

Mobile-C Reference Manual

1.10.0

Generated by Doxygen 1.5.4

Tue Jun 17 11:00:27 2008

Contents

1	MobileC	1
1.1	Introduction	1
2	Mobile-C Namespace Index	3
2.1	Mobile-C Namespace List	3
3	Mobile-C Data Structure Index	5
3.1	Mobile-C Data Structures	5
4	Mobile-C File Index	7
4.1	Mobile-C File List	7
5	Mobile-C Page Index	11
5.1	Mobile-C Related Pages	11
6	Mobile-C Namespace Documentation	13
6.1	LibMC Namespace Reference	13
6.2	LibMC::Properties Namespace Reference	14
6.3	System Namespace Reference	15
6.4	System::Collections::Generic Namespace Reference	16
6.5	System::Reflection Namespace Reference	17
6.6	System::Runtime::CompilerServices Namespace Reference	18
6.7	System::Runtime::InteropServices Namespace Reference	19
6.8	System::Text Namespace Reference	20
7	Mobile-C Data Structure Documentation	21
7.1	_hr_time Struct Reference	21
7.2	_x509_buf Struct Reference	22
7.3	_x509_cert Struct Reference	23
7.4	_x509_name Struct Reference	27

7.5	_x509_time Struct Reference	28
7.6	aes_context Struct Reference	30
7.7	agency_s Struct Reference	31
7.8	agent_datastate_s Struct Reference	34
7.9	agent_mailbox_s Struct Reference	37
7.10	agent_s Struct Reference	38
7.11	agent_task_s Struct Reference	43
7.12	arc4_context Struct Reference	46
7.13	barrier_node_s Struct Reference	47
7.14	barrier_queue_s Struct Reference	48
7.15	cmd_prompt_s Struct Reference	49
7.16	command_s Struct Reference	50
7.17	connection_s Struct Reference	51
7.18	des3_context Struct Reference	53
7.19	des_context Struct Reference	54
7.20	dhm_context Struct Reference	55
7.21	dynstring_s Struct Reference	57
7.22	fipa_acl_envelope_Received_s Struct Reference	58
7.23	fipa_acl_envelope_s Struct Reference	60
7.24	fipa_acl_message_s Struct Reference	61
7.25	fipa_acl_Param_s Struct Reference	64
7.26	fipa_agent_identifier_s Struct Reference	66
7.27	fipa_agent_identifier_set_s Struct Reference	67
7.28	fipa_DateTime_s Struct Reference	68
7.29	fipa_expression_s Struct Reference	70
7.30	fipa_expression_s::content_u Union Reference	71
7.31	fipa_message_string_s Struct Reference	72
7.32	fipa_number_s Struct Reference	73
7.33	fipa_string_s Struct Reference	74
7.34	fipa_url_s Struct Reference	75
7.35	fipa_url_sequence_s Struct Reference	76
7.36	fipa_word_s Struct Reference	77
7.37	foo_c Class Reference	78
7.38	foo_s Struct Reference	81
7.39	havege_state Struct Reference	83
7.40	host_id_s Struct Reference	85

7.41 hr_time Struct Reference	86
7.42 interpreter_variable_data_s Struct Reference	87
7.43 list_s Struct Reference	89
7.44 listNode_s Struct Reference	90
7.45 mc_platform_s Struct Reference	91
7.46 mc_rwlock_s Struct Reference	96
7.47 LibMC::MCAclMessage Class Reference	98
7.48 LibMC::MCAgency Class Reference	104
7.49 LibMC::MCAgency::ChOptions_t Struct Reference	132
7.50 LibMC::MCAgency::MCAgency_t Struct Reference	133
7.51 LibMC::MCAgency::MCAgencyOptions_t Struct Reference	135
7.52 MCAgencyOptions_s Struct Reference	136
7.53 LibMC::MCAGENT Class Reference	138
7.54 md2_context Struct Reference	148
7.55 md4_context Struct Reference	150
7.56 md5_context Struct Reference	152
7.57 message_s Struct Reference	154
7.58 mpi Struct Reference	157
7.59 mtp_http_content_s Struct Reference	159
7.60 mtp_http_s Struct Reference	160
7.61 mxml_attr_s Struct Reference	163
7.62 mxml_custom_s Struct Reference	164
7.63 mxml_fdbuf_s Struct Reference	165
7.64 mxml_index_s Struct Reference	166
7.65 mxml_node_s Struct Reference	168
7.66 mxml_text_s Struct Reference	170
7.67 mxml_value_s Struct Reference	171
7.68 mxml_value_u Union Reference	172
7.69 rsa_context Struct Reference	174
7.70 LibMC::Properties::Settings Class Reference	177
7.71 sha1_context Struct Reference	179
7.72 sha2_context Struct Reference	181
7.73 sha4_context Struct Reference	183
7.74 ssl_context Struct Reference	185
7.75 syncList_s Struct Reference	196
7.76 syncListNode_s Struct Reference	197

8 Mobile-C File Documentation	199
8.1 /home/dko/projects/mobilec/trunk/src/acc.c File Reference	199
8.2 /home/dko/projects/mobilec/trunk/src/agent.c File Reference	202
8.3 /home/dko/projects/mobilec/trunk/src/agent_datastate.c File Reference	205
8.4 /home/dko/projects/mobilec/trunk/src/agent_mailbox.c File Reference	206
8.5 /home/dko/projects/mobilec/trunk/src/agent_return_data.c File Reference	208
8.6 /home/dko/projects/mobilec/trunk/src/agent_task.c File Reference	210
8.7 /home/dko/projects/mobilec/trunk/src/ams.c File Reference	211
8.8 /home/dko/projects/mobilec/trunk/src/barrier.c File Reference	213
8.9 /home/dko/projects/mobilec/trunk/src/cmd_prompt.c File Reference	215
8.10 /home/dko/projects/mobilec/trunk/src/connection.c File Reference	218
8.11 /home/dko/projects/mobilec/trunk/src/data_structures.c File Reference	219
8.12 /home/dko/projects/mobilec/trunk/src/df.c File Reference	221
8.13 /home/dko/projects/mobilec/trunk/src/dynstring.c File Reference	225
8.14 /home/dko/projects/mobilec/trunk/src/fipa_acl.c File Reference	226
8.15 /home/dko/projects/mobilec/trunk/src/fipa_envelope.c File Reference	238
8.16 /home/dko/projects/mobilec/trunk/src/include/acc.h File Reference	246
8.17 /home/dko/projects/mobilec/trunk/src/include/agent.h File Reference	249
8.18 /home/dko/projects/mobilec/trunk/src/include/agent_datastate.h File Reference	252
8.19 /home/dko/projects/mobilec/trunk/src/include/agent_lib.h File Reference	254
8.20 /home/dko/projects/mobilec/trunk/src/include/agent_mailbox.h File Reference	263
8.21 /home/dko/projects/mobilec/trunk/src/include/agent_task.h File Reference	265
8.22 /home/dko/projects/mobilec/trunk/src/include/ams.h File Reference	267
8.23 /home/dko/projects/mobilec/trunk/src/include/ap_queue_template.h File Reference	269
8.24 /home/dko/projects/mobilec/trunk/src/include/barrier.h File Reference	272
8.25 /home/dko/projects/mobilec/trunk/src/include/cmd_prompt.h File Reference	275
8.26 /home/dko/projects/mobilec/trunk/src/include/commands.h File Reference	277
8.27 /home/dko/projects/mobilec/trunk/src/include/commands.x.h File Reference	279
8.28 /home/dko/projects/mobilec/trunk/src/include/connection.h File Reference	281
8.29 /home/dko/projects/mobilec/trunk/src/include/data_structures.h File Reference	283
8.30 /home/dko/projects/mobilec/trunk/src/include/df.h File Reference	286
8.31 /home/dko/projects/mobilec/trunk/src/include/df_request.x.h File Reference	292
8.32 /home/dko/projects/mobilec/trunk/src/include/dynstring.h File Reference	293
8.33 /home/dko/projects/mobilec/trunk/src/include/fipa_acl.h File Reference	295
8.34 /home/dko/projects/mobilec/trunk/src/include/fipa_acl_envelope.h File Reference	310
8.35 /home/dko/projects/mobilec/trunk/src/include/host_id.h File Reference	318

8.36	/home/dko/projects/mobilec/trunk/src/include/interpreter_variable_data.h File Reference	319
8.37	/home/dko/projects/mobilec/trunk/src/include/libmc.h File Reference	321
8.38	/home/dko/projects/mobilec/trunk/src/include/macros.h File Reference	371
8.39	/home/dko/projects/mobilec/trunk/src/include/mc_error.h File Reference	382
8.40	/home/dko/projects/mobilec/trunk/src/include/mc_platform.h File Reference	384
8.41	/home/dko/projects/mobilec/trunk/src/include/mc_rwlock.h File Reference	386
8.42	/home/dko/projects/mobilec/trunk/src/include/message.h File Reference	388
8.43	/home/dko/projects/mobilec/trunk/src/include/mtp_http.h File Reference	391
8.44	/home/dko/projects/mobilec/trunk/src/include/xml_compose.h File Reference	397
8.45	/home/dko/projects/mobilec/trunk/src/include/xml_helper.h File Reference	400
8.46	/home/dko/projects/mobilec/trunk/src/include/xml_parser.h File Reference	402
8.47	/home/dko/projects/mobilec/trunk/src/libmc.c File Reference	406
8.48	/home/dko/projects/mobilec/trunk/src/mc_list/list.c File Reference	461
8.49	/home/dko/projects/mobilec/trunk/src/mc_list/list.h File Reference	463
8.50	/home/dko/projects/mobilec/trunk/src/mc_platform.c File Reference	466
8.51	/home/dko/projects/mobilec/trunk/src/mc_rwlock.c File Reference	468
8.52	/home/dko/projects/mobilec/trunk/src/mc_sync/sync_list.c File Reference	470
8.53	/home/dko/projects/mobilec/trunk/src/mc_sync/sync_list.h File Reference	472
8.54	/home/dko/projects/mobilec/trunk/src/message.c File Reference	475
8.55	/home/dko/projects/mobilec/trunk/src/message_queue.c File Reference	478
8.56	/home/dko/projects/mobilec/trunk/src/mtp_http.c File Reference	479
8.57	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/config.h File Reference	483
8.58	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/vcnet/config.h File Reference	484
8.59	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/vcnet2005/config.h File Reference	485
8.60	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-attr.c File Reference	487
8.61	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-entity.c File Reference	488
8.62	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-file.c File Reference	490
8.63	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-index.c File Reference	497
8.64	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-node.c File Reference	499
8.65	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-private.c File Reference	502
8.66	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-search.c File Reference	504
8.67	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-set.c File Reference	505
8.68	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-string.c File Reference	507
8.69	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml.h File Reference	508
8.70	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxmldoc.c File Reference	520
8.71	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/class.cxx File Reference	524

8.72	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/enum.cxx File Reference	525
8.73	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/function.cxx File Reference	526
8.74	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/struct.cxx File Reference	527
8.75	/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/testmxml.c File Reference	528
8.76	/home/dko/projects/mobilec/trunk/src/security/asm.c File Reference	530
8.77	/home/dko/projects/mobilec/trunk/src/security/asm.h File Reference	531
8.78	/home/dko/projects/mobilec/trunk/src/security/asm_message_composer.c File Reference .	532
8.79	/home/dko/projects/mobilec/trunk/src/security/asm_message_composer.h File Reference .	533
8.80	/home/dko/projects/mobilec/trunk/src/security/asm_message_parser.c File Reference . . .	534
8.81	/home/dko/projects/mobilec/trunk/src/security/asm_message_parser.h File Reference . . .	535
8.82	/home/dko/projects/mobilec/trunk/src/security/asm_node.c File Reference	536
8.83	/home/dko/projects/mobilec/trunk/src/security/asm_node.h File Reference	537
8.84	/home/dko/projects/mobilec/trunk/src/security/mc_dh.c File Reference	538
8.85	/home/dko/projects/mobilec/trunk/src/security/mc_dh.h File Reference	539
8.86	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/aes.h File Reference	540
8.87	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/arc4.h File Reference	543
8.88	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/base64.h File Reference	545
8.89	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/bignum.h File Reference	547
8.90	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/bn_asm.h File Reference	563
8.91	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/certs.h File Reference	565
8.92	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/des.h File Reference	567
8.93	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/dhm.h File Reference	572
8.94	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/havege.h File Reference	577
8.95	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/md2.h File Reference	579
8.96	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/md4.h File Reference	583
8.97	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/md5.h File Reference	587
8.98	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/net.h File Reference	592
8.99	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/rsa.h File Reference	597
8.100	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/sha1.h File Reference	609
8.101	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/sha2.h File Reference	614
8.102	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/sha4.h File Reference	619
8.103	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/ssl.h File Reference	624
8.104	/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/ssl_conf.h File Reference	642

8.105/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/timing.h File Reference	643
8.106/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/x509.h File Reference	645
8.107/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/aes.c File Reference	658
8.108/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/arc4.c File Reference	665
8.109/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/base64.c File Reference	667
8.110/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/bignum.c File Reference	670
8.111/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/certs.c File Reference	685
8.112/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/des.c File Reference	689
8.113/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/dhm.c File Reference	701
8.114/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/dsa.c File Reference	705
8.115/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/havege.c File Reference	710
8.116/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md2.c File Reference	713
8.117/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md4.c File Reference	718
8.118/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md5.c File Reference	724
8.119/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/net.c File Reference	730
8.120/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/rsa.c File Reference	734
8.121/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha1.c File Reference	741
8.122/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha2.c File Reference	748
8.123/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha4.c File Reference	755
8.124/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_cli.c File Reference	762
8.125/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_srv.c File Reference	765
8.126/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_tls.c File Reference	769
8.127/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/timing.c File Reference	779
8.128/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/x509read.c File Reference	782
8.129/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/aes/aescript2.c File Reference	791
8.130/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/hello.c File Reference	793
8.131/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/md5sum.c File Reference	794
8.132/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/sha1sum.c File Reference	796
8.133/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/sha2sum.c File Reference	798
8.134/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/dh_client.c File Reference	800
8.135/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/dh_genprime.c File Reference	802

8.136/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/dh_server.c	File	
Reference		803
8.137/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/mpi_demo.c	File	
Reference		805
8.138/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_genkey.c	File	
Reference		806
8.139/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_sign.c	File	
Reference		807
8.140/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_verify.c	File	
Reference		808
8.141/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/ssl/ssl_client1.c	File	
Reference		809
8.142/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/ssl/ssl_client2.c	File	
Reference		811
8.143/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/ssl/ssl_server.c	File	
Reference		813
8.144/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/test/benchmark.c	File	
Reference		816
8.145/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/test/selftest.c	File	
Reference		818
8.146/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAclMessage.cs	File	
Reference		819
8.147/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgency.cs	File	
Reference		820
8.148/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgent.cs	File	
Reference		821
8.149/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCEExports.cs	File	
Reference		822
8.150/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties/AssemblyInfo.cs	File	
Reference		823
8.151/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties/Settings.Designer.cs	File	
Reference		824
8.152/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Settings.cs	File	
Reference		825
8.153/home/dko/projects/mobilec/trunk/src/winconfig.h	File	
Reference		826
8.154/home/dko/projects/mobilec/trunk/src/xml_compose.c	File	
Reference		827
8.155/home/dko/projects/mobilec/trunk/src/xml_helper.c	File	
Reference		831
8.156/home/dko/projects/mobilec/trunk/src/xml_parser.c	File	
Reference		833
9 Mobile-C Example Documentation		837
9.1 LibMCCConsole/Program.cs		837
9.2 LibMCCppEx/LibMCCppEx.cpp		839
9.3 LibMCFipaTest/Program.cs		842
9.4 LibMCCGui/Form1.cs		844

9.5	LibMCMiscTest/Program.cs	847
9.6	LibMCVbEx/Form1.vb	848
10	Mobile-C Page Documentation	851
10.1	MCACLMessage	851
10.2	Installing LibMC.NET	852
10.3	Getting Started	854
10.4	Using LibMC.NET	855
10.5	Common Operations	856
10.6	MCAgency	857
10.7	MCAgent	859
10.8	Todo List	860
10.9	Bug List	861

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

Mobile-C Namespace Index

2.1 Mobile-C Namespace List

Here is a list of all namespaces with brief descriptions:

LibMC (Namespace for the .NET wrapper for Mobile-C)	13
LibMC::Properties (Namespace for the .NET wrapper properties class)	14
System	15
System::Collections::Generic	16
System::Reflection	17
System::Runtime::CompilerServices	18
System::Runtime::InteropServices	19
System::Text	20

Chapter 3

Mobile-C Data Structure Index

3.1 Mobile-C Data Structures

Here are the data structures with brief descriptions:

_hr_time	21
_x509_buf	22
_x509_cert	23
_x509_name	27
_x509_time	28
aes_context (AES context structure)	30
agency_s (The agency handle)	31
agent_datastate_s	34
agent_mailbox_s	37
agent_s	38
agent_task_s	43
arc4_context (ARC4 context structure)	46
barrier_node_s	47
barrier_queue_s	48
cmd_prompt_s	49
command_s	50
connection_s	51
des3_context (Triple-DES context structure)	53
des_context (DES context structure)	54
dhm_context	55
dynstring_s	57
fipa_acl_envelope_Received_s	58
fipa_acl_envelope_s	60
fipa_acl_message_s	61
fipa_acl_Param_s	64
fipa_agent_identifier_s	66
fipa_agent_identifier_set_s	67
fipa_DateTime_s	68
fipa_expression_s	70
fipa_expression_s::content_u	71
fipa_message_string_s	72
fipa_number_s	73
fipa_string_s	74

fipa_url_s	75
fipa_url_sequence_s	76
fipa_word_s	77
foo_c	78
foo_s	81
havege_state (HAVEGE state structure)	83
host_id_s	85
hr_time (Timer structure)	86
interpreter_variable_data_s	87
list_s	89
listNode_s	90
mc_platform_s	91
mc_rwlock_s	96
LibMC::MCAclMessage (Encapsulates ACL messages in the Mobile-C library)	98
LibMC::MCAgency (Wrapper class for MCAgency_t structure)	104
LibMC::MCAgency::ChOptions_t (ChOptions structures)	132
LibMC::MCAgency::MCAgency_t	133
LibMC::MCAgency::MCAgencyOptions_t	135
MCAgencyOptions_s (User modifiable agency options)	136
LibMC::MCAgent (Wrapper class for MCAgent_t structure)	138
md2_context (MD2 context structure)	148
md4_context (MD4 context structure)	150
md5_context (MD5 context structure)	152
message_s	154
mpi (MPI structure)	157
mtp_http_content_s	159
mtp_http_s	160
mxml_attr_s	163
mxml_custom_s	164
mxml_fdbuf_s	165
mxml_index_s	166
mxml_node_s	168
mxml_text_s	170
mxml_value_s	171
mxml_value_u	172
rsa_context	174
LibMC::Properties::Settings	177
sha1_context (SHA-1 context structure)	179
sha2_context (SHA-256 context structure)	181
sha4_context (SHA-512 context structure)	183
ssl_context	185
syncList_s	196
syncListNode_s	197

Chapter 4

Mobile-C File Index

4.1 Mobile-C File List

Here is a list of all files with brief descriptions:

/home/dko/projects/mobilec/trunk/src/acc.c	199
/home/dko/projects/mobilec/trunk/src/agent.c	202
/home/dko/projects/mobilec/trunk/src/agent_datastate.c	205
/home/dko/projects/mobilec/trunk/src/agent_mailbox.c	206
/home/dko/projects/mobilec/trunk/src/agent_return_data.c	208
/home/dko/projects/mobilec/trunk/src/agent_task.c	210
/home/dko/projects/mobilec/trunk/src/ams.c	211
/home/dko/projects/mobilec/trunk/src/barrier.c	213
/home/dko/projects/mobilec/trunk/src/cmd_prompt.c	215
/home/dko/projects/mobilec/trunk/src/connection.c	218
/home/dko/projects/mobilec/trunk/src/data_structures.c	219
/home/dko/projects/mobilec/trunk/src/df.c	221
/home/dko/projects/mobilec/trunk/src/dynstring.c	225
/home/dko/projects/mobilec/trunk/src/fipa_acl.c	226
/home/dko/projects/mobilec/trunk/src/fipa_envelope.c	238
/home/dko/projects/mobilec/trunk/src/libmc.c	406
/home/dko/projects/mobilec/trunk/src/mc_platform.c	466
/home/dko/projects/mobilec/trunk/src/mc_rwlock.c	468
/home/dko/projects/mobilec/trunk/src/message.c	475
/home/dko/projects/mobilec/trunk/src/message_queue.c	478
/home/dko/projects/mobilec/trunk/src/mtp_http.c	479
/home/dko/projects/mobilec/trunk/src/winconfig.h	826
/home/dko/projects/mobilec/trunk/src/xml_compose.c	827
/home/dko/projects/mobilec/trunk/src/xml_helper.c	831
/home/dko/projects/mobilec/trunk/src/xml_parser.c	833
/home/dko/projects/mobilec/trunk/src/include/acc.h	246
/home/dko/projects/mobilec/trunk/src/include/agent.h	249
/home/dko/projects/mobilec/trunk/src/include/agent_datastate.h	252
/home/dko/projects/mobilec/trunk/src/include/agent_lib.h	254
/home/dko/projects/mobilec/trunk/src/include/agent_mailbox.h	263
/home/dko/projects/mobilec/trunk/src/include/agent_task.h	265
/home/dko/projects/mobilec/trunk/src/include/ams.h	267
/home/dko/projects/mobilec/trunk/src/include/ap_queue_template.h	269

/home/dko/projects/mobilec/trunk/src/include/barrier.h	272
/home/dko/projects/mobilec/trunk/src/include/cmd_prompt.h	275
/home/dko/projects/mobilec/trunk/src/include/commands.h	277
/home/dko/projects/mobilec/trunk/src/include/commands.x.h	279
/home/dko/projects/mobilec/trunk/src/include/connection.h	281
/home/dko/projects/mobilec/trunk/src/include/data_structures.h	283
/home/dko/projects/mobilec/trunk/src/include/df.h	286
/home/dko/projects/mobilec/trunk/src/include/df_request.x.h	292
/home/dko/projects/mobilec/trunk/src/include/dynstring.h	293
/home/dko/projects/mobilec/trunk/src/include/fipa_acl.h	295
/home/dko/projects/mobilec/trunk/src/include/fipa_acl_envelope.h	310
/home/dko/projects/mobilec/trunk/src/include/host_id.h	318
/home/dko/projects/mobilec/trunk/src/include/interpreter_variable_data.h	319
/home/dko/projects/mobilec/trunk/src/include/libmc.h (MobileC api header file)	321
/home/dko/projects/mobilec/trunk/src/include/macros.h	371
/home/dko/projects/mobilec/trunk/src/include/mc_error.h	382
/home/dko/projects/mobilec/trunk/src/include/mc_platform.h	384
/home/dko/projects/mobilec/trunk/src/include/mc_rwlock.h	386
/home/dko/projects/mobilec/trunk/src/include/message.h	388
/home/dko/projects/mobilec/trunk/src/include/mtp_http.h	391
/home/dko/projects/mobilec/trunk/src/include/xml_compose.h	397
/home/dko/projects/mobilec/trunk/src/include/xml_helper.h	400
/home/dko/projects/mobilec/trunk/src/include/xml_parser.h	402
/home/dko/projects/mobilec/trunk/src/mc_list/list.c	461
/home/dko/projects/mobilec/trunk/src/mc_list/list.h	463
/home/dko/projects/mobilec/trunk/src/mc_sync/sync_list.c	470
/home/dko/projects/mobilec/trunk/src/mc_sync/sync_list.h	472
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/config.h	483
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-attr.c	487
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-entity.c	488
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-file.c	490
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-index.c	497
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-node.c	499
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-private.c	502
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-search.c	504
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-set.c	505
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-string.c	507
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml.h	508
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxmldoc.c	520
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/testmxml.c	528
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/class.cxx	524
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/enum.cxx	525
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/function.cxx	526
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/struct.cxx	527
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/vcnet/config.h	484
/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/vcnet2005/config.h	485
/home/dko/projects/mobilec/trunk/src/security/asm.c	530
/home/dko/projects/mobilec/trunk/src/security/asm.h	531
/home/dko/projects/mobilec/trunk/src/security/asm_message_composer.c	532
/home/dko/projects/mobilec/trunk/src/security/asm_message_composer.h	533
/home/dko/projects/mobilec/trunk/src/security/asm_message_parser.c	534
/home/dko/projects/mobilec/trunk/src/security/asm_message_parser.h	535
/home/dko/projects/mobilec/trunk/src/security/asm_node.c	536
/home/dko/projects/mobilec/trunk/src/security/asm_node.h	537

/home/dko/projects/mobilec/trunk/src/security/mc_dh.c	538
/home/dko/projects/mobilec/trunk/src/security/mc_dh.h	539
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/aes.h	540
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/arc4.h	543
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/base64.h	545
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/bignum.h	547
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/bn_asm.h	563
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/certs.h	565
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/des.h	567
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/dhm.h	572
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/havege.h	577
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/md2.h	579
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/md4.h	583
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/md5.h	587
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/net.h	592
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/rsa.h	597
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/sha1.h	609
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/sha2.h	614
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/sha4.h	619
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/ssl.h	624
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/ssl_conf.h	642
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/timing.h	643
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/x509.h	645
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/aes.c	658
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/arc4.c	665
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/base64.c	667
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/bignum.c	670
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/certs.c	685
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/des.c	689
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/dhm.c	701
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/dsa.c	705
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/havege.c	710
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md2.c	713
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md4.c	718
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md5.c	724
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/net.c	730
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/rsa.c	734
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha1.c	741
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha2.c	748
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha4.c	755
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_cli.c	762
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_srv.c	765
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_tls.c	769
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/timing.c	779
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/x509read.c	782
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/aes/aescrypt2.c	791
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/hello.c	793
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/md5sum.c	794
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/sha1sum.c	796
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/sha2sum.c	798
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/dh_client.c	800
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/dh_genprime.c	802
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/dh_server.c	803
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/mpi_demo.c	805

/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_genkey.c	806
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_sign.c	807
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_verify.c	808
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/ssl/ssl_client1.c	809
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/ssl/ssl_client2.c	811
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/ssl/ssl_server.c	813
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/test/benchmark.c	816
/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/test/selftest.c	818
/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAclMessage.cs	819
/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgency.cs	820
/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgent.cs	821
/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCExports.cs	822
/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Settings.cs	825
/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties/AssemblyInfo.cs . .	823
/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties/Settings.Designer.cs	824

Chapter 5

Mobile-C Page Index

5.1 Mobile-C Related Pages

Here is a list of all related documentation pages:

Installing LibMC.NET	852
Getting Started	854
Using LibMC.NET	855
Common Operations	856
Todo List	860
Bug List	861

Chapter 6

Mobile-C Namespace Documentation

6.1 LibMC Namespace Reference

Namespace for the .NET wrapper for Mobile-C.

Data Structures

- class [MCAclMessage](#)
Encapsulates ACL messages in the Mobile-C library.
- class [MCAgency](#)
Wrapper class for [MCAgency_t](#) structure.
- class [MCAgent](#)
Wrapper class for [MCAgent_t](#) structure.

Namespaces

- namespace [Properties](#)
Namespace for the .NET wrapper properties class.

6.1.1 Detailed Description

Namespace for the .NET wrapper for Mobile-C.

[LibMC](#) encapsulates the Mobile-C DLL for windows in an .NET class library. .NET programs can access the library to create agencies, connect to agencies, interact with agents, etc.

6.2 LibMC::Properties Namespace Reference

Namespace for the .NET wrapper properties class.

Data Structures

- class [Settings](#)

6.2.1 Detailed Description

Namespace for the .NET wrapper properties class.

Any user or global properties that should preserved from session to session can be added here through the designer. There are currently no properties in use.

6.3 System Namespace Reference

6.4 System::Collections::Generic Namespace Reference

6.5 System::Reflection Namespace Reference

6.6 System::Runtime::CompilerServices Namespace Reference

6.7 System::Runtime::InteropServices Namespace Reference

6.8 System::Text Namespace Reference

Chapter 7

Mobile-C Data Structure Documentation

7.1 `_hr_time` Struct Reference

Data Fields

- struct timeval [start](#)

7.1.1 Detailed Description

Definition at line 44 of file `timing.c`.

7.1.2 Field Documentation

7.1.2.1 `struct timeval _hr_time::start` `[read]`

Definition at line 46 of file `timing.c`.

Referenced by `set_timer()`.

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

- `/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/timing.c`

7.2 `_x509_buf` Struct Reference

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

Data Fields

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

7.2.1 Detailed Description

Definition at line 85 of file `x509.h`.

7.2.2 Field Documentation

7.2.2.1 `int _x509_buf::tag`

Definition at line 87 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()`.

7.2.2.2 `int _x509_buf::len`

Definition at line 88 of file `x509.h`.

Referenced by `ssl_write_certificate()`, `ssl_write_certificate_request()`, `x509_add_certs()`, `x509_cert_info()`, `x509_dn_gets()`, `x509_get_alg()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_serial()`, `x509_get_sig()`, `x509_get_uid()`, and `x509_verify_cert()`.

7.2.2.3 `unsigned char* _x509_buf::p`

Definition at line 89 of file `x509.h`.

Referenced by `ssl_write_certificate()`, `ssl_write_certificate_request()`, `x509_add_certs()`, `x509_cert_info()`, `x509_dn_gets()`, `x509_free_cert()`, `x509_get_alg()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_serial()`, `x509_get_sig()`, `x509_get_uid()`, and `x509_verify_cert()`.

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

- `/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/x509.h`

7.3 `_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

7.3.1 Detailed Description

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

7.3.2 Field Documentation

7.3.2.1 `x509_buf _x509_cert::raw`

Definition at line 110 of file `x509.h`.

Referenced by `ssl_write_certificate()`, `x509_add_certs()`, and `x509_free_cert()`.

7.3.2.2 `x509_buf _x509_cert::tbs`

Definition at line 111 of file `x509.h`.

Referenced by `x509_add_certs()`, and `x509_verify_cert()`.

7.3.2.3 `int _x509_cert::version`

Definition at line 113 of file `x509.h`.

Referenced by `x509_add_certs()`, `x509_cert_info()`, and `x509_verify_cert()`.

7.3.2.4 x509_buf_x509_cert::serial

Definition at line 114 of file x509.h.

Referenced by x509_add_certs(), and x509_cert_info().

7.3.2.5 x509_buf_x509_cert::sig_oid1

Definition at line 115 of file x509.h.

Referenced by x509_add_certs(), x509_cert_info(), and x509_verify_cert().

7.3.2.6 x509_buf_x509_cert::issuer_raw

Definition at line 117 of file x509.h.

Referenced by x509_add_certs(), and x509_verify_cert().

7.3.2.7 x509_buf_x509_cert::subject_raw

Definition at line 118 of file x509.h.

Referenced by ssl_write_certificate_request(), x509_add_certs(), and x509_verify_cert().

7.3.2.8 x509_name_x509_cert::issuer

Definition at line 120 of file x509.h.

Referenced by x509_add_certs(), x509_cert_info(), and x509_free_cert().

7.3.2.9 x509_name_x509_cert::subject

Definition at line 121 of file x509.h.

Referenced by x509_add_certs(), x509_cert_info(), x509_free_cert(), and x509_verify_cert().

7.3.2.10 x509_time_x509_cert::valid_from

Definition at line 123 of file x509.h.

Referenced by x509_add_certs(), and x509_cert_info().

7.3.2.11 x509_time_x509_cert::valid_to

Definition at line 124 of file x509.h.

Referenced by x509_add_certs(), x509_cert_info(), and x509_is_cert_expired().

7.3.2.12 x509_buf_x509_cert::pk_oid

Definition at line 126 of file x509.h.

Referenced by x509_add_certs().

7.3.2.13 `rsa_context _x509_cert::rsa`

Definition at line 127 of file `x509.h`.

Referenced by `ssl_parse_certificate_verify()`, `ssl_parse_server_key_exchange()`, `ssl_write_client_key_exchange()`, `x509_add_certs()`, `x509_cert_info()`, `x509_free_cert()`, and `x509_verify_cert()`.

7.3.2.14 `x509_buf _x509_cert::issuer_id`

Definition at line 129 of file `x509.h`.

Referenced by `x509_add_certs()`.

7.3.2.15 `x509_buf _x509_cert::subject_id`

Definition at line 130 of file `x509.h`.

Referenced by `x509_add_certs()`.

7.3.2.16 `x509_buf _x509_cert::v3_ext`

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

Referenced by `x509_add_certs()`.

7.3.2.17 `int _x509_cert::ca_istrue`

Definition at line 133 of file `x509.h`.

Referenced by `x509_add_certs()`, and `x509_verify_cert()`.

7.3.2.18 `int _x509_cert::max_pathlen`

Definition at line 134 of file `x509.h`.

Referenced by `x509_add_certs()`, and `x509_verify_cert()`.

7.3.2.19 `x509_buf _x509_cert::sig_oid2`

Definition at line 136 of file `x509.h`.

Referenced by `x509_add_certs()`.

7.3.2.20 `x509_buf _x509_cert::sig`

Definition at line 137 of file `x509.h`.

Referenced by `x509_add_certs()`, and `x509_verify_cert()`.

7.3.2.21 `struct _x509_cert* _x509_cert::next` `[read]`

Definition at line 139 of file `x509.h`.

Referenced by `main()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `x509_add_certs()`, `x509_free_cert()`, and `x509_verify_cert()`.

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

- `/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/x509.h`

7.4 `_x509_name` Struct Reference

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

Data Fields

- [x509_buf oid](#)
- [x509_buf val](#)
- [struct `_x509_name` * next](#)

7.4.1 Detailed Description

Definition at line 93 of file `x509.h`.

7.4.2 Field Documentation

7.4.2.1 `x509_buf _x509_name::oid`

Definition at line 95 of file `x509.h`.

Referenced by `x509_dn_gets()`, `x509_get_name()`, and `x509_verify_cert()`.

7.4.2.2 `x509_buf _x509_name::val`

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

Referenced by `x509_dn_gets()`, `x509_get_name()`, and `x509_verify_cert()`.

7.4.2.3 `struct _x509_name* _x509_name::next` [read]

Definition at line 97 of file `x509.h`.

Referenced by `x509_dn_gets()`, `x509_free_cert()`, `x509_get_name()`, and `x509_verify_cert()`.

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

- `/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/x509.h`

7.5 `_x509_time` Struct Reference

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

Data Fields

- `int year`
- `int mon`
- `int day`
- `int hour`
- `int min`
- `int sec`

7.5.1 Detailed Description

Definition at line 101 of file `x509.h`.

7.5.2 Field Documentation

7.5.2.1 `int _x509_time::year`

Definition at line 103 of file `x509.h`.

Referenced by `x509_cert_info()`, `x509_get_dates()`, and `x509_is_cert_expired()`.

7.5.2.2 `int _x509_time::mon`

Definition at line 103 of file `x509.h`.

Referenced by `x509_cert_info()`, `x509_get_dates()`, and `x509_is_cert_expired()`.

7.5.2.3 `int _x509_time::day`

Definition at line 103 of file `x509.h`.

Referenced by `x509_cert_info()`, `x509_get_dates()`, and `x509_is_cert_expired()`.

7.5.2.4 `int _x509_time::hour`

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

Referenced by `x509_cert_info()`, and `x509_get_dates()`.

7.5.2.5 `int _x509_time::min`

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

Referenced by `x509_cert_info()`, and `x509_get_dates()`.

7.5.2.6 `int _x509_time::sec`

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

Referenced by `x509_cert_info()`, and `x509_get_dates()`.

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

- `/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/x509.h`

7.6 aes_context Struct Reference

AES context structure.

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

Data Fields

- unsigned long [erk](#) [64]
- unsigned long [drk](#) [64]
- [int](#) [nr](#)

7.6.1 Detailed Description

AES context structure.

Definition at line 14 of file [aes.h](#).

7.6.2 Field Documentation

7.6.2.1 unsigned long aes_context::erk[64]

encryption round keys

Definition at line 16 of file [aes.h](#).

Referenced by [aes_set_key\(\)](#).

7.6.2.2 unsigned long aes_context::drk[64]

decryption round keys

Definition at line 17 of file [aes.h](#).

Referenced by [aes_set_key\(\)](#).

7.6.2.3 int aes_context::nr

number of rounds

Definition at line 18 of file [aes.h](#).

Referenced by [aes_set_key\(\)](#).

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

- [/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/aes.h](#)

7.7 agency_s Struct Reference

The agency handle.

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

Data Fields

- [int client](#)
- [int server](#)
- [char * hostName](#)
- [char * filename](#)
- [int portno](#)
- [int portnoc](#)
- [struct mc_platform_s * mc_platform](#)
- [int default_agentstatus](#)
- [int threads](#)
- [int enable_security](#)
- [int stack_size](#) [MC_THREAD_ALL]
- [error_code_t last_error](#)

7.7.1 Detailed Description

The agency handle.

Definition at line 212 of file libmc.h.

7.7.2 Field Documentation

7.7.2.1 int agency_s::client

Definition at line 213 of file libmc.h.

Referenced by MC_Initialize().

7.7.2.2 int agency_s::server

Definition at line 214 of file libmc.h.

Referenced by MC_Initialize().

7.7.2.3 char* agency_s::hostName

Local Hostname

Definition at line 215 of file libmc.h.

Referenced by MC_End(), and MC_Initialize().

7.7.2.4 char* agency_s::filename

Definition at line 216 of file libmc.h.

7.7.2.5 int agency_s::portno

Local port number

Definition at line 217 of file libmc.h.

Referenced by MC_Initialize(), and mc_platform_Initialize().

7.7.2.6 int agency_s::portnoc

Definition at line 218 of file libmc.h.

7.7.2.7 struct mc_platform_s* agency_s::mc_platform [read]

Local MobileC Platform

Definition at line 219 of file libmc.h.

Referenced by MC_AclSend_chdl(), MC_AddAgent(), MC_AddAgent_chdl(), MC_Barrier(), MC_Barrier_chdl(), MC_BarrierDelete(), MC_BarrierDelete_chdl(), MC_BarrierInit(), MC_BarrierInit_chdl(), MC_ChInitializeOptions(), 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_MutexUnlock(), MC_RegisterService(), MC_ResetSignal(), MC_ResumeAgency(), MC_RetrieveAgent(), MC_SearchForService(), MC_SemaphorePost(), MC_SemaphoreWait(), MC_SendAgentMigrationMessage(), MC_SendAgentMigrationMessageFile(), MC_SendSteerCommand(), MC_SetDefaultAgentStatus(), MC_Steer(), MC_SyncDelete(), MC_SyncInit(), MC_WaitAgent(), MC_WaitRetrieveAgent(), and MC_WaitSignal().

7.7.2.8 int agency_s::default_agentstatus

Agency default agent status

Definition at line 220 of file libmc.h.

Referenced by MC_Initialize(), and mc_platform_Initialize().

7.7.2.9 int agency_s::threads

flag which determines which threads to start

Definition at line 221 of file libmc.h.

Referenced by MC_End(), MC_Initialize(), and mc_platform_Initialize().

7.7.2.10 int agency_s::enable_security

Security flag

Definition at line 222 of file libmc.h.

Referenced by MC_Initialize(), and mc_platform_Initialize().

7.7.2.11 int agency_s::stack_size[MC_THREAD_ALL]

Definition at line 223 of file libmc.h.

Referenced by MC_Initialize(), and mc_platform_Initialize().

7.7.2.12 error_code_t agency_s::last_error

Definition at line 224 of file libmc.h.

Referenced by mc_platform_Initialize().

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

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

7.8 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](#)

7.8.1 Detailed Description

Definition at line 38 of file agent_datastate.h.

7.8.2 Field Documentation

7.8.2.1 `char** agent_datastate_s::agent_code_ids`

Definition at line 40 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()`, and `MC_ComposeAgent()`.

7.8.2.2 `char** agent_datastate_s::agent_codes`

Definition at line 41 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()`, and `MC_ComposeAgent()`.

7.8.2.3 `char* agent_datastate_s::agent_code`

Definition at line 42 of file agent_datastate.h.

Referenced by `agent_datastate_Copy()`, `agent_datastate_New()`, `agent_xml_parse__agent_code()`, `agent_xml_parse__tasks()`, `MC_ComposeAgent()`, `MC_PrintAgentCode()`, and `MC_RetrieveAgentCode()`.

7.8.2.4 `agent_task_p* agent_datastate_s::tasks`

Definition at line 45 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_AgentVariableRetrieve(), MC_AgentVariableSave(), MC_ComposeAgent(), MC_GetAgentReturnData(), and message_InitializeFromAgent().

7.8.2.5 mxml_node_t* agent_datastate_s::xml_agent_root

Definition at line 48 of file agent_datastate.h.

Referenced by agent_datastate_New(), agent_Initialize(), agent_xml_parse(), and MC_GetAgentXMLString().

7.8.2.6 mxml_node_t* agent_datastate_s::xml_root

Definition at line 49 of file agent_datastate.h.

Referenced by agent_datastate_Destroy(), agent_datastate_New(), agent_Initialize(), and agent_return_xml_parse().

7.8.2.7 int agent_datastate_s::task_progress

Definition at line 52 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_AgentVariableSave(), MC_PrintAgentCode(), MC_RetrieveAgentCode(), and message_InitializeFromAgent().

7.8.2.8 int agent_datastate_s::return_data

Definition at line 53 of file agent_datastate.h.

Referenced by agent_datastate_Copy(), and agent_datastate_New().

7.8.2.9 int agent_datastate_s::number_of_tasks

Definition at line 56 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_ComposeAgent(), MC_GetAgentNumTasks(), MC_GetAgentReturnData(), MC_PrintAgentCode(), and message_InitializeFromAgent().

7.8.2.10 int agent_datastate_s::persistent

Definition at line 59 of file agent_datastate.h.

Referenced by acc_MessageHandlerThread(), agent_datastate_Copy(), agent_datastate_New(), agent_xml_compose__task(), and MC_ComposeAgent().

7.8.2.11 int agent_datastate_s::init_agent_status

Definition at line 60 of file agent_datastate.h.

Referenced by agent_datastate_Copy(), and agent_datastate_New().

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

- [/home/dko/projects/mobilec/trunk/src/include/agent_datastate.h](#)

7.9 agent_mailbox_s Struct Reference

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

Data Fields

- struct mail_queue_s * [mail_queue](#)

7.9.1 Detailed Description

Definition at line 5 of file agent_mailbox.h.

7.9.2 Field Documentation

7.9.2.1 struct mail_queue_s* agent_mailbox_s::mail_queue [read]

Definition at line 7 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/trunk/src/include/[agent_mailbox.h](#)

7.10 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](#)
- int [home_port](#)
- 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](#)

7.10.1 Detailed Description

Definition at line 45 of file [agent.h](#).

7.10.2 Field Documentation

7.10.2.1 u_long agent_s::id

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

Referenced by [agent_Copy\(\)](#), [agent_Initialize\(\)](#), [agent_RunChScriptThread\(\)](#), [ams_Print\(\)](#), [AP_QUEUE_STD_DEFN_TEMPLATE\(\)](#), [MC_GetAgentID\(\)](#), and [MC_RegisterService\(\)](#).

7.10.2.2 char* agent_s::name

Definition at line 49 of file agent.h.

Referenced by acc_MessageHandlerThread(), agent_AddPersistentVariable(), agent_Copy(), agent_Destroy(), agent_RunChScriptThread(), agent_xml_compose__name(), agent_xml_parse__name(), AP_QUEUE_STD_DEFN_TEMPLATE(), MC_ComposeAgent(), MC_GetAgentName(), and MC_RegisterService().

7.10.2.3 u_long agent_s::connect_id

Definition at line 50 of file agent.h.

Referenced by ams_Print().

7.10.2.4 time_t agent_s::arrival_time

Definition at line 52 of file agent.h.

Referenced by agent_Copy(), agent_Initialize(), and MC_GetAgentArrivalTime().

7.10.2.5 char* agent_s::owner

Definition at line 56 of file agent.h.

Referenced by agent_Copy(), agent_Destroy(), agent_xml_compose__owner(), agent_xml_parse__owner(), and MC_ComposeAgent().

7.10.2.6 char* agent_s::home

Definition at line 57 of file agent.h.

Referenced by agent_Copy(), agent_Destroy(), agent_xml_compose__home(), agent_xml_parse__home(), MC_ComposeAgent(), and message_InitializeFromAgent().

7.10.2.7 int agent_s::home_port

Definition at line 58 of file agent.h.

Referenced by agent_Copy().

7.10.2.8 int agent_s::orphan

Definition at line 60 of file agent.h.

Referenced by agent_Copy(), agent_Initialize(), ams_ManageAgentList(), MC_ComposeAgent(), and MC_SetAgentStatus().

7.10.2.9 agent_datastate_p agent_s::datastate

Definition at line 64 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_AgentVariableRetrieve()`, `MC_AgentVariableSave()`, `MC_ComposeAgent()`, `MC_GetAgentNumTasks()`, `MC_GetAgentReturnData()`, `MC_GetAgentXMLString()`, `MC_PrintAgentCode()`, `MC_RetrieveAgentCode()`, and `message_InitializeFromAgent()`.

7.10.2.10 `enum MC_AgentType_e agent_s::agent_type`

Definition at line 65 of file `agent.h`.

Referenced by `agent_Copy()`, `agent_Initialize()`, `agent_xml_compose__message()`, `MC_ComposeAgent()`, and `MC_GetAgentType()`.

7.10.2.11 `enum MC_AgentStatus_e agent_s::agent_status`

Definition at line 66 of file `agent.h`.

Referenced by `acc_MessageHandlerThread()`, `agent_Copy()`, `agent_Destroy()`, `agent_Initialize()`, `agent_RunChScript()`, `ams_ManageAgentList()`, `ams_Print()`, `MC_ComposeAgent()`, `MC_GetAgentStatus()`, `MC_RetrieveAgent()`, and `MC_SetAgentStatus()`.

7.10.2.12 `int agent_s::return_data`

Definition at line 68 of file `agent.h`.

Referenced by `agent_Copy()`.

7.10.2.13 `ChInterp_t agent_s::agent_interp`

Definition at line 71 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_GetAgentExecEngine()`, and `MC_TerminateAgent()`.

7.10.2.14 `MUTEX_T* agent_s::run_lock`

Definition at line 74 of file `agent.h`.

Referenced by `agent_Copy()`, `agent_Destroy()`, `agent_Initialize()`, `agent_New()`, `interpreter_variable_data_Initialize()`, and `MC_CallAgentFunc()`.

7.10.2.15 `int agent_s::agent_thread_id`

Definition at line 77 of file `agent.h`.

Referenced by `agent_Initialize()`.

7.10.2.16 `THREAD_T* agent_s::agent_thread`

Definition at line 80 of file agent.h.

Referenced by agent_Copy(), agent_Destroy(), agent_Initialize(), and agent_RunChScript().

7.10.2.17 `agent_mailbox_p agent_s::mailbox`

Definition at line 83 of file agent.h.

Referenced by acc_Thread(), agent_Destroy(), agent_Initialize(), MC_AclPost(), MC_AclRetrieve(), and MC_AclWaitRetrieve().

7.10.2.18 `int agent_s::agent_pipe_active`

Definition at line 86 of file agent.h.

Referenced by agent_Initialize().

7.10.2.19 `int agent_s::agent_ready_to_send`

Definition at line 87 of file agent.h.

Referenced by agent_Initialize().

7.10.2.20 `int agent_s::agent_pipe_ready_to_read`

Definition at line 88 of file agent.h.

Referenced by agent_Initialize().

7.10.2.21 `int agent_s::agent_script_ready`

Definition at line 89 of file agent.h.

Referenced by agent_Initialize().

7.10.2.22 `int agent_s::agent_persistent`

Definition at line 90 of file agent.h.

Referenced by agent_Copy().

7.10.2.23 `struct mc_platform_s* agent_s::mc_platform` [read]

Definition at line 92 of file agent.h.

Referenced by agent_Initialize(), agent_RunChScript(), agent_RunChScriptThread(), MC_AddAgent(), and MC_SetAgentStatus().

7.10.2.24 MUTEX_T* agent_s::lock

Definition at line 94 of file agent.h.

Referenced by acc_MessageHandlerThread(), agent_Copy(), agent_Destroy(), agent_Initialize(), agent_New(), ams_ManageAgentList(), MC_GetAgentName(), MC_GetAgentStatus(), MC_PrintAgentCode(), MC_RetrieveAgentCode(), and MC_SetAgentStatus().

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

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

7.11 agent_task_s Struct Reference

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

Data Fields

- [int task_completed](#)
- [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](#)

7.11.1 Detailed Description

Definition at line 37 of file agent_task.h.

7.11.2 Field Documentation

7.11.2.1 int agent_task_s::task_completed

Definition at line 40 of file agent_task.h.

Referenced by agent_task_Copy(), agent_xml_compose__task(), and agent_xml_parse__task().

7.11.2.2 int agent_task_s::number_of_elements

Definition at line 41 of file agent_task.h.

Referenced by agent_task_Copy().

7.11.2.3 int agent_task_s::size_of_element_array

Definition at line 42 of file agent_task.h.

Referenced by agent_task_Copy().

7.11.2.4 int agent_task_s::persistent

Definition at line 43 of file agent_task.h.

Referenced by agent_task_Copy(), agent_xml_compose__task(), and agent_xml_parse__data().

7.11.2.5 `int agent_task_s::init_agent_status`

Definition at line 44 of file `agent_task.h`.

Referenced by `agent_task_Copy()`.

7.11.2.6 `char* agent_task_s::var_name`

Definition at line 46 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()`, and `MC_ComposeAgent()`.

7.11.2.7 `char* agent_task_s::server_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()`, `MC_ComposeAgent()`, and `message_InitializeFromAgent()`.

7.11.2.8 `char* agent_task_s::code_id`

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

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

7.11.2.9 `interpreter_variable_data_t* agent_task_s::agent_return_data`

Definition at line 51 of file `agent_task.h`.

Referenced by `agent_task_Copy()`, `agent_task_Destroy()`, `agent_xml_compose__task()`, `agent_xml_parse__data()`, and `MC_GetAgentReturnData()`.

7.11.2.10 `struct agent_variable_list_s* agent_task_s::agent_variable_list` [read]

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

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

7.11.2.11 `char** agent_task_s::saved_variables`

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

Referenced by `agent_task_Destroy()`, `agent_task_New()`, and `MC_AgentVariableSave()`.

7.11.2.12 `int agent_task_s::num_saved_variables`

Definition at line 54 of file `agent_task.h`.

Referenced by `agent_task_New()`, and `MC_AgentVariableSave()`.

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

- [/home/dko/projects/mobilec/trunk/src/include/agent_task.h](#)

7.12 arc4_context Struct Reference

ARC4 context structure.

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

Data Fields

- [int m](#) [256]
- [int x](#)
- [int y](#)

7.12.1 Detailed Description

ARC4 context structure.

Definition at line 14 of file arc4.h.

7.12.2 Field Documentation

7.12.2.1 int arc4_context::m[256]

permutation table

Definition at line 16 of file arc4.h.

Referenced by arc4_crypt(), and arc4_setup().

7.12.2.2 int arc4_context::x

permutation index

Definition at line 17 of file arc4.h.

Referenced by arc4_crypt(), and arc4_setup().

7.12.2.3 int arc4_context::y

permutation index

Definition at line 18 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/trunk/src/security/xyssl-0.7/include/xyssl/arc4.h](#)

7.13 barrier_node_s Struct Reference

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

Data Fields

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

7.13.1 Detailed Description

Definition at line 39 of file barrier.h.

7.13.2 Field Documentation

7.13.2.1 `MUTEX_T* barrier_node_s::lock`

Definition at line 40 of file barrier.h.

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

7.13.2.2 `COND_T* barrier_node_s::cond`

Definition at line 41 of file barrier.h.

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

7.13.2.3 `int barrier_node_s::id`

Definition at line 42 of file barrier.h.

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

7.13.2.4 `int barrier_node_s::num_registered`

Definition at line 43 of file barrier.h.

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

7.13.2.5 `int barrier_node_s::num_waiting`

Definition at line 44 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/trunk/src/include/barrier.h`

7.14 barrier_queue_s Struct Reference

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

Data Fields

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

7.14.1 Detailed Description

Definition at line 48 of file barrier.h.

7.14.2 Field Documentation

7.14.2.1 `RWLOCK_T* barrier_queue_s::lock`

Definition at line 49 of file barrier.h.

Referenced by `barrier_queue_Add()`, `barrier_queue_Delete()`, `barrier_queue_Destroy()`, `barrier_queue_Get()`, and `barrier_queue_New()`.

7.14.2.2 `list_p barrier_queue_s::list`

Definition at line 51 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()`.

7.14.2.3 `int barrier_queue_s::size`

Definition at line 52 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/trunk/src/include/barrier.h`

7.15 cmd_prompt_s Struct Reference

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

Data Fields

- `THREAD_T` [thread](#)

7.15.1 Detailed Description

Definition at line 38 of file `cmd_prompt.h`.

7.15.2 Field Documentation

7.15.2.1 `THREAD_T cmd_prompt_s::thread`

Definition at line 39 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/trunk/src/include/cmd_prompt.h`

7.16 `command_s` Struct Reference

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

Data Fields

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

7.16.1 Detailed Description

Definition at line 43 of file `cmd_prompt.h`.

7.16.2 Field Documentation

7.16.2.1 `int command_s::index`

Definition at line 44 of file `cmd_prompt.h`.

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

7.16.2.2 `int command_s::num_args`

Definition at line 45 of file `cmd_prompt.h`.

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

7.16.2.3 `char** command_s::args`

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

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

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

- `/home/dko/projects/mobilec/trunk/src/include/cmd_prompt.h`

7.17 connection_s Struct Reference

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

Data Fields

- [int connect_id](#)
- [char * remote_hostname](#)
- [struct sockaddr_in addr](#)
- [u_long clientfd](#)
- [u_long serverfd](#)

7.17.1 Detailed Description

Definition at line 42 of file connection.h.

7.17.2 Field Documentation

7.17.2.1 int connection_s::connect_id

Definition at line 45 of file connection.h.

Referenced by `AP_QUEUE_SEARCH_TEMPLATE()`, `connection_Copy()`, and `message_InitializeFromConnection()`.

7.17.2.2 char* connection_s::remote_hostname

Definition at line 48 of file connection.h.

Referenced by `AP_QUEUE_SEARCH_TEMPLATE()`, `connection_Copy()`, and `connection_Destroy()`.

7.17.2.3 struct sockaddr_in connection_s::addr [read]

Definition at line 50 of file connection.h.

Referenced by `connection_Copy()`, and `message_InitializeFromConnection()`.

7.17.2.4 u_long connection_s::clientfd

Definition at line 53 of file connection.h.

Referenced by `connection_Copy()`, `connection_Destroy()`, `message_InitializeFromConnection()`, and `mtp_http_InitializeFromConnection()`.

7.17.2.5 u_long connection_s::serverfd

Definition at line 54 of file connection.h.

Referenced by `connection_Copy()`.

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

- </home/dko/projects/mobilec/trunk/src/include/connection.h>

7.18 des3_context Struct Reference

Triple-DES context structure.

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

Data Fields

- unsigned long [esk](#) [96]
- unsigned long [dsk](#) [96]

7.18.1 Detailed Description

Triple-DES context structure.

Definition at line 24 of file des.h.

7.18.2 Field Documentation

7.18.2.1 unsigned long des3_context::esk[96]

Triple-DES encryption subkeys

Definition at line 26 of file des.h.

Referenced by [des3_cbc_encrypt\(\)](#), [des3_encrypt\(\)](#), [des3_set_2keys\(\)](#), and [des3_set_3keys\(\)](#).

7.18.2.2 unsigned long des3_context::dsk[96]

Triple-DES decryption subkeys

Definition at line 27 of file des.h.

Referenced by [des3_cbc_decrypt\(\)](#), [des3_decrypt\(\)](#), [des3_set_2keys\(\)](#), and [des3_set_3keys\(\)](#).

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

- [/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/des.h](#)

7.19 des_context Struct Reference

DES context structure.

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

Data Fields

- unsigned long [esk](#) [32]
- unsigned long [dsk](#) [32]

7.19.1 Detailed Description

DES context structure.

Definition at line 14 of file des.h.

7.19.2 Field Documentation

7.19.2.1 unsigned long des_context::esk[32]

DES encryption subkeys

Definition at line 16 of file des.h.

Referenced by [des_cbc_encrypt\(\)](#), [des_encrypt\(\)](#), and [des_set_key\(\)](#).

7.19.2.2 unsigned long des_context::dsk[32]

DES decryption subkeys

Definition at line 17 of file des.h.

Referenced by [des_cbc_decrypt\(\)](#), [des_decrypt\(\)](#), and [des_set_key\(\)](#).

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

- [/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/des.h](#)

7.20 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](#)

7.20.1 Detailed Description

Definition at line 20 of file dhm.h.

7.20.2 Field Documentation

7.20.2.1 int dhm_context::len

size(P) in chars

Definition at line 22 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().

7.20.2.2 mpi dhm_context::P

prime modulus

Definition at line 23 of file dhm.h.

Referenced by dhm_calc_secret(), dhm_free(), dhm_make_params(), dhm_make_public(), dhm_read_params(), and main().

7.20.2.3 mpi dhm_context::G

generator

Definition at line 24 of file dhm.h.

Referenced by dhm_free(), dhm_make_params(), dhm_make_public(), dhm_read_params(), and main().

7.20.2.4 mpi dhm_context::X

secret value

Definition at line 25 of file dhm.h.

Referenced by dhm_calc_secret(), dhm_free(), dhm_make_params(), and dhm_make_public().

7.20.2.5 mpi dhm_context::GX

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

Definition at line 26 of file dhm.h.

Referenced by dhm_free(), dhm_make_params(), and dhm_make_public().

7.20.2.6 mpi dhm_context::GY

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

Definition at line 27 of file dhm.h.

Referenced by dhm_calc_secret(), dhm_free(), dhm_read_params(), and dhm_read_public().

7.20.2.7 mpi dhm_context::K

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

Definition at line 28 of file dhm.h.

Referenced by dhm_calc_secret(), and dhm_free().

7.20.2.8 mpi dhm_context::RP

cached $R^2 \bmod P$

Definition at line 29 of file dhm.h.

Referenced by dhm_calc_secret(), dhm_free(), dhm_make_params(), and dhm_make_public().

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

- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/[dhm.h](#)

7.21 dynstring_s Struct Reference

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

Data Fields

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

7.21.1 Detailed Description

Definition at line 6 of file dynstring.h.

7.21.2 Field Documentation

7.21.2.1 int dynstring_s::len

Definition at line 7 of file dynstring.h.

Referenced by `dynstring_Append()`, `dynstring_New()`, and `mtp_http_CreateMessage()`.

7.21.2.2 int dynstring_s::size

Definition at line 8 of file dynstring.h.

Referenced by `dynstring_Append()`, and `dynstring_New()`.

7.21.2.3 char* dynstring_s::message

Definition at line 9 of file dynstring.h.

Referenced by `dynstring_Append()`, `dynstring_Destroy()`, `dynstring_New()`, `fipa_envelope_Compose__from()`, `MC_AclSend()`, and `mtp_http_CreateMessage()`.

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

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

7.22 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](#)

7.22.1 Detailed Description

Definition at line 4 of file `fipa_acl_envelope.h`.

7.22.2 Field Documentation

7.22.2.1 struct [fipa_url_s](#)* [fipa_acl_envelope_Received_s::received_by](#) [read]

Definition at line 6 of file `fipa_acl_envelope.h`.

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

7.22.2.2 struct [fipa_url_s](#)* [fipa_acl_envelope_Received_s::received_from](#) [read]

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

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

7.22.2.3 struct [fipa_DateTime_s](#)* [fipa_acl_envelope_Received_s::received_date](#) [read]

Definition at line 8 of file `fipa_acl_envelope.h`.

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

7.22.2.4 char* [fipa_acl_envelope_Received_s::received_id](#)

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()`.

7.22.2.5 struct [fipa_url_s](#)* [fipa_acl_envelope_Received_s::received_via](#) [read]

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()`.

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

- `/home/dko/projects/mobilec/trunk/src/include/fipa_acl_envelope.h`

7.23 fipa_acl_envelope_s Struct Reference

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

Data Fields

- [int num_params](#)
- [struct fipa_acl_Param_s ** params](#)

7.23.1 Detailed Description

Definition at line 37 of file [fipa_acl_envelope.h](#).

7.23.2 Field Documentation

7.23.2.1 int fipa_acl_envelope_s::num_params

Definition at line 39 of file [fipa_acl_envelope.h](#).

Referenced by [acc_Thread\(\)](#), [fipa_acl_envelope_Copy\(\)](#), [fipa_acl_envelope_Destroy\(\)](#), and [fipa_envelope_HandleParams\(\)](#).

7.23.2.2 struct fipa_acl_Param_s** fipa_acl_envelope_s::params [read]

Definition at line 40 of file [fipa_acl_envelope.h](#).

Referenced by [acc_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/trunk/src/include/fipa_acl_envelope.h](#)

7.24 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
- struct [fipa_word_s](#) * 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

7.24.1 Detailed Description

Definition at line 78 of file `fipa_acl.h`.

7.24.2 Field Documentation

7.24.2.1 enum fipa_performative_e fipa_acl_message_s::performative

Definition at line 80 of file `fipa_acl.h`.

Referenced by `fipa_acl_Compose()`, `fipa_acl_message_Copy()`, `fipa_acl_Parse()`, `MC_AclSetPerformative()`, and `MC_AclSetPerformative_chdl()`.

7.24.2.2 struct fipa_agent_identifier_s* fipa_acl_message_s::sender [read]

Definition at line 82 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()`, and `MC_AclSetSender()`.

7.24.2.3 struct fipa_agent_identifier_set_s* fipa_acl_message_s::receiver [read]

Definition at line 84 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__to()`, `fipa_message_parameter_Parse()`, `fipa_Reply()`, `MC_AclAddReceiver()`, and `MC_AclSend()`.

7.24.2.4 `int fipa_acl_message_s::receiver_num`

Definition at line 85 of file `fipa_acl.h`.

Referenced by `MC_AclAddReceiver()`.

7.24.2.5 `struct fipa_agent_identifier_set_s* fipa_acl_message_s::reply_to` [read]

Definition at line 87 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()`.

7.24.2.6 `struct fipa_string_s* fipa_acl_message_s::content` [read]

Definition at line 90 of file `fipa_acl.h`.

Referenced by `fipa_acl_Compose()`, `fipa_acl_message_Copy()`, `fipa_acl_message_Destroy()`, `fipa_message_parameter_Parse()`, `MC_AclSetContent()`, and `MC_AclSetContent_chdl()`.

7.24.2.7 `struct fipa_expression_s* fipa_acl_message_s::language` [read]

Definition at line 92 of file `fipa_acl.h`.

Referenced by `fipa_acl_Compose()`, `fipa_acl_message_Copy()`, `fipa_acl_message_Destroy()`, and `fipa_message_parameter_Parse()`.

7.24.2.8 `struct fipa_expression_s* fipa_acl_message_s::encoding` [read]

Definition at line 94 of file `fipa_acl.h`.

Referenced by `fipa_acl_Compose()`, `fipa_acl_message_Copy()`, `fipa_acl_message_Destroy()`, and `fipa_message_parameter_Parse()`.

7.24.2.9 `struct fipa_expression_s* fipa_acl_message_s::ontology` [read]

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

Referenced by `fipa_acl_Compose()`, `fipa_acl_message_Copy()`, `fipa_acl_message_Destroy()`, and `fipa_message_parameter_Parse()`.

7.24.2.10 `struct fipa_word_s* fipa_acl_message_s::protocol` [read]

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

Referenced by `fipa_acl_Compose()`, `fipa_acl_message_Copy()`, `fipa_acl_message_Destroy()`, and `fipa_message_parameter_Parse()`.

7.24.2.11 `struct fipa_expression_s* fipa_acl_message_s::conversation_id` [read]

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

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), and fipa_message_parameter_Parse().

7.24.2.12 struct fipa_expression_s* fipa_acl_message_s::reply_with [read]

Definition at line 102 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), and fipa_message_parameter_Parse().

7.24.2.13 struct fipa_expression_s* fipa_acl_message_s::in_reply_to [read]

Definition at line 104 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), and fipa_message_parameter_Parse().

7.24.2.14 struct fipa_DateTime_s* fipa_acl_message_s::reply_by [read]

Definition at line 106 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), and fipa_message_parameter_Parse().

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

- [/home/dko/projects/mobilec/trunk/src/include/fipa_acl.h](#)

7.25 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

7.25.1 Detailed Description

Definition at line 19 of file `fipa_acl_envelope.h`.

7.25.2 Field Documentation

7.25.2.1 struct fipa_agent_identifier_set_s* fipa_acl_Param_s::to [read]

Definition at line 21 of file `fipa_acl_envelope.h`.

Referenced by `acc_Thread()`, `fipa_acl_Param_Copy()`, `fipa_acl_Param_Destroy()`, and `fipa_envelope_HandleTo()`.

7.25.2.2 struct fipa_agent_identifier_s* fipa_acl_Param_s::from [read]

Definition at line 22 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_Param_Copy()`, `fipa_acl_Param_Destroy()`, and `fipa_envelope_HandleFrom()`.

7.25.2.3 char* fipa_acl_Param_s::comments

Definition at line 23 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_Param_Copy()`, `fipa_acl_Param_Destroy()`, and `fipa_envelope_HandleComments()`.

7.25.2.4 char* fipa_acl_Param_s::acl_representation

Definition at line 24 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_Param_Copy()`, `fipa_acl_Param_Destroy()`, and `fipa_envelope_HandleAclRepresentation()`.

7.25.2.5 char* fipa_acl_Param_s::payload_length

Definition at line 25 of file fipa_acl_envelope.h.

Referenced by fipa_acl_Param_Copy(), fipa_acl_Param_Destroy(), and fipa_envelope_HandlePayloadLength().

7.25.2.6 char* fipa_acl_Param_s::payload_encoding

Definition at line 26 of file fipa_acl_envelope.h.

Referenced by fipa_acl_Param_Copy(), fipa_acl_Param_Destroy(), and fipa_envelope_HandlePayloadEncoding().

7.25.2.7 struct fipa_DateTime_s* fipa_acl_Param_s::date [read]

Definition at line 27 of file fipa_acl_envelope.h.

Referenced by fipa_acl_Param_Copy(), fipa_acl_Param_Destroy(), and fipa_envelope_HandleDate().

7.25.2.8 struct fipa_agent_identifier_set_s* fipa_acl_Param_s::intended_receiver [read]

Definition at line 28 of file fipa_acl_envelope.h.

Referenced by fipa_acl_Param_Copy(), fipa_acl_Param_Destroy(), and fipa_envelope_HandleIntendedReceiver().

7.25.2.9 struct fipa_acl_envelope_Received_s* fipa_acl_Param_s::received [read]

Definition at line 29 of file fipa_acl_envelope.h.

Referenced by fipa_acl_Param_Copy(), fipa_acl_Param_Destroy(), and fipa_envelope_HandleReceived().

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

- /home/dko/projects/mobilec/trunk/src/include/fipa_acl_envelope.h

7.26 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](#)

7.26.1 Detailed Description

Definition at line 152 of file [fipa_acl.h](#).

7.26.2 Field Documentation

7.26.2.1 char* fipa_agent_identifier_s::name

Definition at line 154 of file [fipa_acl.h](#).

Referenced by [acc_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__to\(\)](#), [MC_AclAddReceiver\(\)](#), [MC_AclAddReplyTo\(\)](#), [MC_AclSend\(\)](#), and [MC_AclSetSender\(\)](#).

7.26.2.2 struct fipa_url_sequence_s* fipa_agent_identifier_s::addresses [read]

Definition at line 155 of file [fipa_acl.h](#).

Referenced by [fipa_agent_identifier_Compose\(\)](#), [fipa_agent_identifier_Copy\(\)](#), [fipa_agent_identifier_Destroy\(\)](#), [fipa_envelope_Compose__from\(\)](#), [fipa_envelope_Compose__to\(\)](#), [fipa_envelope_ParseAddresses\(\)](#), [MC_AclAddReceiver\(\)](#), [MC_AclAddReplyTo\(\)](#), [MC_AclSend\(\)](#), and [MC_AclSetSender\(\)](#).

7.26.2.3 struct fipa_agent_identifier_set_s* fipa_agent_identifier_s::resolvers [read]

Definition at line 156 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/trunk/src/include/fipa_acl.h](#)

7.27 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](#)

7.27.1 Detailed Description

Definition at line 138 of file [fipa_acl.h](#).

7.27.2 Field Documentation

7.27.2.1 int fipa_agent_identifier_set_s::num

Definition at line 140 of file [fipa_acl.h](#).

Referenced by [acc_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\(\)](#).

7.27.2.2 int fipa_agent_identifier_set_s::retain_order

Definition at line 141 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\(\)](#).

7.27.2.3 struct fipa_agent_identifier_s** fipa_agent_identifier_set_s::fipa_agent_identifiers [read]

Definition at line 142 of file [fipa_acl.h](#).

Referenced by [acc_Thread\(\)](#), [AP_QUEUE_STD_DEFN_TEMPLATE\(\)](#), [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\(\)](#).

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

- [/home/dko/projects/mobilec/trunk/src/include/fipa_acl.h](#)

7.28 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`

7.28.1 Detailed Description

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

7.28.2 Field Documentation

7.28.2.1 `int fipa_DateTime_s::year`

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

Referenced by `fipa_DateTime_Compose()`.

7.28.2.2 `int fipa_DateTime_s::month`

Definition at line 209 of file `fipa_acl.h`.

Referenced by `fipa_DateTime_Compose()`.

7.28.2.3 `int fipa_DateTime_s::day`

Definition at line 210 of file `fipa_acl.h`.

Referenced by `fipa_DateTime_Compose()`.

7.28.2.4 `int fipa_DateTime_s::hour`

Definition at line 211 of file `fipa_acl.h`.

Referenced by `fipa_DateTime_Compose()`.

7.28.2.5 `int fipa_DateTime_s::minute`

Definition at line 212 of file `fipa_acl.h`.

Referenced by `fipa_DateTime_Compose()`.

7.28.2.6 int fipa_DateTime_s::second

Definition at line 213 of file fipa_acl.h.

Referenced by fipa_DateTime_Compose().

7.28.2.7 int fipa_DateTime_s::millisecond

Definition at line 214 of file fipa_acl.h.

Referenced by fipa_DateTime_Compose().

7.28.2.8 char fipa_DateTime_s::sign

Definition at line 215 of file fipa_acl.h.

Referenced by fipa_DateTime_Compose(), and fipa_datetime_Parse().

7.28.2.9 int fipa_DateTime_s::is_utc

Definition at line 216 of file fipa_acl.h.

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

- [/home/dko/projects/mobilec/trunk/src/include/fipa_acl.h](#)

7.29 fipa_expression_s Struct Reference

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

Data Fields

- enum [fipa_expression_type_e](#) type
- union [fipa_expression_s::content_u](#) content

Data Structures

- union [content_u](#)

7.29.1 Detailed Description

Definition at line 165 of file [fipa_acl.h](#).

7.29.2 Field Documentation

7.29.2.1 enum [fipa_expression_type_e](#) [fipa_expression_s::type](#)

Definition at line 167 of file [fipa_acl.h](#).

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), [fipa_expression_Destroy\(\)](#), and [fipa_expression_Parse\(\)](#).

7.29.2.2 union [fipa_expression_s::content_u](#) [fipa_expression_s::content](#)

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), and [fipa_expression_Destroy\(\)](#).

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

- [/home/dko/projects/mobilec/trunk/src/include/fipa_acl.h](#)

7.30 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](#)

7.30.1 Detailed Description

Definition at line 168 of file [fipa_acl.h](#).

7.30.2 Field Documentation

7.30.2.1 struct [fipa_word_s](#)* [fipa_expression_s::content_u::word](#) [read]

Definition at line 170 of file [fipa_acl.h](#).

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), and [fipa_expression_Destroy\(\)](#).

7.30.2.2 struct [fipa_string_s](#)* [fipa_expression_s::content_u::string](#) [read]

Definition at line 171 of file [fipa_acl.h](#).

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), and [fipa_expression_Destroy\(\)](#).

7.30.2.3 struct [fipa_number_s](#)* [fipa_expression_s::content_u::number](#) [read]

Definition at line 172 of file [fipa_acl.h](#).

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), and [fipa_expression_Destroy\(\)](#).

7.30.2.4 struct [fipa_DateTime_s](#)* [fipa_expression_s::content_u::datetime](#) [read]

Definition at line 173 of file [fipa_acl.h](#).

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), and [fipa_expression_Destroy\(\)](#).

7.30.2.5 struct [fipa_expression_s](#)** [fipa_expression_s::content_u::expression](#) [read]

Definition at line 174 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/trunk/src/include/fipa_acl.h](#)

7.31 fipa_message_string_s Struct Reference

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

Data Fields

- char * [message](#)
- char * [parse](#)

7.31.1 Detailed Description

Definition at line 115 of file `fipa_acl.h`.

7.31.2 Field Documentation

7.31.2.1 char* fipa_message_string_s::message

Definition at line 117 of file `fipa_acl.h`.

Referenced by `acc_Thread()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandleReceived()`, `fipa_message_string_Copy()`, and `fipa_message_string_Destroy()`.

7.31.2.2 char* fipa_message_string_s::parse

Definition at line 118 of file `fipa_acl.h`.

Referenced by `acc_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/trunk/src/include/fipa_acl.h`

7.32 fipa_number_s Struct Reference

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

Data Fields

- char * [str](#)

7.32.1 Detailed Description

Definition at line 236 of file fipa_acl.h.

7.32.2 Field Documentation

7.32.2.1 char* fipa_number_s::str

Definition at line 238 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/trunk/src/include/[fipa_acl.h](#)

7.33 fipa_string_s Struct Reference

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

Data Fields

- char * [content](#)

7.33.1 Detailed Description

Definition at line 195 of file fipa_acl.h.

7.33.2 Field Documentation

7.33.2.1 char* fipa_string_s::content

Definition at line 197 of file fipa_acl.h.

Referenced by fipa_string_Compose(), fipa_string_Copy(), fipa_string_Destroy(), fipa_string_Parse(), and MC_AclSetContent().

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

- [/home/dko/projects/mobilec/trunk/src/include/fipa_acl.h](#)

7.34 fipa_url_s Struct Reference

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

Data Fields

- char * [str](#)

7.34.1 Detailed Description

Definition at line 225 of file fipa_acl.h.

7.34.2 Field Documentation

7.34.2.1 char* fipa_url_s::str

Definition at line 227 of file fipa_acl.h.

Referenced by fipa_envelope_Compose__from(), fipa_envelope_Compose__to(), fipa_envelope_HandleReceived(), fipa_envelope_ParseAddresses(), fipa_url_Compose(), fipa_url_Copy(), fipa_url_Destroy(), MC_AclAddReceiver(), MC_AclAddReplyTo(), MC_AclSend(), and MC_AclSetSender().

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

- [/home/dko/projects/mobilec/trunk/src/include/fipa_acl.h](#)

7.35 fipa_url_sequence_s Struct Reference

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

Data Fields

- [int num](#)
- [struct fipa_url_s ** urls](#)

7.35.1 Detailed Description

Definition at line 126 of file [fipa_acl.h](#).

7.35.2 Field Documentation

7.35.2.1 int fipa_url_sequence_s::num

Definition at line 128 of file [fipa_acl.h](#).

Referenced by [fipa_agent_identifier_Compose\(\)](#), [fipa_envelope_Compose__from\(\)](#), [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\(\)](#).

7.35.2.2 struct fipa_url_s** fipa_url_sequence_s::urls [read]

Definition at line 129 of file [fipa_acl.h](#).

Referenced by [fipa_envelope_Compose__from\(\)](#), [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\(\)](#).

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

- [/home/dko/projects/mobilec/trunk/src/include/fipa_acl.h](#)

7.36 fipa_word_s Struct Reference

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

Data Fields

- char * [content](#)

7.36.1 Detailed Description

Definition at line 184 of file fipa_acl.h.

7.36.2 Field Documentation

7.36.2.1 char* fipa_word_s::content

Definition at line 186 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_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/trunk/src/include/[fipa_acl.h](#)

7.37 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](#)

7.37.1 Detailed Description

Definition at line 1 of file class.cxx.

7.37.2 Constructor & Destructor Documentation

7.37.2.1 foo_c::foo_c (float *f*, int *b*)

7.37.2.2 foo_c::~~foo_c ()

Definition at line 81 of file class.cxx.

7.37.3 Member Function Documentation

7.37.3.1 int foo_c::get_bar () [inline]

Definition at line 13 of file class.cxx.

References [bar](#).

7.37.3.2 float foo_c::get_foo () [inline]

Definition at line 20 of file class.cxx.

References foo.

7.37.3.3 void foo_c::set_bar (int *b*) [inline]

Definition at line 27 of file class.cxx.

References bar.

7.37.3.4 void foo_c::set_foo (float *f*) [inline]

Definition at line 34 of file class.cxx.

References foo.

7.37.3.5 void foo_c::set_foobar (float *f*, int *b* = 0) [inline]

Definition at line 41 of file class.cxx.

References bar, and foo.

7.37.3.6 int foo_c::get_global () [inline, protected]

Definition at line 54 of file class.cxx.

References global.

7.37.3.7 int foo_c::get_barfoo () [inline]

Definition at line 67 of file class.cxx.

References barfoo.

7.37.4 Field Documentation**7.37.4.1 float foo_c::foo** [private]

Definition at line 3 of file class.cxx.

Referenced by get_foo(), set_foo(), and set_foobar().

7.37.4.2 int foo_c::bar [private]

Definition at line 4 of file class.cxx.

Referenced by get_bar(), set_bar(), and set_foobar().

7.37.4.3 `int foo_c::global` `[static, protected]`

Definition at line 50 of file class.cxx.

Referenced by `get_global()`.

7.37.4.4 `int foo_c::barfoo` `[private]`

Definition at line 61 of file class.cxx.

Referenced by `get_barfoo()`.

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

- `/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/class.cxx`

7.38 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](#)

7.38.1 Detailed Description

Definition at line 1 of file struct.cxx.

7.38.2 Constructor & Destructor Documentation

7.38.2.1 foo_s::foo_s (float *f*, int *b*)

Definition at line 39 of file struct.cxx.

References [bar](#), and [foo](#).

7.38.2.2 foo_s::~~foo_s ()

Definition at line 46 of file struct.cxx.

7.38.3 Member Function Documentation

7.38.3.1 int foo_s::get_bar () [inline]

Definition at line 11 of file struct.cxx.

References [bar](#).

7.38.3.2 float foo_s::get_foo () [inline]

Definition at line 18 of file struct.cxx.

References [foo](#).

7.38.3.3 void foo_s::set_bar (int *b*) [inline]

Definition at line 25 of file struct.cxx.

References bar.

7.38.3.4 void foo_s::set_foo (float *f*) [inline]

Definition at line 32 of file struct.cxx.

References foo.

7.38.4 Field Documentation**7.38.4.1 float foo_s::foo**

Definition at line 3 of file struct.cxx.

Referenced by foo_s(), get_foo(), and set_foo().

7.38.4.2 int foo_s::bar

Definition at line 4 of file struct.cxx.

Referenced by foo_s(), get_bar(), and set_bar().

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

- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/[struct.cxx](#)

7.39 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]

7.39.1 Detailed Description

HAVEGE state structure.

Definition at line 16 of file havege.h.

7.39.2 Field Documentation

7.39.2.1 int havege_state::PT1

Definition at line 18 of file havege.h.

Referenced by `havege_fill()`.

7.39.2.2 int havege_state::PT2

Definition at line 18 of file havege.h.

Referenced by `havege_fill()`.

7.39.2.3 int havege_state::offset[2]

Definition at line 18 of file havege.h.

Referenced by `havege_fill()`, and `havege_rand()`.

7.39.2.4 int havege_state::pool[COLLECT_SIZE]

Definition at line 19 of file havege.h.

Referenced by `havege_rand()`.

7.39.2.5 int havege_state::WALK[8192]

Definition at line 20 of file havege.h.

Referenced by `havege_fill()`.

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

- </home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/havege.h>

7.40 host_id_s Struct Reference

```
#include <host_id.h>
```

Data Fields

- char * [hostname](#)
- [int](#) port

7.40.1 Detailed Description

Definition at line 35 of file `host_id.h`.

7.40.2 Field Documentation

7.40.2.1 char* host_id_s::hostname

Definition at line 37 of file `host_id.h`.

7.40.2.2 int host_id_s::port

Definition at line 38 of file `host_id.h`.

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

- `/home/dko/projects/mobilec/trunk/src/include/host_id.h`

7.41 hr_time Struct Reference

timer structure

```
#include <timing.h>
```

Data Fields

- unsigned char [opaque](#) [32]

7.41.1 Detailed Description

timer structure

Definition at line 16 of file timing.h.

7.41.2 Field Documentation

7.41.2.1 unsigned char hr_time::opaque[32]

Definition at line 18 of file timing.h.

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

- [/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/timing.h](#)

7.42 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](#)

7.42.1 Detailed Description

Definition at line 38 of file `interpreter_variable_data.h`.

7.42.2 Field Documentation

7.42.2.1 char* interpreter_variable_data_s::name

Definition at line 39 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()`, and `interpreter_variable_data_New()`.

7.42.2.2 int interpreter_variable_data_s::size

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

Referenced by `agent_AddPersistentVariable()`, `interpreter_variable_data_Copy()`, `interpreter_variable_data_Initialize()`, `interpreter_variable_data_InitializeFromAgent()`, and `interpreter_variable_data_New()`.

7.42.2.3 ChType_t interpreter_variable_data_s::data_type

Definition at line 41 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()`, and `MC_GetAgentReturnData()`.

7.42.2.4 int interpreter_variable_data_s::array_dim

Definition at line 42 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()`, and `MC_GetAgentReturnData()`.

7.42.2.5 `int* interpreter_variable_data_s::array_extent`

Definition at line 43 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()`, and `MC_GetAgentReturnData()`.

7.42.2.6 `void* interpreter_variable_data_s::data`

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_Destroy()`, `interpreter_variable_data_Initialize()`, `interpreter_variable_data_InitializeFromAgent()`, `interpreter_variable_data_New()`, and `MC_AgentVariableRetrieve()`.

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

- `/home/dko/projects/mobilec/trunk/src/include/interpreter_variable_data.h`

7.43 list_s Struct Reference

```
#include <list.h>
```

Data Fields

- [listNode_p listhead](#)
- [int size](#)

7.43.1 Detailed Description

Definition at line 38 of file list.h.

7.43.2 Field Documentation

7.43.2.1 listNode_p list_s::listhead

Definition at line 40 of file list.h.

Referenced by `barrier_queue_Add()`, `barrier_queue_Get()`, `ListAdd()`, `ListDelete()`, `ListGetHead()`, `ListInitialize()`, `ListPop()`, `ListSearch()`, `ListTerminate()`, `syncListAddNode()`, and `syncListFind()`.

7.43.2.2 int list_s::size

Definition at line 41 of file list.h.

Referenced by `barrier_queue_Delete()`, `list_pGetSize()`, `ListAdd()`, `ListDelete()`, `ListInitialize()`, `ListPop()`, `ListSearch()`, `ListTerminate()`, `syncListDelete()`, and `syncListRemove()`.

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

- `/home/dko/projects/mobilec/trunk/src/mc_list/list.h`

7.44 listNode_s Struct Reference

```
#include <list.h>
```

Data Fields

- DATA [node_data](#)
- struct [listNode_s](#) * [next](#)

7.44.1 Detailed Description

Definition at line 29 of file list.h.

7.44.2 Field Documentation

7.44.2.1 DATA listNode_s::node_data

Definition at line 31 of file list.h.

Referenced by [AP_QUEUE_STD_DEFN_TEMPLATE\(\)](#), [barrier_queue_Add\(\)](#), [barrier_queue_Get\(\)](#), [df_SearchForService\(\)](#), [ListAdd\(\)](#), [ListDelete\(\)](#), [ListGetHead\(\)](#), [ListPop\(\)](#), [ListSearch\(\)](#), [syncListAddNode\(\)](#), and [syncListFind\(\)](#).

7.44.2.2 struct listNode_s* listNode_s::next [read]

Definition at line 32 of file list.h.

Referenced by [AP_QUEUE_STD_DEFN_TEMPLATE\(\)](#), [barrier_queue_Add\(\)](#), [barrier_queue_Get\(\)](#), [df_SearchForService\(\)](#), [ListAdd\(\)](#), [ListDelete\(\)](#), [ListPop\(\)](#), [ListSearch\(\)](#), [syncListAddNode\(\)](#), and [syncListFind\(\)](#).

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

- [/home/dko/projects/mobilec/trunk/src/mc_list/list.h](#)

7.45 mc_platform_s Struct Reference

```
#include <mc_platform.h>
```

Data Fields

- `int` `err`
- `char *` `hostname`
- `unsigned short` `port`
- `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`

7.45.1 Detailed Description

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

7.45.2 Field Documentation

7.45.2.1 `int mc_platform_s::err`

Definition at line 50 of file `mc_platform.h`.

Referenced by `agent_Initialize()`, `mc_platform_Initialize()`, `message_InitializeFromAgent()`, and `message_InitializeFromString()`.

7.45.2.2 `char* mc_platform_s::hostname`

Definition at line 55 of file `mc_platform.h`.

Referenced by `fipa_envelope_Compose__from()`, `mc_platform_Initialize()`, `message_InitializeFromAgent()`, `message_InitializeFromString()`, and `message_queue_SendOutgoing()`.

7.45.2.3 `unsigned short mc_platform_s::port`

Definition at line 56 of file `mc_platform.h`.

Referenced by `fipa_envelope_Compose__from()`, `listen_Thread()`, `mc_platform_Initialize()`, `message_InitializeFromAgent()`, `message_InitializeFromString()`, and `message_queue_SendOutgoing()`.

7.45.2.4 `message_queue_p mc_platform_s::asm_message_queue`

Definition at line 59 of file `mc_platform.h`.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `mc_platform_Destroy()`, and `mc_platform_Initialize()`.

7.45.2.5 `message_queue_p mc_platform_s::message_queue`

Definition at line 63 of file `mc_platform.h`.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `MC_End()`, `MC_LoadAgentFromFile()`, `mc_platform_Destroy()`, `mc_platform_Initialize()`, `MC_SendAgentMigrationMessage()`, and `MC_SendAgentMigrationMessageFile()`.

7.45.2.6 `agent_queue_p mc_platform_s::agent_queue`

Definition at line 64 of file `mc_platform.h`.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `MC_AddAgent()`, `MC_FindAgentByID()`, `MC_FindAgentByName()`, `MC_GetAllAgents()`, `mc_platform_Destroy()`, `mc_platform_Initialize()`, `MC_RetrieveAgent()`, `MC_WaitAgent()`, and `MC_WaitRetrieveAgent()`.

7.45.2.7 `connection_queue_p mc_platform_s::connection_queue`

Definition at line 65 of file `mc_platform.h`.

Referenced by `acc_Thread()`, `listen_Thread()`, `MC_End()`, `mc_platform_Destroy()`, and `mc_platform_Initialize()`.

7.45.2.8 `df_p mc_platform_s::df`

Definition at line 67 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()`, and `request_handler_REGISTER()`.

7.45.2.9 ams_p mc_platform_s::ams

Definition at line 68 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), agent_RunChScriptThread(), ams_Start(), ams_Thread(), MC_AddAgent(), MC_End(), MC_MainLoop(), mc_platform_Destroy(), mc_platform_Initialize(), and MC_SetAgentStatus().

7.45.2.10 acc_p mc_platform_s::acc

Definition at line 69 of file mc_platform.h.

Referenced by acc_Start(), listen_Thread(), MC_End(), mc_platform_Destroy(), and mc_platform_Initialize().

7.45.2.11 cmd_prompt_p mc_platform_s::cmd_prompt

Definition at line 70 of file mc_platform.h.

Referenced by cmd_prompt_Start(), MC_End(), mc_platform_Destroy(), and mc_platform_Initialize().

7.45.2.12 syncList_p mc_platform_s::syncList

Definition at line 75 of file mc_platform.h.

Referenced by MC_CondBroadcast(), MC_CondReset(), MC_CondSignal(), MC_CondWait(), MC_MutexLock(), MC_MutexUnlock(), mc_platform_Initialize(), MC_SemaphorePost(), MC_SemaphoreWait(), MC_SyncDelete(), and MC_SyncInit().

7.45.2.13 barrier_queue_p mc_platform_s::barrier_queue

Definition at line 76 of file mc_platform.h.

Referenced by MC_Barrier(), MC_BarrierDelete(), MC_BarrierInit(), mc_platform_Destroy(), and mc_platform_Initialize().

7.45.2.14 listen_thread_arg_p mc_platform_s::listen_thread_arg

Definition at line 78 of file mc_platform.h.

7.45.2.15 listen_thread_arg_p mc_platform_s::client_thread_arg

Definition at line 79 of file mc_platform.h.

7.45.2.16 int mc_platform_s::default_agentstatus

Definition at line 81 of file mc_platform.h.

Referenced by agent_Initialize(), mc_platform_Initialize(), and MC_SetDefaultAgentStatus().

7.45.2.17 int mc_platform_s::stack_size[MC_THREAD_ALL]

Definition at line 83 of file mc_platform.h.

Referenced by acc_Start(), agent_RunChScript(), ams_Start(), cmd_prompt_Start(), df_Start(), and mc_platform_Initialize().

7.45.2.18 ChOptions_t* mc_platform_s::interp_options

Definition at line 85 of file mc_platform.h.

Referenced by agent_RunChScriptThread(), MC_ChInitializeOptions(), mc_platform_Destroy(), and mc_platform_Initialize().

7.45.2.19 COND_T* mc_platform_s::MC_signal_cond

Definition at line 88 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), agent_RunChScriptThread(), mc_platform_Destroy(), mc_platform_Initialize(), and MC_WaitSignal().

7.45.2.20 COND_T* mc_platform_s::MC_sync_cond

Definition at line 89 of file mc_platform.h.

Referenced by mc_platform_Destroy(), and mc_platform_Initialize().

7.45.2.21 MUTEX_T* mc_platform_s::MC_signal_lock

Definition at line 90 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), agent_RunChScriptThread(), mc_platform_Destroy(), mc_platform_Initialize(), and MC_WaitSignal().

7.45.2.22 MUTEX_T* mc_platform_s::MC_sync_lock

Definition at line 91 of file mc_platform.h.

Referenced by mc_platform_Destroy(), and mc_platform_Initialize().

7.45.2.23 enum MC_Signal_e mc_platform_s::MC_signal

Definition at line 92 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), agent_RunChScriptThread(), mc_platform_Initialize(), MC_ResetSignal(), and MC_WaitSignal().

7.45.2.24 enum MC_SteerCommand_e mc_platform_s::MC_steer_command

Definition at line 95 of file mc_platform.h.

Referenced by MC_SendSteerCommand(), MC_Steer(), and MC_SteerControl().

7.45.2.25 MUTEX_T* mc_platform_s::MC_steer_lock

Definition at line 96 of file mc_platform.h.

Referenced by mc_platform_Destroy(), mc_platform_Initialize(), MC_SendSteerCommand(), MC_Steer(), and MC_SteerControl().

7.45.2.26 COND_T* mc_platform_s::MC_steer_cond

Definition at line 97 of file mc_platform.h.

Referenced by mc_platform_Destroy(), mc_platform_Initialize(), MC_SendSteerCommand(), and MC_SteerControl().

7.45.2.27 int mc_platform_s::giant

Definition at line 101 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().

7.45.2.28 MUTEX_T* mc_platform_s::giant_lock

Definition at line 102 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().

7.45.2.29 COND_T* mc_platform_s::giant_cond

Definition at line 103 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), mc_platform_Destroy(), mc_platform_Initialize(), and MC_ResetSignal().

7.45.2.30 int mc_platform_s::quit

Definition at line 105 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), ams_Thread(), df_Thread(), MC_End(), and mc_platform_Initialize().

7.45.2.31 MUTEX_T* mc_platform_s::quit_lock

Definition at line 106 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), ams_Thread(), df_Thread(), MC_End(), mc_platform_Destroy(), and mc_platform_Initialize().

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

- /home/dko/projects/mobilec/trunk/src/include/[mc_platform.h](#)

7.46 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`

7.46.1 Detailed Description

Definition at line 36 of file mc_rwlock.h.

7.46.2 Field Documentation

7.46.2.1 int mc_rwlock_s::num_readers

Definition at line 37 of file mc_rwlock.h.

Referenced by mc_rwlock_init(), mc_rwlock_rdlock(), mc_rwlock_rdunlock(), and mc_rwlock_wrlock().

7.46.2.2 int mc_rwlock_s::write_flag

Definition at line 38 of file mc_rwlock.h.

Referenced by mc_rwlock_init(), mc_rwlock_rdlock(), mc_rwlock_wrlock(), and mc_rwlock_wrunlock().

7.46.2.3 int mc_rwlock_s::write_request

Definition at line 39 of file mc_rwlock.h.

Referenced by mc_rwlock_init(), mc_rwlock_rdlock(), and mc_rwlock_wrlock().

7.46.2.4 MUTEX_T* mc_rwlock_s::lock

Definition at line 40 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().

7.46.2.5 COND_T* mc_rwlock_s::cond

Definition at line 41 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().

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

- /home/dko/projects/mobilec/trunk/src/include/[mc_rwlock.h](#)

7.47 LibMC::MCAclMessage Class Reference

Encapsulates ACL messages in the Mobile-C library.

Public Types

- enum [MC_FipaPerformative_e](#)
Enum for describing the type of an ACL message.

Public Member Functions

- [MCAclMessage](#) ()
Default constructor.
- void [New](#) ()
Creates a new, blank ACL message.
- [MCAclMessage Reply](#) ([MCAclMessage](#) acl_message)
Creates an ACL message that is a response to the argument.
- int [SetPerformative](#) ([MC_FipaPerformative_e](#) performative)
Sets the performative field of the message.
- int [SetSender](#) (String name, String address)
Sets the sender field of the message.
- int [AddReceiver](#) (String name, String address)
Adds a receiver to the list of receivers.
- int [AddReplyTo](#) (String name, String address)
Adds a "reply-to" field to the message.
- int [SetContent](#) (String content)
Sets the content field of the message.
- int [Destroy](#) ()
Destroys a message.

Properties

- internal IntPtr [AclMsg](#) [get, set]

Private Member Functions

- internal [MCAclMessage](#) (IntPtr msg)

Private Attributes

- IntPtr [aclmsg_p](#)

7.47.1 Detailed Description

Encapsulates ACL messages in the Mobile-C library.

This class contains a pointer to an ACL message in the Mobile-C library. Functions are provided to send the message, set its various fields, and destroy the message.

Definition at line 72 of file MCAclMessage.cs.

7.47.2 Member Enumeration Documentation

7.47.2.1 enum LibMC::MCAclMessage::MC_FipaPerformative_e

Enum for describing the type of an ACL message.

Note:

This enum is pulled directly from the Mobile-C library.

Enumerator:

FIPA_ERROR Fipa performative enum value
FIPA_ZERO Fipa performative enum value
FIPA_ACCEPT_PROPOSAL Fipa performative enum value
FIPA_AGREE Fipa performative enum value
FIPA_CANCEL Fipa performative enum value
FIPA_CALL_FOR_PROPOSAL Fipa performative enum value
FIPA_CONFIRM Fipa performative enum value
FIPA_DISCONFIRM Fipa performative enum value
FIPA_FAILURE Fipa performative enum value
FIPA_INFORM Fipa performative enum value
FIPA_INFORM_IF Fipa performative enum value
FIPA_INFORM_REF Fipa performative enum value
FIPA_NOT_UNDERSTOOD Fipa performative enum value
FIPA_PROPOGATE Fipa performative enum value
FIPA_PROPOSE Fipa performative enum value
FIPA_PROXY Fipa performative enum value
FIPA_QUERY_IF Fipa performative enum value
FIPA_QUERY_REF Fipa performative enum value
FIPA_REFUSE Fipa performative enum value
FIPA_REJECT_PROPOSAL Fipa performative enum value
FIPA_REQUEST Fipa performative enum value
FIPA_REQUEST_WHEN Fipa performative enum value
FIPA_REQUEST_WHenever Fipa performative enum value
FIPA_SUBSCRIBE Fipa performative enum value

Definition at line 79 of file MCAclMessage.cs.

7.47.3 Constructor & Destructor Documentation

7.47.3.1 LibMC::MCAclMessage::MCAclMessage () [inline]

Default constructor.

Creates an empty ACL message object.

Definition at line 114 of file MCAclMessage.cs.

References `aclmsg_p`.

Referenced by `Reply()`.

7.47.3.2 internal LibMC::MCAclMessage::MCAclMessage (IntPtr msg) [inline, private]

Definition at line 120 of file MCAclMessage.cs.

References `AclMsg`.

7.47.4 Member Function Documentation

7.47.4.1 void LibMC::MCAclMessage::New () [inline]

Creates a new, blank ACL message.

Creates a new ACL message. The message is blank but valid.

Definition at line 153 of file MCAclMessage.cs.

References `AclMsg`.

7.47.4.2 MCAclMessage LibMC::MCAclMessage::Reply (MCAclMessage acl_message) [inline]

Creates an ACL message that is a response to the argument.

Creates an ACL message to respond to the argument.

Parameters:

acl_message The message from which to create the reply.

Returns:

A new ACL message that is a response to the argument or an empty message if there is an error.

Definition at line 167 of file MCAclMessage.cs.

References `AclMsg`, and `MCAclMessage()`.

7.47.4.3 int LibMC::MCAclMessage::SetPerformative (MC_FipaPerformative_e performative) [inline]

Sets the performative field of the message.

Sets the performative field of the message.

Parameters:

performative The fipa_performative_e enum describing the message.

Returns:

The return value of the underlying MC_AclSetPerformative function.

Note:

The message must be a valid message or this function will fail.

Definition at line 186 of file MCAclMessage.cs.

References AclMsg.

7.47.4.4 int LibMC::MCAclMessage::SetSender (String *name*, String *address*) [inline]

Sets the sender field of the message.

Sets the performative field of the message.

Parameters:

name The name of the sending entity.

address The address of the sending entity.

Returns:

The return value of the underlying MC_AclSetSender function.

Note:

The message must be a valid message or this function will fail.

Definition at line 202 of file MCAclMessage.cs.

References AclMsg.

7.47.4.5 int LibMC::MCAclMessage::AddReceiver (String *name*, String *address*) [inline]

Adds a receiver to the list of receivers.

Adds a receiver to the list of receivers for the message.

Parameters:

name The name of the receiver.

address The address of the receiver.

Returns:

The return value of the underlying MC_AclAddReceiver function.

Note:

The message must be a valid message or this function will fail.

Definition at line 218 of file MCAclMessage.cs.

References AclMsg.

7.47.4.6 int LibMC::MCAclMessage::AddReplyTo (String *name*, String *address*) [inline]

Adds a "reply-to" field to the message.

Adds a "reply-to" field to the message. The reply-to field overrides the sender field when creating a reply.

Parameters:

name The name of the receiver.

address The address of the receiver.

Returns:

The return value of the underlying MC_AclAddAddReplyTo function.

Note:

The message must be a valid message or this function will fail.

Definition at line 235 of file MCAclMessage.cs.

References AclMsg.

7.47.4.7 int LibMC::MCAclMessage::SetContent (String *content*) [inline]

Sets the content field of the message.

Sets the content field of the message.

Parameters:

content The string to copy to the content field.

Returns:

The return value of the underlying MC_AclSetContent function.

Note:

The message must be a valid message or this function will fail.

Definition at line 250 of file MCAclMessage.cs.

References AclMsg.

7.47.4.8 int LibMC::MCAclMessage::Destroy () [inline]

Destroys a message.

This function destroys a message in the Mobile-C library. It releases the underlying memory and must be called when the message is no longer needed.

Returns:

The return value of the underlying MC_AclDestroy function.

Note:

The message must be a valid message or this function will fail. In addition, messages are not automatically destroyed by the garbage collector. Use care when creating messages and ensure they are properly destroyed.

Definition at line 269 of file MCAclMessage.cs.

References AclMsg, and aclmsg_p.

7.47.5 Field Documentation

7.47.5.1 IntPtr LibMC::MCAclMessage::aclmsg_p [private]

Definition at line 105 of file MCAclMessage.cs.

Referenced by Destroy(), and MCAclMessage().

7.47.6 Property Documentation

7.47.6.1 internal IntPtr LibMC::MCAclMessage::AclMsg [get, set, private]

Definition at line 129 of file MCAclMessage.cs.

Referenced by LibMC::MCAgent::AclPost(), LibMC::MCAgency::AclSend(), AddReceiver(), AddReplyTo(), Destroy(), MCAclMessage(), New(), Reply(), SetContent(), SetPerformative(), and SetSender().

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

- [/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAclMessage.cs](#)

7.48 LibMC::MCAgency Class Reference

Wrapper class for [MCAgency_t](#) structure.

Public Types

- enum [MCAgencyState](#)
Enum for describing the state of the agency.
- enum [ChShellType](#)
Ch shell type.
- enum [MC_ThreadIndex_e](#)
Enum for describing the different threads that Mobile-C uses.
- enum [MC_SteerCommand_e](#)
Available commands for MC_Steer.

Public Member Functions

- [MCAgency](#) ()
Default constructor.
- [int Initialize](#) ()
Starts the agency.
- [int End](#) ()
Stops and destroys the agency.
- [int ChInitializeOptions](#) ([ChShellType](#) shellType, String home)
Initializes Ch options for the agency.
- [int SetThreadsAllOn](#) ()
- [int SetThreadsAllOff](#) ()
Sets all threads for the agency to "off".
- [int SetThreadOn](#) ([MC_ThreadIndex_e](#) index)
Sets an individual thread for the agency to "on".
- [int SetThreadOff](#) ([MC_ThreadIndex_e](#) index)
Sets an individual thread for the agency to "off".
- [int HaltAgency](#) ()
Temporarily halts the agency.
- [int ResumeAgency](#) ()
Resumes a halted agency.

- [int SetDefaultAgentStatus](#) ([MCAgent.MC_AgentStatus_e](#) status)
Sets the default state of an agent in the agency.
- [MCAgent WaitRetrieveAgent](#) ()
Waits for an agent to arrive and returns the agent.
- [int WaitAgent](#) ()
Waits for an agent to arrive.
- [int SendAgentMigrationMessageFile](#) (String filename, String hostname, [int](#) port)
Sends an agent migration message file to an agency.
- [int LoadAgentMigrationMessageFile](#) (String filename)
Load an agent migration message.
- [int SendAgentMigrationMessage](#) (String message, String hostname, [int](#) port)
Sends an agent migration message to an agency.
- [int CondBroadcast](#) ([int](#) id)
Broadcast a condition signal.
- [int CondSignal](#) ([int](#) id)
Signal a condition.
- [int CondReset](#) ([int](#) id)
Reset a condition signal.
- [int CondWait](#) ([int](#) id)
Wait for a condition signal.
- [int MutexLock](#) ([int](#) id)
Lock a mutex.
- [int MutexUnlock](#) ([int](#) id)
Unlock a mutex.
- [int SemaphorePost](#) ([int](#) id)
Posts a semaphore.
- [int SemaphoreWait](#) ([int](#) id)
Wait for a semaphore to be posted.
- [int ResetSignal](#) ()
Reset an agency signal.
- [int SyncDelete](#) ([int](#) id)
Delete a synchronization variable.
- [int SyncInit](#) ([int](#) id)
Create a new synchronization variable.

- [int WaitSignal](#) ([int](#) signals)
Wait for agency signals.
- [int BarrierDelete](#) ([int](#) id)
Delete a barrier object.
- [int BarrierInit](#) ([int](#) id, [int](#) num_procs)
Create a new barrier.
- [MC_SteerCommand_e SteerControl](#) ()
Steering control function.
- [int Steer](#) ([IntPtr](#) funcptr, [IntPtr](#) arg)
Steering control function.
- [int RegisterService](#) ([MCAgent](#) agent, [int](#) agentID, [String](#) agentName, [String\[\]](#) serviceNames, [int](#) numServices)
Registers services in the agency.
- [int SearchForService](#) ([String](#) searchString, [IntPtr](#) agentNames, [IntPtr](#) serviceNames, [IntPtr](#) agentIDs, [IntPtr](#) numResults)
Searches for services in the agency.
- [int AddAgent](#) ([MCAgent](#) agent)
Add an agent to the agency.
- [MCAgent FindAgentByName](#) ([String](#) name)
Finds an agent by its name.
- [MCAgent FindAgentByID](#) ([int](#) id)
Find an agent by its ID.
- [MCAgent RetrieveAgent](#) ()
Retrieve an agent from the agency.
- [int AclSend](#) ([MCAclMessage](#) acl_message)
Send an ACL message to the agency.
- [int MainLoop](#) ()
Makes the agency wait indefinitely.

Properties

- [IntPtr Agency](#) [get, set]
- [int Port](#) [get, set]
Accessor for the port number of the agency.
- [MCAgencyState State](#) [get]
Accessor for the agency state.

Private Types

- enum [MC_Signal_e](#)
MobileC system signals.

Private Member Functions

- static [IntPtr _MC_Initialize](#) (int port, ref [MCAgencyOptions_t](#) options)
- static [int _MC_End](#) (IntPtr agency)
- static [int _MC_ChInitializeOptions](#) (IntPtr agency, [ChOptions_t](#) options)
- static [int _MC_InitializeAgencyOptions](#) (ref [MCAgencyOptions_t](#) options)
- static [int _MC_SetThreadsAllOn](#) (ref [MCAgencyOptions_t](#) options)
- static [int _MC_SetThreadsAllOff](#) (ref [MCAgencyOptions_t](#) options)
- static [int _MC_SetThreadOn](#) (ref [MCAgencyOptions_t](#) options, [MC_ThreadIndex_e](#) index)
- static [int _MC_SetThreadOff](#) (ref [MCAgencyOptions_t](#) options, [MC_ThreadIndex_e](#) index)
- static [int _MC_HaltAgency](#) (IntPtr agency)
- static [int _MC_ResumeAgency](#) (IntPtr agency)
- static [int _MC_SetDefaultAgentStatus](#) (IntPtr agency, [MCAgent.MC_AgentStatus_e](#) status)
- static [IntPtr _MC_WaitRetrieveAgent](#) (IntPtr agency)
- static [int _MC_WaitAgent](#) (IntPtr agency)
- static [int _MC_MainLoop](#) (IntPtr agency)
- static [int _MC_SendAgentMigrationMessageFile](#) (IntPtr agency, String filename, String hostname, int port)
- static [int _MC_SendAgentMigrationMessage](#) (IntPtr agency, String message, String hostname, int port)
- static [int _MC_CondBroadcast](#) (IntPtr agency, int id)
- static [int _MC_CondSignal](#) (IntPtr agency, int id)
- static [int _MC_CondReset](#) (IntPtr agency, int id)
- static [int _MC_CondWait](#) (IntPtr agency, int id)
- static [int _MC_MutexLock](#) (IntPtr agency, int id)
- static [int _MC_MutexUnlock](#) (IntPtr agency, int id)
- static [int _MC_SemaphorePost](#) (IntPtr agency, int id)
- static [int _MC_SemaphoreWait](#) (IntPtr agency, int id)
- static [int _MC_ResetSignal](#) (IntPtr agency)
- static [int _MC_SyncDelete](#) (IntPtr agency, int id)
- static [int _MC_SyncInit](#) (IntPtr agency, int id)
- static [int _MC_WaitSignal](#) (IntPtr agency, int signals)
- static [int _MC_BarrierDelete](#) (IntPtr agency, int id)
- static [int _MC_BarrierInit](#) (IntPtr agency, int id, int num_procs)
- static [MC_SteerCommand_e _MC_SteerControl](#) ()
- static [int _MC_Steer](#) (IntPtr agency, IntPtr funcptr, IntPtr arg)
- static [int _MC_RegisterService](#) (IntPtr agency, IntPtr agent, int agentID, String agentName, String[] serviceNames, int numServices)
- static [int _MC_SearchForService](#) (IntPtr agency, String searchString, IntPtr agentNames, IntPtr serviceNames, IntPtr agentIDs, IntPtr numResults)
- static [int _MC_AddAgent](#) (IntPtr agency, IntPtr agent)
- static internal [int _MC_DeleteAgent](#) (IntPtr agent)
- static [IntPtr _MC_FindAgentByName](#) (IntPtr agency, String name)
- static [IntPtr _MC_FindAgentByID](#) (IntPtr agency, int ID)

- static IntPtr [_MC_RetrieveAgent](#) (IntPtr agency)
- static internal [int _MC_GetAgentID](#) (IntPtr agent)
- static internal [String _MC_GetAgentName](#) (IntPtr agent)
- static internal [int _MC_GetAgentNumTasks](#) (IntPtr agent)
- static internal [MCAgent.MC_AgentStatus_e _MC_GetAgentStatus](#) (IntPtr agent)
- static internal [MCAgent.MC_AgentType_e _MC_GetAgentType](#) (IntPtr agent)
- static internal [String _MC_GetAgentXMLString](#) (IntPtr agent)
- static internal [int _MC_PrintAgentCode](#) (IntPtr agent)
- static internal [String _MC_RetrieveAgentCode](#) (IntPtr agent)
- static internal [int _MC_SetAgentStatus](#) (IntPtr agent, [MCAgent.MC_AgentStatus_e](#) status)
- static internal [int _MC_TerminateAgent](#) (IntPtr agent)
- static internal [int _MC_CallAgentFunc](#) (IntPtr agent, String funcName, IntPtr returnVal, IntPtr varg)
- static internal IntPtr [_MC_GetAgentExecEngine](#) (IntPtr agent)
- static internal [int _MC_GetAgentReturnData](#) (IntPtr agent, [int](#) task_num, IntPtr data, IntPtr dim, IntPtr extent)
- static internal IntPtr [_MC_AclNew](#) ()
- static internal [int _MC_AclPost](#) (IntPtr agent, IntPtr message)
- static internal IntPtr [_MC_AclReply](#) (IntPtr acl_message)
- static internal IntPtr [_MC_AclRetrieve](#) (IntPtr agent)
- static internal [int _MC_AclSend](#) (IntPtr agency, IntPtr acl_message)
- static internal IntPtr [_MC_AclWaitRetrieve](#) (IntPtr agent)
- static internal [int _MC_AclSetPerformative](#) (IntPtr acl, [MCAclMessage.MC_FipaPerformative_e](#) performative)
- static internal [int _MC_AclSetSender](#) (IntPtr acl, String name, String address)
- static internal [int _MC_AclAddReceiver](#) (IntPtr acl, String name, String address)
- static internal [int _MC_AclAddReplyTo](#) (IntPtr acl, String name, String address)
- static internal [int _MC_AclSetContent](#) (IntPtr acl, String content)
- static internal [int _MC_AclDestroy](#) (IntPtr acl)

Private Attributes

- IntPtr [agency_p](#) = IntPtr.Zero
- [MCAgencyOptions_t](#) options
- [int](#) port = -1
- [MCAgencyState](#) state = MCAgencyState.NoState
- const String [mcDll](#)

Data Structures

- struct [ChOptions_t](#)
ChOptions structures.
- struct [MCAgency_t](#)
- struct [MCAgencyOptions_t](#)

7.48.1 Detailed Description

Wrapper class for [MCAgency_t](#) structure.

This class provides an interface to the Mobile-C agency. Member functions for the class are generally overloaded versions of the respective functions in the Mobile-C library. The class maintains a pointer to the Mobile-C agency in unmanaged memory. The pointer is not accessible by the user.

Examples:

[LibMCCppEx/LibMCCppEx.cpp](#).

Definition at line 329 of file MCAgency.cs.

7.48.2 Member Enumeration Documentation

7.48.2.1 enum LibMC::MCAgency::MCAgencyState

Enum for describing the state of the agency.

This enum is used to determine whether or not certain actions should be permitted, such as halting, resuming, and ending an agency

Enumerator:

- NoState* Default, uninitialized state
- Initialized* Agency initialized, but not started
- Running* Agency is running
- Halted* Agency has been stopped (can be resumed)
- Ended* Agency is stopped (destroyed)

Definition at line 337 of file MCAgency.cs.

7.48.2.2 enum LibMC::MCAgency::ChShellType

Ch shell type.

Used to set the shell type for the Ch interpreter.

Enumerator:

- CH_REGULARCH* Default, regular shell
- CH_SAFECH* Safe shell

Definition at line 351 of file MCAgency.cs.

7.48.2.3 enum LibMC::MCAgency::MC_ThreadIndex_e

Enum for describing the different threads that Mobile-C uses.

These enums can be used to turn threads on and off before an agency is initialized.

Note:

This enum is pulled directly from the Mobile-C library.

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 79 of file MCEExports.cs.

7.48.2.4 enum LibMC::MCAgency::MC_SteerCommand_e

Available commands for MC_Steer.

Note:

This enum is pulled directly from the Mobile-C library.

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 94 of file MCEExports.cs.

7.48.2.5 enum LibMC::MCAgency::MC_Signal_e [private]

MobileC system signals.

Each signal is activated after the corresponding action. i.e. The 'MC_RECV_MESSAGE' signal is activated after a message is received.

Note:

This enum is pulled directly from the Mobile-C library.

See also:

[MC_WaitSignal\(\)](#), [MC_ResetSignal\(\)](#)

Enumerator:

MC_NO_SIGNAL
MC_RECV_CONNECTION
MC_RECV_MESSAGE
MC_RECV_AGENT
MC_RECV_RETURN
MC_EXEC_AGENT
MC_ALL_SIGNALS

Definition at line 111 of file MCEExports.cs.

7.48.3 Constructor & Destructor Documentation

7.48.3.1 LibMC::MCAgency::MCAgency () [inline]

Default constructor.

The default constructor for the [MCAgency](#) class. It creates a new agency, default options for the agency, and initializes the agency. It does not start the agency.

Definition at line 369 of file MCAgency.cs.

References `_MC_InitializeAgencyOptions()`, `options`, and `state`.

7.48.4 Member Function Documentation

7.48.4.1 int LibMC::MCAgency::Initialize () [inline]

Starts the agency.

Starts the agency and sets the agency state.

Returns:

0 on success, -1 on failure.

Note:

The agency port and any other options must be set before calling this function.

Definition at line 445 of file MCAgency.cs.

References `Agency`, `options`, `port`, and `state`.

7.48.4.2 int LibMC::MCAgency::End () [inline]

Stops and destroys the agency.

Stops the agency and sets the agency state appropriately.

Returns:

The return value of the underlying `MC_End` function.

Note:

This call will fail if the underlying Mobile-C agency is not in the correct state.

Definition at line 467 of file MCAgency.cs.

References `Agency`, and `state`.

7.48.4.3 int LibMC::MCAgency::ChInitializeOptions (ChShellType *shellType*, String *home*) [inline]

Initializes Ch options for the agency.

Can be used to set the home directory and shell mode for the Ch interpreter.

Parameters:

shellType The type of shell Ch should use: CH_REGULARCH or CH_SAFECH.

home The home directory Ch should use.

Returns:

The return value of the underlying MC_ChInitializeOptions function.

Note:

This function must be called before the agency is started.

Definition at line 485 of file MCAgency.cs.

References _MC_ChInitializeOptions(), Agency, LibMC::MCAgency::ChOptions_t::chhome, int, options, and LibMC::MCAgency::ChOptions_t::shelltype.

7.48.4.4 int LibMC::MCAgency::SetThreadsAllOn () [inline]

Definition at line 502 of file MCAgency.cs.

References _MC_SetThreadsAllOn(), and options.

7.48.4.5 int LibMC::MCAgency::SetThreadsAllOff () [inline]

Sets all threads for the agency to "off".

Sets all threads for the agency to "off." Not recommended for use.

Returns:

The return value of the underlying MC_SetThreadsAllOff function.

Note:

This function must be called before the agency is started.

Definition at line 516 of file MCAgency.cs.

References _MC_SetThreadsAllOff(), and options.

7.48.4.6 int LibMC::MCAgency::SetThreadOn (MC_ThreadIndex_e index) [inline]

Sets an individual thread for the agency to "on".

Threads are on by default. If they have been turned off, this function turns them on again.

Parameters:

index The enum that identifies the thread to be turned on.

Returns:

The return value of the underlying MC_SetThreadOn function.

Note:

This function must be called before the agency is started.

Definition at line 532 of file MCAgency.cs.

References `_MC_SetThreadOn()`, and options.

7.48.4.7 int LibMC::MCAgency::SetThreadOff (MC_ThreadIndex_e *index*) [inline]

Sets an individual thread for the agency to "off".

Most commonly used to turn the command prompt thread off.

Parameters:

index The enum that identifies the thread to be turned off.

Returns:

The return value of the underlying `MC_SetThreadOff` function.

Note:

This function must be called before the agency is started.

Definition at line 547 of file MCAgency.cs.

References `_MC_SetThreadOff()`, and options.

7.48.4.8 int LibMC::MCAgency::HaltAgency () [inline]

Temporarily halts the agency.

Halts the agency until it is resumed or ended.

Returns:

The return value of the underlying `MC_HaltAgency` function.

Note:

The underlying Mobile-C agency must be in the correct state to call this function or it will fail.

Definition at line 562 of file MCAgency.cs.

References `_MC_HaltAgency()`, `Agency`, and state.

7.48.4.9 int LibMC::MCAgency::ResumeAgency () [inline]

Resumes a halted agency.

Resumes a halted agency. Cannot be used on ended agencies.

Returns:

The return value of the underlying `MC_ResumeAgency` function.

Note:

The underlying Mobile-C agency must be in the correct state to call this function or it will fail.

Definition at line 579 of file MCAgency.cs.

References `_MC_ResumeAgency()`, `Agency`, and `state`.

7.48.4.10 `int LibMC::MCAgency::SetDefaultAgentStatus (MCAgent.MC_AgentStatus_e status)`
[inline]

Sets the default state of an agent in the agency.

Can be used to set the default status of agents, but most agents managed their state on their own.

Parameters:

status The enum that identifies the desired agent state.

Returns:

The return value of the underlying `MC_SetDefaultAgentStatus` function.

Definition at line 595 of file MCAgency.cs.

References `_MC_SetDefaultAgentStatus()`, and `Agency`.

