795](#)
- sha4\_hmac\_test\_keylen
  - sha4.c, [796](#)
- sha4\_hmac\_test\_sum
  - sha4.c, [796](#)
- sha4\_hmac\_update
  - sha4.c, [793](#)
  - sha4.h, [644](#)
- sha4\_padding
  - sha4.c, [796](#)
- sha4\_process
  - sha4.c, [793](#)
- sha4\_self\_test
  - sha4.c, [793](#)
  - sha4.h, [644](#)
- sha4\_starts
  - sha4.c, [794](#)
  - sha4.h, [644](#)
- sha4\_test\_buf
  - sha4.c, [796](#)
- sha4\_test\_buflen
  - sha4.c, [796](#)
- sha4\_test\_sum
  - sha4.c, [797](#)
- sha4\_update
  - sha4.c, [794](#)
  - sha4.h, [645](#)
- SHA\_CTX
  - openssl.h, [618](#)
- SHR
  - sha2.c, [782](#)
  - sha4.c, [791](#)
- sig
  - \_x509\_cert, [26](#)
- sig\_oid1
  - \_x509\_cert, [26](#)
- sig\_oid2
  - \_x509\_cert, [27](#)
- sighandler
  - timing.c, [819](#)
- sign
  - \_x509\_raw, [31](#)
  - fipa\_DateTime\_s, [88](#)
- SIGNAL
  - macros.h, [374](#)
- signalg
  - \_x509\_raw, [31](#)
- signalled
  - syncListNode\_s, [164](#)
- size
  - barrier\_queue\_s, [55](#)
  - df.h, [271](#)
  - dynstring\_s, [66](#)
  - interpreter\_variable\_data\_s, [107](#)
  - list\_s, [108](#)
  - syncList\_s, [163](#)
- size\_of\_element\_array
  - agent\_task\_s, [50](#)
- sk
  - des3\_context, [62](#)
  - des\_context, [63](#)
- SLEEP\_QUEUE
  - macros.h, [374](#)
- SLEEP\_RESET
  - macros.h, [374](#)
- small\_prime
  - bignum.c, [718](#)
- SOCKET\_ERROR
  - macros.h, [374](#)
- SOCKET\_INPUT\_SIZE
  - message.c, [478](#)
  - mtp\_http.h, [389](#)
- sockfd
  - mc\_platform\_s, [115](#)
- SOME
  - df.h, [267](#)
- sort\_node
  - mxmldoc.c, [526](#)
- split\_string
  - cmd\_prompt.c, [188](#)
  - cmd\_prompt.h, [253](#)
- ssl.h
  - SSL\_CERTIFICATE\_REQUEST, [658](#)
  - SSL\_CERTIFICATE\_VERIFY, [658](#)
  - SSL\_CLIENT\_CERTIFICATE, [658](#)
  - SSL\_CLIENT\_CHANGE\_CIPHER\_SPEC, [658](#)
  - SSL\_CLIENT\_FINISHED, [658](#)
  - SSL\_CLIENT\_HELLO, [658](#)
  - SSL\_CLIENT\_KEY\_EXCHANGE, [658](#)
  - SSL\_FLUSH\_BUFFERS, [658](#)
  - SSL\_HANDSHAKE\_OVER, [658](#)
  - SSL\_HELLO\_REQUEST, [658](#)
  - SSL\_SERVER\_CERTIFICATE, [658](#)
  - SSL\_SERVER\_CHANGE\_CIPHER\_SPEC, [658](#)
  - SSL\_SERVER\_FINISHED, [658](#)
  - SSL\_SERVER\_HELLO, [658](#)
  - SSL\_SERVER\_HELLO\_DONE, [658](#)
  - SSL\_SERVER\_KEY\_EXCHANGE, [658](#)
  - SSL\_ALERT\_CLOSE\_NOTIFY, [650](#)
  - SSL\_ALERT\_FATAL, [650](#)
  - SSL\_ALERT\_NO\_CERTIFICATE, [650](#)
  - SSL\_ALERT\_WARNING, [650](#)
  - SSL\_BUFFER\_LEN, [650](#)

- ssl\_calc\_verify, 658
- ssl\_close\_notify, 658
- SSL\_COMPRESS\_NULL, 650
- ssl\_context, 658
- ssl\_default\_ciphers, 669
- ssl\_derive\_keys, 659
- SSL\_EDH\_RSA\_AES\_256\_SHA, 650
- SSL\_EDH\_RSA\_DES\_168\_SHA, 650
- ssl\_fetch\_input, 659
- ssl\_flush\_output, 659
- ssl\_free, 659
- ssl\_get\_bytes\_avail, 660
- ssl\_get\_cipher, 660
- ssl\_get\_verify\_result, 660
- ssl\_handshake, 660
- ssl\_handshake\_client, 661
- ssl\_handshake\_server, 661
- SSL\_HS\_CERTIFICATE, 651
- SSL\_HS\_CERTIFICATE\_REQUEST, 651
- SSL\_HS\_CERTIFICATE\_VERIFY, 651
- SSL\_HS\_CLIENT\_HELLO, 651
- SSL\_HS\_CLIENT\_KEY\_EXCHANGE, 651
- SSL\_HS\_FINISHED, 651
- SSL\_HS\_HELLO\_REQUEST, 651
- SSL\_HS\_SERVER\_HELLO, 651
- SSL\_HS\_SERVER\_HELLO\_DONE, 651
- SSL\_HS\_SERVER\_KEY\_EXCHANGE, 652
- ssl\_init, 661
- SSL\_IS\_CLIENT, 652
- SSL\_IS\_SERVER, 652
- SSL\_MAJOR\_VERSION\_3, 652
- SSL\_MAX\_CONTENT\_LEN, 652
- SSL\_MINOR\_VERSION\_0, 652
- SSL\_MINOR\_VERSION\_1, 652
- SSL\_MINOR\_VERSION\_2, 653
- SSL\_MSG\_ALERT, 653
- SSL\_MSG\_APPLICATION\_DATA, 653
- SSL\_MSG\_CHANGE\_CIPHER\_SPEC, 653
- SSL\_MSG\_HANDSHAKE, 653
- ssl\_parse\_certificate, 662
- ssl\_parse\_change\_cipher\_spec, 662
- ssl\_parse\_finished, 662
- ssl\_read, 662
- ssl\_read\_record, 663
- SSL\_RSA\_AES\_128\_SHA, 653
- SSL\_RSA\_AES\_256\_SHA, 653
- SSL\_RSA\_DES\_168\_SHA, 653
- SSL\_RSA\_RC4\_128\_MD5, 654
- SSL\_RSA\_RC4\_128\_SHA, 654
- ssl\_session, 658
- ssl\_set\_authmode, 663
- ssl\_set\_bio, 664
- ssl\_set\_ca\_chain, 664
- ssl\_set\_ciphers, 664
- ssl\_set\_dbg, 665
- ssl\_set\_dh\_param, 665
- ssl\_set\_endpoint, 665
- ssl\_set\_hostname, 665
- ssl\_set\_own\_cert, 666
- ssl\_set\_rng, 666
- ssl\_set\_scb, 666
- ssl\_set\_session, 667
- ssl\_states, 658
- SSL\_VERIFY\_NONE, 654
- SSL\_VERIFY\_OPTIONAL, 654
- SSL\_VERIFY\_REQUIRED, 654
- ssl\_write, 667
- ssl\_write\_certificate, 668
- ssl\_write\_change\_cipher\_spec, 668
- ssl\_write\_finished, 668
- ssl\_write\_record, 668
- TLS\_EXT\_SERVERNAME, 654
- TLS\_EXT\_SERVERNAME\_HOSTNAME, 654
- XYSSL\_ERR\_SSL\_BAD\_HS\_-  
CERTIFICATE, 654
- XYSSL\_ERR\_SSL\_BAD\_HS\_-  
CERTIFICATE\_REQUEST, 654
- XYSSL\_ERR\_SSL\_BAD\_HS\_-  
CERTIFICATE\_VERIFY, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_CHANGE\_-  
CIPHER\_SPEC, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_CLIENT\_-  
HELLO, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_CLIENT\_-  