7.48.4.11 `MCAgent LibMC::MCAgency::WaitRetrieveAgent ()` [inline]

Waits for an agent to arrive and returns the agent.

Waits for an agent to arrive in the agency, then returns that agent. The agent is not allowed to execute.

Returns:

The agent that was retrieved or an empty agent if it fails.

Definition at line 608 of file MCAgency.cs.

References `_MC_WaitRetrieveAgent()`, and `Agency`.

7.48.4.12 `int LibMC::MCAgency::WaitAgent ()` [inline]

Waits for an agent to arrive.

Waits for an agent to arrive in the agency. The agent is allowed to execute normally.

Returns:

The return value of the underlying `MC_WaitAgent` function.

Definition at line 625 of file MCAgency.cs.

References `_MC_WaitAgent()`, and `Agency`.

7.48.4.13 int LibMC::MCAgency::SendAgentMigrationMessageFile (String *filename*, String *hostname*, int *port*) [inline]

Sends an agent migration message file to an agency.

Sends the specified XML file to another agency (local or remote).

Parameters:

filename The name of the file to send (fully qualified).

hostname The URL, IP address, or other identifier for the agency host.

port The port to send to.

Returns:

The return value of the underlying MC_SendAgentMigrationMessageFile function.

Definition at line 644 of file MCAgency.cs.

References Agency.

7.48.4.14 int LibMC::MCAgency::LoadAgentMigrationMessageFile (String *filename*) [inline]

Load an agent migration message.

Loads the specified XML file to this agency automatically. There is no need to specify a port or agency location.

Parameters:

filename The name of the file to send (fully qualified).

Returns:

The return value of the underlying MC_SendAgentMigrationMessageFile function.

Definition at line 658 of file MCAgency.cs.

References Agency, and Port.

7.48.4.15 int LibMC::MCAgency::SendAgentMigrationMessage (String *message*, String *hostname*, int *port*) [inline]

Sends an agent migration message to an agency.

Sends an agent migration message to another agency (local or remote).

Parameters:

message The agent migration message.

hostname The URL, IP address, or other identifier for the agency host.

port The port to send to.

Returns:

The return value of the underlying MC_SendAgentMigrationMessageFile function.

Definition at line 673 of file MCAgency.cs.

References `_MC_SendAgentMigrationMessage()`, and `Agency`.

7.48.4.16 `int LibMC::MCAgency::CondBroadcast (int id)` `[inline]`

Broadcast a condition signal.

Broadcasts a signal in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#).

Parameters:

id The ID number of the condition to signal.

Returns:

The return value of the underlying `MC_CondBroadcast` function.

Definition at line 691 of file MCAgency.cs.

References `_MC_CondBroadcast()`, and `Agency`.

7.48.4.17 `int LibMC::MCAgency::CondSignal (int id)` `[inline]`

Signal a condition.

Signals a condition in the agency. The parameter "id" is the ID of the agency sync variable to signal that was created with [SyncInit\(\)](#).

Parameters:

id The ID number of the condition to signal.

Returns:

The return value of the underlying `MC_CondSignal` function.

Definition at line 705 of file MCAgency.cs.

References `_MC_CondSignal()`, and `Agency`.

7.48.4.18 `int LibMC::MCAgency::CondReset (int id)` `[inline]`

Reset a condition signal.

Resets a signal in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#). This function must be called after a condition is received in order to clear it.

Parameters:

id The ID number of the condition to reset.

Returns:

The return value of the underlying `MC_CondReset` function.

Definition at line 720 of file MCAgency.cs.

References `_MC_CondReset()`, and `Agency`.

7.48.4.19 int LibMC::MCAgency::CondWait (int *id*) [inline]

Wait for a condition signal.

Waits for a condition signal in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#). This function blocks until the signal is received.

Parameters:

id The ID number of the condition to wait for.

Returns:

The return value of the underlying MC_CondWait function.

Definition at line 735 of file MCAgency.cs.

References [_MC_CondWait\(\)](#), and [Agency](#).

7.48.4.20 int LibMC::MCAgency::MutexLock (int *id*) [inline]

Lock a mutex.

Locks a mutex in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#). This function blocks until the mutex is locked.

Parameters:

id The ID number of the mutex to lock.

Returns:

The return value of the underlying MC_MutexLock function.

Definition at line 750 of file MCAgency.cs.

References [_MC_MutexLock\(\)](#), and [Agency](#).

7.48.4.21 int LibMC::MCAgency::MutexUnlock (int *id*) [inline]

Unlock a mutex.

Locks a mutex in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#).

Parameters:

id The ID number of the mutex to unlock.

Returns:

The return value of the underlying MC_MutexUnlock function.

Definition at line 764 of file MCAgency.cs.

References [_MC_MutexUnlock\(\)](#), and [Agency](#).

7.48.4.22 `int LibMC::MCAgency::SemaphorePost (int id)` `[inline]`

Posts a semaphore.

Posts a semaphore in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#).

Parameters:

id The ID number of the semaphore to post.

Returns:

The return value of the underlying MC_SemaphorePost function.

Definition at line 778 of file MCAgency.cs.

References `_MC_SemaphorePost()`, and `Agency`.

7.48.4.23 `int LibMC::MCAgency::SemaphoreWait (int id)` `[inline]`

Wait for a semaphore to be posted.

Wait for a semaphore in the agency to be posted. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#). This function blocks until the semaphore is posted.

Parameters:

id The ID number of the semaphore to wait for.

Returns:

The return value of the underlying MC_SemaphoreWait function.

Definition at line 793 of file MCAgency.cs.

References `_MC_SemaphoreWait()`, and `Agency`.

7.48.4.24 `int LibMC::MCAgency::ResetSignal ()` `[inline]`

Reset an agency signal.

Resets a signal in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#).

Returns:

The return value of the underlying MC_ResetSignal function.

Definition at line 806 of file MCAgency.cs.

References `_MC_ResetSignal()`, and `Agency`.

7.48.4.25 `int LibMC::MCAgency::SyncDelete (int id)` `[inline]`

Delete a synchronization variable.

Deletes a synchronization variable in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#).

Parameters:

id The ID number of the variable to delete.

Returns:

The return value of the underlying MC_SyncDelete function.

Definition at line 820 of file MCAgency.cs.

References `_MC_SyncDelete()`, and `Agency`.

7.48.4.26 int LibMC::MCAgency::SyncInit (int *id*) [inline]

Create a new synchronization variable.

Creates a new synchronization variable in the agency. The parameter "id" is desired ID of the variable. A random ID is returned if "id" is already in use.

Parameters:

id The ID number of the condition to signal.

Returns:

The return value of the underlying MC_CondBroadcast function- either a random ID or the desired ID if the desired ID is already in use.

Definition at line 836 of file MCAgency.cs.

References `_MC_SyncInit()`, and `Agency`.

7.48.4.27 int LibMC::MCAgency::WaitSignal (int *signals*) [inline]

Wait for agency signals.

Waits for signals to occur in the agency.

Parameters:

signals The ID number of the condition to signal.

Returns:

The return value of the underlying MC_WaitSignal function.

Definition at line 849 of file MCAgency.cs.

References `_MC_WaitSignal()`, and `Agency`.

7.48.4.28 int LibMC::MCAgency::BarrierDelete (int *id*) [inline]

Delete a barrier object.

Deletes a barrier object from the agency. The parameter "id" is the ID of the agency sync variable created with [BarrierInit\(\)](#).

Parameters:

id The ID number of the barrier to delete.

Returns:

The return value of the underlying MC_BarrierDelete function.

Definition at line 863 of file MCAgency.cs.

References _MC_BarrierDelete(), and Agency.

7.48.4.29 int LibMC::MCAgency::BarrierInit (int *id*, int *num_procs*) [inline]

Create a new barrier.

Creates a new barrier object in the agency.

Parameters:

id The ID number of the condition to signal.

num_procs the number of process to block (?)

Returns:

The return value of the underlying MC_BarrierInit function.

Definition at line 877 of file MCAgency.cs.

References _MC_BarrierInit(), and Agency.

7.48.4.30 MC_SteerCommand_e LibMC::MCAgency::SteerControl () [inline]

Steering control function.

Really not sure.

Returns:

The return value of the underlying MC_SteerControl function.

Todo

Test MC_SteerControl, MC_Steer.

Definition at line 895 of file MCAgency.cs.

References _MC_SteerControl().

7.48.4.31 int LibMC::MCAgency::Steer (IntPtr *funcptr*, IntPtr *arg*) [inline]

Steering control function.

Really not sure.

Parameters:

funcptr Pointer to the steering function

arg Argument to function

Returns:

The return value of the underlying `_MC_Steer` function.

Note:

This function does nothing but throw an exception right now.

Bug

`MC_Steer` is not yet implemented.

Todo

Implement `MC_Steer`

Definition at line 913 of file `MCAgency.cs`.

7.48.4.32 `int LibMC::MCAgency::RegisterService (MCAgent agent, int agentID, String agentName, String[] serviceNames, int numServices)` [inline]

Registers services in the agency.

Registers services provided by agents with the agency. Not really useful in binary space.

Parameters:

agent The agent providing the services.

agentID The agent ID number.

agentName The agent name.

serviceNames An array of service names.

numServices The number of services provided.

Returns:

The return value of the underlying `MC_RegisterService` function.

Todo

Test `MC_RegisterService` and `MC_SearchForService`.

Definition at line 938 of file `MCAgency.cs`.

References `_MC_RegisterService()`, `Agency`, and `LibMC::MCAgent::Agent`.

7.48.4.33 `int LibMC::MCAgency::SearchForService (String searchString, IntPtr agentNames, IntPtr serviceNames, IntPtr agentIDs, IntPtr numResults)` [inline]

Searches for services in the agency.

Searches for services provided by agents with the agency. Not really useful in binary space.

Parameters:

searchString The agent providing the services.
agentNames The agent ID number.
serviceNames The agent name.
agentIDs An array of service names.
numResults The number of services provided.

Returns:

The return value of the underlying MC_SearchForService function.

Note:

This function does nothing but throw an exception right now.

Bug

MC_SearchForService is not yet implemented.

Todo

Implement SearchForService

Definition at line 961 of file MCAgency.cs.

7.48.4.34 int LibMC::MCAgency::AddAgent (MCAgent *agent*) [inline]

Add an agent to the agency.

Adds an agent to the agency.

Parameters:

agent The agent to add.

Returns:

The return value of the underlying MC_AddAgent function.

Definition at line 979 of file MCAgency.cs.

References _MC_AddAgent(), Agency, and LibMC::MCAgent::Agent.

7.48.4.35 MCAgent LibMC::MCAgency::FindAgentByName (String *name*) [inline]

Finds an agent by its name.

Finds an agent in the agency by its name.

Parameters:

name The name of the agent to search for.

Returns:

The return value of the underlying MC_FindAgentByName function.

Definition at line 992 of file MCAgency.cs.

References _MC_FindAgentByName(), and Agency.

7.48.4.36 MCAgent LibMC::MCAgency::FindAgentByID (int *id*) [inline]

Find an agent by its ID.

Finds an agent in the agency by its ID number.

Parameters:

id The ID number of the condition to signal.

Returns:

The return value of the underlying MC_FindAgentByID function.

Definition at line 1005 of file MCAgency.cs.

References _MC_FindAgentByID(), and Agency.

7.48.4.37 MCAgent LibMC::MCAgency::RetrieveAgent () [inline]

Retrieve an agent from the agency.

Really not sure.

Returns:

The return value of the underlying MC_CondBroadcast function.

Definition at line 1017 of file MCAgency.cs.

References _MC_RetrieveAgent(), and Agency.

7.48.4.38 int LibMC::MCAgency::AclSend (MCAclMessage *acl_message*) [inline]

Send an ACL message to the agency.

Sends an ACL message to the agency. The message is delivered appropriately.

Parameters:

acl_message The message to send.

Returns:

The return value of the underlying MC_CondBroadcast function.

Definition at line 1034 of file MCAgency.cs.

References LibMC::MCAclMessage::AclMsg, and Agency.

7.48.4.39 int LibMC::MCAgency::MainLoop () [inline]

Makes the agency wait indefinitely.

Makes the agency wait indefinitely until it receives a "quit" command or is otherwise terminated.

Returns:

The return value of the underlying MC_MainLoop function.

Definition at line 1047 of file MCAgency.cs.

References _MC_MainLoop(), and Agency.

7.48.4.40 static IntPtr LibMC::MCAgency::_MC_Initialize (int *port*, ref MCAgencyOptions_t *options*) [private]

7.48.4.41 static int LibMC::MCAgency::_MC_End (IntPtr *agency*) [private]

7.48.4.42 static int LibMC::MCAgency::_MC_ChInitializeOptions (IntPtr *agency*, ChOptions_t *options*) [private]

Referenced by ChInitializeOptions().

7.48.4.43 static int LibMC::MCAgency::_MC_InitializeAgencyOptions (ref MCAgencyOptions_t *options*) [private]

Referenced by MCAgency().

7.48.4.44 static int LibMC::MCAgency::_MC_SetThreadsAllOn (ref MCAgencyOptions_t *options*) [private]

Referenced by SetThreadsAllOn().

7.48.4.45 static int LibMC::MCAgency::_MC_SetThreadsAllOff (ref MCAgencyOptions_t *options*) [private]

Referenced by SetThreadsAllOff().

7.48.4.46 static int LibMC::MCAgency::_MC_SetThreadOn (ref MCAgencyOptions_t *options*, MC_ThreadIndex_e *index*) [private]

Referenced by SetThreadOn().

7.48.4.47 static int LibMC::MCAgency::_MC_SetThreadOff (ref MCAgencyOptions_t *options*, MC_ThreadIndex_e *index*) [private]

Referenced by SetThreadOff().

7.48.4.48 static int LibMC::MCAgency::_MC_HaltAgency (IntPtr *agency*) [private]

Referenced by HaltAgency().

7.48.4.49 static int LibMC::MCAgency::_MC_ResumeAgency (IntPtr *agency*) [private]

Referenced by ResumeAgency().

7.48.4.50 static int LibMC::MCAgency::_MC_SetDefaultAgentStatus (IntPtr *agency*, MCAgent.MC_AgentStatus_e *status*) [private]

Referenced by SetDefaultAgentStatus().

7.48.4.51 static IntPtr LibMC::MCAgency::_MC_WaitRetrieveAgent (IntPtr *agency*) [private]

Referenced by WaitRetrieveAgent().

7.48.4.52 static int LibMC::MCAgency::_MC_WaitAgent (IntPtr *agency*) [private]

Referenced by WaitAgent().

7.48.4.53 static int LibMC::MCAgency::_MC_MainLoop (IntPtr *agency*) [private]

Referenced by MainLoop().

7.48.4.54 static int LibMC::MCAgency::_MC_SendAgentMigrationMessageFile (IntPtr *agency*, String *filename*, String *hostname*, int *port*) [private]

7.48.4.55 static int LibMC::MCAgency::_MC_SendAgentMigrationMessage (IntPtr *agency*, String *message*, String *hostname*, int *port*) [private]

Referenced by SendAgentMigrationMessage().

7.48.4.56 static int LibMC::MCAgency::_MC_CondBroadcast (IntPtr *agency*, int *id*) [private]

Referenced by CondBroadcast().

7.48.4.57 static int LibMC::MCAgency::_MC_CondSignal (IntPtr *agency*, int *id*) [private]

Referenced by CondSignal().

7.48.4.58 static int LibMC::MCAgency::_MC_CondReset (IntPtr *agency*, int *id*) [private]

Referenced by CondReset().

7.48.4.59 static int LibMC::MCAgency::_MC_CondWait (IntPtr *agency*, int *id*) [private]

Referenced by CondWait().

7.48.4.60 static int LibMC::MCAgency::_MC_MutexLock (IntPtr *agency*, int *id*) [private]

Referenced by MutexLock().

7.48.4.61 `static int LibMC::MCAgency::_MC_MutexUnlock (IntPtr agency, int id)` [private]

Referenced by MutexUnlock().

7.48.4.62 `static int LibMC::MCAgency::_MC_SemaphorePost (IntPtr agency, int id)`
[private]

Referenced by SemaphorePost().

7.48.4.63 `static int LibMC::MCAgency::_MC_SemaphoreWait (IntPtr agency, int id)`
[private]

Referenced by SemaphoreWait().

7.48.4.64 `static int LibMC::MCAgency::_MC_ResetSignal (IntPtr agency)` [private]

Referenced by ResetSignal().

7.48.4.65 `static int LibMC::MCAgency::_MC_SyncDelete (IntPtr agency, int id)` [private]

Referenced by SyncDelete().

7.48.4.66 `static int LibMC::MCAgency::_MC_SyncInit (IntPtr agency, int id)` [private]

Referenced by SyncInit().

7.48.4.67 `static int LibMC::MCAgency::_MC_WaitSignal (IntPtr agency, int signals)`
[private]

Referenced by WaitSignal().

7.48.4.68 `static int LibMC::MCAgency::_MC_BarrierDelete (IntPtr agency, int id)` [private]

Referenced by BarrierDelete().

7.48.4.69 `static int LibMC::MCAgency::_MC_BarrierInit (IntPtr agency, int id, int num_procs)`
[private]

Referenced by BarrierInit().

7.48.4.70 `static MC_SteerCommand_e LibMC::MCAgency::_MC_SteerControl ()` [private]

Referenced by SteerControl().

7.48.4.71 `static int LibMC::MCAgency::_MC_Steer (IntPtr agency, IntPtr funcptr, IntPtr arg)`
[private]

7.48.4.72 `static int LibMC::MCAgency::_MC_RegisterService (IntPtr agency, IntPtr agent, int agentID, String agentName, String[] serviceNames, int numServices)` [private]

Referenced by RegisterService().

7.48.4.73 `static int LibMC::MCAgency::_MC_SearchForService (IntPtr agency, String searchString, IntPtr agentNames, IntPtr serviceNames, IntPtr agentIDs, IntPtr numResults)` [private]

7.48.4.74 `static int LibMC::MCAgency::_MC_AddAgent (IntPtr agency, IntPtr agent)`
[private]

Referenced by AddAgent().

7.48.4.75 `static internal int LibMC::MCAgency::_MC_DeleteAgent (IntPtr agent)` [private]

7.48.4.76 `static IntPtr LibMC::MCAgency::_MC_FindAgentByName (IntPtr agency, String name)` [private]

Referenced by FindAgentByName().

7.48.4.77 `static IntPtr LibMC::MCAgency::_MC_FindAgentByID (IntPtr agency, int ID)`
[private]

Referenced by FindAgentByID().

7.48.4.78 `static IntPtr LibMC::MCAgency::_MC_RetrieveAgent (IntPtr agency)` [private]

Referenced by RetrieveAgent().

-
- 7.48.4.79** static internal int LibMC::MCAgency::_MC_GetAgentID (IntPtr *agent*) [private]
- 7.48.4.80** static internal String LibMC::MCAgency::_MC_GetAgentName (IntPtr *agent*) [private]
- 7.48.4.81** static internal int LibMC::MCAgency::_MC_GetAgentNumTasks (IntPtr *agent*) [private]
- 7.48.4.82** static internal MCAgent.MC_AgentStatus_e LibMC::MCAgency::_MC_GetAgentStatus (IntPtr *agent*) [private]
- 7.48.4.83** static internal MCAgent.MC_AgentType_e LibMC::MCAgency::_MC_GetAgentType (IntPtr *agent*) [private]
- 7.48.4.84** static internal String LibMC::MCAgency::_MC_GetAgentXMLString (IntPtr *agent*) [private]
- 7.48.4.85** static internal int LibMC::MCAgency::_MC_PrintAgentCode (IntPtr *agent*) [private]
- 7.48.4.86** static internal String LibMC::MCAgency::_MC_RetrieveAgentCode (IntPtr *agent*) [private]
- 7.48.4.87** static internal int LibMC::MCAgency::_MC_SetAgentStatus (IntPtr *agent*, MCAgent.MC_AgentStatus_e *status*) [private]
- 7.48.4.88** static internal int LibMC::MCAgency::_MC_TerminateAgent (IntPtr *agent*) [private]
- 7.48.4.89** static internal int LibMC::MCAgency::_MC_CallAgentFunc (IntPtr *agent*, String *funcName*, IntPtr *returnVal*, IntPtr *varg*) [private]
- 7.48.4.90** static internal IntPtr LibMC::MCAgency::_MC_GetAgentExecEngine (IntPtr *agent*) [private]
- 7.48.4.91** static internal int LibMC::MCAgency::_MC_GetAgentReturnData (IntPtr *agent*, int *task_num*, IntPtr *data*, IntPtr *dim*, IntPtr *extent*) [private]
- 7.48.4.92** static internal IntPtr LibMC::MCAgency::_MC_AclNew () [private]
- 7.48.4.93** static internal int LibMC::MCAgency::_MC_AclPost (IntPtr *agent*, IntPtr *message*) [private]
- 7.48.4.94** static internal IntPtr LibMC::MCAgency::_MC_AclReply (IntPtr *acl_message*) [private]
- 7.48.4.95** static internal IntPtr LibMC::MCAgency::_MC_AclRetrieve (IntPtr *agent*) [private]
- 7.48.4.96** static internal int LibMC::MCAgency::_MC_AclSend (IntPtr *agency*, IntPtr *acl_message*) [private]
- 7.48.4.97** static internal IntPtr LibMC::MCAgency::_MC_AclWaitRetrieve (IntPtr *agent*) [private]
-
- 7.48.4.98** static internal int LibMC::MCAgency::_MC_AclSetPerformative (IntPtr *acl*, MCAclMessage.MC_FipaPerformative_e *performative*) [private]
- 7.48.4.99** static internal int LibMC::MCAgency::_MC_AclSetSender (IntPtr *acl*, String *name*, String *address*) [private]

7.48.5.2 MCAgencyOptions_t LibMC::MCAgency::options [private]

Definition at line 358 of file MCAgency.cs.

Referenced by ChInitializeOptions(), Initialize(), MCAgency(), SetThreadOff(), SetThreadOn(), SetThreadsAllOff(), and SetThreadsAllOn().

7.48.5.3 int LibMC::MCAgency::port = -1 [private]

Definition at line 359 of file MCAgency.cs.

Referenced by Initialize().

7.48.5.4 MCAgencyState LibMC::MCAgency::state = MCAgencyState.NoState [private]

Definition at line 360 of file MCAgency.cs.

Referenced by End(), HaltAgency(), Initialize(), MCAgency(), and ResumeAgency().

7.48.5.5 const String LibMC::MCAgency::mcdll [private]

Initial value:

```
"libmc.dll"
```

Definition at line 15 of file MCExports.cs.

7.48.6 Property Documentation**7.48.6.1 IntPtr LibMC::MCAgency::Agency** [get, set, private]

Definition at line 377 of file MCAgency.cs.

Referenced by AclSend(), AddAgent(), BarrierDelete(), BarrierInit(), ChInitializeOptions(), CondBroadcast(), CondReset(), CondSignal(), CondWait(), End(), FindAgentByID(), FindAgentByName(), HaltAgency(), Initialize(), LoadAgentMigrationMessageFile(), MainLoop(), MutexLock(), MutexUnlock(), RegisterService(), ResetSignal(), ResumeAgency(), RetrieveAgent(), SemaphorePost(), SemaphoreWait(), SendAgentMigrationMessage(), SendAgentMigrationMessageFile(), SetDefaultAgentStatus(), SyncDelete(), SyncInit(), WaitAgent(), WaitRetrieveAgent(), and WaitSignal().

7.48.6.2 int LibMC::MCAgency::Port [get, set]

Accessor for the port number of the agency.

Allows the user to set the agency port or get the port number while it is running.

Note:

The port must be set before the agency is started. Once the agency is started, the port cannot be changed.

Definition at line 404 of file MCAgency.cs.

Referenced by LoadAgentMigrationMessageFile().

7.48.6.3 MCAgencyState LibMC::MCAgency::State [get]

Accessor for the agency state.

Allows the user to query the state of the agency.

Note:

The state cannot be set by the user. It is controlled internally.

Definition at line 424 of file MCAgency.cs.

The documentation for this class was generated from the following files:

- [/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgency.cs](#)
- [/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCExports.cs](#)

7.49 LibMC::MCAgency::ChOptions_t Struct Reference

ChOptions structures.

Data Fields

- [int shelltype](#)
- [String chhome](#)

7.49.1 Detailed Description

ChOptions structures.

Allows the user to set the shell type and the home directory.

Note:

This struct is pulled directly from the Mobile-C library.

Definition at line 61 of file MCEExports.cs.

7.49.2 Field Documentation

7.49.2.1 int LibMC::MCAgency::ChOptions_t::shelltype

shell type: CH_REGULARCH or CH_SAFECH

Definition at line 63 of file MCEExports.cs.

Referenced by LibMC::MCAgency::ChInitializeOptions().

7.49.2.2 String LibMC::MCAgency::ChOptions_t::chhome

Embedded Ch home directory. if NULL, use Ch home directory for standard/professional edition as Embedded Ch home directory

Definition at line 66 of file MCEExports.cs.

Referenced by LibMC::MCAgency::ChInitializeOptions().

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

- [/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCEExports.cs](#)

7.50 LibMC::MCAgency::MCAgency_t Struct Reference

Data Fields

- [int client](#)
- [int server](#)
- [String hostName](#)
- [String filename](#)
- [int portno](#)
- [int portnoc](#)
- [IntPtr mc_platform](#)
- [int default_agentstatus](#)
- [int threads](#)
- [int enable_security](#)
- [int\[\] stack_size](#)
- [int last_error](#)

7.50.1 Detailed Description

Definition at line 23 of file MCEExports.cs.

7.50.2 Field Documentation

7.50.2.1 `int LibMC::MCAgency::MCAgency_t::client`

Definition at line 25 of file MCEExports.cs.

7.50.2.2 `int LibMC::MCAgency::MCAgency_t::server`

Definition at line 26 of file MCEExports.cs.

7.50.2.3 `String LibMC::MCAgency::MCAgency_t::hostName`

Local Hostname

Definition at line 27 of file MCEExports.cs.

7.50.2.4 `String LibMC::MCAgency::MCAgency_t::filename`

Definition at line 28 of file MCEExports.cs.

7.50.2.5 `int LibMC::MCAgency::MCAgency_t::portno`

Local port number

Definition at line 29 of file MCEExports.cs.

7.50.2.6 int LibMC::MCAgency::MCAgency_t::portnoc

Definition at line 30 of file MCEExports.cs.

7.50.2.7 IntPtr LibMC::MCAgency::MCAgency_t::mc_platform

Local MobileC Platform

Definition at line 31 of file MCEExports.cs.

7.50.2.8 int LibMC::MCAgency::MCAgency_t::default_agentstatus

Agency default agent status

Definition at line 32 of file MCEExports.cs.

7.50.2.9 int LibMC::MCAgency::MCAgency_t::threads

flag which determines which threads to start

Definition at line 33 of file MCEExports.cs.

7.50.2.10 int LibMC::MCAgency::MCAgency_t::enable_security

Security flag

Definition at line 34 of file MCEExports.cs.

7.50.2.11 int [] LibMC::MCAgency::MCAgency_t::stack_size

Definition at line 37 of file MCEExports.cs.

7.50.2.12 int LibMC::MCAgency::MCAgency_t::last_error

Definition at line 38 of file MCEExports.cs.

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

- [/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCEExports.cs](#)

7.51 LibMC::MCAgency::MCAgencyOptions_t Struct Reference

Data Fields

- [int threads](#)
- [int default_agent_status](#)
- [int modified](#)
- [int enable_security](#)
- [int\[\] stack_size](#)

7.51.1 Detailed Description

Definition at line 42 of file MCExports.cs.

7.51.2 Field Documentation

7.51.2.1 int LibMC::MCAgency::MCAgencyOptions_t::threads

Threads to start

Definition at line 44 of file MCExports.cs.

7.51.2.2 int LibMC::MCAgency::MCAgencyOptions_t::default_agent_status

Default agent status

Definition at line 45 of file MCExports.cs.

7.51.2.3 int LibMC::MCAgency::MCAgencyOptions_t::modified

unused

Definition at line 46 of file MCExports.cs.

7.51.2.4 int LibMC::MCAgency::MCAgencyOptions_t::enable_security

security enable flag

Definition at line 47 of file MCExports.cs.

7.51.2.5 int [] LibMC::MCAgency::MCAgencyOptions_t::stack_size

If the stack size is -1, use system def.

Definition at line 50 of file MCExports.cs.

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

- [/home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCExports.cs](#)

7.52 MCAgencyOptions_s Struct Reference

User modifiable agency options.

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

Data Fields

- [int threads](#)
- [int default_agent_status](#)
- [int modified](#)
- [int enable_security](#)
- [int stack_size](#) [MC_THREAD_ALL]

7.52.1 Detailed Description

User modifiable agency options.

Definition at line 232 of file libmc.h.

7.52.2 Field Documentation

7.52.2.1 int MCAgencyOptions_s::threads

Threads to start

Definition at line 233 of file libmc.h.

Referenced by MC_Initialize(), MC_InitializeAgencyOptions(), MC_SetThreadOff(), MC_SetThreadOn(), MC_SetThreadsAllOff(), and MC_SetThreadsAllOn().

7.52.2.2 int MCAgencyOptions_s::default_agent_status

Default agent status

Definition at line 234 of file libmc.h.

Referenced by MC_Initialize(), and MC_InitializeAgencyOptions().

7.52.2.3 int MCAgencyOptions_s::modified

unused

Definition at line 235 of file libmc.h.

Referenced by MC_InitializeAgencyOptions().

7.52.2.4 int MCAgencyOptions_s::enable_security

security enable flag

Definition at line 236 of file libmc.h.

Referenced by MC_Initialize(), and MC_InitializeAgencyOptions().

7.52.2.5 int MCAgencyOptions_s::stack_size[MC_THREAD_ALL]

If the stack size is -1, use system def.

Definition at line 239 of file libmc.h.

Referenced by MC_Initialize(), and MC_InitializeAgencyOptions().

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

- </home/dko/projects/mobilec/trunk/src/include/libmc.h>

7.53 LibMC::MCAgent Class Reference

Wrapper class for MCAgent_t structure.

Public Types

- enum [MC_AgentType_e](#)
Enum for describing the type of an agent.
- enum [MC_AgentStatus_e](#)
Enum for describing the status of an agent.

Public Member Functions

- [MCAgent](#) ()
Default constructor.
- override string [ToString](#) ()
Display the agent's fields.
- int [DeleteAgent](#) ()
Deletes an agent.
- String [GetAgentXMLString](#) ()
Gets the agent's XML string.
- int [PrintAgentCode](#) ()
Gets the agent's C code string.
- String [RetrieveAgentCode](#) ()
Gets the agent's C code string.
- int [TerminateAgent](#) ()
Terminates an agent.
- int [AclPost](#) (MCAclMessage message)
Posts an ACL message to the agent.
- MCAclMessage [AclRetrieve](#) ()
Retrieve an ACL message from the agent.
- MCAclMessage [AclWaitRetrieve](#) ()
Wait for and retrieve an ACL message from the agent.
- int [CallAgentFunc](#) (String funcName, IntPtr retval, IntPtr varg)
Calls a function in an agent script.
- IntPtr [GetAgentExecEngine](#) ()

Gets an agent's Ch interpreter.

- `int GetAgentReturnData (int task_num, IntPtr data, IntPtr dim, IntPtr extent)`

Calls a function in an agent script.

Static Public Member Functions

- static implicit `operator IntPtr (MCAgent agent)`
- static implicit `operator MCAgent (IntPtr ip)`

Properties

- internal `IntPtr Agent` [get, set]
- `int AgentID` [get]
Gets the agent's ID number.
- `String AgentName` [get]
Gets the agent's name.
- `int AgentNumTasks` [get]
Gets the agent's number of tasks.
- `MC_AgentStatus_e AgentStatus` [get, set]
Gets or sets the agent's status.
- `MC_AgentType_e AgentType` [get]
Gets the agent's type.

Private Member Functions

- internal `MCAgent (IntPtr ip)`
- `void GetAgentFields ()`

Private Attributes

- `IntPtr agent_p`
- `String name = ""`
- `int id = -1`
- `int numTasks = -1`
- `MC_AgentStatus_e status = MC_AgentStatus_e.MC_NO_STATUS`
- `MC_AgentType_e type = MC_AgentType_e.MC_NONE`

7.53.1 Detailed Description

Wrapper class for MCAgent_t structure.

This class provides an interface to the Mobile-C agent structure. Member functions for the class are generally overloaded versions of the respective functions in the Mobile-C library. The class maintains a pointer to a Mobile-C agent in unmanaged memory. The pointer is not accessible by the user.

Definition at line 60 of file MCAgent.cs.

7.53.2 Member Enumeration Documentation

7.53.2.1 enum LibMC::MCAgent::MC_AgentType_e

Enum for describing the type of an agent.

Note:

This enum is pulled directly from the Mobile-C library.

Enumerator:

MC_NONE Default value to describe uninitialized agent.

MC_REMOTE_AGENT A remote agent.

MC_LOCAL_AGENT A local agent.

MC_RETURN_AGENT A returning agent.

Definition at line 74 of file MCAgent.cs.

7.53.2.2 enum LibMC::MCAgent::MC_AgentStatus_e

Enum for describing the status of an agent.

Note:

This enum is pulled directly from the Mobile-C library.

Enumerator:

MC_NO_STATUS Default value for uninitialized agent

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 87 of file MCAgent.cs.

7.53.3 Constructor & Destructor Documentation

7.53.3.1 LibMC::MCAgent::MCAgent () [inline]

Default constructor.

Creates an empty agent.

Definition at line 103 of file MCAgent.cs.

Referenced by operator MCAgent().

7.53.3.2 internal LibMC::MCAgent::MCAgent (IntPtr ip) [inline, private]

Definition at line 109 of file MCAgent.cs.

References Agent.

7.53.4 Member Function Documentation

7.53.4.1 override string LibMC::MCAgent::ToString () [inline]

Display the agent's fields.

Formats and returns a string with all of the agent's properties.

Returns:

A string containing a formatted representation of the agent's properties.

Note:

The agency port and any other options must be set before calling this function.

Definition at line 125 of file MCAgent.cs.

References AgentID, AgentName, AgentNumTasks, AgentStatus, and AgentType.

7.53.4.2 void LibMC::MCAgent::GetAgentFields () [inline, private]

Definition at line 162 of file MCAgent.cs.

References Agent, name, numTasks, status, and type.

7.53.4.3 static implicit LibMC::MCAgent::operator IntPtr (MCAgent agent) [inline, static]

Definition at line 277 of file MCAgent.cs.

References Agent.

7.53.4.4 static implicit LibMC::MCAgent::operator MCAgent (IntPtr ip) [inline, static]

Definition at line 282 of file MCAgent.cs.

References MCAgent().

7.53.4.5 int LibMC::MCAgent::DeleteAgent () [inline]

Deletes an agent.

Deletes an agent from the agency.

Returns:

The return value of the underlying MC_DeleteAgent function call.

Definition at line 299 of file MCAgent.cs.

References Agent.

7.53.4.6 String LibMC::MCAgent::GetAgentXMLString () [inline]

Gets the agent's XML string.

Returns the full XML string associated with the agent.

Returns:

The return value of the underlying MC_GetAgentXMLString function call.

Definition at line 312 of file MCAgent.cs.

References Agent.

7.53.4.7 int LibMC::MCAgent::PrintAgentCode () [inline]

Gets the agent's C code string.

Prints the C code associated with the agent to stdout.

Returns:

The return value of the underlying MC_PrintAgentCode function call.

Definition at line 325 of file MCAgent.cs.

References Agent.

7.53.4.8 String LibMC::MCAgent::RetrieveAgentCode () [inline]

Gets the agent's C code string.

Returns the C code associated with the agent.

Returns:

A string containing the agent's C code.

Definition at line 337 of file MCAgent.cs.

References Agent.

7.53.4.9 int LibMC::MCAgent::TerminateAgent () [inline]

Terminates an agent.

Terminates an agent regardless of the agent's state.

Returns:

The return value of the underlying MC_TerminateAgent function call.

Definition at line 350 of file MCAgent.cs.

References Agent.

7.53.4.10 int LibMC::MCAgent::AclPost (MCAclMessage *message*) [inline]

Posts an ACL message to the agent.

Delivers an ACL message to the agent.

Parameters:

message The ACL message object to deliver.

Returns:

The return value of the underlying MC_AclPost function call.

Note:

The message must be a valid message or this function call will fail.

Definition at line 375 of file MCAgent.cs.

References LibMC::MCAclMessage::AclMsg, and Agent.

7.53.4.11 MCAclMessage LibMC::MCAgent::AclRetrieve () [inline]

Retrieve an ACL message from the agent.

Retrieves an ACL message from the agent if one is available.

Returns:

The ACL message or a blank ACL message if one was not available.

Note:

The message must be a valid message or this function call will fail.

Definition at line 391 of file MCAgent.cs.

References Agent.

7.53.4.12 `MCACLMessage LibMC::MCAgent::AclWaitRetrieve ()` `[inline]`

Wait for and retrieve an ACL message from the agent.

Retrieves an ACL message from the agent when one becomes available.

Returns:

The ACL message or a blank ACL message if the call fails.

Note:

This function call blocks.

Definition at line 410 of file MCAgent.cs.

References Agent.

7.53.4.13 `int LibMC::MCAgent::CallAgentFunc (String funcName, IntPtr retval, IntPtr varg)`
`[inline]`

Calls a function in an agent script.

Calls a function in an agent's script file. This function requires manual marshaling by the user.

Parameters:

funcName The name of the function to call

retval A pointer to memory for the return value

varg A pointer to the argument for the function

Returns:

The return value of the underlying MC_CallAgentFunc function call.

Note:

BE VERY CAREFUL! You must marshal your arguments!

Todo

Find a better way to handle retval and varg

Definition at line 434 of file MCAgent.cs.

References Agent.

7.53.4.14 `IntPtr LibMC::MCAgent::GetAgentExecEngine ()` `[inline]`

Gets an agent's Ch interpreter.

Gets a pointer to the agent's Ch interpreter. Will be improved shortly.

Returns:

A pointer to the Ch interpreter.

Note:

Nothing in the LibMC.NET library can make use of the Ch interpreter yet.

Todo

Wrap MC_GetAgentExecEngine with an object for the void* pointer return type (Ch interpreter).

Definition at line 451 of file MCAgent.cs.

References Agent.

7.53.4.15 int LibMC::MCAgent::GetAgentReturnData (int *task_num*, IntPtr *data*, IntPtr *dim*, IntPtr *extent*) [inline]

Calls a function in an agent script.

Calls a function in an agent's script file. This function requires manual marshaling by the user.

Parameters:

task_num Task number to get data from

data A pointer to memory for the data

dim A pointer to hold the dimensions of the data

extent A pointer to hold the dimensions of the data

Returns:

The return value of the underlying MC_GetAgentReturnData function call.

Note:

This function does nothing but throw an exception right now.

Todo

Implement GetAgentReturnData

Definition at line 472 of file MCAgent.cs.

7.53.5 Field Documentation**7.53.5.1 IntPtr LibMC::MCAgent::agent_p [private]**

Definition at line 62 of file MCAgent.cs.

7.53.5.2 String LibMC::MCAgent::name = "" [private]

Definition at line 63 of file MCAgent.cs.

Referenced by GetAgentFields().

7.53.5.3 int LibMC::MCAgent::id = -1 [private]

Definition at line 64 of file MCAgent.cs.

7.53.5.4 `int LibMC::MCAgent::numTasks = -1` [private]

Definition at line 65 of file MCAgent.cs.

Referenced by GetAgentFields().

7.53.5.5 `MC_AgentStatus_e LibMC::MCAgent::status = MC_AgentStatus_e.MC_NO_STATUS`
[private]

Definition at line 66 of file MCAgent.cs.

Referenced by GetAgentFields().

7.53.5.6 `MC_AgentType_e LibMC::MCAgent::type = MC_AgentType_e.MC_NONE`
[private]

Definition at line 67 of file MCAgent.cs.

Referenced by GetAgentFields().

7.53.6 Property Documentation**7.53.6.1** `internal IntPtr LibMC::MCAgent::Agent` [get, set, private]

Definition at line 142 of file MCAgent.cs.

Referenced by AclPost(), AclRetrieve(), AclWaitRetrieve(), LibMC::MCAgency::AddAgent(), CallAgent-
Func(), DeleteAgent(), GetAgentExecEngine(), GetAgentFields(), GetAgentXMLString(), MCAgent(),
operator IntPtr(), PrintAgentCode(), LibMC::MCAgency::RegisterService(), RetrieveAgentCode(), and
TerminateAgent().

7.53.6.2 `int LibMC::MCAgent::AgentID` [get]

Gets the agent's ID number.

Gets the agent's ID number as assigned by Mobile-C if the agent is a valid agent.

Returns:

The agent's ID number or -1 for an empty agent.

Definition at line 180 of file MCAgent.cs.

Referenced by ToString().

7.53.6.3 `String LibMC::MCAgent::AgentName` [get]

Gets the agent's name.

Gets the agent's name as assigned by Mobile-C or the agent script if the agent is a valid agent.

Returns:

The agent's name or an empty string for an empty agent.

Definition at line 200 of file MCAgent.cs.

Referenced by ToString().

7.53.6.4 int LibMC::MCAgent::AgentNumTasks [get]

Gets the agent's number of tasks.

Gets the agent's ID number of tasks if the agent is a valid agent.

Returns:

The agent's ID number of tasks or -1 for an empty agent.

Definition at line 219 of file MCAgent.cs.

Referenced by ToString().

7.53.6.5 MC_AgentStatus_e LibMC::MCAgent::AgentStatus [get, set]

Gets or sets the agent's status.

Gets or sets the agent's status. When setting the status, the status is double-checked after setting it and may not be set depending on the state of the agent and the agency.

Returns:

The agent's status or MC_NO_STATUS for an empty agent.

Definition at line 241 of file MCAgent.cs.

Referenced by ToString().

7.53.6.6 MC_AgentType_e LibMC::MCAgent::AgentType [get]

Gets the agent's type.

Gets the agent's type.

Returns:

The agent's type or MC_NONE for an empty agent.

Definition at line 263 of file MCAgent.cs.

Referenced by ToString().

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

- </home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgent.cs>

7.54 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](#)

7.54.1 Detailed Description

MD2 context structure.

Definition at line 14 of file md2.h.

7.54.2 Field Documentation

7.54.2.1 unsigned char md2_context::cksum[16]

checksum of the data block

Definition at line 16 of file md2.h.

Referenced by [md2_finish\(\)](#), and [md2_process\(\)](#).

7.54.2.2 unsigned char md2_context::state[48]

intermediate digest state

Definition at line 17 of file md2.h.

Referenced by [md2_finish\(\)](#), and [md2_process\(\)](#).

7.54.2.3 unsigned char md2_context::buffer[16]

data block being processed

Definition at line 18 of file md2.h.

Referenced by [md2_finish\(\)](#), [md2_process\(\)](#), and [md2_update\(\)](#).

7.54.2.4 unsigned char md2_context::ipad[64]

HMAC: inner padding

Definition at line 19 of file md2.h.

Referenced by [md2_hmac_starts\(\)](#).

7.54.2.5 unsigned char md2_context::opad[64]

HMAC: outer padding

Definition at line 20 of file md2.h.

Referenced by md2_hmac_finish(), and md2_hmac_starts().

7.54.2.6 int md2_context::left

amount of data in buffer

Definition at line 21 of file md2.h.

Referenced by md2_finish(), and md2_update().

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

- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/[md2.h](#)

7.55 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]

7.55.1 Detailed Description

MD4 context structure.

Definition at line 14 of file md4.h.

7.55.2 Field Documentation

7.55.2.1 unsigned long md4_context::total[2]

number of bytes processed

Definition at line 16 of file md4.h.

Referenced by md4_finish(), md4_starts(), and md4_update().

7.55.2.2 unsigned long md4_context::state[4]

intermediate digest state

Definition at line 17 of file md4.h.

Referenced by md4_finish(), md4_process(), and md4_starts().

7.55.2.3 unsigned char md4_context::buffer[64]

data block being processed

Definition at line 18 of file md4.h.

Referenced by md4_update().

7.55.2.4 unsigned char md4_context::ipad[64]

HMAC: inner padding

Definition at line 19 of file md4.h.

Referenced by md4_hmac_starts().

7.55.2.5 unsigned char md4_context::opad[64]

HMAC: outer padding

Definition at line 20 of file md4.h.

Referenced by md4_hmac_finish(), and md4_hmac_starts().

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

- [/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/md4.h](#)

7.56 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]

7.56.1 Detailed Description

MD5 context structure.

Definition at line 14 of file md5.h.

7.56.2 Field Documentation

7.56.2.1 unsigned long md5_context::total[2]

number of bytes processed

Definition at line 16 of file md5.h.

Referenced by md5_finish(), md5_starts(), and md5_update().

7.56.2.2 unsigned long md5_context::state[4]

intermediate digest state

Definition at line 17 of file md5.h.

Referenced by md5_finish(), md5_process(), and md5_starts().

7.56.2.3 unsigned char md5_context::buffer[64]

data block being processed

Definition at line 18 of file md5.h.

Referenced by md5_update().

7.56.2.4 unsigned char md5_context::ipad[64]

HMAC: inner padding

Definition at line 19 of file md5.h.

Referenced by md5_hmac_starts().

7.56.2.5 unsigned char md5_context::opad[64]

HMAC: outer padding

Definition at line 20 of file md5.h.

Referenced by md5_hmac_finish(), and md5_hmac_starts().

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

- </home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/md5.h>

7.57 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](#)

7.57.1 Detailed Description

Definition at line 69 of file message.h.

7.57.2 Field Documentation

7.57.2.1 struct sockaddr_in* message_s::addr [read]

Definition at line 71 of file message.h.

Referenced by [message_Destroy\(\)](#), [message_InitializeFromAgent\(\)](#), [message_InitializeFromConnection\(\)](#), [message_InitializeFromString\(\)](#), and [message_New\(\)](#).

7.57.2.2 int message_s::connect_id

Definition at line 74 of file message.h.

Referenced by [message_InitializeFromConnection\(\)](#), [message_InitializeFromString\(\)](#), and [message_New\(\)](#).

7.57.2.3 int message_s::message_id

Definition at line 75 of file message.h.

Referenced by [AP_QUEUE_SEARCH_TEMPLATE\(\)](#), [message_InitializeFromAgent\(\)](#), [message_InitializeFromConnection\(\)](#), [message_InitializeFromString\(\)](#), and [message_New\(\)](#).

7.57.2.4 int message_s::isHTTP

Definition at line 78 of file message.h.

Referenced by message_New(), mtp_http_ComposeMessage(), and mtp_http_CreateMessage().

7.57.2.5 enum message_type_e message_s::message_type

Definition at line 81 of file message.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), agent_Initialize(), MC_AclSend(), message_InitializeFromAgent(), message_InitializeFromString(), and message_New().

7.57.2.6 enum http_performative_e message_s::http_type

Definition at line 82 of file message.h.

Referenced by message_New().

7.57.2.7 mxml_node_t* message_s::xml_root

Definition at line 84 of file message.h.

Referenced by acc_Thread(), agent_Initialize(), MC_LoadAgentFromFile(), message_Destroy(), message_InitializeFromAgent(), message_InitializeFromConnection(), message_InitializeFromString(), and message_New().

7.57.2.8 mxml_node_t* message_s::xml_payload

Definition at line 85 of file message.h.

Referenced by agent_Initialize(), MC_LoadAgentFromFile(), and message_New().

7.57.2.9 char* message_s::message_body

Definition at line 88 of file message.h.

Referenced by acc_Thread(), message_Destroy(), message_InitializeFromAgent(), message_InitializeFromConnection(), message_InitializeFromString(), message_New(), message_Send(), mtp_http_ComposeMessage(), and mtp_http_CreateMessage().

7.57.2.10 char* message_s::update_name

Definition at line 90 of file message.h.

Referenced by message_Destroy(), message_InitializeFromAgent(), message_InitializeFromString(), and message_New().

7.57.2.11 int message_s::update_num

Definition at line 91 of file message.h.

Referenced by message_New().

7.57.2.12 `char* message_s::from_address`

Definition at line 93 of file message.h.

Referenced by `acc_MessageHandlerThread()`, `AP_QUEUE_SEARCH_TEMPLATE()`, `message_Destroy()`, `message_InitializeFromAgent()`, `message_InitializeFromConnection()`, `message_InitializeFromString()`, and `message_New()`.

7.57.2.13 `char* message_s::to_address`

Definition at line 94 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()`, `mtp_http_ComposeMessage()`, and `mtp_http_CreateMessage()`.

7.57.2.14 `char* message_s::target`

Definition at line 96 of file message.h.

Referenced by `MC_AclSend()`, `message_Destroy()`, `message_InitializeFromAgent()`, `message_InitializeFromConnection()`, `message_InitializeFromString()`, `message_New()`, and `mtp_http_ComposeMessage()`.

7.57.2.15 `int message_s::agent_xml_flag`

Definition at line 103 of file message.h.

Referenced by `agent_Initialize()`, `message_Destroy()`, `message_InitializeFromAgent()`, and `message_New()`.

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

- `/home/dko/projects/mobilec/trunk/src/include/message.h`

7.58 mpi Struct Reference

MPI structure.

```
#include <bignum.h>
```

Data Fields

- [int s](#)
- [int n](#)
- [t_int * p](#)

7.58.1 Detailed Description

MPI structure.

Definition at line 53 of file bignum.h.

7.58.2 Field Documentation

7.58.2.1 int mpi::s

integer sign

Definition at line 55 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_gcd\(\)](#), [mpi_grow\(\)](#), [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\(\)](#).

7.58.2.2 int mpi::n

total # of limbs

Definition at line 56 of file bignum.h.

Referenced by [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_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 [x509_cert_info\(\)](#).

7.58.2.3 t_int* mpi::p

pointer to limbs

Definition at line 57 of file bignum.h.

Referenced by [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_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\(\)](#).

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

- </home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/bignum.h>

7.59 mtp_http_content_s Struct Reference

```
#include <mtp_http.h>
```

Data Fields

- char * [content_type](#)
- void * [data](#)

7.59.1 Detailed Description

Definition at line 103 of file mtp_http.h.

7.59.2 Field Documentation

7.59.2.1 char* mtp_http_content_s::content_type

Definition at line 105 of file mtp_http.h.

Referenced by MC_AclSend(), mtp_http_CreateMessage(), mtp_http_Destroy(), and mtp_http_Parse().

7.59.2.2 void* mtp_http_content_s::data

Definition at line 106 of file mtp_http.h.

Referenced by acc_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/trunk/src/include/[mtp_http.h](#)

7.60 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](#)

7.60.1 Detailed Description

Definition at line 109 of file `mtp_http.h`.

7.60.2 Field Documentation

7.60.2.1 enum `http_status_code_e` `mtp_http_s::http_status_code`

Definition at line 111 of file `mtp_http.h`.

7.60.2.2 enum `http_performative_e` `mtp_http_s::http_performative`

Definition at line 112 of file `mtp_http.h`.

Referenced by `acc_Thread()`, `http_ParseHeader()`, and `mtp_http_Parse()`.

7.60.2.3 char* `mtp_http_s::http_version`

Definition at line 113 of file `mtp_http.h`.

Referenced by `mtp_http_Destroy()`.

7.60.2.4 char* mtp_http_s::host

Definition at line 114 of file mtp_http.h.

Referenced by MC_AclSend(), mtp_http_CreateMessage(), and mtp_http_Destroy().

7.60.2.5 char* mtp_http_s::return_code

Definition at line 115 of file mtp_http.h.

Referenced by mtp_http_Destroy().

7.60.2.6 char* mtp_http_s::target

Definition at line 116 of file mtp_http.h.

Referenced by acc_Thread(), http_ParseHeader(), MC_AclSend(), mtp_http_CreateMessage(), and mtp_http_Destroy().

7.60.2.7 char* mtp_http_s::date

Definition at line 118 of file mtp_http.h.

Referenced by mtp_http_Destroy().

7.60.2.8 char* mtp_http_s::server

Definition at line 119 of file mtp_http.h.

Referenced by mtp_http_Destroy().

7.60.2.9 char* mtp_http_s::accept_ranges

Definition at line 120 of file mtp_http.h.

Referenced by mtp_http_Destroy().

7.60.2.10 char* mtp_http_s::content_length

Definition at line 121 of file mtp_http.h.

Referenced by mtp_http_Destroy().

7.60.2.11 char* mtp_http_s::connection

Definition at line 122 of file mtp_http.h.

Referenced by mtp_http_Destroy().

7.60.2.12 char* mtp_http_s::content_type

Definition at line 123 of file mtp_http.h.

Referenced by `mtp_http_Destroy()`, and `mtp_http_Parse()`.

7.60.2.13 char* mtp_http_s::user_agent

Definition at line 124 of file `mtp_http.h`.

Referenced by `mtp_http_Destroy()`.

7.60.2.14 char* mtp_http_s::cache_control

Definition at line 126 of file `mtp_http.h`.

7.60.2.15 char* mtp_http_s::mime_version

Definition at line 127 of file `mtp_http.h`.

7.60.2.16 int mtp_http_s::response_code

Definition at line 130 of file `mtp_http.h`.

Referenced by `http_ParseHeader()`.

7.60.2.17 char* mtp_http_s::response_string

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

Referenced by `http_ParseHeader()`.

7.60.2.18 int mtp_http_s::message_parts

Definition at line 135 of file `mtp_http.h`.

Referenced by `acc_Thread()`, `MC_AclSend()`, `mtp_http_CreateMessage()`, `mtp_http_Destroy()`, and `mtp_http_Parse()`.

7.60.2.19 char* mtp_http_s::boundary

Definition at line 136 of file `mtp_http.h`.

Referenced by `mtp_http_Destroy()`, and `mtp_http_Parse()`.

7.60.2.20 struct mtp_http_content_s* mtp_http_s::content [read]

Definition at line 137 of file `mtp_http.h`.

Referenced by `acc_Thread()`, `MC_AclSend()`, `mtp_http_CreateMessage()`, `mtp_http_Destroy()`, `mtp_http_New()`, and `mtp_http_Parse()`.

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

- `/home/dko/projects/mobilec/trunk/src/include/mtp_http.h`

7.61 mxml_attr_s Struct Reference

```
#include <mxml.h>
```

Data Fields

- char * [name](#)
- char * [value](#)

7.61.1 Detailed Description

Definition at line 83 of file mxml.h.

7.61.2 Field Documentation

7.61.2.1 char* mxml_attr_s::name

Definition at line 85 of file mxml.h.

Referenced by `mxml_write_node()`, `mxmlDelete()`, `mxmlElementGetAttr()`, and `mxmlElementSetAttr()`.

7.61.2.2 char* mxml_attr_s::value

Definition at line 86 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/trunk/src/mxml-2.2.2/mxml.h`

7.62 mxml_custom_s Struct Reference

```
#include <mxml.h>
```

Data Fields

- void * [data](#)
- void(* [destroy](#))(void *)

7.62.1 Detailed Description

Definition at line 102 of file mxml.h.

7.62.2 Field Documentation

7.62.2.1 void* mxml_custom_s::data

Definition at line 104 of file mxml.h.

Referenced by mxmlDelete(), mxmlNewCustom(), and mxmlSetCustom().

7.62.2.2 void(* mxml_custom_s::destroy)(void *)

Referenced by mxmlDelete(), mxmlNewCustom(), and mxmlSetCustom().

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

- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/[mxml.h](#)

7.63 mxml_fdbuf_s Struct Reference

Data Fields

- [int fd](#)
- unsigned char * [current](#)
- unsigned char * [end](#)
- unsigned char [buffer](#) [8192]

7.63.1 Detailed Description

Definition at line 80 of file mxml-file.c.

7.63.2 Field Documentation

7.63.2.1 int mxml_fdbuf_s::fd

Definition at line 82 of file mxml-file.c.

Referenced by [mxml_fd_read\(\)](#), [mxml_fd_write\(\)](#), [mxxmlLoadFd\(\)](#), and [mxxmlSaveFd\(\)](#).

7.63.2.2 unsigned char* mxml_fdbuf_s::current

Definition at line 83 of file mxml-file.c.

Referenced by [mxml_fd_getc\(\)](#), [mxml_fd_putc\(\)](#), [mxml_fd_read\(\)](#), [mxml_fd_write\(\)](#), [mxxmlLoadFd\(\)](#), and [mxxmlSaveFd\(\)](#).

7.63.2.3 unsigned char * mxml_fdbuf_s::end

Definition at line 83 of file mxml-file.c.

Referenced by [mxml_fd_getc\(\)](#), [mxml_fd_putc\(\)](#), [mxml_fd_read\(\)](#), [mxxmlLoadFd\(\)](#), and [mxxmlSaveFd\(\)](#).

7.63.2.4 unsigned char mxml_fdbuf_s::buffer[8192]

Definition at line 83 of file mxml-file.c.

Referenced by [mxml_fd_read\(\)](#), [mxml_fd_write\(\)](#), [mxxmlLoadFd\(\)](#), and [mxxmlSaveFd\(\)](#).

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

- [/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-file.c](#)

7.64 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

7.64.1 Detailed Description

Definition at line 130 of file mxml.h.

7.64.2 Field Documentation

7.64.2.1 char* mxml_index_s::attr

Definition at line 132 of file mxml.h.

Referenced by index_compare(), index_find(), mxmlIndexDelete(), mxmlIndexFind(), and mxmlIndexNew().

7.64.2.2 int mxml_index_s::num_nodes

Definition at line 133 of file mxml.h.

Referenced by main(), mxmlIndexEnum(), mxmlIndexFind(), mxmlIndexNew(), and mxmlIndexReset().

7.64.2.3 int mxml_index_s::alloc_nodes

Definition at line 134 of file mxml.h.

Referenced by mxmlIndexDelete(), and mxmlIndexNew().

7.64.2.4 int mxml_index_s::cur_node

Definition at line 135 of file mxml.h.

Referenced by mxmlIndexEnum(), mxmlIndexFind(), and mxmlIndexReset().

7.64.2.5 mxml_node_t** mxml_index_s::nodes

Definition at line 136 of file mxml.h.

Referenced by index_sort(), mxmlIndexDelete(), mxmlIndexEnum(), mxmlIndexFind(), mxmlIndexNew(), and mxmlIndexReset().

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

- </home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml.h>

7.65 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

7.65.1 Detailed Description

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

7.65.2 Field Documentation

7.65.2.1 mxml_type_t mxml_node_s::type

Definition at line 121 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()`.

7.65.2.2 struct mxml_node_s* mxml_node_s::next [read]

Definition at line 122 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()`.

7.65.2.3 struct mxml_node_s* mxml_node_s::prev [read]

Definition at line 123 of file `mxml.h`.

Referenced by `mxml_write_node()`, `mxmlAdd()`, `mxmlRemove()`, and `mxmlWalkPrev()`.

7.65.2.4 struct mxml_node_s* mxml_node_s::parent [read]

Definition at line 124 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()`.

7.65.2.5 struct mxml_node_s* mxml_node_s::child [read]

Definition at line 125 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().

7.65.2.6 struct mxml_node_s* mxml_node_s::last_child [read]

Definition at line 126 of file mxml.h.

Referenced by add_variable(), main(), mxmlAdd(), mxmlRemove(), mxmlWalkPrev(), and scan_file().

7.65.2.7 mxml_value_t mxml_node_s::value

Definition at line 127 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/trunk/src/mxml-2.2.2/[mxml.h](#)

7.66 mxml_text_s Struct Reference

```
#include <mxml.h>
```

Data Fields

- `int whitespace`
- `char * string`

7.66.1 Detailed Description

Definition at line 96 of file mxml.h.

7.66.2 Field Documentation

7.66.2.1 `int mxml_text_s::whitespace`

Definition at line 98 of file mxml.h.

Referenced by `add_variable()`, `main()`, `mxml_write_node()`, `mxmlNewText()`, `mxmlNewTextf()`, `mxmlSetText()`, `mxmlSetTextf()`, `scan_file()`, and `write_element()`.

7.66.2.2 `char* mxml_text_s::string`

Definition at line 99 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 struct was generated from the following file:

- `/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml.h`

7.67 mxml_value_s Struct Reference

```
#include <mxml.h>
```

Data Fields

- `char * name`
- `int num_attrs`
- `mxml_attr_t * attrs`

7.67.1 Detailed Description

Definition at line 89 of file `mxml.h`.

7.67.2 Field Documentation

7.67.2.1 `char* mxml_value_s::name`

Definition at line 91 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()`, `mxmllIndexNew()`, `mxmlNewElement()`, `mxmlSetElement()`, `scan_file()`, `sort_node()`, `type_cb()`, `update_comment()`, `whitespace_cb()`, `ws_cb()`, `xml_get_element_name()`, and `xml_get_text()`.

7.67.2.2 `int mxml_value_s::num_attrs`

Definition at line 92 of file `mxml.h`.

Referenced by `mxml_write_node()`, `mxmlDelete()`, `mxmlElementGetAttr()`, and `mxmlElementSetAttr()`.

7.67.2.3 `mxml_attr_t* mxml_value_s::attrs`

Definition at line 93 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/trunk/src/mxml-2.2.2/mxml.h`

7.68 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

7.68.1 Detailed Description

Definition at line 109 of file mxml.h.

7.68.2 Field Documentation

7.68.2.1 mxml_element_t mxml_value_u::element

Definition at line 111 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\(\)](#).

7.68.2.2 int mxml_value_u::integer

Definition at line 112 of file mxml.h.

Referenced by [main\(\)](#), [mxml_write_node\(\)](#), [mxmlNewInteger\(\)](#), and [mxmlSetInteger\(\)](#).

7.68.2.3 char* mxml_value_u::opaque

Definition at line 113 of file mxml.h.

Referenced by [main\(\)](#), [mxml_write_node\(\)](#), [mxmlDelete\(\)](#), [mxmlNewOpaque\(\)](#), and [mxmlSetOpaque\(\)](#).

7.68.2.4 double mxml_value_u::real

Definition at line 114 of file mxml.h.

Referenced by [main\(\)](#), [mxml_write_node\(\)](#), [mxmlNewReal\(\)](#), and [mxmlSetReal\(\)](#).

7.68.2.5 mxml_text_t mxml_value_u::text

Definition at line 115 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()`.

7.68.2.6 mxml_custom_t mxml_value_u::custom

Definition at line 116 of file `mxml.h`.

Referenced by `mxmlDelete()`, `mxmlNewCustom()`, and `mxmlSetCustom()`.

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

- `/home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml.h`

7.69 rsa_context Struct Reference

```
#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](#)

7.69.1 Detailed Description

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

7.69.2 Field Documentation

7.69.2.1 `int rsa_context::ver`

should be 0

Definition at line 54 of file `rsa.h`.

Referenced by `x509_parse_key()`.

7.69.2.2 `int rsa_context::len`

size(N) in chars

Definition at line 55 of file `rsa.h`.

Referenced by `main()`, `rsa_gen_key()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, `rsa_pkcs1_verify()`, `rsa_private()`, `rsa_public()`, `rsa_read_private()`, `rsa_read_public()`, `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()`, `x509_add_certs()`, and `x509_parse_key()`.

7.69.2.3 `mpi rsa_context::N`

public modulus

Definition at line 56 of file `rsa.h`.

Referenced by `main()`, `rsa_check_privkey()`, `rsa_check_pubkey()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_public()`, `rsa_read_private()`, `rsa_read_public()`, `rsa_write_private()`, `rsa_write_public()`, `x509_add_certs()`, `x509_cert_info()`, and `x509_parse_key()`.

7.69.2.4 `mpi rsa_context::E`

public exponent

Definition at line 57 of file `rsa.h`.

Referenced by `main()`, `rsa_check_privkey()`, `rsa_check_pubkey()`, `rsa_free()`, `rsa_gen_key()`, `rsa_public()`, `rsa_read_private()`, `rsa_read_public()`, `rsa_write_private()`, `rsa_write_public()`, `x509_add_certs()`, and `x509_parse_key()`.

7.69.2.5 `mpi rsa_context::D`

private exponent

Definition at line 58 of file `rsa.h`.

Referenced by `main()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_read_private()`, `rsa_write_private()`, and `x509_parse_key()`.

7.69.2.6 `mpi rsa_context::P`

1st prime factor

Definition at line 60 of file `rsa.h`.

Referenced by `main()`, `rsa_check_privkey()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_read_private()`, `rsa_write_private()`, and `x509_parse_key()`.

7.69.2.7 `mpi rsa_context::Q`

2nd prime factor

Definition at line 61 of file `rsa.h`.

Referenced by `main()`, `rsa_check_privkey()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_read_private()`, `rsa_write_private()`, and `x509_parse_key()`.

7.69.2.8 `mpi rsa_context::DP`

$D \bmod (P - 1)$

Definition at line 62 of file `rsa.h`.

Referenced by `main()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_read_private()`, `rsa_write_private()`, and `x509_parse_key()`.

7.69.2.9 `mpi rsa_context::DQ`

$D \bmod (Q - 1)$

Definition at line 63 of file `rsa.h`.

Referenced by `main()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_read_private()`, `rsa_write_private()`, and `x509_parse_key()`.

7.69.2.10 `mpi rsa_context::QP`

inverse of $Q \% P$

Definition at line 64 of file `rsa.h`.

Referenced by `main()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_read_private()`, `rsa_write_private()`, and `x509_parse_key()`.

7.69.2.11 `mpi rsa_context::RN`

cached $R^2 \bmod N$

Definition at line 66 of file `rsa.h`.

Referenced by `rsa_free()`, `rsa_private()`, and `rsa_public()`.

7.69.2.12 `mpi rsa_context::RP`

cached $R^2 \bmod P$

Definition at line 67 of file `rsa.h`.

Referenced by `rsa_free()`, and `rsa_private()`.

7.69.2.13 `mpi rsa_context::RQ`

cached $R^2 \bmod Q$

Definition at line 68 of file `rsa.h`.

Referenced by `rsa_free()`, and `rsa_private()`.

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

- `/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/rsa.h`

7.70 LibMC::Properties::Settings Class Reference

Public Member Functions

- [Settings](#) ()

Properties

- static [Settings Default](#) [get]

Private Member Functions

- void [SettingChangingEventHandler](#) (object sender, System.Configuration.SettingChangingEventArgs e)
- void [SettingsSavingEventHandler](#) (object sender, System.ComponentModel.CancelEventArgs e)

Static Private Attributes

- static [Settings defaultInstance](#) = (([Settings](#))(global::System.Configuration.ApplicationSettingsBase.Synchronized(new [Settings](#)())))

7.70.1 Detailed Description

Definition at line 16 of file Settings.Designer.cs.

7.70.2 Constructor & Destructor Documentation

7.70.2.1 LibMC::Properties::Settings::Settings () [inline]

Definition at line 19 of file Settings.cs.

7.70.3 Member Function Documentation

7.70.3.1 void LibMC::Properties::Settings::SettingChangingEventHandler (object sender, System.Configuration.SettingChangingEventArgs e) [inline, private]

Definition at line 28 of file Settings.cs.

7.70.3.2 void LibMC::Properties::Settings::SettingsSavingEventHandler (object sender, System.ComponentModel.CancelEventArgs e) [inline, private]

Definition at line 32 of file Settings.cs.

7.70.4 Field Documentation

7.70.4.1 Settings LibMC::Properties::Settings::defaultInstance =
((Settings)(global::System.Configuration.ApplicationSettingsBase.Synchronized(new
Settings()))) [static, private]

Definition at line 18 of file Settings.Designer.cs.

7.70.5 Property Documentation

7.70.5.1 Settings LibMC::Properties::Settings::Default [static, get]

Definition at line 20 of file Settings.Designer.cs.

The documentation for this class was generated from the following files:

- /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties/[Settings.Designer.cs](#)
- /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/[Settings.cs](#)

7.71 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]

7.71.1 Detailed Description

SHA-1 context structure.

Definition at line 14 of file sha1.h.

7.71.2 Field Documentation

7.71.2.1 unsigned long sha1_context::total[2]

number of bytes processed

Definition at line 16 of file sha1.h.

Referenced by sha1_finish(), sha1_starts(), and sha1_update().

7.71.2.2 unsigned long sha1_context::state[5]

intermediate digest state

Definition at line 17 of file sha1.h.

Referenced by sha1_finish(), sha1_process(), and sha1_starts().

7.71.2.3 unsigned char sha1_context::buffer[64]

data block being processed

Definition at line 18 of file sha1.h.

Referenced by sha1_update().

7.71.2.4 unsigned char sha1_context::ipad[64]

HMAC: inner padding

Definition at line 19 of file sha1.h.

Referenced by sha1_hmac_starts().

7.71.2.5 unsigned char sha1_context::opad[64]

HMAC: outer padding

Definition at line 20 of file sha1.h.

Referenced by sha1_hmac_finish(), and sha1_hmac_starts().

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

- </home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/sha1.h>

7.72 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](#)

7.72.1 Detailed Description

SHA-256 context structure.

Definition at line 14 of file sha2.h.

7.72.2 Field Documentation

7.72.2.1 unsigned long sha2_context::total[2]

number of bytes processed

Definition at line 16 of file sha2.h.

Referenced by sha2_finish(), sha2_starts(), and sha2_update().

7.72.2.2 unsigned long sha2_context::state[8]

intermediate digest state

Definition at line 17 of file sha2.h.

Referenced by sha2_finish(), sha2_process(), and sha2_starts().

7.72.2.3 unsigned char sha2_context::buffer[64]

data block being processed

Definition at line 18 of file sha2.h.

Referenced by sha2_update().

7.72.2.4 unsigned char sha2_context::ipad[64]

HMAC: inner padding

Definition at line 19 of file sha2.h.

Referenced by sha2_hmac_starts().

7.72.2.5 unsigned char sha2_context::opad[64]

HMAC: outer padding

Definition at line 20 of file sha2.h.

Referenced by sha2_hmac_finish(), and sha2_hmac_starts().

7.72.2.6 int sha2_context::is224

0 if SHA-256, 1 if SHA-224

Definition at line 21 of file sha2.h.

Referenced by sha2_finish(), sha2_hmac_finish(), and sha2_starts().

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

- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/[sha2.h](#)

7.73 sha4_context Struct Reference

SHA-512 context structure.

```
#include <sha4.h>
```

Data Fields

- uint64 [total](#) [2]
- uint64 [state](#) [8]
- unsigned char [buffer](#) [128]
- unsigned char [ipad](#) [64]
- unsigned char [opad](#) [64]
- int [is384](#)

7.73.1 Detailed Description

SHA-512 context structure.

Definition at line 22 of file sha4.h.

7.73.2 Field Documentation

7.73.2.1 uint64 sha4_context::total[2]

number of bytes processed

Definition at line 24 of file sha4.h.

Referenced by sha4_finish(), sha4_starts(), and sha4_update().

7.73.2.2 uint64 sha4_context::state[8]

intermediate digest state

Definition at line 25 of file sha4.h.

Referenced by sha4_finish(), sha4_process(), and sha4_starts().

7.73.2.3 unsigned char sha4_context::buffer[128]

data block being processed

Definition at line 26 of file sha4.h.

Referenced by sha4_update().

7.73.2.4 unsigned char sha4_context::ipad[64]

HMAC: inner padding

Definition at line 27 of file sha4.h.

Referenced by sha4_hmac_starts().

7.73.2.5 unsigned char sha4_context::opad[64]

HMAC: outer padding

Definition at line 28 of file sha4.h.

Referenced by sha4_hmac_finish(), and sha4_hmac_starts().

7.73.2.6 int sha4_context::is384

0 if SHA-512, 1 if SHA-384

Definition at line 29 of file sha4.h.

Referenced by sha4_finish(), sha4_hmac_finish(), and sha4_starts().

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

- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/[sha4.h](#)

7.74 ssl_context Struct Reference

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

Data Fields

- `int` `state`
- `int` `major_ver`
- `int` `minor_ver`
- `unsigned char` `max_ver` [2]
- `unsigned char *` `in_ctr`
- `unsigned char *` `in_hdr`
- `unsigned char *` `in_msg`
- `unsigned char *` `in_offt`
- `int` `read_fd`
- `int` `in_msgtype`
- `int` `in_msglen`
- `int` `in_left`
- `int` `in_hslen`
- `int` `nb_zero`
- `unsigned char *` `out_ctr`
- `unsigned char *` `out_hdr`
- `unsigned char *` `out_msg`
- `int` `write_fd`
- `int` `out_msgtype`
- `int` `out_msglen`
- `int` `out_left`
- `int` `out_uoff`
- `rsa_context *` `own_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`
- `int` `resumed`
- `int` `sidlen`
- `unsigned char` `sessid` [32]
- `unsigned char *` `sidtable`
- `md5_context` `hs_md5`
- `sha1_context` `hs_sha1`
- `dhm_context` `dhm_ctx`
- `int`(`* rng_f`)(`void *`)
- `void *` `rng_d`
- `int` `pmslen`
- `unsigned char` `premaster` [256]
- `unsigned char` `randbytes` [64]

- unsigned char `master` [48]
- `int * cipherlist`
- `int cipher`
- `int keylen`
- `int minlen`
- `int ctxlen`
- `void * ctx_enc`
- `void * ctx_dec`
- `int ivlen`
- unsigned char `iv_enc` [16]
- unsigned char `iv_dec` [16]
- `int maclen`
- unsigned char `mac_enc` [32]
- unsigned char `mac_dec` [32]

7.74.1 Detailed Description

Definition at line 129 of file `ssl.h`.

7.74.2 Field Documentation

7.74.2.1 `int ssl_context::state`

SSL handshake: current state

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

Referenced by `ssl_client_start()`, `ssl_close_notify()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `ssl_server_start()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

7.74.2.2 `int ssl_context::major_ver`

equal to `SSLV3_MAJOR_VERSION`

Definition at line 136 of file `ssl.h`.

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

7.74.2.3 `int ssl_context::minor_ver`

either 0: SSLv3, or 1: TLSv1.0

Definition at line 137 of file `ssl.h`.

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

7.74.2.4 unsigned char ssl_context::max_ver[2]

max. version supported by client

Definition at line 138 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().

7.74.2.5 unsigned char* ssl_context::in_ctr

64-bit incoming message counter

Definition at line 143 of file ssl.h.

Referenced by ssl_decrypt_buf(), ssl_free(), and ssl_init().

7.74.2.6 unsigned char* ssl_context::in_hdr

5-byte record header (in_ctr+8)

Definition at line 144 of file ssl.h.

Referenced by ssl_decrypt_buf(), ssl_init(), and ssl_parse_client_hello().

7.74.2.7 unsigned char* ssl_context::in_msg

the message payload (in_hdr+5)

Definition at line 145 of file ssl.h.

Referenced by ssl_decrypt_buf(), ssl_init(), ssl_parse_certificate_request(), ssl_parse_certificate_verify(), ssl_parse_client_hello(), ssl_parse_client_key_exchange(), ssl_parse_server_hello(), ssl_parse_server_hello_done(), ssl_parse_server_key_exchange(), and ssl_read().

7.74.2.8 unsigned char* ssl_context::in_offt

read offset in message payload

Definition at line 146 of file ssl.h.

Referenced by ssl_read().

7.74.2.9 int ssl_context::read_fd

descriptor for read operations

Definition at line 148 of file ssl.h.

Referenced by ssl_parse_client_hello().

7.74.2.10 int ssl_context::in_msgtype

record header: message type

Definition at line 149 of file ssl.h.

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

7.74.2.11 `int ssl_context::in_msglen`

record header: message length

Definition at line 150 of file `ssl.h`.

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

7.74.2.12 `int ssl_context::in_left`

amount of (tcp) data read so far

Definition at line 152 of file `ssl.h`.

Referenced by `ssl_parse_client_hello()`.

7.74.2.13 `int ssl_context::in_hslen`

current handshake message length

Definition at line 153 of file `ssl.h`.

Referenced by `ssl_parse_certificate_verify()`, `ssl_parse_client_key_exchange()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, and `ssl_parse_server_key_exchange()`.

7.74.2.14 `int ssl_context::nb_zero`

of 0-length encrypted messages

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

Referenced by `ssl_decrypt_buf()`.

7.74.2.15 `unsigned char* ssl_context::out_ctr`

64-bit outgoing message counter

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

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

7.74.2.16 `unsigned char* ssl_context::out_hdr`

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

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

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

7.74.2.17 unsigned char* ssl_context::out_msg

the message payload (out_hdr+5)

Definition at line 161 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_client_hello(), ssl_write_client_key_exchange(), ssl_write_record(), ssl_write_server_hello(), ssl_write_server_hello_done(), and ssl_write_server_key_exchange().

7.74.2.18 int ssl_context::write_fd

descriptor for write operations

Definition at line 163 of file ssl.h.

Referenced by ssl_flush_output(), and ssl_write_record().

7.74.2.19 int ssl_context::out_msgtype

record header: message type

Definition at line 164 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_client_hello(), ssl_write_client_key_exchange(), ssl_write_record(), ssl_write_server_hello(), ssl_write_server_hello_done(), and ssl_write_server_key_exchange().

7.74.2.20 int ssl_context::out_msglen

record header: message length

Definition at line 165 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_client_hello(), ssl_write_client_key_exchange(), ssl_write_record(), ssl_write_server_hello(), ssl_write_server_hello_done(), and ssl_write_server_key_exchange().

7.74.2.21 int ssl_context::out_left

amount of data not yet written

Definition at line 167 of file ssl.h.

Referenced by ssl_flush_output(), and ssl_write_record().

7.74.2.22 int ssl_context::out_uoff

offset in user-supplied buffer

Definition at line 168 of file ssl.h.

Referenced by ssl_write().

7.74.2.23 rsa_context* ssl_context::own_key

own RSA private key

Definition at line 173 of file ssl.h.

Referenced by ssl_parse_certificate_request(), ssl_parse_client_key_exchange(), ssl_write_certificate_verify(), and ssl_write_server_key_exchange().

7.74.2.24 x509_cert* ssl_context::own_cert

own X.509 certificate

Definition at line 174 of file ssl.h.

Referenced by ssl_parse_certificate_request(), and ssl_write_certificate().

7.74.2.25 x509_cert* ssl_context::ca_chain

own trusted CA chain

Definition at line 175 of file ssl.h.

Referenced by ssl_write_certificate_request().

7.74.2.26 x509_cert* ssl_context::peer_cert

peer X.509 cert chain

Definition at line 176 of file ssl.h.

Referenced by ssl_free(), ssl_parse_certificate_verify(), ssl_parse_server_key_exchange(), and ssl_write_client_key_exchange().

7.74.2.27 char* ssl_context::peer_cn

expected peer CN

Definition at line 177 of file ssl.h.

7.74.2.28 int ssl_context::endpoint

0: client, 1: server

Definition at line 179 of file ssl.h.

Referenced by ssl_derive_keys(), ssl_handshake(), ssl_set_endpoint(), and ssl_write_certificate().

7.74.2.29 int ssl_context::authmode

verification mode

Definition at line 180 of file ssl.h.

Referenced by ssl_parse_certificate_verify(), and ssl_write_certificate_request().

7.74.2.30 int ssl_context::client_auth

flag for client auth.

Definition at line 181 of file ssl.h.

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

7.74.2.31 int ssl_context::verify_result

verification result

Definition at line 182 of file ssl.h.

Referenced by ssl_get_verify_result().

7.74.2.32 int ssl_context::resumed

session resuming flag

Definition at line 187 of file ssl.h.

Referenced by ssl_derive_keys(), ssl_parse_server_hello(), and ssl_write_server_hello().

7.74.2.33 int ssl_context::sidlen

session id length

Definition at line 188 of file ssl.h.

Referenced by ssl_get_session(), ssl_init(), ssl_parse_client_hello(), ssl_parse_server_hello(), ssl_set_session(), ssl_write_client_hello(), and ssl_write_server_hello().

7.74.2.34 unsigned char ssl_context::sessid[32]

session id

Definition at line 189 of file ssl.h.

Referenced by ssl_get_session(), ssl_init(), ssl_parse_client_hello(), ssl_parse_server_hello(), ssl_set_session(), ssl_write_client_hello(), and ssl_write_server_hello().

7.74.2.35 unsigned char* ssl_context::sidtable

table of session IDs

Definition at line 190 of file ssl.h.

Referenced by ssl_get_session(), and ssl_set_session().

7.74.2.36 md5_context ssl_context::hs_md5

MD5(Handshake msgs)

Definition at line 195 of file ssl.h.

Referenced by `ssl_calc_verify()`, `ssl_parse_client_hello()`, `ssl_write_client_hello()`, and `ssl_write_record()`.

7.74.2.37 sha1_context ssl_context::hs_sha1

SHA1(Handshake msgs)

Definition at line 196 of file `ssl.h`.

Referenced by `ssl_calc_verify()`, `ssl_parse_client_hello()`, `ssl_write_client_hello()`, and `ssl_write_record()`.

7.74.2.38 dhm_context ssl_context::dhm_ctx

DHM key exchange

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

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

7.74.2.39 int(* ssl_context::rng_f)(void *)

RNG function

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

7.74.2.40 void* ssl_context::rng_d

RNG data

Definition at line 200 of file `ssl.h`.

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

7.74.2.41 int ssl_context::pmslen

premaster length

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

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

7.74.2.42 unsigned char ssl_context::premaster[256]

premaster secret

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

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

7.74.2.43 unsigned char ssl_context::randbytes[64]

random bytes

Definition at line 204 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().

7.74.2.44 unsigned char ssl_context::master[48]

master secret

Definition at line 205 of file ssl.h.

Referenced by ssl_calc_verify(), ssl_derive_keys(), ssl_get_session(), ssl_init(), and ssl_set_session().

7.74.2.45 int* ssl_context::cipherlist

accepted ciphersuites

Definition at line 207 of file ssl.h.

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

7.74.2.46 int ssl_context::cipher

current chosen cipher

Definition at line 208 of file ssl.h.

Referenced by ssl_derive_keys(), ssl_get_session(), ssl_parse_client_hello(), ssl_parse_client_key_exchange(), ssl_parse_server_hello(), ssl_parse_server_key_exchange(), ssl_set_session(), ssl_write_client_key_exchange(), ssl_write_server_hello(), and ssl_write_server_key_exchange().

7.74.2.47 int ssl_context::keylen

symmetric key length

Definition at line 209 of file ssl.h.

Referenced by ssl_derive_keys().

7.74.2.48 int ssl_context::minlen

min. ciphertext length

Definition at line 210 of file ssl.h.

Referenced by ssl_decrypt_buf(), and ssl_derive_keys().

7.74.2.49 int ssl_context::ctxlen

cipher context length

Definition at line 212 of file ssl.h.

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

7.74.2.50 void* ssl_context::ctx_enc

encryption context

Definition at line 213 of file ssl.h.

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

7.74.2.51 void* ssl_context::ctx_dec

decryption context

Definition at line 214 of file ssl.h.

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

7.74.2.52 int ssl_context::ivlen

IV length

Definition at line 216 of file ssl.h.

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

7.74.2.53 unsigned char ssl_context::iv_enc[16]

IV (encryption)

Definition at line 217 of file ssl.h.

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

7.74.2.54 unsigned char ssl_context::iv_dec[16]

IV (decryption)

Definition at line 218 of file ssl.h.

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

7.74.2.55 int ssl_context::maclen

MAC length

Definition at line 220 of file ssl.h.

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

7.74.2.56 unsigned char ssl_context::mac_enc[32]

MAC (encryption)

Definition at line 221 of file ssl.h.

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

7.74.2.57 unsigned char ssl_context::mac_dec[32]

MAC (decryption)

Definition at line 222 of file ssl.h.

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

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

- `/home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/ssl.h`

7.75 syncList_s Struct Reference

```
#include <sync_list.h>
```

Data Fields

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

7.75.1 Detailed Description

Definition at line 23 of file `sync_list.h`.

7.75.2 Field Documentation

7.75.2.1 `RWLOCK_T* syncList_s::lock`

Definition at line 24 of file `sync_list.h`.

Referenced by `syncListAddNode()`, `syncListDelete()`, `syncListFind()`, `syncListInit()`, and `syncListRemove()`.

7.75.2.2 `MUTEX_T* syncList_s::giant_lock`

Definition at line 25 of file `sync_list.h`.

Referenced by `MC_SyncDelete()`, and `MC_SyncInit()`.

7.75.2.3 `list_p syncList_s::list`

Definition at line 27 of file `sync_list.h`.

Referenced by `syncListAddNode()`, `syncListDelete()`, `syncListFind()`, and `syncListRemove()`.

7.75.2.4 `int syncList_s::size`

Definition at line 28 of file `sync_list.h`.

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

- `/home/dko/projects/mobilec/trunk/src/mc_sync/sync_list.h`

7.76 syncListNode_s Struct Reference

```
#include <sync_list.h>
```

Data Fields

- `MUTEX_T * lock`
- `COND_T * cond`
- `SEMAPHORE_T * sem`
- `int id`
- `int signalled`

7.76.1 Detailed Description

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

7.76.2 Field Documentation

7.76.2.1 `MUTEX_T* syncListNode_s::lock`

Definition at line 14 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()`.

7.76.2.2 `COND_T* syncListNode_s::cond`

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

Referenced by `MC_CondBroadcast()`, `MC_CondSignal()`, `MC_CondWait()`, `syncListNodeDestroy()`, `syncListNodeInit()`, and `syncListNodeNew()`.

7.76.2.3 `SEMAPHORE_T* syncListNode_s::sem`

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

Referenced by `MC_SemaphorePost()`, `MC_SemaphoreWait()`, `syncListNodeDestroy()`, `syncListNodeInit()`, and `syncListNodeNew()`.

7.76.2.4 `int syncListNode_s::id`

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

Referenced by `MC_SyncInit()`, `syncListAddNode()`, `syncListDelete()`, and `syncListRemove()`.

7.76.2.5 `int syncListNode_s::signalled`

Definition at line 18 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/trunk/src/mc_sync/sync_list.h

Chapter 8

Mobile-C File Documentation

8.1 /home/dko/projects/mobilec/trunk/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 <stdlib.h>
#include "include/acc.h"
#include "include/connection.h"
#include "include/data_structures.h"
#include "include/macros.h"
#include "include/mc_error.h"
#include "include/mc_platform.h"
#include "include/message.h"
#include "include/mtp_http.h"
#include "include/xml_parser.h"
#include "include/fipa_acl_envelope.h"
```

Defines

- #define [BACKLOG](#) 10

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_Start` (`mc_platform_p mc_platform`)
- `void * listen_Thread` (`void *arg`)

8.1.1 Define Documentation

8.1.1.1 #define BACKLOG 10

Definition at line 60 of file `acc.c`.

Referenced by `listen_Thread()`.

8.1.2 Function Documentation

8.1.2.1 int acc_Destroy (acc_p acc)

Definition at line 79 of file `acc.c`.

References `MC_SUCCESS`.

Referenced by `mc_platform_Destroy()`.

8.1.2.2 acc_p acc_Initialize (struct mc_platform_s * mc_platform)

Definition at line 63 of file `acc.c`.

References `COND_INIT`, `COND_T`, `MUTEX_INIT`, and `MUTEX_T`.

Referenced by `mc_platform_Initialize()`.

8.1.2.3 void* acc_MessageHandlerThread (void * arg)

Definition at line 91 of file `acc.c`.

References `agent_Initialize()`, `mc_platform_s::agent_queue`, `agent_s::agent_status`, `AGENT_UPDATE`, `mc_platform_s::ams`, `mc_platform_s::asm_message_queue`, `CANCEL`, `COND_BROADCAST`, `COND_-SIGNAL`, `COND_WAIT`, `agent_s::datastate`, `ENCRYPTED_DATA`, `ENCRYPTION_INITIALIZE`, `FIPA_-ACL`, `message_s::from_address`, `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`, `REQUEST_ENCRYPTION_INITIALIZE`, `RETURN_-MSG`, `SUBSCRIBE`, `message_s::to_address`, and `WARN`.

Referenced by `acc_Start()`.

8.1.2.4 void acc_Start (mc_platform_p mc_platform)

Definition at line 552 of file acc.c.

References mc_platform_s::acc, acc_MessageHandlerThread(), acc_Thread(), listen_Thread(), MC_THREAD_ACC, mc_platform_s::stack_size, and THREAD_CREATE.

Referenced by mc_platform_Initialize().

8.1.2.5 void* acc_Thread (void * arg)

Definition at line 340 of file acc.c.

References agent_mailbox_Post(), mc_platform_s::agent_queue, AGENT_UPDATE, mc_platform_s::asm_message_queue, CANCEL, COND_BROADCAST, COND_WAIT, connection_Destroy(), mc_platform_s::connection_queue, mtp_http_s::content, mtp_http_content_s::data, ENCRYPTED_DATA, ENCRYPTION_INITIALIZE, 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(), mc_platform_s::giant, mc_platform_s::giant_cond, mc_platform_s::giant_lock, mtp_http_s::http_performative, HTTP_POST, HTTP_PUT, agent_s::mailbox, mc_platform, MC_RECV_CONNECTION, mc_platform_s::MC_signal, mc_platform_s::MC_signal_cond, mc_platform_s::MC_signal_lock, 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(), MUTEX_LOCK, MUTEX_UNLOCK, 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, QUER_IF, QUER_REF, mc_platform_s::quit, mc_platform_s::quit_lock, RELAY, REQUEST, REQUEST_ENCRYPTION_INITIALIZE, RETURN_MSG, SUBSCRIBE, mtp_http_s::target, fipa_acl_Param_s::to, WARN, and message_s::xml_root.

Referenced by acc_Start().

8.1.2.6 void* listen_Thread (void * arg)

Definition at line 595 of file acc.c.

References mc_platform_s::acc, BACKLOG, CHECK_NULL, COND_BROADCAST, mc_platform_s::connection_queue, mc_platform, MUTEX_LOCK, MUTEX_UNLOCK, and mc_platform_s::port.

Referenced by acc_Start().

8.2 /home/dko/projects/mobilec/trunk/src/agent.c File Reference

```
#include <unistd.h>
#include <embedch.h>
#include "include/libmc.h"
#include "include/agent.h"
#include "include/mc_platform.h"
#include "include/message.h"
#include "include/agent_lib.h"
#include "include/interpreter_variable_data.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_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_RunChScriptThread](#) (void *ChAgent)

8.2.1 Function Documentation

8.2.1.1 [int agent_AddPersistentVariable](#) ([agent_p](#) agent, [int](#) task_num, const char * var_name)

Definition at line 46 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](#), [agent_datastate_s::task_progress](#), and [agent_datastate_s::tasks](#).

8.2.1.2 [agent_p](#) agent_Copy (const [agent_p](#) agent)

Definition at line 128 of file agent.c.

References [agent_datastate_Copy\(\)](#), [agent_s::agent_interp](#), [agent_s::agent_persistent](#), [agent_s::agent_status](#), [agent_s::agent_thread](#), [agent_s::agent_type](#), [agent_s::arrival_time](#), [agent_s::datastate](#), [agent_s::home](#), [agent_s::home_port](#), [agent_s::id](#), [agent_s::lock](#), [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\(\)](#).

8.2.1.3 [int](#) agent_Destroy ([agent_p](#) agent)

Definition at line 319 of file agent.c.

References agent_datastate_Destroy(), agent_s::agent_interp, agent_mailbox_Destroy(), agent_s::agent_status, agent_s::agent_thread, agent_s::datastate, agent_s::home, agent_s::lock, agent_s::mailbox, MC_AGENT_NEUTRAL, MC_SUCCESS, MUTEX_DESTROY, MUTEX_LOCK, agent_s::name, agent_s::owner, and agent_s::run_lock.

Referenced by agent_Initialize().

8.2.1.4 agent_p agent_Initialize (struct mc_platform_s * mc_platform, message_p message, int id)

Definition at line 213 of file agent.c.

References 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, 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::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, RETURN_MSG, agent_s::run_lock, agent_datastate_s::xml_agent_root, message_s::xml_payload, message_s::xml_root, and agent_datastate_s::xml_root.

Referenced by acc_MessageHandlerThread().

8.2.1.5 agent_p agent_New (void)

Definition at line 190 of file agent.c.

References agent_s::lock, MUTEX_INIT, MUTEX_NEW, and agent_s::run_lock.

Referenced by MC_ComposeAgent().

8.2.1.6 void agent_RunChScript (agent_p agent, mc_platform_p mc_platform)

Definition at line 353 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, THREAD_CREATE, and THREAD_T.

Referenced by ams_ManageAgentList().

8.2.1.7 void* agent_RunChScriptThread (void * ChAgent)

Definition at line 394 of file agent.c.

References agent_s::agent_interp, mc_platform_s::ams, COND_SIGNAL, agent_s::datastate, agent_s::id, mc_platform_s::interp_options, interpreter_variable_data_Destroy(), interpreter_variable_data_Initialize(), interpreter_variable_data_InitializeFromAgent(), MC_AclAddReceiver_chdl(), MC_AclAddReplyTo_chdl(), MC_AclDestroy_chdl(), MC_AclNew_chdl(), MC_AclPost_chdl(), MC_AclReply_chdl(), MC_AclRetrieve_chdl(), MC_AclSend_chdl(), MC_AclSetContent_chdl(), MC_AclSetPerformative_chdl(), MC_AclSetSender_chdl(), MC_AclWaitRetrieve_chdl(), MC_AddAgent_chdl(), MC_AGENT_NEUTRAL, MC_AgentVariableRetrieve_chdl(), MC_AgentVariableSave_chdl(), MC_Barrier_chdl(), MC_BarrierDelete_chdl(), MC_BarrierInit_chdl(), MC_CallAgentFunc_chdl(), MC_ComposeAgent_chdl(), MC_CondBroadcast_chdl(), MC_CondReset_chdl(), MC_CondSignal_chdl(), MC_CondWait_chdl(), MC_DeleteAgent_chdl(), MC_

DeregisterService_chdl(), MC_DestroyServiceSearchResult_chdl(), MC_End_chdl(), MC_EXEC_AGENT, 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_MutexLock_chdl(), MC_MutexUnlock_chdl(), agent_s::mc_platform, mc_platform, MC_PrintAgentCode_chdl(), MC_RegisterService_chdl(), MC_ResumeAgency_chdl(), MC_RetrieveAgent_chdl(), MC_RetrieveAgentCode_chdl(), MC_RETURN_AGENT, 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_platform_s::MC_signal, mc_platform_s::MC_signal_cond, mc_platform_s::MC_signal_lock, MC_SyncDelete_chdl(), MC_SyncInit_chdl(), MC_TerminateAgent_chdl(), MC_WAIT_FINISHED, MC_WAIT_MESSGSEND, MUTEX_LOCK, MUTEX_UNLOCK, agent_s::name, SIGNAL, and agent_datastate_s::task_progress.

Referenced by agent_RunChScript().

8.3 /home/dko/projects/mobilec/trunk/src/agent_datastate.c File Reference

```
#include <mxml.h>
#include "include/agent_datastate.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*)

8.3.1 Function Documentation

8.3.1.1 [agent_datastate_p agent_datastate_Copy](#) (const [agent_datastate_p](#) *datastate*)

Definition at line 38 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()`.

8.3.1.2 [int agent_datastate_Destroy](#) ([agent_datastate_p](#) *agent_datastate*)

Definition at line 126 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()`.

8.3.1.3 [agent_datastate_p agent_datastate_New](#) (void)

Definition at line 106 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::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_ComposeAgent()`.

8.4 /home/dko/projects/mobilec/trunk/src/agent_mailbox.c File Reference

```
#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)

8.4.1 Function Documentation

8.4.1.1 agent_mailbox_p agent_mailbox_Copy (agent_mailbox_p src)

Definition at line 12 of file agent_mailbox.c.

References agent_mailbox_New(), and agent_mailbox_s::mail_queue.

8.4.1.2 int agent_mailbox_Destroy (agent_mailbox_t * mailbox)

Definition at line 20 of file agent_mailbox.c.

References agent_mailbox_s::mail_queue.

Referenced by agent_Destroy().

8.4.1.3 agent_mailbox_p agent_mailbox_New (void)

Definition at line 3 of file agent_mailbox.c.

References agent_mailbox_s::mail_queue.

Referenced by agent_Initialize(), and agent_mailbox_Copy().

8.4.1.4 int agent_mailbox_Post (agent_mailbox_p mailbox, fipa_acl_message_t * message)

Definition at line 29 of file agent_mailbox.c.

References agent_mailbox_s::mail_queue.

Referenced by acc_Thread(), and MC_AclPost().

8.4.1.5 fipa_acl_message_t* agent_mailbox_Retrieve (agent_mailbox_p mailbox)

Definition at line 35 of file agent_mailbox.c.

References agent_mailbox_s::mail_queue.

Referenced by agent_mailbox_WaitRetrieve(), and MC_AclRetrieve().

8.4.1.6 fipa_acl_message_t* agent_mailbox_WaitRetrieve (agent_mailbox_p *mailbox*)

Definition at line 40 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().

8.5 /home/dko/projects/mobilec/trunk/src/agent_return_data.c File Reference

```
#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)

8.5.1 Function Documentation

8.5.1.1 [interpreter_variable_data_p interpreter_variable_data_Copy](#) (interpreter_variable_data_p src)

Definition at line 225 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](#), and [interpreter_variable_data_s::size](#).

8.5.1.2 [int interpreter_variable_data_Destroy](#) (interpreter_variable_data_p agent_variable_data)

Definition at line 206 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\(\)](#).

8.5.1.3 [interpreter_variable_data_p interpreter_variable_data_Initialize](#) (agent_p agent, const char * varname)

Definition at line 135 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](#), and [size](#).

Referenced by [agent_RunChScriptThread\(\)](#).

8.5.1.4 [interpreter_variable_data_p interpreter_variable_data_InitializeFromAgent](#) (agent_p agent)

Definition at line 51 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, agent_datastate_s::task_progress, agent_datastate_s::tasks, and agent_task_s::var_name.

Referenced by agent_RunChScriptThread().

8.5.1.5 interpreter_variable_data_p interpreter_variable_data_New (void)

Definition at line 36 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(), and interpreter_variable_data_Copy().

8.6 /home/dko/projects/mobilec/trunk/src/agent_task.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "include/agent_task.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)

8.6.1 Function Documentation

8.6.1.1 [agent_task_p agent_task_Copy](#) ([agent_task_p](#) task)

Definition at line 57 of file `agent_task.c`.

References `agent_task_s::agent_return_data`, `agent_task_s::code_id`, `agent_task_s::init_agent_status`, `agent_task_s::number_of_elements`, `agent_task_s::persistent`, `agent_task_s::server_name`, `agent_task_s::size_of_element_array`, `agent_task_s::task_completed`, and `agent_task_s::var_name`.

Referenced by `agent_datastate_Copy()`.

8.6.1.2 [int agent_task_Destroy](#) ([agent_task_p](#) agent_task)

Definition at line 99 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()`.

8.6.1.3 [agent_task_p agent_task_New](#) (void)

Definition at line 39 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()`, and `MC_ComposeAgent()`.

8.7 /home/dko/projects/mobilec/trunk/src/ams.c File Reference

```
#include "include/ams.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)

8.7.1 Function Documentation

8.7.1.1 int ams_Destroy (ams_p ams)

Definition at line 38 of file ams.c.

References COND_DESTROY, MC_SUCCESS, and MUTEX_DESTROY.

Referenced by mc_platform_Destroy().

8.7.1.2 ams_p ams_Initialize (mc_platform_p mc_platform)

Definition at line 49 of file ams.c.

References CHECK_NULL, COND_INIT, COND_T, MUTEX_INIT, and MUTEX_T.

Referenced by mc_platform_Initialize().

8.7.1.3 int ams_ManageAgentList (ams_p ams)

Definition at line 108 of file ams.c.

References [agent_RunChScript\(\)](#), [agent_s::agent_status](#), [ListSearch\(\)](#), [agent_s::lock](#), MC_AGENT_ACTIVE, MC_AGENT_NEUTRAL, MC_WAIT_CH, MC_WAIT_FINISHED, MC_WAIT_MESSGSEND, [message_Destroy\(\)](#), [message_InitializeFromAgent\(\)](#), [message_New\(\)](#), MUTEX_LOCK, MUTEX_UNLOCK, and [agent_s::orphan](#).

Referenced by [ams_Thread\(\)](#).

8.7.1.4 void ams_Print (ams_p ams)

Definition at line 76 of file ams.c.

References [agent_s::agent_status](#), [agent_s::connect_id](#), [agent_s::id](#), [ListSearch\(\)](#), MUTEX_LOCK, and MUTEX_UNLOCK.

8.7.1.5 void `ams_Start` (`mc_platform_p` *mc_platform*)

Definition at line 192 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()`.

8.7.1.6 void* `ams_Thread` (`void *` *arg*)

Definition at line 222 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`, and `mc_platform_s::quit_lock`.

Referenced by `ams_Start()`.

8.8 /home/dko/projects/mobilec/trunk/src/barrier.c File Reference

```
#include "include/barrier.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](#))

8.8.1 Function Documentation

8.8.1.1 [int barrier_node_Destroy](#) ([barrier_node_p node](#))

Definition at line 62 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\(\)](#).

8.8.1.2 [barrier_node_p barrier_node_Initialize](#) ([int id](#), [int num_registered](#))

Definition at line 37 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\(\)](#).

8.8.1.3 [int barrier_queue_Add](#) ([barrier_queue_p list](#), [barrier_node_p node](#))

Definition at line 79 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\(\)](#).

8.8.1.4 [int barrier_queue_Delete](#) ([int id](#), [barrier_queue_p list](#))

Definition at line 103 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()`.

8.8.1.5 `int barrier_queue_Destroy (barrier_queue_p queue)`

Definition at line 123 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()`.

8.8.1.6 `barrier_node_p barrier_queue_Get (barrier_queue_p list, int id)`

Definition at line 137 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()`.

8.8.1.7 `barrier_queue_p barrier_queue_New (void)`

Definition at line 154 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()`.

8.8.1.8 `barrier_node_p barrier_queue_Pop (barrier_queue_p queue)`

Definition at line 168 of file `barrier.c`.

References `barrier_queue_s::list`, `ListPop()`, and `node`.

Referenced by `barrier_queue_Destroy()`.

8.9 /home/dko/projects/mobilec/trunk/src/cmd_prompt.c File Reference

```
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#include <string.h>
#include "include/cmd_prompt.h"
#include "include/commands.h"
#include "config.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 \(void *arg, mc_platform_p global\)](#)

8.9.1 Function Documentation

8.9.1.1 [int cmd_prompt_Destroy \(cmd_prompt_p *cmd_prompt*\)](#)

Definition at line 60 of file cmd_prompt.c.

References [MC_SUCCESS](#).

Referenced by [mc_platform_Destroy\(\)](#).

8.9.1.2 [cmd_prompt_p cmd_prompt_Initialize \(mc_platform_p *mc_platform*\)](#)

Definition at line 52 of file cmd_prompt.c.

Referenced by [mc_platform_Initialize\(\)](#).

8.9.1.3 void cmd_prompt_Start (mc_platform_p mc_platform)

Definition at line 67 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().

8.9.1.4 void* cmd_prompt_Thread (void * arg)

Definition at line 98 of file cmd_prompt.c.

References command_s::args, dealloc_command(), exec_command(), command_s::index, mc_platform, command_s::num_args, process_command(), and split_string().

Referenced by cmd_prompt_Start().

8.9.1.5 int dealloc_command (command_t * cmd)

Definition at line 225 of file cmd_prompt.c.

Referenced by cmd_prompt_Thread().

8.9.1.6 int exec_command (command_t cmd, mc_platform_p global)

Definition at line 209 of file cmd_prompt.c.

Referenced by cmd_prompt_Thread().

8.9.1.7 int handler_HELP (void * arg, mc_platform_p global)

Definition at line 247 of file cmd_prompt.c.

8.9.1.8 int handler_PRINT_CONNECTLIST (void * arg, mc_platform_p global)

Definition at line 300 of file cmd_prompt.c.

8.9.1.9 int handler_PRINTLIST_AGENTS (void * arg, mc_platform_p global)

Definition at line 312 of file cmd_prompt.c.

8.9.1.10 int handler_PRINTLIST_MESSAGE (void * arg, mc_platform_p global)

Definition at line 306 of file cmd_prompt.c.

8.9.1.11 int handler_QUIT (void * arg, mc_platform_p global)

Definition at line 240 of file cmd_prompt.c.

References mc_platform_Destroy().

8.9.1.12 int handler_SEND (void * *arg*, mc_platform_p *global*)

Definition at line 285 of file cmd_prompt.c.

8.9.1.13 int process_command (command_t * *cmd*)

Definition at line 191 of file cmd_prompt.c.

References command_s::args, command_cmds, command_s::index, and command_s::num_args.

Referenced by cmd_prompt_Thread().

8.9.1.14 int split_string (char * *args*, const char * *buf*)**

Definition at line 143 of file cmd_prompt.c.

References int.

Referenced by cmd_prompt_Thread().

8.10 /home/dko/projects/mobilec/trunk/src/connection.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "include/connection.h"
#include "include/mc_error.h"
```

Functions

- [int connection_Destroy](#) ([connection_p](#) connection)
- [connection_p connection_New](#) (void)
- [connection_p connection_Copy](#) ([connection_p](#) connection)

8.10.1 Function Documentation

8.10.1.1 [connection_p connection_Copy](#) ([connection_p](#) *connection*)

Definition at line 69 of file connection.c.

References [connection_s::addr](#), [connection_s::clientfd](#), [connection_s::connect_id](#), [connection_New\(\)](#), [connection_s::remote_hostname](#), and [connection_s::serverfd](#).

8.10.1.2 [int connection_Destroy](#) ([connection_p](#) *connection*)

Definition at line 39 of file connection.c.

References [connection_s::clientfd](#), [MC_SUCCESS](#), and [connection_s::remote_hostname](#).

Referenced by [acc_Thread\(\)](#).

8.10.1.3 [connection_p connection_New](#) (void)

Definition at line 58 of file connection.c.

Referenced by [connection_Copy\(\)](#).

8.11 /home/dko/projects/mobilec/trunk/src/data_structures.c File Reference

```
#include "include/ap_queue_template.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)
- [AP_QUEUE_STD_DEFN_TEMPLATE](#) (fipa_acl_message_p mail_queue_SearchReceivers(mail_queue_p mail_queue, fipa_acl_message)

8.11.1 Function Documentation

8.11.1.1 [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 76 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](#).

8.11.1.2 [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 37 of file data_structures.c.

References [connection_s::connect_id](#), [ListSearch\(\)](#), [MUTEX_LOCK](#), [MUTEX_UNLOCK](#), [node](#), and [connection_s::remote_hostname](#).

8.11.1.3 [AP_QUEUE_STD_DEFN_TEMPLATE](#) (*fipa_acl_message_p* mail_queue_SearchReceivers(*mail_queue_p* mail_queue, *fipa_acl_message*)

Definition at line 190 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](#).

8.11.1.4 AP_QUEUE_STD_DEFN_TEMPLATE (AP_QUEUE_SEARCH_TEMPLATE(*agent_queue*, AP_QUEUE_SEARCH_TEMPLATE(*agent*)

Definition at line 137 of file data_structures.c.

References `agent_s::id`, `ListSearch()`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `agent_s::name`, and `node`.

8.12 /home/dko/projects/mobilec/trunk/src/df.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include "include/mc_platform.h"
#include "include/df.h"
#include "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)

8.12.1 Define Documentation

8.12.1.1 #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_MessageHandlerThread()`, and `acc_Thread()`.

8.12.2 Function Documentation

8.12.2.1 `int df_Add (struct df_s * df, struct df_node_s * node)`

Definition at line 44 of file `df.c`.

References `ListAdd()`, `MC_SUCCESS`, and `SIGNAL`.

Referenced by `request_handler_REGISTER()`.

8.12.2.2 `int df_AddRequest (struct df_s * df, struct df_request_list_node_s * node)`

Definition at line 60 of file `df.c`.

References `ListAdd()`, and `SIGNAL`.

Referenced by `MC_DeregisterService()`, `MC_RegisterService()`, and `MC_SearchForService()`.

8.12.2.3 `int df_Destroy (df_p df)`

Definition at line 77 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()`.

8.12.2.4 `df_p df_Initialize (mc_platform_p mc_platform)`

Definition at line 95 of file `df.c`.

References `COND_INIT`, `COND_T`, `df_request_list_New()`, `ListInitialize()`, `MUTEX_INIT`, and `MUTEX_T`.

Referenced by `mc_platform_Initialize()`.

8.12.2.5 `int df_node_Destroy (df_node_p df_node)`

Definition at line 405 of file `df.c`.

References `MC_SUCCESS`, and `MUTEX_LOCK`.

Referenced by `df_Destroy()`.

8.12.2.6 `int df_ProcessRequest (struct mc_platform_s * global)`

Definition at line 126 of file `df.c`.

References `mc_platform_s::df`, `df_request_list_Pop()`, `MC_ERR_EMPTY`, and `MC_ERR_INVALID`.

Referenced by `df_Thread()`.

8.12.2.7 int df_request_list_Destroy (df_request_list_p *df_request_list*)

Definition at line 314 of file df.c.

References df_request_list_node_Destroy(), ListPop(), ListTerminate(), MC_SUCCESS, and node.

Referenced by df_Destroy().

8.12.2.8 df_request_list_p df_request_list_New (void)

Definition at line 336 of file df.c.

References CHECK_NULL, COND_INIT, COND_T, ListInitialize(), MUTEX_INIT, and MUTEX_T.

Referenced by df_Initialize().

8.12.2.9 int df_request_list_node_Destroy (df_request_list_node_p *node*)

Definition at line 285 of file df.c.

References COND_DESTROY, MC_SUCCESS, and MUTEX_DESTROY.

Referenced by df_request_list_Destroy(), and MC_SearchForService().

8.12.2.10 df_request_list_node_p df_request_list_node_New (void)

Definition at line 296 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().

8.12.2.11 df_request_list_node_p df_request_list_Pop (df_request_list_p *requests*)

Definition at line 361 of file df.c.

References ListPop(), MUTEX_LOCK, MUTEX_UNLOCK, and node.

Referenced by df_ProcessRequest().

8.12.2.12 int df_request_search_Destroy (df_request_search_p *node*)

Definition at line 392 of file df.c.

References COND_DESTROY, MC_SUCCESS, and MUTEX_DESTROY.

Referenced by MC_SearchForService().

8.12.2.13 df_request_search_p df_request_search_New (void)

Definition at line 377 of file df.c.

References CHECK_NULL, COND_INIT, COND_T, MUTEX_INIT, MUTEX_T, and search.

Referenced by MC_SearchForService().

8.12.2.14 `int df_SearchForService (df_p df, const char * searchstring, char *** agent_names, char *** service_names, int ** agent_ids, int * num_entries)`

Definition at line 170 of file df.c.

References MC_ERR_NOT_FOUND, MC_SUCCESS, MUTEX_LOCK, MUTEX_UNLOCK, listNode_s::next, and listNode_s::node_data.

8.12.2.15 `void df_Start (mc_platform_p mc_platform)`

Definition at line 255 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().

8.12.2.16 `void* df_Thread (void * arg)`

Definition at line 419 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, and mc_platform_s::quit_lock.

Referenced by df_Start().

8.12.2.17 `int request_handler_DEREGISTER (struct mc_platform_s * global, void * data)`

Definition at line 512 of file df.c.

8.12.2.18 `int request_handler_REGISTER (struct mc_platform_s * global, void * data)`

Definition at line 482 of file df.c.

References mc_platform_s::df, and df_Add().

8.12.2.19 `int request_handler_SEARCH (struct mc_platform_s * global, void * data)`

Definition at line 488 of file df.c.

8.12.2.20 `int request_handler_SUBSCRIBE (struct mc_platform_s * global, void * data)`

Definition at line 507 of file df.c.

8.13 /home/dko/projects/mobilec/trunk/src/dynstring.c File Reference

```
#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)

8.13.1 Function Documentation

8.13.1.1 int dynstring_Append (dynstring_t * msg, char * str)

Definition at line 21 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_string_Compose(), fipa_url_Compose(), fipa_url_sequence_Compose(), fipa_word_Compose(), and mtp_http_CreateMessage().

8.13.1.2 int dynstring_Destroy (dynstring_t * *dynstring*)

Definition at line 46 of file dynstring.c.

References dynstring_s::message.

Referenced by fipa_envelope_Compose__from(), MC_AclSend(), and mtp_http_CreateMessage().

8.13.1.3 dynstring_t* dynstring_New (void)

Definition at line 5 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(), and mtp_http_CreateMessage().

8.14 /home/dko/projects/mobilec/trunk/src/fipa_acl.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <strings.h>
#include "include/fipa_acl.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_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_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)

8.14.1 Define Documentation

8.14.1.1 #define FREEMEM(x) if (x != NULL) free(x)

Definition at line 45 of file [fipa_acl.c](#).

Referenced by [fipa_expression_Destroy\(\)](#).

8.14.2 Function Documentation

8.14.2.1 int fipa_acl_Compose (dynstring_t **msg, fipa_acl_message_t *acl)

Definition at line 1182 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_string_Compose\(\)](#), [fipa_word_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().

8.14.2.2 **fipa_acl_message_t* fipa_acl_message_Copy (fipa_acl_message_t * src)**

Definition at line 78 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_word_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.

8.14.2.3 **int fipa_acl_message_Destroy (fipa_acl_message_t * message)**

Definition at line 57 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_word_Destroy(), fipa_acl_message_s::in_reply_to, fipa_acl_message_s::language, 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 acc_Thread(), and MC_AclDestroy().

8.14.2.4 **fipa_acl_message_t* fipa_acl_message_New (void)**

Definition at line 49 of file fipa_acl.c.

Referenced by acc_Thread(), fipa_acl_message_Copy(), fipa_Reply(), and MC_AclNew().

8.14.2.5 **int fipa_acl_Parse (fipa_acl_message_p acl, fipa_message_string_p message)**

Definition at line 455 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_Thread().

8.14.2.6 **int fipa_agent_identifier_Compose (dynstring_t * msg, fipa_agent_identifier_t * id)**

Definition at line 1352 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().

8.14.2.7 fipa_agent_identifier_t* fipa_agent_identifier_Copy (fipa_agent_identifier_t * src)

Definition at line 226 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, and fipa_agent_identifier_s::resolvers.

Referenced by fipa_acl_message_Copy(), fipa_acl_Param_Copy(), fipa_agent_identifier_set_Copy(), and fipa_Reply().

8.14.2.8 int fipa_agent_identifier_Destroy (fipa_agent_identifier_t * id)

Definition at line 214 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().

8.14.2.9 fipa_agent_identifier_t* fipa_agent_identifier_New (void)

Definition at line 206 of file fipa_acl.c.

Referenced by fipa_agent_identifier_Copy(), fipa_envelope_ParseAgentIdentifier(), MC_AclAddReceiver(), MC_AclAddReplyTo(), and MC_AclSetSender().

8.14.2.10 int fipa_agent_identifier_Parse (fipa_agent_identifier_p * aid, fipa_message_string_p message)

Definition at line 982 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().

8.14.2.11 int fipa_agent_identifier_set_Compose (dynstring_t * msg, fipa_agent_identifier_set_t * ids)

Definition at line 1339 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().

8.14.2.12 fipa_agent_identifier_set_t* fipa_agent_identifier_set_Copy (fipa_agent_identifier_set_t * src)

Definition at line 185 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()`.

8.14.2.13 `int fipa_agent_identifier_set_Destroy (fipa_agent_identifier_set_t * idset)`

Definition at line 173 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()`.

8.14.2.14 `fipa_agent_identifier_set_t* fipa_agent_identifier_set_New (void)`

Definition at line 165 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()`.

8.14.2.15 `int fipa_agent_identifier_set_Parse (fipa_agent_identifier_set_p * agent_ids, fipa_message_string_p message)`

Definition at line 1141 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()`.

8.14.2.16 `int fipa_CheckNextToken (const fipa_message_string_p message, const char * token)`

Definition at line 644 of file `fipa_acl.c`.

References `fipa_message_string_s::parse`.

Referenced by `fipa_expression_Parse()`.

8.14.2.17 `int fipa_DateTime_Compose (dynstring_t * msg, fipa_DateTime_t * date)`

Definition at line 1425 of file `fipa_acl.c`.

References `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()`.

8.14.2.18 `fipa_DateTime_t* fipa_DateTime_Copy (fipa_DateTime_t * src)`

Definition at line 389 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()`.

8.14.2.19 `int fipa_DateTime_Destroy (fipa_DateTime_t * dt)`

Definition at line 382 of file `fipa_acl.c`.

Referenced by `fipa_acl_envelope_Received_Destroy()`, `fipa_acl_message_Destroy()`, `fipa_acl_Param_Destroy()`, and `fipa_expression_Destroy()`.

8.14.2.20 `fipa_DateTime_t* fipa_DateTime_New (void)`

Definition at line 374 of file `fipa_acl.c`.

Referenced by `fipa_DateTime_Copy()`.

8.14.2.21 `int fipa_datetime_Parse (fipa_DateTime_p * datetime, fipa_message_string_p message)`

Definition at line 799 of file `fipa_acl.c`.

References `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()`.

8.14.2.22 `int fipa_expression_Compose (dynstring_t * msg, fipa_expression_t * expr)`

Definition at line 1378 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()`.

8.14.2.23 `fipa_expression_t* fipa_expression_Copy (fipa_expression_t * src)`

Definition at line 277 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()`, and `fipa_expression_Copy()`.

8.14.2.24 `int fipa_expression_Destroy (fipa_expression_t * expr)`

Definition at line 246 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()`.

8.14.2.25 `fipa_expression_t* fipa_expression_New (void)`

Definition at line 238 of file `fipa_acl.c`.

Referenced by `fipa_expression_Copy()`.

8.14.2.26 `int fipa_expression_Parse (fipa_expression_t ** expression, fipa_message_string_p message)`

Definition at line 663 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()`.

8.14.2.27 `int fipa_GetAtom (fipa_message_string_p message, char expected_atom)`

Definition at line 579 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()`.

8.14.2.28 `int fipa_GetNextWord (char ** word, const fipa_message_string_p message)`

Definition at line 714 of file `fipa_acl.c`.

References `ERR`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

8.14.2.29 `int fipa_GetWholeToken (char ** word, fipa_message_string_p message)`

Definition at line 762 of file `fipa_acl.c`.

References `MC_SUCCESS`, and `fipa_message_string_s::parse`.

Referenced by `fipa_datetime_Parse()`.

8.14.2.30 int fipa_message_parameter_Parse (fipa_acl_message_p *acl*, fipa_message_string_p *message*)

Definition at line 475 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_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().