KEY\_EXCHANGE, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_FINISHED, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_-  
HELLO, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_-  
HELLO\_DONE, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_-  
KEY\_EXCHANGE, 655
- XYSSL\_ERR\_SSL\_BAD\_INPUT\_DATA, 655
- XYSSL\_ERR\_SSL\_CA\_CHAIN\_-  
REQUIRED, 656
- XYSSL\_ERR\_SSL\_CERTIFICATE\_-  
REQUIRED, 656
- XYSSL\_ERR\_SSL\_CERTIFICATE\_TOO\_-  
LARGE, 656
- XYSSL\_ERR\_SSL\_FATAL\_ALERT\_-  
MESSAGE, 656
- XYSSL\_ERR\_SSL\_FEATURE\_-  
UNAVAILABLE, 656
- XYSSL\_ERR\_SSL\_INVALID\_MAC, 656

- XYSSL\_ERR\_SSL\_INVALID\_MODULUS\_-  
SIZE, [656](#)
- XYSSL\_ERR\_SSL\_INVALID\_RECORD,  
[656](#)
- XYSSL\_ERR\_SSL\_NO\_CIPHER\_CHOSEN,  
[657](#)
- XYSSL\_ERR\_SSL\_NO\_CLIENT\_-  
CERTIFICATE, [657](#)
- XYSSL\_ERR\_SSL\_NO\_SESSION\_FOUND,  
[657](#)
- XYSSL\_ERR\_SSL\_PEER\_CLOSE\_-  
NOTIFY, [657](#)
- XYSSL\_ERR\_SSL\_PEER\_VERIFY\_-  
FAILED, [657](#)
- XYSSL\_ERR\_SSL\_PRIVATE\_KEY\_-  
REQUIRED, [657](#)
- XYSSL\_ERR\_SSL\_UNEXPECTED\_-  
MESSAGE, [657](#)
- XYSSL\_ERR\_SSL\_UNKNOWN\_CIPHER,  
[657](#)
- SSL\_CERTIFICATE\_REQUEST  
ssl.h, [658](#)
- SSL\_CERTIFICATE\_VERIFY  
ssl.h, [658](#)
- SSL\_CLIENT\_CERTIFICATE  
ssl.h, [658](#)
- SSL\_CLIENT\_CHANGE\_CIPHER\_SPEC  
ssl.h, [658](#)
- SSL\_CLIENT\_FINISHED  
ssl.h, [658](#)
- SSL\_CLIENT\_HELLO  
ssl.h, [658](#)
- SSL\_CLIENT\_KEY\_EXCHANGE  
ssl.h, [658](#)
- SSL\_FLUSH\_BUFFERS  
ssl.h, [658](#)
- SSL\_HANDSHAKE\_OVER  
ssl.h, [658](#)
- SSL\_HELLO\_REQUEST  
ssl.h, [658](#)
- SSL\_SERVER\_CERTIFICATE  
ssl.h, [658](#)
- SSL\_SERVER\_CHANGE\_CIPHER\_SPEC  
ssl.h, [658](#)
- SSL\_SERVER\_FINISHED  
ssl.h, [658](#)
- SSL\_SERVER\_HELLO  
ssl.h, [658](#)
- SSL\_SERVER\_HELLO\_DONE  
ssl.h, [658](#)
- SSL\_SERVER\_KEY\_EXCHANGE  
ssl.h, [658](#)
- SSL\_ALERT\_CLOSE\_NOTIFY  
ssl.h, [650](#)
- SSL\_ALERT\_FATAL  
ssl.h, [650](#)
- SSL\_ALERT\_NO\_CERTIFICATE  
ssl.h, [650](#)
- SSL\_ALERT\_WARNING  
ssl.h, [650](#)
- SSL\_BUFFER\_LEN  
ssl.h, [650](#)
- ssl\_calc\_finished  
ssl\_tls.c, [806](#)
- ssl\_calc\_verify  
ssl.h, [658](#)  
ssl\_tls.c, [806](#)
- ssl\_cli.c
  - ssl\_handshake\_client, [798](#)
  - ssl\_parse\_certificate\_request, [798](#)
  - ssl\_parse\_server\_hello, [798](#)
  - ssl\_parse\_server\_hello\_done, [799](#)
  - ssl\_parse\_server\_key\_exchange, [799](#)
  - ssl\_write\_certificate\_verify, [799](#)
  - ssl\_write\_client\_hello, [799](#)
  - ssl\_write\_client\_key\_exchange, [800](#)
- ssl\_client1.c
  - DEBUG\_LEVEL, [843](#)
  - GET\_REQUEST, [843](#)
  - main, [844](#)
  - my\_debug, [844](#)
  - SERVER\_NAME, [843](#)
  - SERVER\_PORT, [843](#)
- ssl\_client2.c
  - DEBUG\_LEVEL, [845](#)
  - GET\_REQUEST, [845](#)
  - main, [846](#)
  - my\_debug, [846](#)
  - SERVER\_NAME, [845](#)
  - SERVER\_PORT, [845](#)
- ssl\_close\_notify  
ssl.h, [658](#)  
ssl\_tls.c, [806](#)
- SSL\_COMPRESS\_NULL  
ssl.h, [650](#)
- ssl\_context  
ssl.h, [658](#)
- SSL\_DEBUG\_BUF  
debug.h, [583](#)
- SSL\_DEBUG\_CRT  
debug.h, [583](#)
- SSL\_DEBUG\_MPI  
debug.h, [584](#)
- SSL\_DEBUG\_MSG  
debug.h, [584](#)
- SSL\_DEBUG\_RET  
debug.h, [584](#)
- ssl\_decrypt\_buf

- ssl\_tls.c, [806](#)
- ssl\_default\_ciphers
  - ssl.h, [669](#)
  - ssl\_tls.c, [817](#)
- ssl\_derive\_keys
  - ssl.h, [659](#)
  - ssl\_tls.c, [807](#)
- SSL\_EDH\_RSA\_AES\_256\_SHA
  - ssl.h, [650](#)
- SSL\_EDH\_RSA\_DES\_168\_SHA
  - ssl.h, [650](#)
- ssl\_encrypt\_buf
  - ssl\_tls.c, [807](#)
- ssl\_fetch\_input
  - ssl.h, [659](#)
  - ssl\_tls.c, [807](#)
- ssl\_flush\_output
  - ssl.h, [659](#)
  - ssl\_tls.c, [807](#)
- ssl\_free
  - ssl.h, [659](#)
  - ssl\_tls.c, [807](#)
- ssl\_get\_bytes\_avail
  - ssl.h, [660](#)
  - ssl\_tls.c, [808](#)
- ssl\_get\_cipher
  - ssl.h, [660](#)
  - ssl\_tls.c, [808](#)
- ssl\_get\_verify\_result
  - ssl.h, [660](#)
  - ssl\_tls.c, [808](#)
- ssl\_handshake
  - ssl.h, [660](#)
  - ssl\_tls.c, [809](#)
- ssl\_handshake\_client
  - ssl.h, [661](#)
  - ssl\_cli.c, [798](#)
- ssl\_handshake\_server
  - ssl.h, [661](#)
  - ssl\_srv.c, [801](#)
- SSL\_HS\_CERTIFICATE
  - ssl.h, [651](#)
- SSL\_HS\_CERTIFICATE\_REQUEST
  - ssl.h, [651](#)
- SSL\_HS\_CERTIFICATE\_VERIFY
  - ssl.h, [651](#)
- SSL\_HS\_CLIENT\_HELLO
  - ssl.h, [651](#)
- SSL\_HS\_CLIENT\_KEY\_EXCHANGE
  - ssl.h, [651](#)
- SSL\_HS\_FINISHED
  - ssl.h, [651](#)
- SSL\_HS\_HELLO\_REQUEST
  - ssl.h, [651](#)
- SSL\_HS\_SERVER\_HELLO
  - ssl.h, [651](#)
- SSL\_HS\_SERVER\_HELLO\_DONE
  - ssl.h, [651](#)
- SSL\_HS\_SERVER\_KEY\_EXCHANGE
  - ssl.h, [652](#)
- ssl\_init
  - ssl.h, [661](#)
  - ssl\_tls.c, [809](#)
- SSL\_IS\_CLIENT
  - ssl.h, [652](#)
- SSL\_IS\_SERVER
  - ssl.h, [652](#)
- ssl\_mac\_md5
  - ssl\_tls.c, [809](#)
- ssl\_mac\_sha1
  - ssl\_tls.c, [809](#)
- SSL\_MAJOR\_VERSION\_3
  - ssl.h, [652](#)
- SSL\_MAX\_CONTENT\_LEN
  - ssl.h, [652](#)
- SSL\_MINOR\_VERSION\_0
  - ssl.h, [652](#)
- SSL\_MINOR\_VERSION\_1
  - ssl.h, [652](#)
- SSL\_MINOR\_VERSION\_2
  - ssl.h, [653](#)
- SSL\_MSG\_ALERT
  - ssl.h, [653](#)
- SSL\_MSG\_APPLICATION\_DATA
  - ssl.h, [653](#)
- SSL\_MSG\_CHANGE\_CIPHER\_SPEC
  - ssl.h, [653](#)
- SSL\_MSG\_HANDSHAKE
  - ssl.h, [653](#)
- ssl\_parse\_certificate
  - ssl.h, [662](#)
  - ssl\_tls.c, [810](#)
- ssl\_parse\_certificate\_request
  - ssl\_cli.c, [798](#)
- ssl\_parse\_certificate\_verify
  - ssl\_srv.c, [801](#)
- ssl\_parse\_change\_cipher\_spec
  - ssl.h, [662](#)
  - ssl\_tls.c, [810](#)
- ssl\_parse\_client\_hello
  - ssl\_srv.c, [802](#)
- ssl\_parse\_client\_key\_exchange
  - ssl\_srv.c, [802](#)
- ssl\_parse\_finished
  - ssl.h, [662](#)
  - ssl\_tls.c, [810](#)
- ssl\_parse\_server\_hello
  - ssl\_cli.c, [798](#)

- ssl\_parse\_server\_hello\_done
  - ssl\_cli.c, 799
- ssl\_parse\_server\_key\_exchange
  - ssl\_cli.c, 799
- ssl\_read
  - ssl.h, 662
  - ssl\_tls.c, 810
- ssl\_read\_record
  - ssl.h, 663
  - ssl\_tls.c, 811
- SSL\_RSA\_AES\_128\_SHA
  - ssl.h, 653
- SSL\_RSA\_AES\_256\_SHA
  - ssl.h, 653
- SSL\_RSA\_DES\_168\_SHA
  - ssl.h, 653
- SSL\_RSA\_RC4\_128\_MD5
  - ssl.h, 654
- SSL\_RSA\_RC4\_128\_SHA
  - ssl.h, 654
- ssl\_server.c
  - cur, 848
  - DEBUG\_LEVEL, 847
  - HTTP\_RESPONSE, 847
  - main, 848
  - my\_ciphers, 848
  - my\_debug, 848
  - my\_dhm\_G, 849
  - my\_dhm\_P, 849
  - my\_get\_session, 848
  - my\_set\_session, 848
  - prv, 849
  - s\_list\_1st, 849
- ssl\_session
  - ssl.h, 658
- ssl\_set\_authmode
  - ssl.h, 663
  - ssl\_tls.c, 811
- ssl\_set\_bio
  - ssl.h, 664
  - ssl\_tls.c, 811
- ssl\_set\_ca\_chain
  - ssl.h, 664
  - ssl\_tls.c, 812
- ssl\_set\_ciphers
  - ssl.h, 664
  - ssl\_tls.c, 812
- ssl\_set\_dbg
  - ssl.h, 665
  - ssl\_tls.c, 812
- ssl\_set\_dh\_param
  - ssl.h, 665
  - ssl\_tls.c, 813
- ssl\_set\_endpoint
  - ssl.h, 665
  - ssl\_tls.c, 813
- ssl\_set\_hostname
  - ssl.h, 665
  - ssl\_tls.c, 813
- ssl\_set\_own\_cert
  - ssl.h, 666
  - ssl\_tls.c, 814
- ssl\_set\_rng
  - ssl.h, 666
  - ssl\_tls.c, 814
- ssl\_set\_scb
  - ssl.h, 666
  - ssl\_tls.c, 814
- ssl\_set\_session
  - ssl.h, 667
  - ssl\_tls.c, 815
- ssl\_srv.c
  - ssl\_handshake\_server, 801
  - ssl\_parse\_certificate\_verify, 801
  - ssl\_parse\_client\_hello, 802
  - ssl\_parse\_client\_key\_exchange, 802
  - ssl\_write\_certificate\_request, 802
  - ssl\_write\_server\_hello, 802
  - ssl\_write\_server\_hello\_done, 803
  - ssl\_write\_server\_key\_exchange, 803
- ssl\_states
  - ssl.h, 658
- ssl\_test
  - ssl\_test.c, 857
- ssl\_test.c
  - COMMAND\_BOTH, 854
  - COMMAND\_READ, 854
  - COMMAND\_WRITE, 854
  - DFL\_BUFFER\_SIZE, 854
  - DFL\_COMMAND, 854
  - DFL\_CONN\_TIMEOUT, 854
  - DFL\_DEBUG\_LEVEL, 854
  - DFL\_FORCE\_CIPHER, 854
  - DFL\_IOMODE, 855
  - DFL\_MAX\_BYTES, 855
  - DFL\_MAX\_CONNECTIONS, 855
  - DFL\_OPMODE, 855
  - DFL\_SERVER\_NAME, 855
  - DFL\_SERVER\_PORT, 855
  - DFL\_SESSION\_LIFETIME, 855
  - DFL\_SESSION\_REUSE, 855
  - dhm\_G, 857
  - dhm\_P, 858
  - IOMODE\_BLOCK, 855
  - IOMODE\_NONBLOCK, 856
  - lcppm5, 857
  - main, 857
  - my\_debug, 857

- OPMODE\_CLIENT, 856
- OPMODE\_NONE, 856
- OPMODE\_SERVER, 856
- server\_fd, 858
- ssl\_test, 857
- USAGE, 856
- ssl\_tls.c
  - ssl\_calc\_finished, 806
  - ssl\_calc\_verify, 806
  - ssl\_close\_notify, 806
  - ssl\_decrypt\_buf, 806
  - ssl\_default\_ciphers, 817
  - ssl\_derive\_keys, 807
  - ssl\_encrypt\_buf, 807
  - ssl\_fetch\_input, 807
  - ssl\_flush\_output, 807
  - ssl\_free, 807
  - ssl\_get\_bytes\_avail, 808
  - ssl\_get\_cipher, 808
  - ssl\_get\_verify\_result, 808
  - ssl\_handshake, 809
  - ssl\_init, 809
  - ssl\_mac\_md5, 809
  - ssl\_mac\_sha1, 809
  - ssl\_parse\_certificate, 810
  - ssl\_parse\_change\_cipher\_spec, 810
  - ssl\_parse\_finished, 810
  - ssl\_read, 810
  - ssl\_read\_record, 811
  - ssl\_set\_authmode, 811
  - ssl\_set\_bio, 811
  - ssl\_set\_ca\_chain, 812
  - ssl\_set\_ciphers, 812
  - ssl\_set\_dbg, 812
  - ssl\_set\_dh\_param, 813
  - ssl\_set\_endpoint, 813
  - ssl\_set\_hostname, 813
  - ssl\_set\_own\_cert, 814
  - ssl\_set\_rng, 814
  - ssl\_set\_scb, 814
  - ssl\_set\_session, 815
  - ssl\_write, 815
  - ssl\_write\_certificate, 816
  - ssl\_write\_change\_cipher\_spec, 816
  - ssl\_write\_finished, 816
  - ssl\_write\_record, 816
  - tls1\_prf, 817
- SSL\_VERIFY\_NONE
  - ssl.h, 654
- SSL\_VERIFY\_OPTIONAL
  - ssl.h, 654
- SSL\_VERIFY\_REQUIRED
  - ssl.h, 654
- ssl\_write
  - ssl.h, 667
  - ssl\_tls.c, 815
- ssl\_write\_certificate
  - ssl.h, 668
  - ssl\_tls.c, 816
- ssl\_write\_certificate\_request
  - ssl\_srv.c, 802
- ssl\_write\_certificate\_verify
  - ssl\_cli.c, 799
- ssl\_write\_change\_cipher\_spec
  - ssl.h, 668
  - ssl\_tls.c, 816
- ssl\_write\_client\_hello
  - ssl\_cli.c, 799
- ssl\_write\_client\_key\_exchange
  - ssl\_cli.c, 800
- ssl\_write\_finished
  - ssl.h, 668
  - ssl\_tls.c, 816