8.14.2.31 fipa_message_string_t* fipa_message_string_Copy (fipa_message_string_t * *src*)

Definition at line 119 of file fipa_acl.c.

References fipa_message_string_s::message, and fipa_message_string_s::parse.

8.14.2.32 int fipa_message_string_Destroy (fipa_message_string_t * *message*)

Definition at line 109 of file fipa_acl.c.

References fipa_message_string_s::message.

Referenced by acc_Thread().

8.14.2.33 fipa_message_string_t* fipa_message_string_New (void)

Definition at line 101 of file fipa_acl.c.

Referenced by acc_Thread().

8.14.2.34 int fipa_message_type_Parse (enum fipa_performative_e * *performative*, fipa_message_string_p *message*)

Definition at line 517 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().

8.14.2.35 int fipa_number_Compose (dynstring_t * *msg*, fipa_number_t * *number*)

Definition at line 1453 of file fipa_acl.c.

References dynstring_Append(), and fipa_number_s::str.

Referenced by `fipa_expression_Compose()`.

8.14.2.36 `fipa_number_t* fipa_number_Copy (fipa_number_t * src)`

Definition at line 445 of file `fipa_acl.c`.

References `fipa_number_New()`, and `fipa_number_s::str`.

Referenced by `fipa_expression_Copy()`.

8.14.2.37 `int fipa_number_Destroy (fipa_number_t * number)`

Definition at line 435 of file `fipa_acl.c`.

References `fipa_number_s::str`.

Referenced by `fipa_expression_Destroy()`.

8.14.2.38 `fipa_number_t* fipa_number_New (void)`

Definition at line 427 of file `fipa_acl.c`.

Referenced by `fipa_number_Copy()`.

8.14.2.39 `int fipa_performative_Compose (dynstring_t * msg, enum fipa_performative_e performative)`

Definition at line 1251 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()`.

8.14.2.40 `struct fipa_acl_message_s* fipa_Reply (struct fipa_acl_message_s * acl)` [read]

Definition at line 1461 of file `fipa_acl.c`.

References `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_agent_identifier_set_s::num`, `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()`.

8.14.2.41 `int fipa_string_Compose (dynstring_t * msg, fipa_string_t * string)`

Definition at line 1416 of file `fipa_acl.c`.

References `fipa_string_s::content`, and `dynstring_Append()`.

Referenced by fipa_acl_Compose(), and fipa_expression_Compose().

8.14.2.42 fipa_string_t* fipa_string_Copy (fipa_string_t * src)

Definition at line 364 of file fipa_acl.c.

References fipa_string_s::content, and fipa_string_New().

Referenced by fipa_acl_message_Copy(), and fipa_expression_Copy().

8.14.2.43 int fipa_string_Destroy (fipa_string_t * str)

Definition at line 354 of file fipa_acl.c.

References fipa_string_s::content.

Referenced by fipa_acl_message_Destroy(), fipa_expression_Destroy(), and MC_AclSetContent().

8.14.2.44 fipa_string_t* fipa_string_New (void)

Definition at line 346 of file fipa_acl.c.

Referenced by fipa_string_Copy(), and MC_AclSetContent().

8.14.2.45 int fipa_string_Parse (fipa_string_p * fipa_string, fipa_message_string_p message)

Definition at line 937 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().

8.14.2.46 int fipa_url_Compose (dynstring_t * msg, fipa_url_t * url)

Definition at line 1445 of file fipa_acl.c.

References dynstring_Append(), and fipa_url_s::str.

Referenced by fipa_url_sequence_Compose().

8.14.2.47 fipa_url_t* fipa_url_Copy (fipa_url_t * src)

Definition at line 417 of file fipa_acl.c.

References fipa_url_New(), and fipa_url_s::str.

Referenced by fipa_acl_envelope_Received_Copy(), and fipa_url_sequence_Copy().

8.14.2.48 int fipa_url_Destroy (fipa_url_t * url)

Definition at line 407 of file fipa_acl.c.

References fipa_url_s::str.

Referenced by fipa_acl_envelope_Received_Destroy(), and fipa_url_sequence_Destroy().

8.14.2.49 fipa_url_t* fipa_url_New (void)

Definition at line 399 of file fipa_acl.c.

Referenced by fipa_envelope_HandleReceived(), fipa_envelope_ParseAddresses(), fipa_url_Copy(), MC_AclAddReceiver(), MC_AclAddReplyTo(), and MC_AclSetSender().

8.14.2.50 int fipa_url_Parse (fipa_url_p * url, fipa_message_string_p message)

Definition at line 1121 of file fipa_acl.c.

References fipa_word_s::content, fipa_word_Destroy(), and fipa_word_Parse().

Referenced by fipa_url_sequence_Parse().

8.14.2.51 int fipa_url_sequence_Compose (dynstring_t * msg, fipa_url_sequence_t * urls)

Definition at line 1326 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().

8.14.2.52 fipa_url_sequence_t* fipa_url_sequence_Copy (fipa_url_sequence_t * src)

Definition at line 149 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().

8.14.2.53 int fipa_url_sequence_Destroy (fipa_url_sequence_t * sequence)

Definition at line 137 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().

8.14.2.54 fipa_url_sequence_t* fipa_url_sequence_New (void)

Definition at line 129 of file fipa_acl.c.

Referenced by fipa_url_sequence_Copy(), fipa_url_sequence_Parse(), MC_AclAddReceiver(), MC_AclAddReplyTo(), and MC_AclSetSender().

8.14.2.55 int fipa_url_sequence_Parse (fipa_url_sequence_p * urls, fipa_message_string_p message)

Definition at line 1089 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().

8.14.2.56 int fipa_word_Compose (dynstring_t * msg, fipa_word_t * word)

Definition at line 1408 of file fipa_acl.c.

References fipa_word_s::content, and dynstring_Append().

Referenced by fipa_acl_Compose(), and fipa_expression_Compose().

8.14.2.57 fipa_word_t* fipa_word_Copy (fipa_word_t * src)

Definition at line 336 of file fipa_acl.c.

References fipa_word_s::content, and fipa_word_New().

Referenced by fipa_acl_message_Copy(), and fipa_expression_Copy().

8.14.2.58 int fipa_word_Destroy (fipa_word_t * word)

Definition at line 326 of file fipa_acl.c.

References fipa_word_s::content.

Referenced by fipa_acl_message_Destroy(), fipa_agent_identifier_Parse(), fipa_expression_Destroy(), fipa_message_parameter_Parse(), fipa_message_type_Parse(), fipa_url_Parse(), and fipa_url_sequence_Parse().

8.14.2.59 fipa_word_t* fipa_word_New (void)

Definition at line 318 of file fipa_acl.c.

Referenced by fipa_word_Copy().

8.14.2.60 int fipa_word_Parse (fipa_word_t ** word, fipa_message_string_p message)

Definition at line 602 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_url_Parse(), and fipa_url_sequence_Parse().

8.15 /home/dko/projects/mobilec/trunk/src/fipa_envelope.c File Reference

```
#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)

8.15.1 Function Documentation

8.15.1.1 `fipa_acl_envelope_t* fipa_acl_envelope_Copy` (`fipa_acl_envelope_t` * *envelope*)

Definition at line 142 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`.

8.15.1.2 `int fipa_acl_envelope_Destroy` (`fipa_acl_envelope_t` * *envelope*)

Definition at line 129 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_Thread()`.

8.15.1.3 `fipa_acl_envelope_t* fipa_acl_envelope_New` (void)

Definition at line 121 of file `fipa_envelope.c`.

Referenced by `acc_Thread()`, and `fipa_acl_envelope_Copy()`.

8.15.1.4 `fipa_acl_envelope_Received_t* fipa_acl_envelope_Received_Copy` (`fipa_acl_envelope_Received_t` * *received*)

Definition at line 61 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`, and `fipa_acl_envelope_Received_s::received_via`.

Referenced by `fipa_acl_Param_Copy()`.

8.15.1.5 `int fipa_acl_envelope_Received_Destroy` (`fipa_acl_envelope_Received_t` * *received*)

Definition at line 49 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()`.

8.15.1.6 `fipa_acl_envelope_Received_t* fipa_acl_envelope_Received_New (void)`

Definition at line 40 of file `fipa_envelope.c`.

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

8.15.1.7 `fipa_acl_Param_t* fipa_acl_Param_Copy (fipa_acl_Param_t * param)`

Definition at line 102 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`, and `fipa_acl_Param_s::to`.

Referenced by `fipa_acl_envelope_Copy()`.

8.15.1.8 `int fipa_acl_Param_Destroy (fipa_acl_Param_t * param)`

Definition at line 85 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()`.

8.15.1.9 `fipa_acl_Param_t* fipa_acl_Param_New (void)`

Definition at line 77 of file `fipa_envelope.c`.

Referenced by `fipa_acl_Param_Copy()`, and `fipa_envelope_HandleTo()`.

8.15.1.10 `char* fipa_envelope_Compose (fipa_acl_message_t * fipa_acl)`

Definition at line 861 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()`.

8.15.1.11 mxml_node_t* fipa_envelope_Compose__acl_representation (fipa_acl_message_t * *fipa_acl*)

Definition at line 1067 of file fipa_envelope.c.

References mxmlNewElement(), mxmlNewText(), and node.

Referenced by fipa_envelope_Compose__params().

8.15.1.12 mxml_node_t* fipa_envelope_Compose__date (fipa_acl_message_t * *fipa_acl*)

Definition at line 1095 of file fipa_envelope.c.

References mxmlNewElement(), mxmlNewText(), and node.

Referenced by fipa_envelope_Compose__params().

8.15.1.13 mxml_node_t* fipa_envelope_Compose__envelope (fipa_acl_message_t * *fipa_acl*)

Definition at line 886 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().

8.15.1.14 mxml_node_t* fipa_envelope_Compose__from (fipa_acl_message_t * *fipa_acl*)

Definition at line 1006 of file fipa_envelope.c.

References fipa_agent_identifier_s::addresses, 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().

8.15.1.15 mxml_node_t* fipa_envelope_Compose__params (fipa_acl_message_t * *fipa_acl*)

Definition at line 906 of file fipa_envelope.c.

References fipa_envelope_Compose__acl_representation(), fipa_envelope_Compose__date(), fipa_envelope_Compose__from(), 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().

8.15.1.16 mxml_node_t* fipa_envelope_Compose__payload_encoding (fipa_acl_message_t * *fipa_acl*)

Definition at line 1081 of file fipa_envelope.c.

References mxmlNewElement(), mxmlNewText(), and node.

Referenced by fipa_envelope_Compose__params().

8.15.1.17 mxml_node_t* fipa_envelope_Compose__to (fipa_acl_message_t * fipa_acl)

Definition at line 960 of file fipa_envelope.c.

References fipa_agent_identifier_s::addresses, fipa_agent_identifier_set_s::fipa_agent_identifiers, mxml-NewElement(), 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().

8.15.1.18 int fipa_envelope_HandleAclRepresentation (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)

Definition at line 373 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().

8.15.1.19 int fipa_envelope_HandleComments (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)

Definition at line 349 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().

8.15.1.20 int fipa_envelope_HandleDate (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)

Definition at line 444 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().

8.15.1.21 int fipa_envelope_HandleEnvelope (struct fipa_acl_envelope_s * envelope, mxml_node_t * node)

Definition at line 170 of file fipa_envelope.c.

References fipa_envelope_HandleParams(), MC_ERR_PARSE, MXML_DESCEND_FIRST, and mxmlFindElement().

Referenced by fipa_envelope_Parse().

8.15.1.22 `int fipa_envelope_HandleFrom (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)`

Definition at line 320 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().

8.15.1.23 `int fipa_envelope_HandleIntendedReceiver (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)`

Definition at line 474 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().

8.15.1.24 `int fipa_envelope_HandleParams (struct fipa_acl_envelope_s * envelope, mxml_node_t * node)`

Definition at line 190 of file fipa_envelope.c.

References 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().

8.15.1.25 `int fipa_envelope_HandlePayloadEncoding (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)`

Definition at line 420 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().

8.15.1.26 `int fipa_envelope_HandlePayloadLength (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)`

Definition at line 397 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().

8.15.1.27 **int fipa_envelope_HandleReceived** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 552 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().

8.15.1.28 **int fipa_envelope_HandleTo** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 242 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().

8.15.1.29 **int fipa_envelope_Parse** (struct fipa_acl_envelope_s * *envelope*, const char * *message*)

Definition at line 157 of file fipa_envelope.c.

References fipa_envelope_HandleEnvelope(), MXML_NO_CALLBACK, mxmlDelete(), and mxmlLoadString().

Referenced by acc_Thread().

8.15.1.30 **int fipa_envelope_ParseAddresses** (struct fipa_agent_identifier_s * *aid*, mxml_node_t * *addresses_node*)

Definition at line 729 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().

8.15.1.31 **int fipa_envelope_ParseAgentIdentifier** (struct fipa_agent_identifier_s ** *aid*, mxml_node_t * *agent_identifier_node*)

Definition at line 679 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()`.

8.15.1.32 `int fipa_envelope_ParseResolvers (struct fipa_agent_identifier_s * aid, mxml_node_t * resolvers_node)`

Definition at line 796 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()`.

8.16 /home/dko/projects/mobilec/trunk/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](#); [int](#) [waiting](#); [MUTEX_T](#) *[waiting_lock](#); [COND_T](#) *[waiting_cond](#);) **STRUCT**([listen_thread_arg](#)
- [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](#) * [listen_Thread](#) ([void](#) *[arg](#))

Variables

- [u_long](#) [port](#)
- struct [sockaddr_in](#) * [addr](#)
- unsigned long [int](#) [client_fd](#)
- unsigned long [int](#) [server_fd](#)

8.16.1 Function Documentation

8.16.1.1 [int](#) [acc_Destroy](#) ([acc_p](#) [acc](#))

Definition at line 79 of file [acc.c](#).

References [MC_SUCCESS](#).

Referenced by [mc_platform_Destroy\(\)](#).

8.16.1.2 [acc_p](#) [acc_Initialize](#) (struct [mc_platform_s](#) * [mc_platform](#))

Definition at line 63 of file [acc.c](#).

References [COND_INIT](#), [COND_T](#), [MUTEX_INIT](#), and [MUTEX_T](#).

Referenced by [mc_platform_Initialize\(\)](#).

8.16.1.3 [void](#)* [acc_MessageHandlerThread](#) ([void](#) * [arg](#))

Definition at line 91 of file [acc.c](#).

References `agent_Initialize()`, `mc_platform_s::agent_queue`, `agent_s::agent_status`, `AGENT_UPDATE`, `mc_platform_s::ams`, `mc_platform_s::asm_message_queue`, `CANCEL`, `COND_BROADCAST`, `COND_SIGNAL`, `COND_WAIT`, `agent_s::datastate`, `ENCRYPTED_DATA`, `ENCRYPTION_INITIALIZE`, `FIPA_ACL`, `message_s::from_address`, `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`, `REQUEST_ENCRYPTION_INITIALIZE`, `RETURN_MSG`, `SUBSCRIBE`, `message_s::to_address`, and `WARN`.

Referenced by `acc_Start()`.

8.16.1.4 void acc_Start (struct mc_platform_s * mc_platform)

8.16.1.5 void* acc_Thread (void * arg)

Definition at line 340 of file `acc.c`.

References `agent_mailbox_Post()`, `mc_platform_s::agent_queue`, `AGENT_UPDATE`, `mc_platform_s::asm_message_queue`, `CANCEL`, `COND_BROADCAST`, `COND_WAIT`, `connection_Destroy()`, `mc_platform_s::connection_queue`, `mtp_http_s::content`, `mtp_http_content_s::data`, `ENCRYPTED_DATA`, `ENCRYPTION_INITIALIZE`, `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()`, `mc_platform_s::giant`, `mc_platform_s::giant_cond`, `mc_platform_s::giant_lock`, `mtp_http_s::http_performative`, `HTTP_POST`, `HTTP_PUT`, `agent_s::mailbox`, `mc_platform`, `MC_RECV_CONNECTION`, `mc_platform_s::MC_signal`, `mc_platform_s::MC_signal_cond`, `mc_platform_s::MC_signal_lock`, `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()`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `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`, `QUER_IF`, `QUER_REF`, `mc_platform_s::quit`, `mc_platform_s::quit_lock`, `RELAY`, `REQUEST`, `REQUEST_ENCRYPTION_INITIALIZE`, `RETURN_MSG`, `SUBSCRIBE`, `mtp_http_s::target`, `fipa_acl_Param_s::to`, `WARN`, and `message_s::xml_root`.

Referenced by `acc_Start()`.

8.16.1.6 void* listen_Thread (void * arg)

Definition at line 595 of file `acc.c`.

References `mc_platform_s::acc`, `BACKLOG`, `CHECK_NULL`, `COND_BROADCAST`, `mc_platform_s::connection_queue`, `mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `mc_platform_s::port`.

Referenced by `acc_Start()`.

8.16.1.7 **STRUCT** (acc, struct mc_platform_s *mc_platform;THREAD_T thread;THREAD_T message_handler_thread;THREAD_T listen_thread;int waiting;MUTEX_T *waiting_lock;COND_T *waiting_cond;)

8.16.2 Variable Documentation

8.16.2.1 struct sockaddr_in* addr

Definition at line 61 of file acc.h.

8.16.2.2 unsigned long int client_fd

Definition at line 62 of file acc.h.

Referenced by main().

8.16.2.3 u_long port

Definition at line 60 of file acc.h.

Referenced by MC_AclSend(), and message_Send().

8.16.2.4 unsigned long int server_fd

Definition at line 63 of file acc.h.

Referenced by main().

8.17 /home/dko/projects/mobilec/trunk/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](#)

Functions

- [int agent_AddPersistentVariable](#) ([agent_p](#) agent, [int](#) task_num, const char *var_name)
- [agent_p agent_New](#) (void)
- [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)

8.17.1 Function Documentation

8.17.1.1 [int agent_AddPersistentVariable](#) ([agent_p](#) agent, [int](#) task_num, const char * var_name)

Definition at line 46 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](#), [agent_datastate_s::task_progress](#), and [agent_datastate_s::tasks](#).

8.17.1.2 [agent_p agent_Copy](#) (const [agent_p](#) agent)

Definition at line 128 of file agent.c.

References [agent_datastate_Copy\(\)](#), [agent_s::agent_interp](#), [agent_s::agent_persistent](#), [agent_s::agent_status](#), [agent_s::agent_thread](#), [agent_s::agent_type](#), [agent_s::arrival_time](#), [agent_s::datastate](#), [agent_s::home](#), [agent_s::home_port](#), [agent_s::id](#), [agent_s::lock](#), [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\(\)](#).

8.17.1.3 int agent_Destroy (agent_p agent)

Definition at line 319 of file agent.c.

References agent_datastate_Destroy(), agent_s::agent_interp, agent_mailbox_Destroy(), agent_s::agent_status, agent_s::agent_thread, agent_s::datastate, agent_s::home, agent_s::lock, agent_s::mailbox, MC_AGENT_NEUTRAL, MC_SUCCESS, MUTEX_DESTROY, MUTEX_LOCK, agent_s::name, agent_s::owner, and agent_s::run_lock.

Referenced by agent_Initialize().

8.17.1.4 agent_p agent_Initialize (struct mc_platform_s * mc_platform, message_p message, int id)

Definition at line 213 of file agent.c.

References 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, 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::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, RETURN_MSG, agent_s::run_lock, agent_datastate_s::xml_agent_root, message_s::xml_payload, message_s::xml_root, and agent_datastate_s::xml_root.

Referenced by acc_MessageHandlerThread().

8.17.1.5 agent_p agent_New (void)

Definition at line 190 of file agent.c.

References agent_s::lock, MUTEX_INIT, MUTEX_NEW, and agent_s::run_lock.

Referenced by MC_ComposeAgent().

8.17.1.6 void agent_RunChScript (MCAgent_t agent, struct mc_platform_s * global)**8.17.1.7 void* agent_RunChScriptThread (void * agent)**

Definition at line 394 of file agent.c.

References agent_s::agent_interp, mc_platform_s::ams, COND_SIGNAL, agent_s::datastate, agent_s::id, mc_platform_s::interp_options, interpreter_variable_data_Destroy(), interpreter_variable_data_Initialize(), interpreter_variable_data_InitializeFromAgent(), MC_AclAddReceiver_chdl(), MC_AclAddReplyTo_chdl(), MC_AclDestroy_chdl(), MC_AclNew_chdl(), MC_AclPost_chdl(), MC_AclReply_chdl(), MC_AclRetrieve_chdl(), MC_AclSend_chdl(), MC_AclSetContent_chdl(), MC_AclSetPerformative_chdl(), MC_AclSetSender_chdl(), MC_AclWaitRetrieve_chdl(), MC_AddAgent_chdl(), MC_AGENT_NEUTRAL, MC_AgentVariableRetrieve_chdl(), MC_AgentVariableSave_chdl(), MC_Barrier_chdl(), MC_BarrierDelete_chdl(), MC_BarrierInit_chdl(), MC_CallAgentFunc_chdl(), MC_ComposeAgent_chdl(), MC_CondBroadcast_chdl(), MC_CondReset_chdl(), MC_CondSignal_chdl(), MC_CondWait_chdl(), MC_DeleteAgent_chdl(), MC_DeregisterService_chdl(), MC_DestroyServiceSearchResult_chdl(), MC_End_chdl(), MC_EXEC_AGENT, 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_MutexLock_chdl(), MC_MutexUnlock_chdl(),

agent_s::mc_platform, mc_platform, MC_PrintAgentCode_chdl(), MC_RegisterService_chdl(), MC_ResumeAgency_chdl(), MC_RetrieveAgent_chdl(), MC_RetrieveAgentCode_chdl(), MC_RETURN_AGENT, 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_platform_s::MC_signal, mc_platform_s::MC_signal_cond, mc_platform_s::MC_signal_lock, MC_SyncDelete_chdl(), MC_SyncInit_chdl(), MC_TerminateAgent_chdl(), MC_WAIT_FINISHED, MC_WAIT_MESSGSEND, MUTEX_LOCK, MUTEX_UNLOCK, agent_s::name, SIGNAL, and agent_datastate_s::task_progress.

Referenced by agent_RunChScript().

8.18 /home/dko/projects/mobilec/trunk/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*)

8.18.1 Typedef Documentation

8.18.1.1 typedef [agent_datastate_t](#)* [agent_datastate_p](#)

Definition at line 63 of file [agent_datastate.h](#).

8.18.1.2 typedef struct [agent_datastate_s](#) [agent_datastate_t](#)

8.18.2 Function Documentation

8.18.2.1 [agent_datastate_p](#) [agent_datastate_Copy](#) (const [agent_datastate_p](#) *datastate*)

Definition at line 38 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\(\)](#).

8.18.2.2 int [agent_datastate_Destroy](#) ([agent_datastate_p](#) *agent_datastate*)

Definition at line 126 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().

8.18.2.3 agent_datastate_p agent_datastate_New (void)

Definition at line 106 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::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_ComposeAgent().

8.19 /home/dko/projects/mobilec/trunk/src/include/agent_lib.h File Reference

Functions

- EXPORTCH [int MC_AclDestroy_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclNew_chdl](#) (void *varg)
- EXPORTCH void * [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 void * [MC_AclSetPerformative_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclSetSender_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclAddReceiver_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclAddReplyTo_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclSetContent_chdl](#) (void *varg)
- EXPORTCH [int MC_AddAgent_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 [int MC_ComposeAgent_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_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_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 [MCAgent_t MC_RetrieveAgent_chdl](#) (void *varg)
- EXPORTCH char * [MC_RetrieveAgentCode_chdl](#) (void *varg)
- EXPORTCH char * [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_GetAgentStatus_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)

8.19.1 Function Documentation

8.19.1.1 EXPORTCH void* MC_AclAddReceiver_chdl (void * *varg*)

Definition at line 1836 of file libmc.c.

References [MC_AclAddReceiver\(\)](#).

Referenced by [agent_RunChScriptThread\(\)](#).

8.19.1.2 EXPORTCH void* MC_AclAddReplyTo_chdl (void * *varg*)

Definition at line 1856 of file libmc.c.

References [MC_AclAddReplyTo\(\)](#).

Referenced by [agent_RunChScriptThread\(\)](#).

8.19.1.3 EXPORTCH int MC_AclDestroy_chdl (void * *varg*)

Definition at line 1683 of file libmc.c.

References [MC_AclDestroy\(\)](#).

Referenced by [agent_RunChScriptThread\(\)](#).

8.19.1.4 EXPORTCH void* MC_AclNew_chdl (void * *varg*)

Definition at line 1698 of file libmc.c.

References [MC_AclNew\(\)](#).

Referenced by [agent_RunChScriptThread\(\)](#).

8.19.1.5 EXPORTCH void* MC_AclPost_chdl (void * *varg*)

Definition at line 1706 of file libmc.c.

References [MC_AclPost\(\)](#).

Referenced by [agent_RunChScriptThread\(\)](#).

8.19.1.6 EXPORTCH void* MC_AclReply_chdl (void * *varg*)

Definition at line 1724 of file libmc.c.

References MC_AclReply().

Referenced by agent_RunChScriptThread().

8.19.1.7 EXPORTCH void* MC_AclRetrieve_chdl (void * *varg*)

Definition at line 1740 of file libmc.c.

References MC_AclRetrieve().

Referenced by agent_RunChScriptThread().

8.19.1.8 EXPORTCH int MC_AclSend_chdl (void * *varg*)

Definition at line 1756 of file libmc.c.

References CHECK_NULL, MC_AclSend(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.9 EXPORTCH void* MC_AclSetContent_chdl (void * *varg*)

Definition at line 1876 of file libmc.c.

References fipa_acl_message_s::content, and MC_AclSetContent().

Referenced by agent_RunChScriptThread().

8.19.1.10 EXPORTCH void* MC_AclSetPerformative_chdl (void * *varg*)

Definition at line 1798 of file libmc.c.

References MC_AclSetPerformative(), and fipa_acl_message_s::performative.

Referenced by agent_RunChScriptThread().

8.19.1.11 EXPORTCH void* MC_AclSetSender_chdl (void * *varg*)

Definition at line 1816 of file libmc.c.

References MC_AclSetSender().

Referenced by agent_RunChScriptThread().

8.19.1.12 EXPORTCH void* MC_AclWaitRetrieve_chdl (void * *varg*)

Definition at line 1780 of file libmc.c.

References MC_AclWaitRetrieve().

Referenced by agent_RunChScriptThread().

8.19.1.13 EXPORTCH int MC_AddAgent_chdl (void * *varg*)

Definition at line 1896 of file libmc.c.

References CHECK_NULL, MC_AddAgent(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.14 EXPORTCH const void* MC_AgentVariableRetrieve_chdl (void * *varg*)

Definition at line 1920 of file libmc.c.

References MC_AgentVariableRetrieve().

Referenced by agent_RunChScriptThread().

8.19.1.15 EXPORTCH int MC_AgentVariableSave_chdl (void * *varg*)

Definition at line 1944 of file libmc.c.

References MC_AgentVariableSave().

Referenced by agent_RunChScriptThread().

8.19.1.16 EXPORTCH int MC_Barrier_chdl (void * *varg*)

Definition at line 1996 of file libmc.c.

References CHECK_NULL, MC_Barrier(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.17 EXPORTCH int MC_BarrierDelete_chdl (void * *varg*)

Definition at line 2019 of file libmc.c.

References CHECK_NULL, MC_BarrierDelete(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.18 EXPORTCH int MC_BarrierInit_chdl (void * *varg*)

Definition at line 2042 of file libmc.c.

References CHECK_NULL, MC_BarrierInit(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.19 EXPORTCH int MC_CallAgentFunc_chdl (void * *varg*)

Definition at line 1966 of file libmc.c.

References MC_CallAgentFunc().

Referenced by agent_RunChScriptThread().

8.19.1.20 EXPORTCH int MC_ComposeAgent_chdl (void * *varg*)

Definition at line 2090 of file libmc.c.

References MC_ComposeAgent().

Referenced by agent_RunChScriptThread().

8.19.1.21 EXPORTCH int MC_CondBroadcast_chdl (void * *varg*)

Definition at line 2067 of file libmc.c.

References CHECK_NULL, MC_CondBroadcast(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.22 EXPORTCH int MC_CondReset_chdl (void * *varg*)

Definition at line 2153 of file libmc.c.

References CHECK_NULL, MC_CondReset(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.23 EXPORTCH int MC_CondSignal_chdl (void * *varg*)

Definition at line 2130 of file libmc.c.

References CHECK_NULL, MC_CondSignal(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.24 EXPORTCH int MC_CondWait_chdl (void * *varg*)

Definition at line 2176 of file libmc.c.

References CHECK_NULL, MC_CondWait(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.25 EXPORTCH int MC_DeleteAgent_chdl (void * *varg*)

Definition at line 2198 of file libmc.c.

References MC_DeleteAgent().

Referenced by agent_RunChScriptThread().

8.19.1.26 EXPORTCH int MC_DeregisterService_chdl (void * *varg*)

Definition at line 2240 of file libmc.c.

References CHECK_NULL, MC_DeregisterService(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.27 EXPORTCH int MC_DestroyServiceSearchResult_chdl (void * *varg*)

Definition at line 2213 of file libmc.c.

References MC_DestroyServiceSearchResult().

Referenced by agent_RunChScriptThread().

8.19.1.28 EXPORTCH int MC_End_chdl (void * *varg*)

Definition at line 2267 of file libmc.c.

References CHECK_NULL, MC_End(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.29 EXPORTCH MCAgent_t MC_FindAgentByID_chdl (void * *varg*)

Definition at line 2284 of file libmc.c.

References CHECK_NULL, MC_FindAgentByID(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.30 EXPORTCH MCAgent_t MC_FindAgentByName_chdl (void * *varg*)

Definition at line 2307 of file libmc.c.

References CHECK_NULL, MC_FindAgentByName(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.31 EXPORTCH int MC_GetAgentID_chdl (void * *varg*)

Definition at line 2352 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.32 EXPORTCH char* MC_GetAgentName_chdl (void * *varg*)

Definition at line 2368 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.33 EXPORTCH int MC_GetAgentNumTasks_chdl (void * *varg*)

Definition at line 2384 of file libmc.c.

8.19.1.34 EXPORTCH int MC_GetAgentStatus_chdl (void * *varg*)

Definition at line 2400 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.35 EXPORTCH char* MC_GetAgentXMLString_chdl (void * *varg*)

Definition at line 2416 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.36 EXPORTCH int MC_GetTimeOfDay_chdl (void * *varg*)

Definition at line 2432 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.37 EXPORTCH int MC_HaltAgency_chdl (void * *varg*)

Definition at line 2447 of file libmc.c.

References CHECK_NULL, MC_HaltAgency(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.19.1.38 EXPORTCH int MC_MutexLock_chdl (void * *varg*)

Definition at line 2465 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.39 EXPORTCH int MC_MutexUnlock_chdl (void * *varg*)

Definition at line 2489 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.40 EXPORTCH int MC_PrintAgentCode_chdl (void * *varg*)

Definition at line 2513 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.41 EXPORTCH int MC_RegisterService_chdl (void * *varg*)

Definition at line 2529 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.42 EXPORTCH int MC_ResumeAgency_chdl (void * *varg*)

Definition at line 2566 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.43 EXPORTCH MC_Agent_t MC_RetrieveAgent_chdl (void * *varg*)

Definition at line 2584 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.44 EXPORTCH char* MC_RetrieveAgentCode_chdl (void * *varg*)

Definition at line 2601 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.45 EXPORTCH char* MC_SearchForService_chdl (void * *varg*)

Definition at line 2617 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.46 EXPORTCH int MC_SemaphorePost_chdl (void * *varg*)

Definition at line 2658 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.47 EXPORTCH int MC_SemaphoreWait_chdl (void * *varg*)

Definition at line 2682 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.48 EXPORTCH int MC_SendAgentMigrationMessage_chdl (void * *varg*)

Definition at line 2706 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.49 EXPORTCH int MC_SendAgentMigrationMessageFile_chdl (void * *varg*)

Definition at line 2731 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.50 EXPORTCH int MC_SendSteerCommand_chdl (void * *varg*)

Definition at line 2751 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.51 EXPORTCH int MC_SetAgentStatus_chdl (void * *varg*)

Definition at line 2774 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.52 EXPORTCH int MC_SetDefaultAgentStatus_chdl (void * *varg*)

Definition at line 2792 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.53 EXPORTCH int MC_SyncDelete_chdl (void * *varg*)

Definition at line 2815 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.54 EXPORTCH int MC_SyncInit_chdl (void * *varg*)

Definition at line 2838 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.19.1.55 EXPORTCH int MC_TerminateAgent_chdl (void * *varg*)

Definition at line 2862 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.20 /home/dko/projects/mobilec/trunk/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)

8.20.1 Typedef Documentation

8.20.1.1 typedef [agent_mailbox_t](#)* [agent_mailbox_p](#)

Definition at line 9 of file [agent_mailbox.h](#).

8.20.1.2 typedef struct [agent_mailbox_s](#) [agent_mailbox_t](#)

8.20.2 Function Documentation

8.20.2.1 [agent_mailbox_p](#) [agent_mailbox_Copy](#) ([agent_mailbox_p](#) src)

Definition at line 12 of file [agent_mailbox.c](#).

References [agent_mailbox_New\(\)](#), and [agent_mailbox_s::mail_queue](#).

8.20.2.2 int [agent_mailbox_Destroy](#) ([agent_mailbox_t](#) * mailbox)

Definition at line 20 of file [agent_mailbox.c](#).

References [agent_mailbox_s::mail_queue](#).

Referenced by [agent_Destroy\(\)](#).

8.20.2.3 [agent_mailbox_p](#) [agent_mailbox_New](#) (void)

Definition at line 3 of file [agent_mailbox.c](#).

References [agent_mailbox_s::mail_queue](#).

Referenced by `agent_Initialize()`, and `agent_mailbox_Copy()`.

8.20.2.4 `int agent_mailbox_Post (agent_mailbox_p mailbox, fipa_acl_message_t * message)`

Definition at line 29 of file `agent_mailbox.c`.

References `agent_mailbox_s::mail_queue`.

Referenced by `acc_Thread()`, and `MC_AclPost()`.

8.20.2.5 `fipa_acl_message_t* agent_mailbox_Retrieve (agent_mailbox_p mailbox)`

Definition at line 35 of file `agent_mailbox.c`.

References `agent_mailbox_s::mail_queue`.

Referenced by `agent_mailbox_WaitRetrieve()`, and `MC_AclRetrieve()`.

8.20.2.6 `fipa_acl_message_t* agent_mailbox_WaitRetrieve (agent_mailbox_p mailbox)`

Definition at line 40 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()`.

8.21 /home/dko/projects/mobilec/trunk/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)

8.21.1 Typedef Documentation

8.21.1.1 typedef [agent_task_t](#)* [agent_task_p](#)

Definition at line 58 of file [agent_task.h](#).

8.21.1.2 typedef struct [agent_task_s](#) [agent_task_t](#)

8.21.2 Function Documentation

8.21.2.1 [agent_task_p](#) [agent_task_Copy](#) ([agent_task_p](#) task)

Definition at line 57 of file [agent_task.c](#).

References [agent_task_s::agent_return_data](#), [agent_task_s::code_id](#), [agent_task_s::init_agent_status](#), [agent_task_s::number_of_elements](#), [agent_task_s::persistent](#), [agent_task_s::server_name](#), [agent_task_s::size_of_element_array](#), [agent_task_s::task_completed](#), and [agent_task_s::var_name](#).

Referenced by [agent_datastate_Copy](#)().

8.21.2.2 int [agent_task_Destroy](#) ([agent_task_p](#) agent_task)

Definition at line 99 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](#)().

8.21.2.3 `agent_task_p agent_task_New (void)`

Definition at line 39 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()`, and `MC_ComposeAgent()`.

8.22 /home/dko/projects/mobilec/trunk/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)

8.22.1 Function Documentation

8.22.1.1 int [ams_Destroy](#) (ams_p *ams*)

Definition at line 38 of file [ams.c](#).

References [COND_DESTROY](#), [MC_SUCCESS](#), and [MUTEX_DESTROY](#).

Referenced by [mc_platform_Destroy\(\)](#).

8.22.1.2 int [ams_ManageAgentList](#) (ams_p *ams*)

Definition at line 108 of file [ams.c](#).

References [agent_RunChScript\(\)](#), [agent_s::agent_status](#), [ListSearch\(\)](#), [agent_s::lock](#), [MC_AGENT_ACTIVE](#), [MC_AGENT_NEUTRAL](#), [MC_WAIT_CH](#), [MC_WAIT_FINISHED](#), [MC_WAIT_MESSGSEND](#), [message_Destroy\(\)](#), [message_InitializeFromAgent\(\)](#), [message_New\(\)](#), [MUTEX_LOCK](#), [MUTEX_UNLOCK](#), and [agent_s::orphan](#).

Referenced by [ams_Thread\(\)](#).

8.22.1.3 void [ams_Print](#) (ams_p *ams*)

Definition at line 76 of file [ams.c](#).

References [agent_s::agent_status](#), [agent_s::connect_id](#), [agent_s::id](#), [ListSearch\(\)](#), [MUTEX_LOCK](#), and [MUTEX_UNLOCK](#).

8.22.1.4 int [ams_RemoveFinishedAgents](#) (ams_p *ams*)

8.22.1.5 void [ams_Start](#) (struct [mc_platform_s](#) * *mc_platform*)

8.22.1.6 void* [ams_Thread](#) (void * *arg*)

Definition at line 222 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`, and `mc_platform_s::quit_lock`.

Referenced by `ams_Start()`.

8.22.1.7 **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`;)

8.23 /home/dko/projects/mobilec/trunk/src/include/ap_queue_template.h File Reference

```
#include "macros.h"
#include "mc_error.h"
```

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)

8.23.1 Define Documentation

8.23.1.1 #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##_SearchIndex( name##_p name, int index ); \
int name##_RemoveIndex( name##_p name, int index );
```

Definition at line 37 of file ap_queue_template.h.

8.23.1.2 #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 56 of file ap_queue_template.h.

8.23.1.3 #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 200 of file ap_queue_template.h.

8.23.1.4 #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 170 of file ap_queue_template.h.

8.23.1.5 #define AP_QUEUE_STD_DEFN_TEMPLATE(name, node_type)

Definition at line 59 of file ap_queue_template.h.

8.24 /home/dko/projects/mobilec/trunk/src/include/barrier.h File Reference

```
#include "macros.h"
#include "../mc_list/list.h"
#include "mc_rwlock.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)

8.24.1 Typedef Documentation

8.24.1.1 typedef [barrier_node_t](#)* [barrier_node_p](#)

Definition at line 46 of file barrier.h.

8.24.1.2 typedef struct [barrier_node_s](#) [barrier_node_t](#)

8.24.1.3 typedef [barrier_queue_t](#)* [barrier_queue_p](#)

Definition at line 54 of file barrier.h.

8.24.1.4 typedef struct barrier_queue_s barrier_queue_t

8.24.2 Function Documentation

8.24.2.1 int barrier_node_Destroy (barrier_node_p node)

Definition at line 62 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().

8.24.2.2 barrier_node_p barrier_node_Initialize (int id, int num_registered)

Definition at line 37 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().

8.24.2.3 int barrier_queue_Add (barrier_queue_p list, barrier_node_p node)

Definition at line 79 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().

8.24.2.4 int barrier_queue_Delete (int id, barrier_queue_p list)

Definition at line 103 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().

8.24.2.5 int barrier_queue_Destroy (barrier_queue_p queue)

Definition at line 123 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().

8.24.2.6 barrier_node_p barrier_queue_Get (barrier_queue_p list, int id)

Definition at line 137 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()`.

8.24.2.7 `barrier_queue_p barrier_queue_New (void)`

Definition at line 154 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()`.

8.24.2.8 `barrier_node_p barrier_queue_Pop (barrier_queue_p queue)`

Definition at line 168 of file `barrier.c`.

References `barrier_queue_s::list`, `ListPop()`, and `node`.

Referenced by `barrier_queue_Destroy()`.

8.25 /home/dko/projects/mobilec/trunk/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)

8.25.1 Define Documentation

8.25.1.1 #define CMDLINE_SIZE 80

Definition at line 34 of file [cmd_prompt.h](#).

8.25.2 Typedef Documentation

8.25.2.1 typedef cmd_prompt_t* cmd_prompt_p

Definition at line 41 of file [cmd_prompt.h](#).

8.25.2.2 `typedef struct cmd_prompt_s cmd_prompt_t`

8.25.2.3 `typedef command_t* command_p`

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

8.25.2.4 `typedef struct command_s command_t`

8.25.3 Function Documentation

8.25.3.1 `int cmd_prompt_Destroy (cmd_prompt_p cmd_prompt)`

Definition at line 60 of file `cmd_prompt.c`.

References `MC_SUCCESS`.

Referenced by `mc_platform_Destroy()`.

8.25.3.2 `cmd_prompt_p cmd_prompt_Initialize (struct mc_platform_s * mc_platform)`

8.25.3.3 `void cmd_prompt_Start (struct mc_platform_s * mc_platform)`

8.25.3.4 `void* cmd_prompt_Thread (void * arg)`

Definition at line 98 of file `cmd_prompt.c`.

References `command_s::args`, `dealloc_command()`, `exec_command()`, `command_s::index`, `mc_platform`, `command_s::num_args`, `process_command()`, and `split_string()`.

Referenced by `cmd_prompt_Start()`.

8.25.3.5 `int dealloc_command (command_t * cmd)`

Definition at line 225 of file `cmd_prompt.c`.

Referenced by `cmd_prompt_Thread()`.

8.25.3.6 `int exec_command (command_t cmd, struct mc_platform_s * global)`

8.25.3.7 `int process_command (command_t * cmd)`

Definition at line 191 of file `cmd_prompt.c`.

References `command_s::args`, `command_cmds`, `command_s::index`, and `command_s::num_args`.

Referenced by `cmd_prompt_Thread()`.

8.25.3.8 `int split_string (char *** args, const char * buf)`

Definition at line 143 of file `cmd_prompt.c`.

References `int`.

Referenced by `cmd_prompt_Thread()`.

8.26 /home/dko/projects/mobilec/trunk/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](#) { [COMMAND](#) }

Variables

- [cmd_handler_t](#) [cmd_handlers](#) []
- char * [command_descriptions](#) []
- char * [command_cmds](#) []

8.26.1 Define Documentation

8.26.1.1 #define [COMMAND](#)(name, cmd, description) cmd ,

Definition at line 38 of file [commands.h](#).

8.26.1.2 #define [COMMAND](#)(name, cmd, description) description ,

Definition at line 38 of file [commands.h](#).

8.26.1.3 #define [COMMAND](#)(name, cmd, description) &handler_##name ,

Definition at line 38 of file [commands.h](#).

8.26.1.4 #define [COMMAND](#)(name, cmd, description) [COMMAND_##name](#) ,

Definition at line 38 of file [commands.h](#).

8.26.1.5 **#define COMMAND(name, cmd, desc) int handler_##name (void *arg, mc_platform_p global);**

Definition at line 38 of file commands.h.

8.26.2 **Typedef Documentation**

8.26.2.1 **typedef int(* cmd_handler_t)(void *, mc_platform_p global)**

Definition at line 42 of file commands.h.

8.26.3 **Enumeration Type Documentation**

8.26.3.1 **enum command_indices_e**

Enumerator:

COMMAND

Definition at line 44 of file commands.h.

8.26.4 **Variable Documentation**

8.26.4.1 **cmd_handler_t cmd_handlers[]**

Definition at line 51 of file commands.h.

8.26.4.2 **char* command_cmds[]**

Definition at line 177 of file commands.h.

Referenced by process_command().

8.26.4.3 **char* command_descriptions[]**

Definition at line 114 of file commands.h.

8.27 /home/dko/projects/mobilec/trunk/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 a client 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

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 a client to a host Usage [print_connectlist](#)
- This command displays [help](#) for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList [printlist_agents](#)

8.27.1 Function Documentation

8.27.1.1 This command displays help for any command Usage This command sends a client 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

8.27.1.2 [COMMAND](#) (QUIT, "quit", "This command ends the MobileC application.")

8.27.2 Variable Documentation

8.27.2.1 [help](#)

Definition at line 37 of file commands.x.h.

8.27.2.2 This command displays help for any command Usage This command sends a client to a host Usage [is](#)

Examples:

[LibMCVbEx/Form1.vb](#).

Definition at line 37 of file commands.x.h.

8.27.2.3 This command displays help for any command Usage This command sends a client to a host Usage print_connectlist

Definition at line 37 of file commands.x.h.

8.27.2.4 This command displays help for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList printlist_agents

Definition at line 52 of file commands.x.h.

8.27.2.5 This command displays help for any command Usage send

Definition at line 37 of file commands.x.h.

Referenced by message_Send(), and mtp_http_InitializeFromConnection().

8.28 /home/dko/projects/mobilec/trunk/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)

8.28.1 Typedef Documentation

8.28.1.1 typedef [connection_t](#)* [connection_p](#)

Definition at line 56 of file [connection.h](#).

8.28.1.2 typedef struct [connection_s](#) [connection_t](#)

8.28.2 Function Documentation

8.28.2.1 void [connection_Close](#) ([connection_p](#) *connection*)

8.28.2.2 [connection_p](#) [connection_Copy](#) ([connection_p](#) *connection*)

Definition at line 69 of file [connection.c](#).

References [connection_s::addr](#), [connection_s::clientfd](#), [connection_s::connect_id](#), [connection_New\(\)](#), [connection_s::remote_hostname](#), and [connection_s::serverfd](#).

8.28.2.3 int connection_CreateMessage (connection_p *connection*)**8.28.2.4 int connection_Destroy (connection_p *connection*)**

Definition at line 39 of file connection.c.

References connection_s::clientfd, MC_SUCCESS, and connection_s::remote_hostname.

Referenced by acc_Thread().

8.28.2.5 connection_p connection_New (void)

Definition at line 58 of file connection.c.

Referenced by connection_Copy().

8.28.2.6 int connection_RecvFromConnection (connection_p *connection*, struct mc_platform_s * *global*)**8.28.2.7 int connection_SendtoConnection (connection_p *connection*, char * *send_buffer*)****8.28.2.8 void connection_SetHosts (connection_p *connection*, char * *remote_host*, char * *local_host*)**

8.29 /home/dko/projects/mobilec/trunk/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)
- [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

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](#)

8.29.1 Function Documentation

8.29.1.1 `interpreter_variable_data_t char* fipa_acl_message char* AP_QUEUE_DECL_TEMPLATE (mailbox_queue, agent_mailbox)`

Type constraints

8.29.1.2 `AP_QUEUE_DECL_TEMPLATE (agent_variable_list, interpreter_variable_data)`

8.29.1.3 `AP_QUEUE_DECL_TEMPLATE (agent_queue, agent)`

8.29.1.4 `AP_QUEUE_DECL_TEMPLATE (message_queue, message)`

8.29.1.5 `AP_QUEUE_DECL_TEMPLATE (connection_queue, connection)`

8.29.1.6 `interpreter_variable_data_t char* fipa_acl_message char char* AP_QUEUE_GENERIC_DECL_TEMPLATE (mailbox_queue, SearchID, agent_mailbox_p, int)`

Type constraints

8.29.1.7 `interpreter_variable_data_t char* fipa_acl_message AP_QUEUE_GENERIC_DECL_TEMPLATE (mail_queue, SearchReceivers, fipa_acl_message_p, char *)`

8.29.1.8 `interpreter_variable_data_t char* AP_QUEUE_GENERIC_DECL_TEMPLATE (agent_variable_list, Remove, int, char *)`

Type constraints

8.29.1.9 `struct agent_s int int AP_QUEUE_GENERIC_DECL_TEMPLATE (agent_queue, RemoveName, int, char *)`

Type constraints

8.29.1.10 `struct agent_s int AP_QUEUE_GENERIC_DECL_TEMPLATE (agent_queue, SearchName, struct agent_s *, char *) [read]`

Type constraints

8.29.1.11 `connection_t` `int` `AP_QUEUE_GENERIC_DECL_TEMPLATE` (`connection_queue`, `Remove`, `int`, `int`)

Type constraints

8.29.2 Variable Documentation

8.29.2.1 `interpreter_variable_data_t` `char*` `fipa_acl_message` `char` `agent_mailbox_p`

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

8.29.2.2 `interpreter_variable_data_t` `char` `*fipa_acl_message` `char` `char` `int`

Examples:

[LibMCCppEx/LibMCCppEx.cpp](#), and [LibMCGui/Form1.cs](#).

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

Referenced by `LibMC::MCAgency::ChInitializeOptions()`, `default_callback()`, `main()`, `mpi_lsb()`, `mpi_read_string()`, `net_accept()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_verify()`, `sha4_finish()`, `sha4_update()`, `split_string()`, `ssl_decrypt_buf()`, `ssl_get_session()`, `ssl_parse_certificate_verify()`, `ssl_parse_client_hello()`, `ssl_parse_server_hello()`, and `ssl_set_session()`.

8.29.2.3 `struct` `agent_s` `int` `Remove`

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

8.29.2.4 `interpreter_variable_data_t` `char*` `fipa_acl_message` `char` `char*` `RemoveID`

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

8.29.2.5 `interpreter_variable_data_t` `char*` `fipa_acl_message` `RemoveName`

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

8.29.2.6 `Search`

Definition at line 50 of file `data_structures.h`.

8.29.2.7 `interpreter_variable_data_t` `char*` `fipa_acl_message` `char*` `SearchName`

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

8.30 /home/dko/projects/mobilec/trunk/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](#) { [REQUEST](#) }
- enum [service_types_e](#) {
[ZERO](#), [MISC](#), [INSERT](#), [SOME](#),
[TYPES](#), [HERE](#) }

Functions

- [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](#)

8.30.1 Define Documentation

8.30.1.1 `#define REQUEST(name, string, description) int request_handler_##name (struct mc_platform_s *global, void* data);`

Definition at line 189 of file df.h.

8.30.1.2 `#define REQUEST(name, string, description) REQUEST_##name,`

Definition at line 189 of file df.h.

8.30.2 Enumeration Type Documentation

8.30.2.1 `enum df_request_list_index_e`

Enumerator:

REQUEST

Definition at line 43 of file df.h.

8.30.2.2 `enum service_types_e`

Enumerator:

ZERO

MISC

INSERT

SOME

TYPES

HERE

Definition at line 50 of file df.h.

8.30.3 Function Documentation

8.30.3.1 `int df_AddRequest (struct df_s * df, struct df_request_list_node_s * node)`

Definition at line 60 of file df.c.

References ListAdd(), and SIGNAL.

Referenced by MC_DeregisterService(), MC_RegisterService(), and MC_SearchForService().

8.30.3.2 `int df_Destroy (df_p df)`

Definition at line 77 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().

8.30.3.3 `df_p df_Initialize (struct mc_platform_s * mc_platform)`

8.30.3.4 `int df_node_Destroy (df_node_p df_node)`

Definition at line 405 of file df.c.

References MC_SUCCESS, and MUTEX_LOCK.

Referenced by df_Destroy().

8.30.3.5 `int df_ProcessRequest (struct mc_platform_s * global)`

Definition at line 126 of file df.c.

References mc_platform_s::df, df_request_list_Pop(), MC_ERR_EMPTY, and MC_ERR_INVALID.

Referenced by df_Thread().

8.30.3.6 `int df_request_list_Destroy (df_request_list_p df_request_list)`

Definition at line 314 of file df.c.

References df_request_list_node_Destroy(), ListPop(), ListTerminate(), MC_SUCCESS, and node.

Referenced by df_Destroy().

8.30.3.7 `df_request_list_p df_request_list_New (void)`

Definition at line 336 of file df.c.

References CHECK_NULL, COND_INIT, COND_T, ListInitialize(), MUTEX_INIT, and MUTEX_T.

Referenced by df_Initialize().

8.30.3.8 `int df_request_list_node_Destroy (df_request_list_node_p node)`

Definition at line 285 of file df.c.

References COND_DESTROY, MC_SUCCESS, and MUTEX_DESTROY.

Referenced by df_request_list_Destroy(), and MC_SearchForService().

8.30.3.9 df_request_list_node_p df_request_list_node_New (void)

Definition at line 296 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().

8.30.3.10 df_request_list_node_p df_request_list_Pop (df_request_list_p requests)

Definition at line 361 of file df.c.

References ListPop(), MUTEX_LOCK, MUTEX_UNLOCK, and node.

Referenced by df_ProcessRequest().

8.30.3.11 int df_request_search_Destroy (df_request_search_p node)

Definition at line 392 of file df.c.

References COND_DESTROY, MC_SUCCESS, and MUTEX_DESTROY.

Referenced by MC_SearchForService().

8.30.3.12 df_request_search_p df_request_search_New (void)

Definition at line 377 of file df.c.

References CHECK_NULL, COND_INIT, COND_T, MUTEX_INIT, MUTEX_T, and search.

Referenced by MC_SearchForService().

8.30.3.13 int df_SearchForService (df_p df, const char * searchstring, char *** agent_names, char *** service_names, int ** agent_ids, int * num_entries)

Definition at line 170 of file df.c.

References MC_ERR_NOT_FOUND, MC_SUCCESS, MUTEX_LOCK, MUTEX_UNLOCK, listNode_s::next, and listNode_s::node_data.

8.30.3.14 void df_Start (struct mc_platform_s * mc_platform)

8.30.3.15 void* df_Thread (void * arg)

Definition at line 419 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, and mc_platform_s::quit_lock.

Referenced by df_Start().

8.30.3.16 **STRUCT** (df_deregister, int agent_id;char *service_name;)

8.30.3.17 **STRUCT** (df_node, MUTEX_T *lock;int agent_id;char *agent_name;int num_services;char **service_names;enum service_types_e service_types;)

8.30.3.18 **STRUCT** (df_search_results, char **agent_names;char **service_names;int *agent_ids;int num_results;)

8.30.3.19 **STRUCT** (df_request_list_node, MUTEX_T *lock;COND_T *cond;const char *command;void *data;int data_size;)

8.30.4 Variable Documentation

8.30.4.1 **COND_T * cond**

Definition at line 71 of file df.h.

8.30.4.2 **MUTEX_T * lock**

Definition at line 70 of file df.h.

8.30.4.3 **struct mc_platform_s* mc_platform**

Definition at line 104 of file df.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), agent_RunChScriptThread(), ams_Thread(), cmd_prompt_Thread(), listen_Thread(), mc_platform_Initialize(), and MC_WaitAgent().

8.30.4.4 **struct df_node_s* node**

Definition at line 126 of file df.h.

Referenced by add_variable(), 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(), 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__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(), scan_file(), syncListNew(), write_element(), xml_get_deep_child(), and xml_new_cdata().

8.30.4.5 **int num_entries**

Definition at line 110 of file df.h.

8.30.4.6 df_request_list_p request_list

Definition at line 73 of file df.h.

8.30.4.7 df_search_results_p search_results

Definition at line 88 of file df.h.

8.30.4.8 char* search_string

Definition at line 87 of file df.h.

8.30.4.9 list_p service_list

Definition at line 107 of file df.h.

8.30.4.10 int size

Definition at line 72 of file df.h.

Referenced by agent_AddPersistentVariable(), agent_xml_compose__create_row_nodes(), interpreter_variable_data_Initialize(), interpreter_variable_data_InitializeFromAgent(), MC_GetAgentReturnData(), MC_WaitAgent(), and mxml_vsnprintf().

8.30.4.11 THREAD_T thread

Definition at line 114 of file df.h.

8.30.4.12 int waiting

Definition at line 111 of file df.h.

8.30.4.13 COND_T* waiting_cond

Definition at line 113 of file df.h.

8.30.4.14 MUTEX_T* waiting_lock

Definition at line 112 of file df.h.

8.31 /home/dko/projects/mobilec/trunk/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](#)

8.31.1 Function Documentation

- 8.31.1.1 Register services with the directory facilitator [REQUEST](#) (DEREGISTER, "deregister", "Deregisters mobile agent services from a directory facilitator.")**

Type constraints

- 8.31.1.2 [REQUEST](#) (SUBSCRIBE, "subscribe", "Subscribe to a directory facilitator")**

8.31.2 Variable Documentation

8.31.2.1 [register](#)

Definition at line 40 of file df_request.x.h.

8.31.2.2 Register services with the directory facilitator [search](#)

Definition at line 51 of file df_request.x.h.

Referenced by [df_request_search_New\(\)](#), and [MC_SearchForService\(\)](#).

8.32 /home/dko/projects/mobilec/trunk/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)

8.32.1 Define Documentation

8.32.1.1 #define COMPOSE_BLOCKSIZE 1024

Definition at line 5 of file [dynstring.h](#).

Referenced by [dynstring_Append\(\)](#), and [dynstring_New\(\)](#).

8.32.2 Typedef Documentation

8.32.2.1 typedef [dynstring_t](#)* [dynstring_p](#)

Definition at line 11 of file [dynstring.h](#).

8.32.2.2 typedef struct [dynstring_s](#) [dynstring_t](#)

8.32.3 Function Documentation

8.32.3.1 [int](#) [dynstring_Append](#) ([dynstring_t](#) * *msg*, char * *str*)

Definition at line 21 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_string_Compose\(\)](#), [fipa_url_Compose\(\)](#), [fipa_url_sequence_-Compose\(\)](#), [fipa_word_Compose\(\)](#), and [mtp_http_CreateMessage\(\)](#).

8.32.3.2 `int dynstring_Destroy (dynstring_t * dynstring)`

Definition at line 46 of file `dynstring.c`.

References `dynstring_s::message`.

Referenced by `fipa_envelope_Compose__from()`, `MC_AclSend()`, and `mtp_http_CreateMessage()`.

8.32.3.3 `dynstring_t* dynstring_New (void)`

Definition at line 5 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()`, and `mtp_http_CreateMessage()`.

8.33 /home/dko/projects/mobilec/trunk/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` }
- 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`)
- `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_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)

8.33.1 Typedef Documentation

8.33.1.1 typedef fipa_acl_message_t* fipa_acl_message_p

Definition at line 108 of file fipa_acl.h.

8.33.1.2 typedef struct fipa_acl_message_s fipa_acl_message_t

8.33.1.3 typedef fipa_agent_identifier_t* fipa_agent_identifier_p

Definition at line 158 of file fipa_acl.h.

8.33.1.4 typedef fipa_agent_identifier_set_t* fipa_agent_identifier_set_p

Definition at line 144 of file fipa_acl.h.

8.33.1.5 `typedef struct fipa_agent_identifier_set_s fipa_agent_identifier_set_t`

8.33.1.6 `typedef struct fipa_agent_identifier_s fipa_agent_identifier_t`

8.33.1.7 `typedef fipa_DateTime_t* fipa_DateTime_p`

Definition at line 218 of file fipa_acl.h.

8.33.1.8 `typedef struct fipa_DateTime_s fipa_DateTime_t`

8.33.1.9 `typedef fipa_expression_t* fipa_expression_p`

Definition at line 177 of file fipa_acl.h.

8.33.1.10 `typedef struct fipa_expression_s fipa_expression_t`

8.33.1.11 `typedef enum fipa_expression_type_e fipa_expression_type_t`

8.33.1.12 `typedef fipa_message_string_t* fipa_message_string_p`

Definition at line 120 of file fipa_acl.h.

8.33.1.13 `typedef struct fipa_message_string_s fipa_message_string_t`

8.33.1.14 `typedef fipa_number_t* fipa_number_p`

Definition at line 240 of file fipa_acl.h.

8.33.1.15 `typedef struct fipa_number_s fipa_number_t`

8.33.1.16 `typedef fipa_string_t* fipa_string_p`

Definition at line 199 of file fipa_acl.h.

8.33.1.17 `typedef struct fipa_string_s fipa_string_t`

8.33.1.18 `typedef fipa_url_t* fipa_url_p`

Definition at line 229 of file fipa_acl.h.

8.33.1.19 `typedef fipa_url_sequence_t* fipa_url_sequence_p`

Definition at line 131 of file fipa_acl.h.

8.33.1.20 typedef struct fipa_url_sequence_s fipa_url_sequence_t

8.33.1.21 typedef struct fipa_url_s fipa_url_t

8.33.1.22 typedef fipa_word_t* fipa_word_p

Definition at line 188 of file fipa_acl.h.

8.33.1.23 typedef struct fipa_word_s fipa_word_t

8.33.2 Enumeration Type Documentation

8.33.2.1 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 67 of file fipa_acl.h.

8.33.2.2 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*

Definition at line 37 of file `fipa_acl.h`.

8.33.3 Function Documentation

8.33.3.1 `int fipa_acl_Compose (dynstring_t ** msg, fipa_acl_message_t * acl)`

Definition at line 1182 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_string_Compose()`, `fipa_word_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()`.

8.33.3.2 `fipa_acl_message_t* fipa_acl_message_Copy (fipa_acl_message_t * src)`

Definition at line 78 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_word_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`.

8.33.3.3 `int fipa_acl_message_Destroy (fipa_acl_message_t * message)`

Definition at line 57 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_word_Destroy()`, `fipa_acl_message_s::in_reply_to`, `fipa_acl_message_s::language`, `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 `acc_Thread()`, and `MC_AclDestroy()`.

8.33.3.4 fipa_acl_message_t* fipa_acl_message_New (void)

Definition at line 49 of file fipa_acl.c.

Referenced by acc_Thread(), fipa_acl_message_Copy(), fipa_Reply(), and MC_AclNew().

8.33.3.5 int fipa_acl_Parse (struct fipa_acl_message_s * *acl*, fipa_message_string_p *message*)**8.33.3.6 int fipa_agent_identifier_Compose (dynstring_t * *msg*, fipa_agent_identifier_t * *id*)**

Definition at line 1352 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().

8.33.3.7 fipa_agent_identifier_t* fipa_agent_identifier_Copy (fipa_agent_identifier_t * *src*)

Definition at line 226 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, and fipa_agent_identifier_s::resolvers.

Referenced by fipa_acl_message_Copy(), fipa_acl_Param_Copy(), fipa_agent_identifier_set_Copy(), and fipa_Reply().

8.33.3.8 int fipa_agent_identifier_Destroy (fipa_agent_identifier_t * *id*)

Definition at line 214 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().

8.33.3.9 fipa_agent_identifier_t* fipa_agent_identifier_New (void)

Definition at line 206 of file fipa_acl.c.

Referenced by fipa_agent_identifier_Copy(), fipa_envelope_ParseAgentIdentifier(), MC_AclAddReceiver(), MC_AclAddReplyTo(), and MC_AclSetSender().

8.33.3.10 int fipa_agent_identifier_Parse (fipa_agent_identifier_p * *aid*, fipa_message_string_p *message*)

Definition at line 982 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().

8.33.3.11 `int fipa_agent_identifier_set_Compose (dynstring_t * msg, fipa_agent_identifier_set_t * ids)`

Definition at line 1339 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()`.

8.33.3.12 `fipa_agent_identifier_set_t* fipa_agent_identifier_set_Copy (fipa_agent_identifier_set_t * src)`

Definition at line 185 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()`.

8.33.3.13 `int fipa_agent_identifier_set_Destroy (fipa_agent_identifier_set_t * idset)`

Definition at line 173 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()`.

8.33.3.14 `fipa_agent_identifier_set_t* fipa_agent_identifier_set_New (void)`

Definition at line 165 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()`.

8.33.3.15 `int fipa_agent_identifier_set_Parse (fipa_agent_identifier_set_p * agent_ids, fipa_message_string_p message)`

Definition at line 1141 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()`.

8.33.3.16 `int fipa_CheckNextToken (const fipa_message_string_p message, const char * token)`

Definition at line 644 of file fipa_acl.c.

References `fipa_message_string_s::parse`.

Referenced by `fipa_expression_Parse()`.

8.33.3.17 int fipa_DateTime_Compose (dynstring_t * msg, fipa_DateTime_t * date)

Definition at line 1425 of file fipa_acl.c.

References 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().

8.33.3.18 fipa_DateTime_t* fipa_DateTime_Copy (fipa_DateTime_t * src)

Definition at line 389 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().

8.33.3.19 int fipa_DateTime_Destroy (fipa_DateTime_t * dt)

Definition at line 382 of file fipa_acl.c.

Referenced by fipa_acl_envelope_Received_Destroy(), fipa_acl_message_Destroy(), fipa_acl_Param_Destroy(), and fipa_expression_Destroy().

8.33.3.20 fipa_DateTime_t* fipa_DateTime_New (void)

Definition at line 374 of file fipa_acl.c.

Referenced by fipa_DateTime_Copy().

8.33.3.21 int fipa_datetime_Parse (fipa_DateTime_p * datetime, fipa_message_string_p message)

Definition at line 799 of file fipa_acl.c.

References 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().

8.33.3.22 int fipa_expression_Compose (dynstring_t * msg, fipa_expression_t * expr)

Definition at line 1378 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().

8.33.3.23 fipa_expression_t* fipa_expression_Copy (fipa_expression_t * src)

Definition at line 277 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(), and fipa_expression_Copy().

8.33.3.24 int fipa_expression_Destroy (fipa_expression_t * expr)

Definition at line 246 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().

8.33.3.25 fipa_expression_t* fipa_expression_New (void)

Definition at line 238 of file fipa_acl.c.

Referenced by fipa_expression_Copy().

8.33.3.26 int fipa_expression_Parse (fipa_expression_t ** expression, fipa_message_string_p message)

Definition at line 663 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().

8.33.3.27 int fipa_GetAtom (fipa_message_string_p message, char expected_atom)

Definition at line 579 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().

8.33.3.28 int fipa_GetNextWord (char ** *word*, const fipa_message_string_p *message*)

Definition at line 714 of file fipa_acl.c.

References ERR, MC_SUCCESS, and fipa_message_string_s::parse.

8.33.3.29 int fipa_GetWholeToken (char ** *word*, const fipa_message_string_p *message*)

Definition at line 762 of file fipa_acl.c.

References MC_SUCCESS, and fipa_message_string_s::parse.

Referenced by fipa_datetime_Parse().

8.33.3.30 int fipa_message_parameter_Parse (struct fipa_acl_message_s * *acl*, fipa_message_string_p *message*)**8.33.3.31 fipa_message_string_t* fipa_message_string_Copy (fipa_message_string_t * *src*)**

Definition at line 119 of file fipa_acl.c.

References fipa_message_string_s::message, and fipa_message_string_s::parse.

8.33.3.32 int fipa_message_string_Destroy (fipa_message_string_t * *message*)

Definition at line 109 of file fipa_acl.c.

References fipa_message_string_s::message.

Referenced by acc_Thread().

8.33.3.33 fipa_message_string_t* fipa_message_string_New (void)

Definition at line 101 of file fipa_acl.c.

Referenced by acc_Thread().

8.33.3.34 int fipa_message_type_Parse (enum fipa_performative_e * *performative*, fipa_message_string_p *message*)

Definition at line 517 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().

8.33.3.35 int fipa_number_Compose (dynstring_t * msg, fipa_number_t * number)

Definition at line 1453 of file fipa_acl.c.

References dynstring_Append(), and fipa_number_s::str.

Referenced by fipa_expression_Compose().

8.33.3.36 fipa_number_t* fipa_number_Copy (fipa_number_t * src)

Definition at line 445 of file fipa_acl.c.

References fipa_number_New(), and fipa_number_s::str.

Referenced by fipa_expression_Copy().

8.33.3.37 int fipa_number_Destroy (fipa_number_t * number)

Definition at line 435 of file fipa_acl.c.

References fipa_number_s::str.

Referenced by fipa_expression_Destroy().

8.33.3.38 fipa_number_t* fipa_number_New (void)

Definition at line 427 of file fipa_acl.c.

Referenced by fipa_number_Copy().

8.33.3.39 int fipa_performative_Compose (dynstring_t * msg, enum fipa_performative_e performative)

Definition at line 1251 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().

8.33.3.40 struct fipa_acl_message_s* fipa_Reply (struct fipa_acl_message_s * acl) [read]

Definition at line 1461 of file fipa_acl.c.

References 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_agent_identifier_set_s::num, 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().

8.33.3.41 int fipa_string_Compose (dynstring_t * *msg*, fipa_string_t * *string*)

Definition at line 1416 of file fipa_acl.c.

References fipa_string_s::content, and dynstring_Append().

Referenced by fipa_acl_Compose(), and fipa_expression_Compose().

8.33.3.42 fipa_string_t* fipa_string_Copy (fipa_string_t * *src*)

Definition at line 364 of file fipa_acl.c.

References fipa_string_s::content, and fipa_string_New().

Referenced by fipa_acl_message_Copy(), and fipa_expression_Copy().

8.33.3.43 int fipa_string_Destroy (fipa_string_t * *str*)

Definition at line 354 of file fipa_acl.c.

References fipa_string_s::content.

Referenced by fipa_acl_message_Destroy(), fipa_expression_Destroy(), and MC_AclSetContent().

8.33.3.44 fipa_string_t* fipa_string_New (void)

Definition at line 346 of file fipa_acl.c.

Referenced by fipa_string_Copy(), and MC_AclSetContent().

8.33.3.45 int fipa_string_Parse (fipa_string_p * *fipa_string*, fipa_message_string_p *message*)

Definition at line 937 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().

8.33.3.46 int fipa_url_Compose (dynstring_t * *msg*, fipa_url_t * *url*)

Definition at line 1445 of file fipa_acl.c.

References dynstring_Append(), and fipa_url_s::str.

Referenced by fipa_url_sequence_Compose().

8.33.3.47 fipa_url_t* fipa_url_Copy (fipa_url_t * *src*)

Definition at line 417 of file fipa_acl.c.

References fipa_url_New(), and fipa_url_s::str.

Referenced by fipa_acl_envelope_Received_Copy(), and fipa_url_sequence_Copy().

8.33.3.48 int fipa_url_Destroy (fipa_url_t * url)

Definition at line 407 of file fipa_acl.c.

References fipa_url_s::str.

Referenced by fipa_acl_envelope_Received_Destroy(), and fipa_url_sequence_Destroy().

8.33.3.49 fipa_url_t* fipa_url_New (void)

Definition at line 399 of file fipa_acl.c.

Referenced by fipa_envelope_HandleReceived(), fipa_envelope_ParseAddresses(), fipa_url_Copy(), MC_AclAddReceiver(), MC_AclAddReplyTo(), and MC_AclSetSender().

8.33.3.50 int fipa_url_Parse (fipa_url_p * url, fipa_message_string_p message)

Definition at line 1121 of file fipa_acl.c.

References fipa_word_s::content, fipa_word_Destroy(), and fipa_word_Parse().

Referenced by fipa_url_sequence_Parse().

8.33.3.51 int fipa_url_sequence_Compose (dynstring_t * msg, fipa_url_sequence_t * urls)

Definition at line 1326 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().

8.33.3.52 fipa_url_sequence_t* fipa_url_sequence_Copy (fipa_url_sequence_t * src)

Definition at line 149 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().

8.33.3.53 int fipa_url_sequence_Destroy (fipa_url_sequence_t * sequence)

Definition at line 137 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().

8.33.3.54 fipa_url_sequence_t* fipa_url_sequence_New (void)

Definition at line 129 of file fipa_acl.c.

Referenced by fipa_url_sequence_Copy(), fipa_url_sequence_Parse(), MC_AclAddReceiver(), MC_AclAddReplyTo(), and MC_AclSetSender().

8.33.3.55 int fipa_url_sequence_Parse (fipa_url_sequence_p * *urls*, fipa_message_string_p *message*)

Definition at line 1089 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().

8.33.3.56 int fipa_word_Compose (dynstring_t * *msg*, fipa_word_t * *word*)

Definition at line 1408 of file fipa_acl.c.

References fipa_word_s::content, and dynstring_Append().

Referenced by fipa_acl_Compose(), and fipa_expression_Compose().

8.33.3.57 fipa_word_t* fipa_word_Copy (fipa_word_t * *src*)

Definition at line 336 of file fipa_acl.c.

References fipa_word_s::content, and fipa_word_New().

Referenced by fipa_acl_message_Copy(), and fipa_expression_Copy().

8.33.3.58 int fipa_word_Destroy (fipa_word_t * *word*)

Definition at line 326 of file fipa_acl.c.

References fipa_word_s::content.

Referenced by fipa_acl_message_Destroy(), fipa_agent_identifier_Parse(), fipa_expression_Destroy(), fipa_message_parameter_Parse(), fipa_message_type_Parse(), fipa_url_Parse(), and fipa_url_sequence_Parse().

8.33.3.59 fipa_word_t* fipa_word_New (void)

Definition at line 318 of file fipa_acl.c.

Referenced by fipa_word_Copy().

8.33.3.60 int fipa_word_Parse (fipa_word_t ** *word*, fipa_message_string_p *message*)

Definition at line 602 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_url_Parse(), and fipa_url_sequence_Parse().

8.34 /home/dko/projects/mobilec/trunk/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)

8.34.1 Typedef Documentation

8.34.1.1 `typedef fipa_acl_envelope_t* fipa_acl_envelope_p`

Definition at line 42 of file `fipa_acl_envelope.h`.

8.34.1.2 `typedef struct fipa_acl_envelope_Received_s fipa_acl_envelope_Received_t`

8.34.1.3 `typedef struct fipa_acl_envelope_s fipa_acl_envelope_t`

8.34.1.4 `typedef struct fipa_acl_Param_s fipa_acl_Param_t`

8.34.2 Function Documentation

8.34.2.1 `fipa_acl_envelope_t* fipa_acl_envelope_Copy (fipa_acl_envelope_t * envelope)`

Definition at line 142 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`.

8.34.2.2 `int fipa_acl_envelope_Destroy (fipa_acl_envelope_t * envelope)`

Definition at line 129 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_Thread()`.

8.34.2.3 `fipa_acl_envelope_t* fipa_acl_envelope_New (void)`

Definition at line 121 of file `fipa_envelope.c`.

Referenced by `acc_Thread()`, and `fipa_acl_envelope_Copy()`.

8.34.2.4 **fipa_acl_envelope_Received_t* fipa_acl_envelope_Received_Copy (fipa_acl_envelope_Received_t * *received*)**

Definition at line 61 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, and fipa_acl_envelope_Received_s::received_via.

Referenced by fipa_acl_Param_Copy().

8.34.2.5 **int fipa_acl_envelope_Received_Destroy (fipa_acl_envelope_Received_t * *received*)**

Definition at line 49 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().

8.34.2.6 **fipa_acl_envelope_Received_t* fipa_acl_envelope_Received_New (void)**

Definition at line 40 of file fipa_envelope.c.

Referenced by fipa_acl_envelope_Received_Copy(), and fipa_envelope_HandleReceived().

8.34.2.7 **fipa_acl_Param_t* fipa_acl_Param_Copy (fipa_acl_Param_t * *param*)**

Definition at line 102 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, and fipa_acl_Param_s::to.

Referenced by fipa_acl_envelope_Copy().

8.34.2.8 **int fipa_acl_Param_Destroy (fipa_acl_Param_t * *param*)**

Definition at line 85 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().

8.34.2.9 **fipa_acl_Param_t* fipa_acl_Param_New (void)**

Definition at line 77 of file fipa_envelope.c.

Referenced by fipa_acl_Param_Copy(), and fipa_envelope_HandleTo().

8.34.2.10 char* fipa_envelope_Compose (fipa_acl_message_t * *fipa_acl*)

Definition at line 861 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().

8.34.2.11 mxml_node_t* fipa_envelope_Compose__acl_representation (fipa_acl_message_t * *fipa_acl*)

Definition at line 1067 of file fipa_envelope.c.

References mxmlNewElement(), mxmlNewText(), and node.

Referenced by fipa_envelope_Compose__params().

8.34.2.12 mxml_node_t* fipa_envelope_Compose__date (fipa_acl_message_t * *fipa_acl*)

Definition at line 1095 of file fipa_envelope.c.

References mxmlNewElement(), mxmlNewText(), and node.

Referenced by fipa_envelope_Compose__params().

8.34.2.13 mxml_node_t* fipa_envelope_Compose__envelope (fipa_acl_message_t * *fipa_acl*)

Definition at line 886 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().

8.34.2.14 mxml_node_t* fipa_envelope_Compose__from (fipa_acl_message_t * *fipa_acl*)

Definition at line 1006 of file fipa_envelope.c.

References fipa_agent_identifier_s::addresses, 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().

8.34.2.15 mxml_node_t* fipa_envelope_Compose__params (fipa_acl_message_t * *fipa_acl*)

Definition at line 906 of file fipa_envelope.c.

References fipa_envelope_Compose__acl_representation(), fipa_envelope_Compose__date(), fipa_envelope_Compose__from(), 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()`.

8.34.2.16 `mxml_node_t* fipa_envelope_Compose__payload_encoding (fipa_acl_message_t * fipa_acl)`

Definition at line 1081 of file `fipa_envelope.c`.

References `mxmlNewElement()`, `mxmlNewText()`, and `node`.

Referenced by `fipa_envelope_Compose__params()`.

8.34.2.17 `mxml_node_t* fipa_envelope_Compose__to (fipa_acl_message_t * fipa_acl)`

Definition at line 960 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()`.

8.34.2.18 `int fipa_envelope_HandleAclRepresentation (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)`

Definition at line 373 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()`.

8.34.2.19 `int fipa_envelope_HandleComments (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)`

Definition at line 349 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()`.

8.34.2.20 `int fipa_envelope_HandleDate (struct fipa_acl_envelope_s * envelope, mxml_node_t * param_node, int cur_param)`

Definition at line 444 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().

8.34.2.21 **int fipa_envelope_HandleEnvelope** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *node*)

Definition at line 170 of file fipa_envelope.c.

References fipa_envelope_HandleParams(), MC_ERR_PARSE, MXML_DESCEND_FIRST, and mxmlFindElement().

Referenced by fipa_envelope_Parse().

8.34.2.22 **int fipa_envelope_HandleFrom** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 320 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().

8.34.2.23 **int fipa_envelope_HandleIntendedReceiver** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 474 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().

8.34.2.24 **int fipa_envelope_HandleParams** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *node*)

Definition at line 190 of file fipa_envelope.c.

References 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().

8.34.2.25 **int fipa_envelope_HandlePayloadEncoding** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 420 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()`.

8.34.2.26 `int fipa_envelope_HandlePayloadLength (struct fipa_acl_envelope_s * envelope,
mxml_node_t * param_node, int cur_param)`

Definition at line 397 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()`.

8.34.2.27 `int fipa_envelope_HandleReceived (struct fipa_acl_envelope_s * envelope, mxml_node_t
* param_node, int cur_param)`

Definition at line 552 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()`.

8.34.2.28 `int fipa_envelope_HandleTo (struct fipa_acl_envelope_s * envelope, mxml_node_t *
param_node, int cur_param)`

Definition at line 242 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()`.

8.34.2.29 `int fipa_envelope_Parse (struct fipa_acl_envelope_s * envelope, const char * message)`

Definition at line 157 of file `fipa_envelope.c`.

References `fipa_envelope_HandleEnvelope()`, `MXML_NO_CALLBACK`, `mxmlDelete()`, and `mxmlLoadString()`.

Referenced by `acc_Thread()`.

8.34.2.30 `int fipa_envelope_ParseAddresses (struct fipa_agent_identifier_s * aid, mxml_node_t *
addresses_node)`

Definition at line 729 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().

8.34.2.31 `int fipa_envelope_ParseAgentIdentifier (struct fipa_agent_identifier_s ** aid, mxml_node_t * agent_identifier_node)`

Definition at line 679 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().

8.34.2.32 `int fipa_envelope_ParseResolvers (struct fipa_agent_identifier_s * aid, mxml_node_t * resolvers_node)`

Definition at line 796 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().

8.35 /home/dko/projects/mobilec/trunk/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)

8.35.1 Typedef Documentation

8.35.1.1 typedef [host_id_t](#)* [host_id_p](#)

Definition at line 40 of file [host_id.h](#).

8.35.1.2 typedef struct [host_id_s](#) [host_id_t](#)

8.35.2 Function Documentation

8.35.2.1 [int](#) [host_id_Compare](#) ([host_id_p](#) *host1*, [host_id_p](#) *host2*)

8.35.2.2 void [host_id_Destroy](#) ([host_id_p](#) *host_id*)

8.35.2.3 [host_id_p](#) [host_id_Initialize](#) (const char * *hostname*, [int](#) *port*)

8.36 /home/dko/projects/mobilec/trunk/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)

8.36.1 Typedef Documentation

8.36.1.1 typedef [interpreter_variable_data_t](#)* [interpreter_variable_data_p](#)

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

8.36.1.2 typedef struct [interpreter_variable_data_s](#) [interpreter_variable_data_t](#)

Definition at line 47 of file [interpreter_variable_data.h](#).

8.36.2 Function Documentation

8.36.2.1 [interpreter_variable_data_p](#) [interpreter_variable_data_Copy](#) ([interpreter_variable_data_p](#) src)

Definition at line 225 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](#), and [interpreter_variable_data_s::size](#).

8.36.2.2 int interpreter_variable_data_Destroy (interpreter_variable_data_p agent_return_data)

Definition at line 206 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().

8.36.2.3 interpreter_variable_data_p interpreter_variable_data_Initialize (struct agent_s * agent, const char * varname)**8.36.2.4 interpreter_variable_data_p interpreter_variable_data_InitializeFromAgent (struct agent_s * agent)****8.36.2.5 interpreter_variable_data_p interpreter_variable_data_New (void)**

Definition at line 36 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(), and interpreter_variable_data_Copy().

8.37 /home/dko/projects/mobilec/trunk/src/include/libmc.h File Reference

MobileC api header file.

```
#include <embedch.h>
```

Data Structures

- struct [agency_s](#)
The agency handle.
- struct [MCAgencyOptions_s](#)
User modifiable agency options.

Defines

- #define [EXPORTMC](#)
- #define [MC_Wait](#)(arg1) MC_MainLoop(arg1)

Typedefs

- typedef enum [error_code_e](#) error_code_t
- typedef enum [MC_SteerCommand_e](#) MC_SteerCommand_t
- typedef struct [agency_s](#) agency_t
- typedef [agency_t](#) * [agency_p](#)
- typedef [agency_p](#) [MCAgency_t](#)
- typedef struct [MCAgencyOptions_s](#) MCAgencyOptions_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` = 0x0F, `MC_ALL_SIGNALS` = 0x10 }
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.

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` (MC_Agent_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` (MC_Agent_t agent)
Retrieve an ACL message.
- EXPORTMC `int MC_AclSend` (MC_Agency_t attr, struct `fipa_acl_message_s` *acl)
Send a composed ACL Message.
- EXPORTMC struct `fipa_acl_message_s` * `MC_AclWaitRetrieve` (MC_Agent_t agent)
Wait for and retrieve an ACL message.
- 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'.
- const void * [MC_AgentVariableRetrieve](#) ([MCAgent_t](#) agent, const char *var_name, [int](#) task_num)
Retrieve a pointer to a previously saved variable.
- [int MC_AgentVariableSave](#) ([MCAgent_t](#) agent, const char *var_name)
Mark an agent variable for saving.
- 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_ChInitializeOptions](#) ([MCAgency_t](#) attr, [ChOptions_t](#) *options)
Use custom ChOptions_t type for internal Ch interpreter.
- EXPORTMC [int MC_CallAgentFunc](#) ([MCAgent_t](#) agent, const char *funcName, void *returnVal, void *varg)
Calls a function defined in an agent.
- [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 [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_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_MutexLock` (MCAgency_t attr, `int` id)
Locks a MobileC synchronization variable as a mutex.

- EXPORTMC [int MC_MutexUnlock](#) ([MCAgency_t](#) attr, [int](#) id)
Register a new service with the Directory Facilitator.
- EXPORTMC [int MC_RegisterService](#) ([MCAgency_t](#) agency, [MCAgent_t](#) agent, [int](#) agentID, const char *agentName, char **serviceNames, [int](#) numServices)
Resumes a halted agency.
- EXPORTMC [int MC_ResumeAgency](#) ([MCAgency_t](#) agency)
Retrieves the oldest agent from an agency.
- EXPORTMC [MCAgent_t MC_RetrieveAgent](#) ([MCAgency_t](#) attr)
Post to a MobileC synchronization variable semaphore.
- EXPORTMC [int MC_SemaphorePost](#) ([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_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, [int 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_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.

8.37.1 Detailed Description

MobileC api header file.

Definition in file [libmc.h](#).

8.37.2 Define Documentation

8.37.2.1 #define EXPORTMC

Definition at line 135 of file libmc.h.

8.37.2.2 #define MC_Wait(arg1) MC_MainLoop(arg1)

Definition at line 140 of file libmc.h.

8.37.3 Typedef Documentation

8.37.3.1 typedef agency_t* agency_p

Definition at line 226 of file libmc.h.

8.37.3.2 typedef struct agency_s agency_t

8.37.3.3 typedef agent_t* agent_p

Definition at line 246 of file libmc.h.

8.37.3.4 typedef struct agent_s agent_t

Definition at line 244 of file libmc.h.

8.37.3.5 typedef enum error_code_e error_code_t

8.37.3.6 typedef enum MC_SteerCommand_e MC_SteerCommand_t

8.37.3.7 typedef agency_p MCAgency_t

Definition at line 227 of file libmc.h.

8.37.3.8 typedef struct MCAgencyOptions_s MCAgencyOptions_t

8.37.3.9 typedef agent_t* MCAgent_t

Definition at line 245 of file libmc.h.

8.37.4 Enumeration Type Documentation

8.37.4.1 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 154 of file libmc.h.

8.37.4.2 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 254 of file libmc.h.

8.37.4.3 enum MC_AgentType_e

Enumerator:

MC_NONE
MC_REMOTE_AGENT
MC_LOCAL_AGENT
MC_RETURN_AGENT

Definition at line 249 of file libmc.h.

8.37.4.4 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 199 of file libmc.h.

8.37.4.5 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 184 of file libmc.h.

8.37.4.6 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 172 of file libmc.h.

8.37.5 Function Documentation

8.37.5.1 EXPORTMC int MC_AclAddReceiver (struct fipa_acl_message_s * *acl*, const char * *name*, const char * *address*)

Definition at line 226 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, and fipa_url_sequence_s::urls.

Referenced by MC_AclAddReceiver_chdl().

8.37.5.2 EXPORTMC int MC_AclAddReplyTo (struct fipa_acl_message_s * *acl*, const char * *name*, const char * *address*)

Definition at line 262 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, and fipa_url_sequence_s::urls.

Referenced by MC_AclAddReplyTo_chdl().

8.37.5.3 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 69 of file libmc.c.

References fipa_acl_message_Destroy().

Referenced by MC_AclDestroy_chdl().

8.37.5.4 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 75 of file libmc.c.

References fipa_acl_message_New().

Referenced by MC_AclNew_chdl().

8.37.5.5 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 80 of file libmc.c.

References agent_mailbox_Post(), and agent_s::mailbox.

Referenced by MC_AclPost_chdl(), and MC_AclSend().

8.37.5.6 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 86 of file libmc.c.

References fipa_Reply().

Referenced by MC_AclReply_chdl().

8.37.5.7 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 92 of file libmc.c.

References agent_mailbox_Retrieve(), and agent_s::mailbox.

Referenced by MC_AclRetrieve_chdl().

8.37.5.8 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 98 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_agent_identifier_set_s::fipa_agent_identifiers, fipa_envelope_Compose(), mtp_http_s::host, http_to_hostport(), MC_AclPost(), MC_FindAgentByName(), dynstring_s::message, message_Destroy(), 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, fipa_acl_message_s::receiver, fipa_url_s::str, mtp_http_s::target, message_s::target, and fipa_url_sequence_s::urls.

Referenced by MC_AclSend_chdl().

8.37.5.9 EXPORTMC int MC_AclSetContent (struct fipa_acl_message_s * *acl*, const char * *content*)

Definition at line 297 of file libmc.c.

References fipa_string_s::content, fipa_acl_message_s::content, fipa_string_Destroy(), and fipa_string_New().

Referenced by MC_AclSetContent_chdl().

8.37.5.10 EXPORTMC int MC_AclSetPerformative (struct fipa_acl_message_s * *acl*, enum fipa_performative_e *performative*)

Definition at line 195 of file libmc.c.

References fipa_acl_message_s::performative.

Referenced by MC_AclSetPerformative_chdl().

8.37.5.11 EXPORTMC int MC_AclSetSender (struct fipa_acl_message_s * *acl*, const char * *name*, const char * *address*)

Definition at line 203 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, and fipa_url_sequence_s::urls.

Referenced by MC_AclSetSender_chdl().

8.37.5.12 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 188 of file libmc.c.

References agent_mailbox_WaitRetrieve(), and agent_s::mailbox.

Referenced by MC_AclWaitRetrieve_chdl().

8.37.5.13 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 314 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 MC_AddAgent_chdl().

8.37.5.14 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.

8.37.6 Examples

The following example demonstrates usage of [MC_AgentVariableRetrieve\(\)](#) from agent space.

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>mobagent1</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="2" num="0">
          <TASK num="0" complete="0" server="localhost:5051" code_id="1" />
          <TASK num="1" complete="0" server="localhost:5050" code_id="2" />
        </TASKS>
        <AGENT_CODE id="1">
          <![CDATA[
//#include <stdio.h>
#include <math.h>
int savevar;
int another_savevar;
int array_savevar[10];
int main()
{
    int i;
    printf("Hello World!\n");
    printf("This is mobagent1 from the agency at port 5050.\n");
    printf("I am performing the task on the agency at port 5051 now.\n");
    printf("%f\n", hypot(1,2));
    savevar = 10;
    another_savevar = 20;
    mc_AgentVariableSave(mc_current_agent, "savevar");
    mc_AgentVariableSave(mc_current_agent, "another_savevar");
    for(i = 0; i < 10; i++) {
        array_savevar[i] = i*3;
    }
    mc_AgentVariableSave(mc_current_agent, "array_savevar");
    return 0;
}

]]>
        </AGENT_CODE>
      <AGENT_CODE id="2">
        <![CDATA[
#include <stdio.h>
int retvar;
int main()
{
    const int *i;
    i = (int*)mc_AgentVariableRetrieve(mc_current_agent, "savevar", 0);
    if (i==NULL) {
        printf("Variable 'savevar' not found.\n");
    } else {
        printf("Variable 'savevar' has value %d.\n", *i);
    }
    retvar = *i*2;
    return 0;
}

]]>
      </AGENT_CODE>
    </TASKS>
  </AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>
```

Definition at line 327 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()`.

8.37.6.1 `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`

8.37.7 Examples

See `agent_saved_variables_example/test1.xml` for an example of usage of this api function.

8.37.8 Examples

The following example demonstrates usage of `MC_AgentVariableSave()` from agent space.

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>mobagent1</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="2" num="0">
          <TASK num="0" complete="0" server="localhost:5051" code_id="1" />
          <TASK num="1" complete="0" server="localhost:5050" code_id="2" />
        <AGENT_CODE id="1">
          <![CDATA[
// #include <stdio.h>
#include <math.h>
int savevar;
int another_savevar;
int array_savevar[10];
int main()
{
  int i;
  printf("Hello World!\n");
  printf("This is mobagent1 from the agency at port 5050.\n");
}
```

```

printf("I am performing the task on the agency at port 5051 now.\n");
printf("%f\n", hypot(1,2));
savevar = 10;
another_savevar = 20;
mc_AgentVariableSave(mc_current_agent, "savevar");
mc_AgentVariableSave(mc_current_agent, "another_savevar");
for(i = 0; i < 10; i++) {
    array_savevar[i] = i*3;
}
mc_AgentVariableSave(mc_current_agent, "array_savevar");
return 0;
}

]]>
</AGENT_CODE>
<AGENT_CODE id="2">
<![CDATA[
#include <stdio.h>
int retvar;
int main()
{
    const int *i;
    i = (int*)mc_AgentVariableRetrieve(mc_current_agent, "savevar", 0);
    if (i==NULL) {
        printf("Variable 'savevar' not found.\n");
    } else {
        printf("Variable 'savevar' has value %d.\n", *i);
    }
    retvar = *i*2;
    return 0;
}

]]>
</AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 345 of file libmc.c.

References agent_s::datastate, MC_ERR_MEMORY, agent_task_s::num_saved_variables, agent_task_s::saved_variables, agent_datastate_s::task_progress, and agent_datastate_s::tasks.

Referenced by MC_AgentVariableSave_chdl().

8.37.8.1 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 406 of file libmc.c.

References mc_platform_s::barrier_queue, barrier_queue_Delete(), and agency_s::mc_platform.

Referenced by MC_BarrierDelete_chdl().

8.37.8.2 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.

8.37.9 Examples

The following example demonstrates an agent which sets up an MC_Barrier.

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>CommAgent1</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5051</HOME>
        <TASKS task="1" num="0">
          <TASK num="0"complete="0" server="localhost:5050">
            <DATA number_of_elements="0" name="no-return" >
              </DATA>
            </TASK>
          </TASKS>
          <AGENT_CODE>
            <![CDATA[
#include <stdio.h>
#define MC_BARRIER_ID 56
#define NUM_PROCS 2
int main()
{
    printf("Comm agent: Setting up the MC_Barriers now...\n");
    mc_BarrierInit(MC_BARRIER_ID, NUM_PROCS);
    mc_BarrierInit(MC_BARRIER_ID + 1, NUM_PROCS);
    mc_BarrierInit(MC_BARRIER_ID + 2, NUM_PROCS);
    printf("MC_Barrier Initialized.\n");
    return 0;
}

]]>
          </AGENT_CODE>
        </TASKS>
      </AGENT_DATA>
    </MOBILE_AGENT>
  </MESSAGE>
</MOBILEC_MESSAGE>
```

Definition at line 392 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()`.

8.37.9.1 EXPORTMC int MC_CallAgentFunc (MCAgent_t *agent*, const char * *funcName*, void * *returnVal*, void * *varg*)

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

varg 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

8.37.10 Example

Definition at line 412 of file libmc.c.

References agent_s::agent_interp, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::run_lock.

Referenced by MC_CallAgentFunc_chdl().

8.37.10.1 EXPORTMC int MC_ChInitializeOptions (MCAgency_t *attr*, ChOptions_t * *options*)

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

Definition at line 430 of file libmc.c.

References mc_platform_s::interp_options, and agency_s::mc_platform.

8.37.10.2 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.
- 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 448 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_datastate_s::persistent, agent_task_s::server_name, agent_datastate_s::tasks, and agent_task_s::var_name.

Referenced by MC_ComposeAgent_chdl().

8.37.10.3 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 518 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().

8.37.10.4 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 570 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().

8.37.10.5 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

8.37.11 Example

The following example demonstrates the agent-space version of the function, which is nearly identical to the binary space api function.

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>wake_agent</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5051">
            </TASK>
          <AGENT_CODE>
            <![CDATA[
```



```

#include <stdio.h>

#define SYNC_ID 55
int main()
{
    int cond_id;
    cond_id = SYNC_ID;
    printf("This is the wake agent.\n");
    mc_CondSignal(cond_id);

    return 0;
}
]]>
</AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 533 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()`.

8.37.11.1 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

8.37.12 Example

The following example demonstrates the agent-space version of this function.

```

<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>sleep_agent</NAME>
        <OWNER>IEL</OWNER>
      </AGENT_DATA>
    </MOBILE_AGENT>
  </MESSAGE>
</MOBILEC_MESSAGE>

```

```

<HOME>localhost:5050</HOME>
<TASKS task="1" num="0">
  <TASK num="0" complete="0" server="localhost:5051">
    </TASK>
  <AGENT_CODE>
    <![CDATA[
#include <stdio.h>

#define SYNC_ID 55
int main()
{
    int cond_id;
    printf("Sleep agent has arrived.\n");
    cond_id = mc_SyncInit(SYNC_ID);
    if (cond_id != SYNC_ID) {
        printf("Possible error. Aborting...\n");
        exit(1);
    }
    printf("This is the sleep agent.\n");
    printf("I am going to sleep now...\n");
    mc_CondWait(cond_id);
    printf("This is the sleep agent: I am awake now. Continuing...\n");
    mc_SyncDelete(cond_id);

    return 0;
}
]]>
    </AGENT_CODE>
  </TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 548 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().

8.37.12.1 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 588 of file libmc.c.

References agent_Copy(), and MC_SUCCESS.

8.37.12.2 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 595 of file libmc.c.

References CHECK_NULL, MC_ERR_INVALID, MC_SetAgentStatus(), MC_SUCCESS, MC_TerminateAgent(), and MC_WAIT_FINISHED.

Referenced by MC_DeleteAgent_chdl().

8.37.12.3 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 609 of file libmc.c.

Referenced by MC_DestroyServiceSearchResult_chdl().

8.37.12.4 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

8.37.13 Example

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

int main(int argc, char *argv[])
{
    MCAgency_t agency;
```

```

MCAgencyOptions_t options;
int port=5050;
int remote_port = 5051;

MC_InitializeAgencyOptions(&options);
MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */
agency = MC_Initialize(port, &options);

printf("Mobile-C Started\n");
/* Note: The third argument of the following function may also be a
   valid IP address in the form of a string. i.e. 192.168.0.1 */
MC_SendAgentMigrationMessageFile(agency,
    "test1.xml",
    "localhost",
    remote_port);
MC_End(agency);
exit(0);
}

```

Definition at line 658 of file libmc.c.

References `mc_platform_s::acc`, `mc_platform_s::ams`, `mc_platform_s::cmd_prompt`, `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`, `mc_platform_s::quit`, `mc_platform_s::quit_lock`, `cmd_prompt_s::thread`, `THREAD_CANCEL`, `THREAD_JOIN`, and `agency_s::threads`.

Referenced by `MC_End_chdl()`.

8.37.13.1 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 712 of file libmc.c.

References `mc_platform_s::agent_queue`, and `agency_s::mc_platform`.

Referenced by `MC_FindAgentByID_chdl()`.

8.37.13.2 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

8.37.14 Example

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

int main(int argc, char *argv[])
{
    MCAgency_t agency;
    MCAgencyOptions_t options;
    MCAgent_t agent;
    int dim, *extent;
    double *data;
    int i, j, size;
    int port=5050;
    int remote_port=5051;

    MC_InitializeAgencyOptions(&options);
    MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */
    agency = MC_Initialize(port, &options);

    printf("MobileC Started\n");
    /* Note: The third argument of the following function may also be a
       valid IP address in the form of a string. i.e. 192.168.0.1 */
    MC_SendAgentMigrationMessageFile(agency,
        "test.xml",
        "localhost",
        remote_port);

    /* Wait for return-agent arrival signal */
    MC_WaitSignal(agency, MC_RECV_RETURN);

    /* Make sure we caught the correct agent */
    agent = MC_FindAgentByName(agency, "mobagent3");
    if (agent == NULL) {
        fprintf(stderr, "Did not receive correct agent. \n");
        exit(1);
    }

    /* Print relevant information */
    printf("%d tasks.\n", MC_GetAgentNumTasks(agent) );
    for (i = 0; i < MC_GetAgentNumTasks(agent); i++) {
        MC_GetAgentReturnData(
            agent,
            i,
            (void*)&data,
            &dim,
            &extent );
        printf("Task: %d\n", i);
        size = 1;
        printf("dim is %d\n", dim);
        for (j = 0; j < dim; j++) {
            size *= extent[j];
        }
        printf("Size: %d\n", size);
        printf("Data elements: ");
        for (j = 0; j < size; j++) {
            printf("%f ", data[j]);
        }
        printf("\n\n");
        free(data);
        free(extent);
    }
}
```

```

/* We must reset the signal that we previously caught with the
 * MC_WaitSignal() function with MC_ResetSignal() */
MC_ResetSignal(agency);

MC_End(agency);
return 0;
}

```

Definition at line 699 of file libmc.c.

References `mc_platform_s::agent_queue`, and `agency_s::mc_platform`.

Referenced by `MC_AclSend()`, and `MC_FindAgentByName_chdl()`.

8.37.14.1 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 764 of file libmc.c.

References `agent_s::agent_interp`.

8.37.14.2 EXPORTMC int MC_GetAgentID (MCAgent_t agent)

Retrieve an agent's id.

Definition at line 770 of file libmc.c.

References `agent_s::id`.

8.37.14.3 EXPORTMC char* MC_GetAgentName (MCAgent_t agent)

Definition at line 778 of file libmc.c.

References `agent_s::lock`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `agent_s::name`.

8.37.14.4 EXPORTMC int MC_GetAgentNumTasks (MCAgent_t agent)

Retrieve the number of tasks an agent has.

8.37.15 Example

```

#include <stdio.h>
#include <libmc.h>

int main(int argc, char *argv[])
{

```

```

MCAgency_t agency;
MCAgencyOptions_t options;
MCAgent_t agent;
int dim, *extent;
double *data;
int i, j, size;
int port=5050;
int remote_port=5051;

MC_InitializeAgencyOptions(&options);
MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */
agency = MC_Initialize(port, &options);

printf("MobileC Started\n");
/* Note: The third argument of the following function may also be a
   valid IP address in the form of a string. i.e. 192.168.0.1 */
MC_SendAgentMigrationMessageFile(agency,
    "test.xml",
    "localhost",
    remote_port);

/* Wait for return-agent arrival signal */
MC_WaitSignal(agency, MC_RECV_RETURN);

/* Make sure we caught the correct agent */
agent = MC_FindAgentByName(agency, "mobagent3");
if (agent == NULL) {
    fprintf(stderr, "Did not receive correct agent. \n");
    exit(1);
}

/* Print relevant information */
printf("%d tasks.\n", MC_GetAgentNumTasks(agent) );
for (i = 0; i < MC_GetAgentNumTasks(agent); i++) {
    MC_GetAgentReturnData(
        agent,
        i,
        (void**)&data,
        &dim,
        &extent );
    printf("Task: %d\n", i);
    size = 1;
    printf("dim is %d\n", dim);
    for (j = 0; j < dim; j++) {
        size *= extent[j];
    }
    printf("Size: %d\n", size);
    printf("Data elements: ");
    for (j = 0; j < size; j++) {
        printf("%f ", data[j]);
    }
    printf("\n\n");
    free(data);
    free(extent);
}

/* We must reset the signal that we previously caught with the
 * MC_WaitSignal() function with MC_ResetSignal() */
MC_ResetSignal(agency);

MC_End(agency);
return 0;
}

```

Definition at line 859 of file libmc.c.

References `agent_s::datastate`, and `agent_datastate_s::number_of_tasks`.

8.37.15.1 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.

8.37.16 Example

This file demonstrates the retrieval of agent return data from an agent

```
/* mc_sample_app.c
 *
 * This sample program uses the Mobile C library to build
 * a simple command-line driven client/server app.
 *
 * 12/15/2006
 * */

#include <libmc.h>
#include <stdio.h>
#ifdef _WIN32
#include <windows.h>
#endif
int main(int argc, char *argv[])
{
    MCAgency_t agency;
    MCAgent_t agent;
    int *extent;
    int dim;
    short *array;
    int i, size;
    char *xml;
    MCAgencyOptions_t options;
    int port = 5051;
    int remote_port = 5050;

    MC_InitializeAgencyOptions(&options);
    MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */
    if (argc == 2) {
        printf("Starting agency listening on port %d.\n",
            atoi(argv[1]) );
        agency = MC_Initialize(
            atoi(argv[1]),
            &options
        );
    } else {
        agency = MC_Initialize(
            port,
            &options);
    }
    /* Note: The third argument of the following function may also be a
    valid IP address in the form of a string. i.e. 192.168.0.1 */
    MC_SendAgentMigrationMessageFile(
        agency,
```



```

        "agent.xml",
        "localhost",
        remote_port);
while(1) {
    printf("Waiting for agent return...\n");
    MC_WaitSignal(
        agency,
        MC_RECV_RETURN
    );
    agent = MC_FindAgentByName(agency, "mobagent1");
    if (agent == NULL) {
        printf("Found wrong agent. Waiting again...\n");
        MC_ResetSignal(agency);
        continue;
    }
    if (MC_GetAgentType(agent) != MC_RETURN_AGENT) {
        printf("Found wrong agent. Waiting again...\n");
        MC_ResetSignal(agency);
        continue;
    } else {
        break;
    }
}
/* Get and print the agent's XML code */
xml = MC_GetAgentXMLString(agent);
printf("%s\n", xml);
free(xml);

/* Retrieve the agent's return data */
MC_GetAgentReturnData(agent, 0, (void**)&array, &dim, &extent);

/* Since we have previously called MC_WaitSignal(), we must now reset
 * the signal to allow the agency to continue. */
MC_ResetSignal(agency);

/* Print relevant data */
printf("dim is %d\n", dim);
printf("Extents are: ");
size = 1;
for (i = 0; i < dim; i++) {
    printf("%d ", extent[i]);
    size *= extent[i];
}
printf("\n");
printf("%d Elements: \n", size);
for (i = 0; i < size; i++) {
    printf("%d ", array[i]);
}
printf("\n");

MC_End(agency);

return 0;
}

```

This is the agent which gets the data

```

<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>mobagent1</NAME>

```

```

<OWNER>IEL</OWNER>
<HOME>localhost:5051</HOME>
<TASKS task="1" num="0">
  <TASK num="0" complete="0" server="localhost:5050" return="a">
    </TASK>
    <AGENT_CODE>
      <![CDATA[
#include <stdio.h>
short a[2][3][2];
int main()
{
    int i, j, k, l;
    k = 0;
    for (i = 0; i < 2; i++) {
        for (j = 0; j < 3; j++) {
            for(l = 0; l < 2; l++) {
                a[i][j][l] = k;
                k++;
                printf("%d ", i+j);
            }
        }
    }
    printf("\nThis is a mobile agent from port 5050.\n");
    printf("I am performing the task on the agency at port 5051 now.\n");
    sleep(1);

    return 0;
}

]]>
</AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 796 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.

8.37.16.1 EXPORTMC int MC_GetAgentStatus (MCAgent_t agent)

Get an agent's current status.

Returns:

returns type 'enum MC_AgentStatus_e'

Definition at line 742 of file libmc.c.

References agent_s::agent_status, agent_s::lock, MUTEX_LOCK, and MUTEX_UNLOCK.

8.37.16.2 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 865 of file libmc.c.

References `agent_s::agent_type`.

8.37.16.3 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 752 of file libmc.c.

References `agent_s::datastate`, `mxmlSaveAllocString()`, and `agent_datastate_s::xml_agent_root`.

8.37.16.4 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 911 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()`.

8.37.16.5 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

8.37.17 Example

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

#ifdef _WIN32
#include <windows.h>
```

```

#endif

int main(int argc, char *argv[])
{
    MCAgency_t agency;
    int local_port = 5051;

    agency = MC_Initialize(local_port, NULL);

    printf("Mobile-C Started\n");

    MC_MainLoop(agency);

    MC_End(agency);
    return 0;
}

```

Definition at line 920 of file libmc.c.

References CHECK_NULL, agency_s::client, MCAgencyOptions_s::default_agent_status, agency_s::default_agentstatus, MCAgencyOptions_s::enable_security, agency_s::enable_security, HOST_NAME_MAX, agency_s::hostName, MC_InitializeAgencyOptions(), agency_s::mc_platform, mc_platform_initialize(), MC_THREAD_ALL, agency_s::portno, agency_s::server, MCAgencyOptions_s::stack_size, agency_s::stack_size, MCAgencyOptions_s::threads, and agency_s::threads.

8.37.17.1 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.

8.37.18 Example

```

#include <stdio.h>
#include <libmc.h>

#ifdef _WIN32
#include <windows.h>
#endif

int main(int argc, char *argv[])
{
    MCAgency_t agency;
    int local_port = 5051;

    agency = MC_Initialize(local_port, NULL);

    printf("Mobile-C Started\n");

    MC_MainLoop(agency);
}

```

```
MC_End(agency);  
return 0;  
}
```

Definition at line 975 of file libmc.c.

References MCAgencyOptions_s::default_agent_status, MCAgencyOptions_s::enable_security, MC_THREAD_ALL, MC_WAIT_CH, MCAgencyOptions_s::modified, MCAgencyOptions_s::stack_size, and MCAgencyOptions_s::threads.

Referenced by MC_Initialize().

8.37.18.1 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 992 of file libmc.c.

References agency_s::mc_platform, message_Destroy(), message_InitializeFromString(), message_New(), mc_platform_s::message_queue, MXML_DESCEND, mxmFindElement(), mxmLoadString(), message_s::to_address, message_s::xml_payload, and message_s::xml_root.

8.37.18.2 EXPORTMC int MC_MainLoop (MCAgency_t attr)

Wait indefinitely.

Note:

This function is intended to block the calling thread forever.

Definition at line 1617 of file libmc.c.

References mc_platform_s::ams, and agency_s::mc_platform.

8.37.18.3 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

8.37.19 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.

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>sleep_agent</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5051">
            <DATA dim="0" name="no-return">
              </DATA>
            </TASK>
          </TASKS>
        <AGENT_CODE>
          <![CDATA[
#include <stdio.h>
int main()
{
    int mutex_id;
    printf("Sleep agent has arrived.\n");
    mutex_id = mc_SyncInit(55);
    if (mutex_id != 55) {
        printf("Possible error. Aborting...\n");
        exit(1);
    }
    printf("This is agent 1.\n");
    printf("Agent 1: I am locking the mutex now.\n");
    mc_MutexLock(mutex_id);
    printf("Agent 1: Mutex locked. Perform protected operations here\n");
    printf("Agent 1: Waiting for 5 seconds...\n");
    sleep(5);
    printf("Agent 1: Unlocking mutex now...\n");
    mc_MutexUnlock(mutex_id);

    return 0;
}

]]>
          </AGENT_CODE>
        </TASKS>
      </AGENT_DATA>
    </MOBILE_AGENT>
  </MESSAGE>
</MOBILEC_MESSAGE>

<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>wake_agent</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5051">
            <DATA dim="0" name="no-return" complete="0" server="localhost:5051">
              </DATA>
            </TASK>
          </TASKS>
        </AGENT_DATA>
      </MOBILE_AGENT>
    </MESSAGE>
  </MOBILEC_MESSAGE>
```

```

        </DATA>
    </TASK>
    <AGENT_CODE>
    <![CDATA[
#include <stdio.h>
int main()
{
    int mutex_id;
    mutex_id = 55;
    printf("Agent 2: Has arrived");
    printf("Agent 2: Attempting to lock the mutex...\n");
    mc_MutexLock(mutex_id);
    printf("Agent 2: Mutex locked.\n");
    printf("Agent 2: Perform protected operations here.\n");
    sleep(5);
    mc_MutexUnlock(mutex_id);
    printf("Agent 2: Mutex Unlocked\n");
    mc_SyncDelete(mutex_id);

    return 0;
}
    ]]>
    </AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 1062 of file libmc.c.

References syncListNode_s::lock, agency_s::mc_platform, MUTEX_LOCK, mc_platform_s::syncList, and syncListFind().

8.37.19.1 EXPORTMC int MC_MutexUnlock (MCAgency_t attr, int id)

Definition at line 1074 of file libmc.c.

References syncListNode_s::lock, agency_s::mc_platform, MUTEX_UNLOCK, mc_platform_s::syncList, and syncListFind().

8.37.19.2 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 1086 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.

8.37.19.3 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)'.

8.37.20 Example

```
<?xml version="1.0"?>
<!DOCTYPE myMessage SYSTEM "mobilec.dtd">
<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>service_provider_1</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5051">
            <DATA persistent="1" number_of_elements="0"
              name="no-return" >
            </DATA>
          </TASK>
        </TASKS>
      <AGENT_CODE>
        <![CDATA[
#include <stdio.h>

struct arg_struct {
    int a;
    int b;
};

int main() {
    char **services;
    int i;
    services = malloc(sizeof(char*)*2);
    for(i = 0; i < 2; i++) {
        services[i] = malloc(40);
    }
    strcpy(services[0], "addition");
    strcpy(services[1], "subtraction");
    printf("Service provider 1 has arrived.\n");
    printf("I provide addition and subtraction service.\n");
    mc_RegisterService( mc_current_agent, services, 2);
    return 0;
}

int addition(struct arg_struct* arg) {
    printf("Adding %d and %d...\n", arg->a, arg->b);
    return arg->a + arg->b;
}
```



```

}

int subtraction(struct arg_struct* arg) {
    printf("Subtracting %d - %d...\n", arg->a, arg->b);
    return arg->a - arg->b;
}

]]>
</AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 1102 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`.

8.37.20.1 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 1236 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`.

8.37.20.2 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 1181 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()`.

8.37.20.3 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 1190 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.

8.37.20.4 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 1220 of file libmc.c.

References agent_datastate_s::agent_code, agent_s::datastate, agent_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, and agent_datastate_s::task_progress.

8.37.20.5 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

8.37.21 Example

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
```

```

    <NAME>mobagent1</NAME>
    <OWNER>IEL</OWNER>
    <HOME>localhost:5051</HOME>
    <TASKS task="1" num="0">
      <TASK num="0" complete="0" server="localhost:5050">
        <DATA number_of_elements="0" name="no-return" >
          </DATA>
        </TASK>
      <AGENT_CODE>
        <![CDATA[
#include <stdio.h>
struct arg_struct {
    int a;
    int b;
};
int main()
{
    MCAgent_t agent;
    int retval;
    /* Search Return Variables */
    char** agentNames;
    char** serviceNames;
    int *agentIDs;
    int numResults;

    /* Argument Struct */
    struct arg_struct arg;

    /* Search for addition service */
    printf("\n\nSearching for addition service.\n");
    mc_SearchForService(
        "addition",
        &agentNames,
        &serviceNames,
        &agentIDs,
        &numResults );
    printf("Done searching.\n");
    if (numResults < 1) {
        printf("No agents with service 'addition' found.\n");
        exit(0);
    }

    /* Just get the first hit */
    printf("Using agent %s for addition.\n", agentNames[0]);
    agent = mc_FindAgentByID(agentIDs[0]);

    arg.a = 44;
    arg.b = 45;
    mc_CallAgentFunc(agent, "addition", &retval, &arg);
    printf("Result of addition %d + %d is %d.\n", arg.a, arg.b, retval);

    /* Now search for multiplication service */
    printf("\n\n Searching for Multiplication service...\n");
    mc_SearchForService(
        "multiplication",
        &agentNames,
        &serviceNames,
        &agentIDs,
        &numResults );

    if (numResults < 1) {
        printf("No agents with service 'multiplication' found.\n");
        exit(0);
    }

    printf("Using agent %s for multiplication.\n", agentNames[0]);
    agent = mc_FindAgentByID(agentIDs[0]);

```

```

mc_CallAgentFunc(agent, "multiplication", &retval, &arg);
printf("Result of multiplication %d * %d is %d.\n", arg.a, arg.b, retval);

/* Now lets try to deregister a service */
mc_DeregisterService(
    agentIDs[0],
    serviceNames[0]
);

return 0;
}

]]>
</AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 1247 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.

8.37.21.1 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

8.37.22 Example

```

<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>wake_agent</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5051">
            <DATA dim="0" name="no-return">
              </DATA>
            </TASK>
          </TASKS>
        <AGENT_CODE>
          <![CDATA[
#include <stdio.h>
int main()
{

```

```

    int semaphore_id;
    semaphore_id = 55;
    printf("This is the wake agent.\n");
    mc_SemaphorePost(semaphore_id);

    return 0;
}

]]>
</AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 1297 of file libmc.c.

References `agency_s::mc_platform`, `syncListNode_s::sem`, `SEMAPHORE_POST`, `mc_platform_s::syncList`, and `syncListFind()`.

8.37.22.1 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.

8.37.23 Example

```

<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>sleep_agent</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5051">
            <DATA dim="0" name="no-return">
              </DATA>
            </TASK>
          <AGENT_CODE>
            <![CDATA[
#include <stdio.h>
int main()

```

```

{
    int sem_id;
    printf("Sleep agent has arrived.\n");
    sem_id = mc_SyncInit(55);
    if (sem_id != 55) {
        printf("Possible error. Aborting...\n");
        exit(1);
    }
    printf("This is the sleep agent.\n");
    printf("I am going to sleep now...\n");
    mc_SemaphoreWait(sem_id);
    printf("This is the sleep agent: I am awake now. Continuing...\n");
    mc_SyncDelete(sem_id);

    return 0;
}

]]>
</AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 1309 of file libmc.c.

References `agency_s::mc_platform`, `syncListNode_s::sem`, `SEMAPHORE_WAIT`, `mc_platform_s::syncList`, and `syncListFind()`.

8.37.23.1 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 1331 of file libmc.c.

References `MC_ERR`, `agency_s::mc_platform`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, and `mc_platform_s::message_queue`.

8.37.23.2 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 1362 of file libmc.c.

References `agency_s::mc_platform`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, and `mc_platform_s::message_queue`.

8.37.23.3 EXPORTMC int MC_SetAgentStatus (MCAgent_t *agent*, int *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 1448 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_DeleteAgent()`.

8.37.23.4 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 1463 of file libmc.c.

References `mc_platform_s::default_agentstatus`, and `agency_s::mc_platform`.

8.37.23.5 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 1490 of file libmc.c.

References SET_THREAD_OFF, and MCAgencyOptions_s::threads.

8.37.23.6 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 1473 of file libmc.c.

References SET_THREAD_ON, and MCAgencyOptions_s::threads.

8.37.23.7 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 1497 of file libmc.c.

References MC_THREAD_ALL, SET_THREAD_OFF, and MCAgencyOptions_s::threads.

8.37.23.8 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 1480 of file libmc.c.

References [MC_THREAD_ALL](#), [SET_THREAD_ON](#), and [MCAgencyOptions_s::threads](#).

8.37.23.9 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.

8.37.24 Example

```
#include <stdio.h>
#include <libmc.h>
#ifdef _WIN32
#include <windows.h>
#endif

int algorithm(void* boo);

int main() {
    MCAgency_t agency;
    int local_port = 5050;
    MCAgencyOptions_t options;

    MC_InitializeAgencyOptions(&options);

    MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */

    agency = MC_Initialize(local_port, &options);

    printf("Starting algorithm...\n");
    MC_Steer(
```

```

        agency,
        &algorithm,
        NULL
    );

    MC_End(agency);
    return 0;
}

int algorithm(void* boo)
{
    int i=0;
    MC_SteerCommand_t command;
    while(1) {
#ifdef _WIN32
        sleep(1);
#else
        Sleep(1000);
#endif
        printf("%d \n", i);
        i++;
        command = MC_SteerControl();
        if(
            command == MC_RESTART ||
            command == MC_STOP
        )
        {
            return 0;
        }
    }
}

```

Definition at line 1507 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`.

8.37.24.1 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 1526 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`.

8.37.24.2 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 1542 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()`.

8.37.24.3 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 1572 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()`.

8.37.24.4 EXPORTMC int MC_TerminateAgent (MCAgent_t *agent*)

Halt a running agent.

Returns:

0 on success, `error_code_t` on failure

Definition at line 1597 of file `libmc.c`.

References `agent_s::agent_interp`.

Referenced by `MC_DeleteAgent()`.

8.37.24.5 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 1624 of file libmc.c.

References `mc_platform_s::agent_queue`, `COND_WAIT`, `mc_platform`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `size`.

8.37.24.6 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 1644 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`.

8.37.24.7 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.

8.37.25 Example

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

int main(int argc, char *argv[])
```

```

{
    MCAgency_t agency;
    MCAgencyOptions_t options;
    MCAgent_t agent;
    int dim, *extent;
    double *data;
    int i, j, size;
    int port=5050;
    int remote_port=5051;

    MC_InitializeAgencyOptions(&options);
    MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */
    agency = MC_Initialize(port, &options);

    printf("MobileC Started\n");
    /* Note: The third argument of the following function may also be a
       valid IP address in the form of a string. i.e. 192.168.0.1 */
    MC_SendAgentMigrationMessageFile(agency,
        "test.xml",
        "localhost",
        remote_port);

    /* Wait for return-agent arrival signal */
    MC_WaitSignal(agency, MC_RECV_RETURN);

    /* Make sure we caught the correct agent */
    agent = MC_FindAgentByName(agency, "mobagent3");
    if (agent == NULL) {
        fprintf(stderr, "Did not receive correct agent. \n");
        exit(1);
    }

    /* Print relevant information */
    printf("%d tasks.\n", MC_GetAgentNumTasks(agent) );
    for (i = 0; i < MC_GetAgentNumTasks(agent); i++) {
        MC_GetAgentReturnData(
            agent,
            i,
            (void*)&data,
            &dim,
            &extent );
        printf("Task: %d\n", i);
        size = 1;
        printf("dim is %d\n", dim);
        for (j = 0; j < dim; j++) {
            size *= extent[j];
        }
        printf("Size: %d\n", size);
        printf("Data elements: ");
        for (j = 0; j < size; j++) {
            printf("%f ", data[j]);
        }
        printf("\n\n");
        free(data);
        free(extent);
    }

    /* We must reset the signal that we previously caught with the
       * MC_WaitSignal() function with MC_ResetSignal() */
    MC_ResetSignal(agency);

    MC_End(agency);
    return 0;
}

```

Definition at line 1662 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().