- ssl\_write\_record
  - ssl.h, 668
  - ssl\_tls.c, 816
- ssl\_write\_server\_hello
  - ssl\_srv.c, 802
- ssl\_write\_server\_hello\_done
  - ssl\_srv.c, 803
- ssl\_write\_server\_key\_exchange
  - ssl\_srv.c, 803
- stack\_size
  - agency\_s, 38
  - mc\_platform\_s, 115
  - MCAgencyOptions\_s, 120
- start
  - \_hr\_time, 9
  - \_ssl\_session, 23
- state
  - \_ssl\_context, 20
  - fipa\_comm\_protocol\_t, 82
  - md2\_context, 121
  - md4\_context, 123
  - md5\_context, 125
  - sha1\_context, 156
  - sha2\_context, 158
  - sha4\_context, 160
- STATE\_C\_COMMENT
  - mxmldoc.c, 524
- STATE\_CHARACTER
  - mxmldoc.c, 524
- STATE\_CXX\_COMMENT
  - mxmldoc.c, 524
- STATE\_IDENTIFIER
  - mxmldoc.c, 524
- STATE\_NONE
  - mxmldoc.c, 525



- STATE\_PREPROCESSOR
  - mxmldoc.c, [525](#)
- STATE\_STRING
  - mxmldoc.c, [525](#)
- stationary\_agent\_info\_s, [162](#)
  - agency, [162](#)
  - agent, [162](#)
  - args, [162](#)
  - attr, [162](#)
  - thread, [162](#)
- stationary\_agent\_info\_t
  - libmc.h, [321](#)
- str
  - fipa\_number\_s, [92](#)
  - fipa\_url\_s, [94](#)
- strdup
  - mxml-2.2.2/winconfig.h, [533](#)
  - winconfig.h, [535](#)
- string
  - fipa\_expression\_s::content\_u, [60](#)
  - mxml\_text\_s, [145](#)
- strtok\_r
  - winconfig.h, [535](#)
- STRUCT
  - acc.h, [220](#), [221](#)
  - ams.h, [245](#)
  - df.h, [270](#)
  - macros.h, [375](#)
  - xml\_parser.h, [403](#)
- struct.cxx
  - foo\_t, [531](#)
- subject
  - \_x509\_cert, [27](#)
  - \_x509\_raw, [31](#)
- subject\_id
  - \_x509\_cert, [27](#)
- subject\_raw
  - \_x509\_cert, [27](#)
- subpubkey
  - \_x509\_raw, [31](#)
- SUBSCRIBE
  - message.h, [384](#)
- SWAP
  - des.c, [729](#)
  - havege.c, [743](#)
- SWITCHING\_PROTOCOLS
  - mtp\_http.h, [390](#)
- sync\_list.c
  - syncListAddNode, [471](#)
  - syncListDelete, [471](#)
  - syncListDestroy, [471](#)
  - syncListFind, [471](#)
  - syncListInit, [472](#)
  - syncListNew, [472](#)
  - syncListNodeDestroy, [472](#)
  - syncListNodeInit, [472](#)
  - syncListNodeNew, [472](#)
  - syncListRemove, [472](#)
- sync\_list.h
  - syncList\_p, [474](#)
  - syncList\_t, [474](#)
  - syncListAddNode, [475](#)
  - syncListDelete, [475](#)
  - syncListDestroy, [475](#)
  - syncListFind, [475](#)
  - syncListGet, [475](#)
  - syncListInit, [475](#)
  - syncListNew, [475](#)
  - syncListNode\_p, [474](#)
  - syncListNode\_t, [474](#)
  - syncListNodeDestroy, [476](#)
  - syncListNodeInit, [476](#)
  - syncListNodeNew, [476](#)
  - syncListRemove, [476](#)
- syncList
  - mc\_platform\_s, [115](#)
- syncList\_p
  - sync\_list.h, [474](#)
- syncList\_s, [163](#)
  - giant\_lock, [163](#)
  - list, [163](#)
  - lock, [163](#)
  - size, [163](#)
- syncList\_t
  - sync\_list.h, [474](#)
- syncListAddNode
  - sync\_list.c, [471](#)
  - sync\_list.h, [475](#)
- syncListDelete
  - sync\_list.c, [471](#)
  - sync\_list.h, [475](#)
- syncListDestroy
  - sync\_list.c, [471](#)
  - sync\_list.h, [475](#)
- syncListFind
  - sync\_list.c, [471](#)
  - sync\_list.h, [475](#)
- syncListGet
  - sync\_list.h, [475](#)
- syncListInit
  - sync\_list.c, [472](#)
  - sync\_list.h, [475](#)
- syncListNew
  - sync\_list.c, [472](#)
  - sync\_list.h, [475](#)
- syncListNode\_p
  - sync\_list.h, [474](#)
- syncListNode\_s, [164](#)

- cond, 164
- id, 164
- lock, 164
- sem, 164
- signalled, 164
- syncListNode\_t
  - sync\_list.h, 474
- syncListNodeDestroy
  - sync\_list.c, 472
  - sync\_list.h, 476
- syncListNodeInit
  - sync\_list.c, 472
  - sync\_list.h, 476
- syncListNodeNew
  - sync\_list.c, 472
  - sync\_list.h, 476
- syncListRemove
  - sync\_list.c, 472
  - sync\_list.h, 476
- t\_dbl
  - bignum.h, 567
- t\_int
  - bignum.h, 567
- tag
  - \_x509\_buf, 24
- target
  - message\_s, 128
  - mtp\_http\_s, 136
- task\_progress
  - agent\_datastate\_s, 41
- tasks
  - agent\_datastate\_s, 41
- tbs
  - \_x509\_cert, 27
  - \_x509\_raw, 32
- tbs\_signalg
  - \_x509\_raw, 32
- test\_ca\_cert
  - certs.c, 720
  - certs.h, 581
- test\_ca\_key
  - certs.c, 720
  - certs.h, 581
- test\_ca\_pwd
  - certs.c, 720
  - certs.h, 581
- test\_cli\_cert
  - certs.c, 721
  - certs.h, 581
- test\_cli\_key
  - certs.c, 721
  - certs.h, 581
- test\_srv\_cert
  - certs.c, 722
  - certs.h, 581
- test\_srv\_key
  - certs.c, 722
  - certs.h, 582
- testmxml.c
  - main, 532
  - type\_cb, 532
  - whitespace\_cb, 532
- text
  - mxml\_value\_u, 147
- thread
  - cmd\_prompt\_s, 56
  - df.h, 271
  - stationary\_agent\_info\_s, 162
- THREAD\_CANCEL
  - macros.h, 375
- THREAD\_CREATE
  - macros.h, 375
- THREAD\_DETACH
  - macros.h, 375
- THREAD\_EXIT
  - macros.h, 375
- THREAD\_JOIN
  - macros.h, 375
- threads
  - agency\_s, 38
  - MCAgencyOptions\_s, 120
- time\_out
  - fipa\_comm\_protocol\_cn\_t, 80
- timeout
  - \_ssl\_context, 20
- timing.c
  - alarmed, 819
  - get\_timer, 818
  - hardclock, 819
  - hardclock\_init, 819
  - m\_sleep, 819
  - set\_alarm, 819
  - sighandler, 819
  - tv\_init, 819
- timing.h
  - alarmed, 671
  - get\_timer, 670
  - hardclock, 670
  - m\_sleep, 671
  - set\_alarm, 671
- tls1\_prf
  - ssl\_tls.c, 817
- TLS\_EXT\_SERVERNAME
  - ssl.h, 654
- TLS\_EXT\_SERVERNAME\_HOSTNAME
  - ssl.h, 654
- to

- fipa\_acl\_Param\_s, [74](#)
- to\_address
  - message\_s, [129](#)
- total
  - md4\_context, [123](#)
  - md5\_context, [125](#)
  - sha1\_context, [156](#)
  - sha2\_context, [159](#)
  - sha4\_context, [161](#)
- TST1\_ENTER
  - havege.c, [743](#)
- TST1\_LEAVE
  - havege.c, [743](#)
- TST2\_ENTER
  - havege.c, [743](#)
- TST2\_LEAVE
  - havege.c, [744](#)
- tv\_init
  - timing.c, [819](#)
- type
  - fipa\_comm\_action\_t, [77](#)
  - fipa\_comm\_message\_check\_t, [78](#)
  - fipa\_comm\_protocol\_t, [82](#)
  - fipa\_comm\_reply\_t, [84](#)
  - fipa\_expression\_s, [89](#)
  - mxml\_node\_s, [144](#)
- type\_cb
  - testmxml.c, [532](#)
- TYPES
  - df.h, [267](#)
- udplisten\_Thread
  - acc.c, [171](#)
  - acc.h, [221](#)
- UDPPORT
  - acc.c, [169](#)
- UL64
  - sha4.h, [642](#)
- UNAUTHORIZED
  - mtp\_http.h, [390](#)
- UNORDERED\_COLLECTION
  - mtp\_http.h, [391](#)
- UNPROCESSABLE\_ENTITY
  - mtp\_http.h, [391](#)
- UNSUPPORTED\_MEDIA\_TYPE
  - mtp\_http.h, [390](#)
- update\_comment
  - mxml.doc.c, [526](#)
- update\_name
  - message\_s, [129](#)
- update\_num
  - message\_s, [129](#)
- UPGRADE\_REQUIRED
  - mtp\_http.h, [391](#)
- urls
  - fipa\_url\_sequence\_s, [95](#)
- USAGE
  - aescrypt2.c, [830](#)
  - ssl\_test.c, [856](#)
- user\_agent
  - mtp\_http\_s, [137](#)
- v3\_ext
  - \_x509\_cert, [27](#)
- val
  - \_x509\_name, [29](#)
- valid\_from
  - \_x509\_cert, [27](#)
- valid\_to
  - \_x509\_cert, [27](#)
- validity
  - \_x509\_raw, [32](#)
- value
  - mxml\_attr\_s, [138](#)
  - mxml\_node\_s, [144](#)
- var\_name
  - agent\_task\_s, [50](#)
- ver
  - rsa\_context, [155](#)
- verify\_result
  - \_ssl\_context, [21](#)
- version
  - \_x509\_cert, [27](#)
  - \_x509\_raw, [32](#)
- vsnprintf
  - mxml-2.2.2/winconfig.h, [534](#)
- waiting
  - df.h, [271](#)
- waiting\_cond
  - df.h, [272](#)
- waiting\_lock
  - df.h, [272](#)
- WAKE\_QUEUE
  - macros.h, [376](#)
- WALK
  - havege\_state, [102](#)
- WARN
  - macros.h, [376](#)
- wg\_code
  - agent\_s, [48](#)
- whitespace
  - mxml\_text\_s, [145](#)
- whitespace\_cb
  - testmxml.c, [532](#)
  - xml\_helper.c, [864](#)
  - xml\_helper.h, [398](#)
- winconfig.h

- PACKAGE\_STRING, [535](#)
- PACKAGE\_VERSION, [535](#)
- strdup, [535](#)
- strtok\_r, [535](#)
- word
  - fipa\_expression\_s::content\_u, [60](#)
- write\_documentation
  - mxmldoc.c, [526](#)
- write\_element
  - mxmldoc.c, [526](#)
- write\_flag
  - mc\_rwlock\_s, [117](#)
- write\_request
  - mc\_rwlock\_s, [117](#)
- write\_string
  - mxmldoc.c, [526](#)
- ws\_cb
  - mxmldoc.c, [527](#)
- X
  - dhm\_context, [65](#)
- x
  - arc4\_context, [53](#)
- x509.h
  - ASN1\_BIT\_STRING, [674](#)
  - ASN1\_BMP\_STRING, [674](#)
  - ASN1\_BOOLEAN, [674](#)
  - ASN1\_CONSTRUCTED, [675](#)
  - ASN1\_CONTEXT\_SPECIFIC, [675](#)
  - ASN1\_IA5\_STRING, [675](#)
  - ASN1\_INTEGER, [675](#)
  - ASN1\_NULL, [675](#)
  - ASN1\_OCTET\_STRING, [675](#)
  - ASN1\_OID, [675](#)
  - ASN1\_PRIMITIVE, [675](#)
  - ASN1\_PRINTABLE\_STRING, [676](#)
  - ASN1\_SEQUENCE, [676](#)
  - ASN1\_SET, [676](#)
  - ASN1\_T61\_STRING, [676](#)
  - ASN1\_UNIVERSAL\_STRING, [676](#)
  - ASN1\_UTC\_TIME, [676](#)
  - ASN1\_UTF8\_STRING, [676](#)
  - BADCERT\_CN\_MISMATCH, [676](#)
  - BADCERT\_EXPIRED, [676](#)
  - BADCERT\_NOT\_TRUSTED, [677](#)
  - BADCERT\_REVOKED, [677](#)
  - OID\_CN, [677](#)
  - OID\_PKCS1, [677](#)
  - OID\_PKCS1\_RSA, [677](#)
  - OID\_PKCS1\_RSA\_SHA, [677](#)
  - OID\_PKCS9, [677](#)
  - OID\_PKCS9\_EMAIL, [677](#)
  - OID\_X520, [677](#)
  - PEM\_LINE\_LENGTH, [678](#)
  - PKCS9\_EMAIL, [678](#)
  - x509\_buf, [683](#)
  - x509\_cert, [683](#)
  - x509\_free, [683](#)
  - X509\_ISSUER, [678](#)
  - x509\_name, [683](#)
  - x509\_node, [683](#)
  - X509\_OUTPUT\_DER, [678](#)
  - X509\_OUTPUT\_PEM, [678](#)
  - x509\_raw, [683](#)
  - x509\_self\_test, [683](#)
  - X509\_SUBJECT, [678](#)
  - x509\_time, [683](#)
  - x509parse\_cert\_info, [683](#)
  - x509parse\_crt, [683](#)
  - x509parse\_crtfile, [684](#)
  - x509parse\_dn\_gets, [684](#)
  - x509parse\_expired, [685](#)
  - x509parse\_key, [685](#)
  - x509parse\_keyfile, [685](#)
  - x509parse\_verify, [686](#)
  - X520\_COMMON\_NAME, [678](#)
  - X520\_COUNTRY, [678](#)
  - X520\_LOCALITY, [678](#)
  - X520\_ORG\_UNIT, [678](#)
  - X520\_ORGANIZATION, [679](#)
  - X520\_STATE, [679](#)
  - XYSSL\_ERR\_ASN1\_INVALID\_DATA, [679](#)
  - XYSSL\_ERR\_ASN1\_INVALID\_LENGTH, [679](#)
  - XYSSL\_ERR\_ASN1\_LENGTH\_MISMATCH, [679](#)
  - XYSSL\_ERR\_ASN1\_OUT\_OF\_DATA, [679](#)
  - XYSSL\_ERR\_ASN1\_UNEXPECTED\_TAG, [679](#)