8.38 /home/dko/projects/mobilec/trunk/src/include/macros.h File Reference

```
#include <pthread.h>
#include <semaphore.h>
#include "config.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 [PTHREAD_STACK_SIZE](#) 131072
- #define [THREAD_T](#) pthread_t
- #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 [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)

8.38.1 Define Documentation

8.38.1.1 #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 430 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(), and MC_GetAgentReturnData().

8.38.1.2 #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); /*FIXME*/ \
break; \
case CH_FLOATTYPE: \
                                *(float*)val = strtod(string, NULL); \
break; \
case CH_DOUBLETYPE: \
                                *(double*)val = strtod(string, NULL); \
break; \
default: \
    fprintf(stderr, \
        "Unsupported data type: %d %s:%d\n", \
        type, __FILE__, __LINE__ ); \
}

```

Definition at line 544 of file macros.h.

Referenced by agent_xml_parse__data().

8.38.1.3 #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_DOUBLETYPE:
        strcpy(string, "double");
        break;
    default:
        fprintf(stderr,
            "Unsupported data type: %d %s:%d\n",
            type, __FILE__, __LINE__ );
}

```

Definition at line 461 of file macros.h.

Referenced by agent_xml_compose__data().

8.38.1.4 #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 493 of file macros.h.

Referenced by agent_xml_compose__create_row_nodes(), and agent_xml_compose__data().

8.38.1.5 #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 522 of file macros.h.

Referenced by agent_xml_parse__data().

8.38.1.6 #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 413 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__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(), listen_Thread(), 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_DeregisterService_chdl(), MC_End_chdl(), MC_FindAgentByID_chdl(), MC_FindAgentByName_chdl(), MC_HaltAgency_chdl(), MC_Initialize(), mc_platform_Initialize(), MC_RegisterService(), mc_rwlock_init(), MC_SearchForService(), message_InitializeFromAgent(), message_InitializeFromConnection(), message_InitializeFromString(), message_New(), message_queue_SendOutgoing(), message_Send(), mtp_http_InitializeFromConnection(), mtp_http_New(), syncListNodeInit(), syncListNodeNew(), xml_get_cdata(), xml_get_text(), and xml_new_cdata().

8.38.1.7 #define COND_BROADCAST(cond) pthread_cond_broadcast(cond)

Definition at line 202 of file macros.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), ams_Thread(), df_Thread(), listen_Thread(), MC_Barrier(), MC_CondBroadcast(), and MC_SendSteerCommand().

8.38.1.8 #define COND_DESTROY(cond) pthread_cond_destroy(cond)

Definition at line 178 of file macros.h.

Referenced by ams_Destroy(), barrier_node_Destroy(), df_Destroy(), df_request_list_node_Destroy(), df_request_search_Destroy(), mc_platform_Destroy(), mc_rwlock_destroy(), and syncListNodeDestroy().

8.38.1.9 #define COND_INIT(cond) pthread_cond_init(cond, NULL)

Definition at line 175 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().

8.38.1.10 #define COND_RESET(cond, mutex) pthread_mutex_unlock(mutex);

Definition at line 190 of file macros.h.

8.38.1.11 #define COND_SIGNAL(cond) pthread_cond_signal(cond)

Definition at line 204 of file macros.h.

Referenced by `acc_MessageHandlerThread()`, `agent_RunChScriptThread()`, `MC_AddAgent()`, `MC_CondSignal()`, `MC_End()`, `MC_ResetSignal()`, `mc_rwlock_rdunlock()`, `mc_rwlock_wrunlock()`, and `MC_SetAgentStatus()`.

8.38.1.12 **#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 184 of file `macros.h`.

8.38.1.13 **#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 192 of file `macros.h`.

Referenced by `MC_SearchForService()`.

8.38.1.14 **#define COND_T pthread_cond_t**

Definition at line 173 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()`.

8.38.1.15 **#define COND_WAIT(cond, mutex) pthread_cond_wait(cond, mutex)**

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

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `agent_mailbox_WaitRetrieve()`, `ams_Thread()`, `df_Thread()`, `MC_Barrier()`, `MC_CondWait()`, `mc_platform_Initialize()`, `mc_rwlock_rdlock()`, `mc_rwlock_wrlock()`, `MC_SteerControl()`, `MC_WaitAgent()`, and `MC_WaitSignal()`.

8.38.1.16 **#define GET_THREAD_MODE(a, b) ((a & (1<<b)) / (1<<b))**

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

Referenced by `MC_End()`, and `mc_platform_Initialize()`.

8.38.1.17 #define MUTEX_DESTROY(mutex) pthread_mutex_destroy(mutex)

Definition at line 156 of file macros.h.

Referenced by agent_Destroy(), agent_Initialize(), ams_Destroy(), barrier_node_Destroy(), df_Destroy(), df_request_list_node_Destroy(), df_request_search_Destroy(), mc_platform_Destroy(), mc_rwlock_destroy(), and syncListNodeDestroy().

8.38.1.18 #define MUTEX_INIT(mutex) pthread_mutex_init(mutex, NULL)

Definition at line 153 of file macros.h.

Referenced by acc_Initialize(), agent_Copy(), agent_Initialize(), agent_New(), 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().

8.38.1.19 #define MUTEX_LOCK(mutex)

Value:

```
if (pthread_mutex_lock( mutex ))
    fprintf(stderr, "pthread lock error: %s:%d\n", __FILE__, __LINE__) \
```

Definition at line 159 of file macros.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), agent_Copy(), agent_Destroy(), agent_mailbox_WaitRetrieve(), 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(), interpreter_variable_data_Initialize(), listen_Thread(), MC_AddAgent(), MC_Barrier(), MC_CallAgentFunc(), MC_CondBroadcast(), MC_CondReset(), MC_CondSignal(), MC_CondWait(), MC_End(), MC_GetAgentName(), MC_GetAgentStatus(), MC_GetAllAgents(), MC_HaltAgency(), 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_SendSteerCommand(), MC_SetAgentStatus(), MC_Steer(), MC_SteerControl(), MC_SyncDelete(), MC_SyncInit(), MC_WaitAgent(), MC_WaitRetrieveAgent(), MC_WaitSignal(), and message_queue_SendOutgoing().

8.38.1.20 #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 164 of file macros.h.

Referenced by agent_New().

8.38.1.21 #define MUTEX_T pthread_mutex_t

Definition at line 151 of file macros.h.

Referenced by `acc_Initialize()`, `agent_Copy()`, `agent_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_RegisterService()`, `mc_rwlock_init()`, `syncListInit()`, `syncListNodeInit()`, and `syncListNodeNew()`.

8.38.1.22 **#define MUTEX_UNLOCK(mutex) pthread_mutex_unlock(mutex)**

Definition at line 162 of file `macros.h`.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `agent_mailbox_WaitRetrieve()`, `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()`, `interpreter_variable_data_Initialize()`, `listen_Thread()`, `MC_AddAgent()`, `MC_Barrier()`, `MC_CallAgentFunc()`, `MC_CondBroadcast()`, `MC_CondReset()`, `MC_CondSignal()`, `MC_CondWait()`, `MC_End()`, `MC_GetAgentName()`, `MC_GetAgentStatus()`, `MC_GetAllAgents()`, `MC_HaltAgency()`, `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_SendSteerCommand()`, `MC_SetAgentStatus()`, `MC_Steer()`, `MC_SteerControl()`, `MC_SyncDelete()`, `MC_SyncInit()`, `MC_WaitAgent()`, `MC_WaitRetrieveAgent()`, `MC_WaitSignal()`, and `message_queue_SendOutgoing()`.

8.38.1.23 **#define PTHREAD_STACK_SIZE 131072**

Definition at line 130 of file `macros.h`.

8.38.1.24 **#define RWLOCK_DESTROY(rwlock) mc_rwlock_destroy(rwlock)**

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

Referenced by `barrier_queue_Destroy()`.

8.38.1.25 **#define RWLOCK_INIT(rwlock) mc_rwlock_init(rwlock)**

Definition at line 238 of file `macros.h`.

Referenced by `barrier_queue_New()`, and `syncListInit()`.

8.38.1.26 **#define RWLOCK_RDLOCK(rwlock) mc_rwlock_rdlock(rwlock)**

Definition at line 264 of file `macros.h`.

Referenced by `barrier_queue_Get()`, and `syncListFind()`.

8.38.1.27 **#define RWLOCK_RDUNLOCK(rwlock) mc_rwlock_rdunlock(rwlock)**

Definition at line 266 of file `macros.h`.

Referenced by `barrier_queue_Get()`, and `syncListFind()`.

8.38.1.28 **#define RWLOCK_T mc_rwlock_t**

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

Referenced by barrier_queue_New(), and syncListInit().

8.38.1.29 **#define RWLOCK_WRLOCK(rwlock) mc_rwlock_wrlock(rwlock)**

Definition at line 268 of file macros.h.

Referenced by barrier_queue_Add(), barrier_queue_Delete(), syncListAddNode(), syncListDelete(), and syncListRemove().

8.38.1.30 **#define RWLOCK_WRUNLOCK(rwlock) mc_rwlock_wrunlock(rwlock)**

Definition at line 270 of file macros.h.

Referenced by barrier_queue_Add(), barrier_queue_Delete(), syncListAddNode(), syncListDelete(), and syncListRemove().

8.38.1.31 **#define SEMAPHORE_DESTROY(sem) sem_destroy(sem)**

Definition at line 216 of file macros.h.

Referenced by syncListNodeDestroy().

8.38.1.32 **#define SEMAPHORE_INIT(sem) sem_init(sem, 0, 0)**

Definition at line 213 of file macros.h.

Referenced by syncListNodeInit(), and syncListNodeNew().

8.38.1.33 **#define SEMAPHORE_POST(sem) sem_post(sem)**

Definition at line 221 of file macros.h.

Referenced by MC_SemaphorePost().

8.38.1.34 **#define SEMAPHORE_T sem_t**

Definition at line 211 of file macros.h.

Referenced by syncListNodeInit(), and syncListNodeNew().

8.38.1.35 **#define SEMAPHORE_WAIT(sem) sem_wait(sem)**

Definition at line 219 of file macros.h.

Referenced by MC_SemaphoreWait().

8.38.1.36 **#define SET_THREAD_OFF(a, b) a = (a & ~(1<<b)))**

Definition at line 111 of file macros.h.

Referenced by MC_SetThreadOff(), and MC_SetThreadsAllOff().

8.38.1.37 #define SET_THREAD_ON(a, b) a = (a | (1<<b))

Definition at line 110 of file macros.h.

Referenced by MC_SetThreadOn(), and MC_SetThreadsAllOn().

8.38.1.38 #define SIGNAL(cond, mutex, action)

Value:

```
pthread_mutex_lock( mutex ); \
action; \
pthread_cond_signal( cond ); \
pthread_mutex_unlock( mutex )
```

Definition at line 197 of file macros.h.

Referenced by agent_RunChScriptThread(), df_Add(), and df_AddRequest().

8.38.1.39 #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 284 of file macros.h.

8.38.1.40 #define SLEEP_RESET(queue) pthread_mutex_unlock(queue → thread_mutex)

Definition at line 288 of file macros.h.

8.38.1.41 #define STRUCT(name, members)

Value:

```
typedef struct name##_s { \
    members \
} name##_t; \
typedef name##_t* name##_p;
```

Definition at line 115 of file macros.h.

8.38.1.42 #define THREAD_CANCEL(thread_handle) pthread_cancel(thread_handle)

Definition at line 140 of file macros.h.

Referenced by MC_End().

8.38.1.43 #define THREAD_CREATE(thread_handle, function, arg)**Value:**

```
pthread_create( \
    thread_handle, \
    &attr, \
    function, \
    (void*) arg \
)
```

Definition at line 132 of file macros.h.

Referenced by acc_Start(), agent_RunChScript(), ams_Start(), cmd_prompt_Start(), and df_Start().

8.38.1.44 #define THREAD_JOIN(thread_handle) pthread_join(thread_handle, NULL)

Definition at line 143 of file macros.h.

Referenced by MC_End().

8.38.1.45 #define THREAD_T pthread_t

Definition at line 131 of file macros.h.

Referenced by agent_RunChScript().

8.38.1.46 #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 278 of file macros.h.

8.38.1.47 #define WARN(message)**Value:**

```
fprintf(stderr, "WARNING: "); \
    fprintf(stderr, message ); \
    fprintf(stderr, " %s:%d\n", __FILE__, __LINE__ )
```

Definition at line 420 of file macros.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), and message_InitializeFromAgent().

8.39 /home/dko/projects/mobilec/trunk/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](#) }

8.39.1 Typedef Documentation

8.39.1.1 typedef enum [error_code_e](#) [error_code_t](#)

8.39.2 Enumeration Type Documentation

8.39.2.1 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 38 of file mc_error.h.

8.40 /home/dko/projects/mobilec/trunk/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 "../security/asm.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)
- [int](#) [mc_platform_Destroy](#) ([mc_platform_p](#) mc_platform)

8.40.1 Typedef Documentation

8.40.1.1 typedef [mc_platform_t](#)* [mc_platform_p](#)

Definition at line 109 of file [mc_platform.h](#).

8.40.1.2 typedef struct [mc_platform_s](#) [mc_platform_t](#)

Definition at line 108 of file [mc_platform.h](#).

8.40.2 Function Documentation

8.40.2.1 [int](#) [mc_platform_Destroy](#) ([mc_platform_p](#) *mc_platform*)

Definition at line 232 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::asm_message_queue`, `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::interp_options`, `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`, and `mc_platform_s::quit_lock`.

Referenced by `handler_QUIT()`, and `MC_End()`.

8.40.2.2 `mc_platform_p mc_platform_Initialize (MCAgency_t agency)`

Definition at line 44 of file `mc_platform.c`.

References `mc_platform_s::acc`, `acc_Initialize()`, `acc_Start()`, `mc_platform_s::agent_queue`, `mc_platform_s::ams`, `ams_Initialize()`, `ams_Start()`, `mc_platform_s::asm_message_queue`, `mc_platform_s::barrier_queue`, `barrier_queue_New()`, `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()`, `agency_s::enable_security`, `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`, `mc_platform_s::interp_options`, `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_lock`, `agency_s::stack_size`, `mc_platform_s::stack_size`, `mc_platform_s::syncList`, `syncListInit()`, and `agency_s::threads`.

Referenced by `MC_Initialize()`.

8.41 /home/dko/projects/mobilec/trunk/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*)

8.41.1 Typedef Documentation

8.41.1.1 typedef [mc_rwlock_t](#)* [mc_rwlock_p](#)

Definition at line 43 of file [mc_rwlock.h](#).

8.41.1.2 typedef struct [mc_rwlock_s](#) [mc_rwlock_t](#)

8.41.2 Function Documentation

8.41.2.1 [int mc_rwlock_destroy](#) ([mc_rwlock_p](#) *rwlock*)

Definition at line 58 of file [mc_rwlock.c](#).

References [mc_rwlock_s::cond](#), [COND_DESTROY](#), [mc_rwlock_s::lock](#), and [MUTEX_DESTROY](#).

8.41.2.2 [int mc_rwlock_init](#) ([mc_rwlock_p](#) *rwlock*)

Definition at line 38 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](#).

8.41.2.3 int mc_rwlock_rdlock (mc_rwlock_p *rwlock*)

Definition at line 71 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.

8.41.2.4 int mc_rwlock_rdunlock (mc_rwlock_p *rwlock*)

Definition at line 86 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.

8.41.2.5 int mc_rwlock_wrlock (mc_rwlock_p *rwlock*)

Definition at line 99 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.

8.41.2.6 int mc_rwlock_wrunlock (mc_rwlock_p *rwlock*)

Definition at line 118 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.

8.42 /home/dko/projects/mobilec/trunk/src/include/message.h File Reference

```
#include "config.h"
#include "mtp_http.h"
#include <mxml.h>
#include <netinet/in.h>
```

Data Structures

- struct [message_s](#)

Typedefs

- typedef enum [message_type_e](#) [message_type_t](#)
- typedef struct [message_s](#) [message_t](#)
- typedef [message_t](#) * [message_p](#)

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 }

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 [message_Send](#) ([message_p](#) message)
- int [http_to_hostport](#) (const char *http_str, char **host, int *port, char **target)

8.42.1 Typedef Documentation

8.42.1.1 typedef message_t* message_p

Definition at line 105 of file message.h.

8.42.1.2 `typedef struct message_s message_t`

8.42.1.3 `typedef enum message_type_e message_type_t`

8.42.2 Enumeration Type Documentation

8.42.2.1 `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

Definition at line 44 of file message.h.

8.42.3 Function Documentation

8.42.3.1 `int http_to_hostport (const char * http_str, char ** host, int * port, char ** target)`

Definition at line 502 of file message.c.

References MC_ERR_PARSE.

Referenced by MC_AclSend().

8.42.3.2 `message_p message_Copy (message_p src)`

Definition at line 307 of file message.c.

8.42.3.3 `int message_Destroy (message_p message)`

Definition at line 601 of file message.c.

References message_s::addr, message_s::agent_xml_flag, message_s::from_address, MC_SUCCESS, message_s::message_body, mxmDelete(), message_s::target, message_s::to_address, message_s::update_name, and message_s::xml_root.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `ams_ManageAgentList()`, `MC_AclSend()`, `MC_LoadAgentFromFile()`, `MC_SendAgentMigrationMessage()`, `MC_SendAgentMigrationMessageFile()`, `message_InitializeFromConnection()`, and `message_queue_SendOutgoing()`.

8.42.3.4 `int message_InitializeFromAgent (struct mc_platform_s * mc_platform, message_p message, struct agent_s * agent)`

Referenced by `ams_ManageAgentList()`.

8.42.3.5 `int message_InitializeFromConnection (struct mc_platform_s * mc_platform, message_p message, struct connection_s * connection)`

8.42.3.6 `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_SendAgentMigrationMessage()`, and `MC_SendAgentMigrationMessageFile()`.

8.42.3.7 `message_p message_New (void)`

Definition at line 283 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_Thread()`, `ams_ManageAgentList()`, `MC_LoadAgentFromFile()`, `MC_SendAgentMigrationMessage()`, `MC_SendAgentMigrationMessageFile()`, and `mtp_http_CreateMessage()`.

8.42.3.8 `int message_Send (message_p message)`

Definition at line 639 of file `message.c`.

References `CHECK_NULL`, `MC_ERR`, `MC_ERR_CONNECT`, `MC_ERR_SEND`, `message_s::message_body`, `mtp_http_ComposeMessage()`, `mtp_http_New()`, `mtp_http_Parse()`, `port`, `send`, `SOCKET_INPUT_SIZE`, `strtok_r`, and `message_s::to_address`.

Referenced by `acc_MessageHandlerThread()`, `MC_AclSend()`, and `message_queue_SendOutgoing()`.

8.43 /home/dko/projects/mobilec/trunk/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_ParseHeader (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)`
- `mtp_http_p mtp_http_New (void)`
- `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)`

8.43.1 Define Documentation

8.43.1.1 `#define SOCKET_INPUT_SIZE 4096`

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

Referenced by `message_InitializeFromConnection()`, `message_Send()`, and `mtp_http_InitializeFromConnection()`.

8.43.2 Typedef Documentation

8.43.2.1 `typedef struct mtp_http_content_s mtp_http_content_t`

8.43.2.2 `typedef mtp_http_t* mtp_http_p`

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

8.43.2.3 `typedef struct mtp_http_s mtp_http_t`

8.43.3 Enumeration Type Documentation

8.43.3.1 `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 87 of file mtp_http.h.

8.43.3.2 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
REQUST_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 42 of file mtp_http.h.

8.43.4 Function Documentation

8.43.4.1 **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 207 of file mtp_http.c.

Referenced by mtp_http_Parse().

8.43.4.2 **const char* http_GetToken (const char * *string*, char ** *token*)**

Definition at line 630 of file mtp_http.c.

Referenced by http_ParseHeader().

8.43.4.3 **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 266 of file mtp_http.c.

References CHECK_NULL, MC_ERR_PARSE, and MC_SUCCESS.

Referenced by mtp_http_Parse().

8.43.4.4 const char* http_ParseHeader (mtp_http_p *http*, const char * *string*)

Definition at line 541 of file mtp_http.c.

References 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_Parse().

8.43.4.5 int mtp_http_ComposeMessage (struct message_s * *message*)

Referenced by message_Send().

8.43.4.6 struct message_s* mtp_http_CreateMessage (mtp_http_t * *mtp_http*, char * *hostname*, int *port*) [read]

Definition at line 715 of file mtp_http.c.

References 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().

8.43.4.7 int mtp_http_Destroy (mtp_http_p *http*)

Definition at line 50 of file mtp_http.c.

References mtp_http_s::accept_ranges, mtp_http_s::boundary, 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::return_code, SAFE_FREE, mtp_http_s::server, mtp_http_s::target, and mtp_http_s::user_agent.

Referenced by acc_Thread(), and MC_AclSend().

8.43.4.8 int mtp_http_InitializeFromConnection (struct mtp_http_s * *http*, struct connection_s * *connection*)

Referenced by acc_Thread().

8.43.4.9 mtp_http_p mtp_http_New (void)

Definition at line 82 of file mtp_http.c.

References CHECK_NULL, and mtp_http_s::content.

Referenced by acc_Thread(), MC_AclSend(), and message_Send().

8.43.4.10 int mtp_http_Parse (struct mtp_http_s * *http*, const char * *string*)

Definition at line 326 of file mtp_http.c.

References mtp_http_s::boundary, mtp_http_s::content, mtp_http_content_s::content_type, mtp_http_s::content_type, mtp_http_content_s::data, http_GetExpression(), HTTP_PARSE_EXPR, http_ParseExpression(), http_ParseHeader(), mtp_http_s::http_performative, HTTP_POST, HTTP_PUT, HTTP_RESPONSE, MC_SUCCESS, mtp_http_s::message_parts, and SAFE_FREE.

Referenced by message_Send(), and mtp_http_InitializeFromConnection().

8.44 /home/dko/projects/mobilec/trunk/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__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 \(void *data, int index, int *extent, ChType_t type, int dim, int extent_index\)](#)

8.44.1 Function Documentation

8.44.1.1 mxml_node_t* agent_xml_compose (agent_p agent)

Definition at line 38 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()`.

8.44.1.2 mxml_node_t* agent_xml_compose__agent_code (agent_p agent, int index)

Definition at line 498 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()`.

8.44.1.3 mxml_node_t* agent_xml_compose__agent_data (agent_p agent)

Definition at line 142 of file `xml_compose.c`.

References `agent_xml_compose__home()`, `agent_xml_compose__name()`, `agent_xml_compose__owner()`, `agent_xml_compose__tasks()`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlNewElement()`, and `node`.

Referenced by `agent_xml_compose__mobile_agent()`.

8.44.1.4 **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 549 of file xml_compose.c.

References agent_xml_compose__create_row_nodes(), 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().

8.44.1.5 **mxml_node_t* agent_xml_compose__data** (agent_p *agent*, int *index*, interpreter_variable_data_t * *interp_variable*)

Definition at line 424 of file xml_compose.c.

References agent_xml_compose__row(), interpreter_variable_data_s::array_dim, 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().

8.44.1.6 **mxml_node_t* agent_xml_compose__gaf_message** (agent_p *agent*)

Definition at line 58 of file xml_compose.c.

References agent_xml_compose__message(), MXML_ADD_AFTER, mxmlAdd(), mxmlNewElement(), and node.

Referenced by agent_xml_compose().

8.44.1.7 **mxml_node_t* agent_xml_compose__home** (agent_p *agent*)

Definition at line 234 of file xml_compose.c.

References agent_s::home, mxmlNewElement(), mxmlNewText(), and node.

Referenced by agent_xml_compose__agent_data().

8.44.1.8 **mxml_node_t* agent_xml_compose__message** (agent_p *agent*)

Definition at line 77 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().

8.44.1.9 **mxml_node_t* agent_xml_compose__mobile_agent** (agent_p *agent*)

Definition at line 121 of file xml_compose.c.

References agent_xml_compose__agent_data(), MXML_ADD_AFTER, mxmlAdd(), mxmlNewElement(), and node.

Referenced by agent_xml_compose__message().

8.44.1.10 mxml_node_t* agent_xml_compose__name (agent_p agent)

Definition at line 202 of file xml_compose.c.

References mxmlNewElement(), mxmlNewText(), agent_s::name, and node.

Referenced by agent_xml_compose__agent_data().

8.44.1.11 mxml_node_t* agent_xml_compose__owner (agent_p agent)

Definition at line 218 of file xml_compose.c.

References mxmlNewElement(), mxmlNewText(), node, and agent_s::owner.

Referenced by agent_xml_compose__agent_data().

8.44.1.12 mxml_node_t* agent_xml_compose__row (interpreter_variable_data_t *interp_variable, int index)

Definition at line 526 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().

8.44.1.13 mxml_node_t* agent_xml_compose__task (agent_p agent, int index)

Definition at line 307 of file xml_compose.c.

References agent_task_s::agent_return_data, agent_task_s::agent_variable_list, agent_xml_compose__data(), 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_task_s::task_completed, agent_datastate_s::tasks, and agent_task_s::var_name.

Referenced by agent_xml_compose__tasks().

8.44.1.14 mxml_node_t* agent_xml_compose__tasks (agent_p agent)

Definition at line 250 of file xml_compose.c.

References agent_xml_compose__agent_code(), agent_xml_compose__task(), 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().

8.45 /home/dko/projects/mobilec/trunk/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*)
- const 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*)

8.45.1 Function Documentation

8.45.1.1 const char * [whitespace_cb](#) ([mxml_node_t](#) * *node*, [int](#) *where*)

Definition at line 568 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()`.

8.45.1.2 [mxml_node_t * xml_find_sibling](#) (const [mxml_node_t](#) * *node*, const char * *sibling_name*)

Definition at line 46 of file `xml_helper.c`.

References `MXML_NO_DESCEND`, `mxmlFindElement()`, `mxml_node_s::parent`, and `xml_get_element_name()`.

8.45.1.3 char* [xml_get_cdata](#) (const [mxml_node_t](#) * *node*)

Definition at line 68 of file `xml_helper.c`.

References `CHECK_NULL`, `MXML_ELEMENT`, `mxml_node_s::type`, and `xml_get_element_name()`.

Referenced by `xml_get_text()`.

8.45.1.4 [mxml_node_t * xml_get_child](#) (const [mxml_node_t](#) * *node*, const char * *child_name*, [int](#) *descend*)

Definition at line 101 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()`, and `xml_get_deep_child()`.

8.45.1.5 `mxml_node_t* xml_get_deep_child (const mxml_node_t * parent, const char ** child_path)`

Definition at line 120 of file `xml_helper.c`.

References `MXML_NO_DESCEND`, `node`, and `xml_get_child()`.

8.45.1.6 `const char* xml_get_element_name (const mxml_node_t * node)`

Definition at line 214 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()`, `xml_find_sibling()`, and `xml_get_cdata()`.

8.45.1.7 `mxml_node_t* xml_get_next_element (const mxml_node_t * node)`

Definition at line 134 of file `xml_helper.c`.

References `MXML_ELEMENT`, `mxml_node_s::next`, and `mxml_node_s::type`.

8.45.1.8 `char* xml_get_text (const mxml_node_t * node)`

Definition at line 152 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()`, and `agent_xml_parse__owner()`.

8.45.1.9 `mxml_node_t* xml_new_cdata (mxml_node_t * parent, const char * text)`

Definition at line 227 of file `xml_helper.c`.

References `CHECK_NULL`, `mxmlNewElement()`, and `node`.

Referenced by `agent_xml_compose__agent_code()`.

8.46 /home/dko/projects/mobilec/trunk/src/include/xml_parser.h File Reference

```
#include <mxml.h>
#include "macros.h"
#include "agent.h"
```

Functions

- [STRUCT](#) (xml_parser, const [mxml_node_t](#) *root; const [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__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)

8.46.1 Function Documentation

8.46.1.1 [error_code_t agent_xml_parse](#) ([agent_p](#) agent)

Definition at line 49 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\(\)](#).

8.46.1.2 [error_code_t agent_xml_parse__agent_code](#) ([agent_p](#) agent, [int](#) index, [xml_parser_p](#) xml_parser)

Definition at line 779 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\(\)](#).

8.46.1.3 error_code_t agent_xml_parse__agent_data (agent_p agent, xml_parser_p xml_parser)

Definition at line 90 of file xml_parser.c.

References agent_xml_parse__home(), agent_xml_parse__name(), agent_xml_parse__owner(), agent_xml_parse__tasks(), MC_ERR_PARSE, MC_SUCCESS, and xml_get_child().

Referenced by agent_xml_parse__mobile_agent().

8.46.1.4 error_code_t agent_xml_parse__data (agent_p agent, xml_parser_p xml_parser, int index)

Definition at line 454 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().

8.46.1.5 void agent_xml_parse__fill_row_data (void * data, ChType_t type, int * extent, const mxml_node_t * node, int * index)

Definition at line 663 of file xml_parser.c.

References agent_xml_parse__fill_row_data(), 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().

8.46.1.6 error_code_t agent_xml_parse__home (agent_p agent, xml_parser_p xml_parser)

Definition at line 198 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().

8.46.1.7 error_code_t agent_xml_parse__mobile_agent (agent_p agent, xml_parser_p xml_parser)

Definition at line 62 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().

8.46.1.8 error_code_t agent_xml_parse__name (agent_p agent, xml_parser_p xml_parser)

Definition at line 145 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()`.

8.46.1.9 `error_code_t agent_xml_parse__owner (agent_p agent, xml_parser_p xml_parser)`

Definition at line 171 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()`.

8.46.1.10 `error_code_t agent_xml_parse__row (interpreter_variable_data_t * interp_variable, xml_parser_p xml_parser, int index)`

Definition at line 599 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()`.

8.46.1.11 `error_code_t agent_xml_parse__task (agent_p agent, xml_parser_p xml_parser, int index)`

Definition at line 362 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`, `mxmElementGetAttr()`, `mxmFindElement()`, `agent_task_s::server_name`, `agent_task_s::task_completed`, `agent_datastate_s::tasks`, and `agent_task_s::var_name`.

Referenced by `agent_xml_parse__tasks()`.

8.46.1.12 `error_code_t agent_xml_parse__tasks (agent_p agent, xml_parser_p xml_parser)`

Definition at line 224 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()`, `agent_s::datastate`, `MC_ERR_PARSE`, `MXML_DESCEND`, `MXML_DESCEND_FIRST`, `MXML_NO_DESCEND`, `mxmElementGetAttr()`, `mxmFindElement()`, `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()`.

8.46.1.13 `error_code_t message_xml_parse (message_p message)`

Definition at line 855 of file `xml_parser.c`.

Referenced by `acc_Thread()`, and `message_InitializeFromConnection()`.

8.46.1.14 `error_code_t message_xml_parse__message (message_p message, xml_parser_p xml_parser)`

Definition at line 910 of file `xml_parser.c`.

8.46.1.15 `STRUCT (xml_parser, const mxml_node_t *root; const mxml_node_t *node;)`

8.47 /home/dko/projects/mobilec/trunk/src/libmc.c File Reference

```
#include <unistd.h>
#include <pthread.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"
```

Defines

- #define [HOST_NAME_MAX](#) 255

Functions

- [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.

- [int MC_AclSetPerformative](#) ([fipa_acl_message_t](#) *acl, enum [fipa_performative_e](#) performative)
- [int MC_AclSetSender](#) ([fipa_acl_message_t](#) *acl, const char *name, const char *address)
- [int MC_AclAddReceiver](#) ([fipa_acl_message_t](#) *acl, const char *name, const char *address)
- [int MC_AclAddReplyTo](#) ([fipa_acl_message_t](#) *acl, const char *name, const char *address)
- [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'.
- const void * [MC_AgentVariableRetrieve](#) ([MCAGENT_t](#) agent, const char *var_name, [int](#) task_num)
Retrieve a pointer to a previously saved variable.
- [int MC_AgentVariableSave](#) ([MCAGENT_t](#) agent, const char *var_name)
Mark an agent variable for saving.
- [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, void *varg)
Calls a function defined in an agent.
- EXPORTMC [int MC_ChInitializeOptions](#) ([MCAgency_t](#) attr, [ChOptions_t](#) *options)
Use custom ChOptions_t type for internal Ch interpreter.
- [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 [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.
- [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.

- `int MC_DestroyServiceSearchResult` (char **agentName, char **serviceName, int *agentID, int numResult)
Free memory allocated by a Service Search operation.
- `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.
- `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.
- `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_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.
- [int MC_SendCh](#) ([MCAgency_t](#) attr, const char *filename, const char *remotehost, [int](#) port)
- 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)
- [int MC_SetAgentStatus](#) ([MCAgent_t](#) agent, [int](#) status)
Set an agent's status.
- [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.
- `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.
- `int MC_AclDestroy_chdl (void *varg)`
- `void * MC_AclNew_chdl (void *varg)`
- `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_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 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 MC_Agent_t MC_ComposeAgent_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_DestroyServiceSearchResult_chdl (void *varg)
- EXPORTCH int MC_DeregisterService_chdl (void *varg)
- EXPORTCH int MC_End_chdl (void *varg)
- EXPORTCH MC_Agent_t MC_FindAgentByID_chdl (void *varg)
- EXPORTCH MC_Agent_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_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_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)

Variables

- [mc_platform_p g_mc_platform](#)

8.47.1 Define Documentation

8.47.1.1 #define HOST_NAME_MAX 255

Definition at line 58 of file libmc.c.

Referenced by MC_Initialize().

8.47.2 Function Documentation

8.47.2.1 int MC_AclAddReceiver (fipa_acl_message_t * *acl*, const char * *name*, const char * *address*)

Definition at line 226 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, and fipa_url_sequence_s::urls.

Referenced by MC_AclAddReceiver_chdl().

8.47.2.2 EXPORTCH int MC_AclAddReceiver_chdl (void * *varg*)

Definition at line 1836 of file libmc.c.

References MC_AclAddReceiver().

Referenced by agent_RunChScriptThread().

8.47.2.3 int MC_AclAddReplyTo (fipa_acl_message_t * *acl*, const char * *name*, const char * *address*)

Definition at line 262 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, and fipa_url_sequence_s::urls.

Referenced by MC_AclAddReplyTo_chdl().

8.47.2.4 EXPORTCH int MC_AclAddReplyTo_chdl (void * *varg*)

Definition at line 1856 of file libmc.c.

References MC_AclAddReplyTo().

Referenced by agent_RunChScriptThread().

8.47.2.5 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 69 of file libmc.c.

References fipa_acl_message_Destroy().

Referenced by MC_AclDestroy_chdl().

8.47.2.6 int MC_AclDestroy_chdl (void * *varg*)

Definition at line 1683 of file libmc.c.

References MC_AclDestroy().

Referenced by agent_RunChScriptThread().

8.47.2.7 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 75 of file libmc.c.

References fipa_acl_message_New().

Referenced by MC_AclNew_chdl().

8.47.2.8 void* MC_AclNew_chdl (void * *varg*)

Definition at line 1698 of file libmc.c.

References MC_AclNew().

Referenced by agent_RunChScriptThread().

8.47.2.9 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 80 of file `libmc.c`.

References `agent_mailbox_Post()`, and `agent_s::mailbox`.

Referenced by `MC_AclPost_chdl()`, and `MC_AclSend()`.

8.47.2.10 int MC_AclPost_chdl (void * *varg*)

Definition at line 1706 of file `libmc.c`.

References `MC_AclPost()`.

Referenced by `agent_RunChScriptThread()`.

8.47.2.11 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 86 of file `libmc.c`.

References `fipa_Reply()`.

Referenced by `MC_AclReply_chdl()`.

8.47.2.12 EXPORTCH void* MC_AclReply_chdl (void * *varg*)

Definition at line 1724 of file `libmc.c`.

References `MC_AclReply()`.

Referenced by `agent_RunChScriptThread()`.

8.47.2.13 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 92 of file libmc.c.

References agent_mailbox_Retrieve(), and agent_s::mailbox.

Referenced by MC_AclRetrieve_chdl().

8.47.2.14 EXPORTCH void* MC_AclRetrieve_chdl (void * varg)

Definition at line 1740 of file libmc.c.

References MC_AclRetrieve().

Referenced by agent_RunChScriptThread().

8.47.2.15 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 98 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_agent_identifier_set_s::fipa_agent_identifiers, fipa_envelope_Compose(), mtp_http_s::host, http_to_hostport(), MC_AclPost(), MC_FindAgentByName(), dynstring_s::message, message_Destroy(), 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, fipa_acl_message_s::receiver, fipa_url_s::str, mtp_http_s::target, message_s::target, and fipa_url_sequence_s::urls.

Referenced by MC_AclSend_chdl().

8.47.2.16 EXPORTCH int MC_AclSend_chdl (void * varg)

Definition at line 1756 of file libmc.c.

References CHECK_NULL, MC_AclSend(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.47.2.17 int MC_AclSetContent (fipa_acl_message_t * acl, const char * content)

Definition at line 297 of file libmc.c.

References `fipa_string_s::content`, `fipa_acl_message_s::content`, `fipa_string_Destroy()`, and `fipa_string_New()`.

Referenced by `MC_AclSetContent_chdl()`.

8.47.2.18 EXPORTCH int MC_AclSetContent_chdl (void * *varg*)

Definition at line 1876 of file `libmc.c`.

References `fipa_acl_message_s::content`, and `MC_AclSetContent()`.

Referenced by `agent_RunChScriptThread()`.

8.47.2.19 int MC_AclSetPerformative (fipa_acl_message_t * *acl*, enum fipa_performative_e *performative*)

Definition at line 195 of file `libmc.c`.

References `fipa_acl_message_s::performative`.

Referenced by `MC_AclSetPerformative_chdl()`.

8.47.2.20 EXPORTCH int MC_AclSetPerformative_chdl (void * *varg*)

Definition at line 1798 of file `libmc.c`.

References `MC_AclSetPerformative()`, and `fipa_acl_message_s::performative`.

Referenced by `agent_RunChScriptThread()`.

8.47.2.21 int MC_AclSetSender (fipa_acl_message_t * *acl*, const char * *name*, const char * *address*)

Definition at line 203 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`, and `fipa_url_sequence_s::urls`.

Referenced by `MC_AclSetSender_chdl()`.

8.47.2.22 EXPORTCH int MC_AclSetSender_chdl (void * *varg*)

Definition at line 1816 of file `libmc.c`.

References `MC_AclSetSender()`.

Referenced by `agent_RunChScriptThread()`.

8.47.2.23 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 188 of file libmc.c.

References agent_mailbox_WaitRetrieve(), and agent_s::mailbox.

Referenced by MC_AclWaitRetrieve_chdl().

8.47.2.24 EXPORTCH void* MC_AclWaitRetrieve_chdl (void * *varg*)

Definition at line 1780 of file libmc.c.

References MC_AclWaitRetrieve().

Referenced by agent_RunChScriptThread().

8.47.2.25 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 314 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 MC_AddAgent_chdl().

8.47.2.26 EXPORTCH int MC_AddAgent_chdl (void * *varg*)

Definition at line 1896 of file libmc.c.

References CHECK_NULL, MC_AddAgent(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.47.2.27 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.

8.47.3 Examples

The following example demonstrates usage of [MC_AgentVariableRetrieve\(\)](#) from agent space.

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>mobagent1</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="2" num="0">
          <TASK num="0" complete="0" server="localhost:5051" code_id="1" />
          <TASK num="1" complete="0" server="localhost:5050" code_id="2" />
        <AGENT_CODE id="1">
          <![CDATA[
//#include <stdio.h>
#include <math.h>
int savevar;
int another_savevar;
int array_savevar[10];
int main()
{
    int i;
    printf("Hello World!\n");
    printf("This is mobagent1 from the agency at port 5050.\n");
    printf("I am performing the task on the agency at port 5051 now.\n");
    printf("%f\n", hypot(1,2));
    savevar = 10;
    another_savevar = 20;
    mc_AgentVariableSave(mc_current_agent, "savevar");
    mc_AgentVariableSave(mc_current_agent, "another_savevar");
    for(i = 0; i < 10; i++) {
        array_savevar[i] = i*3;
    }
    mc_AgentVariableSave(mc_current_agent, "array_savevar");
    return 0;
}

]]>
      </AGENT_CODE>
    <AGENT_CODE id="2">
      <![CDATA[
#include <stdio.h>
int retvar;
int main()
{
    const int *i;
    i = (int*)mc_AgentVariableRetrieve(mc_current_agent, "savevar", 0);
    if (i==NULL) {
        printf("Variable 'savevar' not found.\n");
    } else {
        printf("Variable 'savevar' has value %d.\n", *i);
    }
    retvar = *i*2;
    return 0;
}

]]>
    </AGENT_CODE>
  </MESSAGE>
</MOBILE_AGENT>
</MOBILEC_MESSAGE>
```

```

    </AGENT_CODE>
  </TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 327 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()`.

8.47.3.1 EXPORTCH `const void* MC_AgentVariableRetrieve_chdl (void * varg)`

Definition at line 1920 of file libmc.c.

References `MC_AgentVariableRetrieve()`.

Referenced by `agent_RunChScriptThread()`.

8.47.3.2 `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`

8.47.4 Examples

See `agent_saved_variables_example/test1.xml` for an example of usage of this api function.

8.47.5 Examples

The following example demonstrates usage of `MC_AgentVariableSave()` from agent space.

```

<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>mobagent1</NAME>

```

```

<OWNER>IEL</OWNER>
<HOME>localhost:5050</HOME>
<TASKS task="2" num="0">
  <TASK num="0" complete="0" server="localhost:5051" code_id="1" />
  <TASK num="1" complete="0" server="localhost:5050" code_id="2" />
  <AGENT_CODE id="1">
    <![CDATA[
//#include <stdio.h>
#include <math.h>
int savevar;
int another_savevar;
int array_savevar[10];
int main()
{
  int i;
  printf("Hello World!\n");
  printf("This is mobagent1 from the agency at port 5050.\n");
  printf("I am performing the task on the agency at port 5051 now.\n");
  printf("%f\n", hypot(1,2));
  savevar = 10;
  another_savevar = 20;
  mc_AgentVariableSave(mc_current_agent, "savevar");
  mc_AgentVariableSave(mc_current_agent, "another_savevar");
  for(i = 0; i < 10; i++) {
    array_savevar[i] = i*3;
  }
  mc_AgentVariableSave(mc_current_agent, "array_savevar");
  return 0;
}
]]>
</AGENT_CODE>
<AGENT_CODE id="2">
  <![CDATA[
#include <stdio.h>
int retvar;
int main()
{
  const int *i;
  i = (int*)mc_AgentVariableRetrieve(mc_current_agent, "savevar", 0);
  if (i==NULL) {
    printf("Variable 'savevar' not found.\n");
  } else {
    printf("Variable 'savevar' has value %d.\n", *i);
  }
  retvar = *i*2;
  return 0;
}
]]>
</AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 345 of file libmc.c.

References agent_s::datastate, MC_ERR_MEMORY, agent_task_s::num_saved_variables, agent_task_s::saved_variables, agent_datastate_s::task_progress, and agent_datastate_s::tasks.

Referenced by MC_AgentVariableSave_chdl().

8.47.5.1 EXPORTCH int MC_AgentVariableSave_chdl (void * varg)

Definition at line 1944 of file libmc.c.

References MC_AgentVariableSave().

Referenced by agent_RunChScriptThread().

8.47.5.2 int MC_Barrier (MCAgency_t *attr*, int *id*)

Definition at line 366 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().

8.47.5.3 EXPORTCH int MC_Barrier_chdl (void * *varg*)

Definition at line 1996 of file libmc.c.

References CHECK_NULL, MC_Barrier(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.47.5.4 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 406 of file libmc.c.

References mc_platform_s::barrier_queue, barrier_queue_Delete(), and agency_s::mc_platform.

Referenced by MC_BarrierDelete_chdl().

8.47.5.5 EXPORTCH int MC_BarrierDelete_chdl (void * *varg*)

Definition at line 2019 of file libmc.c.

References CHECK_NULL, MC_BarrierDelete(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.47.5.6 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.

8.47.6 Examples

The following example demonstrates an agent which sets up an MC_Barrier.

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>CommAgent1</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5051</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5050">
            <DATA number_of_elements="0" name="no-return" >
              </DATA>
            </TASK>
          </TASKS>
          <AGENT_CODE>
            <![CDATA[
#include <stdio.h>
#define MC_BARRIER_ID 56
#define NUM_PROCS 2
int main()
{
    printf("Comm agent: Setting up the MC_Barriers now...\n");
    mc_BarrierInit(MC_BARRIER_ID, NUM_PROCS);
    mc_BarrierInit(MC_BARRIER_ID + 1, NUM_PROCS);
    mc_BarrierInit(MC_BARRIER_ID + 2, NUM_PROCS);
    printf("MC_Barrier Initialized.\n");
    return 0;
}
]]>
          </AGENT_CODE>
        </TASKS>
      </AGENT_DATA>
    </MOBILE_AGENT>
  </MESSAGE>
</MOBILEC_MESSAGE>
```

Definition at line 392 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()`.

8.47.6.1 EXPORTCH int MC_BarrierInit_chdl (void *varg)

Definition at line 2042 of file libmc.c.

References `CHECK_NULL`, `MC_BarrierInit()`, and `agency_s::mc_platform`.

Referenced by `agent_RunChScriptThread()`.

8.47.6.2 EXPORTMC int MC_CallAgentFunc (MCAgent_t *agent*, const char * *funcName*, void * *returnVal*, void * *varg*)

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
varg 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

8.47.7 Example

Definition at line 412 of file libmc.c.

References agent_s::agent_interp, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::run_lock.

Referenced by MC_CallAgentFunc_chdl().

8.47.7.1 EXPORTCH int MC_CallAgentFunc_chdl (void * *varg*)

Definition at line 1966 of file libmc.c.

References MC_CallAgentFunc().

Referenced by agent_RunChScriptThread().

8.47.7.2 EXPORTMC int MC_ChInitializeOptions (MCAgency_t *attr*, ChOptions_t * *options*)

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

Definition at line 430 of file libmc.c.

References mc_platform_s::interp_options, and agency_s::mc_platform.

8.47.7.3 **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.

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 448 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_datastate_s::persistent, agent_task_s::server_name, agent_datastate_s::tasks, and agent_task_s::var_name.

Referenced by MC_ComposeAgent_chdl().

8.47.7.4 **EXPORTCH MCAgent_t MC_ComposeAgent_chdl** (void * *varg*)

Definition at line 2090 of file libmc.c.

References MC_ComposeAgent().

Referenced by agent_RunChScriptThread().

8.47.7.5 **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 518 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()`.

8.47.7.6 EXPORTCH int MC_CondBroadcast_chdl (void * *varg*)

Definition at line 2067 of file libmc.c.

References `CHECK_NULL`, `MC_CondBroadcast()`, and `agency_s::mc_platform`.

Referenced by `agent_RunChScriptThread()`.

8.47.7.7 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 570 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()`.

8.47.7.8 EXPORTCH int MC_CondReset_chdl (void * *varg*)

Definition at line 2153 of file libmc.c.

References `CHECK_NULL`, `MC_CondReset()`, and `agency_s::mc_platform`.

Referenced by `agent_RunChScriptThread()`.

8.47.7.9 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

8.47.8 Example

The following example demonstrates the agent-space version of the function, which is nearly identical to the binary space api function.

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>wake_agent</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5051">
            </TASK>
        </TASKS>
        <AGENT_CODE>
          <![CDATA[
#include <stdio.h>

#define SYNC_ID 55
int main()
{
    int cond_id;
    cond_id = SYNC_ID;
    printf("This is the wake agent.\n");
    mc_CondSignal(cond_id);

    return 0;
}
]]>
      </AGENT_CODE>
    </AGENT_DATA>
  </MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>
```

Definition at line 533 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()`.

8.47.8.1 EXPORTCH int MC_CondSignal_chdl (void * *varg*)

Definition at line 2130 of file libmc.c.

References `CHECK_NULL`, `MC_CondSignal()`, and `agency_s::mc_platform`.

Referenced by agent_RunChScriptThread().

8.47.8.2 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

8.47.9 Example

The following example demonstrates the agent-space version of this function.

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>sleep_agent</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5051">
            </TASK>
        </TASKS>
        <AGENT_CODE>
          <![CDATA[
#include <stdio.h>

#define SYNC_ID 55
int main()
{
    int cond_id;
    printf("Sleep agent has arrived.\n");
    cond_id = mc_SyncInit(SYNC_ID);
    if (cond_id != SYNC_ID) {
        printf("Possible error. Aborting...\n");
        exit(1);
    }
    printf("This is the sleep agent.\n");
    printf("I am going to sleep now...\n");
    mc_CondWait(cond_id);
    printf("This is the sleep agent: I am awake now. Continuing...\n");
    mc_SyncDelete(cond_id);

    return 0;
}

]]>
</AGENT_CODE>
```

```

    </TASKS>
    </AGENT_DATA>
    </MOBILE_AGENT>
    </MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 548 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().

8.47.9.1 EXPORTCH int MC_CondWait_chdl (void * *varg*)

Definition at line 2176 of file libmc.c.

References CHECK_NULL, MC_CondWait(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.47.9.2 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 588 of file libmc.c.

References agent_Copy(), and MC_SUCCESS.

8.47.9.3 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 595 of file libmc.c.

References CHECK_NULL, MC_ERR_INVALID, MC_SetAgentStatus(), MC_SUCCESS, MC_TerminateAgent(), and MC_WAIT_FINISHED.

Referenced by MC_DeleteAgent_chdl().

8.47.9.4 EXPORTCH int MC_DeleteAgent_chdl (void * *varg*)

Definition at line 2198 of file libmc.c.

References MC_DeleteAgent().

Referenced by agent_RunChScriptThread().

8.47.9.5 int MC_DeregisterService (MCAgency_t *agency*, int *agentID*, const char * *serviceName*)

Definition at line 629 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().

8.47.9.6 EXPORTCH int MC_DeregisterService_chdl (void * *varg*)

Definition at line 2240 of file libmc.c.

References CHECK_NULL, MC_DeregisterService(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.47.9.7 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 609 of file libmc.c.

Referenced by MC_DestroyServiceSearchResult_chdl().

8.47.9.8 EXPORTCH int MC_DestroyServiceSearchResult_chdl (void * *varg*)

Definition at line 2213 of file libmc.c.

References MC_DestroyServiceSearchResult().

Referenced by agent_RunChScriptThread().

8.47.9.9 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

8.47.10 Example

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

int main(int argc, char *argv[])
{
    MCAgency_t agency;
    MCAgencyOptions_t options;
    int port=5050;
    int remote_port = 5051;

    MC_InitializeAgencyOptions(&options);
    MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */
    agency = MC_Initialize(port, &options);

    printf("Mobile-C Started\n");
    /* Note: The third argument of the following function may also be a
       valid IP address in the form of a string. i.e. 192.168.0.1 */
    MC_SendAgentMigrationMessageFile(agency,
        "test1.xml",
        "localhost",
        remote_port);
    MC_End(agency);
    exit(0);
}
```

Definition at line 658 of file libmc.c.

References mc_platform_s::acc, mc_platform_s::ams, mc_platform_s::cmd_prompt, 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, mc_platform_s::quit, mc_platform_s::quit_lock, cmd_prompt_s::thread, THREAD_CANCEL, THREAD_JOIN, and agency_s::threads.

Referenced by MC_End_chdl().

8.47.10.1 EXPORTCH int MC_End_chdl (void * varg)

Definition at line 2267 of file libmc.c.

References CHECK_NULL, MC_End(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.47.10.2 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 712 of file libmc.c.

References mc_platform_s::agent_queue, and agency_s::mc_platform.

Referenced by MC_FindAgentByID_chdl().

8.47.10.3 EXPORTCH MCAgent_t MC_FindAgentByID_chdl (void * *varg*)

Definition at line 2284 of file libmc.c.

References CHECK_NULL, MC_FindAgentByID(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.47.10.4 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

8.47.11 Example

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

int main(int argc, char *argv[])
{
    MCAgency_t agency;
    MCAgencyOptions_t options;
    MCAgent_t agent;
    int dim, *extent;
    double *data;
    int i, j, size;
    int port=5050;
    int remote_port=5051;

    MC_InitializeAgencyOptions(&options);
```

```

MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */
agency = MC_Initialize(port, &options);

printf("MobileC Started\n");
/* Note: The third argument of the following function may also be a
   valid IP address in the form of a string. i.e. 192.168.0.1 */
MC_SendAgentMigrationMessageFile(agency,
    "test.xml",
    "localhost",
    remote_port);

/* Wait for return-agent arrival signal */
MC_WaitSignal(agency, MC_RECV_RETURN);

/* Make sure we caught the correct agent */
agent = MC_FindAgentByName(agency, "mobagent3");
if (agent == NULL) {
    fprintf(stderr, "Did not receive correct agent. \n");
    exit(1);
}

/* Print relevant information */
printf("%d tasks.\n", MC_GetAgentNumTasks(agent) );
for (i = 0; i < MC_GetAgentNumTasks(agent); i++) {
    MC_GetAgentReturnData(
        agent,
        i,
        (void**)&data,
        &dim,
        &extent );
    printf("Task: %d\n", i);
    size = 1;
    printf("dim is %d\n", dim);
    for (j = 0; j < dim; j++) {
        size *= extent[j];
    }
    printf("Size: %d\n", size);
    printf("Data elements: ");
    for (j = 0; j < size; j++) {
        printf("%f ", data[j]);
    }
    printf("\n\n");
    free(data);
    free(extent);
}

/* We must reset the signal that we previously caught with the
   * MC_WaitSignal() function with MC_ResetSignal() */
MC_ResetSignal(agency);

MC_End(agency);
return 0;
}

```

Definition at line 699 of file libmc.c.

References mc_platform_s::agent_queue, and agency_s::mc_platform.

Referenced by MC_AclSend(), and MC_FindAgentByName_chdl().

8.47.11.1 EXPORTCH MCAgent_t MC_FindAgentByName_chdl (void * varg)

Definition at line 2307 of file libmc.c.

References CHECK_NULL, MC_FindAgentByName(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.47.11.2 time_t MC_GetAgentArrivalTime (MCAgent_t agent)

Definition at line 728 of file libmc.c.

References agent_s::arrival_time.

Referenced by MC_GetAgentArrivalTime_chdl().

8.47.11.3 EXPORTCH time_t MC_GetAgentArrivalTime_chdl (void * varg)

Definition at line 2333 of file libmc.c.

References MC_GetAgentArrivalTime().

8.47.11.4 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 764 of file libmc.c.

References agent_s::agent_interp.

8.47.11.5 EXPORTMC int MC_GetAgentID (MCAgent_t agent)

Retrieve an agent's id.

Definition at line 770 of file libmc.c.

References agent_s::id.

8.47.11.6 EXPORTCH int MC_GetAgentID_chdl (void * varg)

Definition at line 2352 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.11.7 EXPORTMC char* MC_GetAgentName (MCAgent_t agent)

Definition at line 778 of file libmc.c.

References agent_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::name.

8.47.11.8 EXPORTCH char* MC_GetAgentName_chdl (void * varg)

Definition at line 2368 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.11.9 EXPORTMC int MC_GetAgentNumTasks (MCAgent_t agent)

Retrieve the number of tasks an agent has.

8.47.12 Example

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

int main(int argc, char *argv[])
{
    MCAgency_t agency;
    MCAgencyOptions_t options;
    MCAgent_t agent;
    int dim, *extent;
    double *data;
    int i, j, size;
    int port=5050;
    int remote_port=5051;

    MC_InitializeAgencyOptions(&options);
    MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */
    agency = MC_Initialize(port, &options);

    printf("MobileC Started\n");
    /* Note: The third argument of the following function may also be a
       valid IP address in the form of a string. i.e. 192.168.0.1 */
    MC_SendAgentMigrationMessageFile(agency,
        "test.xml",
        "localhost",
        remote_port);

    /* Wait for return-agent arrival signal */
    MC_WaitSignal(agency, MC_RECV_RETURN);

    /* Make sure we caught the correct agent */
    agent = MC_FindAgentByName(agency, "mobagent3");
    if (agent == NULL) {
        fprintf(stderr, "Did not receive correct agent. \n");
        exit(1);
    }

    /* Print relevant information */
    printf("%d tasks.\n", MC_GetAgentNumTasks(agent) );
    for (i = 0; i < MC_GetAgentNumTasks(agent); i++) {
        MC_GetAgentReturnData(
            agent,
            i,
            (void*)&data,
            &dim,
            &extent );
        printf("Task: %d\n", i);
        size = 1;
        printf("dim is %d\n", dim);
        for (j = 0; j < dim; j++) {
            size *= extent[j];
        }
    }
}
```

```

    printf("Size: %d\n", size);
    printf("Data elements: ");
    for (j = 0; j < size; j++) {
        printf("%f ", data[j]);
    }
    printf("\n\n");
    free(data);
    free(extent);
}

/* We must reset the signal that we previously caught with the
 * MC_WaitSignal() function with MC_ResetSignal() */
MC_ResetSignal(agency);

MC_End(agency);
return 0;
}

```

Definition at line 859 of file libmc.c.

References `agent_s::datastate`, and `agent_datastate_s::number_of_tasks`.

8.47.12.1 EXPORTCH int MC_GetAgentNumTasks_chdl (void * *varg*)

Definition at line 2384 of file libmc.c.

8.47.12.2 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.

8.47.13 Example

This file demonstrates the retrieval of agent return data from an agent

```

/* mc_sample_app.c
 *
 * This sample program uses the Mobile C library to build
 * a simple command-line driven client/server app.
 *
 * 12/15/2006
 * */

#include <libmc.h>
#include <stdio.h>
#ifdef _WIN32
#include <windows.h>
#endif
int main(int argc, char *argv[])

```

```

{
    MCAgency_t agency;
    MCAgent_t agent;
    int *extent;
    int dim;
    short *array;
    int i, size;
    char *xml;
    MCAgencyOptions_t options;
    int port = 5051;
    int remote_port = 5050;

    MC_InitializeAgencyOptions(&options);
    MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */
    if (argc == 2) {
        printf("Starting agency listening on port %d.\n",
            atoi(argv[1]) );
        agency = MC_Initialize(
            atoi(argv[1]),
            &options
        );
    } else {
        agency = MC_Initialize(
            port,
            &options);
    }
    /* Note: The third argument of the following function may also be a
    valid IP address in the form of a string. i.e. 192.168.0.1 */
    MC_SendAgentMigrationMessageFile(
        agency,
        "agent.xml",
        "localhost",
        remote_port);
    while(1) {
        printf("Waiting for agent return...\n");
        MC_WaitSignal(
            agency,
            MC_RECV_RETURN
        );
        agent = MC_FindAgentByName(agency, "mobagent1");
        if (agent == NULL) {
            printf("Found wrong agent. Waiting again...\n");
            MC_ResetSignal(agency);
            continue;
        }
        if (MC_GetAgentType(agent) != MC_RETURN_AGENT) {
            printf("Found wrong agent. Waiting again...\n");
            MC_ResetSignal(agency);
            continue;
        } else {
            break;
        }
    }
    /* Get and print the agent's XML code */
    xml = MC_GetAgentXMLString(agent);
    printf("%s\n", xml);
    free(xml);

    /* Retrieve the agent's return data */
    MC_GetAgentReturnData(agent, 0, (void**)&array, &dim, &extent);

    /* Since we have previously called MC_WaitSignal(), we must now reset
    * the signal to allow the agency to continue. */
    MC_ResetSignal(agency);

    /* Print relevant data */
    printf("dim is %d\n", dim);

```



```

    printf("Extents are: ");
    size = 1;
    for (i = 0; i < dim; i++) {
        printf("%d ", extent[i]);
        size *= extent[i];
    }
    printf("\n");
    printf("%d Elements: \n", size);
    for (i = 0; i < size; i++) {
        printf("%d ", array[i]);
    }
    printf("\n");

    MC_End(agency);

    return 0;
}

```

This is the agent which gets the data

```

<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>mobagent1</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5051</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5050" return="a">
            <AGENT_CODE>
              <![CDATA[
#include <stdio.h>
short a[2][3][2];
int main()
{
    int i, j, k, l;
    k = 0;
    for (i = 0; i < 2; i++) {
        for (j = 0; j < 3; j++) {
            for(l = 0; l < 2; l++) {
                a[i][j][l] = k;
                k++;
                printf("%d ", i+j);
            }
        }
    }
    printf("\nThis is a mobile agent from port 5050.\n");
    printf("I am performing the task on the agency at port 5051 now.\n");
    sleep(1);

    return 0;
}

]]>
      </AGENT_CODE>
    </TASKS>
  </AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 796 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`.

8.47.13.1 EXPORTMC int MC_GetAgentStatus (MCAgent_t agent)

Get an agent's current status.

Returns:

returns type 'enum MC_AgentStatus_e'

Definition at line 742 of file libmc.c.

References `agent_s::agent_status`, `agent_s::lock`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

8.47.13.2 EXPORTCH int MC_GetAgentStatus_chdl (void * varg)

Definition at line 2400 of file libmc.c.

Referenced by `agent_RunChScriptThread()`.

8.47.13.3 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 865 of file libmc.c.

References `agent_s::agent_type`.

8.47.13.4 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 752 of file libmc.c.

References `agent_s::datastate`, `mxmlSaveAllocString()`, and `agent_datastate_s::xml_agent_root`.

8.47.13.5 EXPORTCH char* MC_GetAgentXMLString_chdl (void * varg)

Definition at line 2416 of file libmc.c.

Referenced by `agent_RunChScriptThread()`.

8.47.13.6 int MC_GetAllAgents (MCAgency_t *attr*, MCAgent_t ** *agents*, int * *num_agents*)

Definition at line 875 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.

8.47.13.7 EXPORTCH int MC_GetTimeOfDay_chdl (void * *varg*)

Definition at line 2432 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.13.8 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 911 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().

8.47.13.9 EXPORTCH int MC_HaltAgency_chdl (void * *varg*)

Definition at line 2447 of file libmc.c.

References CHECK_NULL, MC_HaltAgency(), and agency_s::mc_platform.

Referenced by agent_RunChScriptThread().

8.47.13.10 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

8.47.14 Example

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

#ifdef _WIN32
#include <windows.h>
#endif

int main(int argc, char *argv[])
{
    MCAgency_t agency;
    int local_port = 5051;

    agency = MC_Initialize(local_port, NULL);

    printf("Mobile-C Started\n");

    MC_MainLoop(agency);

    MC_End(agency);
    return 0;
}
```

Definition at line 920 of file libmc.c.

References CHECK_NULL, agency_s::client, MCAgencyOptions_s::default_agent_status, agency_s::default_agentstatus, MCAgencyOptions_s::enable_security, agency_s::enable_security, HOST_NAME_MAX, agency_s::hostName, MC_InitializeAgencyOptions(), agency_s::mc_platform, mc_platform_Initialize(), MC_THREAD_ALL, agency_s::portno, agency_s::server, MCAgencyOptions_s::stack_size, agency_s::stack_size, MCAgencyOptions_s::threads, and agency_s::threads.

8.47.14.1 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.

8.47.15 Example

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

#ifdef _WIN32
#include <windows.h>
#endif

int main(int argc, char *argv[])
{
    MCAgency_t agency;
```

```

int local_port = 5051;

agency = MC_Initialize(local_port, NULL);

printf("Mobile-C Started\n");

MC_MainLoop(agency);

MC_End(agency);
return 0;
}

```

Definition at line 975 of file libmc.c.

References MCAgencyOptions_s::default_agent_status, MCAgencyOptions_s::enable_security, MC_THREAD_ALL, MC_WAIT_CH, MCAgencyOptions_s::modified, MCAgencyOptions_s::stack_size, and MCAgencyOptions_s::threads.

Referenced by MC_Initialize().

8.47.15.1 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 992 of file libmc.c.

References 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.

8.47.15.2 int MC_MainLoop (MCAgency_t attr)

Wait indefinitely.

Note:

This function is intended to block the calling thread forever.

Definition at line 1617 of file libmc.c.

References mc_platform_s::ams, and agency_s::mc_platform.

8.47.15.3 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

8.47.16 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.

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>sleep_agent</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5051">
            <DATA dim="0" name="no-return">
              </DATA>
            </TASK>
          </TASKS>
        <AGENT_CODE>
          <![CDATA[
#include <stdio.h>
int main()
{
    int mutex_id;
    printf("Sleep agent has arrived.\n");
    mutex_id = mc_SyncInit(55);
    if (mutex_id != 55) {
        printf("Possible error. Aborting...\n");
        exit(1);
    }
    printf("This is agent 1.\n");
    printf("Agent 1: I am locking the mutex now.\n");
    mc_MutexLock(mutex_id);
    printf("Agent 1: Mutex locked. Perform protected operations here\n");
    printf("Agent 1: Waiting for 5 seconds...\n");
    sleep(5);
    printf("Agent 1: Unlocking mutex now...\n");
    mc_MutexUnlock(mutex_id);

    return 0;
}
          ]]>
        </AGENT_CODE>
      </TASKS>
    </AGENT_DATA>
  </MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
```

```

<MOBILE_AGENT>
<AGENT_DATA>
  <NAME>wake_agent</NAME>
  <OWNER>IEL</OWNER>
  <HOME>localhost:5050</HOME>
  <TASKS task="1" num="0">
    <TASK num="0" complete="0" server="localhost:5051">
      <DATA dim="0" name="no-return" complete="0" server="localhost:5051">
        </DATA>
      </TASK>
    </TASKS>
  <AGENT_CODE>
    <![CDATA[
#include <stdio.h>
int main()
{
    int mutex_id;
    mutex_id = 55;
    printf("Agent 2: Has arrived");
    printf("Agent 2: Attempting to lock the mutex...\n");
    mc_MutexLock(mutex_id);
    printf("Agent 2: Mutex locked.\n");
    printf("Agent 2: Perform protected operations here.\n");
    sleep(5);
    mc_MutexUnlock(mutex_id);
    printf("Agent 2: Mutex Unlocked\n");
    mc_SyncDelete(mutex_id);

    return 0;
}

]]>
  </AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 1062 of file libmc.c.

References syncListNode_s::lock, agency_s::mc_platform, MUTEX_LOCK, mc_platform_s::syncList, and syncListFind().

8.47.16.1 EXPORTCH int MC_MutexLock_chdl (void * *varg*)

Definition at line 2465 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.16.2 EXPORTMC int MC_MutexUnlock (MCAgency_t *attr*, int *id*)

Definition at line 1074 of file libmc.c.

References syncListNode_s::lock, agency_s::mc_platform, MUTEX_UNLOCK, mc_platform_s::syncList, and syncListFind().

8.47.16.3 EXPORTCH int MC_MutexUnlock_chdl (void * *varg*)

Definition at line 2489 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.16.4 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 1086 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.

8.47.16.5 EXPORTCH int MC_PrintAgentCode_chdl (void * varg)

Definition at line 2513 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.16.6 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)'.

8.47.17 Example

```
<?xml version="1.0"?>
<!DOCTYPE myMessage SYSTEM "mobilec.dtd">
<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>service_provider_1</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5051">
```



```

        <DATA persistent="1" number_of_elements="0"
            name="no-return" >
    </DATA>
</TASK>
<AGENT_CODE>
    <![CDATA[
#include <stdio.h>

struct arg_struct {
    int a;
    int b;
};

int main() {
    char **services;
    int i;
    services = malloc(sizeof(char*)*2);
    for(i = 0; i < 2; i++) {
        services[i] = malloc(40);
    }
    strcpy(services[0], "addition");
    strcpy(services[1], "subtraction");
    printf("Service provider 1 has arrived.\n");
    printf("I provide addition and subtraction service.\n");
    mc_RegisterService( mc_current_agent, services, 2);
    return 0;
}

int addition(struct arg_struct* arg) {
    printf("Adding %d and %d...\n", arg->a, arg->b);
    return arg->a + arg->b;
}

int subtraction(struct arg_struct* arg) {
    printf("Subtracting %d - %d...\n", arg->a, arg->b);
    return arg->a - arg->b;
}

    ]]>
</AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 1102 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.

8.47.17.1 EXPORTCH int MC_RegisterService_chdl (void * *varg*)

Definition at line 2529 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.17.2 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 1236 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`.

8.47.17.3 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 1181 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()`.

8.47.17.4 EXPORTCH int MC_ResumeAgency_chdl (void * *varg*)

Definition at line 2566 of file `libmc.c`.

Referenced by `agent_RunChScriptThread()`.

8.47.17.5 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 1190 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`.

8.47.17.6 EXPORTCH MCAgent_t MC_RetrieveAgent_chdl (void * *varg*)

Definition at line 2584 of file `libmc.c`.

Referenced by `agent_RunChScriptThread()`.

8.47.17.7 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 1220 of file libmc.c.

References agent_datastate_s::agent_code, agent_s::datastate, agent_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, and agent_datastate_s::task_progress.

8.47.17.8 EXPORTCH char* MC_RetrieveAgentCode_chdl (void * varg)

Definition at line 2601 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.17.9 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

8.47.18 Example

```
<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>mobagent1</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5051</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5050">
            <DATA number_of_elements="0" name="no-return" >
              </DATA>
            </TASK>
          </TASKS>
        </AGENT_DATA>
      </MOBILE_AGENT>
    </MESSAGE>
  </MOBILEC_MESSAGE>
```

```
<AGENT_CODE>
<![CDATA[
#include <stdio.h>
struct arg_struct {
    int a;
    int b;
};
int main()
{
    MCAgent_t agent;
    int retval;
    /* Search Return Variables */
    char** agentNames;
    char** serviceNames;
    int *agentIDs;
    int numResults;

    /* Argument Struct */
    struct arg_struct arg;

    /* Search for addition service */
    printf("\n\nSearching for addition service.\n");
    mc_SearchForService(
        "addition",
        &agentNames,
        &serviceNames,
        &agentIDs,
        &numResults );
    printf("Done searching.\n");
    if (numResults < 1) {
        printf("No agents with service 'addition' found.\n");
        exit(0);
    }

    /* Just get the first hit */
    printf("Using agent %s for addition.\n", agentNames[0]);
    agent = mc_FindAgentByID(agentIDs[0]);

    arg.a = 44;
    arg.b = 45;
    mc_CallAgentFunc(agent, "addition", &retval, &arg);
    printf("Result of addition %d + %d is %d.\n", arg.a, arg.b, retval);

    /* Now search for multiplication service */
    printf("\n\n Searching for Multiplication service...\n");
    mc_SearchForService(
        "multiplication",
        &agentNames,
        &serviceNames,
        &agentIDs,
        &numResults );

    if (numResults < 1) {
        printf("No agents with service 'multiplication' found.\n");
        exit(0);
    }

    printf("Using agent %s for multiplication.\n", agentNames[0]);
    agent = mc_FindAgentByID(agentIDs[0]);
    mc_CallAgentFunc(agent, "multiplication", &retval, &arg);
    printf("Result of multiplication %d * %d is %d.\n", arg.a, arg.b, retval);

    /* Now lets try to deregister a service */
    mc_DeregisterService(
        agentIDs[0],
        serviceNames[0]
    );
}
```

```

    return 0;
}

    ]]>
    </AGENT_CODE>
  </TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 1247 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.

8.47.18.1 EXPORTCH int MC_SearchForService_chdl (void * *varg*)

Definition at line 2617 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.18.2 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

8.47.19 Example

```

<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>wake_agent</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">
          <TASK num="0" complete="0" server="localhost:5051">
            <DATA dim="0" name="no-return">
              </DATA>
            </TASK>
          <AGENT_CODE>
            <![CDATA[
#include <stdio.h>
int main()
{

```

```

    int semaphore_id;
    semaphore_id = 55;
    printf("This is the wake agent.\n");
    mc_SemaphorePost(semaphore_id);

    return 0;
}
    ]]>
    </AGENT_CODE>
  </TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 1297 of file libmc.c.

References `agency_s::mc_platform`, `syncListNode_s::sem`, `SEMAPHORE_POST`, `mc_platform_s::syncList`, and `syncListFind()`.

8.47.19.1 EXPORTCH int MC_SemaphorePost_chdl (void * *varg*)

Definition at line 2658 of file libmc.c.

Referenced by `agent_RunChScriptThread()`.

8.47.19.2 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.

8.47.20 Example

```

<?xml version="1.0"?>

<!DOCTYPE myMessage SYSTEM "mobilec.dtd">

<MOBILEC_MESSAGE>
  <MESSAGE message="MOBILE_AGENT">
    <MOBILE_AGENT>
      <AGENT_DATA>
        <NAME>sleep_agent</NAME>
        <OWNER>IEL</OWNER>
        <HOME>localhost:5050</HOME>
        <TASKS task="1" num="0">

```

```

        <TASK num="0" complete="0" server="localhost:5051">
        <DATA dim="0" name="no-return">
        </DATA>
    </TASK>
    <AGENT_CODE>
    <![CDATA[
#include <stdio.h>
int main()
{
    int sem_id;
    printf("Sleep agent has arrived.\n");
    sem_id = mc_SyncInit(55);
    if (sem_id != 55) {
        printf("Possible error. Aborting...\n");
        exit(1);
    }
    printf("This is the sleep agent.\n");
    printf("I am going to sleep now...\n");
    mc_SemaphoreWait(sem_id);
    printf("This is the sleep agent: I am awake now. Continuing...\n");
    mc_SyncDelete(sem_id);

    return 0;
}

    ]]>
    </AGENT_CODE>
</TASKS>
</AGENT_DATA>
</MOBILE_AGENT>
</MESSAGE>
</MOBILEC_MESSAGE>

```

Definition at line 1309 of file libmc.c.

References `agency_s::mc_platform`, `syncListNode_s::sem`, `SEMAPHORE_WAIT`, `mc_platform_s::syncList`, and `syncListFind()`.

8.47.20.1 EXPORTCH int MC_SemaphoreWait_chdl (void * *varg*)

Definition at line 2682 of file libmc.c.

Referenced by `agent_RunChScriptThread()`.

8.47.20.2 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 1331 of file libmc.c.

References `MC_ERR`, `agency_s::mc_platform`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, and `mc_platform_s::message_queue`.

8.47.20.3 EXPORTCH int MC_SendAgentMigrationMessage_chdl (void * *varg*)

Definition at line 2706 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.20.4 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 1362 of file libmc.c.

References agency_s::mc_platform, message_Destroy(), message_InitializeFromString(), message_New(), and mc_platform_s::message_queue.

8.47.20.5 EXPORTCH int MC_SendAgentMigrationMessageFile_chdl (void * *varg*)

Definition at line 2731 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.20.6 int MC_SendCh (MCAgency_t *attr*, const char * *filename*, const char * *remotehost*, int *port*)

Definition at line 1321 of file libmc.c.

8.47.20.7 EXPORTMC int MC_SendSteerCommand (MCAgency_t *attr*, enum MC_SteerCommand_e *cmd*)

Definition at line 1438 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.

8.47.20.8 EXPORTCH int MC_SendSteerCommand_chdl (void * *varg*)

Definition at line 2751 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.20.9 int MC_SetAgentStatus (MCAgent_t *agent*, int *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 1448 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_DeleteAgent().

8.47.20.10 EXPORTCH int MC_SetAgentStatus_chdl (void * *varg*)

Definition at line 2774 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.20.11 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 1463 of file libmc.c.

References mc_platform_s::default_agentstatus, and agency_s::mc_platform.

8.47.20.12 EXPORTCH int MC_SetDefaultAgentStatus_chdl (void * *varg*)

Definition at line 2792 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.20.13 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 1490 of file libmc.c.

References SET_THREAD_OFF, and MCAgencyOptions_s::threads.

8.47.20.14 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 1473 of file libmc.c.

References SET_THREAD_ON, and MCAgencyOptions_s::threads.

8.47.20.15 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 1497 of file libmc.c.

References MC_THREAD_ALL, SET_THREAD_OFF, and MCAgencyOptions_s::threads.

8.47.20.16 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 1480 of file libmc.c.

References [MC_THREAD_ALL](#), [SET_THREAD_ON](#), and [MCAgencyOptions_s::threads](#).

8.47.20.17 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.

8.47.21 Example

```
#include <stdio.h>
#include <libmc.h>
#ifdef _WIN32
#include <windows.h>
#endif

int algorithm(void* boo);

int main() {
    MCAgency_t agency;
    int local_port = 5050;
    MCAgencyOptions_t options;

    MC_InitializeAgencyOptions(&options);

    MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */

    agency = MC_Initialize(local_port, &options);

    printf("Starting algorithm...\n");
    MC_Steer(
```

```

        agency,
        &algorithm,
        NULL
    );

    MC_End(agency);
    return 0;
}

int algorithm(void* boo)
{
    int i=0;
    MC_SteerCommand_t command;
    while(1) {
#ifdef _WIN32
        sleep(1);
#else
        Sleep(1000);
#endif
        printf("%d \n", i);
        i++;
        command = MC_SteerControl();
        if(
            command == MC_RESTART ||
            command == MC_STOP
        )
        {
            return 0;
        }
    }
}

```

Definition at line 1507 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`.

8.47.21.1 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 1526 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`.

8.47.21.2 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 1542 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().

8.47.21.3 EXPORTCH int MC_SyncDelete_chdl (void * varg)

Definition at line 2815 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.21.4 EXPORTMC int MC_SyncInit (MC_Agency_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 1572 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().

8.47.21.5 EXPORTCH int MC_SyncInit_chdl (void * varg)

Definition at line 2838 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.21.6 EXPORTMC int MC_TerminateAgent (MCAgent_t *agent*)

Halt a running agent.

Returns:

0 on success, error_code_t on failure

Definition at line 1597 of file libmc.c.

References agent_s::agent_interp.

Referenced by MC_DeleteAgent().

8.47.21.7 EXPORTCH int MC_TerminateAgent_chdl (void * *varg*)

Definition at line 2862 of file libmc.c.

Referenced by agent_RunChScriptThread().

8.47.21.8 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 1624 of file libmc.c.

References mc_platform_s::agent_queue, COND_WAIT, mc_platform, agency_s::mc_platform, MUTEX_LOCK, MUTEX_UNLOCK, and size.

8.47.21.9 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 1644 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.

8.47.21.10 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.

8.47.22 Example

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

int main(int argc, char *argv[])
{
    MCAgency_t agency;
    MCAgencyOptions_t options;
    MCAgent_t agent;
    int dim, *extent;
    double *data;
    int i, j, size;
    int port=5050;
    int remote_port=5051;

    MC_InitializeAgencyOptions(&options);
    MC_SetThreadOff(&options, MC_THREAD_CP); /* Turn off command prompt */
    agency = MC_Initialize(port, &options);

    printf("MobileC Started\n");
    /* Note: The third argument of the following function may also be a
       valid IP address in the form of a string. i.e. 192.168.0.1 */
    MC_SendAgentMigrationMessageFile(agency,
        "test.xml",
        "localhost",
        remote_port);

    /* Wait for return-agent arrival signal */
    MC_WaitSignal(agency, MC_RECV_RETURN);

    /* Make sure we caught the correct agent */
    agent = MC_FindAgentByName(agency, "mobagent3");
    if (agent == NULL) {
        fprintf(stderr, "Did not receive correct agent. \n");
        exit(1);
    }

    /* Print relevant information */
    printf("%d tasks.\n", MC_GetAgentNumTasks(agent) );
    for (i = 0; i < MC_GetAgentNumTasks(agent); i++) {
        MC_GetAgentReturnData(
            agent,
            i,
            (void*)&data,
            &dim,
```

```

        &extent );
    printf("Task: %d\n", i);
    size = 1;
    printf("dim is %d\n", dim);
    for (j = 0; j < dim; j++) {
        size *= extent[j];
    }
    printf("Size: %d\n", size);
    printf("Data elements: ");
    for (j = 0; j < size; j++) {
        printf("%f ", data[j]);
    }
    printf("\n\n");
    free(data);
    free(extent);
}

/* We must reset the signal that we previously caught with the
 * MC_WaitSignal() function with MC_ResetSignal() */
MC_ResetSignal(agency);

MC_End(agency);
return 0;
}

```

Definition at line 1662 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().

8.47.23 Variable Documentation

8.47.23.1 mc_platform_p g_mc_platform

Definition at line 62 of file libmc.c.

Referenced by fipa_envelope_Compose__from().

8.48 /home/dko/projects/mobilec/trunk/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)

8.48.1 Function Documentation

8.48.1.1 [int list_pGetSize](#) ([list_p](#) *list*)

Definition at line 70 of file list.c.

References [list_s::size](#).

8.48.1.2 [int ListAdd](#) ([list_p](#) *list*, DATA *data*)

Definition at line 158 of file list.c.

References [list_s::listhead](#), [listNode_s::next](#), [listNode_s::node_data](#), and [list_s::size](#).

Referenced by [barrier_queue_Add\(\)](#), [df_Add\(\)](#), [df_AddRequest\(\)](#), and [syncListAddNode\(\)](#).

8.48.1.3 [DATA ListDelete](#) ([list_p](#) *list*, const [int](#) *index*)

Definition at line 213 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\(\)](#), and [syncListRemove\(\)](#).

8.48.1.4 [DATA ListGetHead](#) ([list_p](#) *list*)

Definition at line 76 of file list.c.

References [list_s::listhead](#), and [listNode_s::node_data](#).

8.48.1.5 list_p ListInitialize (void)

Definition at line 30 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().

8.48.1.6 int ListInsert (list_p list, DATA data, const int index)

Definition at line 200 of file list.c.

8.48.1.7 DATA ListPop (list_p list)

Definition at line 86 of file list.c.

References DATA, list_s::listhead, listNode_s::next, listNode_s::node_data, and list_s::size.

Referenced by barrier_queue_Pop(), df_Destroy(), df_request_list_Destroy(), and df_request_list_Pop().

8.48.1.8 DATA ListSearch (list_p list, const int index)

Definition at line 121 of file list.c.

References list_s::listhead, listNode_s::next, listNode_s::node_data, and list_s::size.

Referenced by ams_ManageAgentList(), ams_Print(), AP_QUEUE_SEARCH_TEMPLATE(), AP_QUEUE_STD_DEFN_TEMPLATE(), barrier_queue_Delete(), MC_RetrieveAgent(), MC_WaitRetrieveAgent(), message_queue_SendOutgoing(), syncListDelete(), and syncListRemove().

8.48.1.9 void ListTerminate (list_p list)

Definition at line 46 of file list.c.

References list_s::listhead, and list_s::size.

Referenced by barrier_queue_Destroy(), df_Destroy(), and df_request_list_Destroy().

8.49 /home/dko/projects/mobilec/trunk/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)

8.49.1 Define Documentation

8.49.1.1 #define DATA void*

Definition at line 27 of file list.h.

Referenced by [barrier_queue_Add\(\)](#), [ListDelete\(\)](#), [ListPop\(\)](#), and [syncListAddNode\(\)](#).

8.49.1.2 #define QUEUE_TEMPLATE(name, node_type, search_type, search_var_name)

Definition at line 65 of file list.h.

8.49.2 Typedef Documentation

8.49.2.1 typedef list_t* list_p

Definition at line 46 of file list.h.

8.49.2.2 typedef struct list_s list_t

8.49.2.3 typedef listNode_t* listNode_p

Definition at line 36 of file list.h.

8.49.2.4 typedef struct listNode_s listNode_t

8.49.3 Function Documentation

8.49.3.1 int ListAdd (list_p *list*, DATA *data*)

Definition at line 158 of file list.c.

References list_s::listhead, listNode_s::next, listNode_s::node_data, and list_s::size.

Referenced by barrier_queue_Add(), df_Add(), df_AddRequest(), and syncListAddNode().

8.49.3.2 DATA ListDelete (list_p *list*, const int *index*)

Definition at line 213 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(), and syncListRemove().

8.49.3.3 DATA ListGetHead (list_p *list*)

Definition at line 76 of file list.c.

References list_s::listhead, and listNode_s::node_data.

8.49.3.4 int ListGetSize (list_p *list*)

8.49.3.5 list_p ListInitialize (void)

Definition at line 30 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().

8.49.3.6 int ListInsert (list_p *list*, DATA *data*, const int *index*)

Definition at line 200 of file list.c.

8.49.3.7 DATA ListPop (*list_p list*)

Definition at line 86 of file list.c.

References DATA, list_s::listhead, listNode_s::next, listNode_s::node_data, and list_s::size.

Referenced by barrier_queue_Pop(), df_Destroy(), df_request_list_Destroy(), and df_request_list_Pop().

8.49.3.8 DATA ListSearch (*list_p list*, *const int index*)

Definition at line 121 of file list.c.

References list_s::listhead, listNode_s::next, listNode_s::node_data, and list_s::size.

Referenced by ams_ManageAgentList(), ams_Print(), AP_QUEUE_SEARCH_TEMPLATE(), AP_QUEUE_STD_DEFN_TEMPLATE(), barrier_queue_Delete(), MC_RetrieveAgent(), MC_WaitRetrieveAgent(), message_queue_SendOutgoing(), syncListDelete(), and syncListRemove().

8.49.3.9 void ListTerminate (*list_p list*)

Definition at line 46 of file list.c.

References list_s::listhead, and list_s::size.

Referenced by barrier_queue_Destroy(), df_Destroy(), and df_request_list_Destroy().

8.50 /home/dko/projects/mobilec/trunk/src/mc_platform.c File Reference

```
#include <unistd.h>
#include <sys/socket.h>
#include <netdb.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include "include/acc.h"
#include "include/mc_platform.h"
```

Defines

- #define [DEFAULT_HOSTNAME_LENGTH](#) 200

Functions

- [mc_platform_p mc_platform_Initialize](#) ([MCAgency_t](#) agency)
- [int mc_platform_Destroy](#) ([mc_platform_p](#) platform)

8.50.1 Define Documentation

8.50.1.1 #define [DEFAULT_HOSTNAME_LENGTH](#) 200

Definition at line 42 of file mc_platform.c.

Referenced by [mc_platform_Initialize\(\)](#).

8.50.2 Function Documentation

8.50.2.1 [int mc_platform_Destroy](#) ([mc_platform_p](#) platform)

Definition at line 232 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::asm_message_queue](#), [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::interp_options](#), [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](#), and [mc_platform_s::quit_lock](#).

Referenced by [handler_QUIT\(\)](#), and [MC_End\(\)](#).

8.50.2.2 [mc_platform_p mc_platform_Initialize](#) ([MCAgency_t](#) agency)

Definition at line 44 of file mc_platform.c.

References `mc_platform_s::acc`, `acc_Initialize()`, `acc_Start()`, `mc_platform_s::agent_queue`, `mc_platform_s::ams`, `ams_Initialize()`, `ams_Start()`, `mc_platform_s::asm_message_queue`, `mc_platform_s::barrier_queue`, `barrier_queue_New()`, `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()`, `agency_s::enable_security`, `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`, `mc_platform_s::interp_options`, `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_lock`, `agency_s::stack_size`, `mc_platform_s::stack_size`, `mc_platform_s::syncList`, `syncListInit()`, and `agency_s::threads`.

Referenced by `MC_Initialize()`.

8.51 /home/dko/projects/mobilec/trunk/src/mc_rwlock.c File Reference

```
#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 \(mc_rwlock_p rwlock\)](#)

8.51.1 Function Documentation

8.51.1.1 int mc_rwlock_destroy (mc_rwlock_p *rwlock*)

Definition at line 58 of file mc_rwlock.c.

References `mc_rwlock_s::cond`, `COND_DESTROY`, `mc_rwlock_s::lock`, and `MUTEX_DESTROY`.

8.51.1.2 int mc_rwlock_init (mc_rwlock_p *rwlock*)

Definition at line 38 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`.

8.51.1.3 int mc_rwlock_rdlock (mc_rwlock_p *rwlock*)

Definition at line 71 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`.

8.51.1.4 int mc_rwlock_rdunlock (mc_rwlock_p *rwlock*)

Definition at line 86 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`.

8.51.1.5 int mc_rwlock_wrlock (mc_rwlock_p *rwlock*)

Definition at line 99 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.

8.51.1.6 int mc_rwlock_wrunlock (mc_rwlock_p *rwlock*)

Definition at line 118 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.

8.52 /home/dko/projects/mobilec/trunk/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 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*)

8.52.1 Function Documentation

8.52.1.1 int syncListAddNode (struct syncListNode_s * *node*, struct syncList_s * *list*)

Definition at line 83 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()`.

8.52.1.2 int syncListDelete (int *id*, struct syncList_s * *list*)

Definition at line 112 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()`.

8.52.1.3 syncListNode_t* syncListFind (int *id*, struct syncList_s * *list*)

Definition at line 53 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()`.

8.52.1.4 struct syncList_s* syncListInit (void) [read]

Definition at line 69 of file sync_list.c.

References ListInitialize(), syncList_s::lock, MUTEX_INIT, MUTEX_T, RWLOCK_INIT, and RWLOCK_T.

Referenced by mc_platform_Initialize().

8.52.1.5 int syncListNew (int id, struct syncList_s * list)

Definition at line 102 of file sync_list.c.

References node, syncListAddNode(), and syncListNodeInit().

8.52.1.6 int syncListNodeDestroy (struct syncListNode_s * node)

Definition at line 41 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(), and syncListDelete().

8.52.1.7 int syncListNodeInit (struct syncListNode_s * node)

Definition at line 8 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().

8.52.1.8 struct syncListNode_s* syncListNodeNew (void) [read]

Definition at line 23 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().

8.52.1.9 syncListNode_t* syncListRemove (int id, struct syncList_s * list)

Definition at line 129 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().

8.53 /home/dko/projects/mobilec/trunk/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 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](#))

8.53.1 Typedef Documentation

8.53.1.1 typedef [syncList_t](#)* [syncList_p](#)

Definition at line 30 of file [sync_list.h](#).

8.53.1.2 typedef struct [syncList_s](#) [syncList_t](#)

8.53.1.3 typedef [syncListNode_t](#)* [syncListNode_p](#)

Definition at line 20 of file [sync_list.h](#).

8.53.1.4 typedef struct syncListNode_s syncListNode_t

8.53.2 Function Documentation

8.53.2.1 int syncListAddNode (struct syncListNode_s * *node*, struct syncList_s * *list*)

Definition at line 83 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().

8.53.2.2 int syncListDelete (int *id*, struct syncList_s * *list*)

Definition at line 112 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().

8.53.2.3 syncListNode_t* syncListFind (int *id*, struct syncList_s * *list*)

Definition at line 53 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().

8.53.2.4 syncListNode_t* syncListGet (int *id*, struct syncList_s * *list*)

8.53.2.5 struct syncList_s* syncListInit (void) [read]

Definition at line 69 of file sync_list.c.

References ListInitialize(), syncList_s::lock, MUTEX_INIT, MUTEX_T, RWLOCK_INIT, and RWLOCK_T.

Referenced by mc_platform_Initialize().

8.53.2.6 int syncListNew (int *id*, struct syncList_s * *list*)

Definition at line 102 of file sync_list.c.

References node, syncListAddNode(), and syncListNodeInit().

8.53.2.7 int syncListNodeDestroy (struct syncListNode_s * *node*)

Definition at line 41 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(), and syncListDelete().

8.53.2.8 int syncListNodeInit (struct syncListNode_s * *node*)

Definition at line 8 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().

8.53.2.9 struct syncListNode_s* syncListNodeNew (void) [read]

Definition at line 23 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().

8.53.2.10 syncListNode_t* syncListRemove (int *id*, struct syncList_s * *list*)

Definition at line 129 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().

8.54 /home/dko/projects/mobilec/trunk/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 <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_helper.h"
#include "include/xml_parser.h"
#include "security/asm_node.h"
```

Defines

- `#define SOCKET_INPUT_SIZE 4096`

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 message_Send (message_p message)`

8.54.1 Define Documentation

8.54.1.1 `#define SOCKET_INPUT_SIZE 4096`

Definition at line 55 of file message.c.

8.54.2 Function Documentation

8.54.2.1 `int http_to_hostport (const char * http_str, char ** host, int * port, char ** target)`

Definition at line 502 of file message.c.

References MC_ERR_PARSE.

Referenced by MC_AclSend().

8.54.2.2 `message_p message_Copy (message_p src)`

Definition at line 307 of file message.c.

8.54.2.3 `int message_Destroy (message_p message)`

Definition at line 601 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_MessageHandlerThread(), acc_Thread(), ams_ManageAgentList(), MC_AclSend(), MC_LoadAgentFromFile(), MC_SendAgentMigrationMessage(), MC_SendAgentMigrationMessageFile(), message_InitializeFromConnection(), and message_queue_SendOutgoing().

8.54.2.4 `int message_InitializeFromAgent (mc_platform_p mc_platform, message_p message, agent_p agent)`

Definition at line 315 of file message.c.

References message_s::addr, agent_xml_compose(), message_s::agent_xml_flag, 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_datastate_s::number_of_tasks, mc_platform_s::port, agent_task_s::server_name, 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.

8.54.2.5 `int message_InitializeFromConnection (mc_platform_p mc_platform, message_p message, connection_p connection)`

Definition at line 414 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_INPUT_SIZE, message_s::target, message_s::to_address, and message_s::xml_root.

8.54.2.6 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 537 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_SUCCESS, message_s::message_body, message_s::message_id, message_s::message_type, MOBILE_AGENT, mc_platform_s::port, message_s::target, message_s::to_address, message_s::update_name, and message_s::xml_root.

8.54.2.7 message_p message_New (void)

Definition at line 283 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_Thread(), ams_ManageAgentList(), MC_LoadAgentFromFile(), MC_SendAgentMigrationMessage(), MC_SendAgentMigrationMessageFile(), and mtp_http_CreateMessage().

8.54.2.8 int message_Send (message_p message)

Definition at line 639 of file message.c.

References CHECK_NULL, MC_ERR, MC_ERR_CONNECT, MC_ERR_SEND, message_s::message_body, mtp_http_ComposeMessage(), mtp_http_New(), mtp_http_Parse(), port, send, SOCKET_INPUT_SIZE, strtok_r, and message_s::to_address.

Referenced by acc_MessageHandlerThread(), MC_AclSend(), and message_queue_SendOutgoing().

8.55 /home/dko/projects/mobilec/trunk/src/message_queue.c File Reference

```
#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](#))

8.55.1 Function Documentation

8.55.1.1 void [message_queue_SendOutgoing](#) (struct [mc_platform_s](#) * [mc_platform](#), [message_queue_p](#) [mqueue](#))

Definition at line 49 of file [message_queue.c](#).

References [CHECK_NULL](#), [mc_platform_s::hostname](#), [ListSearch\(\)](#), [message_Destroy\(\)](#), [message_Send\(\)](#), [MUTEX_LOCK](#), [MUTEX_UNLOCK](#), [mc_platform_s::port](#), and [message_s::to_address](#).

8.56 /home/dko/projects/mobilec/trunk/src/mtp_http.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <unistd.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 mtp_http_InitializeFromConnection](#) ([mtp_http_p](#) http, [connection_p](#) connection)
- 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.
- [int mtp_http_Parse](#) (struct [mtp_http_s](#) *http, const char *string)
- const char * [http_ParseHeader](#) ([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)

8.56.1 Define Documentation

8.56.1.1 #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_Parse().

8.56.1.2 #define SAFE_FREE(object)

Value:

```
if(object) free(object); \
    object = NULL
```

8.56.1.3 #define SAFE_FREE(elem)

Value:

```
if(elem) \
    free(elem)
```

Referenced by mtp_http_Destroy(), and mtp_http_Parse().

8.56.2 Function Documentation

8.56.2.1 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 207 of file mtp_http.c.

Referenced by mtp_http_Parse().

8.56.2.2 const char* http_GetToken (const char * *string*, char ** *token*)

Definition at line 630 of file mtp_http.c.

Referenced by http_ParseHeader().

8.56.2.3 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 266 of file mtp_http.c.

References CHECK_NULL, MC_ERR_PARSE, and MC_SUCCESS.

Referenced by mtp_http_Parse().

8.56.2.4 const char* http_ParseHeader (mtp_http_p *http*, const char * *string*)

Definition at line 541 of file mtp_http.c.

References 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_Parse().

8.56.2.5 int mtp_http_ComposeMessage (message_p *message*)

Definition at line 669 of file mtp_http.c.

References message_s::isHTTP, MC_SUCCESS, message_s::message_body, PACKAGE_VERSION, message_s::target, and message_s::to_address.

8.56.2.6 struct message_s* mtp_http_CreateMessage (mtp_http_t * *mtp_http*, char * *hostname*, int *port*) [read]

Definition at line 715 of file mtp_http.c.

References 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().

8.56.2.7 `int mtp_http_Destroy (mtp_http_p http)`

Definition at line 50 of file `mtp_http.c`.

References `mtp_http_s::accept_ranges`, `mtp_http_s::boundary`, `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::return_code`, `SAFE_FREE`, `mtp_http_s::server`, `mtp_http_s::target`, and `mtp_http_s::user_agent`.

Referenced by `acc_Thread()`, and `MC_AclSend()`.

8.56.2.8 `int mtp_http_InitializeFromConnection (mtp_http_p http, connection_p connection)`

Definition at line 94 of file `mtp_http.c`.

References `CHECK_NULL`, `connection_s::clientfd`, `ERR`, `MC_ERR_CONNECT`, `mtp_http_Parse()`, `send`, and `SOCKET_INPUT_SIZE`.

8.56.2.9 `mtp_http_p mtp_http_New (void)`

Definition at line 82 of file `mtp_http.c`.

References `CHECK_NULL`, and `mtp_http_s::content`.

Referenced by `acc_Thread()`, `MC_AclSend()`, and `message_Send()`.

8.56.2.10 `int mtp_http_Parse (struct mtp_http_s * http, const char * string)`

Definition at line 326 of file `mtp_http.c`.

References `mtp_http_s::boundary`, `mtp_http_s::content`, `mtp_http_content_s::content_type`, `mtp_http_s::content_type`, `mtp_http_content_s::data`, `http_GetExpression()`, `HTTP_PARSE_EXPR`, `http_ParseExpression()`, `http_ParseHeader()`, `mtp_http_s::http_performative`, `HTTP_POST`, `HTTP_PUT`, `HTTP_RESPONSE`, `MC_SUCCESS`, `mtp_http_s::message_parts`, and `SAFE_FREE`.

Referenced by `message_Send()`, and `mtp_http_InitializeFromConnection()`.

8.56.2.11 `int mtp_http_ParseResponse (struct mtp_http_s * http, const char * string)`

Definition at line 663 of file `mtp_http.c`.

8.57 /home/dko/projects/mobilec/trunk/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`)

8.57.1 Define Documentation

8.57.1.1 `#define HAVE_STRDUP 1`

Definition at line 49 of file config.h.

8.57.1.2 `#define HAVE_VSNPRINTF 1`

Definition at line 42 of file config.h.

8.57.1.3 `#define MXML_VERSION "Mini-XML v2.2.2"`

Definition at line 35 of file config.h.

Referenced by `write_documentation()`.

8.57.2 Function Documentation

8.57.2.1 `char* mxml_strdup (const char *, va_list)`

Definition at line 59 of file mxml-string.c.

Referenced by `mxml_error()`, `mxmlNewTextf()`, and `mxmlSetTextf()`.

8.58 /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/vcnet/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 v2.0"
- #define [HAVE_STRDUP](#) 1
- #define [vsnprintf](#) mxml_vsnprintf

Functions

- char * [mxml_strdup](#) (const char *, va_list)
- int [mxml_vsnprintf](#) (char *, size_t, const char *, va_list)

8.58.1 Define Documentation

8.58.1.1 #define HAVE_STRDUP 1

Definition at line 48 of file config.h.

8.58.1.2 #define MXML_VERSION "Mini-XML v2.0"

Definition at line 34 of file config.h.

8.58.1.3 #define vsnprintf mxml_vsnprintf

Definition at line 64 of file config.h.

8.58.2 Function Documentation

8.58.2.1 char* mxml_strdup (const char *, va_list)

Definition at line 59 of file mxml-string.c.

8.58.2.2 int mxml_vsnprintf (char *, size_t, const char *, va_list)

Definition at line 104 of file mxml-string.c.

8.59 /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/vcnet2005/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 v2.0"
- `#define HAVE_STRDUP` 1
- `#define vsnprintf` mxml_vsnprintf

Functions

- `char * mxml_strdup` (const char *, va_list)
- `int mxml_vsnprintf` (char *, size_t, const char *, va_list)

8.59.1 Define Documentation

8.59.1.1 `#define HAVE_STRDUP` 1

Definition at line 48 of file config.h.

8.59.1.2 `#define MXML_VERSION` "Mini-XML v2.0"

Definition at line 34 of file config.h.

8.59.1.3 `#define vsnprintf` mxml_vsnprintf

Definition at line 64 of file config.h.

8.59.2 Function Documentation

8.59.2.1 `char* mxml_strdup` (const char *, va_list)

Definition at line 59 of file mxml-string.c.

References `mxml_vsnprintf()`.

8.59.2.2 `int mxml_vsnprintf` (char *, size_t, const char *, va_list)

Definition at line 104 of file mxml-string.c.

References size.

Referenced by `mxml_strdup()`.

8.60 /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-attr.c File Reference

```
#include "config.h"
#include "mxml.h"
```

Functions

- const char * [mxmlElementGetAttr](#) (mxml_node_t *node, const char *name)
- void [mxmlElementSetAttr](#) (mxml_node_t *node, const char *name, const char *value)

8.60.1 Function Documentation

8.60.1.1 const char* mxmlElementGetAttr (mxml_node_t * node, const char * name)

Definition at line 40 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\(\)](#), [mxml_parse_element\(\)](#), [mxmlFindElement\(\)](#), [mxmlIndexNew\(\)](#), [scan_file\(\)](#), [sort_node\(\)](#), [type_cb\(\)](#), and [write_documentation\(\)](#).

8.60.1.2 void mxmlElementSetAttr (mxml_node_t * node, const char * name, const char * value)

Definition at line 87 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](#), [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\(\)](#).

8.61 /home/dko/projects/mobilec/trunk/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)

8.61.1 Function Documentation

8.61.1.1 static int default_callback (const char * *name*) [static]

Definition at line 166 of file mxml-entity.c.

References [int](#).

8.61.1.2 int mxmlEntityAddCallback (int(*) (const char *name) *cb*)

Definition at line 62 of file mxml-entity.c.

References [callbacks](#), [mxml_error\(\)](#), and [num_callbacks](#).

8.61.1.3 const char* mxmlEntityGetName (int *val*)

Definition at line 88 of file mxml-entity.c.

Referenced by [mxml_write_name\(\)](#), and [mxml_write_string\(\)](#).

8.61.1.4 int mxmlEntityGetValue (const char * *name*)

Definition at line 118 of file mxml-entity.c.

References [callbacks](#), and [num_callbacks](#).

Referenced by [mxml_get_entity\(\)](#).

8.61.1.5 void mxmlEntityRemoveCallback (int(*) (const char *name) cb)

Definition at line 137 of file mxml-entity.c.

References `callbacks`, and `num_callbacks`.

8.61.2 Variable Documentation

8.61.2.1 int(* callbacks[100])(const char *name) [static]

Initial value:

```
{
    default_callback
}
```

Referenced by `mxmlEntityAddCallback()`, `mxmlEntityGetValue()`, and `mxmlEntityRemoveCallback()`.

8.61.2.2 int num_callbacks = 1 [static]

Definition at line 50 of file mxml-entity.c.

Referenced by `mxmlEntityAddCallback()`, `mxmlEntityGetValue()`, and `mxmlEntityRemoveCallback()`.

8.62 /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-file.c File Reference

```
#include "config.h"
#include "mxml.h"
#include <unistd.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`

8.62.1 Define Documentation

8.62.1.1 `#define ENCODE_UTF16BE 1`

Definition at line 65 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, and `mxml_string_getc()`.

8.62.1.2 `#define ENCODE_UTF16LE 2`

Definition at line 66 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, and `mxml_string_getc()`.

8.62.1.3 `#define ENCODE_UTF8 0`

Definition at line 64 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, `mxml_load_data()`, and `mxml_string_getc()`.

8.62.1.4 `#define mxml_bad_char(ch) ((ch) < ' ' && (ch) != '\n' && (ch) != '\r' && (ch) != '\t')`

Definition at line 73 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, `mxml_get_entity()`, and `mxml_string_getc()`.

8.62.2 Typedef Documentation

8.62.2.1 `typedef struct mxml_fdbuf_s mxml_fdbuf_t`

8.62.3 Function Documentation

8.62.3.1 `static int mxml_add_char (int ch, char **ptr, char **buffer, int *bufsize) [static]`

Definition at line 480 of file mxml-file.c.

References `mxml_error()`.

Referenced by `mxml_load_data()`, and `mxml_parse_element()`.

8.62.3.2 `static int mxml_fd_getc (void *p, int *encoding)` `[static]`

Definition at line 560 of file `mxml-file.c`.

References `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()`.

8.62.3.3 `static int mxml_fd_putc (int ch, void *p)` `[static]`

Definition at line 847 of file `mxml-file.c`.

References `mxml_fdbuf_s::current`, `mxml_fdbuf_s::end`, and `mxml_fd_write()`.

Referenced by `mxmlSaveFd()`.

8.62.3.4 `static int mxml_fd_read (mxml_fdbuf_t *buf)` `[static]`

Definition at line 916 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()`.

8.62.3.5 `static int mxml_fd_write (mxml_fdbuf_t *buf)` `[static]`

Definition at line 955 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()`.

8.62.3.6 `static int mxml_file_getc (void *p, int *encoding)` `[static]`

Definition at line 998 of file `mxml-file.c`.

References `ENCODE_UTF16BE`, `ENCODE_UTF16LE`, `ENCODE_UTF8`, `mxml_bad_char`, and `mxml_error()`.

Referenced by `mxmlLoadFile()`.

8.62.3.7 `static int mxml_file_putc (int ch, void *p)` `[static]`

Definition at line 1197 of file `mxml-file.c`.

Referenced by `mxmlSaveFile()`.

8.62.3.8 `static int mxml_get_entity (mxml_node_t *parent, void *p, int *encoding, int(*) (void *, int *) getc_cb) [static]`

Definition at line 1252 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()`.

8.62.3.9 `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 1313 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()`.

8.62.3.10 `static int mxml_parse_element (mxml_node_t *node, void *p, int *encoding, int(*) (void *, int *) getc_cb) [static]`

Definition at line 1837 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()`.

8.62.3.11 `static int mxml_string_getc (void *p, int *encoding) [static]`

Definition at line 2099 of file mxml-file.c.

References `ENCODE_UTF16BE`, `ENCODE_UTF16LE`, `ENCODE_UTF8`, `mxml_bad_char`, and `mxml_error()`.

Referenced by `mxmlLoadString()`.

8.62.3.12 `static int mxml_string_putc (int ch, void *p) [static]`

Definition at line 2335 of file mxml-file.c.

Referenced by `mxmlSaveString()`.

8.62.3.13 `static int mxml_write_name (const char *s, void *p, int(*) (int, void *) putc_cb) [static]`

Definition at line 2409 of file mxml-file.c.

References `mxmlEntityGetName()`.

Referenced by `mxml_write_node()`.

8.62.3.14 `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 2484 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().

8.62.3.15 `static int mxml_write_string (const char * s, void * p, int(*)(int, void *) putc_cb) [static]`

Definition at line 2765 of file mxml-file.c.

References mxmlEntityGetName().

Referenced by mxml_write_node().

8.62.3.16 `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 2805 of file mxml-file.c.

References MXML_TAB.

Referenced by mxml_write_node().

8.62.3.17 `mxml_node_t* mxmlLoadFd (mxml_node_t * top, int fd, mxml_type_t(*)(mxml_node_t *node) cb)`

Definition at line 156 of file mxml-file.c.

References 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().

8.62.3.18 `mxml_node_t* mxmlLoadFile (mxml_node_t * top, FILE * fp, mxml_type_t(*)(mxml_node_t *node) cb)`

Definition at line 196 of file mxml-file.c.

References mxml_file_getc(), and mxml_load_data().

Referenced by main().

8.62.3.19 mxml_node_t* mxmlLoadString (mxml_node_t * *top*, const char * *s*, mxml_type_t(*) (mxml_node_t **node*) *cb*)

Definition at line 225 of file mxml-file.c.

References mxml_load_data(), and mxml_string_getc().

Referenced by acc_Thread(), agent_xml_compose(), fipa_envelope_Compose(), fipa_envelope_Parse(), main(), MC_LoadAgentFromFile(), and message_InitializeFromConnection().

8.62.3.20 char* mxmlSaveAllocString (mxml_node_t * *node*, const char *(*) (mxml_node_t **node*, int *ws*) *cb*)

Definition at line 255 of file mxml-file.c.

References mxmlSaveString().

Referenced by fipa_envelope_Compose(), MC_GetAgentXMLString(), and message_InitializeFromAgent().

8.62.3.21 int mxmlSaveFd (mxml_node_t * *node*, int *fd*, const char *(*) (mxml_node_t **node*, int *ws*) *cb*)

Definition at line 312 of file mxml-file.c.

References 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().

8.62.3.22 int mxmlSaveFile (mxml_node_t * *node*, FILE * *fp*, const char *(*) (mxml_node_t **node*, int *ws*) *cb*)

Definition at line 359 of file mxml-file.c.

References mxml_file_putc(), and mxml_write_node().

Referenced by main().

8.62.3.23 int mxmlSaveString (mxml_node_t * *node*, char * *buffer*, int *bufsize*, const char *(*) (mxml_node_t **node*, int *ws*) *cb*)

Definition at line 401 of file mxml-file.c.

References mxml_string_putc(), and mxml_write_node().

Referenced by main(), and mxmlSaveAllocString().

8.62.3.24 void mxmlSetCustomHandlers (mxml_custom_load_cb_t *load*, mxml_custom_save_cb_t *save*)

Definition at line 453 of file mxml-file.c.

References mxml_custom_load_cb, and mxml_custom_save_cb.

8.62.3.25 void mxmlSetErrorCallback (void(*) (const char *) *cb*)

Definition at line 468 of file mxml-file.c.

References mxml_error_cb.

8.62.4 Variable Documentation**8.62.4.1 mxml_custom_load_cb_t mxml_custom_load_cb = NULL [static]**

Definition at line 100 of file mxml-file.c.

Referenced by mxml_load_data(), and mxmlSetCustomHandlers().

8.62.4.2 mxml_custom_save_cb_t mxml_custom_save_cb = NULL [static]

Definition at line 101 of file mxml-file.c.

Referenced by mxml_write_node(), and mxmlSetCustomHandlers().

8.62.4.3 void(* mxml_error_cb)(const char *)

Referenced by mxml_error(), and mxmlSetErrorCallback().

8.63 /home/dko/projects/mobilec/trunk/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)

8.63.1 Function Documentation

8.63.1.1 static int [index_compare](#) ([mxml_index_t](#) * *ind*, [mxml_node_t](#) * *first*, [mxml_node_t](#) * *second*) [static]

Definition at line 491 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\(\)](#).

8.63.1.2 static int [index_find](#) ([mxml_index_t](#) * *ind*, const char * *element*, const char * *value*, [mxml_node_t](#) * *node*) [static]

Definition at line 530 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\(\)](#).

8.63.1.3 static void [index_sort](#) ([mxml_index_t](#) * *ind*, int *left*, int *right*) [static]

Definition at line 573 of file mxml-index.c.

References [index_compare\(\)](#), and [mxml_index_s::nodes](#).

Referenced by [mxmlIndexNew\(\)](#).

8.63.1.4 void [mxmlIndexDelete](#) ([mxml_index_t](#) * *ind*)

Definition at line 55 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()`.

8.63.1.5 `mxml_node_t* mxmlIndexEnum (mxml_index_t * ind)`

Definition at line 85 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()`.

8.63.1.6 `mxml_node_t* mxmlIndexFind (mxml_index_t * ind, const char * element, const char * value)`

Definition at line 115 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()`.

8.63.1.7 `mxml_index_t* mxmlIndexNew (mxml_node_t * node, const char * element, const char * attr)`

Definition at line 298 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`, and `mxml_node_s::value`.

Referenced by `main()`.

8.63.1.8 `mxml_node_t* mxmlIndexReset (mxml_index_t * ind)`

Definition at line 456 of file `mxml-index.c`.

References `mxml_index_s::cur_node`, `mxml_index_s::nodes`, and `mxml_index_s::num_nodes`.

Referenced by `main()`.

8.64 /home/dko/projects/mobilec/trunk/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)`

8.64.1 Function Documentation

8.64.1.1 static `mxml_node_t * mxml_new (mxml_node_t * parent, mxml_type_t type)` [static]

Definition at line 614 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().

8.64.1.2 void `mxmlAdd (mxml_node_t * parent, int where, mxml_node_t * child, mxml_node_t * node)`

Definition at line 59 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().

8.64.1.3 void `mxmlDelete (mxml_node_t * node)`

Definition at line 193 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()`.

8.64.1.4 `mxml_node_t* mxmlNewCustom (mxml_node_t * parent, void * data, void(*) (void *) destroy)`

Definition at line 284 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()`.

8.64.1.5 `mxml_node_t* mxmlNewElement (mxml_node_t * parent, const char * name)`

Definition at line 320 of file `mxml-node.c`.

References `mxml_value_u::element`, `MXML_ELEMENT`, `mxml_new()`, `mxml_value_s::name`, `node`, 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()`, `fipa_envelope_Compose__acl_representation()`, `fipa_envelope_Compose__date()`, `fipa_envelope_Compose__envelope()`, `fipa_envelope_Compose__from()`, `fipa_envelope_Compose__params()`, `fipa_envelope_Compose__payload_encoding()`, `fipa_envelope_Compose__to()`, `main()`, `mxml_load_data()`, `scan_file()`, and `xml_new_cdata()`.

8.64.1.6 `mxml_node_t* mxmlNewInteger (mxml_node_t * parent, int integer)`

Definition at line 358 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()`.

8.64.1.7 `mxml_node_t* mxmlNewOpaque (mxml_node_t * parent, const char * opaque)`

Definition at line 389 of file `mxml-node.c`.

References `mxml_new()`, `MXML_OPAQUE`, `node`, `mxml_value_u::opaque`, and `mxml_node_s::value`.

Referenced by `main()`, and `mxml_load_data()`.

8.64.1.8 `mxml_node_t* mxmlNewReal (mxml_node_t * parent, double real)`

Definition at line 427 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()`.

8.64.1.9 `mxml_node_t*` `mxmlNewText (mxml_node_t * parent, int whitespace, const char * string)`

Definition at line 459 of file `mxml-node.c`.

References `mxml_new()`, `MXML_TEXT`, `node`, `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()`, `fipa_envelope_Compose__acl_representation()`, `fipa_envelope_Compose__date()`, `fipa_envelope_Compose__from()`, `fipa_envelope_Compose__payload_encoding()`, `fipa_envelope_Compose__to()`, `main()`, `mxml_load_data()`, and `scan_file()`.

8.64.1.10 `mxml_node_t*` `mxmlNewTextf (mxml_node_t * parent, int whitespace, const char * format, ...)`

Definition at line 503 of file `mxml-node.c`.

References `mxml_new()`, `mxml_strdup()`, `MXML_TEXT`, `node`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::value`, and `mxml_text_s::whitespace`.

8.64.1.11 `void` `mxmlRemove (mxml_node_t * node)`

Definition at line 550 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()`.

8.65 /home/dko/projects/mobilec/trunk/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

8.65.1 Function Documentation

8.65.1.1 void [mxml_error](#) (const char **format*, ...)

Definition at line 46 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\(\)](#).

8.65.1.2 [mxml_type_t mxml_integer_cb](#) ([mxml_node_t](#) **node*)

Definition at line 92 of file mxml-private.c.

References [MXML_INTEGER](#).

8.65.1.3 [mxml_type_t mxml_opaque_cb](#) ([mxml_node_t](#) **node*)

Definition at line 105 of file mxml-private.c.

References [MXML_OPAQUE](#).

8.65.1.4 [mxml_type_t mxml_real_cb](#) ([mxml_node_t](#) **node*)

Definition at line 118 of file mxml-private.c.

References [MXML_REAL](#).

8.65.2 Variable Documentation

8.65.2.1 void(* mxml_error_cb)(const char *) = NULL

Referenced by mxml_error(), and mxmlSetErrorCallback().