  - XYSSL\_ERR\_X509\_CERT\_INVALID\_ALG, [679](#)
  - XYSSL\_ERR\_X509\_CERT\_INVALID\_DATE, [680](#)
  - XYSSL\_ERR\_X509\_CERT\_INVALID\_EXTENSIONS, [680](#)
  - XYSSL\_ERR\_X509\_CERT\_INVALID\_FORMAT, [680](#)
  - XYSSL\_ERR\_X509\_CERT\_INVALID\_NAME, [680](#)
  - XYSSL\_ERR\_X509\_CERT\_INVALID\_PEM, [680](#)
  - XYSSL\_ERR\_X509\_CERT\_INVALID\_PUBKEY, [680](#)
  - XYSSL\_ERR\_X509\_CERT\_INVALID\_SERIAL, [680](#)
  - XYSSL\_ERR\_X509\_CERT\_INVALID\_SIGNATURE, [680](#)

- XYSSL\_ERR\_X509\_CERT\_INVALID\_-  
VERSION, [680](#)
- XYSSL\_ERR\_X509\_CERT\_SIG\_-  
MISMATCH, [681](#)
- XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_-  
PK\_ALG, [681](#)
- XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_-  
SIG\_ALG, [681](#)
- XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_-  
VERSION, [681](#)
- XYSSL\_ERR\_X509\_CERT\_VERIFY\_-  
FAILED, [681](#)
- XYSSL\_ERR\_X509\_FEATURE\_-  
UNAVAILABLE, [681](#)
- XYSSL\_ERR\_X509\_KEY\_INVALID\_-  
ENC\_IV, [681](#)
- XYSSL\_ERR\_X509\_KEY\_INVALID\_-  
FORMAT, [681](#)
- XYSSL\_ERR\_X509\_KEY\_INVALID\_PEM,  
[681](#)
- XYSSL\_ERR\_X509\_KEY\_INVALID\_-  
VERSION, [682](#)
- XYSSL\_ERR\_X509\_KEY\_PASSWORD\_-  
MISMATCH, [682](#)
- XYSSL\_ERR\_X509\_KEY\_PASSWORD\_-  
REQUIRED, [682](#)
- XYSSL\_ERR\_X509\_KEY\_UNKNOWN\_-  
ENC\_ALG, [682](#)
- XYSSL\_ERR\_X509\_POINT\_ERROR, [682](#)
- XYSSL\_ERR\_X509\_VALUE\_TO\_LENGTH,  
[682](#)
- x509\_buf  
x509.h, [683](#)
- x509\_cert  
x509.h, [683](#)
- x509\_des3\_decrypt  
x509parse.c, [823](#)
- x509\_free  
x509.h, [683](#)  
x509parse.c, [823](#)
- x509\_get\_alg  
x509parse.c, [823](#)
- x509\_get\_dates  
x509parse.c, [823](#)
- x509\_get\_ext  
x509parse.c, [824](#)
- x509\_get\_iv  
x509parse.c, [824](#)
- x509\_get\_name  
x509parse.c, [824](#)
- x509\_get\_pubkey  
x509parse.c, [824](#)
- x509\_get\_serial  
x509parse.c, [824](#)
- x509\_get\_sig  
x509parse.c, [825](#)
- x509\_get\_uid  
x509parse.c, [825](#)
- x509\_get\_version  
x509parse.c, [825](#)
- x509\_hash  
x509parse.c, [825](#)
- X509\_ISSUER  
x509.h, [678](#)
- x509\_name  
x509.h, [683](#)
- x509\_node  
x509.h, [683](#)
- X509\_OUTPUT\_DER  
x509.h, [678](#)
- X509\_OUTPUT\_PEM  
x509.h, [678](#)
- x509\_raw  
x509.h, [683](#)
- x509\_self\_test  
x509.h, [683](#)  
x509parse.c, [825](#)
- X509\_SUBJECT  
x509.h, [678](#)
- x509\_time  
x509.h, [683](#)
- x509parse.c  
asn1\_get\_bool, [822](#)  
asn1\_get\_int, [822](#)  
asn1\_get\_len, [822](#)  
asn1\_get\_mpi, [822](#)  
asn1\_get\_tag, [823](#)  
x509\_des3\_decrypt, [823](#)  
x509\_free, [823](#)  
x509\_get\_alg, [823](#)  
x509\_get\_dates, [823](#)  
x509\_get\_ext, [824](#)  
x509\_get\_iv, [824](#)  
x509\_get\_name, [824](#)  
x509\_get\_pubkey, [824](#)  
x509\_get\_serial, [824](#)  
x509\_get\_sig, [825](#)  
x509\_get\_uid, [825](#)  
x509\_get\_version, [825](#)  
x509\_hash, [825](#)  
x509\_self\_test, [825](#)  
x509parse\_cert\_info, [826](#)  
x509parse\_crt, [826](#)  
x509parse\_crtfile, [826](#)  
x509parse\_dn\_gets, [827](#)  
x509parse\_expired, [827](#)  
x509parse\_key, [827](#)  
x509parse\_keyfile, [828](#)

- x509parse\_verify, 828
- x509parse\_cert\_info
  - x509.h, 683
  - x509parse.c, 826
- x509parse\_crt
  - x509.h, 683
  - x509parse.c, 826
- x509parse\_crtfile
  - x509.h, 684
  - x509parse.c, 826
- x509parse\_dn\_gets
  - x509.h, 684
  - x509parse.c, 827
- x509parse\_expired
  - x509.h, 685
  - x509parse.c, 827
- x509parse\_key
  - x509.h, 685
  - x509parse.c, 827
- x509parse\_keyfile
  - x509.h, 685
  - x509parse.c, 828
- x509parse\_verify
  - x509.h, 686
  - x509parse.c, 828
- X520\_COMMON\_NAME
  - x509.h, 678
- X520\_COUNTRY
  - x509.h, 678
- X520\_LOCALITY
  - x509.h, 678
- X520\_ORG\_UNIT
  - x509.h, 678
- X520\_ORGANIZATION
  - x509.h, 679
- X520\_STATE
  - x509.h, 679
- xml\_agent\_root
  - agent\_datastate\_s, 42
- xml\_compose.c
  - agent\_xml\_compose, 860
  - agent\_xml\_compose\_\_agent\_code, 860
  - agent\_xml\_compose\_\_agent\_data, 860
  - agent\_xml\_compose\_\_create\_row\_nodes, 861
  - agent\_xml\_compose\_\_data, 861
  - agent\_xml\_compose\_\_gaf\_message, 861
  - agent\_xml\_compose\_\_home, 861
  - agent\_xml\_compose\_\_message, 861
  - agent\_xml\_compose\_\_mobile\_agent, 862
  - agent\_xml\_compose\_\_name, 862
  - agent\_xml\_compose\_\_owner, 862
  - agent\_xml\_compose\_\_row, 862
  - agent\_xml\_compose\_\_task, 862
  - agent\_xml\_compose\_\_tasks, 862
  - agent\_xml\_compose\_\_wg\_code, 863
- xml\_compose.h
  - agent\_xml\_compose, 394
  - agent\_xml\_compose\_\_agent\_code, 394
  - agent\_xml\_compose\_\_agent\_data, 394
  - agent\_xml\_compose\_\_create\_row\_nodes, 395
  - agent\_xml\_compose\_\_data, 395
  - agent\_xml\_compose\_\_gaf\_message, 395
  - agent\_xml\_compose\_\_home, 395
  - agent\_xml\_compose\_\_message, 395
  - agent\_xml\_compose\_\_mobile\_agent, 395
  - agent\_xml\_compose\_\_name, 396
  - agent\_xml\_compose\_\_owner, 396
  - agent\_xml\_compose\_\_row, 396
  - agent\_xml\_compose\_\_task, 396
  - agent\_xml\_compose\_\_tasks, 396
  - agent\_xml\_compose\_\_wg\_code, 396
- xml\_find\_sibling
  - xml\_helper.c, 864
  - xml\_helper.h, 398
- xml\_get\_cdata
  - xml\_helper.c, 864
  - xml\_helper.h, 398
- xml\_get\_child
  - xml\_helper.c, 864
  - xml\_helper.h, 398
- xml\_get\_deep\_child
  - xml\_helper.c, 865
  - xml\_helper.h, 399
- xml\_get\_element\_name
  - xml\_helper.c, 865
  - xml\_helper.h, 399
- xml\_get\_next\_element
  - xml\_helper.c, 865
  - xml\_helper.h, 399
- xml\_get\_text
  - xml\_helper.c, 865
  - xml\_helper.h, 399
- xml\_helper.c
  - whitespace\_cb, 864
  - xml\_find\_sibling, 864
  - xml\_get\_cdata, 864
  - xml\_get\_child, 864
  - xml\_get\_deep\_child, 865
  - xml\_get\_element\_name, 865
  - xml\_get\_next\_element, 865
  - xml\_get\_text, 865
  - xml\_new\_cdata, 865
- xml\_helper.h
  - whitespace\_cb, 398
  - xml\_find\_sibling, 398
  - xml\_get\_cdata, 398
  - xml\_get\_child, 398
  - xml\_get\_deep\_child, 399

- xml\_get\_element\_name, 399
  - xml\_get\_next\_element, 399
  - xml\_get\_text, 399
  - xml\_new\_cdata, 399
- xml\_new\_cdata
  - xml\_helper.c, 865
  - xml\_helper.h, 399
- xml\_parser.c
  - \_XOPEN\_SOURCE, 866
  - agent\_return\_xml\_parse, 867
  - agent\_xml\_parse, 867
  - agent\_xml\_parse\_\_agent\_code, 867
  - agent\_xml\_parse\_\_agent\_data, 867
  - agent\_xml\_parse\_\_data, 867
  - agent\_xml\_parse\_\_fill\_row\_data, 867
  - agent\_xml\_parse\_\_home, 868
  - agent\_xml\_parse\_\_mobile\_agent, 868
  - agent\_xml\_parse\_\_name, 868
  - agent\_xml\_parse\_\_owner, 868
  - agent\_xml\_parse\_\_row, 868
  - agent\_xml\_parse\_\_sender, 868
  - agent\_xml\_parse\_\_task, 869
  - agent\_xml\_parse\_\_tasks, 869
  - agent\_xml\_parse\_\_wg\_code, 869
  - message\_xml\_parse, 869
  - message\_xml\_parse\_\_message, 869
- xml\_parser.h
  - agent\_xml\_parse, 400
  - agent\_xml\_parse\_\_agent\_code, 400
  - agent\_xml\_parse\_\_agent\_data, 400
  - agent\_xml\_parse\_\_data, 401
  - agent\_xml\_parse\_\_fill\_row\_data, 401
  - agent\_xml\_parse\_\_home, 401
  - agent\_xml\_parse\_\_mobile\_agent, 401
  - agent\_xml\_parse\_\_name, 401
  - agent\_xml\_parse\_\_owner, 402
  - agent\_xml\_parse\_\_row, 402
  - agent\_xml\_parse\_\_sender, 402
  - agent\_xml\_parse\_\_task, 402
  - agent\_xml\_parse\_\_tasks, 402
  - agent\_xml\_parse\_\_wg\_code, 402
  - message\_xml\_parse, 403
  - message\_xml\_parse\_\_message, 403
  - STRUCT, 403
- xml\_payload
  - message\_s, 129
- xml\_root
  - agent\_datastate\_s, 42
  - message\_s, 129
- XTIME
  - aes.c, 690
- XYSSL\_AES\_C
  - security/xyssl-0.9/include/xyssl/config.h, 488
- XYSSL\_ARC4\_C
  - security/xyssl-0.9/include/xyssl/config.h, 488
- XYSSL\_BASE64\_C
  - security/xyssl-0.9/include/xyssl/config.h, 488
- XYSSL\_BIGNUM\_C
  - security/xyssl-0.9/include/xyssl/config.h, 488
- xyssl\_ca\_crt
  - certs.c, 723
  - certs.h, 582
- XYSSL\_CERTS\_C
  - security/xyssl-0.9/include/xyssl/config.h, 488
- XYSSL\_DEBUG\_C
  - security/xyssl-0.9/include/xyssl/config.h, 489
- XYSSL\_DEBUG\_MSG
  - security/xyssl-0.9/include/xyssl/config.h, 489
- XYSSL\_DES\_C
  - security/xyssl-0.9/include/xyssl/config.h, 489
- XYSSL\_DHM\_C
  - security/xyssl-0.9/include/xyssl/config.h, 489
- XYSSL\_ERR\_ASN1\_INVALID\_DATA
  - x509.h, 679
- XYSSL\_ERR\_ASN1\_INVALID\_LENGTH
  - x509.h, 679
- XYSSL\_ERR\_ASN1\_LENGTH\_MISMATCH
  - x509.h, 679
- XYSSL\_ERR\_ASN1\_OUT\_OF\_DATA
  - x509.h, 679
- XYSSL\_ERR\_ASN1\_UNEXPECTED\_TAG
  - x509.h, 679
- XYSSL\_ERR\_BASE64\_BUFFER\_TOO\_SMALL
  - base64.h, 561
- XYSSL\_ERR\_BASE64\_INVALID\_CHARACTER
  - base64.h, 561
- XYSSL\_ERR\_DHM\_BAD\_INPUT\_DATA
  - dhm.h, 592
- XYSSL\_ERR\_DHM\_CALC\_SECRET\_FAILED
  - dhm.h, 592
- XYSSL\_ERR\_DHM\_MAKE\_PARAMS\_FAILED
  - dhm.h, 592
- XYSSL\_ERR\_DHM\_MAKE\_PUBLIC\_FAILED
  - dhm.h, 592
- XYSSL\_ERR\_DHM\_READ\_PARAMS\_FAILED
  - dhm.h, 592
- XYSSL\_ERR\_DHM\_READ\_PUBLIC\_FAILED
  - dhm.h, 592
- XYSSL\_ERR\_MPI\_BAD\_INPUT\_DATA
  - bignum.h, 566
- XYSSL\_ERR\_MPI\_BUFFER\_TOO\_SMALL
  - bignum.h, 566
- XYSSL\_ERR\_MPI\_DIVISION\_BY\_ZERO
  - bignum.h, 566
- XYSSL\_ERR\_MPI\_FILE\_IO\_ERROR
  - bignum.h, 566
- XYSSL\_ERR\_MPI\_INVALID\_CHARACTER
  - bignum.h, 566

- XYSSL\_ERR\_MPI\_NEGATIVE\_VALUE  
bignum.h, 566
- XYSSL\_ERR\_MPI\_NOT\_ACCEPTABLE  
bignum.h, 566
- XYSSL\_ERR\_NET\_ACCEPT\_FAILED  
net.h, 612
- XYSSL\_ERR\_NET\_BIND\_FAILED  
net.h, 612
- XYSSL\_ERR\_NET\_CONN\_RESET  
net.h, 612
- XYSSL\_ERR\_NET\_CONNECT\_FAILED  
net.h, 612
- XYSSL\_ERR\_NET\_LISTEN\_FAILED  
net.h, 612
- XYSSL\_ERR\_NET\_RECV\_FAILED  
net.h, 612
- XYSSL\_ERR\_NET\_SEND\_FAILED  
net.h, 612
- XYSSL\_ERR\_NET\_SOCKET\_FAILED  
net.h, 612
- XYSSL\_ERR\_NET\_TRY\_AGAIN  
net.h, 612
- XYSSL\_ERR\_NET\_UNKNOWN\_HOST  