8.66 /home/dko/projects/mobilec/trunk/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](#))

8.66.1 Function Documentation

8.66.1.1 [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 48 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\(\)](#), [mxmlIndexNew\(\)](#), [scan_file\(\)](#), [sort_node\(\)](#), [write_documentation\(\)](#), [write_element\(\)](#), [xml_find_sibling\(\)](#), and [xml_get_child\(\)](#).

8.66.1.2 [mxml_node_t * mxmlWalkNext](#) ([mxml_node_t * node](#), [mxml_node_t * top](#), [int descend](#))

Definition at line 130 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\(\)](#).

8.66.1.3 [mxml_node_t * mxmlWalkPrev](#) ([mxml_node_t * node](#), [mxml_node_t * top](#), [int descend](#))

Definition at line 166 of file mxml-search.c.

References [mxml_node_s::last_child](#), [mxml_node_s::parent](#), and [mxml_node_s::prev](#).

8.67 /home/dko/projects/mobilec/trunk/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,...)`

8.67.1 Function Documentation

8.67.1.1 `int mxmlSetCustom (mxml_node_t * node, void * data, void(*) (void *) destroy)`

Definition at line 43 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`.

8.67.1.2 `int mxmlSetElement (mxml_node_t * node, const char * name)`

Definition at line 76 of file mxml-set.c.

References `mxml_value_u::element`, `MXML_ELEMENT`, `mxml_value_s::name`, `mxml_node_s::type`, and `mxml_node_s::value`.

8.67.1.3 `int mxmlSetInteger (mxml_node_t * node, int integer)`

Definition at line 106 of file mxml-set.c.

References `mxml_value_u::integer`, `MXML_INTEGER`, `mxml_node_s::type`, and `mxml_node_s::value`.

8.67.1.4 `int mxmlSetOpaque (mxml_node_t * node, const char * opaque)`

Definition at line 133 of file mxml-set.c.

References `MXML_OPAQUE`, `mxml_value_u::opaque`, `mxml_node_s::type`, and `mxml_node_s::value`.

8.67.1.5 `int mxmlSetReal (mxml_node_t * node, double real)`

Definition at line 163 of file mxml-set.c.

References `MXML_REAL`, `mxml_value_u::real`, `mxml_node_s::type`, and `mxml_node_s::value`.

8.67.1.6 int mxmlSetText (mxml_node_t * *node*, int *whitespace*, const char * *string*)

Definition at line 190 of file mxml-set.c.

References MXML_TEXT, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, mxml_node_s::value, and mxml_text_s::whitespace.

8.67.1.7 int mxmlSetTextf (mxml_node_t * *node*, int *whitespace*, const char * *format*, ...)

Definition at line 222 of file mxml-set.c.

References mxml_strdup(), MXML_TEXT, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, mxml_node_s::value, and mxml_text_s::whitespace.

8.68 /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-string.c File Reference

```
#include "config.h"
```

Functions

- `char * mxml_strdup` (`const char *format, va_list ap`)
- `int mxml_vsnprintf` (`char *buffer, size_t bufsize, const char *format, va_list ap`)

8.68.1 Function Documentation

8.68.1.1 `char* mxml_strdup (const char *format, va_list ap)`

Definition at line 59 of file mxml-string.c.

References `mxml_vsnprintf()`.

Referenced by `mxml_error()`, `mxmlNewTextf()`, and `mxmlSetTextf()`.

8.68.1.2 `int mxml_vsnprintf (char *buffer, size_t bufsize, const char *format, va_list ap)`

Definition at line 104 of file mxml-string.c.

References `size`.

Referenced by `mxml_strdup()`.

8.69 /home/dko/projects/mobilec/trunk/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)

8.69.1 Define Documentation

8.69.1.1 #define MXML_ADD_AFTER 1

Definition at line 65 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()`.

8.69.1.2 #define MXML_ADD_BEFORE 0

Definition at line 64 of file `mxml.h`.

Referenced by `mxmlAdd()`, `scan_file()`, and `sort_node()`.

8.69.1.3 #define MXML_ADD_TO_PARENT NULL

Definition at line 66 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()`.

8.69.1.4 #define MXML_DESCEND 1

Definition at line 55 of file `mxml.h`.

Referenced by `agent_xml_parse__tasks()`, `main()`, `MC_LoadAgentFromFile()`, `mxmlFindElement()`, `mxmlIndexNew()`, and `write_element()`.

8.69.1.5 #define MXML_DESCEND_FIRST -1

Definition at line 57 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()`.

8.69.1.6 `#define MXML_INTEGER_CALLBACK mxml_integer_cb`

Definition at line 45 of file `mxml.h`.

Referenced by `main()`.

8.69.1.7 `#define MXML_NO_CALLBACK 0`

Definition at line 44 of file `mxml.h`.

Referenced by `acc_Thread()`, `agent_xml_compose()`, `fipa_envelope_Compose()`, `fipa_envelope_Parse()`, `main()`, `message_InitializeFromAgent()`, and `message_InitializeFromConnection()`.

8.69.1.8 `#define MXML_NO_DESCEND 0`

Definition at line 56 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()`, `write_documentation()`, `write_element()`, `xml_find_sibling()`, and `xml_get_deep_child()`.

8.69.1.9 `#define MXML_NO_PARENT 0`

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

Referenced by `agent_xml_compose__agent_code()`, `agent_xml_compose__create_row_nodes()`, `main()`, and `scan_file()`.

8.69.1.10 `#define MXML_OPAQUE_CALLBACK mxml_opaque_cb`

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

Referenced by `main()`.

8.69.1.11 `#define MXML_REAL_CALLBACK mxml_real_cb`

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

Referenced by `main()`.

8.69.1.12 `#define MXML_TAB 8`

Definition at line 42 of file `mxml.h`.

Referenced by `mxml_write_ws()`.

8.69.1.13 #define MXML_TEXT_CALLBACK 0

Definition at line 51 of file `mxml.h`.

8.69.1.14 #define MXML_WRAP 72

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

Referenced by `mxml_write_node()`.

8.69.1.15 #define MXML_WS_AFTER_CLOSE 3

Definition at line 62 of file `mxml.h`.

Referenced by `mxml_write_node()`, `whitespace_cb()`, and `ws_cb()`.

8.69.1.16 #define MXML_WS_AFTER_OPEN 1

Definition at line 60 of file `mxml.h`.

Referenced by `mxml_write_node()`, `whitespace_cb()`, and `ws_cb()`.

8.69.1.17 #define MXML_WS_BEFORE_CLOSE 2

Definition at line 61 of file `mxml.h`.

Referenced by `mxml_write_node()`, `whitespace_cb()`, and `ws_cb()`.

8.69.1.18 #define MXML_WS_BEFORE_OPEN 0

Definition at line 59 of file `mxml.h`.

Referenced by `mxml_write_node()`, `whitespace_cb()`, and `ws_cb()`.

8.69.2 Typedef Documentation

8.69.2.1 typedef struct mxml_attr_s mxml_attr_t

8.69.2.2 typedef int(* mxml_custom_load_cb_t)(mxml_node_t *, const char *)

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

8.69.2.3 typedef char*(*) mxml_custom_save_cb_t)(mxml_node_t *)

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

8.69.2.4 typedef struct mxml_custom_s mxml_custom_t

8.69.2.5 typedef struct mxml_value_s mxml_element_t

8.69.2.6 typedef struct mxml_index_s mxml_index_t

8.69.2.7 typedef struct mxml_node_s mxml_node_t

8.69.2.8 typedef struct mxml_text_s mxml_text_t

8.69.2.9 typedef enum mxml_type_e mxml_type_t

8.69.2.10 typedef union mxml_value_u mxml_value_t

8.69.3 Enumeration Type Documentation

8.69.3.1 enum mxml_type_e

Enumerator:

MXML_ELEMENT

MXML_INTEGER

MXML_OPAQUE

MXML_REAL

MXML_TEXT

MXML_CUSTOM

Definition at line 73 of file mxml.h.

8.69.4 Function Documentation

8.69.4.1 void mxml_error (const char * *format*, ...)

Definition at line 46 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().

8.69.4.2 mxml_type_t mxml_integer_cb (mxml_node_t * *node*)

Definition at line 92 of file mxml-private.c.

References MXML_INTEGER.

8.69.4.3 mxml_type_t mxml_opaque_cb (mxml_node_t * *node*)

Definition at line 105 of file mxml-private.c.

References MXML_OPAQUE.

8.69.4.4 mxm_type_t mxm_real_cb (mxm_node_t * node)

Definition at line 118 of file mxm-private.c.

References MXML_REAL.

8.69.4.5 void mxmAdd (mxm_node_t * parent, int where, mxm_node_t * child, mxm_node_t * node)

Definition at line 59 of file mxm-node.c.

References mxm_node_s::child, mxm_node_s::last_child, MXML_ADD_AFTER, MXML_ADD_BEFORE, mxmRemove(), mxm_node_s::next, mxm_node_s::parent, and mxm_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(), mxm_new(), scan_file(), and sort_node().

8.69.4.6 void mxmDelete (mxm_node_t * node)

Definition at line 193 of file mxm-node.c.

References mxm_value_s::attrs, mxm_node_s::child, mxm_value_u::custom, mxm_custom_s::data, mxm_custom_s::destroy, mxm_value_u::element, MXML_CUSTOM, MXML_ELEMENT, MXML_INTEGER, MXML_OPAQUE, MXML_REAL, MXML_TEXT, mxmDelete(), mxmRemove(), mxm_attr_s::name, mxm_value_s::name, mxm_value_s::num_attrs, mxm_value_u::opaque, mxm_text_s::string, mxm_value_u::text, mxm_node_s::type, mxm_attr_s::value, and mxm_node_s::value.

Referenced by add_variable(), agent_datastate_Destroy(), fipa_envelope_Compose(), fipa_envelope_Parse(), main(), message_Destroy(), mxm_load_data(), mxmDelete(), scan_file(), and sort_node().

8.69.4.7 const char* mxmElementGetAttr (mxm_node_t * node, const char * name)

Definition at line 40 of file mxm-attr.c.

References mxm_value_s::attrs, mxm_value_u::element, MXML_ELEMENT, mxm_attr_s::name, mxm_value_s::num_attrs, mxm_node_s::type, mxm_attr_s::value, and mxm_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(), mxm_parse_element(), mxmFindElement(), mxmIndexNew(), scan_file(), sort_node(), type_cb(), and write_documentation().

8.69.4.8 void mxmElementSetAttr (mxm_node_t * node, const char * name, const char * value)

Definition at line 87 of file mxm-attr.c.

References mxm_value_s::attrs, mxm_value_u::element, MXML_ELEMENT, mxm_error(), mxm_value_s::name, mxm_attr_s::name, mxm_value_s::num_attrs, mxm_node_s::type, mxm_attr_s::value, and mxm_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().

8.69.4.9 int mxmlEntityAddCallback (int(*) (const char *name) cb)

Definition at line 62 of file mxml-entity.c.

References callbacks, mxml_error(), and num_callbacks.

8.69.4.10 const char* mxmlEntityGetName (int val)

Definition at line 88 of file mxml-entity.c.

Referenced by mxml_write_name(), and mxml_write_string().

8.69.4.11 int mxmlEntityGetValue (const char * name)

Definition at line 118 of file mxml-entity.c.

References callbacks, and num_callbacks.

Referenced by mxml_get_entity().

8.69.4.12 void mxmlEntityRemoveCallback (int(*) (const char *name) cb)

Definition at line 137 of file mxml-entity.c.

References callbacks, and num_callbacks.

8.69.4.13 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 48 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(), mxmlIndexNew(), scan_file(), sort_node(), write_documentation(), write_element(), xml_find_sibling(), and xml_get_child().

8.69.4.14 void mxmlIndexDelete (mxml_index_t * ind)

Definition at line 55 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().

8.69.4.15 mxml_node_t* mxmlIndexEnum (mxml_index_t * ind)

Definition at line 85 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().

8.69.4.16 mxml_node_t* mxmlIndexFind (mxml_index_t * ind, const char * element, const char * value)

Definition at line 115 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().

8.69.4.17 mxml_index_t* mxmlIndexNew (mxml_node_t * node, const char * element, const char * attr)

Definition at line 298 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, and mxml_node_s::value.

Referenced by main().

8.69.4.18 mxml_node_t* mxmlIndexReset (mxml_index_t * ind)

Definition at line 456 of file mxml-index.c.

References mxml_index_s::cur_node, mxml_index_s::nodes, and mxml_index_s::num_nodes.

Referenced by main().

8.69.4.19 mxml_node_t* mxmlLoadFd (mxml_node_t * top, int fd, mxml_type_t(*) (mxml_node_t *) cb)**8.69.4.20 mxml_node_t* mxmlLoadFile (mxml_node_t * top, FILE * fp, mxml_type_t(*) (mxml_node_t *) cb)****8.69.4.21 mxml_node_t* mxmlLoadString (mxml_node_t * top, const char * s, mxml_type_t(*) (mxml_node_t *) cb)****8.69.4.22 mxml_node_t* mxmlNewCustom (mxml_node_t * parent, void * data, void(*) (void *) destroy)**

Definition at line 284 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().

8.69.4.23 mxml_node_t* mxmlNewElement (mxml_node_t * *parent*, const char * *name*)

Definition at line 320 of file mxml-node.c.

References mxml_value_u::element, MXML_ELEMENT, mxml_new(), mxml_value_s::name, node, 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(), fipa_envelope_Compose__acl_representation(), fipa_envelope_Compose__date(), fipa_envelope_Compose__envelope(), fipa_envelope_Compose__from(), fipa_envelope_Compose__params(), fipa_envelope_Compose__payload_encoding(), fipa_envelope_Compose__to(), main(), mxml_load_data(), scan_file(), and xml_new_cdata().

8.69.4.24 mxml_node_t* mxmlNewInteger (mxml_node_t * *parent*, int *integer*)

Definition at line 358 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().

8.69.4.25 mxml_node_t* mxmlNewOpaque (mxml_node_t * *parent*, const char * *opaque*)

Definition at line 389 of file mxml-node.c.

References mxml_new(), MXML_OPAQUE, node, mxml_value_u::opaque, and mxml_node_s::value.

Referenced by main(), and mxml_load_data().

8.69.4.26 mxml_node_t* mxmlNewReal (mxml_node_t * *parent*, double *real*)

Definition at line 427 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().

8.69.4.27 mxml_node_t* mxmlNewText (mxml_node_t * *parent*, int *whitespace*, const char * *string*)

Definition at line 459 of file mxml-node.c.

References mxml_new(), MXML_TEXT, node, 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(), fipa_envelope_Compose__acl_representation(), fipa_envelope_Compose__date(), fipa_envelope_Compose__from(), fipa_envelope_Compose__payload_encoding(), fipa_envelope_Compose__to(), main(), mxml_load_data(), and scan_file().

8.69.4.28 **mxmxml_node_t*** mxmxmlNewTextf (mxmxml_node_t * *parent*, int *whitespace*, const char * *format*, ...)

Definition at line 503 of file mxmxml-node.c.

References mxmxml_new(), mxmxml_strdupf(), MXMXML_TEXT, node, mxmxml_text_s::string, mxmxml_value_u::text, mxmxml_node_s::value, and mxmxml_text_s::whitespace.

8.69.4.29 **void** mxmxmlRemove (mxmxml_node_t * *node*)

Definition at line 550 of file mxmxml-node.c.

References mxmxml_node_s::child, mxmxml_node_s::last_child, mxmxml_node_s::next, mxmxml_node_s::parent, and mxmxml_node_s::prev.

Referenced by mxmxmlAdd(), and mxmxmlDelete().

8.69.4.30 **char*** mxmxmlSaveAllocString (mxmxml_node_t * *node*, const char *(*)(mxmxml_node_t *, int) *cb*)

8.69.4.31 **int** mxmxmlSaveFd (mxmxml_node_t * *node*, int *fd*, const char *(*)(mxmxml_node_t *, int) *cb*)

8.69.4.32 **int** mxmxmlSaveFile (mxmxml_node_t * *node*, FILE * *fp*, const char *(*)(mxmxml_node_t *, int) *cb*)

8.69.4.33 **int** mxmxmlSaveString (mxmxml_node_t * *node*, char * *buffer*, int *bufsize*, const char *(*)(mxmxml_node_t *, int) *cb*)

8.69.4.34 **int** mxmxmlSetCustom (mxmxml_node_t * *node*, void * *data*, void(*)(void *) *destroy*)

Definition at line 43 of file mxmxml-set.c.

References mxmxml_value_u::custom, mxmxml_custom_s::data, mxmxml_custom_s::destroy, MXMXML_CUSTOM, mxmxml_node_s::type, and mxmxml_node_s::value.

8.69.4.35 **void** mxmxmlSetCustomHandlers (mxmxml_custom_load_cb_t *load*, mxmxml_custom_save_cb_t *save*)

Definition at line 453 of file mxmxml-file.c.

References mxmxml_custom_load_cb, and mxmxml_custom_save_cb.

8.69.4.36 **int** mxmxmlSetElement (mxmxml_node_t * *node*, const char * *name*)

Definition at line 76 of file mxmxml-set.c.

References mxmxml_value_u::element, MXMXML_ELEMENT, mxmxml_value_s::name, mxmxml_node_s::type, and mxmxml_node_s::value.

8.69.4.37 **void** mxmxmlSetErrorCallback (void(*)(const char *) *cb*)

Definition at line 468 of file mxmxml-file.c.

References mxmxml_error_cb.

8.69.4.38 int mxmlSetInteger (mxml_node_t * node, int integer)

Definition at line 106 of file mxml-set.c.

References mxml_value_u::integer, MXML_INTEGER, mxml_node_s::type, and mxml_node_s::value.

8.69.4.39 int mxmlSetOpaque (mxml_node_t * node, const char * opaque)

Definition at line 133 of file mxml-set.c.

References MXML_OPAQUE, mxml_value_u::opaque, mxml_node_s::type, and mxml_node_s::value.

8.69.4.40 int mxmlSetReal (mxml_node_t * node, double real)

Definition at line 163 of file mxml-set.c.

References MXML_REAL, mxml_value_u::real, mxml_node_s::type, and mxml_node_s::value.

8.69.4.41 int mxmlSetText (mxml_node_t * node, int whitespace, const char * string)

Definition at line 190 of file mxml-set.c.

References MXML_TEXT, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, mxml_node_s::value, and mxml_text_s::whitespace.

8.69.4.42 int mxmlSetTextf (mxml_node_t * node, int whitespace, const char * format, ...)

Definition at line 222 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.

8.69.4.43 mxml_node_t* mxmlWalkNext (mxml_node_t * node, mxml_node_t * top, int descend)

Definition at line 130 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().

8.69.4.44 mxml_node_t* mxmlWalkPrev (mxml_node_t * node, mxml_node_t * top, int descend)

Definition at line 166 of file mxml-search.c.

References mxml_node_s::last_child, mxml_node_s::parent, and mxml_node_s::prev.

8.70 /home/dko/projects/mobilec/trunk/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[]`)

8.70.1 Define Documentation

8.70.1.1 `#define STATE_C_COMMENT 2`

Definition at line 122 of file `mxmldoc.c`.

Referenced by `scan_file()`.

8.70.1.2 `#define STATE_CHARACTER 5`

Definition at line 125 of file `mxmldoc.c`.

Referenced by `scan_file()`.

8.70.1.3 `#define STATE_CXX_COMMENT 3`

Definition at line 123 of file `mxmldoc.c`.

Referenced by `scan_file()`.

8.70.1.4 `#define STATE_IDENTIFIER 6`

Definition at line 126 of file mxmldoc.c.

Referenced by `scan_file()`.

8.70.1.5 `#define STATE_NONE 0`

Definition at line 120 of file mxmldoc.c.

Referenced by `scan_file()`.

8.70.1.6 `#define STATE_PREPROCESSOR 1`

Definition at line 121 of file mxmldoc.c.

Referenced by `scan_file()`.

8.70.1.7 `#define STATE_STRING 4`

Definition at line 124 of file mxmldoc.c.

Referenced by `scan_file()`.

8.70.2 Function Documentation

8.70.2.1 `static mxml_node_t * add_variable (mxml_node_t * parent, const char * name, mxml_node_t * type)` `[static]`

Definition at line 306 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()`.

8.70.2.2 `int main (int argc, char * argv[])`

Examples:

[LibMCCppEx/LibMCCppEx.cpp](#).

Definition at line 152 of file mxmldoc.c.

References `MXML_DESCEND`, `MXML_NO_CALLBACK`, `mxmlDelete()`, `mxmlElementSetAttr()`, `mxmlFindElement()`, `mxmlLoadFile()`, `mxmlNewElement()`, `mxmlSaveFile()`, `scan_file()`, `write_documentation()`, and `ws_cb()`.

8.70.2.3 `static void safe_strcpy (char * dst, const char * src)` `[static]`

Definition at line 420 of file mxmldoc.c.

Referenced by `update_comment()`.

8.70.2.4 static int scan_file (const char *filename, FILE *fp, mxml_node_t *doc) [static]

Definition at line 435 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().

8.70.2.5 static void sort_node (mxml_node_t *tree, mxml_node_t *func) [static]

Definition at line 1610 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().

8.70.2.6 static void update_comment (mxml_node_t *parent, mxml_node_t *comment) [static]

Definition at line 1698 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().

8.70.2.7 static void write_documentation (mxml_node_t *doc) [static]

Definition at line 1798 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().

8.70.2.8 static void write_element (mxml_node_t *doc, mxml_node_t *element) [static]

Definition at line 2608 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().

8.70.2.9 static void write_string (const char * *s*) [static]

Definition at line 2653 of file mxmldoc.c.

Referenced by write_element().

8.70.2.10 static const char * ws_cb (mxml_node_t * *node*, int *where*) [static]

Definition at line 2711 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().

8.71 /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/class.cxx File Reference

Data Structures

- class `foo_c`

Variables

- `foo_c f`
- `foo_c foo`
- `bar = b`

8.71.1 Variable Documentation

8.71.1.1 `bar = b`

Definition at line 77 of file class.cxx.

8.71.1.2 `foo_c f`

Referenced by `foo_float_function()`, `main()`, `md2_file()`, `md4_file()`, `md5_check()`, `md5_file()`, `sha1_check()`, `sha1_file()`, `sha2_check()`, `sha2_file()`, `sha4_file()`, `x509_read_crtfile()`, and `x509_read_keyfile()`.

8.71.1.3 `foo_c foo`

8.72 /home/dko/projects/mobilec/trunk/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](#) }

8.72.1 Typedef Documentation

8.72.1.1 typedef enum [foo_enum_e](#) [foo_enum_t](#)

8.72.2 Enumeration Type Documentation

8.72.2.1 enum [foo_enum_e](#)

Enumerator:

FOO_ONE
FOO_TWO
FOO_RED
FOO_BLUE

Definition at line 1 of file enum.cxx.

8.73 /home/dko/projects/mobilec/trunk/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)

8.73.1 Function Documentation

8.73.1.1 [int](#) [foo_default_int](#) ([int](#) *one*, [int](#) *two* = 2)

Definition at line 65 of file function.cxx.

8.73.1.2 [int](#) [foo_default_string](#) ([int](#) *one*, const char * *two* = "2")

Definition at line 45 of file function.cxx.

8.73.1.3 [float](#) [foo_float_function](#) ([int](#) *one*, const char * *two*)

Definition at line 26 of file function.cxx.

References [f](#).

8.73.1.4 [void](#) [foo_void_function](#) ([int](#) *one*, float * *two*, const char * *three*)

Definition at line 6 of file function.cxx.

8.74 /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/struct.cxx File Reference

Data Structures

- struct [foo_s](#)

Typedefs

- typedef struct [foo_s](#) [foo_t](#)

8.74.1 Typedef Documentation

8.74.1.1 typedef struct [foo_s](#) [foo_t](#)

8.75 /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/testmxml.c File Reference

```
#include "config.h"
#include "mxml.h"
#include <unistd.h>
#include <fcntl.h>
```

Defines

- `#define O_BINARY 0`

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[])`

8.75.1 Define Documentation

8.75.1.1 `#define O_BINARY 0`

Definition at line 39 of file testmxml.c.

Referenced by `main()`.

8.75.2 Function Documentation

8.75.2.1 `int main (int argc, char * argv[])`

Definition at line 56 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`, `O_BINARY`, `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()`.

8.75.2.2 `mxml_type_t type_cb (mxml_node_t * node)`

Definition at line 539 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().

8.75.2.3 `const char* whitespace_cb (mxml_node_t * node, int where)`

Definition at line 246 of file xml_helper.c.

References MXML_WS_AFTER_CLOSE, and MXML_WS_BEFORE_OPEN.

8.76 /home/dko/projects/mobilec/trunk/src/security/asm.c File Reference

```
#include "../include/mc_platform.h"
#include "../include/message.h"
#include "asm.h"
#include "asm_message_composer.h"
#include "config.h"
#include "mc_dh.h"
```

8.77 /home/dko/projects/mobilec/trunk/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"
```

8.78 `/home/dko/projects/mobilec/trunk/src/security/asm_message_composer.c` File Reference

```
#include "asm_message_composer.h"  
#include "config.h"  
#include "../include/mc_platform.h"
```


8.79 /home/dko/projects/mobilec/trunk/src/security/asm_message_composer.h File Reference

```
#include <mxml.h>
#include "asm.h"
#include "config.h"
```

8.80 `/home/dko/projects/mobilec/trunk/src/security/asm_message_parser.c` File Reference

```
#include "asm_message_parser.h"  
#include "../include/xml_helper.h"  
#include "config.h"
```

8.81 /home/dko/projects/mobilec/trunk/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"
```

8.82 /home/dko/projects/mobilec/trunk/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-0.7/include/xyssl/havege.h"
#include "xyssl-0.7/include/xyssl/bignum.h"
```

8.83 /home/dko/projects/mobilec/trunk/src/security/asm_node.h File Reference

```
#include <mxml.h>
#include <netinet/in.h>
#include "../include/macros.h"
#include "../include/message.h"
#include "xyssl-0.7/include/xyssl/dhm.h"
#include "xyssl-0.7/include/xyssl/rsa.h"
#include "xyssl-0.7/include/xyssl/aes.h"
#include "config.h"
```

8.84 /home/dko/projects/mobilec/trunk/src/security/mc_dh.c File Reference

```
#include <stdio.h>
#include "../include/mc_error.h"
#include "xyssl-0.7/include/xyssl/dhm.h"
#include "xyssl-0.7/include/xyssl/havege.h"
#include "mc_dh.h"
#include "asm_node.h"
#include "config.h"
```

8.85 /home/dko/projects/mobilec/trunk/src/security/mc_dh.h File Reference

```
#include "xyssl-0.7/include/xyssl/rsa.h"
#include "asm_node.h"
#include "config.h"
```

8.86 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/aes.h File Reference

Data Structures

- struct [aes_context](#)
AES context structure.

Functions

- void [aes_set_key](#) ([aes_context](#) *ctx, unsigned char *key, int keysize)
AES key schedule.
- void [aes_encrypt](#) ([aes_context](#) *ctx, unsigned char input[16], unsigned char output[16])
AES block encryption (ECB mode).
- void [aes_decrypt](#) ([aes_context](#) *ctx, unsigned char input[16], unsigned char output[16])
AES block decryption (ECB mode).
- void [aes_cbc_encrypt](#) ([aes_context](#) *ctx, unsigned char iv[16], unsigned char *input, unsigned char *output, int len)
AES-CBC buffer encryption.
- void [aes_cbc_decrypt](#) ([aes_context](#) *ctx, unsigned char iv[16], unsigned char *input, unsigned char *output, int len)
AES-CBC buffer decryption.
- int [aes_self_test](#) (int verbose)
Checkup routine.

8.86.1 Detailed Description

Definition in file [aes.h](#).

8.86.2 Function Documentation

8.86.2.1 void [aes_cbc_decrypt](#) ([aes_context](#) * ctx, unsigned char iv[16], unsigned char * input, unsigned char * output, int len)

AES-CBC buffer decryption.

Parameters:

- ctx* AES context
- iv* initialization vector (modified after use)
- input* buffer holding the ciphertext

output buffer holding the plaintext
len length of the data to be decrypted

Definition at line 822 of file aes.c.

Referenced by ssl_decrypt_buf().

8.86.2.2 void aes_cbc_encrypt (aes_context * ctx, unsigned char iv[16], unsigned char * input, unsigned char * output, int len)

AES-CBC buffer encryption.

Parameters:

ctx AES context
iv initialization vector (modified after use)
input buffer holding the plaintext
output buffer holding the ciphertext
len length of the data to be encrypted

Definition at line 797 of file aes.c.

Referenced by main(), and ssl_encrypt_buf().

8.86.2.3 void aes_decrypt (aes_context * ctx, unsigned char input[16], unsigned char output[16])

AES block decryption (ECB mode).

Parameters:

ctx AES context
input ciphertext block
output plaintext block

Definition at line 706 of file aes.c.

Referenced by main().

8.86.2.4 void aes_encrypt (aes_context * ctx, unsigned char input[16], unsigned char output[16])

AES block encryption (ECB mode).

Parameters:

ctx AES context
input plaintext block
output ciphertext block

AES block encryption (ECB mode)

Definition at line 615 of file aes.c.

Referenced by main().

8.86.2.5 int aes_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 920 of file aes.c.

Referenced by main().

8.86.2.6 void aes_set_key (aes_context * *ctx*, unsigned char * *key*, int *keysize*)

AES key schedule.

Parameters:

ctx AES context to be initialized

key the secret key

keysize must be 128, 192 or 256

Referenced by main(), and ssl_derive_keys().

8.87 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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)

8.87.1 Detailed Description

Definition in file [arc4.h](#).

8.87.2 Function Documentation

8.87.2.1 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 60 of file [arc4.c](#).

References [arc4_context::m](#), [arc4_context::x](#), and [arc4_context::y](#).

Referenced by [main\(\)](#), [ssl_decrypt_buf\(\)](#), and [ssl_encrypt_buf\(\)](#).

8.87.2.2 int [arc4_self_test](#) (int verbose)

Definition at line 147 of file [arc4.c](#).

Referenced by [main\(\)](#).

8.87.2.3 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 main(), and ssl_derive_keys().

8.88 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/base64.h File Reference

Defines

- `#define ERR_BASE64_BUFFER_TOO_SMALL 0x0010`
- `#define 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.

8.88.1 Detailed Description

Definition in file [base64.h](#).

8.88.2 Define Documentation

8.88.2.1 `#define ERR_BASE64_BUFFER_TOO_SMALL 0x0010`

Definition at line 11 of file [base64.h](#).

Referenced by [base64_decode\(\)](#), and [base64_encode\(\)](#).

8.88.2.2 `#define ERR_BASE64_INVALID_CHARACTER 0x0012`

Definition at line 12 of file [base64.h](#).

Referenced by [base64_decode\(\)](#), [x509_add_certs\(\)](#), and [x509_parse_key\(\)](#).

8.88.3 Function Documentation

8.88.3.1 `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 (updated after call)
- src* source buffer

slen amount of data to be decoded

Returns:

0 if successful, ERR_BASE64_BUFFER_TOO_SMALL, or ERR_BASE64_INVALID_DATA if an invalid char is found. *dlen is always updated to reflect to amount of data that was written (or would have been written)

Note:

Call this function with *dlen = 0 to obtain the required buffer size in *dlen

Definition at line 118 of file base64.c.

References base64_dec_map, ERR_BASE64_BUFFER_TOO_SMALL, and ERR_BASE64_INVALID_CHARACTER.

Referenced by x509_add_certs(), and x509_parse_key().

8.88.3.2 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 (updated after call)

src source buffer

slen amount of data to be encoded

Returns:

0 if successful, or ERR_BASE64_BUFFER_TOO_SMALL. *dlen is always updated to reflect to amount of data that was written (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 ERR_BASE64_BUFFER_TOO_SMALL.

8.88.3.3 int base64_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 245 of file base64.c.

Referenced by main().

8.89 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/bignum.h File Reference

```
#include <stdio.h>
#include "ssl_conf.h"
```

Data Structures

- struct [mpi](#)
MPI structure.

Defines

- #define [ERR_MPI_FILE_IO_ERROR](#) 0x0002
- #define [ERR_MPI_INVALID_CHARACTER](#) 0x0004
- #define [ERR_MPI_INVALID_PARAMETER](#) 0x0006
- #define [ERR_MPI_BUFFER_TOO_SMALL](#) 0x0008
- #define [ERR_MPI_NEGATIVE_VALUE](#) 0x000A
- #define [ERR_MPI_DIVISION_BY_ZERO](#) 0x000C
- #define [ERR_MPI_NOT_ACCEPTABLE](#) 0x000E
- #define [CHK](#)(fc) if((ret = fc) != 0) goto cleanup
- #define [ciL](#) ([int](#)) sizeof([t_int](#))
- #define [biL](#) ([ciL](#) << 3)
- #define [biH](#) ([ciL](#) << 2)

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 nblimbs)
Enlarge X 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_read_string (mpi *X, int radix, char *s)`
Import X from an ASCII string.
- `int mpi_write_string (mpi *X, int radix, char *s, int *slen)`
Export X into an ASCII string.
- `int mpi_read_file (mpi *X, int radix, FILE *fin)`
Read X from an opened file.
- `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_msb (mpi *X)`
Return the total size in bits, without leading 0s.
- `int mpi_lsb (mpi *X)`
Return the number of least significant bits.
- `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)`

Miller-Rabin primality test.

- `int mpi_gen_prime (mpi *X, int nbits, int dh_flag, int(*rng_f)(void *), void *rng_d)`

Prime number generation.

- `int mpi_self_test (int verbose)`

Checkup routine.

8.89.1 Detailed Description

Definition in file `bignum.h`.

8.89.2 Define Documentation

8.89.2.1 #define biH (ciL << 2)

Definition at line 48 of file bignum.h.

Referenced by mpi_div_mpi(), and mpi_mod_int().

8.89.2.2 #define biL (ciL << 3)

Definition at line 47 of file bignum.h.

Referenced by mpi_div_mpi(), mpi_exp_mod(), mpi_lsb(), mpi_montg_init(), mpi_msb(), mpi_shift_l(), and mpi_shift_r().

8.89.2.3 #define CHK(fc) if((ret = fc) != 0) goto cleanup

Definition at line 23 of file bignum.h.

Referenced by dhm_calc_secret(), dhm_make_params(), dhm_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_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(), rsa_public(), rsa_read_private(), rsa_read_public(), rsa_write_private(), and rsa_write_public().

8.89.2.4 #define ciL (int) sizeof(t_int)

Definition at line 46 of file bignum.h.

Referenced by mpi_copy(), mpi_free(), mpi_grow(), mpi_lset(), mpi_montmul(), mpi_read_binary(), mpi_read_string(), mpi_write_binary(), and mpi_write_string().

8.89.2.5 #define ERR_MPI_BUFFER_TOO_SMALL 0x0008

Definition at line 18 of file bignum.h.

Referenced by mpi_write_binary(), and mpi_write_string().

8.89.2.6 #define ERR_MPI_DIVISION_BY_ZERO 0x000C

Definition at line 20 of file bignum.h.

Referenced by mpi_div_mpi(), and mpi_mod_int().

8.89.2.7 #define ERR_MPI_FILE_IO_ERROR 0x0002

Definition at line 15 of file bignum.h.

Referenced by mpi_read_file(), and mpi_write_file().

8.89.2.8 #define ERR_MPI_INVALID_CHARACTER 0x0004

Definition at line 16 of file bignum.h.

Referenced by mpi_get_digit().

8.89.2.9 #define ERR_MPI_INVALID_PARAMETER 0x0006

Definition at line 17 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().

8.89.2.10 #define ERR_MPI_NEGATIVE_VALUE 0x000A

Definition at line 19 of file bignum.h.

Referenced by mpi_sub_abs().

8.89.2.11 #define ERR_MPI_NOT_ACCEPTABLE 0x000E

Definition at line 21 of file bignum.h.

Referenced by mpi_gen_prime(), mpi_inv_mod(), and mpi_is_prime().

8.89.3 Typedef Documentation

8.89.3.1 typedef unsigned long long t_dbl

Definition at line 41 of file bignum.h.

8.89.3.2 typedef unsigned long t_int

Definition at line 32 of file bignum.h.

8.89.4 Function Documentation

8.89.4.1 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 641 of file bignum.c.

References CHK, mpi_copy(), mpi_grow(), mpi::n, and mpi::p.

Referenced by mpi_add_mpi(), and mpi_sub_mpi().

8.89.4.2 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 806 of file bignum.c.

References mpi_add_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by mpi_gen_prime(), and mpi_read_string().

8.89.4.3 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 744 of file bignum.c.

References CHK, mpi_add_abs(), 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().

8.89.4.4 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 562 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().

8.89.4.5 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 625 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().

8.89.4.6 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 592 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()`, `rsa_check_privkey()`, `rsa_gen_key()`, `rsa_private()`, and `rsa_public()`.

8.89.4.7 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 115 of file bignum.c.

References `CHK`, `ciL`, `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()`.

8.89.4.8 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, `ERR_MPI_DIVISION_BY_ZERO` if $b == 0$

Note:

Either Q or R can be NULL.

Definition at line 1117 of file bignum.c.

References `mpi_div_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `main()`, and `mpi_write_hlp()`.

8.89.4.9 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, `ERR_MPI_DIVISION_BY_ZERO` if $B == 0$

Note:

Either Q or R can be NULL.

Definition at line 953 of file bignum.c.

References biH, biL, CHK, ERR_MPI_DIVISION_BY_ZERO, mpi_add_mpi(), 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, and mpi::s.

Referenced by mpi_div_int(), and mpi_mod_mpi().

8.89.4.10 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, ERR_MPI_INVALID_PARAMETER 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 1263 of file bignum.c.

References biL, CHK, ERR_MPI_INVALID_PARAMETER, 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, and mpi::p.

Referenced by dhm_calc_secret(), dhm_make_params(), dhm_make_public(), main(), mpi_is_prime(), rsa_private(), and rsa_public().

8.89.4.11 void mpi_free (mpi * X, ...)

Unallocate one or more [mpi](#).

Definition at line 66 of file bignum.c.

References ciL, mpi::n, and mpi::p.

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_sub_abs(), mpi_write_string(), rsa_check_privkey(), rsa_free(), rsa_gen_key(), rsa_private(), and rsa_public().

8.89.4.12 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 1440 of file bignum.c.

References CHK, mpi_cmp_int(), mpi_cmp_mpi(), mpi_copy(), mpi_free(), mpi_init(), mpi_lsb(), mpi_lset(), mpi_mul_mpi(), mpi_shift_l(), mpi_shift_r(), mpi_sub_abs(), mpi::p, and mpi::s.

Referenced by mpi_inv_mod(), rsa_check_privkey(), and rsa_gen_key().

8.89.4.13 int mpi_gen_prime (mpi * X, int nbits, int dh_flag, int(*) (void *) rng_f, void * rng_d)

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

rng_f points to the RNG function

rng_d points to the RNG data

Returns:

0 if successful (probably prime), 1 if memory allocation failed, ERR_MPI_INVALID_PARAMETER if nbits is < 3

Definition at line 1710 of file bignum.c.

References BITS_TO_LIMBS, CHK, ERR_MPI_INVALID_PARAMETER, ERR_MPI_NOT_ACCEPTABLE, mpi_add_int(), mpi_free(), mpi_grow(), mpi_init(), mpi_is_prime(), mpi_lset(), mpi_msb(), mpi_shift_l(), mpi_shift_r(), mpi_sub_int(), and mpi::p.

Referenced by main(), and rsa_gen_key().

8.89.4.14 int mpi_grow (mpi * X, int nlimbs)

Enlarge X to the specified number of limbs.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 91 of file bignum.c.

References ciL, mpi::n, mpi::p, and mpi::s.

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().

8.89.4.15 void mpi_init (mpi * X, ...)

Initialize one or more [mpi](#).

Definition at line 48 of file bignum.c.

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_sub_abs(), mpi_write_string(), rsa_check_privkey(), rsa_gen_key(), rsa_private(), and rsa_public().

8.89.4.16 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, ERR_MPI_INVALID_PARAMETER if N is negative or nil ERR_MPI_NOT_ACCEPTABLE if A has no inverse mod N

Definition at line 1488 of file bignum.c.

References CHK, ERR_MPI_INVALID_PARAMETER, ERR_MPI_NOT_ACCEPTABLE, mpi_add_mpi(), 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(), and mpi::p.

Referenced by main(), and rsa_gen_key().

8.89.4.17 int mpi_is_prime (mpi * X)

Miller-Rabin primality test.

Returns:

0 if successful (probably prime), 1 if memory allocation failed, ERR_MPI_NOT_ACCEPTABLE if X is not prime

Definition at line 1614 of file bignum.c.

References CHK, ERR_MPI_NOT_ACCEPTABLE, 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, and small_prime.

Referenced by main(), and mpi_gen_prime().

8.89.4.18 int mpi_lsb (mpi * X)

Return the number of least significant bits.

Definition at line 458 of file bignum.c.

References biL, int, mpi::n, and mpi::p.

Referenced by mpi_gcd(), and mpi_is_prime().

8.89.4.19 int mpi_lset (mpi * X, int z)

Set value from integer.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 154 of file bignum.c.

References CHK, ciL, mpi_grow(), mpi::n, mpi::p, and mpi::s.

Referenced by 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().

8.89.4.20 `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, `ERR_MPI_DIVISION_BY_ZERO` if $b == 0$,

Definition at line 1153 of file `bignum.c`.

References `biH`, `ERR_MPI_DIVISION_BY_ZERO`, `mpi::n`, and `mpi::p`.

Referenced by `mpi_is_prime()`, and `mpi_write_hlp()`.

8.89.4.21 `int mpi_mod_mpi (mpi * R, mpi * A, mpi * B)`

Modulo: $R = A \bmod B$.

Returns:

0 if successful, 1 if memory allocation failed, `ERR_MPI_DIVISION_BY_ZERO` if $B == 0$

Definition at line 1133 of file `bignum.c`.

References `CHK`, `mpi_add_mpi()`, `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_gen_key()`, and `rsa_private()`.

8.89.4.22 `int mpi_msb (mpi * X)`

Return the total size in bits, without leading 0s.

Definition at line 440 of file `bignum.c`.

References `biL`, `mpi::n`, and `mpi::p`.

Referenced by `dhm_calc_secret()`, `dhm_make_params()`, `dhm_read_params()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gen_prime()`, `mpi_is_prime()`, `mpi_shift_l()`, `mpi_write_binary()`, `mpi_write_string()`, `rsa_check_pubkey()`, `rsa_gen_key()`, `rsa_read_private()`, `rsa_read_public()`, `x509_add_certs()`, and `x509_parse_key()`.

8.89.4.23 `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 937 of file `bignum.c`.

References `mpi_mul_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_div_mpi()`, and `mpi_read_string()`.

8.89.4.24 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 902 of file bignum.c.

References 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(), rsa_check_privkey(), rsa_gen_key(), and rsa_private().

8.89.4.25 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 395 of file bignum.c.

References CHARS_TO_LIMBS, CHK, ciL, mpi_grow(), mpi_lset(), and mpi::p.

Referenced by asn1_get_mpi(), dhm_read_bignum(), dhm_read_public(), rsa_private(), and rsa_public().

8.89.4.26 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 ERR_MPI_XXX error code

Definition at line 332 of file bignum.c.

References ERR_MPI_FILE_IO_ERROR, mpi_get_digit(), and mpi_read_string().

Referenced by main(), rsa_read_private(), and rsa_read_public().

8.89.4.27 int mpi_read_string (mpi * X, int radix, char * s)

Import X from an ASCII string.

Parameters:

X destination [mpi](#)
radix input numeric base
s null-terminated string buffer

Returns:

0 if successful, or an ERR_MPI_XXX error code

Definition at line 188 of file bignum.c.

References [BITS_TO_LIMBS](#), [CHK](#), [ciL](#), [ERR_MPI_INVALID_PARAMETER](#), [int](#), [mpi_add_int\(\)](#), [mpi_free\(\)](#), [mpi_get_digit\(\)](#), [mpi_grow\(\)](#), [mpi_init\(\)](#), [mpi_lset\(\)](#), [mpi_mul_int\(\)](#), [mpi::p](#), and [mpi::s](#).

Referenced by [main\(\)](#), and [mpi_read_file\(\)](#).

8.89.4.28 int mpi_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 1912 of file bignum.c.

Referenced by [main\(\)](#).

8.89.4.29 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 473 of file bignum.c.

References [biL](#), [BITS_TO_LIMBS](#), [CHK](#), [mpi_grow\(\)](#), [mpi_msb\(\)](#), [mpi::n](#), and [mpi::p](#).

Referenced by [mpi_div_mpi\(\)](#), [mpi_exp_mod\(\)](#), [mpi_gcd\(\)](#), and [mpi_gen_prime\(\)](#).

8.89.4.30 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 522 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()`.

8.89.4.31 `int mpi_sub_abs (mpi * X, mpi * A, mpi * B)`

Unsigned subtraction: $X = |A| - |B|$.

Returns:

0 if successful, `ERR_MPI_NEGATIVE_VALUE` if B is greater than A

Definition at line 708 of file bignum.c.

References `CHK`, `ERR_MPI_NEGATIVE_VALUE`, `mpi_cmp_abs()`, `mpi_copy()`, `mpi_free()`, `mpi_init()`, `mpi_sub_hlp()`, `mpi::n`, and `mpi::p`.

Referenced by `mpi_add_mpi()`, `mpi_gcd()`, and `mpi_sub_mpi()`.

8.89.4.32 `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 822 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()`.

8.89.4.33 `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 775 of file bignum.c.

References `CHK`, `mpi_add_abs()`, `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()`.

8.89.4.34 `void mpi_swap (mpi * X, mpi * Y)`

Swap the contents of X and Y.

Definition at line 142 of file bignum.c.

Referenced by `rsa_gen_key()`.

8.89.4.35 `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, 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 417 of file `bignum.c`.

References `ciL`, `ERR_MPI_BUFFER_TOO_SMALL`, `mpi_msb()`, and `mpi::p`.

Referenced by `dhm_calc_secret()`, `dhm_make_public()`, `rsa_private()`, and `rsa_public()`.

8.89.4.36 `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 ERR_MPI_XXX error code

Note:

Set *fout* == NULL to print *X* on the console.

Definition at line 357 of file `bignum.c`.

References `CHK`, `ERR_MPI_FILE_IO_ERROR`, and `mpi_write_string()`.

Referenced by `main()`, `rsa_write_private()`, and `rsa_write_public()`.

8.89.4.37 `int mpi_write_string (mpi * X, int radix, char * s, int * slen)`

Export *X* 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 ERR_MPI_XXX error code

Note:

Call this function with **slen* = 0 to obtain the minimum required buffer size in **slen*.

Definition at line 270 of file bignum.c.

References CHK, ciL, ERR_MPI_BUFFER_TOO_SMALL, ERR_MPI_INVALID_PARAMETER, mpi_copy(), mpi_free(), mpi_init(), mpi_msb(), mpi_write_hlp(), mpi::n, mpi::p, and mpi::s.

Referenced by mpi_write_file().

8.90 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/bn_asm.h File Reference

Defines

- #define [MULADDC_INIT](#)
- #define [MULADDC_CORE](#)
- #define [MULADDC_STOP](#) }

8.90.1 Detailed Description

Multiply source vector [s] with b, add result to destination vector [d] and set carry c.

Currently supports:

. IA-32 (386+) . IA-32 (SSE2) . AMD64 / EM64T . Motorola 68000 . PowerPC, 32-bit . PowerPC, 64-bit
. SPARC v8 . MicroBlaze . TriCore . ARM v3+ . Alpha . MIPS32 . C, generic . C, longlong

Definition in file [bn_asm.h](#).

8.90.2 Define Documentation

8.90.2.1 #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 594 of file [bn_asm.h](#).

Referenced by [mpi_mul_hlp\(\)](#).

8.90.2.2 #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 587 of file [bn_asm.h](#).

Referenced by [mpi_mul_hlp\(\)](#).

8.90.2.3 #define MULADDC_STOP }

Definition at line 608 of file bn_asm.h.

Referenced by mpi_mul_hlp().

8.91 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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](#) []

8.91.1 Detailed Description

Definition in file [certs.h](#).

8.91.2 Variable Documentation

8.91.2.1 char test_ca_cert[]

Definition at line 21 of file [certs.c](#).

Referenced by [main\(\)](#).

8.91.2.2 char test_ca_key[]

Definition at line 45 of file [certs.c](#).

8.91.2.3 char test_ca_pwd[]

Definition at line 77 of file [certs.c](#).

8.91.2.4 char test_cli_cert[]

Definition at line 130 of file [certs.c](#).

Referenced by [main\(\)](#).

8.91.2.5 char test_cli_key[]

Definition at line 152 of file [certs.c](#).

Referenced by [main\(\)](#).

8.91.2.6 char test_srv_cert[]

Definition at line 79 of file certs.c.

Referenced by main().

8.91.2.7 char test_srv_key[]

Definition at line 101 of file certs.c.

Referenced by main().

8.91.2.8 char xyssl_ca_cert[]

Definition at line 181 of file certs.c.

Referenced by main().

8.92 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/des.h File Reference

Data Structures

- struct [des_context](#)
DES context structure.
- struct [des3_context](#)
Triple-DES context structure.

Functions

- void [des_set_key](#) ([des_context](#) *ctx, unsigned char key[8])
DES key schedule (56-bit).
- void [des_encrypt](#) ([des_context](#) *ctx, unsigned char input[8], unsigned char output[8])
DES block encryption (ECB mode).
- void [des_decrypt](#) ([des_context](#) *ctx, unsigned char input[8], unsigned char output[8])
DES block decryption (ECB mode).
- void [des_cbc_encrypt](#) ([des_context](#) *ctx, unsigned char iv[8], unsigned char *input, unsigned char *output, [int](#) len)
DES-CBC buffer encryption.
- void [des_cbc_decrypt](#) ([des_context](#) *ctx, unsigned char iv[8], unsigned char *input, unsigned char *output, [int](#) len)
DES-CBC buffer decryption.
- void [des3_set_2keys](#) ([des3_context](#) *ctx, unsigned char key[16])
Triple-DES key schedule (112-bit).
- void [des3_set_3keys](#) ([des3_context](#) *ctx, unsigned char key[24])
Triple-DES key schedule (168-bit).
- void [des3_encrypt](#) ([des3_context](#) *ctx, unsigned char input[8], unsigned char output[8])
Triple-DES block encryption (ECB mode).
- void [des3_decrypt](#) ([des3_context](#) *ctx, unsigned char input[8], unsigned char output[8])
Triple-DES block decryption (ECB mode).
- void [des3_cbc_encrypt](#) ([des3_context](#) *ctx, unsigned char iv[8], unsigned char *input, unsigned char *output, [int](#) len)
3DES-CBC buffer encryption
- void [des3_cbc_decrypt](#) ([des3_context](#) *ctx, unsigned char iv[8], unsigned char *input, unsigned char *output, [int](#) len)

3DES-CBC buffer decryption

- [int des_self_test](#) ([int](#) verbose)

8.92.1 Detailed Description

Definition in file [des.h](#).

8.92.2 Function Documentation

8.92.2.1 **void des3_cbc_decrypt** (**des3_context** * *ctx*, unsigned char *iv*[8], unsigned char * *input*, unsigned char * *output*, int *len*)

3DES-CBC buffer decryption

Parameters:

- ctx* 3DES context
- iv* initialization vector (modified after use)
- input* buffer holding the ciphertext
- output* buffer holding the plaintext
- len* length of the data to be decrypted

Definition at line 611 of file [des.c](#).

References [des3_crypt\(\)](#), and [des3_context::dsk](#).

Referenced by [ssl_decrypt_buf\(\)](#), and [x509_des3_decrypt\(\)](#).

8.92.2.2 **void des3_cbc_encrypt** (**des3_context** * *ctx*, unsigned char *iv*[8], unsigned char * *input*, unsigned char * *output*, int *len*)

3DES-CBC buffer encryption

Parameters:

- ctx* 3DES context
- iv* initialization vector (modified after use)
- input* buffer holding the plaintext
- output* buffer holding the ciphertext
- len* length of the data to be encrypted

Definition at line 586 of file [des.c](#).

References [des3_crypt\(\)](#), and [des3_context::esk](#).

Referenced by [main\(\)](#), and [ssl_encrypt_buf\(\)](#).

8.92.2.3 void des3_decrypt (des3_context * ctx, unsigned char input[8], unsigned char output[8])

Triple-DES block decryption (ECB mode).

Parameters:

ctx 3DES context
input ciphertext block
output plaintext block

Definition at line 576 of file des.c.

References des3_crypt(), and des3_context::dsk.

8.92.2.4 void des3_encrypt (des3_context * ctx, unsigned char input[8], unsigned char output[8])

Triple-DES block encryption (ECB mode).

Parameters:

ctx 3DES context
input plaintext block
output ciphertext block

Definition at line 566 of file des.c.

References des3_crypt(), and des3_context::esk.

8.92.2.5 void des3_set_2keys (des3_context * ctx, unsigned char key[16])

Triple-DES key schedule (112-bit).

Parameters:

ctx 3DES context to be initialized
key 16-byte secret key

Definition at line 472 of file des.c.

References des_main_ks(), des3_context::dsk, and des3_context::esk.

8.92.2.6 void des3_set_3keys (des3_context * ctx, unsigned char key[24])

Triple-DES key schedule (168-bit).

Parameters:

ctx 3DES context to be initialized
key 24-byte secret key

Definition at line 498 of file des.c.

References des_main_ks(), des3_context::dsk, and des3_context::esk.

Referenced by main(), ssl_derive_keys(), and x509_des3_decrypt().

8.92.2.7 void des_cbc_decrypt (des_context * ctx, unsigned char iv[8], unsigned char * input, unsigned char * output, int len)

DES-CBC buffer decryption.

Parameters:

ctx DES context
iv initialization vector (modified after use)
input buffer holding the ciphertext
output buffer holding the plaintext
len length of the data to be decrypted

Definition at line 444 of file des.c.

References des_crypt(), and des_context::dsk.

8.92.2.8 void des_cbc_encrypt (des_context * ctx, unsigned char iv[8], unsigned char * input, unsigned char * output, int len)

DES-CBC buffer encryption.

Parameters:

ctx DES context
iv initialization vector (modified after use)
input buffer holding the plaintext
output buffer holding the ciphertext
len length of the data to be encrypted

Definition at line 419 of file des.c.

References des_crypt(), and des_context::esk.

Referenced by main().

8.92.2.9 void des_decrypt (des_context * ctx, unsigned char input[8], unsigned char output[8])

DES block decryption (ECB mode).

Parameters:

ctx DES context
input ciphertext block
output plaintext block

Definition at line 409 of file des.c.

References des_crypt(), and des_context::dsk.

8.92.2.10 void des_encrypt (des_context * *ctx*, unsigned char *input*[8], unsigned char *output*[8])

DES block encryption (ECB mode).

Parameters:

ctx DES context
input plaintext block
output ciphertext block

Definition at line 399 of file des.c.

References des_crypt(), and des_context::esk.

8.92.2.11 int des_self_test (int *verbose*)

Definition at line 733 of file des.c.

Referenced by main().

8.92.2.12 void des_set_key (des_context * *ctx*, unsigned char *key*[8])

DES key schedule (56-bit).

Parameters:

ctx DES context to be initialized
key 8-byte secret key

Definition at line 357 of file des.c.

References des_main_ks(), des_context::dsk, and des_context::esk.

Referenced by main().

8.93 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/dhm.h File Reference

```
#include "bignum.h"
```

Data Structures

- struct [dhm_context](#)

Defines

- #define [ERR_DHM_BAD_INPUT_DATA](#) 0x0380
- #define [ERR_DHM_READ_PARAMS_FAILED](#) 0x0390
- #define [ERR_DHM_MAKE_PARAMS_FAILED](#) 0x03A0
- #define [ERR_DHM_READ_PUBLIC_FAILED](#) 0x03B0
- #define [ERR_DHM_MAKE_PUBLIC_FAILED](#) 0x03C0
- #define [ERR_DHM_CALC_SECRET_FAILED](#) 0x03D0

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](#)(*rng_f)(void *), void *rng_d, unsigned char *output, [int](#) *olen)
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, unsigned char *output, [int](#) olen, [int](#)(*rng_f)(void *), void *rng_d)
Create 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.

8.93.1 Detailed Description

Definition in file [dhm.h](#).

8.93.2 Define Documentation

8.93.2.1 `#define ERR_DHM_BAD_INPUT_DATA 0x0380`

Definition at line 13 of file dhm.h.

Referenced by `dhm_calc_secret()`, `dhm_make_public()`, `dhm_read_bignum()`, `dhm_read_params()`, and `dhm_read_public()`.

8.93.2.2 `#define ERR_DHM_CALC_SECRET_FAILED 0x03D0`

Definition at line 18 of file dhm.h.

Referenced by `dhm_calc_secret()`.

8.93.2.3 `#define ERR_DHM_MAKE_PARAMS_FAILED 0x03A0`

Definition at line 15 of file dhm.h.

Referenced by `dhm_make_params()`.

8.93.2.4 `#define ERR_DHM_MAKE_PUBLIC_FAILED 0x03C0`

Definition at line 17 of file dhm.h.

Referenced by `dhm_make_public()`.

8.93.2.5 `#define ERR_DHM_READ_PARAMS_FAILED 0x0390`

Definition at line 14 of file dhm.h.

Referenced by `dhm_read_bignum()`.

8.93.2.6 `#define ERR_DHM_READ_PUBLIC_FAILED 0x03B0`

Definition at line 16 of file dhm.h.

Referenced by `dhm_read_public()`.

8.93.3 Function Documentation

8.93.3.1 `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 ERR_DHM_MAKE_PUBLIC_FAILED

Definition at line 203 of file dhm.c.

References CHK, ERR_DHM_BAD_INPUT_DATA, ERR_DHM_CALC_SECRET_FAILED, dhm_context::GY, dhm_context::K, mpi_exp_mod(), mpi_msb(), mpi_write_binary(), dhm_context::P, dhm_context::RP, and dhm_context::X.

Referenced by main(), ssl_parse_client_key_exchange(), and ssl_write_client_key_exchange().

8.93.3.2 void dhm_free (dhm_context * ctx)

Definition at line 229 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().

8.93.3.3 int dhm_make_params (dhm_context * ctx, int(*) (void *) rng_f, void * rng_d, unsigned char * output, int * olen)

Setup and write the ServerKeyExchange parameters.

Parameters:

ctx DHM context
rng_f points to the RNG function
rng_d points to the RNG data
output destination buffer
olen number of chars written

Note:

This function assumes that ctx->P and ctx->G have already been properly set (for example using mpi_read_string).

Returns:

0 if successful, or an MPI error code

Definition at line 93 of file dhm.c.

References CHK, DHM_MPI_EXPORT, ERR_DHM_MAKE_PARAMS_FAILED, dhm_context::G, dhm_context::GX, dhm_context::len, mpi_cmp_mpi(), mpi_exp_mod(), mpi_grow(), mpi_msb(), mpi_shift_r(), dhm_context::P, mpi::p, dhm_context::RP, and dhm_context::X.

Referenced by main(), and ssl_write_server_key_exchange().

8.93.3.4 int dhm_make_public (dhm_context * ctx, unsigned char * output, int olen, int(*) (void *) rng_f, void * rng_d)

Create private value X and export G^X .

Parameters:

ctx DHM context
output destination buffer
olen must be == *ctx*->P.len
rng_f points to the RNG function
rng_d points to the RNG data

Returns:

0 if successful, or ERR_DHM_MAKE_PUBLIC_FAILED

Definition at line 164 of file dhm.c.

References CHK, ERR_DHM_BAD_INPUT_DATA, ERR_DHM_MAKE_PUBLIC_FAILED, dhm_context::G, dhm_context::GX, dhm_context::len, mpi_cmp_mpi(), mpi_exp_mod(), mpi_grow(), mpi_shift_r(), mpi_write_binary(), dhm_context::P, mpi::p, dhm_context::RP, and dhm_context::X.

Referenced by main(), and ssl_write_client_key_exchange().

8.93.3.5 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 ERR_DHM_READ_PARAMS_FAILED

Definition at line 63 of file dhm.c.

References dhm_read_bignum(), ERR_DHM_BAD_INPUT_DATA, dhm_context::G, dhm_context::GY, dhm_context::len, mpi_msb(), and dhm_context::P.

Referenced by main(), and ssl_parse_server_key_exchange().

8.93.3.6 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 ERR_DHM_READ_PUBLIC_FAILED

Definition at line 147 of file dhm.c.

References ERR_DHM_BAD_INPUT_DATA, ERR_DHM_READ_PUBLIC_FAILED, dhm_context::GY, dhm_context::len, and mpi_read_binary().

Referenced by main(), and ssl_parse_client_key_exchange().

8.93.3.7 int dhm_self_test (int *verbose*)

Checkup routine.

Returns:

 0 if successful, or 1 if the test failed

Definition at line 243 of file dhm.c.

8.94 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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 *rng_d)
HAVEGE rand function.

8.94.1 Detailed Description

Definition in file [havege.h](#).

8.94.2 Define Documentation

8.94.2.1 #define COLLECT_SIZE 1024

Definition at line 11 of file [havege.h](#).

Referenced by [havege_fill\(\)](#), and [havege_rand\(\)](#).

8.94.3 Function Documentation

8.94.3.1 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 [main\(\)](#).

8.94.3.2 int havege_rand (void * *rng_d*)

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 main().

8.95 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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)
MD2 final digest.
- void [md2](#) (unsigned char *input, [int](#) ilen, unsigned char *output)
Output = MD2(input buffer).
- [int](#) [md2_file](#) (char *path, unsigned char *output)
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)
MD2 HMAC final digest.
- void [md2_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char *output)
Output = HMAC-MD2(hmac key, input buffer).
- [int](#) [md2_self_test](#) ([int](#) verbose)
Checkup routine.

8.95.1 Detailed Description

Definition in file [md2.h](#).

8.95.2 Function Documentation

8.95.2.1 void md2 (unsigned char * *input*, int *ilen*, unsigned char * *output*)

Output = MD2(input buffer).

Parameters:

input buffer holding the data
ilen length of the input data
output MD2 checksum result

Definition at line 151 of file md2.c.

References md2_finish(), md2_starts(), and md2_update().

Referenced by x509_hash().

8.95.2.2 int md2_file (char * *path*, unsigned char * *output*)

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

Definition at line 166 of file md2.c.

References f, md2_finish(), md2_starts(), and md2_update().

8.95.2.3 void md2_finish (md2_context * *ctx*, unsigned char * *output*)

MD2 final digest.

Parameters:

ctx MD2 context
output MD2 checksum result

Definition at line 130 of file md2.c.

References md2_context::buffer, md2_context::cksum, md2_context::left, md2_process(), and md2_context::state.

Referenced by md2(), md2_file(), and md2_hmac_finish().

8.95.2.4 void md2_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char * *output*)

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

Definition at line 246 of file md2.c.

References md2_hmac_finish(), md2_hmac_starts(), and md2_hmac_update().

8.95.2.5 void md2_hmac_finish (md2_context * *ctx*, unsigned char * *output*)

MD2 HMAC final digest.

Parameters:

ctx HMAC context

output MD2 HMAC checksum result

Definition at line 230 of file md2.c.

References md2_finish(), md2_starts(), md2_update(), and md2_context::opad.

Referenced by md2_hmac().

8.95.2.6 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

Definition at line 198 of file md2.c.

References md2_context::ipad, md2_starts(), md2_update(), and md2_context::opad.

Referenced by md2_hmac().

8.95.2.7 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

Definition at line 221 of file md2.c.

References md2_update().

Referenced by md2_hmac().

8.95.2.8 int md2_self_test (int *verbose*)

Checksum routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 327 of file md2.c.

Referenced by main().

8.95.2.9 void md2_starts (md2_context * *ctx*)

MD2 context setup.

Parameters:

ctx context to be initialized

Definition at line 69 of file md2.c.

Referenced by md2(), md2_file(), md2_hmac_finish(), and md2_hmac_starts().

8.95.2.10 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

Definition at line 102 of file md2.c.

References md2_context::buffer, md2_context::left, and md2_process().

Referenced by md2(), md2_file(), md2_hmac_finish(), md2_hmac_starts(), and md2_hmac_update().

8.96 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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)
MD4 final digest.
- void [md4](#) (unsigned char *input, [int](#) ilen, unsigned char *output)
Output = MD4(input buffer).
- [int](#) [md4_file](#) (char *path, unsigned char *output)
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)
MD4 HMAC final digest.
- void [md4_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char *output)
Output = HMAC-MD4(hmac key, input buffer).
- [int](#) [md4_self_test](#) ([int](#) verbose)
Checkup routine.

8.96.1 Detailed Description

Definition in file [md4.h](#).

8.96.2 Function Documentation

8.96.2.1 void md4 (unsigned char * *input*, int *ilen*, unsigned char * *output*)

Output = MD4(input buffer).

Parameters:

input buffer holding the data
ilen length of the input data
output MD4 checksum result

Definition at line 259 of file md4.c.

References md4_finish(), md4_starts(), and md4_update().

Referenced by main(), and x509_hash().

8.96.2.2 int md4_file (char * *path*, unsigned char * *output*)

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

Definition at line 274 of file md4.c.

References f, md4_finish(), md4_starts(), and md4_update().

8.96.2.3 void md4_finish (md4_context * *ctx*, unsigned char * *output*)

MD4 final digest.

Parameters:

ctx MD4 context
output MD4 checksum result

Definition at line 231 of file md4.c.

References md4_padding, md4_update(), PUT_UINT32_LE, md4_context::state, and md4_context::total.

Referenced by md4(), md4_file(), and md4_hmac_finish().

8.96.2.4 void md4_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char * *output*)

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

Definition at line 354 of file md4.c.

References md4_hmac_finish(), md4_hmac_starts(), and md4_hmac_update().

8.96.2.5 void md4_hmac_finish (md4_context * *ctx*, unsigned char * *output*)

MD4 HMAC final digest.

Parameters:

ctx HMAC context
output MD4 HMAC checksum result

Definition at line 338 of file md4.c.

References md4_finish(), md4_starts(), md4_update(), and md4_context::opad.

Referenced by md4_hmac().

8.96.2.6 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

Definition at line 306 of file md4.c.

References md4_context::ipad, md4_starts(), md4_update(), and md4_context::opad.

Referenced by md4_hmac().

8.96.2.7 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

Definition at line 329 of file md4.c.

References md4_update().

Referenced by md4_hmac().

8.96.2.8 int md4_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 435 of file md4.c.

Referenced by main().

8.96.2.9 void md4_starts (md4_context * *ctx*)

MD4 context setup.

Parameters:

ctx context to be initialized

Definition at line 62 of file md4.c.

References md4_context::state, and md4_context::total.

Referenced by md4(), md4_file(), md4_hmac_finish(), and md4_hmac_starts().

8.96.2.10 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

Definition at line 179 of file md4.c.

References md4_context::buffer, md4_process(), and md4_context::total.

Referenced by md4(), md4_file(), md4_finish(), md4_hmac_finish(), md4_hmac_starts(), and md4_hmac_update().

8.97 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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)
MD5 final digest.
- void [md5](#) (unsigned char *input, [int](#) ilen, unsigned char *output)
Output = MD5(input buffer).
- [int](#) [md5_file](#) (char *path, unsigned char *output)
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)
MD5 HMAC final digest.
- void [md5_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char *output)
Output = HMAC-MD5(hmac key, input buffer).
- [int](#) [md5_self_test](#) ([int](#) verbose)
Checkup routine.

8.97.1 Detailed Description

Definition in file [md5.h](#).

8.97.2 Function Documentation

8.97.2.1 void md5 (unsigned char * *input*, int *ilen*, unsigned char * *output*)

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 main(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_server_key_exchange(), ssl_write_server_key_exchange(), and x509_hash().

8.97.2.2 int md5_file (char * *path*, unsigned char * *output*)

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 293 of file md5.c.

References f, md5_finish(), md5_starts(), and md5_update().

Referenced by md5_wrapper().

8.97.2.3 void md5_finish (md5_context * *ctx*, unsigned char * *output*)

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_UINT32_LE, md5_context::state, and md5_context::total.

Referenced by md5(), md5_file(), md5_hmac_finish(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_server_key_exchange(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

8.97.2.4 void md5_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char * *output*)

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 373 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().

8.97.2.5 void md5_hmac_finish (md5_context * *ctx*, unsigned char * *output*)

MD5 HMAC final digest.

Parameters:

ctx HMAC context

output MD5 HMAC checksum result

Definition at line 357 of file md5.c.

References md5_finish(), md5_starts(), md5_update(), and md5_context::opad.

Referenced by md5_hmac().

8.97.2.6 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 325 of file md5.c.

References md5_context::ipad, md5_starts(), md5_update(), and md5_context::opad.

Referenced by md5_hmac().

8.97.2.7 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 348 of file md5.c.

References md5_update().

Referenced by md5_hmac().

8.97.2.8 int md5_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 454 of file md5.c.

Referenced by main().

8.97.2.9 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_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_client_hello(), ssl_parse_server_key_exchange(), ssl_write_client_hello(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

8.97.2.10 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_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_client_hello(), ssl_parse_server_key_exchange(), ssl_write_record(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

8.98 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/net.h File Reference

Defines

- `#define ERR_NET_UNKNOWN_HOST 0xFF10`
- `#define ERR_NET_SOCKET_FAILED 0xFF20`
- `#define ERR_NET_CONNECT_FAILED 0xFF20`
- `#define ERR_NET_BIND_FAILED 0xFF30`
- `#define ERR_NET_LISTEN_FAILED 0xFF40`
- `#define ERR_NET_ACCEPT_FAILED 0xFF50`
- `#define ERR_NET_RECV_FAILED 0xFF60`
- `#define ERR_NET_SEND_FAILED 0xFF70`
- `#define ERR_NET_CONN_RESET 0xFF80`
- `#define ERR_NET_WOULD_BLOCK 0xFF90`

Functions

- `int net_connect (int *fd, char *host, unsigned int port)`
Initiate a TCP connection with host:port.
- `int net_bind (int *fd, char *bind_ip, unsigned 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, unsigned char client_ip[4])`
Accept a connection from a remote client.
- `int net_set_nonblock (int fd)`
Set the socket non-blocking.
- `void net_usleep (unsigned long usec)`
Portable usleep helper.
- `int net_recv (int fd, unsigned char *buf, int *len)`
Read at most 'len' characters. len is updated to reflect the actual number of characters read.
- `int net_send (int fd, 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.

8.98.1 Detailed Description

Definition in file `net.h`.

8.98.2 Define Documentation

8.98.2.1 **#define ERR_NET_ACCEPT_FAILED 0xFF50**

Definition at line 16 of file net.h.

Referenced by net_accept().

8.98.2.2 **#define ERR_NET_BIND_FAILED 0xFF30**

Definition at line 14 of file net.h.

Referenced by net_bind().

8.98.2.3 **#define ERR_NET_CONN_RESET 0xFF80**

Definition at line 19 of file net.h.

Referenced by main(), net_recv(), and net_send().

8.98.2.4 **#define ERR_NET_CONNECT_FAILED 0xFF20**

Definition at line 13 of file net.h.

Referenced by net_connect().

8.98.2.5 **#define ERR_NET_LISTEN_FAILED 0xFF40**

Definition at line 15 of file net.h.

Referenced by net_bind().

8.98.2.6 **#define ERR_NET_RECV_FAILED 0xFF60**

Definition at line 17 of file net.h.

Referenced by net_recv().

8.98.2.7 **#define ERR_NET_SEND_FAILED 0xFF70**

Definition at line 18 of file net.h.

Referenced by net_send().

8.98.2.8 **#define ERR_NET_SOCKET_FAILED 0xFF20**

Definition at line 12 of file net.h.

Referenced by net_bind(), and net_connect().

8.98.2.9 `#define ERR_NET_UNKNOWN_HOST 0xFF10`

Definition at line 11 of file net.h.

Referenced by `net_connect()`.

8.98.2.10 `#define ERR_NET_WOULD_BLOCK 0xFF90`

Definition at line 20 of file net.h.

Referenced by `main()`, `net_accept()`, `net_recv()`, and `net_send()`.

8.98.3 Function Documentation

8.98.3.1 `int net_accept (int bind_fd, int * client_fd, unsigned char client_ip[4])`

Accept a connection from a remote client.

Returns:

0 if successful, `ERR_NET_ACCEPT_FAILED`, or `ERR_NET_WOULD_BLOCK` if `bind_fd` was set to non-blocking and `accept()` is blocking.

Definition at line 200 of file net.c.

References `ERR_NET_ACCEPT_FAILED`, `ERR_NET_WOULD_BLOCK`, `int`, and `net_is_blocking()`.

Referenced by `main()`.

8.98.3.2 `int net_bind (int * fd, char * bind_ip, unsigned 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: `ERR_NET_SOCKET_FAILED`, `ERR_NET_BIND_FAILED`, `ERR_NET_LISTEN_FAILED`

Definition at line 112 of file net.c.

References `ERR_NET_BIND_FAILED`, `ERR_NET_LISTEN_FAILED`, and `ERR_NET_SOCKET_FAILED`.

Referenced by `main()`.

8.98.3.3 `void net_close (int fd)`

Gracefully shutdown the connection.

Definition at line 337 of file net.c.

Referenced by `main()`.

8.98.3.4 int net_connect (int *fd, char *host, unsigned int port)

Initiate a TCP connection with host:port.

Returns:

0 if successful, or one of: ERR_NET_SOCKET_FAILED, ERR_NET_UNKNOWN_HOST, ERR_NET_CONNECT_FAILED

Definition at line 67 of file net.c.

References ERR_NET_CONNECT_FAILED, ERR_NET_SOCKET_FAILED, and ERR_NET_UNKNOWN_HOST.

Referenced by main().

8.98.3.5 int net_recv (int fd, unsigned char *buf, int *len)

Read at most 'len' characters. len is updated to reflect the actual number of characters read.

Returns:

0 if successful, ERR_NET_CONN_RESET if the connection was closed from the other side, or ERR_NET_WOULD_BLOCK if read() is blocking.

Definition at line 265 of file net.c.

References ERR_NET_CONN_RESET, ERR_NET_RECV_FAILED, ERR_NET_WOULD_BLOCK, and net_is_blocking().

Referenced by main(), and ssl_parse_client_hello().

8.98.3.6 int net_send (int fd, unsigned char *buf, int *len)

Write at most 'len' characters. len is updated to reflect the number of characters _not_ written.

Returns:

0 if successful, ERR_NET_CONN_RESET if the connection was closed from the other side, or ERR_NET_WOULD_BLOCK if write() is blocking.

Definition at line 302 of file net.c.

References ERR_NET_CONN_RESET, ERR_NET_SEND_FAILED, ERR_NET_WOULD_BLOCK, and net_is_blocking().

Referenced by main(), ssl_flush_output(), and ssl_write_record().

8.98.3.7 int net_set_nonblock (int fd)

Set the socket non-blocking.

Returns:

0 if successful, or 1 if the operation failed

Definition at line 241 of file net.c.

8.98.3.8 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 254 of file net.c.

8.99 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/rsa.h File Reference

```
#include "bignum.h"
```

Data Structures

- struct [rsa_context](#)

Defines

- #define [ERR_RSA_BAD_INPUT_DATA](#) 0x0300
- #define [ERR_RSA_INVALID_PADDING](#) 0x0310
- #define [ERR_RSA_KEY_GEN_FAILED](#) 0x0320
- #define [ERR_RSA_KEY_CHK_FAILED](#) 0x0330
- #define [ERR_RSA_KEY_RD_FAILED](#) 0x0340
- #define [ERR_RSA_KEY_WR_FAILED](#) 0x0350
- #define [ERR_RSA_PUBLIC_FAILED](#) 0x0360
- #define [ERR_RSA_PRIVATE_FAILED](#) 0x0370
- #define [ERR_RSA_VERIFY_FAILED](#) 0x0380
- #define [RSA_RAW](#) 0
- #define [RSA_MD2](#) 2
- #define [RSA_MD4](#) 3
- #define [RSA_MD5](#) 4
- #define [RSA_SHA1](#) 5
- #define [RSA_SIGN](#) 0x01
- #define [RSA_CRYPT](#) 0x02
- #define [ASN1_HASH_MDX](#)
- #define [ASN1_HASH_SHA1](#)
- #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](#)

Functions

- [int rsa_gen_key](#) ([rsa_context](#) *ctx, [int](#) nbits, [int](#) exponent, [int](#)(*rng_f)(void *), void *rng_d)
Generate an RSA keypair.
- [int rsa_read_public](#) ([rsa_context](#) *ctx, FILE *f)
Read the public key from a file.
- [int rsa_read_private](#) ([rsa_context](#) *ctx, FILE *f)

Read the private key from a file.

- `int rsa_write_public (rsa_context *ctx, FILE *f)`

Write the public key into a file.

- `int rsa_write_private (rsa_context *ctx, FILE *f)`

Write the private key into a file.

- `int rsa_public (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int olen)`

Perform an RSA public key operation.

- `int rsa_private (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int olen)`

Perform an RSA private key operation.

- `int rsa_check_pubkey (rsa_context *ctx)`

Return 0 if the public key is valid, or ERR_RSA_KEY_CHECK_FAILED.

- `int rsa_check_privkey (rsa_context *ctx)`

Return 0 if the private key is valid, or ERR_RSA_KEY_CHECK_FAILED.

- `int rsa_pkcs1_encrypt (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int olen)`

Add the PKCS#1 v1.5 padding and do a public RSA.

- `int rsa_pkcs1_decrypt (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int *olen)`

Do a private RSA, removes the PKCS#1 v1.5 padding.

- `int rsa_pkcs1_sign (rsa_context *ctx, int alg_id, unsigned char *hash, int hashlen, unsigned char *sig, int siglen)`

Perform a private RSA to sign a message digest.

- `int rsa_pkcs1_verify (rsa_context *ctx, int alg_id, unsigned char *hash, int hashlen, unsigned char *sig, int siglen)`

Perform 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.

8.99.1 Detailed Description

Definition in file [rsa.h](#).

8.99.2 Define Documentation

8.99.2.1 #define ASN1_HASH_MDX

Value:

```
"\x30\x20\x30\x0C\x06\x08\x2A\x86\x48" \
"\x86\xF7\x0D\x02\x00\x05\x00\x04\x10"
```

Definition at line 44 of file rsa.h.

Referenced by rsa_pkcs1_sign(), and rsa_pkcs1_verify().

8.99.2.2 #define ASN1_HASH_SHA1

Value:

```
"\x30\x21\x30\x09\x06\x05\x2B\x0E\x03" \
"\x02\x1A\x05\x00\x04\x14"
```

Definition at line 48 of file rsa.h.

Referenced by rsa_pkcs1_sign(), and rsa_pkcs1_verify().

8.99.2.3 #define ERR_RSA_BAD_INPUT_DATA 0x0300

Definition at line 13 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().

8.99.2.4 #define ERR_RSA_INVALID_PADDING 0x0310

Definition at line 14 of file rsa.h.

Referenced by rsa_pkcs1_decrypt(), and rsa_pkcs1_verify().

8.99.2.5 #define ERR_RSA_KEY_CHK_FAILED 0x0330

Definition at line 16 of file rsa.h.

Referenced by rsa_check_privkey(), and rsa_check_pubkey().

8.99.2.6 #define ERR_RSA_KEY_GEN_FAILED 0x0320

Definition at line 15 of file rsa.h.

Referenced by rsa_gen_key().

8.99.2.7 #define ERR_RSA_KEY_RD_FAILED 0x0340

Definition at line 17 of file rsa.h.

Referenced by rsa_read_private(), and rsa_read_public().

8.99.2.8 #define ERR_RSA_KEY_WR_FAILED 0x0350

Definition at line 18 of file rsa.h.

Referenced by rsa_write_private(), and rsa_write_public().

8.99.2.9 #define ERR_RSA_PRIVATE_FAILED 0x0370

Definition at line 20 of file rsa.h.

Referenced by rsa_private().

8.99.2.10 #define ERR_RSA_PUBLIC_FAILED 0x0360

Definition at line 19 of file rsa.h.

Referenced by rsa_public().

8.99.2.11 #define ERR_RSA_VERIFY_FAILED 0x0380

Definition at line 21 of file rsa.h.

Referenced by rsa_pkcs1_verify().

8.99.2.12 #define KEY_LEN 128

Definition at line 249 of file rsa.h.

Referenced by main().

8.99.2.13 #define RSA_CRYPT 0x02

Definition at line 33 of file rsa.h.

Referenced by rsa_pkcs1_decrypt(), and rsa_pkcs1_encrypt().

8.99.2.14 #define RSA_D

Value:

```
"24BF6185468786FDD303083D25E64EFC" \
    "66CA472BC44D253102F8B4A9D3BFA750" \
    "91386C0077937FE33FA3252D28855837" \
    "AE1B484A8A9A45F7EE8C0C634F99E8CD" \
    "DF79C5CE07EE72C7F123142198164234" \
    "CABB724CF78B8173B9F880FC86322407" \
    "AF1FEDFDDE2BEB674CA15F3E81A1521E" \
    "071513A1E85B5DFA031F21ECAE91A34D"
```

Definition at line 262 of file rsa.h.

Referenced by main().

8.99.2.15 #define RSA_DP**Value:**

```
"C1ACF567564274FB07A0BBAD5D26E298" \
    "3C94D22288ACD763FD8E5600ED4A702D" \
    "F84198A5F06C2E72236AE490C93F07F8" \
    "3CC559CD27BC2D1CA488811730BB5725"
```

Definition at line 281 of file rsa.h.

Referenced by main().

8.99.2.16 #define RSA_DQ**Value:**

```
"4959CBF6F8FEF750AEE6977C155579C7" \
    "D8AAEA56749EA28623272E4F7D0592AF" \
    "7C1F1313CAC9471B5C523BFE592F517B" \
    "407A1BD76C164B93DA2D32A383E58357"
```

Definition at line 286 of file rsa.h.

Referenced by main().

8.99.2.17 #define RSA_E "10001"

Definition at line 260 of file rsa.h.

Referenced by main().

8.99.2.18 #define RSA_MD2 2

Definition at line 27 of file rsa.h.

Referenced by rsa_pkcs1_sign(), rsa_pkcs1_verify(), x509_cert_info(), and x509_hash().

8.99.2.19 #define RSA_MD4 3

Definition at line 28 of file rsa.h.

Referenced by rsa_pkcs1_sign(), rsa_pkcs1_verify(), x509_cert_info(), and x509_hash().

8.99.2.20 #define RSA_MD5 4

Definition at line 29 of file rsa.h.

Referenced by rsa_pkcs1_sign(), rsa_pkcs1_verify(), x509_cert_info(), and x509_hash().

8.99.2.21 #define RSA_N**Value:**

```
"9292758453063D803DD603D5E777D788" \
    "8ED1D5BF35786190FA2F23EBC0848AEA" \
    "DDA92CA6C3D80B32C4D109BE0F36D6AE" \
    "7130B9CED7ACDF54CFC7555AC14EEBAB" \
    "93A89813FBF3C4F8066D2D800F7C38A8" \
    "1AE31942917403FF4946B0A83D3D3E05" \
    "EE57C6F5F5606FB5D4BC6CD34EE0801A" \
    "5E94BB77B07507233A0BC7BAC8F90F79"
```

Definition at line 251 of file rsa.h.

Referenced by main().

8.99.2.22 #define RSA_P

Value:

```
"C36D0EB7FCD285223CFB5AABA5BDA3D8" \
    "2C01CAD19EA484A87EA4377637E75500" \
    "FCB2005C5C7DD6EC4AC023CDA285D796" \
    "C3D9E75E1EFC42488BB4F1D13AC30A57"
```

Definition at line 271 of file rsa.h.

Referenced by main().

8.99.2.23 #define RSA_Q

Value:

```
"C000DF51A7C77AE8D7C7370C1FF55B69" \
    "E211C2B9E5DB1ED0BF61D0D9899620F4" \
    "910E4168387E3C30AA1E00C339A79508" \
    "8452DD96A9A5EA5D9DCA68DA636032AF"
```

Definition at line 276 of file rsa.h.

Referenced by main().

8.99.2.24 #define RSA_QP

Value:

```
"9AE7FBC99546432DF71896FC239EADAE" \
    "F38D18D2B2F0E2DD275AA977E2BF4411" \
    "F5A3B2A5D33605AEBBCCBA7FEB9F2D2F" \
    "A74206CEC169D74BF5A8C50D6F48EA08"
```

Definition at line 291 of file rsa.h.

Referenced by main().

8.99.2.25 #define RSA_RAW 0

Definition at line 26 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().

8.99.2.26 #define RSA_SHA1 5

Definition at line 30 of file rsa.h.

Referenced by main(), rsa_pkcs1_sign(), rsa_pkcs1_verify(), x509_cert_info(), and x509_hash().

8.99.2.27 #define RSA_SIGN 0x01

Definition at line 32 of file rsa.h.

Referenced by rsa_pkcs1_sign(), and rsa_pkcs1_verify().

8.99.3 Function Documentation**8.99.3.1 int rsa_check_privkey (rsa_context * ctx)**

Return 0 if the private key is valid, or ERR_RSA_KEY_CHECK_FAILED.

Definition at line 228 of file dsa.c.

References CHK, rsa_context::E, ERR_RSA_KEY_CHK_FAILED, mpi_cmp_int(), mpi_cmp_mpi(), mpi_free(), mpi_gcd(), mpi_init(), mpi_mul_mpi(), mpi_sub_int(), rsa_context::N, rsa_context::P, and rsa_context::Q.

Referenced by x509_parse_key().

8.99.3.2 int rsa_check_pubkey (rsa_context * ctx)

Return 0 if the public key is valid, or ERR_RSA_KEY_CHECK_FAILED.

Definition at line 208 of file dsa.c.

References rsa_context::E, ERR_RSA_KEY_CHK_FAILED, mpi_msb(), rsa_context::N, and mpi::p.

Referenced by x509_add_certs().

8.99.3.3 void rsa_free (rsa_context * ctx)

Free the components of an RSA key.

Definition at line 471 of file dsa.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_read_private(), rsa_read_public(), x509_free_cert(), and x509_parse_key().

8.99.3.4 int rsa_gen_key (rsa_context * ctx, int nbits, int exponent, int(*) (void *) rng_f, void * rng_d)

Generate an RSA keypair.

Parameters:

ctx RSA context to be initialized

nbits size of the public key in bits
exponent public exponent (e.g., 65537)
rng_f points to the RNG function
rng_d points to the RNG data

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 41 of file dsa.c.

References CHK, rsa_context::D, rsa_context::DP, rsa_context::DQ, rsa_context::E, ERR_RSA_BAD_INPUT_DATA, ERR_RSA_KEY_GEN_FAILED, rsa_context::len, 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::Q, rsa_context::QP, and rsa_free().

Referenced by main().

8.99.3.5 `int rsa_pkcs1_decrypt(rsa_context * ctx, unsigned char * input, int ilen, unsigned char * output, int * olen)`

Do a private RSA, removes the PKCS#1 v1.5 padding.

Parameters:

ctx RSA context
input buffer holding the encrypted data
ilen must be the same as the modulus size
output buffer that will hold the plaintext
olen size of output buffer, will be updated to contain the length of the plaintext

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 287 of file dsa.c.

References ERR_RSA_BAD_INPUT_DATA, ERR_RSA_INVALID_PADDING, int, rsa_context::len, RSA_CRYPT, and rsa_private().

Referenced by ssl_parse_client_key_exchange().

8.99.3.6 `int rsa_pkcs1_encrypt(rsa_context * ctx, unsigned char * input, int ilen, unsigned char * output, int olen)`

Add the PKCS#1 v1.5 padding and do a public RSA.

Parameters:

ctx RSA context
input buffer holding the data to be encrypted
ilen length of the plaintext; cannot be longer than the modulus, minus 3+8 for padding

output buffer that will hold the ciphertext

olen must be the same as the modulus size (for example, 128 if RSA-1024 is used)

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 257 of file dsa.c.

References ERR_RSA_BAD_INPUT_DATA, rsa_context::len, RSA_CRYPT, and rsa_public().

Referenced by ssl_write_client_key_exchange().

8.99.3.7 int rsa_pkcs1_sign (rsa_context * ctx, int alg_id, unsigned char * hash, int hashlen, unsigned char * sig, int siglen)

Perform a private RSA to sign a message digest.

Parameters:

ctx RSA context

alg_id RSA_RAW, RSA_MD2/4/5 or RSA_SHA1

hash buffer holding the message digest

hashlen message digest length

sig buffer that will hold the ciphertext

siglen must be the same as the modulus size (for example, 128 if RSA-1024 is used)

Returns:

0 if the signing operation was successful, or an ERR_RSA_XXX error code

Definition at line 325 of file dsa.c.

References ASN1_HASH_MDX, ASN1_HASH_SHA1, ERR_RSA_BAD_INPUT_DATA, rsa_context::len, RSA_MD2, RSA_MD4, RSA_MD5, rsa_private(), RSA_RAW, RSA_SHA1, and RSA_SIGN.

Referenced by main(), ssl_write_certificate_verify(), and ssl_write_server_key_exchange().

8.99.3.8 int rsa_pkcs1_verify (rsa_context * ctx, int alg_id, unsigned char * hash, int hashlen, unsigned char * sig, int siglen)

Perform a public RSA and check the message digest.

Parameters:

ctx points to an RSA public key

alg_id RSA_RAW, RSA_MD2/4/5 or RSA_SHA1

hash buffer holding the message digest

hashlen message digest length

sig buffer holding the ciphertext

siglen must be the same as the modulus size

Returns:

0 if the verify operation was successful, or an ERR_RSA_XXX error code

Definition at line 401 of file dsa.c.

References ASN1_HASH_MDX, ASN1_HASH_SHA1, ERR_RSA_BAD_INPUT_DATA, ERR_RSA_INVALID_PADDING, ERR_RSA_VERIFY_FAILED, int, rsa_context::len, RSA_MD2, RSA_MD4, RSA_MD5, rsa_public(), RSA_RAW, RSA_SHA1, and RSA_SIGN.

Referenced by main(), ssl_parse_certificate_verify(), ssl_parse_server_key_exchange(), and x509_verify_cert().

8.99.3.9 int rsa_private (rsa_context * ctx, unsigned char * input, int ilen, unsigned char * output, int olen)

Perform an RSA private key operation.

Returns:

0 if successful, or an ERR_RSA_XXX error code

Note:

This function does not take care of message padding: both ilen and olen must be equal to the modulus size (ctx->len). Also, be sure to set input[0] = 0.

Definition at line 147 of file dsa.c.

References CHK, rsa_context::D, rsa_context::DP, rsa_context::DQ, ERR_RSA_BAD_INPUT_DATA, ERR_RSA_PRIVATE_FAILED, rsa_context::len, mpi_add_mpi(), 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, and rsa_context::RQ.

Referenced by main(), rsa_pkcs1_decrypt(), and rsa_pkcs1_sign().

8.99.3.10 int rsa_public (rsa_context * ctx, unsigned char * input, int ilen, unsigned char * output, int olen)

Perform an RSA public key operation.

Returns:

0 if successful, or an ERR_RSA_XXX error code

Note:

This function does not take care of message padding: both ilen and olen must be equal to the modulus size (ctx->len). Also, be sure to set input[0] = 0.

Definition at line 111 of file dsa.c.

References CHK, rsa_context::E, ERR_RSA_BAD_INPUT_DATA, ERR_RSA_PUBLIC_FAILED, rsa_context::len, mpi_cmp_mpi(), mpi_exp_mod(), mpi_free(), mpi_init(), mpi_read_binary(), mpi_write_binary(), rsa_context::N, and rsa_context::RN.

Referenced by main(), rsa_pkcs1_encrypt(), and rsa_pkcs1_verify().

8.99.3.11 `int rsa_read_private (rsa_context * ctx, FILE * f)`

Read the private key from a file.

Parameters:

ctx RSA context to be initialized

f Handle of the source file

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 136 of file rsa.c.

References CHK, rsa_context::D, rsa_context::DP, rsa_context::DQ, rsa_context::E, ERR_RSA_KEY_RD_FAILED, rsa_context::len, mpi_msb(), mpi_read_file(), rsa_context::N, rsa_context::P, rsa_context::Q, rsa_context::QP, and rsa_free().

Referenced by main().

8.99.3.12 `int rsa_read_public (rsa_context * ctx, FILE * f)`

Read the public key from a file.

Parameters:

ctx RSA context to be initialized

f Handle of the source file

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 111 of file rsa.c.

References CHK, rsa_context::E, ERR_RSA_KEY_RD_FAILED, rsa_context::len, mpi_msb(), mpi_read_file(), rsa_context::N, and rsa_free().

Referenced by main().

8.99.3.13 `int rsa_self_test (int verbose)`

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 585 of file dsa.c.

Referenced by main().

8.99.3.14 int rsa_write_private (rsa_context * *ctx*, FILE * *f*)

Write the private key into a file.

Parameters:

ctx RSA context holding the key
f Handle of the destination file

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 185 of file rsa.c.

References CHK, rsa_context::D, rsa_context::DP, rsa_context::DQ, rsa_context::E, ERR_RSA_KEY_WR_FAILED, mpi_write_file(), rsa_context::N, rsa_context::P, rsa_context::Q, and rsa_context::QP.

Referenced by main().

8.99.3.15 int rsa_write_public (rsa_context * *ctx*, FILE * *f*)

Write the public key into a file.

Parameters:

ctx RSA context holding the key
f Handle of the destination file

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 167 of file rsa.c.

References CHK, rsa_context::E, ERR_RSA_KEY_WR_FAILED, mpi_write_file(), and rsa_context::N.

Referenced by main().

8.100 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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)
SHA-1 final digest.
- void [sha1](#) (unsigned char *input, [int](#) ilen, unsigned char *output)
Output = SHA-1(input buffer).
- [int](#) [sha1_file](#) (char *path, unsigned char *output)
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)
SHA-1 HMAC final digest.
- void [sha1_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char *output)
Output = HMAC-SHA-1(hmac key, input buffer).
- [int](#) [sha1_self_test](#) ([int](#) verbose)
Checkup routine.

8.100.1 Detailed Description

Definition in file [sha1.h](#).

8.100.2 Function Documentation

8.100.2.1 void sha1 (unsigned char * *input*, int *ilen*, unsigned char * *output*)

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(), ssl_calc_verify(), ssl_derive_keys(), ssl_get_session(), ssl_mac_sha1(), ssl_parse_server_key_exchange(), ssl_set_session(), ssl_write_server_key_exchange(), and x509_hash().

8.100.2.2 int sha1_file (char * *path*, unsigned char * *output*)

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 328 of file sha1.c.

References f, sha1_finish(), sha1_starts(), and sha1_update().

Referenced by main(), and sha1_wrapper().

8.100.2.3 void sha1_finish (sha1_context * *ctx*, unsigned char * *output*)

SHA-1 final digest.

Parameters:

ctx SHA-1 context
output SHA-1 checksum result

Definition at line 284 of file sha1.c.

References PUT_UINT32_BE, sha1_padding, sha1_update(), sha1_context::state, and sha1_context::total.

Referenced by sha1(), sha1_file(), sha1_hmac_finish(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_sha1(), ssl_parse_server_key_exchange(), and ssl_write_server_key_exchange().

8.100.2.4 void sha1_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char * *output*)

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 408 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().

8.100.2.5 void sha1_hmac_finish (sha1_context * *ctx*, unsigned char * *output*)

SHA-1 HMAC final digest.

Parameters:

ctx HMAC context

output SHA-1 HMAC checksum result

Definition at line 392 of file sha1.c.

References sha1_context::opad, sha1_finish(), sha1_starts(), and sha1_update().

Referenced by sha1_hmac().

8.100.2.6 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 360 of file sha1.c.

References sha1_context::ipad, sha1_context::opad, sha1_starts(), and sha1_update().

Referenced by sha1_hmac().

8.100.2.7 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 383 of file sha1.c.

References sha1_update().

Referenced by sha1_hmac().

8.100.2.8 int sha1_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 489 of file sha1.c.

Referenced by main().

8.100.2.9 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(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_sha1(), ssl_parse_client_hello(), ssl_parse_server_key_exchange(), ssl_write_client_hello(), and ssl_write_server_key_exchange().

8.100.2.10 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(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_sha1(), ssl_parse_client_hello(), ssl_parse_server_key_exchange(), ssl_write_record(), and ssl_write_server_key_exchange().

8.101 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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)
SHA-256 final digest.
- void [sha2](#) (unsigned char *input, [int](#) ilen, unsigned char *output, [int](#) is224)
Output = SHA-256(input buffer).
- [int](#) [sha2_file](#) (char *path, unsigned char *output, [int](#) is224)
Output = SHA-256(file contents).
- void [sha2_hmac_starts](#) ([sha2_context](#) *ctx, [int](#) is224, unsigned char *key, [int](#) keylen)
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)
SHA-256 HMAC final digest.
- void [sha2_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char *output, [int](#) is224)
Output = HMAC-SHA-256(hmac key, input buffer).
- [int](#) [sha2_self_test](#) ([int](#) verbose)
Checkup routine.

8.101.1 Detailed Description

Definition in file [sha2.h](#).

8.101.2 Function Documentation

8.101.2.1 void sha2 (unsigned char * *input*, int *ilen*, unsigned char * *output*, 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().

8.101.2.2 int sha2_file (char * *path*, unsigned char * *output*, 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 f, sha2_finish(), sha2_starts(), and sha2_update().

Referenced by sha2_wrapper().

8.101.2.3 void sha2_finish (sha2_context * *ctx*, unsigned char * *output*)

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_UINT32_BE, sha2_padding, sha2_update(), sha2_context::state, and sha2_context::total.

Referenced by main(), sha2(), sha2_file(), and sha2_hmac_finish().

8.101.2.4 `void sha2_hmac (unsigned char * key, int keylen, unsigned char * input, int ilen, unsigned char * output, 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 413 of file sha2.c.

References sha2_hmac_finish(), sha2_hmac_starts(), and sha2_hmac_update().

8.101.2.5 `void sha2_hmac_finish (sha2_context * ctx, unsigned char * output)`

SHA-256 HMAC final digest.

Parameters:

ctx HMAC context
output SHA-224/256 HMAC checksum result

Definition at line 393 of file sha2.c.

References sha2_context::is224, sha2_context::opad, sha2_finish(), sha2_starts(), and sha2_update().

Referenced by main(), and sha2_hmac().

8.101.2.6 `void sha2_hmac_starts (sha2_context * ctx, int is224, unsigned char * key, int keylen)`

SHA-256 HMAC context setup.

Parameters:

ctx HMAC context to be initialized
is224 0 = use SHA256, 1 = use SHA224
key HMAC secret key
keylen length of the HMAC key

Definition at line 361 of file sha2.c.

References sha2_context::ipad, sha2_context::opad, sha2_starts(), and sha2_update().

Referenced by main(), and sha2_hmac().

8.101.2.7 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 384 of file sha2.c.

References sha2_update().

Referenced by main(), and sha2_hmac().

8.101.2.8 int sha2_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 524 of file sha2.c.

Referenced by main().

8.101.2.9 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 main(), sha2(), sha2_file(), sha2_hmac_finish(), and sha2_hmac_starts().

8.101.2.10 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 main(), sha2(), sha2_file(), sha2_finish(), sha2_hmac_finish(), sha2_hmac_starts(), and sha2_hmac_update().

8.102 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/sha4.h File Reference

Data Structures

- struct [sha4_context](#)
SHA-512 context structure.

Defines

- #define [UL64](#)(x) x##ULL
- #define [uint64](#) unsigned 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)
SHA-512 final digest.
- void [sha4](#) (unsigned char *input, [int](#) ilen, unsigned char *output, [int](#) is384)
Output = SHA-512(input buffer).
- [int](#) [sha4_file](#) (char *path, unsigned char *output, [int](#) is384)
Output = SHA-512(file contents).
- void [sha4_hmac_starts](#) ([sha4_context](#) *ctx, [int](#) is384, unsigned char *key, [int](#) keylen)
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)
SHA-512 HMAC final digest.
- void [sha4_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char *output, [int](#) is384)
Output = HMAC-SHA-512(hmac key, input buffer).
- [int](#) [sha4_self_test](#) ([int](#) verbose)
Checkup routine.

8.102.1 Detailed Description

Definition in file [sha4.h](#).

8.102.2 Define Documentation

8.102.2.1 #define uint64 unsigned long long

Definition at line 16 of file sha4.h.

Referenced by sha4_finish(), sha4_process(), and sha4_update().

8.102.2.2 #define UL64(x) x##ULL

Definition at line 15 of file sha4.h.

Referenced by sha4_starts().

8.102.3 Function Documentation

8.102.3.1 void sha4 (unsigned char * *input*, int *ilen*, unsigned char * *output*, 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().

8.102.3.2 int sha4_file (char * *path*, unsigned char * *output*, 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 f, sha4_finish(), sha4_starts(), and sha4_update().

8.102.3.3 void sha4_finish (sha4_context * ctx, unsigned char * output)

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, sha4_context::is384, PUT_UINT64_BE, sha4_padding, sha4_update(), sha4_context::state, sha4_context::total, and uint64.

Referenced by sha4(), sha4_file(), and sha4_hmac_finish().

8.102.3.4 void sha4_hmac (unsigned char * key, int keylen, unsigned char * input, int ilen, unsigned char * output, 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 411 of file sha4.c.

References sha4_hmac_finish(), sha4_hmac_starts(), and sha4_hmac_update().

8.102.3.5 void sha4_hmac_finish (sha4_context * ctx, unsigned char * output)

SHA-512 HMAC final digest.

Parameters:

ctx HMAC context

output SHA-384/512 HMAC checksum result

Definition at line 391 of file sha4.c.

References sha4_context::is384, sha4_context::opad, sha4_finish(), sha4_starts(), and sha4_update().

Referenced by sha4_hmac().

8.102.3.6 void sha4_hmac_starts (sha4_context * ctx, int is384, unsigned char * key, int keylen)

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_starts(), and sha4_update().

Referenced by sha4_hmac().

8.102.3.7 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 382 of file sha4.c.

References sha4_update().

Referenced by sha4_hmac().

8.102.3.8 int sha4_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 541 of file sha4.c.

8.102.3.9 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(), and sha4_hmac_starts().

8.102.3.10 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, sha4_process(), sha4_context::total, and uint64.

Referenced by sha4(), sha4_file(), sha4_finish(), sha4_hmac_finish(), sha4_hmac_starts(), and sha4_hmac_update().

8.103 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/ssl.h File Reference

```
#include "x509.h"
#include "rsa.h"
#include "dhm.h"
#include "md5.h"
#include "sha1.h"
```

Data Structures

- struct [ssl_context](#)

Defines

- #define [ERR_SSL_FEATURE_UNAVAILABLE](#) 0x1000
- #define [ERR_SSL_INVALID_MAC](#) 0x1800
- #define [ERR_SSL_INVALID_RECORD](#) 0x2000
- #define [ERR_SSL_INVALID_MODULUS_SIZE](#) 0x2800
- #define [ERR_SSL_UNKNOWN_CIPHER](#) 0x3000
- #define [ERR_SSL_NO_CIPHER_CHOSEN](#) 0x3800
- #define [ERR_SSL_NO_SESSION_FOUND](#) 0x4000
- #define [ERR_SSL_NO_CLIENT_CERTIFICATE](#) 0x4800
- #define [ERR_SSL_CERTIFICATE_TOO_LARGE](#) 0x5000
- #define [ERR_SSL_CERTIFICATE_REQUIRED](#) 0x5800
- #define [ERR_SSL_PRIVATE_KEY_REQUIRED](#) 0x6000
- #define [ERR_SSL_CA_CHAIN_REQUIRED](#) 0x6800
- #define [ERR_SSL_UNEXPECTED_MESSAGE](#) 0x7000
- #define [ERR_SSL_FATAL_ALERT_MESSAGE](#) 0x7800
- #define [ERR_SSL_PEER_VERIFY_FAILED](#) 0x8000
- #define [ERR_SSL_PEER_CLOSE_NOTIFY](#) 0x8800
- #define [ERR_SSL_BAD_HS_CLIENT_HELLO](#) 0x9000
- #define [ERR_SSL_BAD_HS_SERVER_HELLO](#) 0x9800
- #define [ERR_SSL_BAD_HS_CERTIFICATE](#) 0xA000
- #define [ERR_SSL_BAD_HS_CERTIFICATE_REQUEST](#) 0xA800
- #define [ERR_SSL_BAD_HS_SERVER_KEY_EXCHANGE](#) 0xB000
- #define [ERR_SSL_BAD_HS_SERVER_HELLO_DONE](#) 0xB800
- #define [ERR_SSL_BAD_HS_CLIENT_KEY_EXCHANGE](#) 0xC000
- #define [ERR_SSL_BAD_HS_CERTIFICATE_VERIFY](#) 0xC800
- #define [ERR_SSL_BAD_HS_CHANGE_CIPHER_SPEC](#) 0xD000
- #define [ERR_SSL_BAD_HS_FINISHED](#) 0xD800
- #define [SSLV3_MAJOR_VERSION](#) 3
- #define [SSLV3_MINOR_VERSION](#) 0
- #define [TLS10_MINOR_VERSION](#) 1
- #define [TLS11_MINOR_VERSION](#) 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_SESSION_TBL_LEN](#) 8192
- #define [SSL_MAX_CONTENT_LEN](#) 16384
- #define [SSL_EXPIRATION_TIME](#) 86400
- #define [SSL_BUFFER_LEN](#) (SSL_MAX_CONTENT_LEN + 512)
- #define [SSL3_RSA_RC4_128_MD5](#) 4
- #define [SSL3_RSA_RC4_128_SHA](#) 5
- #define [SSL3_RSA_DES_168_SHA](#) 10
- #define [SSL3_EDH_RSA_DES_168_SHA](#) 22
- #define [TLS1_RSA_AES_256_SHA](#) 53
- #define [TLS1_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

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_HANDSHAKE_OVER](#) }

Functions

- `int ssl_client_start (ssl_context *ssl)`
- `int ssl_server_start (ssl_context *ssl)`
- `int ssl_derive_keys (ssl_context *ssl)`
- `int ssl_calc_verify (ssl_context *ssl, unsigned char hash[36])`
- `int ssl_read_record (ssl_context *ssl, int do_crypt)`
- `int ssl_write_record (ssl_context *ssl, int do_crypt)`
- `int ssl_flush_output (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)`
- `int ssl_write_finished (ssl_context *ssl)`
- `int ssl_parse_finished (ssl_context *ssl)`
- `int ssl_init (ssl_context *ssl, int client_resume)`
Initialize the SSL context. If client_resume is not null, the session id and premaster secret are preserved (client-side only).
- `void ssl_set_endpoint (ssl_context *ssl, int endpoint)`
Set the current endpoint type, SSL_IS_CLIENT or SSL_IS_SERVER.
- `void ssl_set_authmode (ssl_context *ssl, int authmode)`
Set the certificate verification mode.
- `void ssl_set_rng_func (ssl_context *ssl, int(*rng_f)(void *), void *rng_d)`
Set the random number generator function.
- `void ssl_set_io_files (ssl_context *ssl, int read_fd, int write_fd)`
Set the read and write file descriptors.
- `void ssl_set_ciphlist (ssl_context *ssl, int *ciphers)`
Set the list of allowed ciphersuites.
- `void ssl_set_ca_chain (ssl_context *ssl, x509_cert *ca, char *cn)`
Set the CA certificate chain used to verify peer cert, and the peer's expected CommonName (or NULL).
- `void ssl_set_rsa_cert (ssl_context *ssl, x509_cert *own_cert, rsa_context *own_key)`
Set own certificate and private RSA key.
- `void ssl_set_sidtable (ssl_context *ssl, unsigned char *sidtable)`
Set the global session ID table (server-side only).
- `int ssl_set_dhm_vals (ssl_context *ssl, char *dhm_P, char *dhm_G)`
Set the Diffie-Hellman public P and G values, provided as hexadecimal strings (server-side only).
- `int ssl_get_verify_result (ssl_context *ssl)`
Return the result of the certificate verification.
- `char * ssl_get_cipher_name (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 '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](#) []

8.103.1 Detailed Description

Definition in file [ssl.h](#).

8.103.2 Define Documentation

8.103.2.1 **#define ERR_SSL_BAD_HS_CERTIFICATE 0xA000**

Definition at line 35 of file [ssl.h](#).

8.103.2.2 **#define ERR_SSL_BAD_HS_CERTIFICATE_REQUEST 0xA800**

Definition at line 36 of file [ssl.h](#).

8.103.2.3 **#define ERR_SSL_BAD_HS_CERTIFICATE_VERIFY 0xC800**

Definition at line 40 of file [ssl.h](#).

Referenced by [ssl_parse_certificate_verify\(\)](#).

8.103.2.4 **#define ERR_SSL_BAD_HS_CHANGE_CIPHER_SPEC 0xD000**

Definition at line 41 of file [ssl.h](#).

8.103.2.5 #define ERR_SSL_BAD_HS_CLIENT_HELLO 0x9000

Definition at line 33 of file ssl.h.

Referenced by ssl_parse_client_hello().

8.103.2.6 #define ERR_SSL_BAD_HS_CLIENT_KEY_EXCHANGE 0xC000

Definition at line 39 of file ssl.h.

Referenced by ssl_parse_client_key_exchange().

8.103.2.7 #define ERR_SSL_BAD_HS_FINISHED 0xD800

Definition at line 42 of file ssl.h.

8.103.2.8 #define ERR_SSL_BAD_HS_SERVER_HELLO 0x9800

Definition at line 34 of file ssl.h.

Referenced by ssl_parse_server_hello().

8.103.2.9 #define ERR_SSL_BAD_HS_SERVER_HELLO_DONE 0xB800

Definition at line 38 of file ssl.h.

Referenced by ssl_parse_server_hello_done().

8.103.2.10 #define ERR_SSL_BAD_HS_SERVER_KEY_EXCHANGE 0xB000

Definition at line 37 of file ssl.h.

Referenced by ssl_parse_server_key_exchange().

8.103.2.11 #define ERR_SSL_CA_CHAIN_REQUIRED 0x6800

Definition at line 28 of file ssl.h.

8.103.2.12 #define ERR_SSL_CERTIFICATE_REQUIRED 0x5800

Definition at line 26 of file ssl.h.

Referenced by ssl_parse_certificate_request(), and ssl_write_certificate().

8.103.2.13 #define ERR_SSL_CERTIFICATE_TOO_LARGE 0x5000

Definition at line 25 of file ssl.h.

Referenced by ssl_write_certificate().

8.103.2.14 #define ERR_SSL_FATAL_ALERT_MESSAGE 0x7800

Definition at line 30 of file ssl.h.

8.103.2.15 #define ERR_SSL_FEATURE_UNAVAILABLE 0x1000

Definition at line 17 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().

8.103.2.16 #define ERR_SSL_INVALID_MAC 0x1800

Definition at line 18 of file ssl.h.

Referenced by ssl_decrypt_buf().

8.103.2.17 #define ERR_SSL_INVALID_MODULUS_SIZE 0x2800

Definition at line 20 of file ssl.h.

8.103.2.18 #define ERR_SSL_INVALID_RECORD 0x2000

Definition at line 19 of file ssl.h.

8.103.2.19 #define ERR_SSL_NO_CIPHER_CHOSEN 0x3800

Definition at line 22 of file ssl.h.

Referenced by ssl_parse_client_hello(), and ssl_parse_server_hello().

8.103.2.20 #define ERR_SSL_NO_CLIENT_CERTIFICATE 0x4800

Definition at line 24 of file ssl.h.

8.103.2.21 #define ERR_SSL_NO_SESSION_FOUND 0x4000

Definition at line 23 of file ssl.h.

Referenced by ssl_get_session().

8.103.2.22 #define ERR_SSL_PEER_CLOSE_NOTIFY 0x8800

Definition at line 32 of file ssl.h.

Referenced by main().

8.103.2.23 #define ERR_SSL_PEER_VERIFY_FAILED 0x8000

Definition at line 31 of file ssl.h.

8.103.2.24 #define ERR_SSL_PRIVATE_KEY_REQUIRED 0x6000

Definition at line 27 of file ssl.h.

Referenced by ssl_parse_certificate_request().

8.103.2.25 #define ERR_SSL_UNEXPECTED_MESSAGE 0x7000

Definition at line 29 of file ssl.h.

Referenced by ssl_parse_certificate_request(), ssl_parse_server_hello(), ssl_parse_server_hello_done(), ssl_parse_server_key_exchange(), and ssl_read().

8.103.2.26 #define ERR_SSL_UNKNOWN_CIPHER 0x3000

Definition at line 21 of file ssl.h.

8.103.2.27 #define SSL3_EDH_RSA_DES_168_SHA 22

Definition at line 76 of file ssl.h.

Referenced by ssl_derive_keys(), ssl_parse_client_key_exchange(), ssl_parse_server_key_exchange(), ssl_write_client_key_exchange(), and ssl_write_server_key_exchange().

8.103.2.28 #define SSL3_RSA_DES_168_SHA 10

Definition at line 75 of file ssl.h.

Referenced by ssl_derive_keys().

8.103.2.29 #define SSL3_RSA_RC4_128_MD5 4

Definition at line 73 of file ssl.h.

Referenced by ssl_derive_keys().

8.103.2.30 #define SSL3_RSA_RC4_128_SHA 5

Definition at line 74 of file ssl.h.

Referenced by ssl_derive_keys().

8.103.2.31 #define SSL_ALERT_CLOSE_NOTIFY 0

Definition at line 90 of file ssl.h.

Referenced by ssl_close_notify().

8.103.2.32 #define SSL_ALERT_FATAL 2

Definition at line 92 of file ssl.h.

8.103.2.33 #define SSL_ALERT_NO_CERTIFICATE 41

Definition at line 93 of file ssl.h.

Referenced by ssl_write_certificate().

8.103.2.34 #define SSL_ALERT_WARNING 1

Definition at line 91 of file ssl.h.

Referenced by ssl_close_notify(), and ssl_write_certificate().

8.103.2.35 #define SSL_BUFFER_LEN (SSL_MAX_CONTENT_LEN + 512)

Definition at line 68 of file ssl.h.

Referenced by ssl_free(), and ssl_init().

8.103.2.36 #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().

8.103.2.37 #define SSL_EXPIRATION_TIME 86400

Definition at line 62 of file ssl.h.

Referenced by ssl_get_session().

8.103.2.38 #define SSL_HS_CERTIFICATE 11

Definition at line 98 of file ssl.h.

Referenced by ssl_write_certificate().

8.103.2.39 #define SSL_HS_CERTIFICATE_REQUEST 13

Definition at line 100 of file ssl.h.

Referenced by ssl_parse_certificate_request(), and ssl_write_certificate_request().

8.103.2.40 #define SSL_HS_CERTIFICATE_VERIFY 15

Definition at line 102 of file ssl.h.

Referenced by ssl_parse_certificate_verify(), and ssl_write_certificate_verify().

8.103.2.41 #define SSL_HS_CLIENT_HELLO 1

Definition at line 96 of file ssl.h.

Referenced by ssl_parse_client_hello(), and ssl_write_client_hello().

8.103.2.42 #define SSL_HS_CLIENT_KEY_EXCHANGE 16

Definition at line 103 of file ssl.h.

Referenced by ssl_parse_client_key_exchange(), and ssl_write_client_key_exchange().

8.103.2.43 #define SSL_HS_FINISHED 20

Definition at line 104 of file ssl.h.

8.103.2.44 #define SSL_HS_HELLO_REQUEST 0

Definition at line 95 of file ssl.h.

8.103.2.45 #define SSL_HS_SERVER_HELLO 2

Definition at line 97 of file ssl.h.

Referenced by ssl_parse_server_hello(), and ssl_write_server_hello().

8.103.2.46 #define SSL_HS_SERVER_HELLO_DONE 14

Definition at line 101 of file ssl.h.

Referenced by ssl_parse_server_hello_done(), and ssl_write_server_hello_done().

8.103.2.47 #define SSL_HS_SERVER_KEY_EXCHANGE 12

Definition at line 99 of file ssl.h.

Referenced by ssl_parse_server_key_exchange(), and ssl_write_server_key_exchange().

8.103.2.48 #define SSL_IS_CLIENT 0

Definition at line 52 of file ssl.h.

Referenced by main(), ssl_derive_keys(), ssl_handshake(), and ssl_write_certificate().

8.103.2.49 #define SSL_IS_SERVER 1

Definition at line 53 of file ssl.h.

Referenced by main(), and ssl_handshake().

8.103.2.50 #define SSL_MAX_CONTENT_LEN 16384

Definition at line 61 of file ssl.h.

Referenced by ssl_write(), and ssl_write_certificate().

8.103.2.51 #define SSL_MSG_ALERT 21

Definition at line 86 of file ssl.h.

Referenced by ssl_close_notify(), and ssl_write_certificate().

8.103.2.52 #define SSL_MSG_APPLICATION_DATA 23

Definition at line 88 of file ssl.h.

Referenced by ssl_read(), and ssl_write().

8.103.2.53 #define SSL_MSG_CHANGE_CIPHER_SPEC 20

Definition at line 85 of file ssl.h.

8.103.2.54 #define SSL_MSG_HANDSHAKE 22

Definition at line 87 of file ssl.h.

Referenced by ssl_parse_certificate_request(), ssl_parse_certificate_verify(), ssl_parse_client_hello(), ssl_parse_client_key_exchange(), ssl_parse_server_hello(), ssl_parse_server_hello_done(), ssl_parse_server_key_exchange(), ssl_write_certificate(), ssl_write_certificate_request(), ssl_write_certificate_verify(), ssl_write_client_hello(), ssl_write_client_key_exchange(), ssl_write_record(), ssl_write_server_hello(), ssl_write_server_hello_done(), and ssl_write_server_key_exchange().

8.103.2.55 #define SSL_SESSION_TBL_LEN 8192

Definition at line 60 of file ssl.h.

Referenced by ssl_get_session(), and ssl_set_session().

8.103.2.56 #define SSL_VERIFY_NONE 0

Definition at line 56 of file ssl.h.

Referenced by main(), and ssl_write_certificate_request().

8.103.2.57 #define SSL_VERIFY_OPTIONAL 1

Definition at line 57 of file ssl.h.

Referenced by main().

8.103.2.58 #define SSL_VERIFY_REQUIRED 2

Definition at line 58 of file ssl.h.

Referenced by ssl_parse_certificate_verify().

8.103.2.59 #define SSLV3_MAJOR_VERSION 3

Definition at line 47 of file ssl.h.

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

8.103.2.60 #define SSLV3_MINOR_VERSION 0

Definition at line 48 of file ssl.h.

Referenced by ssl_calc_verify(), ssl_decrypt_buf(), ssl_derive_keys(), ssl_encrypt_buf(), ssl_parse_client_key_exchange(), ssl_parse_server_hello(), ssl_write_certificate(), ssl_write_client_hello(), and ssl_write_client_key_exchange().

8.103.2.61 #define TLS10_MINOR_VERSION 1

Definition at line 49 of file ssl.h.

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

8.103.2.62 #define TLS11_MINOR_VERSION 2

Definition at line 50 of file ssl.h.

8.103.2.63 #define TLS1_EDH_RSA_AES_256_SHA 57

Definition at line 78 of file ssl.h.

Referenced by ssl_derive_keys(), ssl_parse_client_key_exchange(), ssl_parse_server_key_exchange(), ssl_write_client_key_exchange(), and ssl_write_server_key_exchange().

8.103.2.64 #define TLS1_RSA_AES_256_SHA 53

Definition at line 77 of file ssl.h.

Referenced by ssl_derive_keys().

8.103.3 Enumeration Type Documentation**8.103.3.1 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_HANDSHAKE_OVER

Definition at line 109 of file ssl.h.

8.103.4 Function Documentation

8.103.4.1 `int ssl_calc_verify (ssl_context * ssl, unsigned char hash[36])`

Definition at line 326 of file ssl_tls.c.

References `ssl_context::hs_md5`, `ssl_context::hs_sha1`, `ssl_context::master`, `md5()`, `md5_finish()`, `md5_starts()`, `md5_update()`, `ssl_context::minor_ver`, `sha1()`, `sha1_finish()`, `sha1_starts()`, `sha1_update()`, and `SSLV3_MINOR_VERSION`.

Referenced by `ssl_parse_certificate_verify()`, and `ssl_write_certificate_verify()`.

8.103.4.2 `int ssl_client_start (ssl_context * ssl)`

Definition at line 438 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_flush_output()`, `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()`, and `ssl_context::state`.

Referenced by `ssl_handshake()`.

8.103.4.3 `int ssl_close_notify (ssl_context * ssl)`

Notify the peer that the connection is being closed.

Definition at line 1450 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_flush_output()`, `SSL_HANDSHAKE_OVER`, `SSL_MSG_ALERT`, `ssl_write_record()`, and `ssl_context::state`.

Referenced by `main()`.

8.103.4.4 `int ssl_derive_keys (ssl_context * ssl)`

Definition at line 101 of file `ssl_tls.c`.

References `aes_set_key()`, `arc4_setup()`, `ssl_context::cipher`, `ssl_context::ctx_dec`, `ssl_context::ctx_enc`, `ssl_context::ctxlen`, `des3_set_3keys()`, `ssl_context::endpoint`, `ERR_SSL_FEATURE_UNAVAILABLE`, `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_context::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::resumed`, `sha1()`, `sha1_finish()`, `sha1_starts()`, `sha1_update()`, `SSL3_EDH_RSA_DES_168_SHA`, `SSL3_RSA_DES_168_SHA`, `SSL3_RSA_RC4_128_MD5`, `SSL3_RSA_RC4_128_SHA`, `SSL_IS_CLIENT`, `SSLV3_MINOR_VERSION`, `TLS1_EDH_RSA_AES_256_SHA`, `tls1_prf()`, and `TLS1_RSA_AES_256_SHA`.

Referenced by `ssl_parse_client_key_exchange()`, `ssl_parse_server_hello()`, `ssl_write_client_key_exchange()`, and `ssl_write_server_hello()`.

8.103.4.5 `int ssl_flush_output (ssl_context * ssl)`

Definition at line 823 of file `ssl_tls.c`.

References `net_send()`, `ssl_context::out_hdr`, `ssl_context::out_left`, `ssl_context::out_msglen`, and `ssl_context::write_fd`.

Referenced by `ssl_client_start()`, `ssl_close_notify()`, and `ssl_server_start()`.

8.103.4.6 `void ssl_free (ssl_context * ssl)`

Free an SSL context.

Definition at line 1473 of file `ssl_tls.c`.

References `ssl_context::ctx_dec`, `ssl_context::ctx_enc`, `ssl_context::ctxlen`, `ssl_context::dhm_ctx`, `dhm_free()`, `ssl_context::in_ctr`, `ssl_context::out_ctr`, `ssl_context::peer_cert`, `SSL_BUFFER_LEN`, and `x509_free_cert()`.

Referenced by `main()`.

8.103.4.7 `char* ssl_get_cipher_name (ssl_context * ssl)`

Return the name of the current cipher.

Definition at line 1308 of file `ssl_tls.c`.

Referenced by `main()`.

8.103.4.8 `int ssl_get_verify_result (ssl_context * ssl)`

Return the result of the certificate verification.

Definition at line 1303 of file `ssl_tls.c`.

References `ssl_context::verify_result`.

Referenced by `main()`.

8.103.4.9 int ssl_handshake (ssl_context * ssl)

Perform the SSL handshake.

Returns:

0 if successful, ERR_NET_WOULD_BLOCK (only when the socket is set to non-blocking), or a specific SSL error code.

Definition at line 1369 of file ssl_tls.c.

References ssl_context::endpoint, ERR_SSL_FEATURE_UNAVAILABLE, ssl_client_start(), SSL_IS_CLIENT, SSL_IS_SERVER, and ssl_server_start().

Referenced by main(), ssl_read(), and ssl_write().

8.103.4.10 int ssl_init (ssl_context * ssl, int client_resume)

Initialize the SSL context. If client_resume is not null, the session id and premaster secret are preserved (client-side only).

Returns:

0 if successful, or 1 if memory allocation failed

Definition at line 1199 of file ssl_tls.c.

References ssl_context::in_ctr, ssl_context::in_hdr, ssl_context::in_msg, ssl_context::master, ssl_context::out_ctr, ssl_context::out_hdr, ssl_context::out_msg, ssl_context::sessid, ssl_context::sidlen, and SSL_BUFFER_LEN.

Referenced by main().

8.103.4.11 int ssl_parse_certificate (ssl_context * ssl)

Definition at line 914 of file ssl_tls.c.

Referenced by ssl_client_start(), and ssl_server_start().

8.103.4.12 int ssl_parse_change_cipher_spec (ssl_context * ssl)

Definition at line 1028 of file ssl_tls.c.

Referenced by ssl_client_start(), and ssl_server_start().

8.103.4.13 int ssl_parse_finished (ssl_context * ssl)

Definition at line 1154 of file ssl_tls.c.

Referenced by ssl_client_start(), and ssl_server_start().

8.103.4.14 int ssl_read (ssl_context * ssl, unsigned char * buf, int * len)

Read at most 'len' application data bytes.

Returns:

0 if successful, ERR_NET_WOULD_BLOCK (only when the socket is set to non-blocking), or a specific SSL error code.

Note:

len is updated to reflect the actual number of data bytes read.

Definition at line 1387 of file ssl_tls.c.

References ERR_SSL_UNEXPECTED_MESSAGE, ssl_context::in_msg, ssl_context::in_msglen, ssl_context::in_msgtype, ssl_context::in_offt, ssl_handshake(), SSL_MSG_APPLICATION_DATA, and ssl_read_record().

Referenced by main().

8.103.4.15 int ssl_read_record (ssl_context * ssl, int do_crypt)

Definition at line 682 of file ssl_tls.c.

Referenced by ssl_parse_certificate_request(), ssl_parse_certificate_verify(), ssl_parse_client_key_exchange(), ssl_parse_server_hello(), ssl_parse_server_hello_done(), ssl_parse_server_key_exchange(), and ssl_read().

8.103.4.16 int ssl_server_start (ssl_context * ssl)

Definition at line 663 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_flush_output(), 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(), and ssl_context::state.

Referenced by main(), and ssl_handshake().

8.103.4.17 void ssl_set_authmode (ssl_context * ssl, int authmode)

Set the certificate verification mode.

Parameters:

mode can be:

SSL_VERIFY_NONE: peer certificate is not checked (default).

SSL_VERIFY_OPTIONAL: peer certificate is checked, however the handshake continues even if verification failed; you may want to check ssl->verify_result after.

SSL_VERIFY_REQUIRED: peer *must* present a valid certificate, handshake is aborted if verification failed.

Definition at line 1249 of file ssl_tls.c.

Referenced by main().

8.103.4.18 void ssl_set_ca_chain (ssl_context * ssl, x509_cert * ca, char * cn)

Set the CA certificate chain used to verify peer cert, and the peer's expected CommonName (or NULL).

Definition at line 1273 of file ssl_tls.c.

Referenced by main().

8.103.4.19 void ssl_set_ciphlist (ssl_context * ssl, int * ciphers)

Set the list of allowed ciphersuites.

Definition at line 1268 of file ssl_tls.c.

Referenced by main().

8.103.4.20 int ssl_set_dhm_vals (ssl_context * ssl, char * dhm_P, char * dhm_G)

Set the Diffie-Hellman public P and G values, provided as hexadecimal strings (server-side only).

Returns:

0 if successful, or 1 if the values could not be read

Definition at line 1291 of file ssl_tls.c.

Referenced by main().

8.103.4.21 void ssl_set_endpoint (ssl_context * ssl, int endpoint)

Set the current endpoint type, SSL_IS_CLIENT or SSL_IS_SERVER.

Definition at line 1244 of file ssl_tls.c.

References ssl_context::endpoint.

Referenced by main().

8.103.4.22 void ssl_set_io_files (ssl_context * ssl, int read_fd, int write_fd)

Set the read and write file descriptors.

Definition at line 1262 of file ssl_tls.c.

Referenced by main().

8.103.4.23 void ssl_set_rng_func (ssl_context * ssl, int(*) (void *) rng_f, void * rng_d)

Set the random number generator function.

Definition at line 1254 of file ssl_tls.c.

Referenced by main().

8.103.4.24 void ssl_set_rsa_cert (ssl_context * ssl, x509_cert * own_cert, rsa_context * own_key)

Set own certificate and private RSA key.

Definition at line 1279 of file ssl_tls.c.

Referenced by main().

8.103.4.25 void ssl_set_sidtable (ssl_context * ssl, unsigned char * sidtable)

Set the global session ID table (server-side only).

Definition at line 1286 of file ssl_tls.c.

Referenced by main().

8.103.4.26 int ssl_write (ssl_context * ssl, unsigned char * buf, int len)

Write 'len' application data bytes.

Returns:

0 if successful, ERR_NET_WOULD_BLOCK (only when the socket is set to non-blocking), or a specific SSL error code.

Note:

When the socket is set to non-blocking and this function returns ERR_NET_WOULD_BLOCK, it should be called again with the *same* arguments until it returns 0.

Definition at line 1422 of file ssl_tls.c.

References ssl_context::out_msg, ssl_context::out_msglen, ssl_context::out_msgtype, ssl_context::out_uoff, ssl_handshake(), SSL_MAX_CONTENT_LEN, SSL_MSG_APPLICATION_DATA, and ssl_write_record().

Referenced by main().

8.103.4.27 int ssl_write_certificate (ssl_context * ssl)

Definition at line 843 of file ssl_tls.c.

References ssl_context::client_auth, ssl_context::endpoint, ERR_SSL_CERTIFICATE_REQUIRED, ERR_SSL_CERTIFICATE_TOO_LARGE, _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_HS_CERTIFICATE, SSL_IS_CLIENT, SSL_MAX_CONTENT_LEN, SSL_MSG_ALERT, SSL_MSG_HANDSHAKE, ssl_write_record(), SSLV3_MINOR_VERSION, and ssl_context::state.

Referenced by ssl_client_start(), and ssl_server_start().

8.103.4.28 int ssl_write_change_cipher_spec (ssl_context * ssl)

Definition at line 1019 of file ssl_tls.c.

Referenced by ssl_client_start(), and ssl_server_start().

8.103.4.29 int ssl_write_finished (ssl_context * ssl)

Definition at line 1117 of file ssl_tls.c.

Referenced by ssl_client_start(), and ssl_server_start().

8.103.4.30 int ssl_write_record (ssl_context * ssl, int do_crypt)

Definition at line 645 of file ssl_tls.c.

References ssl_context::hs_md5, ssl_context::hs_sha1, ssl_context::major_ver, md5_update(), ssl_context::minor_ver, net_send(), ssl_context::out_hdr, ssl_context::out_left, ssl_context::out_msg, ssl_context::out_msglen, ssl_context::out_msgtype, sha1_update(), ssl_encrypt_buf(), SSL_MSG_HANDSHAKE, and ssl_context::write_fd.

Referenced by ssl_close_notify(), ssl_write(), ssl_write_certificate(), ssl_write_certificate_request(), ssl_write_certificate_verify(), ssl_write_client_hello(), ssl_write_client_key_exchange(), ssl_write_server_hello(), ssl_write_server_hello_done(), and ssl_write_server_key_exchange().

8.103.5 Variable Documentation

8.103.5.1 int ssl_default_ciphers[]

Definition at line 1343 of file ssl_tls.c.

Referenced by main().

8.104 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/ssl_conf.h File Reference

8.104.1 Detailed Description

Defines the set of features `_not_` available, this is mostly useful for embedded systems.

Both MD5 and SHA-1 are required for SSL and therefore cannot be disabled here.

When compiling for ARMv3 / ARMv4T with gcc 3.4.6 -Os:

NO_GENPRIME saves 2196 bytes NO_DHM saves 2220 bytes NO_MD2 saves 2056 bytes NO_MD4 saves 3948 bytes NO_ARC4 saves 840 bytes NO_DES saves 18016 bytes NO_AES saves 19228 bytes NO_SSL_CLI saves 2644 bytes NO_SSL_SRV saves 3348 bytes

Definition in file [ssl_conf.h](#).

8.105 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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 [set_timer](#) (struct [hr_time](#) *val, int reset)
Return the elapsed time in milliseconds.
- void [set_alarm](#) (int seconds)
Setup an alarm clock.

Variables

- int [alarmed](#)

8.105.1 Detailed Description

Definition in file [timing.h](#).

8.105.2 Function Documentation

8.105.2.1 unsigned long hardclock (void)

Return the CPU cycle counter value.

Definition at line 135 of file timing.c.

References [tv_init](#).

Referenced by [main\(\)](#).

8.105.2.2 void set_alarm (int seconds)

Setup an alarm clock.

Parameters:

seconds delay before the "alarmed" flag is set

Definition at line 227 of file timing.c.

References sighandler().

Referenced by main().

8.105.2.3 unsigned long set_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 201 of file timing.c.

References _hr_time::start.

8.105.3 Variable Documentation

8.105.3.1 int alarmed

Definition at line 160 of file timing.c.

Referenced by main().

8.106 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/x509.h File Reference

```
#include "rsa.h"
```

Data Structures

- struct [_x509_buf](#)
- struct [_x509_name](#)
- struct [_x509_time](#)
- struct [_x509_cert](#)

Defines

- #define [ERR_ASN1_OUT_OF_DATA](#) 0x0014
- #define [ERR_ASN1_UNEXPECTED_TAG](#) 0x0016
- #define [ERR_ASN1_INVALID_LENGTH](#) 0x0018
- #define [ERR_ASN1_LENGTH_MISMATCH](#) 0x001A
- #define [ERR_ASN1_INVALID_DATA](#) 0x001C
- #define [ERR_X509_FEATURE_UNAVAILABLE](#) 0x0020
- #define [ERR_X509_CERT_INVALID_PEM](#) 0x0040
- #define [ERR_X509_CERT_INVALID_FORMAT](#) 0x0060
- #define [ERR_X509_CERT_INVALID_VERSION](#) 0x0080
- #define [ERR_X509_CERT_INVALID_SERIAL](#) 0x00A0
- #define [ERR_X509_CERT_INVALID_ALG](#) 0x00C0
- #define [ERR_X509_CERT_INVALID_NAME](#) 0x00E0
- #define [ERR_X509_CERT_INVALID_DATE](#) 0x0100
- #define [ERR_X509_CERT_INVALID_PUBKEY](#) 0x0120
- #define [ERR_X509_CERT_INVALID_SIGNATURE](#) 0x0140
- #define [ERR_X509_CERT_INVALID_EXTENSIONS](#) 0x0160
- #define [ERR_X509_CERT_UNKNOWN_VERSION](#) 0x0180
- #define [ERR_X509_CERT_UNKNOWN_SIG_ALG](#) 0x01A0
- #define [ERR_X509_CERT_UNKNOWN_PK_ALG](#) 0x01C0
- #define [ERR_X509_CERT_SIG_MISMATCH](#) 0x01E0
- #define [ERR_X509_KEY_INVALID_PEM](#) 0x0200
- #define [ERR_X509_KEY_INVALID_VERSION](#) 0x0220
- #define [ERR_X509_KEY_INVALID_FORMAT](#) 0x0240
- #define [ERR_X509_KEY_INVALID_ENC_IV](#) 0x0260
- #define [ERR_X509_KEY_UNKNOWN_ENC_ALG](#) 0x0280
- #define [ERR_X509_KEY_PASSWORD_REQUIRED](#) 0x02A0
- #define [ERR_X509_KEY_PASSWORD_MISMATCH](#) 0x02C0
- #define [ERR_X509_SIG_VERIFY_FAILED](#) 0x02E0
- #define [BADCERT_HAS_EXPIRED](#) 1
- #define [BADCERT_CN_MISMATCH](#) 2
- #define [BADCERT_NOT_TRUSTED](#) 4
- #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 `OID_X520` "\x55\x04"
- #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_PKCS9` "\x2A\x86\x48\x86\xF7\x0D\x01\x09"

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`

Functions

- `int x509_add_certs` (`x509_cert` *chain, unsigned char *buf, `int` buflen)
Parse one or more certificates and add them to the chain.
- `int x509_read_crtfile` (`x509_cert` *chain, char *path)
Load one or more certificates and add them to the chain.
- `int x509_parse_key` (`rsa_context` *rsa, unsigned char *buf, `int` buflen, unsigned char *pwd, `int` pwrlen)
Parse a private RSA key.
- `int x509_read_keyfile` (`rsa_context` *rsa, char *path, char *password)
Load and parse a private RSA key.
- `int x509_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 * x509_cert_info (x509_cert *crt)`

Return an informational string about the certificate, or NULL if memory allocation failed.

- `int x509_is_cert_expired (x509_cert *crt)`

Return 0 if the certificate is still valid, or BADCERT_HAS_EXPIRED.

- `int x509_verify_cert (x509_cert *crt, x509_cert *trust_ca, char *cn, int *flags)`

Verify the certificate signature.

- `void x509_free_cert (x509_cert *crt)`

Unallocate all certificate data.

- `int x509_self_test (int verbose)`

Checkup routine.

8.106.1 Detailed Description

Definition in file [x509.h](#).

8.106.2 Define Documentation

8.106.2.1 #define ASN1_BIT_STRING 0x03

Definition at line 52 of file [x509.h](#).

Referenced by [x509_get_pubkey\(\)](#), and [x509_get_sig\(\)](#).

8.106.2.2 #define ASN1_BMP_STRING 0x1E

Definition at line 64 of file [x509.h](#).

Referenced by [x509_get_name\(\)](#).

8.106.2.3 #define ASN1_BOOLEAN 0x01

Definition at line 50 of file [x509.h](#).

Referenced by [asn1_get_bool\(\)](#).

8.106.2.4 #define ASN1_CONSTRUCTED 0x20

Definition at line 66 of file [x509.h](#).

Referenced by [x509_add_certs\(\)](#), [x509_get_alg\(\)](#), [x509_get_dates\(\)](#), [x509_get_ext\(\)](#), [x509_get_name\(\)](#), [x509_get_pubkey\(\)](#), [x509_get_uid\(\)](#), [x509_get_version\(\)](#), and [x509_parse_key\(\)](#).

8.106.2.5 #define ASN1_CONTEXT_SPECIFIC 0x80

Definition at line 67 of file x509.h.

Referenced by x509_get_ext(), x509_get_serial(), x509_get_uid(), and x509_get_version().

8.106.2.6 #define ASN1_IA5_STRING 0x16

Definition at line 61 of file x509.h.

Referenced by x509_get_name().

8.106.2.7 #define ASN1_INTEGER 0x02

Definition at line 51 of file x509.h.

Referenced by asn1_get_int(), asn1_get_mpi(), and x509_get_serial().

8.106.2.8 #define ASN1_NULL 0x05

Definition at line 54 of file x509.h.

Referenced by x509_get_alg().

8.106.2.9 #define ASN1_OCTET_STRING 0x04

Definition at line 53 of file x509.h.

Referenced by x509_get_ext().

8.106.2.10 #define ASN1_OID 0x06

Definition at line 55 of file x509.h.

Referenced by x509_get_alg(), and x509_get_name().

8.106.2.11 #define ASN1_PRIMITIVE 0x00

Definition at line 65 of file x509.h.

Referenced by x509_get_serial().

8.106.2.12 #define ASN1_PRINTABLE_STRING 0x13

Definition at line 59 of file x509.h.

Referenced by x509_get_name().

8.106.2.13 #define ASN1_SEQUENCE 0x10

Definition at line 57 of file x509.h.

Referenced by x509_add_certs(), x509_get_alg(), x509_get_dates(), x509_get_ext(), x509_get_name(), x509_get_pubkey(), and x509_parse_key().

8.106.2.14 #define ASN1_SET 0x11

Definition at line 58 of file x509.h.

Referenced by x509_get_name().

8.106.2.15 #define ASN1_T61_STRING 0x14

Definition at line 60 of file x509.h.

Referenced by x509_get_name().

8.106.2.16 #define ASN1_UNIVERSAL_STRING 0x1C

Definition at line 63 of file x509.h.

Referenced by x509_get_name().

8.106.2.17 #define ASN1_UTC_TIME 0x17

Definition at line 62 of file x509.h.

Referenced by x509_get_dates().

8.106.2.18 #define ASN1_UTF8_STRING 0x0C

Definition at line 56 of file x509.h.

Referenced by x509_get_name().

8.106.2.19 #define BADCERT_CN_MISMATCH 2

Definition at line 44 of file x509.h.

Referenced by main(), and x509_verify_cert().

8.106.2.20 #define BADCERT_HAS_EXPIRED 1

Definition at line 43 of file x509.h.

Referenced by main(), and x509_is_cert_expired().

8.106.2.21 #define BADCERT_NOT_TRUSTED 4

Definition at line 45 of file x509.h.

Referenced by main(), and x509_verify_cert().

8.106.2.22 #define ERR_ASN1_INVALID_DATA 0x001C

Definition at line 17 of file x509.h.

8.106.2.23 #define ERR_ASN1_INVALID_LENGTH 0x0018

Definition at line 15 of file x509.h.

Referenced by `asn1_get_bool()`, `asn1_get_int()`, and `asn1_get_len()`.

8.106.2.24 #define ERR_ASN1_LENGTH_MISMATCH 0x001A

Definition at line 16 of file x509.h.

Referenced by `x509_add_certs()`, `x509_get_alg()`, `x509_get_dates()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_version()`, and `x509_parse_key()`.

8.106.2.25 #define ERR_ASN1_OUT_OF_DATA 0x0014

Definition at line 13 of file x509.h.

Referenced by `asn1_get_len()`, `asn1_get_tag()`, `x509_get_name()`, `x509_get_pubkey()`, and `x509_get_serial()`.

8.106.2.26 #define ERR_ASN1_UNEXPECTED_TAG 0x0016

Definition at line 14 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()`.

8.106.2.27 #define ERR_X509_CERT_INVALID_ALG 0x00C0

Definition at line 24 of file x509.h.

Referenced by `x509_get_alg()`.

8.106.2.28 #define ERR_X509_CERT_INVALID_DATE 0x0100

Definition at line 26 of file x509.h.

Referenced by `x509_get_dates()`.

8.106.2.29 #define ERR_X509_CERT_INVALID_EXTENSIONS 0x0160

Definition at line 29 of file x509.h.

Referenced by `x509_get_ext()`.

8.106.2.30 #define ERR_X509_CERT_INVALID_FORMAT 0x0060

Definition at line 21 of file x509.h.

Referenced by x509_add_certs().

8.106.2.31 #define ERR_X509_CERT_INVALID_NAME 0x00E0

Definition at line 25 of file x509.h.

Referenced by x509_get_name().

8.106.2.32 #define ERR_X509_CERT_INVALID_PEM 0x0040

Definition at line 20 of file x509.h.

Referenced by x509_add_certs().

8.106.2.33 #define ERR_X509_CERT_INVALID_PUBKEY 0x0120

Definition at line 27 of file x509.h.

Referenced by x509_get_pubkey().

8.106.2.34 #define ERR_X509_CERT_INVALID_SERIAL 0x00A0

Definition at line 23 of file x509.h.

Referenced by x509_get_serial().

8.106.2.35 #define ERR_X509_CERT_INVALID_SIGNATURE 0x0140

Definition at line 28 of file x509.h.

Referenced by x509_get_sig().

8.106.2.36 #define ERR_X509_CERT_INVALID_VERSION 0x0080

Definition at line 22 of file x509.h.

Referenced by x509_get_version().

8.106.2.37 #define ERR_X509_CERT_SIG_MISMATCH 0x01E0

Definition at line 33 of file x509.h.

Referenced by x509_add_certs().

8.106.2.38 #define ERR_X509_CERT_UNKNOWN_PK_ALG 0x01C0

Definition at line 32 of file x509.h.

Referenced by x509_get_pubkey().

8.106.2.39 #define ERR_X509_CERT_UNKNOWN_SIG_ALG 0x01A0

Definition at line 31 of file x509.h.

Referenced by x509_add_certs().

8.106.2.40 #define ERR_X509_CERT_UNKNOWN_VERSION 0x0180

Definition at line 30 of file x509.h.

Referenced by x509_add_certs().

8.106.2.41 #define ERR_X509_FEATURE_UNAVAILABLE 0x0020

Definition at line 19 of file x509.h.

Referenced by x509_parse_key().

8.106.2.42 #define ERR_X509_KEY_INVALID_ENC_IV 0x0260

Definition at line 37 of file x509.h.

Referenced by x509_des3_getiv(), and x509_parse_key().

8.106.2.43 #define ERR_X509_KEY_INVALID_FORMAT 0x0240

Definition at line 36 of file x509.h.

Referenced by x509_parse_key().

8.106.2.44 #define ERR_X509_KEY_INVALID_PEM 0x0200

Definition at line 34 of file x509.h.

Referenced by x509_parse_key().

8.106.2.45 #define ERR_X509_KEY_INVALID_VERSION 0x0220

Definition at line 35 of file x509.h.

Referenced by x509_parse_key().

8.106.2.46 #define ERR_X509_KEY_PASSWORD_MISMATCH 0x02C0

Definition at line 40 of file x509.h.

Referenced by x509_parse_key().

8.106.2.47 #define ERR_X509_KEY_PASSWORD_REQUIRED 0x02A0

Definition at line 39 of file x509.h.

Referenced by x509_parse_key().

8.106.2.48 #define ERR_X509_KEY_UNKNOWN_ENC_ALG 0x0280

Definition at line 38 of file x509.h.

Referenced by x509_parse_key().

8.106.2.49 #define ERR_X509_SIG_VERIFY_FAILED 0x02E0

Definition at line 41 of file x509.h.

Referenced by x509_verify_cert().

8.106.2.50 #define OID_PKCS1 "\x2A\x86\x48\x86\xF7\x0D\x01\x01"

Definition at line 81 of file x509.h.

Referenced by x509_add_certs().

8.106.2.51 #define OID_PKCS1_RSA "\x2A\x86\x48\x86\xF7\x0D\x01\x01\x01"

Definition at line 82 of file x509.h.

Referenced by x509_get_pubkey().

8.106.2.52 #define OID_PKCS9 "\x2A\x86\x48\x86\xF7\x0D\x01\x09"

Definition at line 83 of file x509.h.

Referenced by x509_dn_gets().

8.106.2.53 #define OID_X520 "\x55\x04"

Definition at line 80 of file x509.h.

Referenced by x509_dn_gets().

8.106.2.54 #define PKCS9_EMAIL 1

Definition at line 78 of file x509.h.

Referenced by x509_dn_gets().

8.106.2.55 #define X520_COMMON_NAME 3

Definition at line 72 of file x509.h.

Referenced by x509_dn_gets().

8.106.2.56 #define X520_COUNTRY 6

Definition at line 73 of file x509.h.

Referenced by x509_dn_gets().

8.106.2.57 #define X520_LOCALITY 7

Definition at line 74 of file x509.h.

Referenced by x509_dn_gets().

8.106.2.58 #define X520_ORG_UNIT 11

Definition at line 77 of file x509.h.

Referenced by x509_dn_gets().

8.106.2.59 #define X520_ORGANIZATION 10

Definition at line 76 of file x509.h.

Referenced by x509_dn_gets().

8.106.2.60 #define X520_STATE 8

Definition at line 75 of file x509.h.

Referenced by x509_dn_gets().

8.106.3 Typedef Documentation**8.106.3.1 typedef struct _x509_buf x509_buf****8.106.3.2 typedef struct _x509_cert x509_cert****8.106.3.3 typedef struct _x509_name x509_name****8.106.3.4 typedef struct _x509_time x509_time****8.106.4 Function Documentation****8.106.4.1 int x509_add_certs (x509_cert * *chain*, unsigned char * *buf*, int *buflen*)**

Parse one or more certificates and add them to the chain.

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 643 of file x509read.c.

References ASN1_CONSTRUCTED, asn1_get_tag(), ASN1_SEQUENCE, base64_decode(), _x509_cert::ca_istrue, rsa_context::E, ERR_ASN1_LENGTH_MISMATCH, ERR_BASE64_INVALID_CHARACTER, ERR_X509_CERT_INVALID_FORMAT, ERR_X509_CERT_INVALID_PEM, ERR_X509_CERT_SIG_MISMATCH, ERR_X509_CERT_UNKNOWN_SIG_ALG, ERR_X509_CERT_UNKNOWN_VERSION, _x509_cert::issuer, _x509_cert::issuer_id, _x509_cert::issuer_raw, rsa_context::len, _x509_buf::len, _x509_cert::max_pathlen, mpi_msb(), 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_add_certs(), x509_free_cert(), x509_get_alg(), x509_get_dates(), x509_get_ext(), x509_get_name(), x509_get_pubkey(), x509_get_serial(), x509_get_sig(), x509_get_uid(), and x509_get_version().

Referenced by main(), x509_add_certs(), and x509_read_crtfile().

8.106.4.2 char* x509_cert_info (x509_cert * crt)

Return an informational string about the certificate, or NULL if memory allocation failed.

Definition at line 1388 of file x509read.c.

References _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, x509_dn_gets(), and _x509_time::year.

8.106.4.3 int x509_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 1305 of file x509read.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 x509_cert_info().

8.106.4.4 void x509_free_cert (x509_cert * crt)

Unallocate all certificate data.

Definition at line 1595 of file x509read.c.

References _x509_cert::issuer, _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(), and x509_add_certs().

8.106.4.5 int x509_is_cert_expired (x509_cert * crt)

Return 0 if the certificate is still valid, or BADCERT_HAS_EXPIRED.

Definition at line 1449 of file x509read.c.

References `BADCERT_HAS_EXPIRED`, `_x509_time::day`, `_x509_time::mon`, `_x509_cert::valid_to`, and `_x509_time::year`.

Referenced by `x509_verify_cert()`.

8.106.4.6 `int x509_parse_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 1074 of file `x509read.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`, `ERR_ASN1_LENGTH_MISMATCH`, `ERR_BASE64_INVALID_CHARACTER`, `ERR_X509_FEATURE_UNAVAILABLE`, `ERR_X509_KEY_INVALID_ENC_IV`, `ERR_X509_KEY_INVALID_FORMAT`, `ERR_X509_KEY_INVALID_PEM`, `ERR_X509_KEY_INVALID_VERSION`, `ERR_X509_KEY_PASSWORD_MISMATCH`, `ERR_X509_KEY_PASSWORD_REQUIRED`, `ERR_X509_KEY_UNKNOWN_ENC_ALG`, `rsa_context::len`, `mpi_msb()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_check_privkey()`, `rsa_free()`, `rsa_context::ver`, `x509_des3_decrypt()`, and `x509_des3_getiv()`.

Referenced by `main()`, and `x509_read_keyfile()`.

8.106.4.7 `int x509_read_crfile(x509_cert *chain, char *path)`

Load one or more certificates and add them to the chain.

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 971 of file `x509read.c`.

References `f`, and `x509_add_certs()`.

8.106.4.8 `int x509_read_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 1258 of file x509read.c.

References `f`, and `x509_parse_key()`.

8.106.4.9 `int x509_self_test (int verbose)`

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 1723 of file x509read.c.

Referenced by `main()`.

8.106.4.10 `int x509_verify_cert (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 `ERR_X509_SIG_VERIFY_FAILED`, in which case **flags* will have one or more of the following values set: `BADCERT_HAS_EXPIRED` – `BADCERT_CN_MISMATCH` – `BADCERT_NOT_TRUSTED`

Definition at line 1494 of file x509read.c.

References `BADCERT_CN_MISMATCH`, `BADCERT_NOT_TRUSTED`, `_x509_cert::ca_istrue`, `ERR_X509_SIG_VERIFY_FAILED`, `_x509_cert::issuer_raw`, `_x509_buf::len`, `_x509_cert::max_pathlen`, `_x509_cert::next`, `_x509_name::next`, `_x509_name::oid`, `_x509_buf::p`, `_x509_cert::rsa`, `rsa_pkcs1_verify()`, `_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()`, and `x509_is_cert_expired()`.

8.107 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/aes.c File Reference

```
#include <string.h>
#include "xyssl/aes.h"
```

Defines

- #define [_CRT_SECURE_NO_DEPRECATED](#) 1
- #define [uint8](#) unsigned char
- #define [uint32](#) unsigned long
- #define [GET_UINT32_BE](#)(n, b, i)
- #define [PUT_UINT32_BE](#)(n, b, i)
- #define [ROTR8](#)(x)
- #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_set_key](#) ([aes_context](#) *ctx, uint8 *key, int keysize)
- void [aes_encrypt](#) ([aes_context](#) *ctx, unsigned char input[16], unsigned char output[16])
AES block encryption (ECB mode).
- void [aes_decrypt](#) ([aes_context](#) *ctx, unsigned char input[16], unsigned char output[16])
AES block decryption (ECB mode).
- void [aes_cbc_encrypt](#) ([aes_context](#) *ctx, unsigned char iv[16], unsigned char *input, unsigned char *output, int len)
AES-CBC buffer encryption.
- void [aes_cbc_decrypt](#) ([aes_context](#) *ctx, unsigned char iv[16], unsigned char *input, unsigned char *output, int len)
AES-CBC buffer decryption.
- int [aes_self_test](#) (int verbose)
Checkup routine.

Variables

- static uint8 [FSb](#) [256]
- static uint32 [FT0](#) [256]
- static uint32 [FT1](#) [256]
- static uint32 [FT2](#) [256]
- static uint32 [FT3](#) [256]

- static uint8 [RSb](#) [256]
- static uint32 [RT0](#) [256]
- static uint32 [RT1](#) [256]
- static uint32 [RT2](#) [256]
- static uint32 [RT3](#) [256]
- static uint32 [RCON](#) [10]
- static uint32 [KT0](#) [256]
- static uint32 [KT1](#) [256]
- static uint32 [KT2](#) [256]
- static uint32 [KT3](#) [256]
- static const char [_aes_src](#) [] = "_aes_src"

8.107.1 Define Documentation

8.107.1.1 #define _CRT_SECURE_NO_DEPRECATED 1

Definition at line 28 of file aes.c.

8.107.1.2 #define AES_FROUND(X0, X1, X2, X3, Y0, Y1, Y2, Y3)

Value:

```
{
    RK += 4;

    X0 = RK[0] ^ FT0[ (uint8) ( Y0 >> 24 ) ] ^
             FT1[ (uint8) ( Y1 >> 16 ) ] ^
             FT2[ (uint8) ( Y2 >>  8 ) ] ^
             FT3[ (uint8) ( Y3          ) ];

    X1 = RK[1] ^ FT0[ (uint8) ( Y1 >> 24 ) ] ^
             FT1[ (uint8) ( Y2 >> 16 ) ] ^
             FT2[ (uint8) ( Y3 >>  8 ) ] ^
             FT3[ (uint8) ( Y0          ) ];

    X2 = RK[2] ^ FT0[ (uint8) ( Y2 >> 24 ) ] ^
             FT1[ (uint8) ( Y3 >> 16 ) ] ^
             FT2[ (uint8) ( Y0 >>  8 ) ] ^
             FT3[ (uint8) ( Y1          ) ];

    X3 = RK[3] ^ FT0[ (uint8) ( Y3 >> 24 ) ] ^
             FT1[ (uint8) ( Y0 >> 16 ) ] ^
             FT2[ (uint8) ( Y1 >>  8 ) ] ^
             FT3[ (uint8) ( Y2          ) ];
}
```

8.107.1.3 #define AES_RROUND(X0, X1, X2, X3, Y0, Y1, Y2, Y3)

Value:

```
{
    RK += 4;

    X0 = RK[0] ^ RT0[ (uint8) ( Y0 >> 24 ) ] ^
             RT1[ (uint8) ( Y3 >> 16 ) ] ^
```

```

        RT2[ (uint8) ( Y2 >> 8 ) ] ^ \
        RT3[ (uint8) ( Y1          ) ]; \
    \
X1 = RK[1] ^ RT0[ (uint8) ( Y1 >> 24 ) ] ^ \
        RT1[ (uint8) ( Y0 >> 16 ) ] ^ \
        RT2[ (uint8) ( Y3 >> 8 ) ] ^ \
        RT3[ (uint8) ( Y2          ) ]; \
    \
X2 = RK[2] ^ RT0[ (uint8) ( Y2 >> 24 ) ] ^ \
        RT1[ (uint8) ( Y1 >> 16 ) ] ^ \
        RT2[ (uint8) ( Y0 >> 8 ) ] ^ \
        RT3[ (uint8) ( Y3          ) ]; \
    \
X3 = RK[3] ^ RT0[ (uint8) ( Y3 >> 24 ) ] ^ \
        RT1[ (uint8) ( Y2 >> 16 ) ] ^ \
        RT2[ (uint8) ( Y1 >> 8 ) ] ^ \
        RT3[ (uint8) ( Y0          ) ]; \
}

```

8.107.1.4 #define GET_UINT32_BE(n, b, i)

Value:

```

{
    (n) = ( (uint32) (b)[(i)      ] << 24 ) \
        | ( (uint32) (b)[(i) + 1] << 16 ) \
        | ( (uint32) (b)[(i) + 2] << 8  ) \
        | ( (uint32) (b)[(i) + 3]      ); \
}

```

Definition at line 47 of file aes.c.

Referenced by aes_set_key(), des3_crypt(), des_crypt(), des_main_ks(), sha1_process(), and sha2_process().

8.107.1.5 #define MUL(x, y) ((x && y) ? pow[(log[x] + log[y]) % 255] : 0)

Definition at line 103 of file aes.c.

Referenced by aes_gen_tables().

8.107.1.6 #define PUT_UINT32_BE(n, b, i)

Value:

```

{
    (b)[(i)      ] = (uint8) ( (n) >> 24 ); \
    (b)[(i) + 1] = (uint8) ( (n) >> 16 ); \
    (b)[(i) + 2] = (uint8) ( (n) >> 8  ); \
    (b)[(i) + 3] = (uint8) ( (n)          ); \
}

```

Definition at line 56 of file aes.c.

Referenced by des3_crypt(), des_crypt(), sha1_finish(), and sha2_finish().

8.107.1.7 #define ROTR8(x)

Value:

```
( ( ( x << 24 ) & 0xFFFFFFFF ) | \
    ( ( x & 0xFFFFFFFF ) >> 8 ) )
```

Definition at line 100 of file aes.c.

Referenced by aes_gen_tables().

8.107.1.8 #define uint32 unsigned long

Definition at line 40 of file aes.c.

Referenced by aes_gen_tables(), and aes_set_key().

8.107.1.9 #define uint8 unsigned char

Definition at line 36 of file aes.c.

Referenced by aes_gen_tables(), and aes_set_key().

8.107.1.10 #define XTIME(x) ((x << 1) ^ ((x & 0x80) ? 0x1B : 0x00))

Definition at line 102 of file aes.c.

Referenced by aes_gen_tables().

8.107.2 Function Documentation

8.107.2.1 void aes_cbc_decrypt (aes_context * ctx, unsigned char iv[16], unsigned char * input, unsigned char * output, int len)

AES-CBC buffer decryption.

Parameters:

ctx AES context
iv initialization vector (modified after use)
input buffer holding the ciphertext
output buffer holding the plaintext
len length of the data to be decrypted

Definition at line 822 of file aes.c.

Referenced by ssl_decrypt_buf().

8.107.2.2 void aes_cbc_encrypt (aes_context * ctx, unsigned char iv[16], unsigned char * input, unsigned char * output, int len)

AES-CBC buffer encryption.

Parameters:

ctx AES context
iv initialization vector (modified after use)
input buffer holding the plaintext
output buffer holding the ciphertext
len length of the data to be encrypted

Definition at line 797 of file aes.c.

Referenced by main(), and ssl_encrypt_buf().

8.107.2.3 void aes_decrypt (aes_context * ctx, unsigned char input[16], unsigned char output[16])

AES block decryption (ECB mode).

Parameters:

ctx AES context
input ciphertext block
output plaintext block

Definition at line 706 of file aes.c.

Referenced by main().

8.107.2.4 void aes_encrypt (aes_context * ctx, unsigned char input[16], unsigned char output[16])

AES block encryption (ECB mode).

AES block encryption (ECB mode)

Definition at line 615 of file aes.c.

Referenced by main().

8.107.2.5 static void aes_gen_tables (void) [static]

Definition at line 105 of file aes.c.

References FSb, FT0, FT1, FT2, FT3, MUL, RCON, ROTR8, RSb, RT0, RT1, RT2, RT3, uint32, uint8, and XTIME.

Referenced by aes_set_key().

8.107.2.6 int aes_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 920 of file aes.c.

Referenced by main().

8.107.2.7 void aes_set_key (aes_context * ctx, uint8 * key, int keysize)

Definition at line 457 of file aes.c.

References aes_gen_tables(), aes_context::drk, aes_context::erk, FSb, GET_UINT32_BE, KT0, KT1, KT2, KT3, aes_context::nr, RCON, RT0, RT1, RT2, RT3, uint32, and uint8.

8.107.3 Variable Documentation**8.107.3.1 const char _aes_src[] = "_aes_src" [static]**

Definition at line 847 of file aes.c.

8.107.3.2 uint8 FSb[256] [static]

Definition at line 77 of file aes.c.

Referenced by aes_gen_tables(), and aes_set_key().

8.107.3.3 uint32 FT0[256] [static]

Definition at line 78 of file aes.c.

Referenced by aes_gen_tables().

8.107.3.4 uint32 FT1[256] [static]

Definition at line 79 of file aes.c.

Referenced by aes_gen_tables().

8.107.3.5 uint32 FT2[256] [static]

Definition at line 80 of file aes.c.

Referenced by aes_gen_tables().

8.107.3.6 uint32 FT3[256] [static]

Definition at line 81 of file aes.c.

Referenced by aes_gen_tables().

8.107.3.7 uint32 KT0[256] [static]

Definition at line 449 of file aes.c.

Referenced by aes_set_key().

8.107.3.8 uint32 KT1[256] [static]

Definition at line 450 of file aes.c.

Referenced by aes_set_key().

8.107.3.9 uint32 KT2[256] [static]

Definition at line 451 of file aes.c.

Referenced by aes_set_key().

8.107.3.10 uint32 KT3[256] [static]

Definition at line 452 of file aes.c.

Referenced by aes_set_key().

8.107.3.11 uint32 RCON[10] [static]

Definition at line 95 of file aes.c.

Referenced by aes_gen_tables(), and aes_set_key().

8.107.3.12 uint8 RSb[256] [static]

Definition at line 86 of file aes.c.

Referenced by aes_gen_tables().

8.107.3.13 uint32 RT0[256] [static]

Definition at line 87 of file aes.c.

Referenced by aes_gen_tables(), and aes_set_key().

8.107.3.14 uint32 RT1[256] [static]

Definition at line 88 of file aes.c.

Referenced by aes_gen_tables(), and aes_set_key().

8.107.3.15 uint32 RT2[256] [static]

Definition at line 89 of file aes.c.

Referenced by aes_gen_tables(), and aes_set_key().

8.107.3.16 uint32 RT3[256] [static]

Definition at line 90 of file aes.c.

Referenced by aes_gen_tables(), and aes_set_key().

8.108 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/arc4.c File Reference

```
#include "xyssl/arc4.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

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 char `_arc4_src` [] = "_arc4_src"

8.108.1 Define Documentation

8.108.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 27 of file arc4.c.

8.108.2 Function Documentation

8.108.2.1 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 60 of file arc4.c.

References `arc4_context::m`, `arc4_context::x`, and `arc4_context::y`.

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

8.108.2.2 `int arc4_self_test (int verbose)`

Definition at line 147 of file arc4.c.

Referenced by main().

8.108.2.3 `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 main(), and ssl_derive_keys().

8.108.3 Variable Documentation

8.108.3.1 `const char _arc4_src[] = "_arc4_src" [static]`

Definition at line 81 of file arc4.c.

8.109 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/base64.c File Reference

```
#include "xyssl/base64.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED` 1

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 int base64_enc_map [64]
- static const int base64_dec_map [128]
- static const char _base64_src [] = "_base64_src"

8.109.1 Define Documentation

8.109.1.1 `#define _CRT_SECURE_NO_DEPRECATED` 1

Definition at line 22 of file base64.c.

8.109.2 Function Documentation

8.109.2.1 `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 (updated after call)
- src* source buffer
- slen* amount of data to be decoded

Returns:

0 if successful, ERR_BASE64_BUFFER_TOO_SMALL, or ERR_BASE64_INVALID_DATA if an invalid char is found. *dlen is always updated to reflect to amount of data that was written (or would have been written)

Note:

Call this function with *dlen = 0 to obtain the required buffer size in *dlen

Definition at line 118 of file base64.c.

References base64_dec_map, ERR_BASE64_BUFFER_TOO_SMALL, and ERR_BASE64_INVALID_CHARACTER.

Referenced by x509_add_certs(), and x509_parse_key().

8.109.2.2 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 (updated after call)
src source buffer
slen amount of data to be encoded

Returns:

0 if successful, or ERR_BASE64_BUFFER_TOO_SMALL. *dlen is always updated to reflect to amount of data that was written (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 ERR_BASE64_BUFFER_TOO_SMALL.

8.109.2.3 int base64_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 245 of file base64.c.

Referenced by main().

8.109.3 Variable Documentation**8.109.3.1 const char _base64_src[] = "_base64_src" [static]**

Definition at line 179 of file base64.c.

8.109.3.2 const int 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, 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().

8.109.3.3 const int 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().

8.110 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/bignum.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <stdarg.h>
#include <stdio.h>
#include "xyssl/bignum.h"
#include "xyssl/bn_asm.h"
```

Defines

- #define [_CRT_SECURE_NO_DEPRECATED](#) 1
- #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](#) nblimbs)
Enlarge X 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.
- static [int](#) [mpi_get_digit](#) ([t_int](#) *d, [int](#) radix, char c)
- [int](#) [mpi_read_string](#) ([mpi](#) *X, [int](#) radix, char *s)
Import X 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 X into an ASCII string.
- [int](#) [mpi_read_file](#) ([mpi](#) *X, [int](#) radix, FILE *fin)
Read X from an opened file.

- `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_msb` (`mpi *X`)
Return the total size in bits, without leading 0s.
- `int mpi_lsb` (`mpi *X`)
Return the number of least significant bits.
- `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)`
Miller-Rabin primality test.
- `int mpi_gen_prime (mpi *X, int nbits, int dh_flag, int(*rng_f)(void *), void *rng_d)`
Prime number generation.
- `int mpi_self_test (int verbose)`
Checkup routine.

Variables

- `static const int small_prime []`
- `static const char _bignum_src [] = "_bignum_src"`

8.110.1 Define Documentation

8.110.1.1 #define _CRT_SECURE_NO_DEPRECATED 1

Definition at line 28 of file bignum.c.

8.110.1.2 #define BITS_TO_LIMBS(i) (((i) + biL - 1) / biL)

Definition at line 42 of file bignum.c.

Referenced by mpi_gen_prime(), mpi_read_string(), and mpi_shift_l().

8.110.1.3 #define CHARS_TO_LIMBS(i) (((i) + ciL - 1) / ciL)

Definition at line 43 of file bignum.c.

Referenced by mpi_read_binary().

8.110.2 Function Documentation**8.110.2.1 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 641 of file bignum.c.

References CHK, mpi_copy(), mpi_grow(), mpi::n, and mpi::p.

Referenced by mpi_add_mpi(), and mpi_sub_mpi().

8.110.2.2 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 806 of file bignum.c.

References mpi_add_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by mpi_gen_prime(), and mpi_read_string().

8.110.2.3 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 744 of file bignum.c.

References CHK, mpi_add_abs(), 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().

8.110.2.4 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 562 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().

8.110.2.5 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 625 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().

8.110.2.6 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 592 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(), rsa_check_privkey(), rsa_gen_key(), rsa_private(), and rsa_public().

8.110.2.7 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 115 of file bignum.c.

References CHK, ciL, 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().

8.110.2.8 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, ERR_MPI_DIVISION_BY_ZERO if $b == 0$

Note:

Either *Q* or *R* can be NULL.

Definition at line 1117 of file bignum.c.

References mpi_div_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by main(), and mpi_write_hlp().

8.110.2.9 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, ERR_MPI_DIVISION_BY_ZERO if $B == 0$

Note:

Either *Q* or *R* can be NULL.

Definition at line 953 of file bignum.c.

References biH, biL, CHK, ERR_MPI_DIVISION_BY_ZERO, mpi_add_mpi(), 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, and mpi::s.

Referenced by mpi_div_int(), and mpi_mod_mpi().

8.110.2.10 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, ERR_MPI_INVALID_PARAMETER 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 1263 of file bignum.c.

References biL, CHK, ERR_MPI_INVALID_PARAMETER, 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, and mpi::p.

Referenced by dhm_calc_secret(), dhm_make_params(), dhm_make_public(), main(), mpi_is_prime(), rsa_private(), and rsa_public().

8.110.2.11 void mpi_free (mpi * X, ...)

Unallocate one or more [mpi](#).

Definition at line 66 of file bignum.c.

References [ciL](#), [mpi::n](#), and [mpi::p](#).

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_sub_abs\(\)](#), [mpi_write_string\(\)](#), [rsa_check_privkey\(\)](#), [rsa_free\(\)](#), [rsa_gen_key\(\)](#), [rsa_private\(\)](#), and [rsa_public\(\)](#).

8.110.2.12 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 1440 of file bignum.c.

References [CHK](#), [mpi_cmp_int\(\)](#), [mpi_cmp_mpi\(\)](#), [mpi_copy\(\)](#), [mpi_free\(\)](#), [mpi_init\(\)](#), [mpi_lsb\(\)](#), [mpi_lset\(\)](#), [mpi_mul_mpi\(\)](#), [mpi_shift_l\(\)](#), [mpi_shift_r\(\)](#), [mpi_sub_abs\(\)](#), [mpi::p](#), and [mpi::s](#).

Referenced by [mpi_inv_mod\(\)](#), [rsa_check_privkey\(\)](#), and [rsa_gen_key\(\)](#).

8.110.2.13 int mpi_gen_prime (mpi * X, int nbits, int dh_flag, int(*) (void *) rng_f, void * rng_d)

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

rng_f points to the RNG function

rng_d points to the RNG data

Returns:

0 if successful (probably prime), 1 if memory allocation failed, [ERR_MPI_INVALID_PARAMETER](#) if nbits is < 3

Definition at line 1710 of file bignum.c.

References [BITS_TO_LIMBS](#), [CHK](#), [ERR_MPI_INVALID_PARAMETER](#), [ERR_MPI_NOT_ACCEPTABLE](#), [mpi_add_int\(\)](#), [mpi_free\(\)](#), [mpi_grow\(\)](#), [mpi_init\(\)](#), [mpi_is_prime\(\)](#), [mpi_lset\(\)](#), [mpi_msb\(\)](#), [mpi_shift_l\(\)](#), [mpi_shift_r\(\)](#), [mpi_sub_int\(\)](#), and [mpi::p](#).

Referenced by [main\(\)](#), and [rsa_gen_key\(\)](#).

8.110.2.14 static int mpi_get_digit (t_int * d, int radix, char c) [static]

Definition at line 171 of file bignum.c.

References ERR_MPI_INVALID_CHARACTER.

Referenced by mpi_read_file(), and mpi_read_string().

8.110.2.15 int mpi_grow (mpi * X, int nlimbs)

Enlarge X to the specified number of limbs.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 91 of file bignum.c.

References ciL, mpi::n, mpi::p, and mpi::s.

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().

8.110.2.16 void mpi_init (mpi * X, ...)

Initialize one or more [mpi](#).

Definition at line 48 of file bignum.c.

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_sub_abs(), mpi_write_string(), rsa_check_privkey(), rsa_gen_key(), rsa_private(), and rsa_public().

8.110.2.17 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, ERR_MPI_INVALID_PARAMETER if N is negative or nil ERR_MPI_NOT_ACCEPTABLE if A has no inverse mod N

Definition at line 1488 of file bignum.c.

References CHK, ERR_MPI_INVALID_PARAMETER, ERR_MPI_NOT_ACCEPTABLE, mpi_add_mpi(), 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(), and mpi::p.

Referenced by main(), and rsa_gen_key().

8.110.2.18 int mpi_is_prime (mpi * X)

Miller-Rabin primality test.

Returns:

0 if successful (probably prime), 1 if memory allocation failed, ERR_MPI_NOT_ACCEPTABLE if X is not prime

Definition at line 1614 of file bignum.c.

References CHK, ERR_MPI_NOT_ACCEPTABLE, 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, and small_prime.

Referenced by main(), and mpi_gen_prime().

8.110.2.19 int mpi_lsb (mpi * X)

Return the number of least significant bits.

Definition at line 458 of file bignum.c.

References biL, int, mpi::n, and mpi::p.

Referenced by mpi_gcd(), and mpi_is_prime().

8.110.2.20 int mpi_lset (mpi * X, int z)

Set value from integer.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 154 of file bignum.c.

References CHK, ciL, mpi_grow(), mpi::n, mpi::p, and mpi::s.

Referenced by 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().

8.110.2.21 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, ERR_MPI_DIVISION_BY_ZERO if $b == 0$,

Definition at line 1153 of file bignum.c.

References biH, ERR_MPI_DIVISION_BY_ZERO, mpi::n, and mpi::p.

Referenced by mpi_is_prime(), and mpi_write_hlp().

8.110.2.22 int mpi_mod_mpi (mpi * R, mpi * A, mpi * B)

Modulo: $R = A \bmod B$.

Returns:

0 if successful, 1 if memory allocation failed, ERR_MPI_DIVISION_BY_ZERO if $B == 0$

Definition at line 1133 of file bignum.c.

References CHK, mpi_add_mpi(), 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_gen_key(), and rsa_private().

8.110.2.23 static void mpi_montg_init (t_int * *mm*, mpi * *N*) [static]

Definition at line 1194 of file bignum.c.

References biL, and mpi::p.

Referenced by mpi_exp_mod().

8.110.2.24 static void mpi_montmul (mpi * *A*, mpi * *B*, mpi * *N*, t_int *mm*, mpi * *T*) [static]

Definition at line 1212 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().

8.110.2.25 static void mpi_montred (mpi * *A*, mpi * *N*, t_int *mm*, mpi * *T*) [static]

Definition at line 1249 of file bignum.c.

References mpi_montmul(), mpi::n, mpi::p, and mpi::s.

Referenced by mpi_exp_mod().

8.110.2.26 int mpi_msb (mpi * *X*)

Return the total size in bits, without leading 0s.

Definition at line 440 of file bignum.c.

References biL, mpi::n, and mpi::p.

Referenced by dhm_calc_secret(), dhm_make_params(), dhm_read_params(), mpi_div_mpi(), mpi_exp_mod(), mpi_gen_prime(), mpi_is_prime(), mpi_shift_l(), mpi_write_binary(), mpi_write_string(), rsa_check_pubkey(), rsa_gen_key(), rsa_read_private(), rsa_read_public(), x509_add_certs(), and x509_parse_key().

8.110.2.27 static void mpi_mul_hlp (int *i*, t_int * *s*, t_int * *d*, t_int *b*) [static]

Definition at line 838 of file bignum.c.

References MULADDC_CORE, MULADDC_INIT, and MULADDC_STOP.

Referenced by mpi_montmul(), and mpi_mul_mpi().

8.110.2.28 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 937 of file bignum.c.

References mpi_mul_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by mpi_div_mpi(), and mpi_read_string().

8.110.2.29 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 902 of file bignum.c.

References 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(), rsa_check_privkey(), rsa_gen_key(), and rsa_private().

8.110.2.30 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 395 of file bignum.c.

References CHARS_TO_LIMBS, CHK, ciL, mpi_grow(), mpi_lset(), and mpi::p.

Referenced by asn1_get_mpi(), dhm_read_bignum(), dhm_read_public(), rsa_private(), and rsa_public().

8.110.2.31 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 ERR_MPI_XXX error code

Definition at line 332 of file bignum.c.

References ERR_MPI_FILE_IO_ERROR, mpi_get_digit(), and mpi_read_string().

Referenced by main(), rsa_read_private(), and rsa_read_public().

8.110.2.32 `int mpi_read_string (mpi * X, int radix, char * s)`

Import X from an ASCII string.

Parameters:

X destination [mpi](#)
radix input numeric base
s null-terminated string buffer

Returns:

0 if successful, or an ERR_MPI_XXX error code

Definition at line 188 of file bignum.c.

References [BITS_TO_LIMBS](#), [CHK](#), [ciL](#), [ERR_MPI_INVALID_PARAMETER](#), [int](#), [mpi_add_int\(\)](#), [mpi_free\(\)](#), [mpi_get_digit\(\)](#), [mpi_grow\(\)](#), [mpi_init\(\)](#), [mpi_lset\(\)](#), [mpi_mul_int\(\)](#), [mpi::p](#), and [mpi::s](#).

Referenced by [main\(\)](#), and [mpi_read_file\(\)](#).

8.110.2.33 `int mpi_self_test (int verbose)`

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 1912 of file bignum.c.

Referenced by [main\(\)](#).

8.110.2.34 `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 473 of file bignum.c.

References [biL](#), [BITS_TO_LIMBS](#), [CHK](#), [mpi_grow\(\)](#), [mpi_msb\(\)](#), [mpi::n](#), and [mpi::p](#).

Referenced by [mpi_div_mpi\(\)](#), [mpi_exp_mod\(\)](#), [mpi_gcd\(\)](#), and [mpi_gen_prime\(\)](#).

8.110.2.35 `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 522 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()`.

8.110.2.36 `int mpi_sub_abs (mpi * X, mpi * A, mpi * B)`

Unsigned subtraction: $X = |A| - |B|$.

Returns:

0 if successful, `ERR_MPI_NEGATIVE_VALUE` if B is greater than A

Definition at line 708 of file bignum.c.

References `CHK`, `ERR_MPI_NEGATIVE_VALUE`, `mpi_cmp_abs()`, `mpi_copy()`, `mpi_free()`, `mpi_init()`, `mpi_sub_hlp()`, `mpi::n`, and `mpi::p`.

Referenced by `mpi_add_mpi()`, `mpi_gcd()`, and `mpi_sub_mpi()`.

8.110.2.37 `static void mpi_sub_hlp (int n, t_int * s, t_int * d)` `[static]`

Definition at line 687 of file bignum.c.

Referenced by `mpi_montmul()`, and `mpi_sub_abs()`.

8.110.2.38 `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 822 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()`.

8.110.2.39 `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 775 of file bignum.c.

References `CHK`, `mpi_add_abs()`, `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()`.

8.110.2.40 void mpi_swap (mpi * *X*, mpi * *Y*)

Swap the contents of *X* and *Y*.

Definition at line 142 of file bignum.c.

Referenced by rsa_gen_key().

8.110.2.41 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, 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 417 of file bignum.c.

References ciL, ERR_MPI_BUFFER_TOO_SMALL, mpi_msb(), and mpi::p.

Referenced by dhm_calc_secret(), dhm_make_public(), rsa_private(), and rsa_public().

8.110.2.42 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 ERR_MPI_XXX error code

Note:

Set *fout* == NULL to print *X* on the console.

Definition at line 357 of file bignum.c.

References CHK, ERR_MPI_FILE_IO_ERROR, and mpi_write_string().

Referenced by main(), rsa_write_private(), and rsa_write_public().

8.110.2.43 `static int mpi_write_hlp (mpi * X, int radix, char ** p)` `[static]`

Definition at line 245 of file `bignum.c`.

References `CHK`, `ERR_MPI_INVALID_PARAMETER`, `mpi_cmp_int()`, `mpi_div_int()`, and `mpi_mod_int()`.

Referenced by `mpi_write_string()`.

8.110.2.44 `int mpi_write_string (mpi * X, int radix, char * s, int * slen)`

Export *X* 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 `ERR_MPI_XXX` error code

Note:

Call this function with `*slen = 0` to obtain the minimum required buffer size in `*slen`.

Definition at line 270 of file `bignum.c`.

References `CHK`, `ciL`, `ERR_MPI_BUFFER_TOO_SMALL`, `ERR_MPI_INVALID_PARAMETER`, `mpi_copy()`, `mpi_free()`, `mpi_init()`, `mpi_msb()`, `mpi_write_hlp()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_write_file()`.

8.110.3 Variable Documentation**8.110.3.1** `const char _bignum_src[] = "_bignum_src"` `[static]`

Definition at line 1778 of file `bignum.c`.

8.110.3.2 `const int small_prime[]` `[static]`

Definition at line 1579 of file `bignum.c`.

Referenced by `mpi_is_prime()`.

8.111 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/certs.c File Reference

Variables

- char `test_ca_crt` []
- char `test_ca_key` []
- char `test_ca_pwd` [] = "test"
- char `test_srv_crt` []
- char `test_srv_key` []
- char `test_cli_crt` []
- char `test_cli_key` []
- char `xyssl_ca_crt` []
- static const char `_certs_src` [] = "_certs_src"

8.111.1 Variable Documentation

8.111.1.1 `const char _certs_src[] = "_certs_src"` [static]

Definition at line 206 of file certs.c.

8.111.1.2 `char test_ca_crt[]`

Initial value:

```
"-----BEGIN CERTIFICATE-----\r\n"
"MIIDpTCCAo2gAwIBAgIBADANBgkqhkiG9w0BAQUFADBQMswCQYDVQQGEwJGUjEO\r\n"
"MAwGA1UEBxMFUGFyaXNkZjAMBgNVBAoTBVh5U1NMMRYwFAYDVQQDEw1YeVNTTCBU\r\n"
"ZXN0IENBMB4XDTA3MDcwNzA1MDAxOFoXDTE3MDcwNzA1MDAxOFowRTElMAkGA1UE\r\n"
"BhMCRlIxZjAMBgNVBAcTBVBhcm1zMQ4wDAYDVQQKEwVYeVNTTDEWMBQGA1UEAxMN\r\n"
"WH1TU0wgVGZzdCBDQTCASiWdQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAM+k\r\n"
"gt70fIPiYqmvXr+9uPmWoN405eoSzxdiRLCHqL4V/Ts0E/H+JNfHS4DH1AgxrJu\r\n"
"+ZIadvSJuhKI6eliMkAh5SU1DqaF3jrrFdJCooM6a077M4CRkE1tdAeZDf+BYp0q\r\n"
"BeMU9Y2+j7ibsQaPAizunbXLf4QdteExCwhYRJ8OVXSEaSnt339gJzTD6kOhES3b\r\n"
"lEN3qbx6lqFaJ5MLHTix5uNVc2rvbOizV5oLhJNqm52AKOp1ltv6WTiI8loagvAc\r\n"
"jlhEZRNB9el5SL6Jai/uFcqXKzfXNKW3FYFQHFgobmiMfGt1lUSBj3F2mrqEC7gC\r\n"
"whY/FDvAI64/k5LZAFkCAwEAAoBnzCBnDAMBgNVHRMEBTADAQH/MB0GA1UdDgQW\r\n"
"BBS87h+Y6Porg+SkfV7DdXKTMdKyZzBtBgNVHSMEZjBkgBS87h+Y6Porg+SkfV7D\r\n"
"dXKTMdKyZ6FJpEcwRTElMAkGA1UEBhMCRlIxZjAMBgNVBAcTBVBhcm1zMQ4wDAYD\r\n"
"VQQKEwVYeVNTTDEWMBQGA1UEAxMNWH1TU0wgVGZzdCBDQYIBADANBgkqhkiG9w0B\r\n"
"QAQUFAAOCAQEAIHdohONCg6KAhWdSmfEgSbKUI8/Zr/d56uw42H00sj/uKPQzUco\r\n"
"3Mx2BYElm1itg7q5OhrkB7J4ZB78EtNZM84nV+y6od3YpR0Z9VUxCx7948MozYRy\r\n"
"TKF5x/lKHXlPJkFE04clKdWTFAtWtGhewXrHJQ8C+ENh2Up2wTVh3Z+pEzuZNv3\r\n"
"u/JYu1H+vkt3l1WCy/9mxUnu+anW1DzxPwnjy4lX6Mi0BD2qfKBWLjVS+7v6ALcj\r\n"
"S2oRWWr4LUvXT7z9BBAvw2eJQD+a4uAya6EURG7AsAvr5MnWn/r0wLWmBJ6fB1Yp\r\n"
"F1kOmamOfvstLMf74rLX+LGKeJ/nwui5FQ==\r\n"
"-----END CERTIFICATE-----\r\n"
```

Definition at line 21 of file certs.c.

Referenced by `main()`.

8.111.1.3 `char test_ca_key[]`

Definition at line 45 of file certs.c.

8.111.1.4 char test_ca_pwd[] = "test"

Definition at line 77 of file certs.c.

8.111.1.5 char test_cli_crt[]

Initial value:

```

"-----BEGIN CERTIFICATE-----\r\n"
"MIIDPTCCAiwGAwIBAgIBATANBgqhkiG9w0BAQUFADBQMswCQYDVQQGEwJGUjEO\r\n"
"MAwGA1UEBxMFUGFyaXMxMDU1NMMRYwFAYDVQQDEw1YeVNTTCBU\r\n"
"ZXN0IENBMjA4MDcwNzA1MDEyMFOxDTA4MDcwNjA1MDEyMFOwMDELMakGA1UE\r\n"
"BhMCR1IxDjAMBGNVBAoTBVh5U1NMMREwDwYDVQQDEwhKb2UgVXN1cjcCASIwDQYJ\r\n"
"KoZIHvcNAQEBAQADggEPADCCAQoCggEBAAKZkg8AN16kGmLGqKc6KUHb9IsZwb2+K\r\n"
"jBw83Qb0KuvPVnu3MzEcXFvOZ83g0PL/z8ob5PKr8HP6bVYzhsD65imcDEEVPk2\r\n"
"9r0XGTggGjB601Fd8aTShUWE4NLrKw6YNXTXgTndvhHNxXwqmdNVLkmZjj3ZwYUc\r\n"
"CEE8eE5jHs8cMDXJLMCwgKIM7Sax220hSHQHKwifVO4/Fdw5G+Suys8PhMX2jDXM\r\n"
"ICFwq8ld+bZGoNUtgp48FWhAMfJyTEaHh9LC46KkqGSDRIzx7/4cPB6QqrpzJN0o\r\n"
"Kr8kh7vdRDTFdmO23D4C510Bw/2aC76DhEJpB2bGA4iIszJs+F/PIL8CAwEAAaNN\r\n"
"MEswCQYDVROTBAlwADAdBgNVHQ4EFgQUiWX1IvjrDYgt0zz5Sq16x01k0o4wHwYD\r\n"
"VR0jBBgwFoAUvO4fmOj6K4PkpH1ew3VykzHZMmcwDQYJKoZIhvcNAQEFBQADggEB\r\n"
"AGdQd7VThJmC+oeMUHk2TQX2wZNU+GsC+RLjtleckny95Kn1jGvMtCznyLkSSD\r\n"
"faJlKfR1No8pk5grDscggyIuQx5WnHNv4QBZmMsmvDICxZRQaxuPFHbS4aLxldL\r\n"
"yOWm5Z4qkMHPCKvA86blYsEkksGDV47fF9ZkOQ8nkh7Z4eY4/5TwqTY72ww5g4NL\r\n"
"6DZtWpGpGbx99NRANvZc9D+ElxkgHnH4YWafOKBclSgqrutbRLi2uZx/QpvuF+i\r\n"
"sUbe+HFPMMWU51Bv/oOhQkz0VD+HusYtXWS2lG88cT40aNLy2CkYUugdTR/b9Uea\r\n"
"p/i862sL/1040q1Q0xV5N7U=\r\n"
"-----END CERTIFICATE-----\r\n"

```

Definition at line 130 of file certs.c.

Referenced by main().

8.111.1.6 char test_cli_key[]

Initial value:

```

"-----BEGIN RSA PRIVATE KEY-----\r\n"
"MIIEowIBAAKCAQEApmsDwA2XqQaYsaopzopQdv0ixnBvb4qMHDzdBvQq689We7cz\r\n"
"MRxcW85nzeDQ8v/Pyhvk8qvw/ptVjOGwPrmKZwIMQRU+Tb2vRcZOCaAMHrTUV3x\r\n"
"pNKFRTg0usrDpg1dNeBM12+Ec3FfCqZ01UuSZmOPdnBhRxwQTx4TmMezxwNcks\r\n"
"wlCAogztJrHbY6FIaAcrcJ9U7j8V3Dkb5K7Kzw+ExfaMncwJIXCryV35tkag1S2C\r\n"
"njwVaeA8x8nJMRoeH0sLjoqSoZINEjPhv/hw8HpCqunMk3SgqvYQfu91ENMU0Y7bc\r\n"
"PgImXQHD/ZoLvoOEQmkHZSYDiizMmz4X88gvwIDAQABAoIBAEOBnBkAjdVN+j4ax\r\n"
"1dJewZKqXVkmAUXDBDyDrCjxRoWY2gz7YW1ALUMUbeV0fO5v1zVrwbkUKKZeVBxI\r\n"
"QA9zRw28H8A6tfvolHgRiCxdixMh3ePC+DVDJ6zglvKV2ipAwBufKYIrX0r4Io2\r\n"
"ZqUrNg9CeEYNlkHWcean12rhYwO82pgHxnB1p5pI42pY7lzyLgSddf5n+M5UBOJI\r\n"
"gsNckvbGdv7WQPVFTRDiRgEnCJ3rI8oPSK6MOUWJw3rh2hbKx+ex8NPvEKbzEXiU\r\n"
"p5jlAlbHIWP5sYBbAlYviFtryAV4fyfLcWPfoqa33Oozofjlwoj0Aixz+6rerLjZ\r\n"
"cpTSrAECgYEA2oCffUo6HH3Lq9oeWhFCOoYg3YjZmFrJaJwjHnvroX9/pXhQYKog\r\n"
"TeHcjUJbtFZw0klcetYbZCFqT8v9nf0uPlgaiVGctXf1MSbFXDUFKkYBiFwzWMT\r\n"
"Ysmvff82jMWZ8ecsXTyDR1858SR5WPZ52qEsCc5X2un7QENm6FtVT8CgYEAwvKS\r\n"
"zQNzuoJETqZX7AalmK3JM8Fdam+Qm5LNMcbvbkKwI8HKMS1VMuqaR0XdAX/iMXAx\r\n"
"PlVhSsmoSDbsMpxBEZlptpCen/GcqcITxANTakrBHxqb2aQ5EEu7SzgfHZWse3/\r\n"
"vQEYfcfTBlPIdcZUDzk4/w7WmyivpYtCWoAh1IECgYEA0UYZ+1UJfVpapRj+swMP\r\n"
"DrQbo7i7t71UaFYLKnpFX2OPLTWc5txqn1OruTu5VHDqE+5hneDNUUTT3uOg4B2q\r\n"
"mdmmanjh2M6wz0e0BVfexhNQynqMaqTe32IOM8DFs3L0xaccg7JfVn6P7CeQGOv\r\n"
"wc96kICw6ZxhtJSqpOGipt8CgYBI/OPw+IXxJK4nNSpe+u4vCYP5mUI9hKEFYCbt\r\n"
"qKwvyAUknn/zqiQ+R/iSErFMPmlwXjvWi0gL/qPb+FP4hCLX8u2zNhY08Pp4Gin\r\n"
"Ej+pAntWxq+kHyfKEI5dyRwV/snfv1qwjy404JSf3VMhIMdYDPzbb72Qnni5w51\r\n"
"j00eAQKBgBqt9jJmDlJdpemC2dm0BuuDIz2h3/MH+CMjfaDLenVpKykn17B6N92h\r\n"

```

```
"klMesqK3RQzDGwauDw431LQw0R69onn9fCM3wJw2yEC6wC9sF8I8hsNZbt64yZhZ\r\n"
"4Bi2YRTiHhpEuBqKlhHLDHFHneo3SMYh8PU/PDQQcyWGHUUi9z1RE\r\n"
"-----END RSA PRIVATE KEY-----\r\n"
```

Definition at line 152 of file certs.c.

Referenced by main().

8.111.1.7 char test_srv_crt[]

Initial value:

```
"-----BEGIN CERTIFICATE-----\r\n"
"MIIDPjCCAiaGAwIBAgIBA jANBgkqhkiG9w0BAQUFADBFMQswCQYDVQQGEwJGU jEO\r\n"
"MAwGA1UEBxMFUGFyaXNzA jAMBgNVBAoTBVh5U1NMMRYwFAYDVQQDEw1YeVNTTCBU\r\n"
"ZXN0IENBMB4XDTA3MDcwNzA1MDEyOVoxDTA4MDcwNjA1MDEyOVowMTELMakGA1UE\r\n"
"BhMCRlIXD jAMBgNVBAoTBVh5U1NMMRIwEAYDVQQDEw1sb2NhbgHvc3QwgGFiMA0G\r\n"
"CSqGSIB3DQEBAAQUAA4IBDwAwggEKAoIBAQC40PDcGTgmHkt6noXDfk jVuymjiNYB\r\n"
"gjtiL7uA1Ke3tXStacEecQek/OJxYqYr7ffcWals29LL6HbKpi0xLZKBbD9ACKdH\r\n"
"1Z/SvHlyQPILJdYb9DMw+kzZds5myXUjzn7Aem1YjoxMZUAMyc34i2900X2pL0v2\r\n"
"SfCeJ9Ym4MOnZxYl217+dX9ZbkGIGrT6uY2IYK4boDwxbTcyT8i/NPsvsiMwtWPM\r\n"
"rnQMr+XbgS98sUzcZE70Pe1TlV9Iy8j/8d2OiFo+qTyMu/6UpM2s3gdkQkMzx+Sm\r\n"
"4QitRU jzmEXeUePRU jEGHiv7vz069xuvBzrks36w5BXiVAhLke/OTKVPagMBAAGj\r\n"
"TTBLMAkGA1UdEwQCAAwHQYDVROBBYEFNkOyCTx64SDdPySGWl/tzD7/WMSMB8G\r\n"
"A1UdIwQYMBAAFLzuH5jo+iuD5KR9XsN1cpMx2TJnMA0GCSqGSIB3DQEBBQUAA4IB\r\n"
"AQBelJv5t+suaqy5Lo5bjNeH jNZfgg8EigDQ7Nqaosv1QZAsh2N34Gg5YdkGyVdg\r\n"
"s32I/K5aaywyUbG9qVXQxCM2T95qBqyK56h9yJoZKWQD9H//+zB8kCK/16WvRfv3\r\n"
"VA7eSR19qQFWLHe+1qGh2YhxeDUfyi+fm4D36dGxqc2A34tzjo0QPHKtIeqM0kJy\r\n"
"zzL65T1bJQKkyTurHofFv0jW9ZFG2wkGysVgCY5fjuLiLdo/sWUaXd2987iNFa+K\r\n"
"FrHsTi6urSfZuG1ZnXDXDHEE7Q2snAvvev+KR7DD9X4DJGcPX9ga4CGJj+9ZzyAA\r\n"
"ZTGpOzk1hIH44RFs21JmZR1E\r\n"
"-----END CERTIFICATE-----\r\n"
```

Definition at line 79 of file certs.c.

Referenced by main().

8.111.1.8 char test_srv_key[]

Initial value:

```
"-----BEGIN RSA PRIVATE KEY-----\r\n"
"MIIEowIBAAKCAQEAuNDw3Bk4Jh5Lep6Fw35I1lbsp04jWAYI7Yi+7gNSnt7V0rWnB\r\n"
"HnEHpPzicwKmk+333FmpUtvSy+h2yqYtMS2SgWw/QApA4dWf0rx5ckDyCyXWG/Qz\r\n"
"MPpM2XbOZs11I85+wHptWI6MTGVADmnN+ItvdNF9qS9L9knwni fWJuDDp2cWJdte\r\n"
"/nV/WW5ICIK0+rmNiGcuG6A8MW03Mk/IvzT7FbI jMLVjzK50DK/124EvfLFM3GRO\r\n"
"9D3tU5VfSMvI//HdjohaPqk8 jLv+1KTNrN4HZEJDM8fkpuEIrUVI85hf31Hj0ViX\r\n"
"IByL+789OvcblQc65LN+sOQV41QIS5HvzkylTwIDAQABAoIBABeah8h0aBlmMRmd\r\n"
"+vN4Y3D4kf7UcRCMQ21Mz1Oq1Si/QgGLyiBLK0DFE16LzNE7eTZpNRjh/lAQhmtn\r\n"
"QcpQGa/x1TomlRbCo8DUVWZkKQWHdYroa0lMDliPt dimzhEepE2M1T5EJmLzY3S+\r\n"
"qVge7UMsJjJfWgJAezyXteANQK+2YSt+CjPIqIHch1KexUnvdN9++1oEx6AbuZ8T\r\n"
"4avhFYZQP15tZNGsk2LfQ1YS/NfbowkCsd0/TVubJBmDGUML/E5MbxjxLz1aNB2M\r\n"
"V59cBNsgsA35CODAUF4xOyoSfZGg1Rb9qQrv1E6Jz56dG8SsKF3HqnDjxiPOVBN\r\n"
"FBnVJ+ECgYEA29MhAsKMM4XqBUPk6pIMFTgm/s1E5vxiG70vqiL+guvBhhQ7zs1\r\n"
"8UMTNXZoMELNoB/ev9fN0CjclVr46b/x/yDw7wMb96i+vzEN0zu4RHwi3OWpCPbp\r\n"
"qBKEi3hzN8M+BulPX8CDQx3aLRRfxw51J5EuA0NeybngbItgxTi0u6kCgYEA1zr0\r\n"
"6P5YdOhYHTSWDlkeD49MApcVuzaHnsHZVAhUqu3RwiY9LRAJLzfr7fQDb9DjBz\r\n"
"sxTRLG6LSAcSR7mw+m+GvNqGt/9pSqbtW+L/VwVWSyF+YYklxZUD3UAAyrdVcDEC\r\n"
"a5S+ja4dCsi/1VHT5ulWiCkWL1fJvadmSubKNDCcGYA+71xVGPP+1sFgTiytfrc8\r\n"
"5n2rl4MxinJ9+w0I+EbZCKNMYGvTgiU4dJasSMEdiBKs1FMGo7df8F0BLHF1IsIa\r\n"
"5Ah2tXtXn9154o90iTXMMK6qmRaneM6fhOoeaCwYAhpGxYIpqx/Xr4TOhiag46\r\n"
"-----"
```

```
"jMMaphAeOvw4t1K2RDziOQKBgQCYPCCU0gxuw/o1jda2CxbZy9EmU/erEX09+0n+\r\n"
"TOFQpSEFq/z9WaxAFY9LfsdZ0ZktoeHmalbNdL3i6A3DWAM3YSQzQMRPmzOWnqXx\r\n"
"cgoCBmlvzkzaeLjO5phMoLQHJmmafvuCG6uxov3F8Hi3LyHUF2c8k0nL6ucmJ3vj\r\n"
"uzu4AQKBgBSASMAJS63M9UJB1Eazy2v2NWw04CmzNxUfWrHuKpd/C2ik4QKu0sRO\r\n"
"r9KnkDgxxEhjDm7lXhlW12PU42yORst5I3Eaa1Cfi4KPFn/ozt+iNBYrzd8Tyvnb\r\n"
"qkdEC10+G2Fo/ER4NRCv7a24WNEsOMGzGRqw5cnSJrjbZLYMaIyK\r\n"
"-----END RSA PRIVATE KEY-----\r\n"
```

Definition at line 101 of file certs.c.

Referenced by main().

8.111.1.9 char xyssl_ca_crt[]

Initial value:

```
"-----BEGIN CERTIFICATE-----\r\n"
"MIID4DCCAsigAwIBAgIJAOLw9BMV1jxMMA0GCSqGSIb3DQEBBQUAMFMxCzAJBgNV\r\n"
"BAYTAKZSMQ4wDAYDVQQIEwVQYXJpczEOMAwGA1UEChMFWH1TU0wxJDAiBgNVBAMT\r\n"
"G1h5U1NMIENlcnRpZmljYXRlIEF1dGhvcm10eTAeFw0wNjEwMzEyMjU5MjRaFw0x\r\n"
"NjEwMzEyMjU5MjRaMFEMxCzAJBgNVBAYTAKZSMQ4wDAYDVQQIEwVQYXJpczEOMAwG\r\n"
"A1UEChMFWH1TU0wxJDAiBgNVBAMTG1h5U1NMIENlcnRpZmljYXRlIEF1dGhvcm10\r\n"
"eTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAKnprf1RQ7IYPI3FmI/h\r\n"
"f2EJGfaIP+Jt551VZWFrS3A56Nn4KS57zTAKZUA7YGBLwLAznfaphJ7SvENALeZR\r\n"
"/J0c/n9jwMpFXReQL7RVpgg/zlR+t2DUi3DAwigPZiHHCSJSBC73vpMc6uH0eV2d\r\n"
"itqjjUnJG5F8Zg9/gX4UMRAdlwGWqxvs+jc9i0XFKEZga8+rONe6WvKyBM4e20I\r\n"
"HcO4BPF92d6sm4qLgyR4oXUkBz6NfDWX8ZdTvXuRaK9qMy1327cCT48sis9F6/eK\r\n"
"QAxx0VHlqGTtxDYjHJPsmLNejuyUDvsaC8TMCSpFTXpMTUvJdmxsF1LZWmIo5lY5\r\n"
"zh0CAwEAaAObtjCBsZAdBgNVHQ4EFgQUb1VzoNJrkkGPKJ8xjuFvVP2E4GswgYMG\r\n"
"A1UdIwR8MHQAFAZVc6DSa5JID5CfMY7hb1T9hOBroVekVTBTMQswCQYDVQQGEwJG\r\n"
"UjEOMAwGA1UEChMFUjEOMAwGA1UEChMFUjEOMAwGA1UEChMFUjEOMAwGA1UEChMF\r\n"
"TCBDZXJ0aWZpY2F0ZSBDbXR0b3JpdHmCCQDi8PQTFdY8TDAMBGNVHRMEBTADAQH\r\n"
"MA0GCSqGSIb3DQEBBQUAA4IBAQCgd65b215BASfsPv1rcRnLzu/99eWTVAJwJbbD\r\n"
"VhPAQiET0W4U/85EDK7uoFo/SEjyMB/m4T20A8FIDaK7jBPo/1gtbuQjGMR17h+z\r\n"
"F2iGuNhZ6T2d6Uzqc1t3oiFtSvDRoZ/9kqkEy7Lrs7FBzOmvfTvrqvADf7cLMA2D\r\n"
"ri/otDpzPr4XoDnwd4C+4bQC/Gr3Uder4VAeTOJtKdGqfYLVpWpSPVBDuVLUybKi\r\n"
"8cMAT6p9IG1e12u6vFqcBT/I67Q0bGU6gzVVz9ZVULXOYZMjjLafVXC1gesUH2WT\r\n"
"gTEAnEBkSRrkfAi+RezoEFAbmEl3fPt09dwSPku3x7cB3zaJ\r\n"
"-----END CERTIFICATE-----\r\n"
```

Definition at line 181 of file certs.c.

Referenced by main().

8.112 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/des.c File Reference

```
#include <string.h>
#include "xyssl/des.h"
```

Defines

- #define [_CRT_SECURE_NO_DEPRECATED](#) 1
- #define [GET_UINT32_BE](#)(n, b, i)
- #define [PUT_UINT32_BE](#)(n, b, i)
- #define [DES_IP](#)(X, Y)
- #define [DES_FP](#)(X, Y)
- #define [DES_ROUND](#)(X, Y)

Functions

- static void [des_main_k](#)s (unsigned long SK[32], unsigned char key[8])
- void [des_set_key](#) ([des_context](#) *ctx, unsigned char key[8])
DES key schedule (56-bit).
- static void [des_crypt](#) (unsigned long SK[32], unsigned char input[8], unsigned char output[8])
- void [des_encrypt](#) ([des_context](#) *ctx, unsigned char input[8], unsigned char output[8])
DES block encryption (ECB mode).
- void [des_decrypt](#) ([des_context](#) *ctx, unsigned char input[8], unsigned char output[8])
DES block decryption (ECB mode).
- void [des_cbc_encrypt](#) ([des_context](#) *ctx, unsigned char iv[8], unsigned char *input, unsigned char *output, int len)
DES-CBC buffer encryption.
- void [des_cbc_decrypt](#) ([des_context](#) *ctx, unsigned char iv[8], unsigned char *input, unsigned char *output, int len)
DES-CBC buffer decryption.
- void [des3_set_2keys](#) ([des3_context](#) *ctx, unsigned char key[16])
Triple-DES key schedule (112-bit).
- void [des3_set_3keys](#) ([des3_context](#) *ctx, unsigned char key[24])
Triple-DES key schedule (168-bit).
- static void [des3_crypt](#) (unsigned long SK[96], unsigned char input[8], unsigned char output[8])
- void [des3_encrypt](#) ([des3_context](#) *ctx, unsigned char input[8], unsigned char output[8])
Triple-DES block encryption (ECB mode).
- void [des3_decrypt](#) ([des3_context](#) *ctx, unsigned char input[8], unsigned char output[8])
Triple-DES block decryption (ECB mode).

- void `des3_cbc_encrypt` (`des3_context` *ctx, unsigned char iv[8], unsigned char *input, unsigned char *output, int len)
3DES-CBC buffer encryption
- void `des3_cbc_decrypt` (`des3_context` *ctx, unsigned char iv[8], unsigned char *input, unsigned char *output, int len)
3DES-CBC buffer 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 char `_des_src` [] = "_des_src"

8.112.1 Define Documentation

8.112.1.1 #define _CRT_SECURE_NO_DEPRECATED 1

Definition at line 28 of file des.c.

8.112.1.2 #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) & 0xF0F0F0F; Y ^= T; X ^= (T << 4); \
}
```

Definition at line 256 of file des.c.

Referenced by `des3_crypt()`, and `des_crypt()`.

8.112.1.3 #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 242 of file des.c.

Referenced by des3_crypt(), and des_crypt().

8.112.1.4 #define DES_ROUND(X, Y)**Value:**

```

{
    T = *SK++ ^ X; \
    Y ^= SB8[ (T >> 8) & 0x3F ] ^ \
    SB6[ (T >> 16) & 0x3F ] ^ \
    SB4[ (T >> 24) & 0x3F ] ^ \
    SB2[ (T >> 31) & 0x3F ]; \
    T = *SK++ ^ ((X << 28) | (X >> 4)); \
    Y ^= SB7[ (T >> 8) & 0x3F ] ^ \
    SB5[ (T >> 16) & 0x3F ] ^ \
    SB3[ (T >> 24) & 0x3F ] ^ \
    SB1[ (T >> 31) & 0x3F ]; \
}

```

Definition at line 270 of file des.c.

Referenced by des3_crypt(), and des_crypt().

8.112.1.5 #define GET_UINT32_BE(n, b, i)**Value:**

```

{
    (n) = ( (unsigned long) (b)[(i) << 4] << 24 ) \
    | ( (unsigned long) (b)[(i) << 4 + 1] << 16 ) \
    | ( (unsigned long) (b)[(i) << 4 + 2] << 8 ) \
    | ( (unsigned long) (b)[(i) << 4 + 3] ); \
}

```

Definition at line 39 of file des.c.

8.112.1.6 #define PUT_UINT32_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 48 of file des.c.

8.112.2 Function Documentation

8.112.2.1 void des3_cbc_decrypt (des3_context * *ctx*, unsigned char *iv*[8], unsigned char * *input*, unsigned char * *output*, int *len*)

3DES-CBC buffer decryption

Parameters:

ctx 3DES context
iv initialization vector (modified after use)
input buffer holding the ciphertext
output buffer holding the plaintext
len length of the data to be decrypted

Definition at line 611 of file des.c.

References des3_crypt(), and des3_context::dsk.

Referenced by ssl_decrypt_buf(), and x509_des3_decrypt().

8.112.2.2 void des3_cbc_encrypt (des3_context * *ctx*, unsigned char *iv*[8], unsigned char * *input*, unsigned char * *output*, int *len*)

3DES-CBC buffer encryption

Parameters:

ctx 3DES context
iv initialization vector (modified after use)
input buffer holding the plaintext
output buffer holding the ciphertext
len length of the data to be encrypted

Definition at line 586 of file des.c.

References des3_crypt(), and des3_context::esk.

Referenced by main(), and ssl_encrypt_buf().

8.112.2.3 static void des3_crypt (unsigned long *SK*[96], unsigned char *input*[8], unsigned char *output*[8]) [static]

Definition at line 519 of file des.c.

References DES_FP, DES_IP, DES_ROUND, GET_UINT32_BE, and PUT_UINT32_BE.

Referenced by des3_cbc_decrypt(), des3_cbc_encrypt(), des3_decrypt(), and des3_encrypt().

8.112.2.4 void des3_decrypt (des3_context * *ctx*, unsigned char *input*[8], unsigned char *output*[8])

Triple-DES block decryption (ECB mode).

Parameters:

ctx 3DES context

input ciphertext block

output plaintext block

Definition at line 576 of file des.c.

References des3_crypt(), and des3_context::dsk.

8.112.2.5 void des3_encrypt (des3_context * *ctx*, unsigned char *input*[8], unsigned char *output*[8])

Triple-DES block encryption (ECB mode).

Parameters:

ctx 3DES context

input plaintext block

output ciphertext block

Definition at line 566 of file des.c.

References des3_crypt(), and des3_context::esk.

8.112.2.6 void des3_set_2keys (des3_context * *ctx*, unsigned char *key*[16])

Triple-DES key schedule (112-bit).

Parameters:

ctx 3DES context to be initialized

key 16-byte secret key

Definition at line 472 of file des.c.

References des_main_ks(), des3_context::dsk, and des3_context::esk.

8.112.2.7 void des3_set_3keys (des3_context * ctx, unsigned char key[24])

Triple-DES key schedule (168-bit).

Parameters:

ctx 3DES context to be initialized

key 24-byte secret key

Definition at line 498 of file des.c.

References des_main_ks(), des3_context::dsk, and des3_context::esk.

Referenced by main(), ssl_derive_keys(), and x509_des3_decrypt().

8.112.2.8 void des_cbc_decrypt (des_context * ctx, unsigned char iv[8], unsigned char * input, unsigned char * output, int len)

DES-CBC buffer decryption.

Parameters:

ctx DES context

iv initialization vector (modified after use)

input buffer holding the ciphertext

output buffer holding the plaintext

len length of the data to be decrypted

Definition at line 444 of file des.c.

References des_crypt(), and des_context::dsk.

8.112.2.9 void des_cbc_encrypt (des_context * ctx, unsigned char iv[8], unsigned char * input, unsigned char * output, int len)

DES-CBC buffer encryption.

Parameters:

ctx DES context

iv initialization vector (modified after use)

input buffer holding the plaintext

output buffer holding the ciphertext

len length of the data to be encrypted

Definition at line 419 of file des.c.

References des_crypt(), and des_context::esk.

Referenced by main().

8.112.2.10 static void des_crypt (unsigned long SK[32], unsigned char *input*[8], unsigned char *output*[8]) *[static]*

Definition at line 370 of file des.c.

References DES_FP, DES_IP, DES_ROUND, GET_UINT32_BE, and PUT_UINT32_BE.

Referenced by des_cbc_decrypt(), des_cbc_encrypt(), des_decrypt(), and des_encrypt().

8.112.2.11 void des_decrypt (des_context * *ctx*, unsigned char *input*[8], unsigned char *output*[8])

DES block decryption (ECB mode).

Parameters:

ctx DES context

input ciphertext block

output plaintext block

Definition at line 409 of file des.c.

References des_crypt(), and des_context::dsk.

8.112.2.12 void des_encrypt (des_context * *ctx*, unsigned char *input*[8], unsigned char *output*[8])

DES block encryption (ECB mode).

Parameters:

ctx DES context

input plaintext block

output ciphertext block

Definition at line 399 of file des.c.

References des_crypt(), and des_context::esk.

8.112.2.13 static void des_main_ks (unsigned long SK[32], unsigned char *key*[8]) *[static]*

Definition at line 285 of file des.c.

References GET_UINT32_BE, LHs, and RHs.

Referenced by des3_set_2keys(), des3_set_3keys(), and des_set_key().

8.112.2.14 int des_self_test (int *verbose*)

Definition at line 733 of file des.c.

Referenced by main().

8.112.2.15 void des_set_key (des_context * ctx, unsigned char key[8])

DES key schedule (56-bit).

Parameters:

ctx DES context to be initialized
key 8-byte secret key

Definition at line 357 of file des.c.

References des_main_ks(), des_context::dsk, and des_context::esk.

Referenced by main().

8.112.3 Variable Documentation

8.112.3.1 const char _des_src[] = "_des_src" [static]

Definition at line 636 of file des.c.

8.112.3.2 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 223 of file des.c.

Referenced by des_main_ks().

8.112.3.3 const unsigned long RHs[16] [static]

Initial value:

```
{
    0x00000000, 0x01000000, 0x00010000, 0x01010000,
    0x00000100, 0x01000100, 0x00010100, 0x01010100,
    0x00000001, 0x01000001, 0x00010001, 0x01010001,
    0x00000101, 0x01000101, 0x00010101, 0x01010101,
}
```

Definition at line 231 of file des.c.

Referenced by des_main_ks().

8.112.3.4 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 60 of file des.c.

8.112.3.5 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 80 of file des.c.

8.112.3.6 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 100 of file des.c.

8.112.3.7 `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 120 of file des.c.

8.112.3.8 `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 140 of file des.c.

8.112.3.9 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 160 of file des.c.

8.112.3.10 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 180 of file des.c.

8.112.3.11 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 200 of file des.c.

8.113 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/dhm.c File Reference

```
#include <string.h>
#include "xyssl/dhm.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`
- `#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(*rng_f)(void *)`, void *rng_d, unsigned char *output, `int *olen`)
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`, unsigned char *output, `int olen`, `int(*rng_f)(void *)`, void *rng_d)
Create 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.

Variables

- static const char `_dhm_src` [] = "_dhm_src"

8.113.1 Define Documentation

8.113.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 27 of file dhm.c.

8.113.1.2 #define DHM_MPI_EXPORT(X, n)

Value:

```
CHK( mpi_write_binary( X, p + 2, &n ) );    \
    *p++ = ( n >> 8 ); *p++ = n; p += n;
```

Referenced by `dhm_make_params()`.

8.113.2 Function Documentation

8.113.2.1 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 `ERR_DHM_MAKE_PUBLIC_FAILED`

Definition at line 203 of file `dhm.c`.

References `CHK`, `ERR_DHM_BAD_INPUT_DATA`, `ERR_DHM_CALC_SECRET_FAILED`, `dhm_context::GY`, `dhm_context::K`, `mpi_exp_mod()`, `mpi_msb()`, `mpi_write_binary()`, `dhm_context::P`, `dhm_context::RP`, and `dhm_context::X`.

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

8.113.2.2 void dhm_free (dhm_context * ctx)

Definition at line 229 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()`.

8.113.2.3 int dhm_make_params (dhm_context * ctx, int(*) (void *) rng_f, void * rng_d, unsigned char * output, int * olen)

Setup and write the `ServerKeyExchange` parameters.

Parameters:

ctx DHM context
rng_f points to the RNG function
rng_d points to the RNG data
output destination buffer

olen number of chars written

Note:

This function assumes that `ctx->P` and `ctx->G` have already been properly set (for example using `mpi_read_string`).

Returns:

0 if successful, or an MPI error code

Definition at line 93 of file `dhm.c`.

References `CHK`, `DHM_MPI_EXPORT`, `ERR_DHM_MAKE_PARAMS_FAILED`, `dhm_context::G`, `dhm_context::GX`, `dhm_context::len`, `mpi_cmp_mpi()`, `mpi_exp_mod()`, `mpi_grow()`, `mpi_msb()`, `mpi_shift_r()`, `dhm_context::P`, `mpi::p`, `dhm_context::RP`, and `dhm_context::X`.

Referenced by `main()`, and `ssl_write_server_key_exchange()`.

8.113.2.4 `int dhm_make_public(dhm_context * ctx, unsigned char * output, int olen, int(*) (void *) rng_f, void * rng_d)`

Create private value X and export G^X .

Parameters:

ctx DHM context

output destination buffer

olen must be == `ctx->P.len`

rng_f points to the RNG function

rng_d points to the RNG data

Returns:

0 if successful, or `ERR_DHM_MAKE_PUBLIC_FAILED`

Definition at line 164 of file `dhm.c`.

References `CHK`, `ERR_DHM_BAD_INPUT_DATA`, `ERR_DHM_MAKE_PUBLIC_FAILED`, `dhm_context::G`, `dhm_context::GX`, `dhm_context::len`, `mpi_cmp_mpi()`, `mpi_exp_mod()`, `mpi_grow()`, `mpi_shift_r()`, `mpi_write_binary()`, `dhm_context::P`, `mpi::p`, `dhm_context::RP`, and `dhm_context::X`.

Referenced by `main()`, and `ssl_write_client_key_exchange()`.

8.113.2.5 `static int dhm_read_bignum(mpi * X, unsigned char ** p, unsigned char * end)` [static]

Definition at line 37 of file `dhm.c`.

References `ERR_DHM_BAD_INPUT_DATA`, `ERR_DHM_READ_PARAMS_FAILED`, and `mpi_read_binary()`.

Referenced by `dhm_read_params()`.

8.113.2.6 `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 ERR_DHM_READ_PARAMS_FAILED

Definition at line 63 of file dhm.c.

References `dhm_read_bignum()`, `ERR_DHM_BAD_INPUT_DATA`, `dhm_context::G`, `dhm_context::GY`, `dhm_context::len`, `mpi_msb()`, and `dhm_context::P`.

Referenced by `main()`, and `ssl_parse_server_key_exchange()`.

8.113.2.7 `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 ERR_DHM_READ_PUBLIC_FAILED

Definition at line 147 of file dhm.c.

References `ERR_DHM_BAD_INPUT_DATA`, `ERR_DHM_READ_PUBLIC_FAILED`, `dhm_context::GY`, `dhm_context::len`, and `mpi_read_binary()`.

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

8.113.2.8 `int dhm_self_test (int verbose)`

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 243 of file dhm.c.

8.113.3 Variable Documentation

8.113.3.1 `const char _dhm_src[] = "_dhm_src" [static]`

Definition at line 236 of file dhm.c.

8.114 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/dsa.c File Reference

```
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
#include "xyssl/rsa.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- `int rsa_gen_key (rsa_context *ctx, int nbits, int exponent, int(*rng_f)(void *), void *rng_d)`
Generate an RSA keypair.
- `int rsa_public (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int olen)`
Perform an RSA public key operation.
- `int rsa_private (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int olen)`
Perform an RSA private key operation.
- `int rsa_check_pubkey (rsa_context *ctx)`
Return 0 if the public key is valid, or ERR_RSA_KEY_CHECK_FAILED.
- `int rsa_check_privkey (rsa_context *ctx)`
Return 0 if the private key is valid, or ERR_RSA_KEY_CHECK_FAILED.
- `int rsa_pkcs1_encrypt (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int olen)`
Add the PKCS#1 v1.5 padding and do a public RSA.
- `int rsa_pkcs1_decrypt (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int *olen)`
Do a private RSA, removes the PKCS#1 v1.5 padding.
- `int rsa_pkcs1_sign (rsa_context *ctx, int alg_id, unsigned char *hash, int hashlen, unsigned char *sig, int siglen)`
Perform a private RSA to sign a message digest.
- `int rsa_pkcs1_verify (rsa_context *ctx, int alg_id, unsigned char *hash, int hashlen, unsigned char *sig, int siglen)`
Perform 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.

8.114.1 Define Documentation

8.114.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 28 of file dsa.c.

8.114.2 Function Documentation

8.114.2.1 `int rsa_check_privkey (rsa_context * ctx)`

Return 0 if the private key is valid, or `ERR_RSA_KEY_CHECK_FAILED`.

Definition at line 228 of file dsa.c.

References `CHK`, `rsa_context::E`, `ERR_RSA_KEY_CHK_FAILED`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_free()`, `mpi_gcd()`, `mpi_init()`, `mpi_mul_mpi()`, `mpi_sub_int()`, `rsa_context::N`, `rsa_context::P`, and `rsa_context::Q`.

8.114.2.2 `int rsa_check_pubkey (rsa_context * ctx)`

Return 0 if the public key is valid, or `ERR_RSA_KEY_CHECK_FAILED`.

Definition at line 208 of file dsa.c.

References `rsa_context::E`, `ERR_RSA_KEY_CHK_FAILED`, `mpi_msb()`, `rsa_context::N`, and `mpi::p`.

8.114.2.3 `void rsa_free (rsa_context * ctx)`

Free the components of an RSA key.

Definition at line 471 of file dsa.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`.

8.114.2.4 `int rsa_gen_key (rsa_context * ctx, int nbits, int exponent, int(*) (void *) rng_f, void * rng_d)`

Generate an RSA keypair.

Parameters:

ctx RSA context to be initialized

nbits size of the public key in bits

exponent public exponent (e.g., 65537)

rng_f points to the RNG function

rng_d points to the RNG data

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 41 of file dsa.c.

References CHK, rsa_context::D, rsa_context::DP, rsa_context::DQ, rsa_context::E, ERR_RSA_BAD_INPUT_DATA, ERR_RSA_KEY_GEN_FAILED, rsa_context::len, 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::Q, rsa_context::QP, and rsa_free().

8.114.2.5 **int rsa_pkcs1_decrypt** (rsa_context * *ctx*, unsigned char * *input*, int *ilen*, unsigned char * *output*, int * *olen*)

Do a private RSA, removes the PKCS#1 v1.5 padding.

Parameters:

ctx RSA context

input buffer holding the encrypted data

ilen must be the same as the modulus size

output buffer that will hold the plaintext

olen size of output buffer, will be updated to contain the length of the plaintext

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 287 of file dsa.c.

References ERR_RSA_BAD_INPUT_DATA, ERR_RSA_INVALID_PADDING, int, rsa_context::len, RSA_CRYPT, and rsa_private().

8.114.2.6 **int rsa_pkcs1_encrypt** (rsa_context * *ctx*, unsigned char * *input*, int *ilen*, unsigned char * *output*, int * *olen*)

Add the PKCS#1 v1.5 padding and do a public RSA.

Parameters:

ctx RSA context

input buffer holding the data to be encrypted

ilen length of the plaintext; cannot be longer than the modulus, minus 3+8 for padding

output buffer that will hold the ciphertext

olen must be the same as the modulus size (for example, 128 if RSA-1024 is used)

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 257 of file dsa.c.

References ERR_RSA_BAD_INPUT_DATA, rsa_context::len, RSA_CRYPT, and rsa_public().

8.114.2.7 int rsa_pkcs1_sign (rsa_context * ctx, int alg_id, unsigned char * hash, int hashlen, unsigned char * sig, int siglen)

Perform a private RSA to sign a message digest.

Parameters:

ctx RSA context

alg_id RSA_RAW, RSA_MD2/4/5 or RSA_SHA1

hash buffer holding the message digest

hashlen message digest length

sig buffer that will hold the ciphertext

siglen must be the same as the modulus size (for example, 128 if RSA-1024 is used)

Returns:

0 if the signing operation was successful, or an ERR_RSA_XXX error code

Definition at line 325 of file dsa.c.

References ASN1_HASH_MDX, ASN1_HASH_SHA1, ERR_RSA_BAD_INPUT_DATA, rsa_context::len, RSA_MD2, RSA_MD4, RSA_MD5, rsa_private(), RSA_RAW, RSA_SHA1, and RSA_SIGN.

8.114.2.8 int rsa_pkcs1_verify (rsa_context * ctx, int alg_id, unsigned char * hash, int hashlen, unsigned char * sig, int siglen)

Perform a public RSA and check the message digest.

Parameters:

ctx points to an RSA public key

alg_id RSA_RAW, RSA_MD2/4/5 or RSA_SHA1

hash buffer holding the message digest

hashlen message digest length

sig buffer holding the ciphertext

siglen must be the same as the modulus size

Returns:

0 if the verify operation was successful, or an ERR_RSA_XXX error code

Definition at line 401 of file dsa.c.

References ASN1_HASH_MDX, ASN1_HASH_SHA1, ERR_RSA_BAD_INPUT_DATA, ERR_RSA_INVALID_PADDING, ERR_RSA_VERIFY_FAILED, int, rsa_context::len, RSA_MD2, RSA_MD4, RSA_MD5, rsa_public(), RSA_RAW, RSA_SHA1, and RSA_SIGN.

8.114.2.9 int rsa_private (rsa_context * ctx, unsigned char * input, int ilen, unsigned char * output, int olen)

Perform an RSA private key operation.

Returns:

0 if successful, or an ERR_RSA_XXX error code

Note:

This function does not take care of message padding: both ilen and olen must be equal to the modulus size (ctx->len). Also, be sure to set input[0] = 0.

Definition at line 147 of file dsa.c.

References CHK, rsa_context::D, rsa_context::DP, rsa_context::DQ, ERR_RSA_BAD_INPUT_DATA, ERR_RSA_PRIVATE_FAILED, rsa_context::len, mpi_add_mpi(), 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, and rsa_context::RQ.

8.114.2.10 int rsa_public (rsa_context * ctx, unsigned char * input, int ilen, unsigned char * output, int olen)

Perform an RSA public key operation.

Returns:

0 if successful, or an ERR_RSA_XXX error code

Note:

This function does not take care of message padding: both ilen and olen must be equal to the modulus size (ctx->len). Also, be sure to set input[0] = 0.

Definition at line 111 of file dsa.c.

References CHK, rsa_context::E, ERR_RSA_BAD_INPUT_DATA, ERR_RSA_PUBLIC_FAILED, rsa_context::len, mpi_cmp_mpi(), mpi_exp_mod(), mpi_free(), mpi_init(), mpi_read_binary(), mpi_write_binary(), rsa_context::N, and rsa_context::RN.

8.114.2.11 int rsa_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 585 of file dsa.c.

8.115 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/havege.c File Reference

```
#include <string.h>
#include <time.h>
#include "xyssl/timing.h"
#include "xyssl/havege.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`
- `#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 *rng_d)
HAVEGE rand function.

Variables

- static const char `_havege_src` [] = "_havege_src"

8.115.1 Define Documentation

8.115.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 29 of file havege.c.

8.115.1.2 `#define ONE_ITERATION`

Definition at line 60 of file havege.c.

Referenced by `havege_fill()`.

8.115.1.3 `#define SWAP(X, Y) { int *T = X; X = Y; Y = T; }`

Definition at line 52 of file havege.c.

8.115.1.4 `#define TST1_ENTER if(PTEST & 1) { PTEST ^= 3; PTEST >>= 1;`

Definition at line 54 of file havege.c.

8.115.1.5 `#define TST1_LEAVE U1++; }`

Definition at line 57 of file havege.c.

8.115.1.6 `#define TST2_ENTER if(PTEST & 1) { PTEST ^= 3; PTEST >>= 1;`

Definition at line 55 of file havege.c.

8.115.1.7 `#define TST2_LEAVE U2++; }`

Definition at line 58 of file havege.c.

8.115.2 Function Documentation

8.115.2.1 `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().

8.115.2.2 `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 main().

8.115.2.3 `int havege_rand (void * rng_d)`

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 main().

8.115.3 Variable Documentation

8.115.3.1 `const char _havege_src[] = "_havege_src"` [static]

Definition at line 209 of file havege.c.

8.116 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md2.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/md2.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- void [md2_starts](#) ([md2_context](#) *ctx)
MD2 context setup.
- static void [md2_process](#) ([md2_context](#) *ctx)
- void [md2_update](#) ([md2_context](#) *ctx, unsigned char *input, [int](#) ilen)
MD2 process buffer.
- void [md2_finish](#) ([md2_context](#) *ctx, unsigned char *output)
MD2 final digest.
- void [md2](#) (unsigned char *input, [int](#) ilen, unsigned char *output)
Output = MD2(input buffer).
- [int](#) [md2_file](#) (char *path, unsigned char *output)
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)
MD2 HMAC final digest.
- void [md2_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char *output)
Output = HMAC-MD2(hmac key, input buffer).
- [int](#) [md2_self_test](#) ([int](#) verbose)
Checkup routine.

Variables

- static const unsigned char `PI_SUBST` [256]
- static const char `_md2_src` [] = "_md2_src"

8.116.1 Define Documentation

8.116.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 28 of file md2.c.

8.116.2 Function Documentation

8.116.2.1 `void md2 (unsigned char * input, int ilen, unsigned char * output)`

Output = MD2(*input* buffer).

Parameters:

- input* buffer holding the data
- ilen* length of the input data
- output* MD2 checksum result

Definition at line 151 of file md2.c.

References `md2_finish()`, `md2_starts()`, and `md2_update()`.

Referenced by `x509_hash()`.

8.116.2.2 `int md2_file (char * path, unsigned char * output)`

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

Definition at line 166 of file md2.c.

References `f`, `md2_finish()`, `md2_starts()`, and `md2_update()`.

8.116.2.3 `void md2_finish (md2_context * ctx, unsigned char * output)`

MD2 final digest.

Parameters:

- ctx* MD2 context

output MD2 checksum result

Definition at line 130 of file md2.c.

References md2_context::buffer, md2_context::cksum, md2_context::left, md2_process(), and md2_context::state.

Referenced by md2(), md2_file(), and md2_hmac_finish().

8.116.2.4 void md2_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char * *output*)

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

Definition at line 246 of file md2.c.

References md2_hmac_finish(), md2_hmac_starts(), and md2_hmac_update().

8.116.2.5 void md2_hmac_finish (md2_context * *ctx*, unsigned char * *output*)

MD2 HMAC final digest.

Parameters:

ctx HMAC context

output MD2 HMAC checksum result

Definition at line 230 of file md2.c.

References md2_finish(), md2_starts(), md2_update(), and md2_context::opad.

Referenced by md2_hmac().

8.116.2.6 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

Definition at line 198 of file md2.c.

References md2_context::ipad, md2_starts(), md2_update(), and md2_context::opad.

Referenced by md2_hmac().

8.116.2.7 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

Definition at line 221 of file md2.c.

References md2_update().

Referenced by md2_hmac().

8.116.2.8 static void md2_process (md2_context * *ctx*) [static]

Definition at line 74 of file md2.c.

References md2_context::buffer, md2_context::cksum, PI_SUBST, and md2_context::state.

Referenced by md2_finish(), and md2_update().

8.116.2.9 int md2_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 327 of file md2.c.

Referenced by main().

8.116.2.10 void md2_starts (md2_context * *ctx*)

MD2 context setup.

Parameters:

ctx context to be initialized

Definition at line 69 of file md2.c.

Referenced by md2(), md2_file(), md2_hmac_finish(), and md2_hmac_starts().

8.116.2.11 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

Definition at line 102 of file md2.c.

References md2_context::buffer, md2_context::left, and md2_process().

Referenced by md2(), md2_file(), md2_hmac_finish(), md2_hmac_starts(), and md2_hmac_update().

8.116.3 Variable Documentation

8.116.3.1 `const char _md2_src[] = "_md2_src" [static]`

Definition at line 259 of file md2.c.

8.116.3.2 `const unsigned char PI_SUBST[256] [static]`

Initial value:

```
{
    0x29, 0x2E, 0x43, 0xC9, 0xA2, 0xD8, 0x7C, 0x01, 0x3D, 0x36,
    0x54, 0xA1, 0xEC, 0xF0, 0x06, 0x13, 0x62, 0xA7, 0x05, 0xF3,
    0xC0, 0xC7, 0x73, 0x8C, 0x98, 0x93, 0x2B, 0xD9, 0xBC, 0x4C,
    0x82, 0xCA, 0x1E, 0x9B, 0x57, 0x3C, 0xFD, 0xD4, 0xE0, 0x16,
    0x67, 0x42, 0x6F, 0x18, 0x8A, 0x17, 0xE5, 0x12, 0xBE, 0x4E,
    0xC4, 0xD6, 0xDA, 0x9E, 0xDE, 0x49, 0xA0, 0xFB, 0xF5, 0x8E,
    0xBB, 0x2F, 0xEE, 0x7A, 0xA9, 0x68, 0x79, 0x91, 0x15, 0xB2,
    0x07, 0x3F, 0x94, 0xC2, 0x10, 0x89, 0x0B, 0x22, 0x5F, 0x21,
    0x80, 0x7F, 0x5D, 0x9A, 0x5A, 0x90, 0x32, 0x27, 0x35, 0x3E,
    0xCC, 0xE7, 0xBF, 0xF7, 0x97, 0x03, 0xFF, 0x19, 0x30, 0xB3,
    0x48, 0xA5, 0xB5, 0xD1, 0xD7, 0x5E, 0x92, 0x2A, 0xAC, 0x56,
    0xAA, 0xC6, 0x4F, 0xB8, 0x38, 0xD2, 0x96, 0xA4, 0x7D, 0xB6,
    0x76, 0xFC, 0x6B, 0xE2, 0x9C, 0x74, 0x04, 0xF1, 0x45, 0x9D,
    0x70, 0x59, 0x64, 0x71, 0x87, 0x20, 0x86, 0x5B, 0xCF, 0x65,
    0xE6, 0x2D, 0xA8, 0x02, 0x1B, 0x60, 0x25, 0xAD, 0xAE, 0xB0,
    0xB9, 0xF6, 0x1C, 0x46, 0x61, 0x69, 0x34, 0x40, 0x7E, 0x0F,
    0x55, 0x47, 0xA3, 0x23, 0xDD, 0x51, 0xAF, 0x3A, 0xC3, 0x5C,
    0xF9, 0xCE, 0xBA, 0xC5, 0xEA, 0x26, 0x2C, 0x53, 0x0D, 0x6E,
    0x85, 0x28, 0x84, 0x09, 0xD3, 0xDF, 0xCD, 0xF4, 0x41, 0x81,
    0x4D, 0x52, 0x6A, 0xDC, 0x37, 0xC8, 0x6C, 0xC1, 0xAB, 0xFA,
    0x24, 0xE1, 0x7B, 0x08, 0x0C, 0xBD, 0xB1, 0x4A, 0x78, 0x88,
    0x95, 0x8B, 0xE3, 0x63, 0xE8, 0x6D, 0xE9, 0xCB, 0xD5, 0xFE,
    0x3B, 0x00, 0x1D, 0x39, 0xF2, 0xEF, 0xB7, 0x0E, 0x66, 0x58,
    0xD0, 0xE4, 0xA6, 0x77, 0x72, 0xF8, 0xEB, 0x75, 0x4B, 0x0A,
    0x31, 0x44, 0x50, 0xB4, 0x8F, 0xED, 0x1F, 0x1A, 0xDB, 0x99,
    0x8D, 0x33, 0x9F, 0x11, 0x83, 0x14
}
```

Definition at line 36 of file md2.c.

Referenced by md2_process().