net.h, 613
- XYSSL\_ERR\_RSA\_BAD\_INPUT\_DATA  
rsa.h, 624
- XYSSL\_ERR\_RSA\_INVALID\_PADDING  
rsa.h, 624
- XYSSL\_ERR\_RSA\_KEY\_CHECK\_FAILED  
rsa.h, 624
- XYSSL\_ERR\_RSA\_KEY\_GEN\_FAILED  
rsa.h, 624
- XYSSL\_ERR\_RSA\_PRIVATE\_FAILED  
rsa.h, 625
- XYSSL\_ERR\_RSA\_PUBLIC\_FAILED  
rsa.h, 625
- XYSSL\_ERR\_RSA\_VERIFY\_FAILED  
rsa.h, 625
- XYSSL\_ERR\_SSL\_BAD\_HS\_CERTIFICATE  
ssl.h, 654
- XYSSL\_ERR\_SSL\_BAD\_HS\_CERTIFICATE\_-  
REQUEST  
ssl.h, 654
- XYSSL\_ERR\_SSL\_BAD\_HS\_CERTIFICATE\_-  
VERIFY  
ssl.h, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_CHANGE\_-  
CIPHER\_SPEC  
ssl.h, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_CLIENT\_HELLO  
ssl.h, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_CLIENT\_KEY\_-  
EXCHANGE  
ssl.h, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_FINISHED  
ssl.h, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_HELLO  
ssl.h, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_-  
HELLO\_DONE  
ssl.h, 655
- XYSSL\_ERR\_SSL\_BAD\_HS\_SERVER\_KEY\_-  
EXCHANGE  
ssl.h, 655
- XYSSL\_ERR\_SSL\_BAD\_INPUT\_DATA  
ssl.h, 655
- XYSSL\_ERR\_SSL\_CA\_CHAIN\_REQUIRED  
ssl.h, 656
- XYSSL\_ERR\_SSL\_CERTIFICATE\_REQUIRED  
ssl.h, 656
- XYSSL\_ERR\_SSL\_CERTIFICATE\_TOO\_-  
LARGE  
ssl.h, 656
- XYSSL\_ERR\_SSL\_FATAL\_ALERT\_MESSAGE  
ssl.h, 656
- XYSSL\_ERR\_SSL\_FEATURE\_UNAVAILABLE  
ssl.h, 656
- XYSSL\_ERR\_SSL\_INVALID\_MAC  
ssl.h, 656
- XYSSL\_ERR\_SSL\_INVALID\_MODULUS\_SIZE  
ssl.h, 656
- XYSSL\_ERR\_SSL\_INVALID\_RECORD  
ssl.h, 656
- XYSSL\_ERR\_SSL\_NO\_CIPHER\_CHOSEN  
ssl.h, 657
- XYSSL\_ERR\_SSL\_NO\_CLIENT\_CERTIFICATE  
ssl.h, 657
- XYSSL\_ERR\_SSL\_NO\_SESSION\_FOUND  
ssl.h, 657
- XYSSL\_ERR\_SSL\_PEER\_CLOSE\_NOTIFY  
ssl.h, 657
- XYSSL\_ERR\_SSL\_PEER\_VERIFY\_FAILED  
ssl.h, 657
- XYSSL\_ERR\_SSL\_PRIVATE\_KEY\_REQUIRED  
ssl.h, 657
- XYSSL\_ERR\_SSL\_UNEXPECTED\_MESSAGE  
ssl.h, 657
- XYSSL\_ERR\_SSL\_UNKNOWN\_CIPHER  
ssl.h, 657
- XYSSL\_ERR\_X509\_CERT\_INVALID\_ALG  
x509.h, 679
- XYSSL\_ERR\_X509\_CERT\_INVALID\_DATE  
x509.h, 680
- XYSSL\_ERR\_X509\_CERT\_INVALID\_-  
EXTENSIONS  
x509.h, 680
- XYSSL\_ERR\_X509\_CERT\_INVALID\_FORMAT  
x509.h, 680



- XYSSL\_ERR\_X509\_CERT\_INVALID\_NAME  
x509.h, [680](#)
- XYSSL\_ERR\_X509\_CERT\_INVALID\_PEM  
x509.h, [680](#)
- XYSSL\_ERR\_X509\_CERT\_INVALID\_PUBKEY  
x509.h, [680](#)
- XYSSL\_ERR\_X509\_CERT\_INVALID\_SERIAL  
x509.h, [680](#)
- XYSSL\_ERR\_X509\_CERT\_INVALID\_-  
SIGNATURE  
x509.h, [680](#)
- XYSSL\_ERR\_X509\_CERT\_INVALID\_VERSION  
x509.h, [680](#)
- XYSSL\_ERR\_X509\_CERT\_SIG\_MISMATCH  
x509.h, [681](#)
- XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_PK\_-  
ALG  
x509.h, [681](#)
- XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_SIG\_-  
ALG  
x509.h, [681](#)
- XYSSL\_ERR\_X509\_CERT\_UNKNOWN\_-  
VERSION  
x509.h, [681](#)
- XYSSL\_ERR\_X509\_CERT\_VERIFY\_FAILED  
x509.h, [681](#)
- XYSSL\_ERR\_X509\_FEATURE\_UNAVAILABLE  
x509.h, [681](#)
- XYSSL\_ERR\_X509\_KEY\_INVALID\_ENC\_IV  
x509.h, [681](#)
- XYSSL\_ERR\_X509\_KEY\_INVALID\_FORMAT  
x509.h, [681](#)
- XYSSL\_ERR\_X509\_KEY\_INVALID\_PEM  
x509.h, [681](#)
- XYSSL\_ERR\_X509\_KEY\_INVALID\_VERSION  
x509.h, [682](#)
- XYSSL\_ERR\_X509\_KEY\_PASSWORD\_-  
MISMATCH  
x509.h, [682](#)
- XYSSL\_ERR\_X509\_KEY\_PASSWORD\_-  
REQUIRED  
x509.h, [682](#)
- XYSSL\_ERR\_X509\_KEY\_UNKNOWN\_ENC\_-  
ALG  
x509.h, [682](#)
- XYSSL\_ERR\_X509\_POINT\_ERROR  
x509.h, [682](#)
- XYSSL\_ERR\_X509\_VALUE\_TO\_LENGTH  
x509.h, [682](#)
- XYSSL\_GENPRIME  
security/xyssl-0.9/include/xyssl/config.h, [489](#)
- XYSSL\_HAVE\_ASM  
security/xyssl-0.9/include/xyssl/config.h, [489](#)
- XYSSL\_HAVEGE\_C  
security/xyssl-0.9/include/xyssl/config.h, [489](#)
- XYSSL\_MD5\_C  
security/xyssl-0.9/include/xyssl/config.h, [489](#)
- XYSSL\_NET\_C  
security/xyssl-0.9/include/xyssl/config.h, [489](#)
- XYSSL\_PADLOCK\_C  
security/xyssl-0.9/include/xyssl/config.h, [489](#)
- XYSSL\_RSA\_C  
security/xyssl-0.9/include/xyssl/config.h, [489](#)
- XYSSL\_SELF\_TEST  
security/xyssl-0.9/include/xyssl/config.h, [490](#)
- XYSSL\_SHA1\_C  
security/xyssl-0.9/include/xyssl/config.h, [490](#)
- XYSSL\_SHA2\_C  
security/xyssl-0.9/include/xyssl/config.h, [490](#)
- XYSSL\_SHA4\_C  
security/xyssl-0.9/include/xyssl/config.h, [490](#)
- XYSSL\_SSL\_CLI\_C  
security/xyssl-0.9/include/xyssl/config.h, [490](#)
- XYSSL\_SSL\_SRV\_C  
security/xyssl-0.9/include/xyssl/config.h, [490](#)
- XYSSL\_SSL\_TLS\_C  
security/xyssl-0.9/include/xyssl/config.h, [490](#)
- XYSSL\_TIMING\_C  
security/xyssl-0.9/include/xyssl/config.h, [490](#)
- XYSSL\_X509\_PARSE\_C  
security/xyssl-0.9/include/xyssl/config.h, [490](#)
- XYSSL\_X509\_WRITE\_C  
security/xyssl-0.9/include/xyssl/config.h, [490](#)
- y  
arc4\_context, [53](#)
- year  
\_x509\_time, [33](#)  
fipa\_DateTime\_s, [88](#)
- ZERO  
df.h, [267](#)