8.117 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md4.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/md4.h"
```

Defines

- #define [_CRT_SECURE_NO_DEPRECATED](#) 1
- #define [GET_UINT32_LE](#)(n, b, i)
- #define [PUT_UINT32_LE](#)(n, b, i)
- #define [S](#)(x, n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))
- #define [F](#)(x, y, z) ((x & y) | ((~x) & z))
- #define [P](#)(a, b, c, d, x, s) { a += F(b,c,d) + x; a = S(a,s); }
- #define [F](#)(x, y, z) ((x & y) | (x & z) | (y & z))
- #define [P](#)(a, b, c, d, x, s) { a += F(b,c,d) + x + 0x5A827999; a = S(a,s); }
- #define [F](#)(x, y, z) (x ^ y ^ z)
- #define [P](#)(a, b, c, d, x, s) { a += F(b,c,d) + x + 0x6ED9EBA1; a = S(a,s); }

Functions

- void [md4_starts](#) ([md4_context](#) *ctx)
MD4 context setup.
- static void [md4_process](#) ([md4_context](#) *ctx, unsigned char data[64])
- void [md4_update](#) ([md4_context](#) *ctx, unsigned char *input, [int](#) ilen)
MD4 process buffer.
- void [md4_finish](#) ([md4_context](#) *ctx, unsigned char *output)
MD4 final digest.
- void [md4](#) (unsigned char *input, [int](#) ilen, unsigned char *output)
Output = MD4(input buffer).
- [int](#) [md4_file](#) (char *path, unsigned char *output)
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)
MD4 HMAC final digest.

- void `md4_hmac` (unsigned char *key, int keylen, unsigned char *input, int ilen, unsigned char *output)

Output = HMAC-MD4(hmac key, input buffer).

- int `md4_self_test` (int verbose)

Checkup routine.

Variables

- static const unsigned char `md4_padding` [64]
- static const char `_md4_src` [] = "_md4_src"

8.117.1 Define Documentation

8.117.1.1 #define _CRT_SECURE_NO_DEPRECATED 1

Definition at line 28 of file md4.c.

8.117.1.2 #define F(x, y, z) (x ^ y ^ z)

8.117.1.3 #define F(x, y, z) ((x & y) | (x & z) | (y & z))

8.117.1.4 #define F(x, y, z) ((x & y) | ((~x) & z))

Referenced by `sha2_process()`, and `sha4_process()`.

8.117.1.5 #define GET_UINT32_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 md4.c.

Referenced by `md4_process()`, and `md5_process()`.

8.117.1.6 #define P(a, b, c, d, x, s) { a += F(b,c,d) + x + 0x6ED9EBA1; a = S(a,s); }

8.117.1.7 #define P(a, b, c, d, x, s) { a += F(b,c,d) + x + 0x5A827999; a = S(a,s); }

8.117.1.8 #define P(a, b, c, d, x, s) { a += F(b,c,d) + x; a = S(a,s); }

Referenced by `main()`, `md4_process()`, `md5_process()`, `sha1_process()`, `sha2_process()`, and `sha4_process()`.

8.117.1.9 #define PUT_UINT32_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 50 of file md4.c.

Referenced by md4_finish(), and md5_finish().

8.117.1.10 #define S(x, n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))**8.117.2 Function Documentation****8.117.2.1 void md4 (unsigned char * *input*, int *ilen*, unsigned char * *output*)**

Output = MD4(*input* buffer).

Parameters:

input buffer holding the data
ilen length of the input data
output MD4 checksum result

Definition at line 259 of file md4.c.

References md4_finish(), md4_starts(), and md4_update().

Referenced by main(), and x509_hash().

8.117.2.2 int md4_file (char * *path*, unsigned char * *output*)

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

Definition at line 274 of file md4.c.

References f, md4_finish(), md4_starts(), and md4_update().

8.117.2.3 void md4_finish (md4_context * *ctx*, unsigned char * *output*)

MD4 final digest.

Parameters:

ctx MD4 context
output MD4 checksum result

Definition at line 231 of file md4.c.

References md4_padding, md4_update(), PUT_UINT32_LE, md4_context::state, and md4_context::total.

Referenced by md4(), md4_file(), and md4_hmac_finish().

8.117.2.4 void md4_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char * *output*)

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

Definition at line 354 of file md4.c.

References md4_hmac_finish(), md4_hmac_starts(), and md4_hmac_update().

8.117.2.5 void md4_hmac_finish (md4_context * *ctx*, unsigned char * *output*)

MD4 HMAC final digest.

Parameters:

ctx HMAC context
output MD4 HMAC checksum result

Definition at line 338 of file md4.c.

References md4_finish(), md4_starts(), md4_update(), and md4_context::opad.

Referenced by md4_hmac().

8.117.2.6 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

Definition at line 306 of file md4.c.

References md4_context::ipad, md4_starts(), md4_update(), and md4_context::opad.

Referenced by md4_hmac().

8.117.2.7 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

Definition at line 329 of file md4.c.

References md4_update().

Referenced by md4_hmac().

8.117.2.8 static void md4_process (md4_context * ctx, unsigned char data[64]) [static]

Definition at line 73 of file md4.c.

References GET_UINT32_LE, P, and md4_context::state.

Referenced by md4_update().

8.117.2.9 int md4_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 435 of file md4.c.

Referenced by main().

8.117.2.10 void md4_starts (md4_context * ctx)

MD4 context setup.

Parameters:

ctx context to be initialized

Definition at line 62 of file md4.c.

References md4_context::state, and md4_context::total.

Referenced by md4(), md4_file(), md4_hmac_finish(), and md4_hmac_starts().

8.117.2.11 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

Definition at line 179 of file md4.c.

References md4_context::buffer, md4_process(), and md4_context::total.

Referenced by md4(), md4_file(), md4_finish(), md4_hmac_finish(), md4_hmac_starts(), and md4_hmac_update().

8.117.3 Variable Documentation**8.117.3.1 const char _md4_src[] = "_md4_src" [static]**

Definition at line 367 of file md4.c.

8.117.3.2 const unsigned char md4_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 220 of file md4.c.

Referenced by md4_finish().

8.118 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md5.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/md5.h"
```

Defines

- #define [_CRT_SECURE_NO_DEPRECATED](#) 1
- #define [GET_UINT32_LE](#)(n, b, i)
- #define [PUT_UINT32_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)
MD5 final digest.
- void [md5](#) (unsigned char *input, [int](#) ilen, unsigned char *output)
Output = MD5(input buffer).
- [int](#) [md5_file](#) (char *path, unsigned char *output)
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)
MD5 HMAC final digest.
- void [md5_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char *output)

Output = HMAC-MD5(hmac key, input buffer).

- `int md5_self_test` (`int verbose`)

Checkup routine.

Variables

- static const unsigned char `md5_padding` [64]
- static const char `_md5_src` [] = "_md5_src"

8.118.1 Define Documentation

8.118.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 27 of file md5.c.

8.118.1.2 `#define F(x, y, z) (y ^ (x | ~z))`

8.118.1.3 `#define F(x, y, z) (x ^ y ^ z)`

8.118.1.4 `#define F(x, y, z) (y ^ (z & (x ^ y)))`

8.118.1.5 `#define F(x, y, z) (z ^ (x & (y ^ z)))`

8.118.1.6 `#define GET_UINT32_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.

8.118.1.7 `#define P(a, b, c, d, k, s, t)`

Value:

```
{
    a += F(b, c, d) + X[k] + t; a = S(a, s) + b;
}
```

8.118.1.8 `#define PUT_UINT32_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 49 of file md5.c.

8.118.1.9 `#define S(x, n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))`

8.118.2 Function Documentation

8.118.2.1 `void md5 (unsigned char * input, int ilen, unsigned char * output)`

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 main(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_server_key_exchange(), ssl_write_server_key_exchange(), and x509_hash().

8.118.2.2 `int md5_file (char * path, unsigned char * output)`

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 293 of file md5.c.

References f, md5_finish(), md5_starts(), and md5_update().

Referenced by md5_wrapper().

8.118.2.3 `void md5_finish (md5_context * ctx, unsigned char * output)`

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_UINT32_LE, md5_context::state, and md5_context::total.

Referenced by md5(), md5_file(), md5_hmac_finish(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_server_key_exchange(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

8.118.2.4 void md5_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char * *output*)

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 373 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().

8.118.2.5 void md5_hmac_finish (md5_context * *ctx*, unsigned char * *output*)

MD5 HMAC final digest.

Parameters:

ctx HMAC context

output MD5 HMAC checksum result

Definition at line 357 of file md5.c.

References md5_finish(), md5_starts(), md5_update(), and md5_context::opad.

Referenced by md5_hmac().

8.118.2.6 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 325 of file md5.c.

References md5_context::ipad, md5_starts(), md5_update(), and md5_context::opad.

Referenced by md5_hmac().

8.118.2.7 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 348 of file md5.c.

References md5_update().

Referenced by md5_hmac().

8.118.2.8 static void md5_process (md5_context * ctx, unsigned char data[64]) [static]

Definition at line 72 of file md5.c.

References GET_UINT32_LE, P, and md5_context::state.

Referenced by md5_update().

8.118.2.9 int md5_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 454 of file md5.c.

Referenced by main().

8.118.2.10 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_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_client_hello(), ssl_parse_server_key_exchange(), ssl_write_client_hello(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

8.118.2.11 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_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_client_hello(), ssl_parse_server_key_exchange(), ssl_write_record(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

8.118.3 Variable Documentation**8.118.3.1 const char _md5_src[] = "_md5_src" [static]**

Definition at line 386 of file md5.c.

8.118.3.2 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().

8.119 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/net.c File Reference

```
#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>
#include "xyssl/net.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- `int net_connect (int *fd, char *host, unsigned int port)`
Initiate a TCP connection with host:port.
- `int net_bind (int *fd, char *bind_ip, unsigned 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, unsigned char client_ip[4])`
Accept a connection from a remote client.
- `int net_set_block (int fd)`
- `int net_set_nonblock (int fd)`
Set the socket non-blocking.
- `void net_usleep (unsigned long usec)`
Portable usleep helper.
- `int net_recv (int fd, unsigned char *buf, int *len)`

Read at most 'len' characters. len is updated to reflect the actual number of characters read.

- `int net_send (int fd, 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.

8.119.1 Define Documentation

8.119.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file net.c.

8.119.2 Function Documentation

8.119.2.1 `int net_accept (int bind_fd, int * client_fd, unsigned char client_ip[4])`

Accept a connection from a remote client.

Returns:

0 if successful, `ERR_NET_ACCEPT_FAILED`, or `ERR_NET_WOULD_BLOCK` if `bind_fd` was set to non-blocking and `accept()` is blocking.

Definition at line 200 of file net.c.

References `ERR_NET_ACCEPT_FAILED`, `ERR_NET_WOULD_BLOCK`, `int`, and `net_is_blocking()`.

Referenced by `main()`.

8.119.2.2 `int net_bind (int * fd, char * bind_ip, unsigned 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: `ERR_NET_SOCKET_FAILED`, `ERR_NET_BIND_FAILED`, `ERR_NET_LISTEN_FAILED`

Definition at line 112 of file net.c.

References `ERR_NET_BIND_FAILED`, `ERR_NET_LISTEN_FAILED`, and `ERR_NET_SOCKET_FAILED`.

Referenced by `main()`.

8.119.2.3 `void net_close (int fd)`

Gracefully shutdown the connection.

Definition at line 337 of file net.c.

Referenced by `main()`.

8.119.2.4 int net_connect (int *fd, char *host, unsigned int port)

Initiate a TCP connection with host:port.

Returns:

0 if successful, or one of: ERR_NET_SOCKET_FAILED, ERR_NET_UNKNOWN_HOST, ERR_NET_CONNECT_FAILED

Definition at line 67 of file net.c.

References ERR_NET_CONNECT_FAILED, ERR_NET_SOCKET_FAILED, and ERR_NET_UNKNOWN_HOST.

Referenced by main().

8.119.2.5 static int net_is_blocking (void) [static]

Definition at line 178 of file net.c.

Referenced by net_accept(), net_rcv(), and net_send().

8.119.2.6 int net_rcv (int fd, unsigned char *buf, int *len)

Read at most 'len' characters. len is updated to reflect the actual number of characters read.

Returns:

0 if successful, ERR_NET_CONN_RESET if the connection was closed from the other side, or ERR_NET_WOULD_BLOCK if read() is blocking.

Definition at line 265 of file net.c.

References ERR_NET_CONN_RESET, ERR_NET_RECV_FAILED, ERR_NET_WOULD_BLOCK, and net_is_blocking().

Referenced by main(), and ssl_parse_client_hello().

8.119.2.7 int net_send (int fd, unsigned char *buf, int *len)

Write at most 'len' characters. len is updated to reflect the number of characters _not_ written.

Returns:

0 if successful, ERR_NET_CONN_RESET if the connection was closed from the other side, or ERR_NET_WOULD_BLOCK if write() is blocking.

Definition at line 302 of file net.c.

References ERR_NET_CONN_RESET, ERR_NET_SEND_FAILED, ERR_NET_WOULD_BLOCK, and net_is_blocking().

Referenced by main(), ssl_flush_output(), and ssl_write_record().

8.119.2.8 int net_set_block (int fd)

Definition at line 231 of file net.c.

8.119.2.9 `int net_set_nonblock (int fd)`

Set the socket non-blocking.

Returns:

0 if successful, or 1 if the operation failed

Definition at line 241 of file net.c.

8.119.2.10 `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 254 of file net.c.

8.120 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/rsa.c File Reference

```
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
#include "xyssl/rsa.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- `int rsa_gen_key (rsa_context *ctx, int nbits, int exponent, int(*rng_f)(void *), void *rng_d)`
Generate an RSA keypair.
- `int rsa_read_public (rsa_context *ctx, FILE *f)`
Read the public key from a file.
- `int rsa_read_private (rsa_context *ctx, FILE *f)`
Read the private key from a file.
- `int rsa_write_public (rsa_context *ctx, FILE *f)`
Write the public key into a file.
- `int rsa_write_private (rsa_context *ctx, FILE *f)`
Write the private key into a file.
- `int rsa_public (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int olen)`
Perform an RSA public key operation.
- `int rsa_private (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int olen)`
Perform an RSA private key operation.
- `int rsa_check_pubkey (rsa_context *ctx)`
Return 0 if the public key is valid, or ERR_RSA_KEY_CHECK_FAILED.
- `int rsa_check_privkey (rsa_context *ctx)`
Return 0 if the private key is valid, or ERR_RSA_KEY_CHECK_FAILED.
- `int rsa_pkcs1_encrypt (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int olen)`
Add the PKCS#1 v1.5 padding and do a public RSA.
- `int rsa_pkcs1_decrypt (rsa_context *ctx, unsigned char *input, int ilen, unsigned char *output, int *olen)`

Do a private RSA, removes the PKCS#1 v1.5 padding.

- `int rsa_pkcs1_sign` (`rsa_context` *ctx, `int` alg_id, unsigned char *hash, `int` hashlen, unsigned char *sig, `int` siglen)

Perform a private RSA to sign a message digest.

- `int rsa_pkcs1_verify` (`rsa_context` *ctx, `int` alg_id, unsigned char *hash, `int` hashlen, unsigned char *sig, `int` siglen)

Perform 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.

8.120.1 Define Documentation

8.120.1.1 #define _CRT_SECURE_NO_DEPRECATED 1

Definition at line 28 of file rsa.c.

8.120.2 Function Documentation

8.120.2.1 int rsa_check_privkey (rsa_context * ctx)

Return 0 if the private key is valid, or ERR_RSA_KEY_CHECK_FAILED.

Definition at line 326 of file rsa.c.

References `CHK`, `rsa_context::E`, `ERR_RSA_KEY_CHK_FAILED`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_free()`, `mpi_gcd()`, `mpi_init()`, `mpi_mul_mpi()`, `mpi_sub_int()`, `rsa_context::N`, `rsa_context::P`, and `rsa_context::Q`.

Referenced by `x509_parse_key()`.

8.120.2.2 int rsa_check_pubkey (rsa_context * ctx)

Return 0 if the public key is valid, or ERR_RSA_KEY_CHECK_FAILED.

Definition at line 306 of file rsa.c.

References `rsa_context::E`, `ERR_RSA_KEY_CHK_FAILED`, `mpi_msb()`, `rsa_context::N`, and `mpi::p`.

Referenced by `x509_add_certs()`.

8.120.2.3 void rsa_free (rsa_context * ctx)

Free the components of an RSA key.

Definition at line 569 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_read_private()`, `rsa_read_public()`, `x509_free_cert()`, and `x509_parse_key()`.

8.120.2.4 `int rsa_gen_key (rsa_context * ctx, int nbits, int exponent, int(*) (void *) rng_f, void * rng_d)`

Generate an RSA keypair.

Parameters:

ctx RSA context to be initialized
nbits size of the public key in bits
exponent public exponent (e.g., 65537)
rng_f points to the RNG function
rng_d points to the RNG data

Returns:

0 if successful, or an `ERR_RSA_XXX` error code

Definition at line 41 of file `rsa.c`.

References `CHK`, `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `ERR_RSA_BAD_INPUT_DATA`, `ERR_RSA_KEY_GEN_FAILED`, `rsa_context::len`, `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::Q`, `rsa_context::QP`, and `rsa_free()`.

Referenced by `main()`.

8.120.2.5 `int rsa_pkcs1_decrypt (rsa_context * ctx, unsigned char * input, int ilen, unsigned char * output, int * olen)`

Do a private RSA, removes the PKCS#1 v1.5 padding.

Parameters:

ctx RSA context
input buffer holding the encrypted data
ilen must be the same as the modulus size
output buffer that will hold the plaintext
olen size of output buffer, will be updated to contain the length of the plaintext

Returns:

0 if successful, or an `ERR_RSA_XXX` error code

Definition at line 385 of file `rsa.c`.

References `ERR_RSA_BAD_INPUT_DATA`, `ERR_RSA_INVALID_PADDING`, `int`, `rsa_context::len`, `RSA_CRYPT`, and `rsa_private()`.

Referenced by `ssl_parse_client_key_exchange()`.

8.120.2.6 int rsa_pkcs1_encrypt (rsa_context * ctx, unsigned char * input, int ilen, unsigned char * output, int olen)

Add the PKCS#1 v1.5 padding and do a public RSA.

Parameters:

ctx RSA context
input buffer holding the data to be encrypted
ilen length of the plaintext; cannot be longer than the modulus, minus 3+8 for padding
output buffer that will hold the ciphertext
olen must be the same as the modulus size (for example, 128 if RSA-1024 is used)

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 355 of file rsa.c.

References ERR_RSA_BAD_INPUT_DATA, rsa_context::len, RSA_CRYPT, and rsa_public().

Referenced by ssl_write_client_key_exchange().

8.120.2.7 int rsa_pkcs1_sign (rsa_context * ctx, int alg_id, unsigned char * hash, int hashlen, unsigned char * sig, int siglen)

Perform a private RSA to sign a message digest.

Parameters:

ctx RSA context
alg_id RSA_RAW, RSA_MD2/4/5 or RSA_SHA1
hash buffer holding the message digest
hashlen message digest length
sig buffer that will hold the ciphertext
siglen must be the same as the modulus size (for example, 128 if RSA-1024 is used)

Returns:

0 if the signing operation was successful, or an ERR_RSA_XXX error code

Definition at line 423 of file rsa.c.

References ASN1_HASH_MDX, ASN1_HASH_SHA1, ERR_RSA_BAD_INPUT_DATA, rsa_context::len, RSA_MD2, RSA_MD4, RSA_MD5, rsa_private(), RSA_RAW, RSA_SHA1, and RSA_SIGN.

Referenced by main(), ssl_write_certificate_verify(), and ssl_write_server_key_exchange().

8.120.2.8 int rsa_pkcs1_verify (rsa_context * ctx, int alg_id, unsigned char * hash, int hashlen, unsigned char * sig, int siglen)

Perform a public RSA and check the message digest.

Parameters:

ctx points to an RSA public key
alg_id RSA_RAW, RSA_MD2/4/5 or RSA_SHA1
hash buffer holding the message digest
hashlen message digest length
sig buffer holding the ciphertext
siglen must be the same as the modulus size

Returns:

0 if the verify operation was successful, or an ERR_RSA_XXX error code

Definition at line 499 of file rsa.c.

References ASN1_HASH_MDX, ASN1_HASH_SHA1, ERR_RSA_BAD_INPUT_DATA, ERR_RSA_INVALID_PADDING, ERR_RSA_VERIFY_FAILED, int, rsa_context::len, RSA_MD2, RSA_MD4, RSA_MD5, rsa_public(), RSA_RAW, RSA_SHA1, and RSA_SIGN.

Referenced by main(), ssl_parse_certificate_verify(), ssl_parse_server_key_exchange(), and x509_verify_cert().

8.120.2.9 **int rsa_private (rsa_context * ctx, unsigned char * input, int ilen, unsigned char * output, int olen)**

Perform an RSA private key operation.

Returns:

0 if successful, or an ERR_RSA_XXX error code

Note:

This function does not take care of message padding: both ilen and olen must be equal to the modulus size (ctx->len). Also, be sure to set input[0] = 0.

Definition at line 245 of file rsa.c.

References CHK, rsa_context::D, rsa_context::DP, rsa_context::DQ, ERR_RSA_BAD_INPUT_DATA, ERR_RSA_PRIVATE_FAILED, rsa_context::len, mpi_add_mpi(), 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, and rsa_context::RQ.

Referenced by main(), rsa_pkcs1_decrypt(), and rsa_pkcs1_sign().

8.120.2.10 **int rsa_public (rsa_context * ctx, unsigned char * input, int ilen, unsigned char * output, int olen)**

Perform an RSA public key operation.

Returns:

0 if successful, or an ERR_RSA_XXX error code

Note:

This function does not take care of message padding: both *ilen* and *olen* must be equal to the modulus size (*ctx->len*). Also, be sure to set *input[0] = 0*.

Definition at line 209 of file *rsa.c*.

References *CHK*, *rsa_context::E*, *ERR_RSA_BAD_INPUT_DATA*, *ERR_RSA_PUBLIC_FAILED*, *rsa_context::len*, *mpi_cmp_mpi()*, *mpi_exp_mod()*, *mpi_free()*, *mpi_init()*, *mpi_read_binary()*, *mpi_write_binary()*, *rsa_context::N*, and *rsa_context::RN*.

Referenced by *main()*, *rsa_pkcs1_encrypt()*, and *rsa_pkcs1_verify()*.

8.120.2.11 int rsa_read_private (rsa_context * ctx, FILE * f)

Read the private key from a file.

Parameters:

ctx RSA context to be initialized

f Handle of the source file

Returns:

0 if successful, or an *ERR_RSA_XXX* error code

Definition at line 136 of file *rsa.c*.

References *CHK*, *rsa_context::D*, *rsa_context::DP*, *rsa_context::DQ*, *rsa_context::E*, *ERR_RSA_KEY_RD_FAILED*, *rsa_context::len*, *mpi_msb()*, *mpi_read_file()*, *rsa_context::N*, *rsa_context::P*, *rsa_context::Q*, *rsa_context::QP*, and *rsa_free()*.

Referenced by *main()*.

8.120.2.12 int rsa_read_public (rsa_context * ctx, FILE * f)

Read the public key from a file.

Parameters:

ctx RSA context to be initialized

f Handle of the source file

Returns:

0 if successful, or an *ERR_RSA_XXX* error code

Definition at line 111 of file *rsa.c*.

References *CHK*, *rsa_context::E*, *ERR_RSA_KEY_RD_FAILED*, *rsa_context::len*, *mpi_msb()*, *mpi_read_file()*, *rsa_context::N*, and *rsa_free()*.

Referenced by *main()*.

8.120.2.13 int rsa_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 683 of file rsa.c.

Referenced by main().

8.120.2.14 int rsa_write_private (rsa_context * *ctx*, FILE * *f*)

Write the private key into a file.

Parameters:

ctx RSA context holding the key

f Handle of the destination file

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 185 of file rsa.c.

References CHK, rsa_context::D, rsa_context::DP, rsa_context::DQ, rsa_context::E, ERR_RSA_KEY_WR_FAILED, mpi_write_file(), rsa_context::N, rsa_context::P, rsa_context::Q, and rsa_context::QP.

Referenced by main().

8.120.2.15 int rsa_write_public (rsa_context * *ctx*, FILE * *f*)

Write the public key into a file.

Parameters:

ctx RSA context holding the key

f Handle of the destination file

Returns:

0 if successful, or an ERR_RSA_XXX error code

Definition at line 167 of file rsa.c.

References CHK, rsa_context::E, ERR_RSA_KEY_WR_FAILED, mpi_write_file(), and rsa_context::N.

Referenced by main().

8.121 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha1.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/sha1.h"
```

Defines

- #define [_CRT_SECURE_NO_DEPRECATED](#) 1
- #define [GET_UINT32_BE](#)(n, b, i)
- #define [PUT_UINT32_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)
SHA-1 final digest.
- void [sha1](#) (unsigned char *input, [int](#) ilen, unsigned char *output)
Output = SHA-1(input buffer).
- [int](#) [sha1_file](#) (char *path, unsigned char *output)
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)
SHA-1 HMAC final digest.
- void `sha1_hmac` (unsigned char *key, int keylen, unsigned char *input, int ilen, unsigned char *output)
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 const char `_sha1_src` [] = "_sha1_src"

8.121.1 Define Documentation

8.121.1.1 #define _CRT_SECURE_NO_DEPRECATED 1

Definition at line 27 of file sha1.c.

8.121.1.2 #define F(x, y, z) (x ^ y ^ z)

8.121.1.3 #define F(x, y, z) ((x & y) | (z & (x | y)))

8.121.1.4 #define F(x, y, z) (x ^ y ^ z)

8.121.1.5 #define F(x, y, z) (z ^ (x & (y ^ z)))

8.121.1.6 #define GET_UINT32_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.

8.121.1.7 #define K 0xCA62C1D6

8.121.1.8 #define K 0x8F1BBCDC

8.121.1.9 #define K 0x6ED9EBA1

8.121.1.10 #define K 0x5A827999

Referenced by `sha4_process()`.

8.121.1.11 #define P(a, b, c, d, e, x)**Value:**

```
{
    e += S(a, 5) + F(b, c, d) + K + x; b = S(b, 30);
}
```

8.121.1.12 #define PUT_UINT32_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.

8.121.1.13 #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().

8.121.1.14 #define S(x, n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))**8.121.2 Function Documentation****8.121.2.1 void sha1 (unsigned char * *input*, int *ilen*, unsigned char * *output*)**

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(), ssl_calc_verify(), ssl_derive_keys(), ssl_get_session(), ssl_mac_sha1(), ssl_parse_server_key_exchange(), ssl_set_session(), ssl_write_server_key_exchange(), and x509_hash().

8.121.2.2 int sha1_file (char * *path*, unsigned char * *output*)

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 328 of file sha1.c.

References f, sha1_finish(), sha1_starts(), and sha1_update().

Referenced by main(), and sha1_wrapper().

8.121.2.3 void sha1_finish (sha1_context * *ctx*, unsigned char * *output*)

SHA-1 final digest.

Parameters:

ctx SHA-1 context
output SHA-1 checksum result

Definition at line 284 of file sha1.c.

References PUT_UINT32_BE, sha1_padding, sha1_update(), sha1_context::state, and sha1_context::total.

Referenced by sha1(), sha1_file(), sha1_hmac_finish(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_sha1(), ssl_parse_server_key_exchange(), and ssl_write_server_key_exchange().

8.121.2.4 void sha1_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char * *output*)

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 408 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().

8.121.2.5 void sha1_hmac_finish (sha1_context * *ctx*, unsigned char * *output*)

SHA-1 HMAC final digest.

Parameters:

ctx HMAC context

output SHA-1 HMAC checksum result

Definition at line 392 of file sha1.c.

References sha1_context::opad, sha1_finish(), sha1_starts(), and sha1_update().

Referenced by sha1_hmac().

8.121.2.6 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 360 of file sha1.c.

References sha1_context::ipad, sha1_context::opad, sha1_starts(), and sha1_update().

Referenced by sha1_hmac().

8.121.2.7 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 383 of file sha1.c.

References sha1_update().

Referenced by sha1_hmac().

8.121.2.8 static void sha1_process (sha1_context * *ctx*, unsigned char *data*[64]) [static]

Definition at line 73 of file sha1.c.

References GET_UINT32_BE, P, R, and sha1_context::state.

Referenced by sha1_update().

8.121.2.9 `int sha1_self_test (int verbose)`

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 489 of file sha1.c.

Referenced by main().

8.121.2.10 `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(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_sha1(), ssl_parse_client_hello(), ssl_parse_server_key_exchange(), ssl_write_client_hello(), and ssl_write_server_key_exchange().

8.121.2.11 `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(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_sha1(), ssl_parse_client_hello(), ssl_parse_server_key_exchange(), ssl_write_record(), and ssl_write_server_key_exchange().

8.121.3 Variable Documentation

8.121.3.1 `const char _sha1_src[] = "_sha1_src" [static]`

Definition at line 421 of file sha1.c.

8.121.3.2 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().

8.122 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha2.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/sha2.h"
```

Defines

- #define [_CRT_SECURE_NO_DEPRECATED](#) 1
- #define [GET_UINT32_BE](#)(n, b, i)
- #define [PUT_UINT32_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)
SHA-256 final digest.
- void [sha2](#) (unsigned char *input, [int](#) ilen, unsigned char *output, [int](#) is224)
Output = SHA-256(input buffer).
- [int](#) [sha2_file](#) (char *path, unsigned char *output, [int](#) is224)
Output = SHA-256(file contents).
- void [sha2_hmac_starts](#) ([sha2_context](#) *ctx, [int](#) is224, unsigned char *key, [int](#) keylen)
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)

SHA-256 HMAC final digest.

- void `sha2_hmac` (unsigned char *key, int keylen, unsigned char *input, int ilen, unsigned char *output, 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 const char `_sha2_src` [] = "_sha2_src"

8.122.1 Define Documentation

8.122.1.1 #define _CRT_SECURE_NO_DEPRECATED 1

Definition at line 27 of file sha2.c.

8.122.1.2 #define F0(x, y, z) ((x & y) | (z & (x | y)))

8.122.1.3 #define F1(x, y, z) (z ^ (x & (y ^ z)))

8.122.1.4 #define GET_UINT32_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.

8.122.1.5 #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; \
}
```

8.122.1.6 #define PUT_UINT32_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.

8.122.1.7 #define R(t)**Value:**

```

(
    W[t] = S1(W[t - 2]) + W[t - 7] + \
    S0(W[t - 15]) + W[t - 16] \
)

```

8.122.1.8 #define ROTR(x, n) (SHR(x, n) | (x << (32 - n)))**8.122.1.9 #define S0(x) (ROTR(x, 7) ^ ROTR(x, 18) ^ SHR(x, 3))**

Referenced by sha4_process().

8.122.1.10 #define S1(x) (ROTR(x, 17) ^ ROTR(x, 19) ^ SHR(x, 10))

Referenced by sha4_process(), and tls1_prf().

8.122.1.11 #define S2(x) (ROTR(x, 2) ^ ROTR(x, 13) ^ ROTR(x, 22))

Referenced by tls1_prf().

8.122.1.12 #define S3(x) (ROTR(x, 6) ^ ROTR(x, 11) ^ ROTR(x, 25))**8.122.1.13 #define SHR(x, n) ((x & 0xFFFFFFFF) >> n)****8.122.2 Function Documentation****8.122.2.1 void sha2 (unsigned char * *input*, int *ilen*, unsigned char * *output*, 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().

8.122.2.2 int sha2_file (char * *path*, unsigned char * *output*, 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 f, sha2_finish(), sha2_starts(), and sha2_update().

Referenced by sha2_wrapper().

8.122.2.3 void sha2_finish (sha2_context * *ctx*, unsigned char * *output*)

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_UINT32_BE, sha2_padding, sha2_update(), sha2_context::state, and sha2_context::total.

Referenced by main(), sha2(), sha2_file(), and sha2_hmac_finish().

8.122.2.4 void sha2_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char * *output*, 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 413 of file sha2.c.

References sha2_hmac_finish(), sha2_hmac_starts(), and sha2_hmac_update().

8.122.2.5 void sha2_hmac_finish (sha2_context * ctx, unsigned char * output)

SHA-256 HMAC final digest.

Parameters:

ctx HMAC context
output SHA-224/256 HMAC checksum result

Definition at line 393 of file sha2.c.

References sha2_context::is224, sha2_context::opad, sha2_finish(), sha2_starts(), and sha2_update().

Referenced by main(), and sha2_hmac().

8.122.2.6 void sha2_hmac_starts (sha2_context * ctx, int is224, unsigned char * key, int keylen)

SHA-256 HMAC context setup.

Parameters:

ctx HMAC context to be initialized
is224 0 = use SHA256, 1 = use SHA224
key HMAC secret key
keylen length of the HMAC key

Definition at line 361 of file sha2.c.

References sha2_context::ipad, sha2_context::opad, sha2_starts(), and sha2_update().

Referenced by main(), and sha2_hmac().

8.122.2.7 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 384 of file sha2.c.

References sha2_update().

Referenced by main(), and sha2_hmac().

8.122.2.8 static void sha2_process (sha2_context * *ctx*, unsigned char *data*[64]) [static]

Definition at line 94 of file sha2.c.

References F, GET_UINT32_BE, P, R, and sha2_context::state.

Referenced by sha2_update().

8.122.2.9 int sha2_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 524 of file sha2.c.

Referenced by main().

8.122.2.10 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 main(), sha2(), sha2_file(), sha2_hmac_finish(), and sha2_hmac_starts().

8.122.2.11 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 main(), sha2(), sha2_file(), sha2_finish(), sha2_hmac_finish(), sha2_hmac_starts(), and sha2_hmac_update().

8.122.3 Variable Documentation

8.122.3.1 `const char _sha2_src[] = "_sha2_src"` `[static]`

Definition at line 426 of file sha2.c.

8.122.3.2 `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().

8.123 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha4.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/sha4.h"
```

Defines

- #define [_CRT_SECURE_NO_DEPRECATED](#) 1
- #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)
SHA-512 final digest.
- void [sha4](#) (unsigned char *input, [int](#) ilen, unsigned char *output, [int](#) is384)
Output = SHA-512(input buffer).
- [int](#) [sha4_file](#) (char *path, unsigned char *output, [int](#) is384)
Output = SHA-512(file contents).
- void [sha4_hmac_starts](#) ([sha4_context](#) *ctx, [int](#) is384, unsigned char *key, [int](#) keylen)
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)
SHA-512 HMAC final digest.

- void `sha4_hmac` (unsigned char *key, int keylen, unsigned char *input, int ilen, unsigned char *output, int is384)
Output = HMAC-SHA-512(hmac key, input buffer).
- int `sha4_self_test` (int verbose)
Checkup routine.

Variables

- static const uint64 `K` [80]
- static const unsigned char `sha4_padding` [128]
- static const char `_sha4_src` [] = "_sha4_src"

8.123.1 Define Documentation

8.123.1.1 #define _CRT_SECURE_NO_DEPRECATED 1

Definition at line 27 of file sha4.c.

8.123.1.2 #define F0(x, y, z) ((x & y) | (z & (x | y)))

8.123.1.3 #define F1(x, y, z) (z ^ (x & (y ^ z)))

8.123.1.4 #define GET_UINT64_BE(n, b, i)

Value:

```
{
    (n) = ( (uint64) (b) [(i)      ] << 56 )
          | ( (uint64) (b) [(i) + 1] << 48 )
          | ( (uint64) (b) [(i) + 2] << 40 )
          | ( (uint64) (b) [(i) + 3] << 32 )
          | ( (uint64) (b) [(i) + 4] << 24 )
          | ( (uint64) (b) [(i) + 5] << 16 )
          | ( (uint64) (b) [(i) + 6] <<  8 )
          | ( (uint64) (b) [(i) + 7]      ) ;
}
```

Definition at line 39 of file sha4.c.

Referenced by `sha4_process()`.

8.123.1.5 #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;
}
```


8.123.1.6 #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().

8.123.1.7 #define ROTR(x, n) (SHR(x,n) | (x << (64 - n)))**8.123.1.8 #define S0(x) (ROTR(x, 1) ^ ROTR(x, 8) ^ SHR(x, 7))****8.123.1.9 #define S1(x) (ROTR(x,19) ^ ROTR(x,61) ^ SHR(x, 6))****8.123.1.10 #define S2(x) (ROTR(x,28) ^ ROTR(x,34) ^ ROTR(x,39))****8.123.1.11 #define S3(x) (ROTR(x,14) ^ ROTR(x,18) ^ ROTR(x,41))****8.123.1.12 #define SHR(x, n) (x >> n)****8.123.2 Function Documentation****8.123.2.1 void sha4 (unsigned char * *input*, int *ilen*, unsigned char * *output*, 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().

8.123.2.2 int sha4_file (char * *path*, unsigned char * *output*, 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 f, sha4_finish(), sha4_starts(), and sha4_update().

8.123.2.3 void sha4_finish (sha4_context * ctx, unsigned char * output)

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, sha4_context::is384, PUT_UINT64_BE, sha4_padding, sha4_update(), sha4_context::state, sha4_context::total, and uint64.

Referenced by sha4(), sha4_file(), and sha4_hmac_finish().

8.123.2.4 void sha4_hmac (unsigned char * key, int keylen, unsigned char * input, int ilen, unsigned char * output, 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 411 of file sha4.c.

References sha4_hmac_finish(), sha4_hmac_starts(), and sha4_hmac_update().

8.123.2.5 void sha4_hmac_finish (sha4_context * ctx, unsigned char * output)

SHA-512 HMAC final digest.

Parameters:

ctx HMAC context

output SHA-384/512 HMAC checksum result

Definition at line 391 of file sha4.c.

References sha4_context::is384, sha4_context::opad, sha4_finish(), sha4_starts(), and sha4_update().

Referenced by sha4_hmac().

8.123.2.6 void sha4_hmac_starts (sha4_context * ctx, int is384, unsigned char * key, int keylen)

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_starts(), and sha4_update().

Referenced by sha4_hmac().

8.123.2.7 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 382 of file sha4.c.

References sha4_update().

Referenced by sha4_hmac().

8.123.2.8 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, K, P, S0, S1, sha4_context::state, and uint64.

Referenced by sha4_update().

8.123.2.9 int sha4_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 541 of file sha4.c.

8.123.2.10 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(), and sha4_hmac_starts().

8.123.2.11 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, sha4_process(), sha4_context::total, and uint64.

Referenced by sha4(), sha4_file(), sha4_finish(), sha4_hmac_finish(), sha4_hmac_starts(), and sha4_hmac_update().

8.123.3 Variable Documentation**8.123.3.1 const char _sha4_src[] = "_sha4_src" [static]**

Definition at line 424 of file sha4.c.

8.123.3.2 const uint64 K[80] [static]

Definition at line 69 of file sha4.c.

8.123.3.3 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().

8.124 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_cli.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <time.h>
#include "xyssl/ssl.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

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_client_start (ssl_context *ssl)`

Variables

- static const char `_ssl_cli_src [] = "_ssl_cli_src"`

8.124.1 Define Documentation

8.124.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file `ssl_cli.c`.

8.124.2 Function Documentation

8.124.2.1 `int ssl_client_start (ssl_context * ssl)`

Definition at line 438 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_flush_output()`, `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()`, and `ssl_context::state`.

Referenced by `ssl_handshake()`.

8.124.2.2 `static int ssl_parse_certificate_request (ssl_context *ssl)` [static]

Definition at line 264 of file `ssl_cli.c`.

References `ssl_context::client_auth`, `ERR_SSL_CERTIFICATE_REQUIRED`, `ERR_SSL_PRIVATE_KEY_REQUIRED`, `ERR_SSL_UNEXPECTED_MESSAGE`, `ssl_context::in_msg`, `ssl_context::in_msgtype`, `ssl_context::own_cert`, `ssl_context::own_key`, `SSL_HS_CERTIFICATE_REQUEST`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, and `ssl_context::state`.

Referenced by `ssl_client_start()`.

8.124.2.3 `static int ssl_parse_server_hello (ssl_context *ssl)` [static]

Definition at line 102 of file `ssl_cli.c`.

References `ssl_context::cipher`, `ssl_context::cipherlist`, `ERR_SSL_BAD_HS_SERVER_HELLO`, `ERR_SSL_NO_CIPHER_CHOSEN`, `ERR_SSL_UNEXPECTED_MESSAGE`, `ssl_context::in_hhlen`, `ssl_context::in_msg`, `ssl_context::in_msgtype`, `int`, `ssl_context::minor_ver`, `ssl_context::randbytes`, `ssl_context::resumed`, `ssl_context::sessid`, `ssl_context::sidlen`, `SSL_COMPRESS_NULL`, `ssl_derive_keys()`, `SSL_HS_SERVER_HELLO`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, `SSL_SERVER_CHANGE_CIPHER_SPEC`, `SSLV3_MAJOR_VERSION`, `SSLV3_MINOR_VERSION`, `ssl_context::state`, and `TLS10_MINOR_VERSION`.

Referenced by `ssl_client_start()`.

8.124.2.4 `static int ssl_parse_server_hello_done (ssl_context *ssl)` [static]

Definition at line 307 of file `ssl_cli.c`.

References `ssl_context::client_auth`, `ERR_SSL_BAD_HS_SERVER_HELLO_DONE`, `ERR_SSL_UNEXPECTED_MESSAGE`, `ssl_context::in_hhlen`, `ssl_context::in_msg`, `ssl_context::in_msgtype`, `SSL_HS_SERVER_HELLO_DONE`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, and `ssl_context::state`.

Referenced by `ssl_client_start()`.

8.124.2.5 `static int ssl_parse_server_key_exchange (ssl_context *ssl)` [static]

Definition at line 181 of file `ssl_cli.c`.

References `ssl_context::cipher`, `ssl_context::dhm_ctx`, `dhm_read_params()`, `ERR_SSL_BAD_HS_SERVER_KEY_EXCHANGE`, `ERR_SSL_FEATURE_UNAVAILABLE`, `ERR_SSL_UNEXPECTED_MESSAGE`, `ssl_context::in_hhlen`, `ssl_context::in_msg`, `ssl_context::in_msgtype`, `dhm_context::len`, `rsa_context::len`, `md5()`, `md5_finish()`, `md5_starts()`, `md5_update()`, `ssl_context::peer_cert`, `ssl_context::randbytes`, `_x509_cert::rsa`, `rsa_pkcs1_verify()`, `RSA_RAW`, `sha1()`, `sha1_finish()`, `sha1_starts()`, `sha1_update()`, `SSL3_EDH_RSA_DES_168_SHA`, `SSL_HS_SERVER_KEY_EXCHANGE`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, `ssl_context::state`, and `TLS1_EDH_RSA_AES_256_SHA`.

Referenced by `ssl_client_start()`.

8.124.2.6 `static int ssl_write_certificate_verify (ssl_context *ssl)` [static]

Definition at line 401 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::own_key`, `rsa_pkcs1_sign()`, `RSA_RAW`, `ssl_calc_verify()`, `SSL_HS_CERTIFICATE_VERIFY`, `SSL_MSG_HANDSHAKE`, `ssl_write_record()`, and `ssl_context::state`.

Referenced by `ssl_client_start()`.

8.124.2.7 `static int ssl_write_client_hello (ssl_context *ssl) [static]`

Definition at line 31 of file `ssl_cli.c`.

References `ssl_context::cipherlist`, `ssl_context::hs_md5`, `ssl_context::hs_sha1`, `ssl_context::major_ver`, `ssl_context::max_ver`, `md5_starts()`, `ssl_context::minor_ver`, `ssl_context::out_msg`, `ssl_context::out_msglen`, `ssl_context::out_msgtype`, `ssl_context::randbytes`, `ssl_context::rng_d`, `ssl_context::rng_f`, `ssl_context::sessid`, `sha1_starts()`, `ssl_context::sidlen`, `SSL_COMPRESS_NULL`, `SSL_HS_CLIENT_HELLO`, `SSL_MSG_HANDSHAKE`, `ssl_write_record()`, `SSLV3_MAJOR_VERSION`, `SSLV3_MINOR_VERSION`, `ssl_context::state`, and `TLS10_MINOR_VERSION`.

Referenced by `ssl_client_start()`.

8.124.2.8 `static int ssl_write_client_key_exchange (ssl_context *ssl) [static]`

Definition at line 328 of file `ssl_cli.c`.

References `ssl_context::cipher`, `dhm_calc_secret()`, `ssl_context::dhm_ctx`, `dhm_make_public()`, `ERR_SSL_FEATURE_UNAVAILABLE`, `rsa_context::len`, `dhm_context::len`, `ssl_context::max_ver`, `ssl_context::minor_ver`, `ssl_context::out_msg`, `ssl_context::out_msglen`, `ssl_context::out_msgtype`, `ssl_context::peer_cert`, `ssl_context::pmslen`, `ssl_context::premaster`, `ssl_context::rng_d`, `ssl_context::rng_f`, `_x509_cert::rsa`, `rsa_pkcs1_encrypt()`, `SSL3_EDH_RSA_DES_168_SHA`, `ssl_derive_keys()`, `SSL_HS_CLIENT_KEY_EXCHANGE`, `SSL_MSG_HANDSHAKE`, `ssl_write_record()`, `SSLV3_MINOR_VERSION`, `ssl_context::state`, and `TLS1_EDH_RSA_AES_256_SHA`.

Referenced by `ssl_client_start()`.

8.124.3 Variable Documentation

8.124.3.1 `const char _ssl_cli_src[] = "_ssl_cli_src" [static]`

Definition at line 433 of file `ssl_cli.c`.

8.125 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_srv.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include "xyssl/net.h"
#include "xyssl/ssl.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- static `int ssl_get_session (ssl_context *ssl)`
- static void `ssl_set_session (ssl_context *ssl)`
- 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_server_start (ssl_context *ssl)`

Variables

- static const char `_ssl_srv_src [] = "_ssl_srv_src"`

8.125.1 Define Documentation

8.125.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file `ssl_srv.c`.

8.125.2 Function Documentation

8.125.2.1 `static int ssl_get_session (ssl_context *ssl)` [static]

Definition at line 33 of file `ssl_srv.c`.

References `ssl_context::cipher`, `ERR_SSL_NO_SESSION_FOUND`, `int`, `ssl_context::master`, `ssl_context::sessid`, `sha1()`, `ssl_context::sidlen`, `ssl_context::sidtable`, `SSL_EXPIRATION_TIME`, and `SSL_SESSION_TBL_LEN`.

Referenced by `ssl_write_server_hello()`.

8.125.2.2 static int ssl_parse_certificate_verify (ssl_context * ssl) [static]

Definition at line 621 of file ssl_srv.c.

References ssl_context::authmode, ERR_SSL_BAD_HS_CERTIFICATE_VERIFY, ssl_context::in_hslen, ssl_context::in_msg, ssl_context::in_msgtype, int, rsa_context::len, ssl_context::peer_cert, _x509_cert::rsa, rsa_pkcs1_verify(), RSA_RAW, ssl_calc_verify(), SSL_HS_CERTIFICATE_VERIFY, SSL_MSG_HANDSHAKE, ssl_read_record(), SSL_VERIFY_REQUIRED, and ssl_context::state.

Referenced by ssl_server_start().

8.125.2.3 static int ssl_parse_client_hello (ssl_context * ssl) [static]

Definition at line 116 of file ssl_srv.c.

References ssl_context::cipher, ssl_context::cipherlist, ERR_SSL_BAD_HS_CLIENT_HELLO, ERR_SSL_NO_CIPHER_CHOSEN, ssl_context::hs_md5, ssl_context::hs_sha1, ssl_context::in_hdr, ssl_context::in_left, ssl_context::in_msg, int, ssl_context::major_ver, ssl_context::max_ver, md5_starts(), md5_update(), ssl_context::minor_ver, net_recv(), ssl_context::randbytes, ssl_context::read_fd, ssl_context::sessid, sha1_starts(), sha1_update(), ssl_context::sidlen, SSL_HS_CLIENT_HELLO, SSL_MSG_HANDSHAKE, SSLV3_MAJOR_VERSION, ssl_context::state, and TLS10_MINOR_VERSION.

Referenced by ssl_server_start().

8.125.2.4 static int ssl_parse_client_key_exchange (ssl_context * ssl) [static]

Definition at line 540 of file ssl_srv.c.

References ssl_context::cipher, dhm_calc_secret(), ssl_context::dhm_ctx, dhm_read_public(), ERR_SSL_BAD_HS_CLIENT_KEY_EXCHANGE, ERR_SSL_FEATURE_UNAVAILABLE, ssl_context::in_hslen, ssl_context::in_msg, ssl_context::in_msgtype, rsa_context::len, dhm_context::len, ssl_context::max_ver, ssl_context::minor_ver, ssl_context::own_key, ssl_context::pmslen, ssl_context::premaster, rsa_pkcs1_decrypt(), SSL3_EDH_RSA_DES_168_SHA, ssl_derive_keys(), SSL_HS_CLIENT_KEY_EXCHANGE, SSL_MSG_HANDSHAKE, ssl_read_record(), ssl_set_session(), SSLV3_MINOR_VERSION, ssl_context::state, and TLS1_EDH_RSA_AES_256_SHA.

Referenced by ssl_server_start().

8.125.2.5 int ssl_server_start (ssl_context * ssl)

Definition at line 663 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_flush_output(), 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(), and ssl_context::state.

Referenced by main(), and ssl_handshake().

8.125.2.6 static void ssl_set_session (ssl_context * ssl) [static]

Definition at line 78 of file ssl_srv.c.

References ssl_context::cipher, int, ssl_context::master, ssl_context::sessid, sha1(), ssl_context::sidlen, ssl_context::sidtable, and SSL_SESSION_TBL_LEN.

Referenced by ssl_parse_client_key_exchange().

8.125.2.7 static int ssl_write_certificate_request (ssl_context * ssl) [static]

Definition at line 406 of file ssl_srv.c.

References ssl_context::authmode, 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_HS_CERTIFICATE_REQUEST, SSL_MSG_HANDSHAKE, SSL_VERIFY_NONE, ssl_write_record(), ssl_context::state, and _x509_cert::subject_raw.

Referenced by ssl_server_start().

8.125.2.8 static int ssl_write_server_hello (ssl_context * ssl) [static]

Definition at line 336 of file ssl_srv.c.

References ssl_context::cipher, ssl_context::major_ver, ssl_context::minor_ver, ssl_context::out_msg, ssl_context::out_msglen, ssl_context::out_msgtype, ssl_context::randbytes, ssl_context::resumed, ssl_context::rng_d, ssl_context::rng_f, ssl_context::sessid, ssl_context::sidlen, SSL_COMPRESS_NULL, ssl_derive_keys(), ssl_get_session(), SSL_HS_SERVER_HELLO, SSL_MSG_HANDSHAKE, SSL_SERVER_CHANGE_CIPHER_SPEC, ssl_write_record(), and ssl_context::state.

Referenced by ssl_server_start().

8.125.2.9 static int ssl_write_server_hello_done (ssl_context * ssl) [static]

Definition at line 530 of file ssl_srv.c.

References ssl_context::out_msg, ssl_context::out_msglen, ssl_context::out_msgtype, SSL_HS_SERVER_HELLO_DONE, SSL_MSG_HANDSHAKE, ssl_write_record(), and ssl_context::state.

Referenced by ssl_server_start().

8.125.2.10 static int ssl_write_server_key_exchange (ssl_context * ssl) [static]

Definition at line 460 of file ssl_srv.c.

References ssl_context::cipher, ssl_context::dhm_ctx, dhm_make_params(), ERR_SSL_FEATURE_UNAVAILABLE, rsa_context::len, md5(), md5_finish(), md5_starts(), md5_update(), ssl_context::out_msg, ssl_context::out_msglen, ssl_context::out_msgtype, ssl_context::own_key, ssl_context::randbytes, ssl_context::rng_d, ssl_context::rng_f, rsa_pkcs1_sign(), RSA_RAW, sha1(), sha1_finish(), sha1_starts(), sha1_update(), SSL3_EDH_RSA_DES_168_SHA, SSL_HS_SERVER_KEY_EXCHANGE, SSL_MSG_HANDSHAKE, ssl_write_record(), ssl_context::state, and TLS1_EDH_RSA_AES_256_SHA.

Referenced by ssl_server_start().

8.125.3 Variable Documentation

8.125.3.1 `const char _ssl_srv_src[] = "_ssl_srv_src"` `[static]`

Definition at line 658 of file `ssl_srv.c`.

8.126 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_tls.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <time.h>
#include "xyssl/net.h"
#include "xyssl/ssl.h"
#include "xyssl/aes.h"
#include "xyssl/arc4.h"
#include "xyssl/des.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- static void `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)
- int `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_write_record` (ssl_context *ssl, int do_crypt)
- int `ssl_read_record` (ssl_context *ssl, int do_crypt)
- int `ssl_flush_output` (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, int client_resume)

Initialize the SSL context. If client_resume is not null, the session id and premaster secret are preserved (client-side only).
- void `ssl_set_endpoint` (ssl_context *ssl, int endpoint)

Set the current endpoint type, SSL_IS_CLIENT or SSL_IS_SERVER.

- void `ssl_set_authmode` (`ssl_context` *ssl, int authmode)
Set the certificate verification mode.
- void `ssl_set_rng_func` (`ssl_context` *ssl, int(*rng_f)(void *), void *rng_d)
Set the random number generator function.
- void `ssl_set_io_files` (`ssl_context` *ssl, int read_fd, int write_fd)
Set the read and write file descriptors.
- void `ssl_set_ciphlist` (`ssl_context` *ssl, int *ciphers)
Set the list of allowed ciphersuites.
- void `ssl_set_ca_chain` (`ssl_context` *ssl, `x509_cert` *ca, char *cn)
Set the CA certificate chain used to verify peer cert, and the peer's expected CommonName (or NULL).
- void `ssl_set_rsa_cert` (`ssl_context` *ssl, `x509_cert` *own_cert, `rsa_context` *own_key)
Set own certificate and private RSA key.
- void `ssl_set_sidtable` (`ssl_context` *ssl, unsigned char *sidtable)
Set the global session ID table (server-side only).
- int `ssl_set_dhm_vals` (`ssl_context` *ssl, char *dhm_P, char *dhm_G)
Set the Diffie-Hellman public P and G values, provided as hexadecimal strings (server-side only).
- int `ssl_get_verify_result` (`ssl_context` *ssl)
Return the result of the certificate verification.
- char * `ssl_get_cipher_name` (`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 '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` []
- static const char `_ssl_tls_src` [] = "_ssl_tls_src"

8.126.1 Define Documentation

8.126.1.1 #define _CRT_SECURE_NO_DEPRECATED 1

Definition at line 30 of file ssl_tls.c.

8.126.2 Function Documentation

8.126.2.1 static void ssl_calc_finished (ssl_context *ssl, unsigned char *buf, int from, md5_context *md5, sha1_context *sha1) [static]

Definition at line 1045 of file ssl_tls.c.

8.126.2.2 int ssl_calc_verify (ssl_context *ssl, unsigned char hash[36])

Definition at line 326 of file ssl_tls.c.

References ssl_context::hs_md5, ssl_context::hs_sha1, ssl_context::master, md5(), md5_finish(), md5_starts(), md5_update(), ssl_context::minor_ver, sha1(), sha1_finish(), sha1_starts(), sha1_update(), and SSLV3_MINOR_VERSION.

Referenced by ssl_parse_certificate_verify(), and ssl_write_certificate_verify().

8.126.2.3 int ssl_close_notify (ssl_context *ssl)

Notify the peer that the connection is being closed.

Definition at line 1450 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_flush_output(), SSL_HANDSHAKE_OVER, SSL_MSG_ALERT, ssl_write_record(), and ssl_context::state.

Referenced by main().

8.126.2.4 static int ssl_decrypt_buf (ssl_context *ssl) [static]

Definition at line 516 of file ssl_tls.c.

References aes_cbc_decrypt(), arc4_crypt(), ssl_context::ctx_dec, des3_cbc_decrypt(), ERR_SSL_FEATURE_UNAVAILABLE, ERR_SSL_INVALID_MAC, ssl_context::in_ctr, ssl_context::in_hdr, ssl_context::in_msg, ssl_context::in_msglen, ssl_context::in_msgtype, int, 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_mac_md5(), ssl_mac_sha1(), and SSLV3_MINOR_VERSION.

8.126.2.5 int ssl_derive_keys (ssl_context *ssl)

Definition at line 101 of file ssl_tls.c.

References aes_set_key(), arc4_setup(), ssl_context::cipher, ssl_context::ctx_dec, ssl_context::ctx_enc, ssl_context::ctxlen, des3_set_3keys(), ssl_context::endpoint, ERR_SSL_FEATURE_UNAVAILABLE, 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_context::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::resumed, sha1(), sha1_finish(), sha1_starts(), sha1_update(), SSL3_EDH_RSA_DES_168_SHA, SSL3_RSA_DES_168_SHA, SSL3_RSA_RC4_128_MD5, SSL3_RSA_RC4_128_SHA, SSL_IS_CLIENT, SSLV3_MINOR_VERSION, TLS1_EDH_RSA_AES_256_SHA, tls1_prf(), and TLS1_RSA_AES_256_SHA.

Referenced by ssl_parse_client_key_exchange(), ssl_parse_server_hello(), ssl_write_client_key_exchange(), and ssl_write_server_hello().

8.126.2.6 static int ssl_encrypt_buf (ssl_context *ssl) [static]

Definition at line 433 of file ssl_tls.c.

References aes_cbc_encrypt(), arc4_crypt(), ssl_context::ctx_enc, des3_cbc_encrypt(), ERR_SSL_FEATURE_UNAVAILABLE, 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_mac_md5(), ssl_mac_sha1(), and SSLV3_MINOR_VERSION.

Referenced by ssl_write_record().

8.126.2.7 int ssl_flush_output (ssl_context *ssl)

Definition at line 823 of file ssl_tls.c.

References net_send(), ssl_context::out_hdr, ssl_context::out_left, ssl_context::out_msglen, and ssl_context::write_fd.

Referenced by ssl_client_start(), ssl_close_notify(), and ssl_server_start().

8.126.2.8 void ssl_free (ssl_context *ssl)

Free an SSL context.

Definition at line 1473 of file ssl_tls.c.

References ssl_context::ctx_dec, ssl_context::ctx_enc, ssl_context::ctxlen, ssl_context::dhm_ctx, dhm_free(), ssl_context::in_ctr, ssl_context::out_ctr, ssl_context::peer_cert, SSL_BUFFER_LEN, and x509_free_cert().

Referenced by main().

8.126.2.9 char* ssl_get_cipher_name (ssl_context *ssl)

Return the name of the current cipher.

Definition at line 1308 of file ssl_tls.c.

Referenced by main().

8.126.2.10 int ssl_get_verify_result (ssl_context *ssl)

Return the result of the certificate verification.

Definition at line 1303 of file ssl_tls.c.

References `ssl_context::verify_result`.

Referenced by `main()`.

8.126.2.11 `int ssl_handshake(ssl_context *ssl)`

Perform the SSL handshake.

Returns:

0 if successful, `ERR_NET_WOULD_BLOCK` (only when the socket is set to non-blocking), or a specific SSL error code.

Definition at line 1369 of file `ssl_tls.c`.

References `ssl_context::endpoint`, `ERR_SSL_FEATURE_UNAVAILABLE`, `ssl_client_start()`, `SSL_IS_CLIENT`, `SSL_IS_SERVER`, and `ssl_server_start()`.

Referenced by `main()`, `ssl_read()`, and `ssl_write()`.

8.126.2.12 `int ssl_init(ssl_context *ssl, int client_resume)`

Initialize the SSL context. If `client_resume` is not null, the session id and premaster secret are preserved (client-side only).

Returns:

0 if successful, or 1 if memory allocation failed

Definition at line 1199 of file `ssl_tls.c`.

References `ssl_context::in_ctr`, `ssl_context::in_hdr`, `ssl_context::in_msg`, `ssl_context::master`, `ssl_context::out_ctr`, `ssl_context::out_hdr`, `ssl_context::out_msg`, `ssl_context::sessid`, `ssl_context::sidlen`, and `SSL_BUFFER_LEN`.

Referenced by `main()`.

8.126.2.13 `static void ssl_mac_md5(unsigned char *secret, unsigned char *buf, int len, unsigned char *ctr, int type)` [static]

Definition at line 372 of file `ssl_tls.c`.

References `md5()`, `md5_finish()`, `md5_starts()`, and `md5_update()`.

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

8.126.2.14 `static void ssl_mac_sha1(unsigned char *secret, unsigned char *buf, int len, unsigned char *ctr, int type)` [static]

Definition at line 401 of file `ssl_tls.c`.

References `sha1()`, `sha1_finish()`, `sha1_starts()`, and `sha1_update()`.

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

8.126.2.15 int ssl_parse_certificate (ssl_context * ssl)

Definition at line 914 of file ssl_tls.c.

Referenced by ssl_client_start(), and ssl_server_start().

8.126.2.16 int ssl_parse_change_cipher_spec (ssl_context * ssl)

Definition at line 1028 of file ssl_tls.c.

Referenced by ssl_client_start(), and ssl_server_start().

8.126.2.17 int ssl_parse_finished (ssl_context * ssl)

Definition at line 1154 of file ssl_tls.c.

Referenced by ssl_client_start(), and ssl_server_start().

8.126.2.18 int ssl_read (ssl_context * ssl, unsigned char * buf, int * len)

Read at most 'len' application data bytes.

Returns:

0 if successful, ERR_NET_WOULD_BLOCK (only when the socket is set to non-blocking), or a specific SSL error code.

Note:

len is updated to reflect the actual number of data bytes read.

Definition at line 1387 of file ssl_tls.c.

References ERR_SSL_UNEXPECTED_MESSAGE, ssl_context::in_msg, ssl_context::in_msglen, ssl_context::in_msgtype, ssl_context::in_offt, ssl_handshake(), SSL_MSG_APPLICATION_DATA, and ssl_read_record().

Referenced by main().

8.126.2.19 int ssl_read_record (ssl_context * ssl, int do_crypt)

Definition at line 682 of file ssl_tls.c.

Referenced by ssl_parse_certificate_request(), ssl_parse_certificate_verify(), ssl_parse_client_key_exchange(), ssl_parse_server_hello(), ssl_parse_server_hello_done(), ssl_parse_server_key_exchange(), and ssl_read().

8.126.2.20 void ssl_set_authmode (ssl_context * ssl, int authmode)

Set the certificate verification mode.

Parameters:

mode can be:

SSL_VERIFY_NONE: peer certificate is not checked (default).

SSL_VERIFY_OPTIONAL: peer certificate is checked, however the handshake continues even if verification failed; you may want to check `ssl->verify_result` after.

SSL_VERIFY_REQUIRED: peer *must* present a valid certificate, handshake is aborted if verification failed.

Definition at line 1249 of file `ssl_tls.c`.

Referenced by `main()`.

8.126.2.21 void ssl_set_ca_chain (ssl_context * ssl, x509_cert * ca, char * cn)

Set the CA certificate chain used to verify peer cert, and the peer's expected CommonName (or NULL).

Definition at line 1273 of file `ssl_tls.c`.

Referenced by `main()`.

8.126.2.22 void ssl_set_ciphlist (ssl_context * ssl, int * ciphers)

Set the list of allowed ciphersuites.

Definition at line 1268 of file `ssl_tls.c`.

Referenced by `main()`.

8.126.2.23 int ssl_set_dhm_vals (ssl_context * ssl, char * dhm_P, char * dhm_G)

Set the Diffie-Hellman public P and G values, provided as hexadecimal strings (server-side only).

Returns:

0 if successful, or 1 if the values could not be read

Definition at line 1291 of file `ssl_tls.c`.

Referenced by `main()`.

8.126.2.24 void ssl_set_endpoint (ssl_context * ssl, int endpoint)

Set the current endpoint type, `SSL_IS_CLIENT` or `SSL_IS_SERVER`.

Definition at line 1244 of file `ssl_tls.c`.

References `ssl_context::endpoint`.

Referenced by `main()`.

8.126.2.25 void ssl_set_io_files (ssl_context * ssl, int read_fd, int write_fd)

Set the read and write file descriptors.

Definition at line 1262 of file `ssl_tls.c`.

Referenced by `main()`.

8.126.2.26 void ssl_set_rng_func (ssl_context * ssl, int(*) (void *) rng_f, void * rng_d)

Set the random number generator function.

Definition at line 1254 of file ssl_tls.c.

Referenced by main().

8.126.2.27 void ssl_set_rsa_cert (ssl_context * ssl, x509_cert * own_cert, rsa_context * own_key)

Set own certificate and private RSA key.

Definition at line 1279 of file ssl_tls.c.

Referenced by main().

8.126.2.28 void ssl_set_sidtable (ssl_context * ssl, unsigned char * sidtable)

Set the global session ID table (server-side only).

Definition at line 1286 of file ssl_tls.c.

Referenced by main().

8.126.2.29 int ssl_write (ssl_context * ssl, unsigned char * buf, int len)

Write 'len' application data bytes.

Returns:

0 if successful, ERR_NET_WOULD_BLOCK (only when the socket is set to non-blocking), or a specific SSL error code.

Note:

When the socket is set to non-blocking and this function returns ERR_NET_WOULD_BLOCK, it should be called again with the **same** arguments until it returns 0.

Definition at line 1422 of file ssl_tls.c.

References ssl_context::out_msg, ssl_context::out_msglen, ssl_context::out_msgtype, ssl_context::out_uoff, ssl_handshake(), SSL_MAX_CONTENT_LEN, SSL_MSG_APPLICATION_DATA, and ssl_write_record().

Referenced by main().

8.126.2.30 int ssl_write_certificate (ssl_context * ssl)

Definition at line 843 of file ssl_tls.c.

References ssl_context::client_auth, ssl_context::endpoint, ERR_SSL_CERTIFICATE_REQUIRED, ERR_SSL_CERTIFICATE_TOO_LARGE, _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_HS_CERTIFICATE, SSL_IS_CLIENT, SSL_MAX_CONTENT_LEN, SSL_MSG_ALERT, SSL_MSG_HANDSHAKE, ssl_write_record(), SSLV3_MINOR_VERSION, and ssl_context::state.

Referenced by ssl_client_start(), and ssl_server_start().

8.126.2.31 int ssl_write_change_cipher_spec (ssl_context * ssl)

Definition at line 1019 of file ssl_tls.c.

Referenced by ssl_client_start(), and ssl_server_start().

8.126.2.32 int ssl_write_finished (ssl_context * ssl)

Definition at line 1117 of file ssl_tls.c.

Referenced by ssl_client_start(), and ssl_server_start().

8.126.2.33 int ssl_write_record (ssl_context * ssl, int do_crypt)

Definition at line 645 of file ssl_tls.c.

References ssl_context::hs_md5, ssl_context::hs_sha1, ssl_context::major_ver, md5_update(), ssl_context::minor_ver, net_send(), ssl_context::out_hdr, ssl_context::out_left, ssl_context::out_msg, ssl_context::out_msglen, ssl_context::out_msgtype, sha1_update(), ssl_encrypt_buf(), SSL_MSG_HANDSHAKE, and ssl_context::write_fd.

Referenced by ssl_close_notify(), ssl_write(), ssl_write_certificate(), ssl_write_certificate_request(), ssl_write_certificate_verify(), ssl_write_client_hello(), ssl_write_client_key_exchange(), ssl_write_server_hello(), ssl_write_server_hello_done(), and ssl_write_server_key_exchange().

8.126.2.34 static void 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, and sha1_hmac().

Referenced by ssl_derive_keys().

8.126.3 Variable Documentation**8.126.3.1 const char _ssl_tls_src[] = "_ssl_tls_src" [static]**

Definition at line 1468 of file ssl_tls.c.

8.126.3.2 int ssl_default_ciphers[]

Initial value:

```
{
    TLS1_EDH_RSA_AES_256_SHA,

    SSL3_EDH_RSA_DES_168_SHA,

    TLS1_RSA_AES_256_SHA,
```

```
    SSL3_RSA_DES_168_SHA,  
  
    SSL3_RSA_RC4_128_SHA,  
    SSL3_RSA_RC4_128_MD5,  
  
    0  
}
```

Definition at line 1343 of file ssl_tls.c.

Referenced by main().

8.127 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/timing.c File Reference

```
#include "xyssl/timing.h"
#include <unistd.h>
#include <signal.h>
#include <sys/time.h>
#include <time.h>
```

Data Structures

- struct [_hr_time](#)

Defines

- [#define _CRT_SECURE_NO_DEPRECATED 1](#)

Functions

- unsigned long [hardclock](#) (void)
Return the CPU cycle counter value.
- unsigned long [set_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.

Variables

- static int [hardclock_init](#) = 0
- static struct timeval [tv_init](#)
- static const char [_timing_src](#) [] = "_timing_src"
- int [alarmed](#) = 0

8.127.1 Define Documentation

8.127.1.1 [#define _CRT_SECURE_NO_DEPRECATED 1](#)

Definition at line 22 of file timing.c.

8.127.2 Function Documentation

8.127.2.1 unsigned long hardclock (void)

Return the CPU cycle counter value.

Definition at line 135 of file timing.c.

References tv_init.

Referenced by main().

8.127.2.2 void set_alarm (int *seconds*)

Setup an alarm clock.

Parameters:

seconds delay before the "alarmed" flag is set

Definition at line 227 of file timing.c.

References sighandler().

Referenced by main().

8.127.2.3 unsigned long set_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 201 of file timing.c.

References _hr_time::start.

8.127.2.4 static void sighandler (int *signum*) [static]

Definition at line 221 of file timing.c.

Referenced by set_alarm().

8.127.3 Variable Documentation

8.127.3.1 const char _timing_src[] = "_timing_src" [static]

Definition at line 158 of file timing.c.

8.127.3.2 int alarmed = 0

Definition at line 160 of file timing.c.

Referenced by main().

8.127.3.3 `int hardclock_init = 0` `[static]`

Definition at line 132 of file timing.c.

8.127.3.4 `struct timeval tv_init` `[static]`

Definition at line 133 of file timing.c.

Referenced by `hardclock()`.

8.128 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/x509read.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include "xyssl/x509.h"
#include "xyssl/base64.h"
#include "xyssl/des.h"
#include "xyssl/sha1.h"
#include "xyssl/md5.h"
#include "xyssl/md4.h"
#include "xyssl/md2.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

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 x509_add_certs` (`x509_cert` *chain, unsigned char *buf, `int` buflen)

Parse one or more certificates and add them to the chain.
- `int x509_read_crtfile` (`x509_cert` *chain, char *path)

Load one or more certificates and add them to the chain.
- static `int x509_des3_getiv` (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 pwrlen)
- int `x509_parse_key` (rsa_context *rsa, unsigned char *buf, int buflen, unsigned char *pwd, int pwrlen)
Parse a private RSA key.
- int `x509_read_keyfile` (rsa_context *rsa, char *path, char *pwd)
Load and parse a private RSA key.
- int `x509_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 * `x509_cert_info` (x509_cert *crt)
Return an informational string about the certificate, or NULL if memory allocation failed.
- int `x509_is_cert_expired` (x509_cert *crt)
Return 0 if the certificate is still valid, or BADCERT_HAS_EXPIRED.
- static void `x509_hash` (unsigned char *in, int len, int alg, unsigned char *out)
- int `x509_verify_cert` (x509_cert *crt, x509_cert *trust_ca, char *cn, int *flags)
Verify the certificate signature.
- void `x509_free_cert` (x509_cert *crt)
Unallocate all certificate data.
- int `x509_self_test` (int verbose)
Checkup routine.

Variables

- static const char `_x509_read_src` [] = "_x509read_src"

8.128.1 Define Documentation

8.128.1.1 #define _CRT_SECURE_NO_DEPRECATED 1

Definition at line 33 of file x509read.c.

8.128.2 Function Documentation

8.128.2.1 static int `asn1_get_bool` (unsigned char **p, unsigned char *end, int *val) [static]

Definition at line 113 of file x509read.c.

References `ASN1_BOOLEAN`, `asn1_get_tag()`, and `ERR_ASN1_INVALID_LENGTH`.

Referenced by `x509_get_ext()`.

8.128.2.2 static int asn1_get_int (unsigned char *p*, unsigned char * *end*, int * *val*)** [static]

Definition at line 131 of file x509read.c.

References `asn1_get_tag()`, `ASN1_INTEGER`, and `ERR_ASN1_INVALID_LENGTH`.

Referenced by `x509_get_ext()`, `x509_get_version()`, and `x509_parse_key()`.

8.128.2.3 static int asn1_get_len (unsigned char *p*, unsigned char * *end*, int * *len*)** [static]

Definition at line 57 of file x509read.c.

References `ERR_ASN1_INVALID_LENGTH`, and `ERR_ASN1_OUT_OF_DATA`.

Referenced by `asn1_get_tag()`, `x509_get_name()`, and `x509_get_serial()`.

8.128.2.4 static int asn1_get_mpi (unsigned char *p*, unsigned char * *end*, mpi * *X*)** [static]

Definition at line 154 of file x509read.c.

References `asn1_get_tag()`, `ASN1_INTEGER`, and `mpi_read_binary()`.

Referenced by `x509_get_pubkey()`, and `x509_parse_key()`.

8.128.2.5 static int asn1_get_tag (unsigned char *p*, unsigned char * *end*, int * *len*, int *tag*)**
[static]

Definition at line 98 of file x509read.c.

References `asn1_get_len()`, `ERR_ASN1_OUT_OF_DATA`, and `ERR_ASN1_UNEXPECTED_TAG`.

Referenced by `asn1_get_bool()`, `asn1_get_int()`, `asn1_get_mpi()`, `x509_add_certs()`, `x509_get_alg()`, `x509_get_dates()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_sig()`, `x509_get_uid()`, `x509_get_version()`, and `x509_parse_key()`.

8.128.2.6 int x509_add_certs (x509_cert * *chain*, unsigned char * *buf*, int *buflen*)

Parse one or more certificates and add them to the chain.

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 643 of file x509read.c.

References `ASN1_CONSTRUCTED`, `asn1_get_tag()`, `ASN1_SEQUENCE`, `base64_decode()`, `_x509_cert::ca_istrue`, `rsa_context::E`, `ERR_ASN1_LENGTH_MISMATCH`, `ERR_BASE64_INVALID_CHARACTER`, `ERR_X509_CERT_INVALID_FORMAT`, `ERR_X509_CERT_INVALID_PEM`, `ERR_X509_CERT_SIG_MISMATCH`, `ERR_X509_CERT_UNKNOWN_SIG_ALG`, `ERR_X509_CERT_UNKNOWN_VERSION`, `_x509_cert::issuer`, `_x509_cert::issuer_id`, `_x509_cert::issuer_raw`,

rsa_context::len, _x509_buf::len, _x509_cert::max_pathlen, mpi_msb(), 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_add_certs(), x509_free_cert(), x509_get_alg(), x509_get_dates(), x509_get_ext(), x509_get_name(), x509_get_pubkey(), x509_get_serial(), x509_get_sig(), x509_get_uid(), and x509_get_version().

Referenced by main(), x509_add_certs(), and x509_read_crtfile().

8.128.2.7 char* x509_cert_info (x509_cert * crt)

Return an informational string about the certificate, or NULL if memory allocation failed.

Definition at line 1388 of file x509read.c.

References _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, x509_dn_gets(), and _x509_time::year.

8.128.2.8 static void x509_des3_decrypt (unsigned char des3_iv[8], unsigned char * buf, int buflen, unsigned char * pwd, int pwrlen) [static]

Definition at line 1035 of file x509read.c.

References des3_cbc_decrypt(), des3_set_3keys(), md5_finish(), md5_starts(), and md5_update().

Referenced by x509_parse_key().

8.128.2.9 static int x509_des3_getiv (unsigned char * s, unsigned char iv[8]) [static]

Definition at line 1010 of file x509read.c.

References ERR_X509_KEY_INVALID_ENC_IV.

Referenced by x509_parse_key().

8.128.2.10 int x509_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 1305 of file x509read.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 x509_cert_info().

8.128.2.11 void x509_free_cert (x509_cert * crt)

Unallocate all certificate data.

Definition at line 1595 of file x509read.c.

References `_x509_cert::issuer`, `_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()`, and `x509_add_certs()`.

8.128.2.12 `static int x509_get_alg (unsigned char ** p, unsigned char * end, x509_buf * alg)`
[static]

Definition at line 234 of file `x509read.c`.

References `ASN1_CONSTRUCTED`, `asn1_get_tag()`, `ASN1_NULL`, `ASN1_OID`, `ASN1_SEQUENCE`, `ERR_ASN1_LENGTH_MISMATCH`, `ERR_X509_CERT_INVALID_ALG`, `_x509_buf::len`, `_x509_buf::p`, and `_x509_buf::tag`.

Referenced by `x509_add_certs()`, and `x509_get_pubkey()`.

8.128.2.13 `static int x509_get_dates (unsigned char ** p, unsigned char * end, x509_time * from, x509_time * to)` [static]

Definition at line 363 of file `x509read.c`.

References `ASN1_CONSTRUCTED`, `asn1_get_tag()`, `ASN1_SEQUENCE`, `ASN1_UTCTIME`, `_x509_time::day`, `ERR_ASN1_LENGTH_MISMATCH`, `ERR_X509_CERT_INVALID_DATE`, `_x509_time::hour`, `_x509_time::min`, `_x509_time::mon`, `_x509_time::sec`, and `_x509_time::year`.

Referenced by `x509_add_certs()`.

8.128.2.14 `static int x509_get_ext (unsigned char ** p, unsigned char * end, x509_buf * ext, int * ca_istrue, int * max_pathlen)` [static]

Definition at line 535 of file `x509read.c`.

References `ASN1_CONSTRUCTED`, `ASN1_CONTEXT_SPECIFIC`, `asn1_get_bool()`, `asn1_get_int()`, `asn1_get_tag()`, `ASN1_OCTET_STRING`, `ASN1_SEQUENCE`, `ERR_ASN1_LENGTH_MISMATCH`, `ERR_ASN1_UNEXPECTED_TAG`, `ERR_X509_CERT_INVALID_EXTENSIONS`, `_x509_buf::len`, `_x509_buf::p`, and `_x509_buf::tag`.

Referenced by `x509_add_certs()`.

8.128.2.15 `static int x509_get_name (unsigned char ** p, unsigned char * end, x509_name * cur)`
[static]

Definition at line 281 of file `x509read.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`, `ERR_ASN1_LENGTH_MISMATCH`, `ERR_ASN1_OUT_OF_DATA`, `ERR_ASN1_UNEXPECTED_TAG`, `ERR_X509_CERT_INVALID_NAME`, `_x509_buf::len`, `_x509_name::next`, `_x509_name::oid`, `_x509_buf::p`, `_x509_buf::tag`, and `_x509_name::val`.

Referenced by `x509_add_certs()`.

8.128.2.16 `static int x509_get_pubkey (unsigned char **p, unsigned char * end, x509_buf *
pk_alg_oid, mpi * N, mpi * E)` `[static]`

Definition at line 426 of file x509read.c.

References ASN1_BIT_STRING, ASN1_CONSTRUCTED, asn1_get_mpi(), asn1_get_tag(), ASN1_SEQUENCE, ERR_ASN1_LENGTH_MISMATCH, ERR_ASN1_OUT_OF_DATA, ERR_X509_CERT_INVALID_PUBKEY, ERR_X509_CERT_UNKNOWN_PK_ALG, _x509_buf::len, OID_PKCS1_RSA, _x509_buf::p, and x509_get_alg().

Referenced by x509_add_certs().

8.128.2.17 `static int x509_get_serial (unsigned char **p, unsigned char * end, x509_buf * serial)`
`[static]`

Definition at line 203 of file x509read.c.

References ASN1_CONTEXT_SPECIFIC, asn1_get_len(), ASN1_INTEGER, ASN1_PRIMITIVE, ERR_ASN1_OUT_OF_DATA, ERR_ASN1_UNEXPECTED_TAG, ERR_X509_CERT_INVALID_SERIAL, _x509_buf::len, _x509_buf::p, and _x509_buf::tag.

Referenced by x509_add_certs().

8.128.2.18 `static int x509_get_sig (unsigned char **p, unsigned char * end, x509_buf * sig)`
`[static]`

Definition at line 481 of file x509read.c.

References ASN1_BIT_STRING, asn1_get_tag(), ERR_X509_CERT_INVALID_SIGNATURE, _x509_buf::len, _x509_buf::p, and _x509_buf::tag.

Referenced by x509_add_certs().

8.128.2.19 `static int x509_get_uid (unsigned char **p, unsigned char * end, x509_buf * uid, int n)`
`[static]`

Definition at line 506 of file x509read.c.

References ASN1_CONSTRUCTED, ASN1_CONTEXT_SPECIFIC, asn1_get_tag(), ERR_ASN1_UNEXPECTED_TAG, _x509_buf::len, _x509_buf::p, and _x509_buf::tag.

Referenced by x509_add_certs().

8.128.2.20 `static int x509_get_version (unsigned char **p, unsigned char * end, int * ver)`
`[static]`

Definition at line 173 of file x509read.c.

References ASN1_CONSTRUCTED, ASN1_CONTEXT_SPECIFIC, asn1_get_int(), asn1_get_tag(), ERR_ASN1_LENGTH_MISMATCH, ERR_ASN1_UNEXPECTED_TAG, and ERR_X509_CERT_INVALID_VERSION.

Referenced by x509_add_certs().

8.128.2.21 `static void x509_hash (unsigned char * in, int len, int alg, unsigned char * out)`
`[static]`

Definition at line 1472 of file x509read.c.

References md2(), md4(), md5(), RSA_MD2, RSA_MD4, RSA_MD5, RSA_SHA1, and sha1().

Referenced by x509_verify_cert().

8.128.2.22 `int x509_is_cert_expired (x509_cert * crt)`

Return 0 if the certificate is still valid, or BADCERT_HAS_EXPIRED.

Definition at line 1449 of file x509read.c.

References BADCERT_HAS_EXPIRED, _x509_time::day, _x509_time::mon, _x509_cert::valid_to, and _x509_time::year.

Referenced by x509_verify_cert().

8.128.2.23 `int x509_parse_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 1074 of file x509read.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, ERR_ASN1_LENGTH_MISMATCH, ERR_BASE64_INVALID_CHARACTER, ERR_X509_FEATURE_UNAVAILABLE, ERR_X509_KEY_INVALID_ENC_IV, ERR_X509_KEY_INVALID_FORMAT, ERR_X509_KEY_INVALID_PEM, ERR_X509_KEY_INVALID_VERSION, ERR_X509_KEY_PASSWORD_MISMATCH, ERR_X509_KEY_PASSWORD_REQUIRED, ERR_X509_KEY_UNKNOWN_ENC_ALG, rsa_context::len, mpi_msb(), rsa_context::N, rsa_context::P, rsa_context::Q, rsa_context::QP, rsa_check_privkey(), rsa_free(), rsa_context::ver, x509_des3_decrypt(), and x509_des3_getiv().

Referenced by main(), and x509_read_keyfile().

8.128.2.24 `int x509_read_crtfile (x509_cert * chain, char * path)`

Load one or more certificates and add them to the chain.

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 971 of file x509read.c.

References f, and x509_add_certs().

8.128.2.25 int x509_read_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 1258 of file x509read.c.

References f, and x509_parse_key().

8.128.2.26 int x509_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 1723 of file x509read.c.

Referenced by main().

8.128.2.27 int x509_verify_cert (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 ERR_X509_SIG_VERIFY_FAILED, in which case *flags will have one or more of the following values set: BADCERT_HAS_EXPIRED – BADCERT_CN_MISMATCH – BADCERT_NOT_TRUSTED

Definition at line 1494 of file x509read.c.

References BADCERT_CN_MISMATCH, BADCERT_NOT_TRUSTED, _x509_cert::ca_istrue, ERR_X509_SIG_VERIFY_FAILED, _x509_cert::issuer_raw, _x509_buf::len, _x509_cert::max_pathlen, _x509_cert::next, _x509_name::next, _x509_name::oid, _x509_buf::p, _x509_cert::rsa, rsa_pkcs1_verify(), _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(), and x509_is_cert_expired().

8.128.3 Variable Documentation

8.128.3.1 `const char _x509_read_src[] = "_x509read_src" [static]`

Definition at line 1644 of file x509read.c.

8.129 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/aes/aesencrypt2.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 _CRT_SECURE_NO_DEPRECATED 1`
- `#define MODE_ENCRYPT 0`
- `#define MODE_DECRYPT 1`
- `#define USAGE`

Functions

- `void scanf_argv (char *prompt, char **arg)`
- `int main (int argc, char *argv[])`

8.129.1 Define Documentation

8.129.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file aesencrypt2.c.

8.129.1.2 `#define MODE_DECRYPT 1`

Definition at line 41 of file aesencrypt2.c.

Referenced by `main()`.

8.129.1.3 `#define MODE_ENCRYPT 0`

Definition at line 40 of file aesencrypt2.c.

Referenced by `main()`.

8.129.1.4 `#define USAGE`

Value:

```
"\n aescrypt2 <mode> <input filename> <output filename> <key>\n" \  
"\n  <mode>: 0 = encrypt, 1 = decrypt\n" \  
"\n  example: aescrypt2 0 file file.aes hex:E76B2413958B00E193\n" \  
"\n"
```

Definition at line 43 of file aescrypt2.c.

Referenced by main().

8.129.2 Function Documentation

8.129.2.1 int main (int *argc*, char * *argv*[])

Definition at line 57 of file aescrypt2.c.

References aes_decrypt(), aes_encrypt(), aes_set_key(), int, MODE_DECRYPT, MODE_ENCRYPT, scanf_argv(), sha2_finish(), sha2_hmac_finish(), sha2_hmac_starts(), sha2_hmac_update(), sha2_starts(), sha2_update(), and USAGE.

8.129.2.2 void scanf_argv (char * *prompt*, char ** *arg*)

Definition at line 49 of file aescrypt2.c.

Referenced by main().

8.130 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/hello.c File Reference

```
#include <stdio.h>
#include "xyssl/md5.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- `int main (void)`

8.130.1 Define Documentation

8.130.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file hello.c.

8.130.2 Function Documentation

8.130.2.1 `int main (void)`

Definition at line 29 of file hello.c.

References `md5()`.

8.131 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/md5sum.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/md5.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- `int md5_wrapper (char *filename, unsigned char *sum)`
- `int md5_print (char *filename)`
- `int md5_check (char *filename)`
- `int main (int argc, char *argv[])`

8.131.1 Define Documentation

8.131.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file md5sum.c.

8.131.2 Function Documentation

8.131.2.1 `int main (int argc, char * argv[])`

Definition at line 131 of file md5sum.c.

References `md5_check()`, and `md5_print()`.

8.131.2.2 `int md5_check (char * filename)`

Definition at line 58 of file md5sum.c.

References `f`, and `md5_wrapper()`.

Referenced by `main()`.

8.131.2.3 `int md5_print (char * filename)`

Definition at line 43 of file md5sum.c.

References `md5_wrapper()`.

Referenced by `main()`.

8.131.2.4 int md5_wrapper (char **filename*, unsigned char **sum*)

Definition at line 30 of file md5sum.c.

References md5_file().

Referenced by md5_check(), and md5_print().

8.132 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/sha1sum.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/sha1.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- `int sha1_wrapper (char *filename, unsigned char *sum)`
- `int sha1_print (char *filename)`
- `int sha1_check (char *filename)`
- `int main (int argc, char *argv[])`

8.132.1 Define Documentation

8.132.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file sha1sum.c.

8.132.2 Function Documentation

8.132.2.1 `int main (int argc, char * argv[])`

Definition at line 131 of file sha1sum.c.

References [sha1_check\(\)](#), and [sha1_print\(\)](#).

8.132.2.2 `int sha1_check (char * filename)`

Definition at line 58 of file sha1sum.c.

References [f](#), and [sha1_wrapper\(\)](#).

Referenced by [main\(\)](#).

8.132.2.3 `int sha1_print (char * filename)`

Definition at line 43 of file sha1sum.c.

References [sha1_wrapper\(\)](#).

Referenced by [main\(\)](#).

8.132.2.4 int sha1_wrapper (char * *filename*, unsigned char * *sum*)

Definition at line 30 of file sha1sum.c.

References sha1_file().

Referenced by sha1_check(), and sha1_print().

8.133 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/sha2sum.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/sha2.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- `int sha2_wrapper (char *filename, unsigned char *sum)`
- `int sha2_print (char *filename)`
- `int sha2_check (char *filename)`
- `int main (int argc, char *argv[])`

8.133.1 Define Documentation

8.133.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file sha2sum.c.

8.133.2 Function Documentation

8.133.2.1 `int main (int argc, char * argv[])`

Definition at line 131 of file sha2sum.c.

References [sha2_check\(\)](#), and [sha2_print\(\)](#).

8.133.2.2 `int sha2_check (char * filename)`

Definition at line 58 of file sha2sum.c.

References [f](#), and [sha2_wrapper\(\)](#).

Referenced by [main\(\)](#).

8.133.2.3 `int sha2_print (char * filename)`

Definition at line 43 of file sha2sum.c.

References [sha2_wrapper\(\)](#).

Referenced by [main\(\)](#).

8.133.2.4 int sha2_wrapper (char * *filename*, unsigned char * *sum*)

Definition at line 30 of file sha2sum.c.

References sha2_file().

Referenced by sha2_check(), and sha2_print().

8.134 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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 _CRT_SECURE_NO_DEPRECATED 1`
- `#define SERVER_NAME "localhost"`
- `#define SERVER_PORT 11999`

Functions

- `int main (void)`

8.134.1 Define Documentation

8.134.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file dh_client.c.

8.134.1.2 `#define SERVER_NAME "localhost"`

Definition at line 35 of file dh_client.c.

Referenced by `main()`.

8.134.1.3 `#define SERVER_PORT 11999`

Definition at line 36 of file dh_client.c.

Referenced by `main()`.

8.134.2 Function Documentation

8.134.2.1 `int main (void)`

Definition at line 38 of file dh_client.c.

References aes_decrypt(), aes_set_key(), dhm_calc_secret(), dhm_free(), dhm_make_public(), dhm_read_params(), f, havege_init(), havege_rand(), rsa_context::len, dhm_context::len, net_close(), net_connect(), net_recv(), net_send(), rsa_free(), rsa_pkcs1_verify(), rsa_read_public(), RSA_SHA1, server_fd, SERVER_NAME, SERVER_PORT, and sha1().

8.135 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/dh_genprime.c File Reference

```
#include <stdio.h>
#include "xyssl/havege.h"
#include "xyssl/bignum.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED` 1
- `#define DH_P_SIZE` 1024
- `#define GENERATOR` "4"

Functions

- `int main` (void)

8.135.1 Define Documentation

8.135.1.1 `#define _CRT_SECURE_NO_DEPRECATED` 1

Definition at line 22 of file dh_genprime.c.

8.135.1.2 `#define DH_P_SIZE` 1024

Definition at line 34 of file dh_genprime.c.

Referenced by main().

8.135.1.3 `#define GENERATOR` "4"

Definition at line 35 of file dh_genprime.c.

Referenced by main().

8.135.2 Function Documentation

8.135.2.1 `int main` (void)

Definition at line 37 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.

8.136 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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 _CRT_SECURE_NO_DEPRECATED 1`
- `#define SERVER_PORT 11999`
- `#define PLAINTEXT "0123456_89ABCDE_"`

Functions

- `int main (void)`

8.136.1 Define Documentation

8.136.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file dh_server.c.

8.136.1.2 `#define PLAINTEXT "0123456_89ABCDE_"`

Definition at line 36 of file dh_server.c.

Referenced by `main()`.

8.136.1.3 `#define SERVER_PORT 11999`

Definition at line 35 of file dh_server.c.

8.136.2 Function Documentation

8.136.2.1 `int main (void)`

Definition at line 38 of file dh_server.c.

References `aes_encrypt()`, `aes_set_key()`, `client_fd`, `dhm_calc_secret()`, `dhm_free()`, `dhm_make_params()`, `dhm_read_public()`, `f`, `dhm_context::G`, `havege_init()`, `havege_rand()`, `dhm_context::len`, `rsa_context::len`,

mpi_read_file(), net_accept(), net_bind(), net_close(), net_recv(), net_send(), dhm_context::P, PLAIN-TEXT, rsa_free(), rsa_pkcs1_sign(), rsa_read_private(), RSA_SHA1, SERVER_PORT, and sha1().

8.137 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/mpi_demo.c File Reference

```
#include <stdio.h>
#include "xyssl/bignum.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- `int main (void)`

8.137.1 Define Documentation

8.137.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file mpi_demo.c.

8.137.2 Function Documentation

8.137.2.1 `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`.

8.138 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_genkey.c File Reference

```
#include <stdio.h>
#include "xyssl/havege.h"
#include "xyssl/bignum.h"
#include "xyssl/rsa.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`
- `#define KEY_SIZE 1024`
- `#define EXPONENT 65537`

Functions

- `int main (void)`

8.138.1 Define Documentation

8.138.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file `rsa_genkey.c`.

8.138.1.2 `#define EXPONENT 65537`

Definition at line 32 of file `rsa_genkey.c`.

Referenced by `main()`.

8.138.1.3 `#define KEY_SIZE 1024`

Definition at line 31 of file `rsa_genkey.c`.

Referenced by `main()`.

8.138.2 Function Documentation

8.138.2.1 `int main (void)`

Definition at line 34 of file `rsa_genkey.c`.

References `EXPONENT`, `havege_init()`, `havege_rand()`, `KEY_SIZE`, `rsa_free()`, `rsa_gen_key()`, `rsa_write_private()`, and `rsa_write_public()`.

8.139 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_sign.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- `int main (int argc, char *argv[])`

8.139.1 Define Documentation

8.139.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file `rsa_sign.c`.

8.139.2 Function Documentation

8.139.2.1 `int main (int argc, char * argv[])`

Definition at line 31 of file `rsa_sign.c`.

References `f`, `rsa_context::len`, `rsa_pkcs1_sign()`, `rsa_read_private()`, `RSA_SHA1`, and `sha1_file()`.

8.140 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_verify.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- `int main (int argc, char *argv[])`

8.140.1 Define Documentation

8.140.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file `rsa_verify.c`.

8.140.2 Function Documentation

8.140.2.1 `int main (int argc, char * argv[])`

Definition at line 31 of file `rsa_verify.c`.

References `f`, `rsa_context::len`, `rsa_pkcs1_verify()`, `rsa_read_public()`, `RSA_SHA1`, and `sha1_file()`.

8.141 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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 _CRT_SECURE_NO_DEPRECATED 1`
- `#define SERVER_PORT 4433`
- `#define SERVER_NAME "xyssl.org"`
- `#define GET_REQUEST`

Functions

- `int main (void)`

8.141.1 Define Documentation

8.141.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file `ssl_client1.c`.

8.141.1.2 `#define GET_REQUEST`

Value:

```
"GET /hello/ HTTP/1.0\r\n" \
"Host: xyssl.org\r\n\r\n"
```

Definition at line 38 of file `ssl_client1.c`.

Referenced by `main()`.

8.141.1.3 `#define SERVER_NAME "xyssl.org"`

Definition at line 37 of file `ssl_client1.c`.

8.141.1.4 `#define SERVER_PORT 4433`

Definition at line 32 of file `ssl_client1.c`.

8.141.2 Function Documentation

8.141.2.1 `int main (void)`

Definition at line 42 of file `ssl_client1.c`.

References `ERR_NET_WOULD_BLOCK`, `ERR_SSL_PEER_CLOSE_NOTIFY`, `GET_REQUEST`, `havege_init()`, `havege_rand()`, `net_close()`, `net_connect()`, `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_ciphlist()`, `ssl_set_endpoint()`, `ssl_set_io_files()`, `ssl_set_rng_func()`, `SSL_VERIFY_NONE`, and `ssl_write()`.

8.142 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/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 _CRT_SECURE_NO_DEPRECATED 1`
- `#define SERVER_PORT 4433`
- `#define SERVER_NAME "xyssl.org"`
- `#define GET_REQUEST`

Functions

- `int main (void)`

8.142.1 Define Documentation

8.142.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file `ssl_client2.c`.

8.142.1.2 `#define GET_REQUEST`

Value:

```
"GET /hello/ HTTP/1.0\r\n" \
"Host: xyssl.org\r\n\r\n"
```

Definition at line 40 of file `ssl_client2.c`.

8.142.1.3 `#define SERVER_NAME "xyssl.org"`

Definition at line 39 of file `ssl_client2.c`.

8.142.1.4 `#define SERVER_PORT 4433`

Definition at line 34 of file `ssl_client2.c`.

8.142.2 Function Documentation

8.142.2.1 `int main (void)`

Definition at line 44 of file `ssl_client2.c`.

References `BADCERT_CN_MISMATCH`, `BADCERT_HAS_EXPIRED`, `BADCERT_NOT_TRUSTED`, `ERR_SSL_PEER_CLOSE_NOTIFY`, `GET_REQUEST`, `havege_init()`, `havege_rand()`, `net_close()`, `net_connect()`, `rsa_free()`, `server_fd`, `SERVER_NAME`, `SERVER_PORT`, `ssl_close_notify()`, `ssl_default_ciphers`, `ssl_free()`, `ssl_get_cipher_name()`, `ssl_get_verify_result()`, `ssl_handshake()`, `ssl_init()`, `SSL_IS_CLIENT`, `ssl_read()`, `ssl_set_authmode()`, `ssl_set_ca_chain()`, `ssl_set_ciphlist()`, `ssl_set_endpoint()`, `ssl_set_io_files()`, `ssl_set_rng_func()`, `ssl_set_rsa_cert()`, `SSL_VERIFY_OPTIONAL`, `ssl_write()`, `test_cli_cert`, `test_cli_key`, `x509_add_certs()`, `x509_free_cert()`, `x509_parse_key()`, and `xyssl_ca_cert`.

8.143 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/ssl/ssl_server.c File Reference

```
#include <string.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 _CRT_SECURE_NO_DEPRECATED` 1
- `#define HTTP_RESPONSE`

Functions

- `int main` (void)

Variables

- `char * dhm_P`
- `char * dhm_G` = "4"
- `unsigned char session_table` [SSL_SESSION_TBL_LEN]
- `int my_preferred_ciphers` []

8.143.1 Define Documentation

8.143.1.1 `#define _CRT_SECURE_NO_DEPRECATED` 1

Definition at line 22 of file `ssl_server.c`.

8.143.1.2 `#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 38 of file `ssl_server.c`.

Referenced by `http_ParseHeader()`, `main()`, and `mtp_http_Parse()`.

8.143.2 Function Documentation

8.143.2.1 `int main (void)`

Definition at line 75 of file `ssl_server.c`.

References `client_fd`, `dhm_G`, `dhm_P`, `ERR_NET_CONN_RESET`, `ERR_SSL_PEER_CLOSE_NOTIFY`, `havege_init()`, `havege_rand()`, `HTTP_RESPONSE`, `my_preferred_ciphers`, `net_accept()`, `net_bind()`, `net_close()`, `_x509_cert::next`, `rsa_free()`, `session_table`, `ssl_close_notify()`, `ssl_free()`, `ssl_get_cipher_name()`, `ssl_init()`, `SSL_IS_SERVER`, `ssl_read()`, `ssl_server_start()`, `ssl_set_authmode()`, `ssl_set_ca_chain()`, `ssl_set_ciphlist()`, `ssl_set_dhm_vals()`, `ssl_set_endpoint()`, `ssl_set_io_files()`, `ssl_set_rng_func()`, `ssl_set_rsa_cert()`, `ssl_set_sidtable()`, `SSL_VERIFY_NONE`, `ssl_write()`, `test_ca_cert`, `test_srv_cert`, `test_srv_key`, `x509_add_certs()`, `x509_free_cert()`, and `x509_parse_key()`.

8.143.3 Variable Documentation

8.143.3.1 `char* dhm_G = "4"`

Definition at line 57 of file `ssl_server.c`.

Referenced by `main()`.

8.143.3.2 `char* dhm_P`

Initial value:

```
"E4004C1F94182000103D883A448B3F80"
"2CE4B44A83301270002C20D0321CFD00"
"11CCEF784C26A400F43DFB901BCA7538"
"F2C6B176001CF5A0FD16D2C48B1D0C1C"
"F6AC8E1DA6BCC3B4E1F96B0564965300"
"FFA1D0B601EB2800F489AA512C4B248C"
"01F76949A60BB7F00A40B1EAB64BDD48"
"E8A700D60B7F1200FA8E77B0A979DABF"
```

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

Referenced by `main()`.

8.143.3.3 `int my_preferred_ciphers[]`

Initial value:

```
{
    TLS1_EDH_RSA_AES_256_SHA,
    SSL3_EDH_RSA_DES_168_SHA,
    TLS1_RSA_AES_256_SHA,
    SSL3_RSA_DES_168_SHA,
    SSL3_RSA_RC4_128_SHA,
    SSL3_RSA_RC4_128_MD5,
    0
}
```

Definition at line 64 of file `ssl_server.c`.

Referenced by `main()`.

8.143.3.4 unsigned char session_table[SSL_SESSION_TBL_LEN]

Definition at line 59 of file ssl_server.c.

Referenced by main().

8.144 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/test/benchmark.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <stdio.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 _CRT_SECURE_NO_DEPRECATED` 1
- `#define BUFSIZE` 510

Functions

- `int myrand` (void *rng_state)
- `int main` (void)

8.144.1 Define Documentation

8.144.1.1 `#define _CRT_SECURE_NO_DEPRECATED` 1

Definition at line 22 of file benchmark.c.

8.144.1.2 `#define BUFSIZE` 510

Definition at line 39 of file benchmark.c.

Referenced by `main()`.

8.144.2 Function Documentation

8.144.2.1 `int main` (void)

Definition at line 47 of file benchmark.c.

References aes_cbc_encrypt(), aes_set_key(), alarmed, arc4_crypt(), arc4_setup(), BUFSIZE, rsa_context::D, des3_cbc_encrypt(), des3_set_3keys(), des_cbc_encrypt(), des_set_key(), rsa_context::DP, rsa_context::DQ, rsa_context::E, hardclock(), KEY_LEN, rsa_context::len, md4(), md5(), mpi_read_string(), rsa_context::N, rsa_context::P, rsa_context::Q, rsa_context::QP, RSA_D, RSA_DP, RSA_DQ, RSA_E, rsa_free(), RSA_N, RSA_P, rsa_private(), rsa_public(), RSA_Q, RSA_QP, set_alarm(), sha1(), and sha2().

8.144.2.2 int myrand (void * *rng_state*)

Definition at line 41 of file benchmark.c.

8.145 /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/test/selftest.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/md2.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/bignum.h"
#include "xyssl/base64.h"
#include "xyssl/rsa.h"
#include "xyssl/x509.h"
```

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`

Functions

- `int main (int argc, char *argv[])`

8.145.1 Define Documentation

8.145.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 22 of file selftest.c.

8.145.2 Function Documentation

8.145.2.1 `int main (int argc, char * argv[])`

Definition at line 41 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\(\)`, and `x509_self_test\(\)`.

8.146 /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAclMessage.cs File Reference

Namespaces

- namespace [LibMC](#)
- namespace [System](#)
- namespace [System::Collections::Generic](#)
- namespace [System::Text](#)

Data Structures

- class [LibMC::MCAclMessage](#)
Encapsulates ACL messages in the Mobile-C library.

8.146.1 Detailed Description

Defines the MCAclMessage object and its member functions.

Definition in file [MCAclMessage.cs](#).

8.147 /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgency File Reference

Namespaces

- namespace [LibMC](#)
- namespace [System::Runtime::InteropServices](#)

Data Structures

- class [LibMC::MCAgency](#)
Wrapper class for [MCAgency_t](#) structure.

8.147.1 Detailed Description

Defines the MCAgency object and its member functions.

Definition in file [MCAgency.cs](#).

8.148 /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgent.cs File Reference

Namespaces

- namespace [LibMC](#)

Data Structures

- class [LibMC::MCAgent](#)
Wrapper class for MCAgent_t structure.

8.148.1 Detailed Description

Defines the MCAgent object and its member functions.

Definition in file [MCAgent.cs](#).

8.149 /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCExport File Reference

Namespaces

- namespace [LibMC](#)

Data Structures

- class [LibMC::MCAgency](#)
Wrapper class for [MCAgency_t](#) structure.
- struct [LibMC::MCAgency::MCAgency_t](#)
- struct [LibMC::MCAgency::MCAgencyOptions_t](#)
- struct [LibMC::MCAgency::ChOptions_t](#)
ChOptions structures.

8.149.1 Detailed Description

Imports functions, structs, and enums from the Mobile-C library.

Definition in file [MCExports.cs](#).

8.150 /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties File Reference

Namespaces

- namespace [System::Reflection](#)
- namespace [System::Runtime::CompilerServices](#)

8.151 /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties File Reference

Namespaces

- namespace [LibMC::Properties](#)

Data Structures

- class [LibMC::Properties::Settings](#)

8.152 /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Settings.cs File Reference

Namespaces

- namespace [LibMC::Properties](#)

Data Structures

- class [LibMC::Properties::Settings](#)

8.153 /home/dko/projects/mobilec/trunk/src/winconfig.h File Reference

Defines

- #define `PACKAGE_STRING` "MobileC V1.10.0"
- #define `PACKAGE_VERSION` "1.10.0"
- #define `strtok_r`(buf, delim, save_ptr) strtok(buf, delim)

8.153.1 Define Documentation

8.153.1.1 #define `PACKAGE_STRING` "MobileC V1.10.0"

Definition at line 1 of file winconfig.h.

8.153.1.2 #define `PACKAGE_VERSION` "1.10.0"

Definition at line 2 of file winconfig.h.

Referenced by `mtp_http_ComposeMessage()`, and `mtp_http_CreateMessage()`.

8.153.1.3 #define `strtok_r`(buf, delim, save_ptr) strtok(buf, delim)

Definition at line 3 of file winconfig.h.

Referenced by `agent_xml_parse__fill_row_data()`, `message_InitializeFromAgent()`, and `message_Send()`.

8.154 /home/dko/projects/mobilec/trunk/src/xml_compose.c File Reference

```
#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__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 \(void *data, int index, int *extent, ChType_t type, int dim, int extent_index\)](#)

8.154.1 Function Documentation

8.154.1.1 mxml_node_t* agent_xml_compose (agent_p agent)

Definition at line 38 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()`.

8.154.1.2 mxml_node_t* agent_xml_compose__agent_code (agent_p agent, int index)

Definition at line 498 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()`.

8.154.1.3 mxml_node_t* agent_xml_compose__agent_data (agent_p agent)

Definition at line 142 of file xml_compose.c.

References agent_xml_compose__home(), agent_xml_compose__name(), agent_xml_compose__owner(), agent_xml_compose__tasks(), MXML_ADD_AFTER, mxmlAdd(), mxmlNewElement(), and node.

Referenced by agent_xml_compose__mobile_agent().

8.154.1.4 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 549 of file xml_compose.c.

References agent_xml_compose__create_row_nodes(), 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().

8.154.1.5 mxml_node_t* agent_xml_compose__data (agent_p agent, int index, interpreter_variable_data_t * interp_variable)

Definition at line 424 of file xml_compose.c.

References agent_xml_compose__row(), interpreter_variable_data_s::array_dim, 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().

8.154.1.6 mxml_node_t* agent_xml_compose__gaf_message (agent_p agent)

Definition at line 58 of file xml_compose.c.

References agent_xml_compose__message(), MXML_ADD_AFTER, mxmlAdd(), mxmlNewElement(), and node.

Referenced by agent_xml_compose().

8.154.1.7 mxml_node_t* agent_xml_compose__home (agent_p agent)

Definition at line 234 of file xml_compose.c.

References agent_s::home, mxmlNewElement(), mxmlNewText(), and node.

Referenced by agent_xml_compose__agent_data().

8.154.1.8 mxml_node_t* agent_xml_compose__message (agent_p agent)

Definition at line 77 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()`.

8.154.1.9 `mxml_node_t* agent_xml_compose__mobile_agent (agent_p agent)`

Definition at line 121 of file `xml_compose.c`.

References `agent_xml_compose__agent_data()`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlNewElement()`, and `node`.

Referenced by `agent_xml_compose__message()`.

8.154.1.10 `mxml_node_t* agent_xml_compose__name (agent_p agent)`

Definition at line 202 of file `xml_compose.c`.

References `mxmlNewElement()`, `mxmlNewText()`, `agent_s::name`, and `node`.

Referenced by `agent_xml_compose__agent_data()`.

8.154.1.11 `mxml_node_t* agent_xml_compose__owner (agent_p agent)`

Definition at line 218 of file `xml_compose.c`.

References `mxmlNewElement()`, `mxmlNewText()`, `node`, and `agent_s::owner`.

Referenced by `agent_xml_compose__agent_data()`.

8.154.1.12 `mxml_node_t* agent_xml_compose__row (interpreter_variable_data_t * interp_variable, int index)`

Definition at line 526 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()`.

8.154.1.13 `mxml_node_t* agent_xml_compose__task (agent_p agent, int index)`

Definition at line 307 of file `xml_compose.c`.

References `agent_task_s::agent_return_data`, `agent_task_s::agent_variable_list`, `agent_xml_compose__data()`, `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_task_s::task_completed`, `agent_datastate_s::tasks`, and `agent_task_s::var_name`.

Referenced by `agent_xml_compose__tasks()`.

8.154.1.14 mxml_node_t* agent_xml_compose__tasks (agent_p *agent*)

Definition at line 250 of file xml_compose.c.

References agent_xml_compose__agent_code(), agent_xml_compose__task(), 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().

8.155 /home/dko/projects/mobilec/trunk/src/xml_helper.c File Reference

```
#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`)
- const 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`)

8.155.1 Function Documentation

8.155.1.1 const char* whitespace_cb (mxml_node_t * node, int where)

Definition at line 246 of file `xml_helper.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()`.

8.155.1.2 mxml_node_t* xml_find_sibling (const mxml_node_t * node, const char * sibling_name)

Definition at line 46 of file `xml_helper.c`.

References `MXML_NO_DESCEND`, `mxmlFindElement()`, `mxml_node_s::parent`, and `xml_get_element_name()`.

8.155.1.3 char* xml_get_cdata (const mxml_node_t * node)

Definition at line 68 of file `xml_helper.c`.

References `CHECK_NULL`, `MXML_ELEMENT`, `mxml_node_s::type`, and `xml_get_element_name()`.

Referenced by `xml_get_text()`.

8.155.1.4 mxmml_node_t* xml_get_child (const mxmml_node_t * *node*, const char * *child_name*, int *descend*)

Definition at line 101 of file xml_helper.c.

References mxmmlFindElement().

Referenced by agent_return_xml_parse(), agent_xml_parse__agent_data(), agent_xml_parse__data(), agent_xml_parse__mobile_agent(), and xml_get_deep_child().

8.155.1.5 mxmml_node_t* xml_get_deep_child (const mxmml_node_t * *parent*, const char ** *child_path*)

Definition at line 120 of file xml_helper.c.

References MXML_NO_DESCEND, node, and xml_get_child().

8.155.1.6 const char* xml_get_element_name (const mxmml_node_t * *node*)

Definition at line 214 of file xml_helper.c.

References mxmml_value_u::element, MXML_ELEMENT, mxmml_value_s::name, mxmml_node_s::type, and mxmml_node_s::value.

Referenced by agent_xml_parse__data(), agent_xml_parse__mobile_agent(), agent_xml_parse__row(), xml_find_sibling(), and xml_get_cdata().

8.155.1.7 mxmml_node_t* xml_get_next_element (const mxmml_node_t * *node*)

Definition at line 134 of file xml_helper.c.

References MXML_ELEMENT, mxmml_node_s::next, and mxmml_node_s::type.

8.155.1.8 char* xml_get_text (const mxmml_node_t * *node*)

Definition at line 152 of file xml_helper.c.

References CHECK_NULL, mxmml_node_s::child, mxmml_value_u::element, MXML_ELEMENT, MXML_TEXT, mxmml_value_s::name, mxmml_node_s::next, mxmml_text_s::string, mxmml_value_u::text, mxmml_node_s::type, mxmml_node_s::value, and xml_get_cdata().

Referenced by agent_xml_parse__agent_code(), agent_xml_parse__home(), agent_xml_parse__name(), and agent_xml_parse__owner().

8.155.1.9 mxmml_node_t* xml_new_cdata (mxmml_node_t * *parent*, const char * *text*)

Definition at line 227 of file xml_helper.c.

References CHECK_NULL, mxmmlNewElement(), and node.

Referenced by agent_xml_compose__agent_code().

8.156 /home/dko/projects/mobilec/trunk/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__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)`

8.156.1 Define Documentation

8.156.1.1 `#define _XOPEN_SOURCE 600`

Definition at line 35 of file `xml_parser.c`.

8.156.2 Function Documentation

8.156.2.1 `error_code_t agent_return_xml_parse (agent_p agent)`

Definition at line 820 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`.

8.156.2.2 `error_code_t agent_xml_parse (agent_p agent)`

Definition at line 49 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()`.

8.156.2.3 `error_code_t agent_xml_parse__agent_code (agent_p agent, int index, xml_parser_p xml_parser)`

Definition at line 779 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()`.

8.156.2.4 `error_code_t agent_xml_parse__agent_data (agent_p agent, xml_parser_p xml_parser)`

Definition at line 90 of file `xml_parser.c`.

References `agent_xml_parse__home()`, `agent_xml_parse__name()`, `agent_xml_parse__owner()`, `agent_xml_parse__tasks()`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `xml_get_child()`.

Referenced by `agent_xml_parse__mobile_agent()`.

8.156.2.5 `error_code_t agent_xml_parse__data (agent_p agent, xml_parser_p xml_parser, int index)`

Definition at line 454 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()`.

8.156.2.6 void agent_xml_parse__fill_row_data (void * *data*, ChType_t *type*, int * *extent*, const mxml_node_t * *node*, int * *index*)

Definition at line 663 of file xml_parser.c.

References agent_xml_parse__fill_row_data(), 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().

8.156.2.7 error_code_t agent_xml_parse__home (agent_p *agent*, xml_parser_p *xml_parser*)

Definition at line 198 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().

8.156.2.8 error_code_t agent_xml_parse__mobile_agent (agent_p *agent*, xml_parser_p *xml_parser*)

Definition at line 62 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().

8.156.2.9 error_code_t agent_xml_parse__name (agent_p *agent*, xml_parser_p *xml_parser*)

Definition at line 145 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().

8.156.2.10 error_code_t agent_xml_parse__owner (agent_p *agent*, xml_parser_p *xml_parser*)

Definition at line 171 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().

8.156.2.11 error_code_t agent_xml_parse__row (interpreter_variable_data_t * *interp_variable*, xml_parser_p *xml_parser*, int *index*)

Definition at line 599 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().

8.156.2.12 error_code_t agent_xml_parse__task (agent_p agent, xml_parser_p xml_parser, int index)

Definition at line 362 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::server_name, agent_task_s::task_completed, agent_datastate_s::tasks, and agent_task_s::var_name.

Referenced by agent_xml_parse__tasks().

8.156.2.13 error_code_t agent_xml_parse__tasks (agent_p agent, xml_parser_p xml_parser)

Definition at line 224 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(), 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().

8.156.2.14 error_code_t message_xml_parse (message_p message)

Definition at line 855 of file xml_parser.c.

Referenced by acc_Thread(), and message_InitializeFromConnection().

8.156.2.15 error_code_t message_xml_parse__message (message_p message, xml_parser_p xml_parser)

Definition at line 910 of file xml_parser.c.

Chapter 9

Mobile-C Example Documentation

9.1 LibMCConsole/Program.cs

Basic Mobile-C console demo program

```
using System;
using System.Collections.Generic;
using System.Text;
using LibMC;
using System.Threading;

namespace LibMCConsole
{
    class Program
    {
        public static MCAgency Agency = new MCAgency();
        public static int localport = 5050;
        public static int remoteport = 5051;

        static void Main(string[] args)
        {
            int temp;
            MCAgent agent;

            Console.WriteLine("Starting Mobile-C on port " +
                localport.ToString() + ".\n");

            Agency.Port = localport;
            temp = Agency.Initialize();
            if (temp != 0)
                Console.WriteLine("Initialize: " + temp.ToString() + "\n");

            while (true)
            {
                Thread.Sleep(5000);
                try
                {
                    agent = Agency.FindAgentByName("persistent1");
                    Console.WriteLine("Found agent\n" + agent.ToString() + ".");
                    temp = agent.CallAgentFunc("func", IntPtr.Zero, IntPtr.Zero);
                    Console.WriteLine("CallAgentFunc() returned " +
                        temp.ToString() + ".");
                    temp = agent.TerminateAgent();
                    Console.WriteLine("TerminateAgent() returned " +
                        temp.ToString() + ".");
                }
            }
        }
    }
}
```

```
        catch (Exception e)
        {
            Console.WriteLine("Exception: " + e.Message);
        }
    }
}
```

9.2 LibMCCppEx/LibMCCppEx.cpp

```
#pragma once

#include <windows.h>

namespace LibMCCppEx {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;
    using namespace LibMC;
    using namespace System::Diagnostics;

    public ref class Form1 : public System::Windows::Forms::Form
    {
    public:
        Form1(void)
        {
            InitializeComponent();
            //
            //TODO: Add the constructor code here
            //
            Agency = gcnew MCAgency();
            AllocConsole();
        }

    protected:
        ~Form1()
        {
            if (components)
            {
                delete components;
            }
        }

    private: System::Windows::Forms::GroupBox^ groupBox1;
    protected:
    private: System::Windows::Forms::Button^ button1;
    private: System::Windows::Forms::NumericUpDown^ numericUpDown1;
    private: System::Windows::Forms::Label^ label2;
    private: System::Windows::Forms::Label^ label1;
    private: LibMC::MCAgency^ Agency;
    private: System::Windows::Forms::RichTextBox^ richTextBox1;
    private: System::Windows::Forms::GroupBox^ groupBox2;
    private: System::Windows::Forms::TextBox^ textBox1;
    private: System::Windows::Forms::Button^ button2;
    private: System::Windows::Forms::NumericUpDown^ numericUpDown2;
    private: System::Windows::Forms::Label^ label5;
    private: System::Windows::Forms::Label^ label4;
    private: System::Windows::Forms::Button^ button3;
    private: System::Windows::Forms::Label^ label3;

    private:
        System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
#pragma endregion

    private: System::Void button1_Click(System::Object^ sender,
        System::EventArgs^ e)
    {
        int temp;

        if (Agency->State == MCAgency::MCAgencyState::Initialized)
```

```

{
    Agency->Port = (int)numericUpDown1->Value;
    temp = Agency->SetThreadOff(MCAgency::MC_ThreadIndex_e::MC_THREAD_CP);
    if (temp != 0)
        richTextBox1->Text += "SetThreadOff: " + temp.ToString() + "\n";
    temp = Agency->Initialize();
    if (temp != 0)
        richTextBox1->Text += "Initialize: " + temp.ToString() + "\n";
    else
    {
        richTextBox1->Text += "Started agency\n";
        button1->Text = "Halt Agency";
    }
}
else if (Agency->State == MCAgency::MCAgencyState::Running)
{
    temp = Agency->HaltAgency();
    if (temp != 0)
        richTextBox1->Text += "Error halting agency: " +
            temp.ToString() + "\n";
    else
        button1->Text = "Resume Agency";
}
else if (Agency->State == MCAgency::MCAgencyState::Halted)
{
    temp = Agency->ResumeAgency();
    if (temp != 0)
        richTextBox1->Text += "Error resuming agency: " +
            temp.ToString() + "\n";
    else
        button1->Text = "Halt Agency";
}
else
{
    richTextBox1->Text += "Error: Agency is in an unknown state: " +
        Agency->State.ToString();
}
label2->Text = Agency->State.ToString();
}
private: System::Void Form1_Load(System::Object^ sender, System::EventArgs^ e) {
    label2->Text = Agency->State.ToString();
}
private: System::Void Form1_FormClosing(System::Object^ sender,
    System::Windows::Forms::FormClosingEventArgs^ e) {
    if ((Agency->State == MCAgency::MCAgencyState::Running) ||
        (Agency->State == MCAgency::MCAgencyState::Halted))
        Agency->End();
}
private: System::Void button2_Click(System::Object^ sender,
    System::EventArgs^ e) {
    OpenFileDialog^ ofd = gcnew OpenFileDialog();
    ofd->Filter = "XML files (*.xml)|*.xml|All files (*.*)|*.*";

    if (ofd->ShowDialog() == System::Windows::Forms::DialogResult::OK)
    {
        String^ filename = ofd->FileName;
        try
        {
            Agency->SendAgentMigrationMessageFile(filename,
                textBox1->Text, (int)numericUpDown2->Value);
        }
        catch (Exception ^ex)
        {
            richTextBox1->Text += "Error sending file: " +
                ex->Message + "\n";
            richTextBox1->Text += "Is the local agency running?\n";
        }
    }
}

```

```

    }
}
private: System::Void button3_Click(System::Object^ sender,
    System::EventArgs^ e) {
    OpenFileDialog^ ofd = gcnew OpenFileDialog();
    ofd->Filter = "XML files (*.xml)|*.xml|All files (*.*)|*.*";

    if (ofd->ShowDialog() == System::Windows::Forms::DialogResult::OK)
    {
        String^ filename = ofd->FileName;
        try
        {
            Agency->SendAgentMigrationMessageFile(filename,
                "localhost", (int)numericUpDown2->Value);
        }
        catch (Exception ^ex)
        {
            richTextBox1->Text += "Error sending file: " +
                ex->Message + "\n";
            richTextBox1->Text += "Is the local agency running?\n";
        }
    }
};
}

```

Demonstrates using LibMC.NET from a VC++ program.

```

// LibMCCppEx.cpp : main project file.

#include "stdafx.h"
#include "Form1.h"

using namespace LibMCCppEx;

[STAThreadAttribute]
int main(array<System::String ^> ^args)
{
    // Enabling Windows XP visual effects before any controls are created
    Application::EnableVisualStyles();
    Application::SetCompatibleTextRenderingDefault(false);

    // Create the main window and run it
    Application::Run(gcnew Form1());
    return 0;
}

```

9.3 LibMCFipaTest/Program.cs

Mobile-C FIPA ACL message demo program.

```
using System;
using System.Collections.Generic;
using System.Text;
using LibMC;
using System.Threading;

/*
 * Steps for using LibMC:
 * 1. Create your desired type of project.
 * 2. Add a reference to the LibMC project.
 * 3. Add "using LibMC;" where necessary.
 * 4. For any XML files you want to use,
 *    set "Copy to output directory" to "Copy Always" or "Copy if newer."
 * 5. Set working directory if desired.
 * 6. Enable unmanaged code debugging if desired.
 */

namespace LibMCFipaTest
{
    class Program
    {
        public static MCAgency Agency = new MCAgency();
        public static int localport = 5050;
        public static String host = "localhost";

        static void Main(string[] args)
        {
            int temp;
            Console.WriteLine("Starting Mobile-C.\n");

            Agency.Port = localport;
            Agency.SetThreadOff(MCAgency.MC_ThreadIndex_e.MC_THREAD_CP);
            temp = Agency.Initialize();
            if (temp != 0)
                Console.WriteLine("Initialize: " + temp.ToString() + "\n");

            Agency.SendAgentMigrationMessageFile("xml/test1.xml", host, localport);
            Agency.SendAgentMigrationMessageFile("xml/test2.xml", host, localport);
            Console.WriteLine("Sleeping while the agents load.");
            Thread.Sleep(3000);

            Console.WriteLine("Press 'enter' to send " +
                "an acl message to the agent.");
            Console.ReadLine();

            MCAclMessage tmp = new MCAclMessage();
            tmp.New();
            tmp.SetPerformative(MCAclMessage.MC_FipaPerformative_e.FIPA_INFORM);

            tmp.SetSender("agency", "http://" + host + ":" +
                localport.ToString() + "/acc");
            tmp.AddReplyTo("mobagent2", "http://" + host + ":" +
                localport.ToString() + "/acc");

            tmp.AddReceiver("mobagent1", "http://" + host + ":" +
                localport.ToString() + "/acc");
            tmp.SetContent("This is content. Yay!");
            Agency.AclSend(tmp);
            tmp.Destroy();

            Thread.Sleep(1000);
            Console.WriteLine("Press 'enter' to exit when messaging is done.");
        }
    }
}
```

```
        Console.ReadLine();  
        Agency.End();  
    }  
}
```

9.4 LibMCGui/Form1.cs

```
using System;
using System.Collections.Generic;
using System.Windows.Forms;

namespace LibMCTest
{
    static class Program
    {
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

Basic Mobile-C Windows Forms demo program

```
using System;
using System.IO;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.Runtime.InteropServices;
using LibMC;

namespace LibMCTest
{
    public partial class Form1 : Form
    {
        private MCAgency Agency = new MCAgency();
        private static int localport = 5051;
        private static int remoteport = 5050;

        public Form1()
        {
            InitializeComponent();
            numericUpDown1.Value = localport;
            numericUpDown2.Value = remoteport;
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            label3.Text = Agency.State.ToString();
        }

        private void Form1_FormClosing(object sender, FormClosingEventArgs e)
        {
            if ((Agency.State == MCAgency.MCAgencyState.Running) ||
                (Agency.State == MCAgency.MCAgencyState.Halted))
                Agency.End();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            OpenFileDialog ofd = new OpenFileDialog();
            ofd.Filter = "XML files (*.xml)|*.xml|All files (*.*)|*.*";

            if (ofd.ShowDialog() == DialogResult.OK)
```



```

        {
            String filename = ofd.FileName;
            try
            {
                Agency.SendAgentMigrationMessageFile(filename,
                    "localhost", Agency.Port);
            }
            catch (Exception ex)
            {
                richTextBox1.Text += "Error sending file: " + ex.Message + "\n";
                richTextBox1.Text += "Is the local agency running?\n";
            }
        }
    }

private void button2_Click(object sender, EventArgs e)
{
    OpenFileDialog ofd = new OpenFileDialog();
    ofd.Filter = "XML files (*.xml)|*.xml|All files (*.*)|*.*";

    if (ofd.ShowDialog() == DialogResult.OK)
    {
        String filename = ofd.FileName;
        try
        {
            Agency.SendAgentMigrationMessageFile(filename,
                textBox1.Text, (int)numericUpDown2.Value);
        }
        catch (Exception ex)
        {
            richTextBox1.Text += "Error sending file: " + ex.Message + "\n";
            richTextBox1.Text += "Is the local agency running?\n";
        }
    }
}

private void button3_Click(object sender, EventArgs e)
{
    int temp;

    if (Agency.State == MCAgency.MCAgencyState.Initialized)
    {
        Agency.Port = (int)numericUpDown1.Value;
        temp = Agency.SetThreadOff(MCAgency.MC_ThreadIndex_e.MC_THREAD_CP);
        if (temp != 0)
            richTextBox1.Text += "SetThreadOff: " + temp.ToString() + "\n";
        temp = Agency.Initialize();
        if (temp != 0)
            richTextBox1.Text += "Initialize: " + temp.ToString() + "\n";
        else
        {
            richTextBox1.Text += "Started agency.\n";
            button3.Text = "Halt Agency";
        }
    }
    else if (Agency.State == MCAgency.MCAgencyState.Running)
    {
        temp = Agency.HaltAgency();
        if (temp != 0)
            richTextBox1.Text += "Error halting agency: " +
                temp.ToString() + "\n";
        else
            button3.Text = "Resume Agency";
    }
    else if (Agency.State == MCAgency.MCAgencyState.Halted)
    {
        temp = Agency.ResumeAgency();
    }
}

```

```
        if (temp != 0)
            richTextBox1.Text += "Error resuming agency: " +
                temp.ToString() + "\n";
        else
            button3.Text = "Halt Agency";
    }
    else
    {
        richTextBox1.Text += "Error: Agency is in an unknown state: " +
            Agency.State.ToString();
    }
    label3.Text = Agency.State.ToString();
}
}
```

9.5 LibMCMiscTest/Program.cs

Demonstrates miscellaneous Mobile-C functions.

```
using System;
using System.Collections.Generic;
using System.Text;
using LibMC;
using System.Threading;

namespace LibMCMiscTest
{
    class Program
    {
        public static MCAgency Agency = new MCAgency();
        public static int localport = 5050;
        public static int remoteport = 5051;

        static void Main(string[] args)
        {
            int temp;

            Console.WriteLine("Starting Mobile-C.\n");

            Agency.Port = localport;
            temp = Agency.Initialize();
            if (temp != 0)
                Console.WriteLine("Initialize: " + temp.ToString() + "\n");
            Agency.ResetSignal();

            Console.WriteLine("Waiting for an agent to arrive.");

            MCAgent agent;
            agent = Agency.WaitRetrieveAgent();

            String xml = agent.GetAgentXMLString();
            Console.Write("Agent XML String:");
            Console.WriteLine(xml);

            String str = agent.RetrieveAgentCode();
            Console.Write("Agent Code:");
            Console.WriteLine(str);
            Agency.ResetSignal();

            Console.WriteLine("\nType \"quit\" to exit...");

            Agency.MainLoop();
        }
    }
}
```

9.6 LibMCVbEx/Form1.vb

Demonstrates using LibMC.NET from a VB program.

```
Public Class Form1
    Dim Agency As LibMC.MCAgency = New LibMC.MCAgency
    Private Sub Form1_Load(ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles MyBase.Load
        Label2.Text = Agency.State.ToString
    End Sub

    Private Sub Form1_FormClosing(ByVal sender As System.Object, _
        ByVal e As System.Windows.Forms.FormClosingEventArgs) _
        Handles MyBase.FormClosing
        If ((Agency.State = LibMC.MCAgency.MCAgencyState.Running) Or _
            (Agency.State = LibMC.MCAgency.MCAgencyState.Halted)) Then
            Agency.End()
        End If
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles Button1.Click
        Dim temp As Integer
        If (Agency.State = LibMC.MCAgency.MCAgencyState.Initialized) Then
            Agency.Port = NumericUpDown1.Value
            temp = Agency.SetThreadOff(LibMC.MCAgency.MC_ThreadIndex_e.MC_THREAD_CP)
            If (temp <> 0) Then
                Console.WriteLine("SetThreadOff: " + temp.ToString())
            End If
            temp = Agency.Initialize()
            If (temp <> 0) Then
                Console.WriteLine("Initialize: " + temp.ToString())
            Else
                Console.WriteLine("Started agency.")
                Button1.Text = "Halt Agency"
            End If
        Else
            If (Agency.State = LibMC.MCAgency.MCAgencyState.Running) Then
                temp = Agency.HaltAgency()
                If (temp <> 0) Then
                    Console.WriteLine("Error halting agency: " + temp.ToString())
                Else
                    Button1.Text = "Resume Agency"
                End If
            Else
                If (Agency.State = LibMC.MCAgency.MCAgencyState.Halted) Then
                    temp = Agency.ResumeAgency()
                    If (temp <> 0) Then
                        Console.WriteLine("Error resuming agency: " + temp.ToString())
                    Else
                        Button1.Text = "Halt Agency"
                    End If
                Else
                    Console.WriteLine("Error: Agency is in an unknown state: " + _
                        Agency.State.ToString())
                End If
            End If
        End If

        Label2.Text = Agency.State.ToString()

    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles Button2.Click
```

```
End Sub

Private Sub Button3_Click(ByVal sender As System.Object, _
    ByVal e As System.EventArgs) Handles Button3.Click

End Sub

End Class
```


Chapter 10

Mobile-C Page Documentation

10.1 MCAclMessage

Examples of commonly used MCAclMessage operations:

Create a new, blank ACL message:

```
MCAclMessage tmp = new MCAclMessage();  
tmp.New();
```

Set the performative field:

```
tmp.SetPerformative(MCAclMessage.MC_FipaPerformative_e.FIPA_INFORM);
```

Set the sender:

```
tmp.SetSender("agency", "http://" + host + ":" +  
    localport.ToString() + "/acc");
```

Add an alternate reply-to field:

```
tmp.AddReplyTo("mobagent2", "http://" + host + ":" +  
    localport.ToString() + "/acc");
```

Add a receiver to the message:

```
tmp.AddReceiver("mobagent1", "http://" + host + ":" +  
    localport.ToString() + "/acc");
```

Set the content of the message:

```
tmp.SetContent("This is content. Yay!");
```

Finally, send and destroy the message:

```
Agency.AclSend(tmp);  
tmp.Destroy();
```

Note that messages contain a pointer to allocated unmanaged memory and need to be disposed of after they are used. The agency creates a copy of the message when it is sent, and therefore the MCAclMessage object is no longer needed.

10.2 Installing LibMC.NET

Installing LibMC.NET is straightforward but involves several steps.

10.2.1 Requirements

In order to use LibMC.NET you will need the following:

- Ch version 6.0.0 or greater from <http://www.softintegration.com/>.
- Embedded Ch version 6.0.0 or greater, also from <http://www.softintegration.com/>.
- Mobile-C 1.10.0 or greater. See Section [Downloading Mobile-C](#) for instructions on how to obtain Mobile-C.
- Visual Studio 2005 or later. Express versions of Visual Studio can be found at <http://www.microsoft.com/express/>.

10.2.2 Downloading Mobile-C

First, you must obtain a version of the Mobile-C source code. If you are reading this, chances are you have already completed this step. If you have not already downloaded the source code, it can be done in one of three ways:

- Download a supported release of Mobile-C. Visit [the Mobile-C website](#) for more information on supported releases.
- Download the latest source (unsupported) from [Sourceforge](#). This will give you the most current version of Mobile-C, but not necessarily the most stable version.
- Check out the latest source code from the SVN repository. This requires that you have a subversion client installed. More information can be found at [this location](#).

10.2.3 Building the Mobile-C Libraries

Once you have obtained the Mobile-C source, please see the Mobile-C User's Guide for information on compiling Mobile-C under Windows. Currently, only the Visual Studio .NET 2005 project is supported for LibMC.NET. Section 2.3 of the User's Guide describes how to compile Mobile-C into a static library. For LibMC.NET, at least one of two configurations are required: the "Debug_DLL" or "Release_DLL" versions. To build either one, select the appropriate configuration (this replaces step 3 in the User's Guide, Section 2.3.1) and build the solution (step 4). Alternatively, you may select "Batch Build" from the "Build" menu and build all four possible configurations.

10.2.4 Install the Mobile-C Libraries

After building the Mobile-C DLL files, the project will automatically copy the files to the system directory. By default, the files are copied to C:/Windows/System32/. If your system is configured differently or you wish to change the installation directory, right-click on the mc_lib_win32 project in the Solution Explorer and select "Properties." In the mc_lib_win32 Property Pages treeview, select "Configuration Properties," then "Build Events," and finally "Post-Build Event." You can then change the "Command Line" field to copy the files to the directory of your choice. If you change the installation directory, be sure that your chosen directory is in the system path and that you remove any other versions of the files. You will also need to execute a "Rebuild" on the project to ensure the files are copied to the new location.

10.2.5 Build LibMC.NET

Once you have built the Mobile-C DLL files, you can build LibMC.NET. Open the LibMC.NET solution file located in the directory you installed or checked out Mobile-C to at src/win32/LibMC.NET/LibMC.Net.sln. From the "Build" menu, select "Rebuild Solution." You may want to build both the "Debug" and "Release" versions, or perform a batch build as described previously.

10.3 Getting Started

LibMC.NET is very easy to use. The demo programs provided with the download are a good place to start. Please see the Examples section for more information.

10.3.1 Build the Demo Programs

The LibMC.NET demo program solution is located in the directory you installed or checked out Mobile-C to at `demos/win32/LibMC.NET/LibMCDemos.sln`. As before, select "Rebuild Solution" from the "Build" menu. Note that the demo program solution contains the LibMC.NET project as well. You may also build LibMC.NET from within the demo program solution.

By default, the LibMCGui demo is selected in the demo program solution. You may run this program by selecting the "Debug" menu then "Start Debugging" or by pressing F5. Other demo programs can be started by right-clicking the project in the Solution Explorer and selecting "Debug" then "Start new instance." The demo programs have their own documentation as well. See the README file in `src/win32/LibMC.NET` or `demos/win32/LibMC.NET` for information on how to build the demo program documentation.

10.4 Using LibMC.NET

This section explains how to use the LibMC.NET class library in your .NET project. Currently, it only describes the process for using the library in a C# console or GUI application. Other languages, such as VB and managed C++, will require similar actions.

10.4.1 Create a Project

First, create the type of project you would like to use from the Visual Studio "Start Page" or the "File" menu. Select the name and location of the project as you would any other project. Second, add a reference to the configuration of LibMC.NET you would like to use. For debugging purposes, the "Debug" configuration is probably best. To add the reference, right click the "References" item in the Solution Explorer for the project you just created. Select the "Browse" tab and navigate to the output directory of the LibMC.NET project. The directory is located at src/win32/LibMC.NET/bin/Configuration/ in the Mobile-C source directory, where Configuration is either "Debug" or "Release." Select the DLL file and click "Ok." The References item in the Solution Explorer should now list "LibMC." Be sure to save the solution at this point.

10.4.2 Using LibMC.NET Classes and Functions

As with any other namespace, you must add the declaration "using LibMC;" to any file you want to have access to the class libraries. Once you have added the using statement, you can declare objects from the library as you normally would declare any other objects. See the example programs for more details.

10.4.3 Other Options

You may want to enable one or more features in your project that can help you use LibMC.NET or debug problems. If you add any XML files to your project, you probably will want to set their properties in the project to copy the files to the output directory. This is done by selecting the file in the Solution Explorer, opening its properties, and setting two fields:

- Set the "Build Action" field to "Content" if it is not already set. This will make the file part of the project should you decide to publish or package it.
- Set the "Copy to output directory" to "Copy if newer" or "Copy always." This will copy the file when you build the project.

There is also one important note regarding XML files in Visual Studio. *Do not create XML files from within Visual Studio.* The Visual Studio XML file template contains a few leading characters that specify the encoding of the file. They are hidden and you will not be able to change them. These characters are not currently supported by Mobile-C and will crash a receiving agency.

To open the project properties, right-click the project in the Solution Explorer and select "Properties." In the "Debug" pane, you may wish to set an alternate working directory for the project if you want easy access to XML files outside of the project. This is useful for debugging, but may result in errors finding files if you package the project or create an installer. In general, it is best to specify all files with full paths because the Mobile-C library loads from a different location than the project. If you would like to be able to debug the Mobile-C library, you should select the "Enable unmanaged code debugging" check box. This will allow you to more easily see any errors that may occur in the unmanaged library, though hopefully none will.

10.5 Common Operations

This section contains examples of commonly used operations for three main [LibMC](#) classes:

- [MCAgency](#) The mobile agent agency.
- [MCAgent](#) Mobile agents.
- [MCAclMessage](#) Agent communication language messages.

For complete programs and more detailed examples, see the Examples section.

10.6 MCAgency

Examples of commonly used MCAgency operations:

Declare an agency as a member of a class:

```
public static MCAgency Agency = new MCAgency();
```

Set the agency's port:

```
int temp = 5051;
Agency.Port = temp;
```

Start an agency:

```
int temp;
temp = Agency.Initialize();
if (temp != 0)
    Console.WriteLine("Initialize: " + temp.ToString());
```

Pause and resume an agency:

```
Agency.HaltAgency();
Agency.ResumeAgency();
```

Turn off the command prompt thread:

```
temp = Agency.SetThreadOff(MCAgency.MC_ThreadIndex_e.MC_THREAD_CP);
if (temp != 0)
    Console.WriteLine("SetThreadOff: " + temp.ToString());
```

Load an agent into a local agency:

```
String filename = "agent.xml";
try
{
    Agency.LoadAgentMigrationMessageFile(filename);
}
catch (Exception ex)
{
    Console.WriteLine("Error loading file: " + ex.Message);
}
```

Ideally, the file name should be specified absolutely.

Send an agent to a remote agency:

```
String filename = "agent.xml";
String ip = "192.168.23.93";
int port = 5051;
try
{
    Agency.SendAgentMigrationMessageFile(filename, ip, port);
}
catch (Exception ex)
{
    Console.WriteLine("Error sending file: " + ex.Message);
}
```

Find an agent by name:

```
MCAgent agent;
try
{
    agent = Agency.FindAgentByName("persistent1");
}
catch (Exception e)
{
    Console.WriteLine("Exception: " + e.Message);
}
```

Wait for an agent to arrive:

```
MCAgent agent;
Agency.ResetSignal();
try
{
    agent = Agency.WaitRetrieveAgent();
}
catch (Exception e)
{
    Console.WriteLine("Exception: " + e.Message);
}
```

Wait indefinitely while an agency runs:

```
Agency.MainLoop();
```

10.7 MCAgent

Examples of commonly used MCAgent operations: Find an agent by name (assumes an MCAgency named Agency):

```
MCAgent agent;
try
{
    agent = Agency.FindAgentByName("persistent1");
}
catch (Exception e)
{
    Console.WriteLine("Exception: " + e.Message);
}
```

Terminate an agent

```
int temp;
try
{
    temp = agent.TerminateAgent();
    Console.WriteLine("TerminateAgent() returned " +
        temp.ToString() + ".");
}
catch (Exception e)
{
    Console.WriteLine("Exception: " + e.Message);
}
```

Print information about an agent:

```
Console.WriteLine(agent.ToString());
Console.WriteLine(agent.GetAgentXMLString());
Console.WriteLine(agent.RetrieveAgentCode());
```

10.8 Todo List

Global **LibMC::MCAgency::SteerControl()** Test MC_SteerControl, MC_Steer.

Global **LibMC::MCAgency::Steer(IntPtr funcptr, IntPtr arg)** Implement MC_Steer

Global **LibMC::MCAgency::RegisterService(MCAgent agent, int agentID, String agentName, String[] serviceNames, int agentType)** Test MC_RegisterService and MC_SearchForService.

Global **LibMC::MCAgency::SearchForService(String searchString, IntPtr agentNames, IntPtr serviceNames, IntPtr agentType)** Implement SearchForService

Global **LibMC::MCAgent::CallAgentFunc(String funcName, IntPtr retval, IntPtr varg)** Find a better way to handle retval and varg

Global **LibMC::MCAgent::GetAgentExecEngine()** Wrap MC_GetAgentExecEngine with an object for the void* pointer return type (Ch interpreter).

Global **LibMC::MCAgent::GetAgentReturnData(int task_num, IntPtr data, IntPtr dim, IntPtr extent)** Implement GetAgentReturnData

10.9 Bug List

Global [LibMC::MCAgency::Steer](#)(IntPtr funcptr, IntPtr arg) MC_Steer is not yet implemented.

Global [LibMC::MCAgency::SearchForService](#)(String searchString, IntPtr agentNames, IntPtr serviceNames, IntPtr agentNames) MC_SearchForService is not yet implemented.

Index

- ~foo_c
 - foo_c, [78](#)
- ~foo_s
 - foo_s, [81](#)
- /home/dko/projects/mobilec/trunk/src/acc.c, [199](#)
- /home/dko/projects/mobilec/trunk/src/agent.c, [202](#)
- /home/dko/projects/mobilec/trunk/src/agent_
datastate.c, [205](#)
- /home/dko/projects/mobilec/trunk/src/agent_
mailbox.c, [206](#)
- /home/dko/projects/mobilec/trunk/src/agent_
return_data.c, [208](#)
- /home/dko/projects/mobilec/trunk/src/agent_task.c,
[210](#)
- /home/dko/projects/mobilec/trunk/src/ams.c, [211](#)
- /home/dko/projects/mobilec/trunk/src/barrier.c, [213](#)
- /home/dko/projects/mobilec/trunk/src/cmd_
prompt.c, [215](#)
- /home/dko/projects/mobilec/trunk/src/connection.c,
[218](#)
- /home/dko/projects/mobilec/trunk/src/data_
structures.c, [219](#)
- /home/dko/projects/mobilec/trunk/src/df.c, [221](#)
- /home/dko/projects/mobilec/trunk/src/dynstring.c,
[225](#)
- /home/dko/projects/mobilec/trunk/src/fipa_acl.c,
[226](#)
- /home/dko/projects/mobilec/trunk/src/fipa_
envelope.c, [238](#)
- /home/dko/projects/mobilec/trunk/src/include/acc.h,
[246](#)
- /home/dko/projects/mobilec/trunk/src/include/agent.h,
[249](#)
- /home/dko/projects/mobilec/trunk/src/include/agent_
datastate.h, [252](#)
- /home/dko/projects/mobilec/trunk/src/include/agent_
lib.h, [254](#)
- /home/dko/projects/mobilec/trunk/src/include/agent_
mailbox.h, [263](#)
- /home/dko/projects/mobilec/trunk/src/include/agent_
task.h, [265](#)
- /home/dko/projects/mobilec/trunk/src/include/ams.h,
[267](#)
- /home/dko/projects/mobilec/trunk/src/include/ap_
queue_template.h, [269](#)
- /home/dko/projects/mobilec/trunk/src/include/barrier.h,
[272](#)
- /home/dko/projects/mobilec/trunk/src/include/cmd_
prompt.h, [275](#)
- /home/dko/projects/mobilec/trunk/src/include/commands.h,
[277](#)
- /home/dko/projects/mobilec/trunk/src/include/commands.x.h,
[279](#)
- /home/dko/projects/mobilec/trunk/src/include/connection.h,
[281](#)
- /home/dko/projects/mobilec/trunk/src/include/data_
structures.h, [283](#)
- /home/dko/projects/mobilec/trunk/src/include/df.h,
[286](#)
- /home/dko/projects/mobilec/trunk/src/include/df_
request.x.h, [292](#)
- /home/dko/projects/mobilec/trunk/src/include/dynstring.h,
[293](#)
- /home/dko/projects/mobilec/trunk/src/include/fipa_
acl.h, [295](#)
- /home/dko/projects/mobilec/trunk/src/include/fipa_
acl_envelope.h, [310](#)
- /home/dko/projects/mobilec/trunk/src/include/host_
id.h, [318](#)
- /home/dko/projects/mobilec/trunk/src/include/interpreter_
variable_data.h, [319](#)
- /home/dko/projects/mobilec/trunk/src/include/libmc.h,
[321](#)
- /home/dko/projects/mobilec/trunk/src/include/macros.h,
[371](#)
- /home/dko/projects/mobilec/trunk/src/include/mc_
error.h, [382](#)
- /home/dko/projects/mobilec/trunk/src/include/mc_
platform.h, [384](#)
- /home/dko/projects/mobilec/trunk/src/include/mc_
rlock.h, [386](#)
- /home/dko/projects/mobilec/trunk/src/include/message.h,
[388](#)
- /home/dko/projects/mobilec/trunk/src/include/mtp_
http.h, [391](#)
- /home/dko/projects/mobilec/trunk/src/include/xml_
compose.h, [397](#)
- /home/dko/projects/mobilec/trunk/src/include/xml_
helper.h, [400](#)

- /home/dko/projects/mobilec/trunk/src/include/xml_ - parser.h, 402
- /home/dko/projects/mobilec/trunk/src/libmc.c, 406
- /home/dko/projects/mobilec/trunk/src/mc_list/list.c, 461
- /home/dko/projects/mobilec/trunk/src/mc_list/list.h, 463
- /home/dko/projects/mobilec/trunk/src/mc_platform.c, 466
- /home/dko/projects/mobilec/trunk/src/mc_rwlock.c, 468
- /home/dko/projects/mobilec/trunk/src/mc_sync/sync_list.c, 470
- /home/dko/projects/mobilec/trunk/src/mc_sync/sync_list.h, 472
- /home/dko/projects/mobilec/trunk/src/message.c, 475
- /home/dko/projects/mobilec/trunk/src/message_queue.c, 478
- /home/dko/projects/mobilec/trunk/src/mtp_http.c, 479
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/config.h, 483
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-attr.c, 487
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-entity.c, 488
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-file.c, 490
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-index.c, 497
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-node.c, 499
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-private.c, 502
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-search.c, 504
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-set.c, 505
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml-string.c, 507
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxml.h, 508
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/mxmldoc.c, 520
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/class.cxx, 524
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/enum.cxx, 525
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/function.cxx, 526
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/test/struct.cxx, 527
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/testmxml.c, 528
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/vcnet/config.h, 484
- /home/dko/projects/mobilec/trunk/src/mxml-2.2.2/vcnet2005/config.h, 485
- /home/dko/projects/mobilec/trunk/src/security/asm.c, 530
- /home/dko/projects/mobilec/trunk/src/security/asm.h, 531
- /home/dko/projects/mobilec/trunk/src/security/asm_message_composer.c, 532
- /home/dko/projects/mobilec/trunk/src/security/asm_message_composer.h, 533
- /home/dko/projects/mobilec/trunk/src/security/asm_message_parser.c, 534
- /home/dko/projects/mobilec/trunk/src/security/asm_message_parser.h, 535
- /home/dko/projects/mobilec/trunk/src/security/asm_node.c, 536
- /home/dko/projects/mobilec/trunk/src/security/asm_node.h, 537
- /home/dko/projects/mobilec/trunk/src/security/mc_dh.c, 538
- /home/dko/projects/mobilec/trunk/src/security/mc_dh.h, 539
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/aes.h, 540
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/arc4.h, 543
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/base64.h, 545
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/bignum.h, 547
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/bn_asm.h, 563
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/certs.h, 565
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/des.h, 567
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/dhm.h, 572
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/havege.h, 577
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/md2.h, 579
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/md4.h, 583
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/md5.h, 587
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/net.h, 592
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/rsa.h, 597

- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/sha1.h, 609
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/sha2.h, 614
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/sha4.h, 619
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/ssl.h, 624
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/ssl_conf.h, 642
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/timing.h, 643
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/include/xyssl/x509.h, 645
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/aes.c, 658
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/arc4.c, 665
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/base64.c, 667
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/bignum.c, 670
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/certs.c, 685
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/des.c, 689
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/dhm.c, 701
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/dsa.c, 705
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/havege.c, 710
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md2.c, 713
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md4.c, 718
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/md5.c, 724
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/net.c, 730
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/rsa.c, 734
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha1.c, 741
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha2.c, 748
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/sha4.c, 755
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_cli.c, 762
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_srv.c, 765
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/ssl_tls.c, 769
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/timing.c, 779
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/library/x509read.c, 782
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/aes/aesencrypt2.c, 791
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/hello.c, 793
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/md5sum.c, 794
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/sha1sum.c, 796
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/hash/sha2sum.c, 798
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/dh_client.c, 800
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/dh_genprime.c, 802
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/dh_server.c, 803
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/mpi_demo.c, 805
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_genkey.c, 806
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_sign.c, 807
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/pkey/rsa_verify.c, 808
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/ssl/ssl_client1.c, 809
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/ssl/ssl_client2.c, 811
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/ssl/ssl_server.c, 813
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/test/benchmark.c, 816
- /home/dko/projects/mobilec/trunk/src/security/xyssl-0.7/programs/test/selftest.c, 818
- /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAcl, 819
- /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAg, 820
- /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAg, 821
- /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCExp, 822
- /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Propert, 823
- /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Propert, 824
- /home/dko/projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Setting, 825
- /home/dko/projects/mobilec/trunk/src/winconfig.h, 826

- /home/dko/projects/mobilec/trunk/src/xml_-
compose.c, [827](#)
- /home/dko/projects/mobilec/trunk/src/xml_-
helper.c, [831](#)
- /home/dko/projects/mobilec/trunk/src/xml_parser.c,
[833](#)
- _CRT_SECURE_NO_DEPRECATED
 - aes.c, [659](#)
 - aesencrypt2.c, [791](#)
 - arc4.c, [665](#)
 - base64.c, [667](#)
 - benchmark.c, [816](#)
 - bignum.c, [672](#)
 - des.c, [690](#)
 - dh_client.c, [800](#)
 - dh_genprime.c, [802](#)
 - dh_server.c, [803](#)
 - dhm.c, [701](#)
 - dsa.c, [706](#)
 - havege.c, [710](#)
 - hello.c, [793](#)
 - md2.c, [714](#)
 - md4.c, [719](#)
 - md5.c, [725](#)
 - md5sum.c, [794](#)
 - mpi_demo.c, [805](#)
 - net.c, [731](#)
 - rsa.c, [735](#)
 - rsa_genkey.c, [806](#)
 - rsa_sign.c, [807](#)
 - rsa_verify.c, [808](#)
 - selftest.c, [818](#)
 - sha1.c, [742](#)
 - sha1sum.c, [796](#)
 - sha2.c, [749](#)
 - sha2sum.c, [798](#)
 - sha4.c, [756](#)
 - ssl_cli.c, [762](#)
 - ssl_client1.c, [809](#)
 - ssl_client2.c, [811](#)
 - ssl_server.c, [813](#)
 - ssl_srv.c, [765](#)
 - ssl_tls.c, [771](#)
 - timing.c, [779](#)
 - x509read.c, [783](#)
- _MC_AclAddReceiver
 - LibMC::MCAgency, [129](#)
- _MC_AclAddReplyTo
 - LibMC::MCAgency, [129](#)
- _MC_AclDestroy
 - LibMC::MCAgency, [129](#)
- _MC_AclNew
 - LibMC::MCAgency, [129](#)
- _MC_AclPost
 - LibMC::MCAgency, [129](#)
- _MC_AclReply
 - LibMC::MCAgency, [129](#)
- _MC_AclRetrieve
 - LibMC::MCAgency, [129](#)
- _MC_AclSend
 - LibMC::MCAgency, [129](#)
- _MC_AclSetContent
 - LibMC::MCAgency, [129](#)
- _MC_AclSetPerformative
 - LibMC::MCAgency, [129](#)
- _MC_AclSetSender
 - LibMC::MCAgency, [129](#)
- _MC_AclWaitRetrieve
 - LibMC::MCAgency, [129](#)
- _MC_AddAgent
 - LibMC::MCAgency, [127](#)
- _MC_BarrierDelete
 - LibMC::MCAgency, [126](#)
- _MC_BarrierInit
 - LibMC::MCAgency, [126](#)
- _MC_CallAgentFunc
 - LibMC::MCAgency, [129](#)
- _MC_ChInitializeOptions
 - LibMC::MCAgency, [124](#)
- _MC_CondBroadcast
 - LibMC::MCAgency, [125](#)
- _MC_CondReset
 - LibMC::MCAgency, [125](#)
- _MC_CondSignal
 - LibMC::MCAgency, [125](#)
- _MC_CondWait
 - LibMC::MCAgency, [125](#)
- _MC_DeleteAgent
 - LibMC::MCAgency, [127](#)
- _MC_End
 - LibMC::MCAgency, [124](#)
- _MC_FindAgentByID
 - LibMC::MCAgency, [127](#)
- _MC_FindAgentByName
 - LibMC::MCAgency, [127](#)
- _MC_GetAgentExecEngine
 - LibMC::MCAgency, [129](#)
- _MC_GetAgentID
 - LibMC::MCAgency, [127](#)
- _MC_GetAgentName
 - LibMC::MCAgency, [129](#)
- _MC_GetAgentNumTasks
 - LibMC::MCAgency, [129](#)
- _MC_GetAgentReturnData
 - LibMC::MCAgency, [129](#)
- _MC_GetAgentStatus
 - LibMC::MCAgency, [129](#)
- _MC_GetAgentType

- LibMC::MCAgency, 129
- _MC_GetAgentXMLString
 - LibMC::MCAgency, 129
- _MC_HaltAgency
 - LibMC::MCAgency, 124
- _MC_Initialize
 - LibMC::MCAgency, 124
- _MC_InitializeAgencyOptions
 - LibMC::MCAgency, 124
- _MC_MainLoop
 - LibMC::MCAgency, 125
- _MC_MutexLock
 - LibMC::MCAgency, 125
- _MC_MutexUnlock
 - LibMC::MCAgency, 125
- _MC_PrintAgentCode
 - LibMC::MCAgency, 129
- _MC_RegisterService
 - LibMC::MCAgency, 127
- _MC_ResetSignal
 - LibMC::MCAgency, 126
- _MC_ResumeAgency
 - LibMC::MCAgency, 124
- _MC_RetrieveAgent
 - LibMC::MCAgency, 127
- _MC_RetrieveAgentCode
 - LibMC::MCAgency, 129
- _MC_SearchForService
 - LibMC::MCAgency, 127
- _MC_SemaphorePost
 - LibMC::MCAgency, 126
- _MC_SemaphoreWait
 - LibMC::MCAgency, 126
- _MC_SendAgentMigrationMessage
 - LibMC::MCAgency, 125
- _MC_SendAgentMigrationMessageFile
 - LibMC::MCAgency, 125
- _MC_SetAgentStatus
 - LibMC::MCAgency, 129
- _MC_SetDefaultAgentStatus
 - LibMC::MCAgency, 124
- _MC_SetThreadOff
 - LibMC::MCAgency, 124
- _MC_SetThreadOn
 - LibMC::MCAgency, 124
- _MC_SetThreadsAllOff
 - LibMC::MCAgency, 124
- _MC_SetThreadsAllOn
 - LibMC::MCAgency, 124
- _MC_Steer
 - LibMC::MCAgency, 126
- _MC_SteerControl
 - LibMC::MCAgency, 126
- _MC_SyncDelete
 - LibMC::MCAgency, 126
- _MC_SyncInit
 - LibMC::MCAgency, 126
- _MC_TerminateAgent
 - LibMC::MCAgency, 129
- _MC_WaitAgent
 - LibMC::MCAgency, 125
- _MC_WaitRetrieveAgent
 - LibMC::MCAgency, 125
- _MC_WaitSignal
 - LibMC::MCAgency, 126
- _XOPEN_SOURCE
 - xml_parser.c, 833
- _aes_src
 - aes.c, 663
- _arc4_src
 - arc4.c, 666
- _base64_src
 - base64.c, 668
- _bignum_src
 - bignum.c, 684
- _certs_src
 - certs.c, 685
- _des_src
 - des.c, 696
- _dhm_src
 - dhm.c, 704
- _havege_src
 - havege.c, 712
- _hr_time, 21
 - start, 21
- _md2_src
 - md2.c, 717
- _md4_src
 - md4.c, 723
- _md5_src
 - md5.c, 729
- _sha1_src
 - sha1.c, 746
- _sha2_src
 - sha2.c, 754
- _sha4_src
 - sha4.c, 760
- _ssl_cli_src
 - ssl_cli.c, 764
- _ssl_srv_src
 - ssl_srv.c, 768
- _ssl_tls_src
 - ssl_tls.c, 777
- _timing_src
 - timing.c, 780
- _x509_buf, 22
 - len, 22
 - p, 22

- tag, 22
- _x509_cert, 23
 - ca_istrue, 25
 - issuer, 24
 - issuer_id, 25
 - issuer_raw, 24
 - max_pathlen, 25
 - next, 25
 - pk_oid, 24
 - raw, 23
 - rsa, 24
 - serial, 23
 - sig, 25
 - sig_oid1, 24
 - sig_oid2, 25
 - subject, 24
 - subject_id, 25
 - subject_raw, 24
 - tbs, 23
 - v3_ext, 25
 - valid_from, 24
 - valid_to, 24
 - version, 23
- _x509_name, 27
 - next, 27
 - oid, 27
 - val, 27
- _x509_read_src
 - x509read.c, 790
- _x509_time, 28
 - day, 28
 - hour, 28
 - min, 28
 - mon, 28
 - sec, 28
 - year, 28
- acc
 - mc_platform_s, 93
- acc.c
 - acc_Destroy, 200
 - acc_Initialize, 200
 - acc_MessageHandlerThread, 200
 - acc_Start, 200
 - acc_Thread, 201
 - BACKLOG, 200
 - listen_Thread, 201
- acc.h
 - acc_Destroy, 246
 - acc_Initialize, 246
 - acc_MessageHandlerThread, 246
 - acc_Start, 247
 - acc_Thread, 247
 - addr, 248
 - client_fd, 248
 - listen_Thread, 247
 - port, 248
 - server_fd, 248
 - STRUCT, 247
- acc_Destroy
 - acc.c, 200
 - acc.h, 246
- acc_Initialize
 - acc.c, 200
 - acc.h, 246
- acc_MessageHandlerThread
 - acc.c, 200
 - acc.h, 246
- acc_Start
 - acc.c, 200
 - acc.h, 247
- acc_Thread
 - acc.c, 201
 - acc.h, 247
- accept_ranges
 - mtp_http_s, 161
- ACCEPTED
 - mtp_http.h, 393
- acl_representation
 - fipa_acl_Param_s, 64
- AcIMsg
 - LibMC::MCAclMessage, 103
- aclmsg_p
 - LibMC::MCAclMessage, 103
- AcIPost
 - LibMC::MCAgent, 143
- AcIRetrieve
 - LibMC::MCAgent, 143
- AcISend
 - LibMC::MCAgency, 123
- AcIWaitRetrieve
 - LibMC::MCAgent, 143
- add_variable
 - mxmldoc.c, 521
- AddAgent
 - LibMC::MCAgency, 122
- addr
 - acc.h, 248
 - connection_s, 51
 - message_s, 154
- AddReceiver
 - LibMC::MCAclMessage, 101
- AddReplyTo
 - LibMC::MCAclMessage, 101
- addresses
 - fipa_agent_identifier_s, 66
- aes.c
 - _CRT_SECURE_NO_DEPRECATED, 659

- [_aes_src](#), 663
- [aes_cbc_decrypt](#), 661
- [aes_cbc_encrypt](#), 661
- [aes_decrypt](#), 662
- [aes_encrypt](#), 662
- [AES_FROUND](#), 659
- [aes_gen_tables](#), 662
- [AES_RROUND](#), 659
- [aes_self_test](#), 662
- [aes_set_key](#), 662
- [FSb](#), 663
- [FT0](#), 663
- [FT1](#), 663
- [FT2](#), 663
- [FT3](#), 663
- [GET_UINT32_BE](#), 660
- [KT0](#), 663
- [KT1](#), 663
- [KT2](#), 664
- [KT3](#), 664
- [MUL](#), 660
- [PUT_UINT32_BE](#), 660
- [RCON](#), 664
- [ROTR8](#), 660
- [RSb](#), 664
- [RT0](#), 664
- [RT1](#), 664
- [RT2](#), 664
- [RT3](#), 664
- [uint32](#), 661
- [uint8](#), 661
- [XTIME](#), 661
- [aes.h](#)
 - [aes_cbc_decrypt](#), 540
 - [aes_cbc_encrypt](#), 541
 - [aes_decrypt](#), 541
 - [aes_encrypt](#), 541
 - [aes_self_test](#), 541
 - [aes_set_key](#), 542
- [aes_cbc_decrypt](#)
 - [aes.c](#), 661
 - [aes.h](#), 540
- [aes_cbc_encrypt](#)
 - [aes.c](#), 661
 - [aes.h](#), 541
- [aes_context](#), 30
 - [drk](#), 30
 - [erk](#), 30
 - [nr](#), 30
- [aes_decrypt](#)
 - [aes.c](#), 662
 - [aes.h](#), 541
- [aes_encrypt](#)
 - [aes.c](#), 662
 - [aes.h](#), 541
- [AES_FROUND](#)
 - [aes.c](#), 659
- [aes_gen_tables](#)
 - [aes.c](#), 662
- [AES_RROUND](#)
 - [aes.c](#), 659
- [aes_self_test](#)
 - [aes.c](#), 662
 - [aes.h](#), 541
- [aes_set_key](#)
 - [aes.c](#), 662
 - [aes.h](#), 542
- [aescript2.c](#)
 - [_CRT_SECURE_NO_DEPRECATED](#), 791
 - [main](#), 792
 - [MODE_DECRYPT](#), 791
 - [MODE_ENCRYPT](#), 791
 - [scanf_argv](#), 792
 - [USAGE](#), 791
- [Agency](#)
 - [LibMC::MCAgency](#), 130
- [agency_p](#)
 - [libmc.h](#), 327
 - [LibMC::MCAgency](#), 129
- [agency_s](#), 31
 - [client](#), 31
 - [default_agentstatus](#), 32
 - [enable_security](#), 32
 - [filename](#), 31
 - [hostName](#), 31
 - [last_error](#), 33
 - [mc_platform](#), 32
 - [portno](#), 31
 - [portnoc](#), 32
 - [server](#), 31
 - [stack_size](#), 32
 - [threads](#), 32
- [agency_t](#)
 - [libmc.h](#), 327
- [Agent](#)
 - [LibMC::MCAgent](#), 146
- [agent.c](#)
 - [agent_AddPersistentVariable](#), 202
 - [agent_Copy](#), 202
 - [agent_Destroy](#), 202
 - [agent_Initialize](#), 203
 - [agent_New](#), 203
 - [agent_RunChScript](#), 203
 - [agent_RunChScriptThread](#), 203
- [agent.h](#)
 - [agent_AddPersistentVariable](#), 249
 - [agent_Copy](#), 249
 - [agent_Destroy](#), 249

- agent_Initialize, 250
- agent_New, 250
- agent_RunChScript, 250
- agent_RunChScriptThread, 250
- AGENT_UPDATE
 - message.h, 389
- agent_AddPersistentVariable
 - agent.c, 202
 - agent.h, 249
- agent_code
 - agent_datastate_s, 34
- agent_code_ids
 - agent_datastate_s, 34
- agent_codes
 - agent_datastate_s, 34
- agent_Copy
 - agent.c, 202
 - agent.h, 249
- agent_datastate.c
 - agent_datastate_Copy, 205
 - agent_datastate_Destroy, 205
 - agent_datastate_New, 205
- agent_datastate.h
 - agent_datastate_Copy, 252
 - agent_datastate_Destroy, 252
 - agent_datastate_New, 253
 - agent_datastate_p, 252
 - agent_datastate_t, 252
- agent_datastate_Copy
 - agent_datastate.c, 205
 - agent_datastate.h, 252
- agent_datastate_Destroy
 - agent_datastate.c, 205
 - agent_datastate.h, 252
- agent_datastate_New
 - agent_datastate.c, 205
 - agent_datastate.h, 253
- agent_datastate_p
 - agent_datastate.h, 252
- agent_datastate_s, 34
 - agent_code, 34
 - agent_code_ids, 34
 - agent_codes, 34
 - init_agent_status, 35
 - number_of_tasks, 35
 - persistent, 35
 - return_data, 35
 - task_progress, 35
 - tasks, 34
 - xml_agent_root, 35
 - xml_root, 35
- agent_datastate_t
 - agent_datastate.h, 252
- agent_Destroy
 - agent.c, 202
 - agent.h, 249
- agent_Initialize
 - agent.c, 203
 - agent.h, 250
- agent_interp
 - agent_s, 40
- agent_lib.h
 - MC_AclAddReceiver_chdl, 255
 - MC_AclAddReplyTo_chdl, 255
 - MC_AclDestroy_chdl, 255
 - MC_AclNew_chdl, 255
 - MC_AclPost_chdl, 255
 - MC_AclReply_chdl, 255
 - MC_AclRetrieve_chdl, 256
 - MC_AclSend_chdl, 256
 - MC_AclSetContent_chdl, 256
 - MC_AclSetPerformative_chdl, 256
 - MC_AclSetSender_chdl, 256
 - MC_AclWaitRetrieve_chdl, 256
 - MC_AddAgent_chdl, 256
 - MC_AgentVariableRetrieve_chdl, 257
 - MC_AgentVariableSave_chdl, 257
 - MC_Barrier_chdl, 257
 - MC_BarrierDelete_chdl, 257
 - MC_BarrierInit_chdl, 257
 - MC_CallAgentFunc_chdl, 257
 - MC_ComposeAgent_chdl, 257
 - MC_CondBroadcast_chdl, 258
 - MC_CondReset_chdl, 258
 - MC_CondSignal_chdl, 258
 - MC_CondWait_chdl, 258
 - MC_DeleteAgent_chdl, 258
 - MC_DeregisterService_chdl, 258
 - MC_DestroyServiceSearchResult_chdl, 258
 - MC_End_chdl, 259
 - MC_FindAgentByID_chdl, 259
 - MC_FindAgentByName_chdl, 259
 - MC_GetAgentID_chdl, 259
 - MC_GetAgentName_chdl, 259
 - MC_GetAgentNumTasks_chdl, 259
 - MC_GetAgentStatus_chdl, 259
 - MC_GetAgentXMLString_chdl, 259
 - MC_GetTimeOfDay_chdl, 260
 - MC_HaltAgency_chdl, 260
 - MC_MutexLock_chdl, 260
 - MC_MutexUnlock_chdl, 260
 - MC_PrintAgentCode_chdl, 260
 - MC_RegisterService_chdl, 260
 - MC_ResumeAgency_chdl, 260
 - MC_RetrieveAgent_chdl, 260
 - MC_RetrieveAgentCode_chdl, 261
 - MC_SearchForService_chdl, 261
 - MC_SemaphorePost_chdl, 261

- MC_SemaphoreWait_chdl, 261
- MC_SendAgentMigrationMessage_chdl, 261
- MC_SendAgentMigrationMessageFile_chdl, 261
- MC_SendSteerCommand_chdl, 261
- MC_SetAgentStatus_chdl, 261
- MC_SetDefaultAgentStatus_chdl, 261
- MC_SyncDelete_chdl, 262
- MC_SyncInit_chdl, 262
- MC_TerminateAgent_chdl, 262
- agent_mailbox.c
 - agent_mailbox_Copy, 206
 - agent_mailbox_Destroy, 206
 - agent_mailbox_New, 206
 - agent_mailbox_Post, 206
 - agent_mailbox_Retrieve, 206
 - agent_mailbox_WaitRetrieve, 206
- agent_mailbox.h
 - agent_mailbox_Copy, 263
 - agent_mailbox_Destroy, 263
 - agent_mailbox_New, 263
 - agent_mailbox_p, 263
 - agent_mailbox_Post, 264
 - agent_mailbox_Retrieve, 264
 - agent_mailbox_t, 263
 - agent_mailbox_WaitRetrieve, 264
- agent_mailbox_Copy
 - agent_mailbox.c, 206
 - agent_mailbox.h, 263
- agent_mailbox_Destroy
 - agent_mailbox.c, 206
 - agent_mailbox.h, 263
- agent_mailbox_New
 - agent_mailbox.c, 206
 - agent_mailbox.h, 263
- agent_mailbox_p
 - agent_mailbox.h, 263
 - data_structures.h, 285
- agent_mailbox_Post
 - agent_mailbox.c, 206
 - agent_mailbox.h, 264
- agent_mailbox_Retrieve
 - agent_mailbox.c, 206
 - agent_mailbox.h, 264
- agent_mailbox_s, 37
 - mail_queue, 37
- agent_mailbox_t
 - agent_mailbox.h, 263
- agent_mailbox_WaitRetrieve
 - agent_mailbox.c, 206
 - agent_mailbox.h, 264
- agent_New
 - agent.c, 203
 - agent.h, 250
- agent_p
 - libmc.h, 327
 - LibMC::MCAgent, 145
- agent_persistent
 - agent_s, 41
- agent_pipe_active
 - agent_s, 41
- agent_pipe_ready_to_read
 - agent_s, 41
- agent_queue
 - mc_platform_s, 92
- agent_ready_to_send
 - agent_s, 41
- agent_return_data
 - agent_task_s, 44
- agent_return_data.c
 - interpreter_variable_data_Copy, 208
 - interpreter_variable_data_Destroy, 208
 - interpreter_variable_data_Initialize, 208
 - interpreter_variable_data_InitializeFromAgent, 208
 - interpreter_variable_data_New, 209
- agent_return_xml_parse
 - xml_parser.c, 834
- agent_RunChScript
 - agent.c, 203
 - agent.h, 250
- agent_RunChScriptThread
 - agent.c, 203
 - agent.h, 250
- agent_s, 38
 - agent_interp, 40
 - agent_persistent, 41
 - agent_pipe_active, 41
 - agent_pipe_ready_to_read, 41
 - agent_ready_to_send, 41
 - agent_script_ready, 41
 - agent_status, 40
 - agent_thread, 40
 - agent_thread_id, 40
 - agent_type, 40
 - arrival_time, 39
 - connect_id, 39
 - datastate, 39
 - home, 39
 - home_port, 39
 - id, 38
 - lock, 41
 - mailbox, 41
 - mc_platform, 41
 - name, 38
 - orphan, 39
 - owner, 39
 - return_data, 40

- run_lock, 40
- agent_script_ready
 - agent_s, 41
- agent_status
 - agent_s, 40
- agent_t
 - libmc.h, 327
- agent_task.c
 - agent_task_Copy, 210
 - agent_task_Destroy, 210
 - agent_task_New, 210
- agent_task.h
 - agent_task_Copy, 265
 - agent_task_Destroy, 265
 - agent_task_New, 265
 - agent_task_p, 265
 - agent_task_t, 265
- agent_task_Copy
 - agent_task.c, 210
 - agent_task.h, 265
- agent_task_Destroy
 - agent_task.c, 210
 - agent_task.h, 265
- agent_task_New
 - agent_task.c, 210
 - agent_task.h, 265
- agent_task_p
 - agent_task.h, 265
- agent_task_s, 43
 - agent_return_data, 44
 - agent_variable_list, 44
 - code_id, 44
 - init_agent_status, 43
 - num_saved_variables, 44
 - number_of_elements, 43
 - persistent, 43
 - saved_variables, 44
 - server_name, 44
 - size_of_element_array, 43
 - task_completed, 43
 - var_name, 44
- agent_task_t
 - agent_task.h, 265
- agent_thread
 - agent_s, 40
- agent_thread_id
 - agent_s, 40
- agent_type
 - agent_s, 40
- agent_variable_list
 - agent_task_s, 44
- agent_xml_compose
 - xml_compose.c, 827
 - xml_compose.h, 397
- agent_xml_compose__agent_code
 - xml_compose.c, 827
 - xml_compose.h, 397
- agent_xml_compose__agent_data
 - xml_compose.c, 827
 - xml_compose.h, 397
- agent_xml_compose__create_row_nodes
 - xml_compose.c, 828
 - xml_compose.h, 397
- agent_xml_compose__data
 - xml_compose.c, 828
 - xml_compose.h, 398
- agent_xml_compose__gaf_message
 - xml_compose.c, 828
 - xml_compose.h, 398
- agent_xml_compose__home
 - xml_compose.c, 828
 - xml_compose.h, 398
- agent_xml_compose__message
 - xml_compose.c, 828
 - xml_compose.h, 398
- agent_xml_compose__mobile_agent
 - xml_compose.c, 829
 - xml_compose.h, 398
- agent_xml_compose__name
 - xml_compose.c, 829
 - xml_compose.h, 399
- agent_xml_compose__owner
 - xml_compose.c, 829
 - xml_compose.h, 399
- agent_xml_compose__row
 - xml_compose.c, 829
 - xml_compose.h, 399
- agent_xml_compose__task
 - xml_compose.c, 829
 - xml_compose.h, 399
- agent_xml_compose__tasks
 - xml_compose.c, 829
 - xml_compose.h, 399
- agent_xml_flag
 - message_s, 156
- agent_xml_parse
 - xml_parser.c, 834
 - xml_parser.h, 402
- agent_xml_parse__agent_code
 - xml_parser.c, 834
 - xml_parser.h, 402
- agent_xml_parse__agent_data
 - xml_parser.c, 834
 - xml_parser.h, 402
- agent_xml_parse__data
 - xml_parser.c, 834
 - xml_parser.h, 403
- agent_xml_parse__fill_row_data

- xml_parser.c, 834
- xml_parser.h, 403
- agent_xml_parse__home
 - xml_parser.c, 835
 - xml_parser.h, 403
- agent_xml_parse__mobile_agent
 - xml_parser.c, 835
 - xml_parser.h, 403
- agent_xml_parse__name
 - xml_parser.c, 835
 - xml_parser.h, 403
- agent_xml_parse__owner
 - xml_parser.c, 835
 - xml_parser.h, 404
- agent_xml_parse__row
 - xml_parser.c, 835
 - xml_parser.h, 404
- agent_xml_parse__task
 - xml_parser.c, 835
 - xml_parser.h, 404
- agent_xml_parse__tasks
 - xml_parser.c, 836
 - xml_parser.h, 404
- AgentID
 - LibMC::MCAgent, 146
- AgentName
 - LibMC::MCAgent, 146
- AgentNumTasks
 - LibMC::MCAgent, 147
- AgentStatus
 - LibMC::MCAgent, 147
- AgentType
 - LibMC::MCAgent, 147
- alarmed
 - timing.c, 780
 - timing.h, 644
- alloc_nodes
 - mxml_index_s, 166
- ams
 - mc_platform_s, 92
- ams.c
 - ams_Destroy, 211
 - ams_Initialize, 211
 - ams_ManageAgentList, 211
 - ams_Print, 211
 - ams_Start, 211
 - ams_Thread, 212
- ams.h
 - ams_Destroy, 267
 - ams_ManageAgentList, 267
 - ams_Print, 267
 - ams_RemoveFinishedAgents, 267
 - ams_Start, 267
 - ams_Thread, 267
- STRUCT, 268
- ams_Destroy
 - ams.c, 211
 - ams.h, 267
- ams_Initialize
 - ams.c, 211
- ams_ManageAgentList
 - ams.c, 211
 - ams.h, 267
- ams_Print
 - ams.c, 211
 - ams.h, 267
- ams_RemoveFinishedAgents
 - ams.h, 267
- ams_Start
 - ams.c, 211
 - ams.h, 267
- ams_Thread
 - ams.c, 212
 - ams.h, 267
- AP_QUEUE_DECL_TEMPLATE
 - ap_queue_template.h, 269
 - data_structures.h, 284
- AP_QUEUE_GENERIC_DECL_TEMPLATE
 - ap_queue_template.h, 269
 - data_structures.h, 284
- AP_QUEUE_REMOVE_TEMPLATE
 - ap_queue_template.h, 269
- AP_QUEUE_SEARCH_TEMPLATE
 - ap_queue_template.h, 270
 - data_structures.c, 219
- AP_QUEUE_STD_DEFN_TEMPLATE
 - ap_queue_template.h, 270
 - data_structures.c, 219
- ap_queue_template.h
 - AP_QUEUE_DECL_TEMPLATE, 269
 - AP_QUEUE_GENERIC_DECL_TEMPLATE, 269
 - AP_QUEUE_REMOVE_TEMPLATE, 269
 - AP_QUEUE_SEARCH_TEMPLATE, 270
 - AP_QUEUE_STD_DEFN_TEMPLATE, 270
- arc4.c
 - _CRT_SECURE_NO_DEPRECATED, 665
 - _arc4_src, 666
 - arc4_crypt, 665
 - arc4_self_test, 665
 - arc4_setup, 666
- arc4.h
 - arc4_crypt, 543
 - arc4_self_test, 543
 - arc4_setup, 543
- arc4_context, 46
 - m, 46
 - x, 46

- y, [46](#)
- arc4_crypt
 - arc4.c, [665](#)
 - arc4.h, [543](#)
- arc4_self_test
 - arc4.c, [665](#)
 - arc4.h, [543](#)
- arc4_setup
 - arc4.c, [666](#)
 - arc4.h, [543](#)
- args
 - command_s, [50](#)
- array_dim
 - interpreter_variable_data_s, [87](#)
- array_extent
 - interpreter_variable_data_s, [87](#)
- arrival_time
 - agent_s, [39](#)
- asm_message_queue
 - mc_platform_s, [92](#)
- ASN1_BIT_STRING
 - x509.h, [647](#)
- ASN1_BMP_STRING
 - x509.h, [647](#)
- ASN1_BOOLEAN
 - x509.h, [647](#)
- ASN1_CONSTRUCTED
 - x509.h, [647](#)
- ASN1_CONTEXT_SPECIFIC
 - x509.h, [647](#)
- asn1_get_bool
 - x509read.c, [783](#)
- asn1_get_int
 - x509read.c, [783](#)
- asn1_get_len
 - x509read.c, [784](#)
- asn1_get_mpi
 - x509read.c, [784](#)
- asn1_get_tag
 - x509read.c, [784](#)
- ASN1_HASH_MDX
 - rsa.h, [599](#)
- ASN1_HASH_SHA1
 - rsa.h, [599](#)
- ASN1_IA5_STRING
 - x509.h, [648](#)
- ASN1_INTEGER
 - x509.h, [648](#)
- ASN1_NULL
 - x509.h, [648](#)
- ASN1_OCTET_STRING
 - x509.h, [648](#)
- ASN1_OID
 - x509.h, [648](#)
- ASN1_PRIMITIVE
 - x509.h, [648](#)
- ASN1_PRINTABLE_STRING
 - x509.h, [648](#)
- ASN1_SEQUENCE
 - x509.h, [648](#)
- ASN1_SET
 - x509.h, [649](#)
- ASN1_T61_STRING
 - x509.h, [649](#)
- ASN1_UNIVERSAL_STRING
 - x509.h, [649](#)
- ASN1_UTC_TIME
 - x509.h, [649](#)
- ASN1_UTF8_STRING
 - x509.h, [649](#)
- attr
 - mxml_index_s, [166](#)
- attrs
 - mxml_value_s, [171](#)
- authmode
 - ssl_context, [190](#)
- BACKLOG
 - acc.c, [200](#)
- BAD_REQUEST
 - mtp_http.h, [393](#)
- BADCERT_CN_MISMATCH
 - x509.h, [649](#)
- BADCERT_HAS_EXPIRED
 - x509.h, [649](#)
- BADCERT_NOT_TRUSTED
 - x509.h, [649](#)
- bar
 - class.cxx, [524](#)
 - foo_c, [79](#)
 - foo_s, [82](#)
- barfoo
 - foo_c, [80](#)
- barrier.c
 - barrier_node_Destroy, [213](#)
 - barrier_node_Initialize, [213](#)
 - barrier_queue_Add, [213](#)
 - barrier_queue_Delete, [213](#)
 - barrier_queue_Destroy, [214](#)
 - barrier_queue_Get, [214](#)
 - barrier_queue_New, [214](#)
 - barrier_queue_Pop, [214](#)
- barrier.h
 - barrier_node_Destroy, [273](#)
 - barrier_node_Initialize, [273](#)
 - barrier_node_p, [272](#)
 - barrier_node_t, [272](#)
 - barrier_queue_Add, [273](#)

- barrier_queue_Delete, 273
- barrier_queue_Destroy, 273
- barrier_queue_Get, 273
- barrier_queue_New, 274
- barrier_queue_p, 272
- barrier_queue_Pop, 274
- barrier_queue_t, 272
- barrier_node_Destroy
 - barrier.c, 213
 - barrier.h, 273
- barrier_node_Initialize
 - barrier.c, 213
 - barrier.h, 273
- barrier_node_p
 - barrier.h, 272
- barrier_node_s, 47
 - cond, 47
 - id, 47
 - lock, 47
 - num_registered, 47
 - num_waiting, 47
- barrier_node_t
 - barrier.h, 272
- barrier_queue
 - mc_platform_s, 93
- barrier_queue_Add
 - barrier.c, 213
 - barrier.h, 273
- barrier_queue_Delete
 - barrier.c, 213
 - barrier.h, 273
- barrier_queue_Destroy
 - barrier.c, 214
 - barrier.h, 273
- barrier_queue_Get
 - barrier.c, 214
 - barrier.h, 273
- barrier_queue_New
 - barrier.c, 214
 - barrier.h, 274
- barrier_queue_p
 - barrier.h, 272
- barrier_queue_Pop
 - barrier.c, 214
 - barrier.h, 274
- barrier_queue_s, 48
 - list, 48
 - lock, 48
 - size, 48
- barrier_queue_t
 - barrier.h, 272
- BarrierDelete
 - LibMC::MCAgency, 119
- BarrierInit
 - LibMC::MCAgency, 120
- base64.c
 - _CRT_SECURE_NO_DEPRECATED, 667
 - _base64_src, 668
 - base64_dec_map, 668
 - base64_decode, 667
 - base64_enc_map, 669
 - base64_encode, 668
 - base64_self_test, 668
- base64.h
 - base64_decode, 545
 - base64_encode, 546
 - base64_self_test, 546
 - ERR_BASE64_BUFFER_TOO_SMALL, 545
 - ERR_BASE64_INVALID_CHARACTER, 545
- base64_dec_map
 - base64.c, 668
- base64_decode
 - base64.c, 667
 - base64.h, 545
- base64_enc_map
 - base64.c, 669
- base64_encode
 - base64.c, 668
 - base64.h, 546
- base64_self_test
 - base64.c, 668
 - base64.h, 546
- benchmark.c
 - _CRT_SECURE_NO_DEPRECATED, 816
 - BUFSIZE, 816
 - main, 816
 - myrand, 817
- bignum.c
 - _CRT_SECURE_NO_DEPRECATED, 672
 - _bignum_src, 684
 - BITS_TO_LIMBS, 672
 - CHARS_TO_LIMBS, 673
 - mpi_add_abs, 673
 - mpi_add_int, 673
 - mpi_add_mpi, 673
 - mpi_cmp_abs, 673
 - mpi_cmp_int, 674
 - mpi_cmp_mpi, 674
 - mpi_copy, 674
 - mpi_div_int, 674
 - mpi_div_mpi, 675
 - mpi_exp_mod, 675
 - mpi_free, 675
 - mpi_gcd, 676
 - mpi_gen_prime, 676
 - mpi_get_digit, 676
 - mpi_grow, 677

- mpi_init, 677
- mpi_inv_mod, 677
- mpi_is_prime, 677
- mpi_lsb, 678
- mpi_lset, 678
- mpi_mod_int, 678
- mpi_mod_mpi, 678
- mpi_montg_init, 678
- mpi_montmul, 679
- mpi_montred, 679
- mpi_msb, 679
- mpi_mul_hlp, 679
- mpi_mul_int, 679
- mpi_mul_mpi, 679
- mpi_read_binary, 680
- mpi_read_file, 680
- mpi_read_string, 680
- mpi_self_test, 681
- mpi_shift_l, 681
- mpi_shift_r, 681
- mpi_sub_abs, 682
- mpi_sub_hlp, 682
- mpi_sub_int, 682
- mpi_sub_mpi, 682
- mpi_swap, 682
- mpi_write_binary, 683
- mpi_write_file, 683
- mpi_write_hlp, 683
- mpi_write_string, 684
- small_prime, 684
- bignum.h
 - biH, 550
 - biL, 550
 - CHK, 550
 - ciL, 550
 - ERR_MPI_BUFFER_TOO_SMALL, 550
 - ERR_MPI_DIVISION_BY_ZERO, 550
 - ERR_MPI_FILE_IO_ERROR, 550
 - ERR_MPI_INVALID_CHARACTER, 550
 - ERR_MPI_INVALID_PARAMETER, 551
 - ERR_MPI_NEGATIVE_VALUE, 551
 - ERR_MPI_NOT_ACCEPTABLE, 551
 - mpi_add_abs, 551
 - mpi_add_int, 551
 - mpi_add_mpi, 552
 - mpi_cmp_abs, 552
 - mpi_cmp_int, 552
 - mpi_cmp_mpi, 552
 - mpi_copy, 553
 - mpi_div_int, 553
 - mpi_div_mpi, 553
 - mpi_exp_mod, 554
 - mpi_free, 554
 - mpi_gcd, 554
 - mpi_gen_prime, 555
 - mpi_grow, 555
 - mpi_init, 555
 - mpi_inv_mod, 555
 - mpi_is_prime, 556
 - mpi_lsb, 556
 - mpi_lset, 556
 - mpi_mod_int, 556
 - mpi_mod_mpi, 557
 - mpi_msb, 557
 - mpi_mul_int, 557
 - mpi_mul_mpi, 557
 - mpi_read_binary, 558
 - mpi_read_file, 558
 - mpi_read_string, 558
 - mpi_self_test, 559
 - mpi_shift_l, 559
 - mpi_shift_r, 559
 - mpi_sub_abs, 560
 - mpi_sub_int, 560
 - mpi_sub_mpi, 560
 - mpi_swap, 560
 - mpi_write_binary, 560
 - mpi_write_file, 561
 - mpi_write_string, 561
 - t_dbl, 551
 - t_int, 551
- biH
 - bignum.h, 550
- biL
 - bignum.h, 550
- BITS_TO_LIMBS
 - bignum.c, 672
- bn_asm.h
 - MULADDC_CORE, 563
 - MULADDC_INIT, 563
 - MULADDC_STOP, 563
- boundary
 - mtp_http_s, 162
- buffer
 - md2_context, 148
 - md4_context, 150
 - md5_context, 152
 - mxml_fdbuf_s, 165
 - sha1_context, 179
 - sha2_context, 181
 - sha4_context, 183
- BUFSIZE
 - benchmark.c, 816
- ca_chain
 - ssl_context, 190
- ca_istrue
 - _x509_cert, 25

- cache_control
 - mtp_http_s, [162](#)
- CallAgentFunc
 - LibMC::MCAgency, [144](#)
- callbacks
 - mxml-entity.c, [489](#)
- CANCEL
 - message.h, [389](#)
- certs.c
 - _certs_src, [685](#)
 - test_ca_cert, [685](#)
 - test_ca_key, [685](#)
 - test_ca_pwd, [685](#)
 - test_cli_cert, [686](#)
 - test_cli_key, [686](#)
 - test_srv_cert, [687](#)
 - test_srv_key, [687](#)
 - xyssl_ca_cert, [688](#)
- certs.h
 - test_ca_cert, [565](#)
 - test_ca_key, [565](#)
 - test_ca_pwd, [565](#)
 - test_cli_cert, [565](#)
 - test_cli_key, [565](#)
 - test_srv_cert, [565](#)
 - test_srv_key, [566](#)
 - xyssl_ca_cert, [566](#)
- CH_REGULARCH
 - LibMC::MCAgency, [109](#)
- CH_SAFECH
 - LibMC::MCAgency, [109](#)
- CH_DATATYPE_SIZE
 - macros.h, [372](#)
- CH_DATATYPE_STR_TO_VAL
 - macros.h, [372](#)
- CH_DATATYPE_STRING
 - macros.h, [373](#)
- CH_DATATYPE_VALUE_STRING
 - macros.h, [373](#)
- CH_STRING_DATATYPE
 - macros.h, [374](#)
- CHARS_TO_LIMBS
 - bignum.c, [673](#)
- CHECK_NULL
 - macros.h, [374](#)
- chhome
 - LibMC::MCAgency::ChOptions_t, [132](#)
- child
 - mxml_node_s, [168](#)
- ChInitializeOptions
 - LibMC::MCAgency, [111](#)
- CHK
 - bignum.h, [550](#)
- ChShellType
 - LibMC::MCAgency, [109](#)
- ciL
 - bignum.h, [550](#)
- cipher
 - ssl_context, [193](#)
- cipherlist
 - ssl_context, [193](#)
- cksum
 - md2_context, [148](#)
- class.cxx
 - bar, [524](#)
 - f, [524](#)
 - foo, [524](#)
- client
 - agency_s, [31](#)
 - LibMC::MCAgency::MCAgency_t, [133](#)
- client_auth
 - ssl_context, [190](#)
- client_fd
 - acc.h, [248](#)
- client_thread_arg
 - mc_platform_s, [93](#)
- clientfd
 - connection_s, [51](#)
- cmd_handler_t
 - commands.h, [278](#)
- cmd_handlers
 - commands.h, [278](#)
- cmd_prompt
 - mc_platform_s, [93](#)
- cmd_prompt.c
 - cmd_prompt_Destroy, [215](#)
 - cmd_prompt_Initialize, [215](#)
 - cmd_prompt_Start, [215](#)
 - cmd_prompt_Thread, [216](#)
 - dealloc_command, [216](#)
 - exec_command, [216](#)
 - handler_HELP, [216](#)
 - handler_PRINT_CONNECTLIST, [216](#)
 - handler_PRINTLIST_AGENTS, [216](#)
 - handler_PRINTLIST_MESSAGE, [216](#)
 - handler_QUIT, [216](#)
 - handler_SEND, [216](#)
 - process_command, [217](#)
 - split_string, [217](#)
- cmd_prompt.h
 - cmd_prompt_Destroy, [276](#)
 - cmd_prompt_Initialize, [276](#)
 - cmd_prompt_p, [275](#)
 - cmd_prompt_Start, [276](#)
 - cmd_prompt_t, [275](#)
 - cmd_prompt_Thread, [276](#)
 - CMDLINE_SIZE, [275](#)
 - command_p, [276](#)

- command_t, 276
- dealloc_command, 276
- exec_command, 276
- process_command, 276
- split_string, 276
- cmd_prompt_Destroy
 - cmd_prompt.c, 215
 - cmd_prompt.h, 276
- cmd_prompt_Initialize
 - cmd_prompt.c, 215
 - cmd_prompt.h, 276
- cmd_prompt_p
 - cmd_prompt.h, 275
- cmd_prompt_s, 49
 - thread, 49
- cmd_prompt_Start
 - cmd_prompt.c, 215
 - cmd_prompt.h, 276
- cmd_prompt_t
 - cmd_prompt.h, 275
- cmd_prompt_Thread
 - cmd_prompt.c, 216
 - cmd_prompt.h, 276
- CMDLINE_SIZE
 - cmd_prompt.h, 275
- code_id
 - agent_task_s, 44
- COLLECT_SIZE
 - havege.h, 577
- COMMAND
 - commands.h, 277, 278
 - commands.x.h, 279
- command_cmds
 - commands.h, 278
- command_descriptions
 - commands.h, 278
- command_indices_e
 - commands.h, 278
- command_p
 - cmd_prompt.h, 276
- command_s, 50
 - args, 50
 - index, 50
 - num_args, 50
- command_t
 - cmd_prompt.h, 276
- commands.h
 - cmd_handler_t, 278
 - cmd_handlers, 278
 - COMMAND, 277, 278
 - command_cmds, 278
 - command_descriptions, 278
 - command_indices_e, 278
- commands.x.h
 - COMMAND, 279
 - help, 279
 - is, 279
 - print_connectlist, 279
 - printlist_agents, 280
 - send, 280
- comments
 - fipa_acl_Param_s, 64
- COMPOSE_BLOCKSIZE
 - dynstring.h, 293
- cond
 - barrier_node_s, 47
 - df.h, 290
 - mc_rwlock_s, 96
 - syncListNode_s, 197
- COND_BROADCAST
 - macros.h, 375
- COND_DESTROY
 - macros.h, 375
- COND_INIT
 - macros.h, 375
- COND_RESET
 - macros.h, 375
- COND_SIGNAL
 - macros.h, 375
- COND_SLEEP
 - macros.h, 376
- COND_SLEEP_ACTION
 - macros.h, 376
- COND_T
 - macros.h, 376
- COND_WAIT
 - macros.h, 376
- CondBroadcast
 - LibMC::MCAgency, 116
- CondReset
 - LibMC::MCAgency, 116
- CondSignal
 - LibMC::MCAgency, 116
- CondWait
 - LibMC::MCAgency, 116
- config.h
 - HAVE_STRDUP, 483
 - HAVE_VSNPRINTF, 483
 - mxml_strdupf, 483
 - MXML_VERSION, 483
- CONFLICT
 - mtp_http.h, 393
- connect_id
 - agent_s, 39
 - connection_s, 51
 - message_s, 154
- connection
 - mtp_http_s, 161

- connection.c
 - connection_Copy, 218
 - connection_Destroy, 218
 - connection_New, 218
- connection.h
 - connection_Close, 281
 - connection_Copy, 281
 - connection_CreateMessage, 281
 - connection_Destroy, 282
 - connection_New, 282
 - connection_p, 281
 - connection_RecvFromConnection, 282
 - connection_SendtoConnection, 282
 - connection_SetHosts, 282
 - connection_t, 281
- connection_Close
 - connection.h, 281
- connection_Copy
 - connection.c, 218
 - connection.h, 281
- connection_CreateMessage
 - connection.h, 281
- connection_Destroy
 - connection.c, 218
 - connection.h, 282
- connection_New
 - connection.c, 218
 - connection.h, 282
- connection_p
 - connection.h, 281
- connection_queue
 - mc_platform_s, 92
- connection_RecvFromConnection
 - connection.h, 282
- connection_s, 51
 - addr, 51
 - clientfd, 51
 - connect_id, 51
 - remote_hostname, 51
 - serverfd, 51
- connection_SendtoConnection
 - connection.h, 282
- connection_SetHosts
 - connection.h, 282
- connection_t
 - connection.h, 281
- content
 - fipa_acl_message_s, 62
 - fipa_expression_s, 70
 - fipa_string_s, 74
 - fipa_word_s, 77
 - mtp_http_s, 162
- content_length
 - mtp_http_s, 161
- content_type
 - mtp_http_content_s, 159
 - mtp_http_s, 161
- CONTINUE
 - mtp_http.h, 393
- conversation_id
 - fipa_acl_message_s, 62
- CREATED
 - mtp_http.h, 393
- ctx_dec
 - ssl_context, 194
- ctx_enc
 - ssl_context, 194
- ctxlen
 - ssl_context, 193
- cur_node
 - mxml_index_s, 166
- current
 - mxml_fdbuf_s, 165
- custom
 - mxml_value_u, 173
- D
 - rsa_context, 175
- DATA
 - list.h, 463
- data
 - interpreter_variable_data_s, 88
 - mtp_http_content_s, 159
 - mxml_custom_s, 164
- data_structures.c
 - AP_QUEUE_SEARCH_TEMPLATE, 219
 - AP_QUEUE_STD_DEFN_TEMPLATE, 219
- data_structures.h
 - agent_mailbox_p, 285
 - AP_QUEUE_DECL_TEMPLATE, 284
 - AP_QUEUE_GENERIC_DECL_TEMPLATE, 284
 - int, 285
 - Remove, 285
 - RemoveID, 285
 - RemoveName, 285
 - Search, 285
 - SearchName, 285
- data_type
 - interpreter_variable_data_s, 87
- datastate
 - agent_s, 39
- date
 - fipa_acl_Param_s, 65
 - mtp_http_s, 161
- datetime
 - fipa_expression_s::content_u, 71
- day

- [_x509_time](#), 28
 - [fipa_DateTime_s](#), 68
- [dealloc_command](#)
 - [cmd_prompt.c](#), 216
 - [cmd_prompt.h](#), 276
- [Default](#)
 - [LibMC::Properties::Settings](#), 178
- [default_agent_status](#)
 - [LibMC::MCAgency::MCAgencyOptions_t](#), 135
 - [MCAgencyOptions_s](#), 136
- [default_agentstatus](#)
 - [agency_s](#), 32
 - [LibMC::MCAgency::MCAgency_t](#), 134
 - [mc_platform_s](#), 93
- [default_callback](#)
 - [mxml-entity.c](#), 488
- [DEFAULT_HOSTNAME_LENGTH](#)
 - [mc_platform.c](#), 466
- [defaultInstance](#)
 - [LibMC::Properties::Settings](#), 178
- [DeleteAgent](#)
 - [LibMC::MCAgent](#), 141
- [des.c](#)
 - [_CRT_SECURE_NO_DEPRECATED](#), 690
 - [_des_src](#), 696
 - [des3_cbc_decrypt](#), 692
 - [des3_cbc_encrypt](#), 692
 - [des3_crypt](#), 692
 - [des3_decrypt](#), 693
 - [des3_encrypt](#), 693
 - [des3_set_2keys](#), 693
 - [des3_set_3keys](#), 693
 - [des_cbc_decrypt](#), 694
 - [des_cbc_encrypt](#), 694
 - [des_crypt](#), 694
 - [des_decrypt](#), 695
 - [des_encrypt](#), 695
 - [DES_FP](#), 690
 - [DES_IP](#), 690
 - [des_main_ks](#), 695
 - [DES_ROUND](#), 691
 - [des_self_test](#), 695
 - [des_set_key](#), 695
 - [GET_UINT32_BE](#), 691
 - [LHs](#), 696
 - [PUT_UINT32_BE](#), 691
 - [RHs](#), 696
 - [SB1](#), 696
 - [SB2](#), 697
 - [SB3](#), 697
 - [SB4](#), 698
 - [SB5](#), 698
 - [SB6](#), 698
 - [SB7](#), 699
 - [SB8](#), 699
- [des.h](#)
 - [des3_cbc_decrypt](#), 568
 - [des3_cbc_encrypt](#), 568
 - [des3_decrypt](#), 568
 - [des3_encrypt](#), 569
 - [des3_set_2keys](#), 569
 - [des3_set_3keys](#), 569
 - [des_cbc_decrypt](#), 569
 - [des_cbc_encrypt](#), 570
 - [des_decrypt](#), 570
 - [des_encrypt](#), 570
 - [des_self_test](#), 571
 - [des_set_key](#), 571
- [des3_cbc_decrypt](#)
 - [des.c](#), 692
 - [des.h](#), 568
- [des3_cbc_encrypt](#)
 - [des.c](#), 692
 - [des.h](#), 568
- [des3_context](#), 53
 - [dsk](#), 53
 - [esk](#), 53
- [des3_crypt](#)
 - [des.c](#), 692
- [des3_decrypt](#)
 - [des.c](#), 693
 - [des.h](#), 568
- [des3_encrypt](#)
 - [des.c](#), 693
 - [des.h](#), 569
- [des3_set_2keys](#)
 - [des.c](#), 693
 - [des.h](#), 569
- [des3_set_3keys](#)
 - [des.c](#), 693
 - [des.h](#), 569
- [des_cbc_decrypt](#)
 - [des.c](#), 694
 - [des.h](#), 569
- [des_cbc_encrypt](#)
 - [des.c](#), 694
 - [des.h](#), 570
- [des_context](#), 54
 - [dsk](#), 54
 - [esk](#), 54
- [des_crypt](#)
 - [des.c](#), 694
- [des_decrypt](#)
 - [des.c](#), 695
 - [des.h](#), 570
- [des_encrypt](#)
 - [des.c](#), 695

- des.h, 570
- DES_FP
 - des.c, 690
- DES_IP
 - des.c, 690
- des_main_ks
 - des.c, 695
- DES_ROUND
 - des.c, 691
- des_self_test
 - des.c, 695
 - des.h, 571
- des_set_key
 - des.c, 695
 - des.h, 571
- Destroy
 - LibMC::MCAclMessage, 102
- destroy
 - mxml_custom_s, 164
- df
 - mc_platform_s, 92
- df.c
 - df_Add, 222
 - df_AddRequest, 222
 - df_Destroy, 222
 - df_Initialize, 222
 - df_node_Destroy, 222
 - df_ProcessRequest, 222
 - df_request_list_Destroy, 222
 - df_request_list_New, 223
 - df_request_list_node_Destroy, 223
 - df_request_list_node_New, 223
 - df_request_list_Pop, 223
 - df_request_search_Destroy, 223
 - df_request_search_New, 223
 - df_SearchForService, 223
 - df_Start, 224
 - df_Thread, 224
 - REQUEST, 221
 - request_handler_DEREGISTER, 224
 - request_handler_REGISTER, 224
 - request_handler_SEARCH, 224
 - request_handler_SUBSCRIBE, 224
- df.h
 - cond, 290
 - df_AddRequest, 288
 - df_Destroy, 288
 - df_Initialize, 288
 - df_node_Destroy, 288
 - df_ProcessRequest, 288
 - df_request_list_Destroy, 288
 - df_request_list_index_e, 287
 - df_request_list_New, 288
 - df_request_list_node_Destroy, 288
 - df_request_list_node_New, 289
 - df_request_list_Pop, 289
 - df_request_search_Destroy, 289
 - df_request_search_New, 289
 - df_SearchForService, 289
 - df_Start, 289
 - df_Thread, 289
 - HERE, 287
 - INSERT, 287
 - lock, 290
 - mc_platform, 290
 - MISC, 287
 - node, 290
 - num_entries, 290
 - REQUEST, 287
 - request_list, 290
 - search_results, 291
 - search_string, 291
 - service_list, 291
 - service_types_e, 287
 - size, 291
 - SOME, 287
 - STRUCT, 289, 290
 - thread, 291
 - TYPES, 287
 - waiting, 291
 - waiting_cond, 291
 - waiting_lock, 291
 - ZERO, 287
- df_Add
 - df.c, 222
- df_AddRequest
 - df.c, 222
 - df.h, 288
- df_Destroy
 - df.c, 222
 - df.h, 288
- df_Initialize
 - df.c, 222
 - df.h, 288
- df_node_Destroy
 - df.c, 222
 - df.h, 288
- df_ProcessRequest
 - df.c, 222
 - df.h, 288
- df_request.x.h
 - register, 292
 - REQUEST, 292
 - search, 292
- df_request_list_Destroy
 - df.c, 222
 - df.h, 288
- df_request_list_index_e

- df.h, 287
- df_request_list_New
 - df.c, 223
 - df.h, 288
- df_request_list_node_Destroy
 - df.c, 223
 - df.h, 288
- df_request_list_node_New
 - df.c, 223
 - df.h, 289
- df_request_list_Pop
 - df.c, 223
 - df.h, 289
- df_request_search_Destroy
 - df.c, 223
 - df.h, 289
- df_request_search_New
 - df.c, 223
 - df.h, 289
- df_SearchForService
 - df.c, 223
 - df.h, 289
- df_Start
 - df.c, 224
 - df.h, 289
- df_Thread
 - df.c, 224
 - df.h, 289
- dh_client.c
 - _CRT_SECURE_NO_DEPRECATED, 800
 - main, 800
 - SERVER_NAME, 800
 - SERVER_PORT, 800
- dh_genprime.c
 - _CRT_SECURE_NO_DEPRECATED, 802
 - DH_P_SIZE, 802
 - GENERATOR, 802
 - main, 802
- DH_P_SIZE
 - dh_genprime.c, 802
- dh_server.c
 - _CRT_SECURE_NO_DEPRECATED, 803
 - main, 803
 - PLAINTEXT, 803
 - SERVER_PORT, 803
- dhm.c
 - _CRT_SECURE_NO_DEPRECATED, 701
 - _dhm_src, 704
 - dhm_calc_secret, 702
 - dhm_free, 702
 - dhm_make_params, 702
 - dhm_make_public, 703
 - DHM_MPI_EXPORT, 701
 - dhm_read_bignum, 703
 - dhm_read_params, 703
 - dhm_read_public, 704
 - dhm_self_test, 704
- dhm.h
 - dhm_calc_secret, 573
 - dhm_free, 574
 - dhm_make_params, 574
 - dhm_make_public, 574
 - dhm_read_params, 575
 - dhm_read_public, 575
 - dhm_self_test, 576
 - ERR_DHM_BAD_INPUT_DATA, 573
 - ERR_DHM_CALC_SECRET_FAILED, 573
 - ERR_DHM_MAKE_PARAMS_FAILED, 573
 - ERR_DHM_MAKE_PUBLIC_FAILED, 573
 - ERR_DHM_READ_PARAMS_FAILED, 573
 - ERR_DHM_READ_PUBLIC_FAILED, 573
- dhm_calc_secret
 - dhm.c, 702
 - dhm.h, 573
- dhm_context, 55
 - G, 55
 - GX, 56
 - GY, 56
 - K, 56
 - len, 55
 - P, 55
 - RP, 56
 - X, 55
- dhm_ctx
 - ssl_context, 192
- dhm_free
 - dhm.c, 702
 - dhm.h, 574
- dhm_G
 - ssl_server.c, 814
- dhm_make_params
 - dhm.c, 702
 - dhm.h, 574
- dhm_make_public
 - dhm.c, 703
 - dhm.h, 574
- DHM_MPI_EXPORT
 - dhm.c, 701
- dhm_P
 - ssl_server.c, 814
- dhm_read_bignum
 - dhm.c, 703
- dhm_read_params
 - dhm.c, 703
 - dhm.h, 575
- dhm_read_public
 - dhm.c, 704
 - dhm.h, 575

- dhm_self_test
 - dhm.c, [704](#)
 - dhm.h, [576](#)
- DP
 - rsa_context, [175](#)
- DQ
 - rsa_context, [175](#)
- drk
 - aes_context, [30](#)
- dsa.c
 - _CRT_SECURE_NO_DEPRECATED, [706](#)
 - rsa_check_privkey, [706](#)
 - rsa_check_pubkey, [706](#)
 - rsa_free, [706](#)
 - rsa_gen_key, [706](#)
 - rsa_pkcs1_decrypt, [707](#)
 - rsa_pkcs1_encrypt, [707](#)
 - rsa_pkcs1_sign, [707](#)
 - rsa_pkcs1_verify, [708](#)
 - rsa_private, [708](#)
 - rsa_public, [709](#)
 - rsa_self_test, [709](#)
- dsk
 - des3_context, [53](#)
 - des_context, [54](#)
- dynstring.c
 - dynstring_Append, [225](#)
 - dynstring_Destroy, [225](#)
 - dynstring_New, [225](#)
- dynstring.h
 - COMPOSE_BLOCKSIZE, [293](#)
 - dynstring_Append, [293](#)
 - dynstring_Destroy, [293](#)
 - dynstring_New, [294](#)
 - dynstring_p, [293](#)
 - dynstring_t, [293](#)
- dynstring_Append
 - dynstring.c, [225](#)
 - dynstring.h, [293](#)
- dynstring_Destroy
 - dynstring.c, [225](#)
 - dynstring.h, [293](#)
- dynstring_New
 - dynstring.c, [225](#)
 - dynstring.h, [294](#)
- dynstring_p
 - dynstring.h, [293](#)
- dynstring_s, [57](#)
 - len, [57](#)
 - message, [57](#)
 - size, [57](#)
- dynstring_t
 - dynstring.h, [293](#)
- E
 - rsa_context, [175](#)
- element
 - mxml_value_u, [172](#)
- enable_security
 - agency_s, [32](#)
 - LibMC::MCAgency::MCAgency_t, [134](#)
 - LibMC::MCAgency::MCAgencyOptions_t, [135](#)
 - MCAgencyOptions_s, [136](#)
- ENCODE_UTF16BE
 - mxml-file.c, [491](#)
- ENCODE_UTF16LE
 - mxml-file.c, [491](#)
- ENCODE_UTF8
 - mxml-file.c, [491](#)
- encoding
 - fipa_acl_message_s, [62](#)
- ENCRYPTED_DATA
 - message.h, [389](#)
- ENCRYPTION_INITIALIZE
 - message.h, [389](#)
- End
 - LibMC::MCAgency, [111](#)
- end
 - mxml_fdbuf_s, [165](#)
- Ended
 - LibMC::MCAgency, [109](#)
- endpoint
 - ssl_context, [190](#)
- enum.cxx
 - FOO_BLUE, [525](#)
 - FOO_ONE, [525](#)
 - FOO_RED, [525](#)
 - FOO_TWO, [525](#)
 - foo_enum_e, [525](#)
 - foo_enum_t, [525](#)
- erk
 - aes_context, [30](#)
- ERR
 - libmc.h, [328](#)
 - mc_error.h, [382](#)
- err
 - mc_platform_s, [91](#)
- ERR_ASN1_INVALID_DATA
 - x509.h, [649](#)
- ERR_ASN1_INVALID_LENGTH
 - x509.h, [650](#)
- ERR_ASN1_LENGTH_MISMATCH
 - x509.h, [650](#)
- ERR_ASN1_OUT_OF_DATA
 - x509.h, [650](#)
- ERR_ASN1_UNEXPECTED_TAG
 - x509.h, [650](#)

- ERR_BASE64_BUFFER_TOO_SMALL
base64.h, [545](#)
- ERR_BASE64_INVALID_CHARACTER
base64.h, [545](#)
- ERR_DHM_BAD_INPUT_DATA
dhm.h, [573](#)
- ERR_DHM_CALC_SECRET_FAILED
dhm.h, [573](#)
- ERR_DHM_MAKE_PARAMS_FAILED
dhm.h, [573](#)
- ERR_DHM_MAKE_PUBLIC_FAILED
dhm.h, [573](#)
- ERR_DHM_READ_PARAMS_FAILED
dhm.h, [573](#)
- ERR_DHM_READ_PUBLIC_FAILED
dhm.h, [573](#)
- ERR_MPI_BUFFER_TOO_SMALL
bignum.h, [550](#)
- ERR_MPI_DIVISION_BY_ZERO
bignum.h, [550](#)
- ERR_MPI_FILE_IO_ERROR
bignum.h, [550](#)
- ERR_MPI_INVALID_CHARACTER
bignum.h, [550](#)
- ERR_MPI_INVALID_PARAMETER
bignum.h, [551](#)
- ERR_MPI_NEGATIVE_VALUE
bignum.h, [551](#)
- ERR_MPI_NOT_ACCEPTABLE
bignum.h, [551](#)
- ERR_NET_ACCEPT_FAILED
net.h, [593](#)
- ERR_NET_BIND_FAILED
net.h, [593](#)
- ERR_NET_CONN_RESET
net.h, [593](#)
- ERR_NET_CONNECT_FAILED
net.h, [593](#)
- ERR_NET_LISTEN_FAILED
net.h, [593](#)
- ERR_NET_RECV_FAILED
net.h, [593](#)
- ERR_NET_SEND_FAILED
net.h, [593](#)
- ERR_NET_SOCKET_FAILED
net.h, [593](#)
- ERR_NET_UNKNOWN_HOST
net.h, [593](#)
- ERR_NET_WOULD_BLOCK
net.h, [594](#)
- ERR_RSA_BAD_INPUT_DATA
rsa.h, [599](#)
- ERR_RSA_INVALID_PADDING
rsa.h, [599](#)
- ERR_RSA_KEY_CHK_FAILED
rsa.h, [599](#)
- ERR_RSA_KEY_GEN_FAILED
rsa.h, [599](#)
- ERR_RSA_KEY_RD_FAILED
rsa.h, [599](#)
- ERR_RSA_KEY_WR_FAILED
rsa.h, [599](#)
- ERR_RSA_PRIVATE_FAILED
rsa.h, [600](#)
- ERR_RSA_PUBLIC_FAILED
rsa.h, [600](#)
- ERR_RSA_VERIFY_FAILED
rsa.h, [600](#)
- ERR_SSL_BAD_HS_CERTIFICATE
ssl.h, [627](#)
- ERR_SSL_BAD_HS_CERTIFICATE_REQUEST
ssl.h, [627](#)
- ERR_SSL_BAD_HS_CERTIFICATE_VERIFY
ssl.h, [627](#)
- ERR_SSL_BAD_HS_CHANGE_CIPHER_SPEC
ssl.h, [627](#)
- ERR_SSL_BAD_HS_CLIENT_HELLO
ssl.h, [627](#)
- ERR_SSL_BAD_HS_CLIENT_KEY_-
EXCHANGE
ssl.h, [628](#)
- ERR_SSL_BAD_HS_FINISHED
ssl.h, [628](#)
- ERR_SSL_BAD_HS_SERVER_HELLO
ssl.h, [628](#)
- ERR_SSL_BAD_HS_SERVER_HELLO_DONE
ssl.h, [628](#)
- ERR_SSL_BAD_HS_SERVER_KEY_-
EXCHANGE
ssl.h, [628](#)
- ERR_SSL_CA_CHAIN_REQUIRED
ssl.h, [628](#)
- ERR_SSL_CERTIFICATE_REQUIRED
ssl.h, [628](#)
- ERR_SSL_CERTIFICATE_TOO_LARGE
ssl.h, [628](#)
- ERR_SSL_FATAL_ALERT_MESSAGE
ssl.h, [628](#)
- ERR_SSL_FEATURE_UNAVAILABLE
ssl.h, [629](#)
- ERR_SSL_INVALID_MAC
ssl.h, [629](#)
- ERR_SSL_INVALID_MODULUS_SIZE
ssl.h, [629](#)
- ERR_SSL_INVALID_RECORD
ssl.h, [629](#)
- ERR_SSL_NO_CIPHER_CHOSEN
ssl.h, [629](#)

- ERR_SSL_NO_CLIENT_CERTIFICATE
 - ssl.h, [629](#)
- ERR_SSL_NO_SESSION_FOUND
 - ssl.h, [629](#)
- ERR_SSL_PEER_CLOSE_NOTIFY
 - ssl.h, [629](#)
- ERR_SSL_PEER_VERIFY_FAILED
 - ssl.h, [629](#)
- ERR_SSL_PRIVATE_KEY_REQUIRED
 - ssl.h, [630](#)
- ERR_SSL_UNEXPECTED_MESSAGE
 - ssl.h, [630](#)
- ERR_SSL_UNKNOWN_CIPHER
 - ssl.h, [630](#)
- ERR_X509_CERT_INVALID_ALG
 - x509.h, [650](#)
- ERR_X509_CERT_INVALID_DATE
 - x509.h, [650](#)
- ERR_X509_CERT_INVALID_EXTENSIONS
 - x509.h, [650](#)
- ERR_X509_CERT_INVALID_FORMAT
 - x509.h, [650](#)
- ERR_X509_CERT_INVALID_NAME
 - x509.h, [651](#)
- ERR_X509_CERT_INVALID_PEM
 - x509.h, [651](#)
- ERR_X509_CERT_INVALID_PUBKEY
 - x509.h, [651](#)
- ERR_X509_CERT_INVALID_SERIAL
 - x509.h, [651](#)
- ERR_X509_CERT_INVALID_SIGNATURE
 - x509.h, [651](#)
- ERR_X509_CERT_INVALID_VERSION
 - x509.h, [651](#)
- ERR_X509_CERT_SIG_MISMATCH
 - x509.h, [651](#)
- ERR_X509_CERT_UNKNOWN_PK_ALG
 - x509.h, [651](#)
- ERR_X509_CERT_UNKNOWN_SIG_ALG
 - x509.h, [651](#)
- ERR_X509_CERT_UNKNOWN_VERSION
 - x509.h, [652](#)
- ERR_X509_FEATURE_UNAVAILABLE
 - x509.h, [652](#)
- ERR_X509_KEY_INVALID_ENC_IV
 - x509.h, [652](#)
- ERR_X509_KEY_INVALID_FORMAT
 - x509.h, [652](#)
- ERR_X509_KEY_INVALID_PEM
 - x509.h, [652](#)
- ERR_X509_KEY_INVALID_VERSION
 - x509.h, [652](#)
- ERR_X509_KEY_PASSWORD_MISMATCH
 - x509.h, [652](#)
- ERR_X509_KEY_PASSWORD_REQUIRED
 - x509.h, [652](#)
- ERR_X509_KEY_UNKNOWN_ENC_ALG
 - x509.h, [652](#)
- ERR_X509_SIG_VERIFY_FAILED
 - x509.h, [653](#)
- error_code_e
 - libmc.h, [327](#)
 - mc_error.h, [382](#)
- error_code_t
 - libmc.h, [327](#)
 - mc_error.h, [382](#)
- esk
 - des3_context, [53](#)
 - des_context, [54](#)
- exec_command
 - cmd_prompt.c, [216](#)
 - cmd_prompt.h, [276](#)
- EXPECTATION_FAILED
 - mtp_http.h, [394](#)
- EXPONENT
 - rsa_genkey.c, [806](#)
- EXPORTMC
 - libmc.h, [326](#)
- expression
 - fipa_expression_s::content_u, [71](#)
- F
 - md4.c, [719](#)
 - md5.c, [725](#)
 - sha1.c, [742](#)
- f
 - class.cxx, [524](#)
- F0
 - sha2.c, [749](#)
 - sha4.c, [756](#)
- F1
 - sha2.c, [749](#)
 - sha4.c, [756](#)
- FAILED_DEPENDANCY
 - mtp_http.h, [394](#)
- fd
 - mxml_fdbuf_s, [165](#)
- filename
 - agency_s, [31](#)
 - LibMC::MCAgency::MCAgency_t, [133](#)
- FindAgentByID
 - LibMC::MCAgency, [122](#)
- FindAgentByName
 - LibMC::MCAgency, [122](#)
- FIPA_ACCEPT_PROPOSAL
 - fipa_acl.h, [299](#)
 - LibMC::MCAclMessage, [99](#)
- FIPA_ACL

- message.h, 389
- fipa_acl.h
 - FIPA_ACCEPT_PROPOSAL, 299
 - FIPA_AGREE, 299
 - FIPA_CALL_FOR_PROPOSAL, 299
 - FIPA_CANCEL, 299
 - FIPA_CONFIRM, 299
 - FIPA_DISCONFIRM, 299
 - FIPA_ERROR, 299
 - FIPA_EXPR_DATETIME, 299
 - FIPA_EXPR_EXPRESSION, 299
 - FIPA_EXPR_NUMBER, 299
 - FIPA_EXPR_STRING, 299
 - FIPA_EXPR_WORD, 299
 - FIPA_EXPR_ZERO, 299
 - FIPA_FAILURE, 299
 - FIPA_INFORM, 299
 - FIPA_INFORM_IF, 299
 - FIPA_INFORM_REF, 299
 - FIPA_NOT_UNDERSTOOD, 299
 - FIPA_PROPOGATE, 299
 - FIPA_PROPOSE, 299
 - FIPA_PROXY, 299
 - FIPA_QUERY_IF, 299
 - FIPA_QUERY_REF, 299
 - FIPA_REFUSE, 300
 - FIPA_REJECT_PROPOSAL, 300
 - FIPA_REQUEST, 300
 - FIPA_REQUEST_WHEN, 300
 - FIPA_REQUEST_WHENEVER, 300
 - FIPA_SUBSCRIBE, 300
 - FIPA_ZERO, 299
- FIPA_AGREE
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_CALL_FOR_PROPOSAL
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_CANCEL
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_CONFIRM
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_DISCONFIRM
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_ERROR
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_EXPR_DATETIME
 - fipa_acl.h, 299
- FIPA_EXPR_EXPRESSION
 - fipa_acl.h, 299
- FIPA_EXPR_NUMBER
 - fipa_acl.h, 299
- FIPA_EXPR_STRING
 - fipa_acl.h, 299
- FIPA_EXPR_WORD
 - fipa_acl.h, 299
- FIPA_EXPR_ZERO
 - fipa_acl.h, 299
- FIPA_FAILURE
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_INFORM
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_INFORM_IF
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_INFORM_REF
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_NOT_UNDERSTOOD
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_PROPOGATE
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_PROPOSE
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_PROXY
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_QUERY_IF
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_QUERY_REF
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- FIPA_REFUSE
 - fipa_acl.h, 300
 - LibMC::MCAclMessage, 99
- FIPA_REJECT_PROPOSAL
 - fipa_acl.h, 300
 - LibMC::MCAclMessage, 99
- FIPA_REQUEST
 - fipa_acl.h, 300
 - LibMC::MCAclMessage, 99
- FIPA_REQUEST_WHEN
 - fipa_acl.h, 300
 - LibMC::MCAclMessage, 99
- FIPA_REQUEST_WHENEVER
 - fipa_acl.h, 300
 - LibMC::MCAclMessage, 99
- FIPA_SUBSCRIBE
 - LibMC::MCAclMessage, 99

- fipa_acl.h, 300
 - LibMC::MCAclMessage, 99
- FIPA_ZERO
 - fipa_acl.h, 299
 - LibMC::MCAclMessage, 99
- fipa_acl.c
 - fipa_acl_Compose, 227
 - fipa_acl_message_Copy, 228
 - fipa_acl_message_Destroy, 228
 - fipa_acl_message_New, 228
 - fipa_acl_Parse, 228
 - fipa_agent_identifier_Compose, 228
 - fipa_agent_identifier_Copy, 228
 - fipa_agent_identifier_Destroy, 229
 - fipa_agent_identifier_New, 229
 - fipa_agent_identifier_Parse, 229
 - fipa_agent_identifier_set_Compose, 229
 - fipa_agent_identifier_set_Copy, 229
 - fipa_agent_identifier_set_Destroy, 230
 - fipa_agent_identifier_set_New, 230
 - fipa_agent_identifier_set_Parse, 230
 - fipa_CheckNextToken, 230
 - fipa_DateTime_Compose, 230
 - fipa_DateTime_Copy, 230
 - fipa_DateTime_Destroy, 231
 - fipa_DateTime_New, 231
 - fipa_datetime_Parse, 231
 - fipa_expression_Compose, 231
 - fipa_expression_Copy, 231
 - fipa_expression_Destroy, 231
 - fipa_expression_New, 232
 - fipa_expression_Parse, 232
 - fipa_GetAtom, 232
 - fipa_GetNextWord, 232
 - fipa_GetWholeToken, 232
 - fipa_message_parameter_Parse, 232
 - fipa_message_string_Copy, 233
 - fipa_message_string_Destroy, 233
 - fipa_message_string_New, 233
 - fipa_message_type_Parse, 233
 - fipa_number_Compose, 233
 - fipa_number_Copy, 234
 - fipa_number_Destroy, 234
 - fipa_number_New, 234
 - fipa_performative_Compose, 234
 - fipa_Reply, 234
 - fipa_string_Compose, 234
 - fipa_string_Copy, 235
 - fipa_string_Destroy, 235
 - fipa_string_New, 235
 - fipa_string_Parse, 235
 - fipa_url_Compose, 235
 - fipa_url_Copy, 235
 - fipa_url_Destroy, 235
 - fipa_url_New, 235
 - fipa_url_Parse, 236
 - fipa_url_sequence_Compose, 236
 - fipa_url_sequence_Copy, 236
 - fipa_url_sequence_Destroy, 236
 - fipa_url_sequence_New, 236
 - fipa_url_sequence_Parse, 236
 - fipa_word_Compose, 237
 - fipa_word_Copy, 237
 - fipa_word_Destroy, 237
 - fipa_word_New, 237
 - fipa_word_Parse, 237
 - FREEMEM, 227
- fipa_acl.h
 - fipa_acl_Compose, 300
 - fipa_acl_message_Copy, 300
 - fipa_acl_message_Destroy, 300
 - fipa_acl_message_New, 300
 - fipa_acl_message_p, 297
 - fipa_acl_message_t, 297
 - fipa_acl_Parse, 301
 - fipa_agent_identifier_Compose, 301
 - fipa_agent_identifier_Copy, 301
 - fipa_agent_identifier_Destroy, 301
 - fipa_agent_identifier_New, 301
 - fipa_agent_identifier_p, 297
 - fipa_agent_identifier_Parse, 301
 - fipa_agent_identifier_set_Compose, 301
 - fipa_agent_identifier_set_Copy, 302
 - fipa_agent_identifier_set_Destroy, 302
 - fipa_agent_identifier_set_New, 302
 - fipa_agent_identifier_set_p, 297
 - fipa_agent_identifier_set_Parse, 302
 - fipa_agent_identifier_set_t, 297
 - fipa_agent_identifier_t, 298
 - fipa_CheckNextToken, 302
 - fipa_DateTime_Compose, 302
 - fipa_DateTime_Copy, 303
 - fipa_DateTime_Destroy, 303
 - fipa_DateTime_New, 303
 - fipa_DateTime_p, 298
 - fipa_datetime_Parse, 303
 - fipa_DateTime_t, 298
 - fipa_expression_Compose, 303
 - fipa_expression_Copy, 303
 - fipa_expression_Destroy, 304
 - fipa_expression_New, 304
 - fipa_expression_p, 298
 - fipa_expression_Parse, 304
 - fipa_expression_t, 298
 - fipa_expression_type_e, 299
 - fipa_expression_type_t, 298
 - fipa_GetAtom, 304
 - fipa_GetNextWord, 304

- fipa_GetWholeToken, 305
- fipa_message_parameter_Parse, 305
- fipa_message_string_Copy, 305
- fipa_message_string_Destroy, 305
- fipa_message_string_New, 305
- fipa_message_string_p, 298
- fipa_message_string_t, 298
- fipa_message_type_Parse, 305
- fipa_number_Compose, 305
- fipa_number_Copy, 306
- fipa_number_Destroy, 306
- fipa_number_New, 306
- fipa_number_p, 298
- fipa_number_t, 298
- fipa_performative_Compose, 306
- fipa_performative_e, 299
- fipa_Reply, 306
- fipa_string_Compose, 306
- fipa_string_Copy, 307
- fipa_string_Destroy, 307
- fipa_string_New, 307
- fipa_string_p, 298
- fipa_string_Parse, 307
- fipa_string_t, 298
- fipa_url_Compose, 307
- fipa_url_Copy, 307
- fipa_url_Destroy, 307
- fipa_url_New, 308
- fipa_url_p, 298
- fipa_url_Parse, 308
- fipa_url_sequence_Compose, 308
- fipa_url_sequence_Copy, 308
- fipa_url_sequence_Destroy, 308
- fipa_url_sequence_New, 308
- fipa_url_sequence_p, 298
- fipa_url_sequence_Parse, 308
- fipa_url_sequence_t, 298
- fipa_url_t, 299
- fipa_word_Compose, 309
- fipa_word_Copy, 309
- fipa_word_Destroy, 309
- fipa_word_New, 309
- fipa_word_p, 299
- fipa_word_Parse, 309
- fipa_word_t, 299
- fipa_acl_Compose
 - fipa_acl.c, 227
 - fipa_acl.h, 300
- fipa_acl_envelope.h
 - fipa_acl_envelope_Copy, 311
 - fipa_acl_envelope_Destroy, 311
 - fipa_acl_envelope_New, 311
 - fipa_acl_envelope_p, 311
 - fipa_acl_envelope_Received_Copy, 311
 - fipa_acl_envelope_Received_Destroy, 312
 - fipa_acl_envelope_Received_New, 312
 - fipa_acl_envelope_Received_t, 311
 - fipa_acl_envelope_t, 311
 - fipa_acl_Param_Copy, 312
 - fipa_acl_Param_Destroy, 312
 - fipa_acl_Param_New, 312
 - fipa_acl_Param_t, 311
 - fipa_envelope_Compose, 313
 - fipa_envelope_Compose_acl_representation, 313
 - fipa_envelope_Compose_date, 313
 - fipa_envelope_Compose_envelope, 313
 - fipa_envelope_Compose_from, 313
 - fipa_envelope_Compose_params, 313
 - fipa_envelope_Compose_payload_encoding, 314
 - fipa_envelope_Compose_to, 314
 - fipa_envelope_HandleAclRepresentation, 314
 - fipa_envelope_HandleComments, 314
 - fipa_envelope_HandleDate, 314
 - fipa_envelope_HandleEnvelope, 315
 - fipa_envelope_HandleFrom, 315
 - fipa_envelope_HandleIntendedReceiver, 315
 - fipa_envelope_HandleParams, 315
 - fipa_envelope_HandlePayloadEncoding, 315
 - fipa_envelope_HandlePayloadLength, 316
 - fipa_envelope_HandleReceived, 316
 - fipa_envelope_HandleTo, 316
 - fipa_envelope_Parse, 316
 - fipa_envelope_ParseAddresses, 316
 - fipa_envelope_ParseAgentIdentifier, 317
 - fipa_envelope_ParseResolvers, 317
- fipa_acl_envelope_Copy
 - fipa_acl_envelope.h, 311
 - fipa_envelope.c, 239
- fipa_acl_envelope_Destroy
 - fipa_acl_envelope.h, 311
 - fipa_envelope.c, 239
- fipa_acl_envelope_New
 - fipa_acl_envelope.h, 311
 - fipa_envelope.c, 239
- fipa_acl_envelope_p
 - fipa_acl_envelope.h, 311
- fipa_acl_envelope_Received_Copy
 - fipa_acl_envelope.h, 311
 - fipa_envelope.c, 239
- fipa_acl_envelope_Received_Destroy
 - fipa_acl_envelope.h, 312
 - fipa_envelope.c, 239
- fipa_acl_envelope_Received_New
 - fipa_acl_envelope.h, 312
 - fipa_envelope.c, 240
- fipa_acl_envelope_Received_s, 58

- received_by, 58
- received_date, 58
- received_from, 58
- received_id, 58
- received_via, 58
- fipa_acl_envelope_Received_t
 - fipa_acl_envelope.h, 311
- fipa_acl_envelope_s, 60
 - num_params, 60
 - params, 60
- fipa_acl_envelope_t
 - fipa_acl_envelope.h, 311
- fipa_acl_message_Copy
 - fipa_acl.c, 228
 - fipa_acl.h, 300
- fipa_acl_message_Destroy
 - fipa_acl.c, 228
 - fipa_acl.h, 300
- fipa_acl_message_New
 - fipa_acl.c, 228
 - fipa_acl.h, 300
- fipa_acl_message_p
 - fipa_acl.h, 297
- fipa_acl_message_s, 61
 - content, 62
 - conversation_id, 62
 - encoding, 62
 - in_reply_to, 63
 - language, 62
 - ontology, 62
 - performative, 61
 - protocol, 62
 - receiver, 61
 - receiver_num, 61
 - reply_by, 63
 - reply_to, 62
 - reply_with, 63
 - sender, 61
- fipa_acl_message_t
 - fipa_acl.h, 297
- fipa_acl_Param_Copy
 - fipa_acl_envelope.h, 312
 - fipa_envelope.c, 240
- fipa_acl_Param_Destroy
 - fipa_acl_envelope.h, 312
 - fipa_envelope.c, 240
- fipa_acl_Param_New
 - fipa_acl_envelope.h, 312
 - fipa_envelope.c, 240
- fipa_acl_Param_s, 64
 - acl_representation, 64
 - comments, 64
 - date, 65
 - from, 64
 - intended_receiver, 65
 - payload_encoding, 65
 - payload_length, 64
 - received, 65
 - to, 64
- fipa_acl_Param_t
 - fipa_acl_envelope.h, 311
- fipa_acl_Parse
 - fipa_acl.c, 228
 - fipa_acl.h, 301
- fipa_agent_identifier_Compose
 - fipa_acl.c, 228
 - fipa_acl.h, 301
- fipa_agent_identifier_Copy
 - fipa_acl.c, 228
 - fipa_acl.h, 301
- fipa_agent_identifier_Destroy
 - fipa_acl.c, 229
 - fipa_acl.h, 301
- fipa_agent_identifier_New
 - fipa_acl.c, 229
 - fipa_acl.h, 301
- fipa_agent_identifier_p
 - fipa_acl.h, 297
- fipa_agent_identifier_Parse
 - fipa_acl.c, 229
 - fipa_acl.h, 301
- fipa_agent_identifier_s, 66
 - addresses, 66
 - name, 66
 - resolvers, 66
- fipa_agent_identifier_set_Compose
 - fipa_acl.c, 229
 - fipa_acl.h, 301
- fipa_agent_identifier_set_Copy
 - fipa_acl.c, 229
 - fipa_acl.h, 302
- fipa_agent_identifier_set_Destroy
 - fipa_acl.c, 230
 - fipa_acl.h, 302
- fipa_agent_identifier_set_New
 - fipa_acl.c, 230
 - fipa_acl.h, 302
- fipa_agent_identifier_set_p
 - fipa_acl.h, 297
- fipa_agent_identifier_set_Parse
 - fipa_acl.c, 230
 - fipa_acl.h, 302
- fipa_agent_identifier_set_s, 67
 - fipa_agent_identifiers, 67
 - num, 67
 - retain_order, 67
- fipa_agent_identifier_set_t
 - fipa_acl.h, 297

- fipa_agent_identifier_t
 - fipa_acl.h, 298
- fipa_agent_identifiers
 - fipa_agent_identifier_set_s, 67
- fipa_CheckNextToken
 - fipa_acl.c, 230
 - fipa_acl.h, 302
- fipa_DateTime_Compose
 - fipa_acl.c, 230
 - fipa_acl.h, 302
- fipa_DateTime_Copy
 - fipa_acl.c, 230
 - fipa_acl.h, 303
- fipa_DateTime_Destroy
 - fipa_acl.c, 231
 - fipa_acl.h, 303
- fipa_DateTime_New
 - fipa_acl.c, 231
 - fipa_acl.h, 303
- fipa_DateTime_p
 - fipa_acl.h, 298
- fipa_datetime_Parse
 - fipa_acl.c, 231
 - fipa_acl.h, 303
- fipa_DateTime_s, 68
 - day, 68
 - hour, 68
 - is_utc, 69
 - millisecond, 69
 - minute, 68
 - month, 68
 - second, 68
 - sign, 69
 - year, 68
- fipa_DateTime_t
 - fipa_acl.h, 298
- fipa_envelope.c
 - fipa_acl_envelope_Copy, 239
 - fipa_acl_envelope_Destroy, 239
 - fipa_acl_envelope_New, 239
 - fipa_acl_envelope_Received_Copy, 239
 - fipa_acl_envelope_Received_Destroy, 239
 - fipa_acl_envelope_Received_New, 240
 - fipa_acl_Param_Copy, 240
 - fipa_acl_Param_Destroy, 240
 - fipa_acl_Param_New, 240
 - fipa_envelope_Compose, 240
 - fipa_envelope_Compose__acl_representation, 240
 - fipa_envelope_Compose__date, 241
 - fipa_envelope_Compose__envelope, 241
 - fipa_envelope_Compose__from, 241
 - fipa_envelope_Compose__params, 241
 - fipa_envelope_Compose__payload_encoding, 241
 - fipa_envelope_Compose__to, 241
 - fipa_envelope_HandleAclRepresentation, 242
 - fipa_envelope_HandleComments, 242
 - fipa_envelope_HandleDate, 242
 - fipa_envelope_HandleEnvelope, 242
 - fipa_envelope_HandleFrom, 242
 - fipa_envelope_HandleIntendedReceiver, 243
 - fipa_envelope_HandleParams, 243
 - fipa_envelope_HandlePayloadEncoding, 243
 - fipa_envelope_HandlePayloadLength, 243
 - fipa_envelope_HandleReceived, 243
 - fipa_envelope_HandleTo, 244
 - fipa_envelope_Parse, 244
 - fipa_envelope_ParseAddresses, 244
 - fipa_envelope_ParseAgentIdentifier, 244
 - fipa_envelope_ParseResolvers, 245
- fipa_envelope_Compose
 - fipa_acl_envelope.h, 313
 - fipa_envelope.c, 240
- fipa_envelope_Compose__acl_representation
 - fipa_acl_envelope.h, 313
 - fipa_envelope.c, 240
- fipa_envelope_Compose__date
 - fipa_acl_envelope.h, 313
 - fipa_envelope.c, 241
- fipa_envelope_Compose__envelope
 - fipa_acl_envelope.h, 313
 - fipa_envelope.c, 241
- fipa_envelope_Compose__from
 - fipa_acl_envelope.h, 313
 - fipa_envelope.c, 241
- fipa_envelope_Compose__params
 - fipa_acl_envelope.h, 313
 - fipa_envelope.c, 241
- fipa_envelope_Compose__payload_encoding
 - fipa_acl_envelope.h, 314
 - fipa_envelope.c, 241
- fipa_envelope_Compose__to
 - fipa_acl_envelope.h, 314
 - fipa_envelope.c, 241
- fipa_envelope_HandleAclRepresentation
 - fipa_acl_envelope.h, 314
 - fipa_envelope.c, 242
- fipa_envelope_HandleComments
 - fipa_acl_envelope.h, 314
 - fipa_envelope.c, 242
- fipa_envelope_HandleDate
 - fipa_acl_envelope.h, 314
 - fipa_envelope.c, 242
- fipa_envelope_HandleEnvelope
 - fipa_acl_envelope.h, 315
 - fipa_envelope.c, 242

- fipa_envelope_HandleFrom
 - fipa_acl_envelope.h, [315](#)
 - fipa_envelope.c, [242](#)
- fipa_envelope_HandleIntendedReceiver
 - fipa_acl_envelope.h, [315](#)
 - fipa_envelope.c, [243](#)
- fipa_envelope_HandleParams
 - fipa_acl_envelope.h, [315](#)
 - fipa_envelope.c, [243](#)
- fipa_envelope_HandlePayloadEncoding
 - fipa_acl_envelope.h, [315](#)
 - fipa_envelope.c, [243](#)
- fipa_envelope_HandlePayloadLength
 - fipa_acl_envelope.h, [316](#)
 - fipa_envelope.c, [243](#)
- fipa_envelope_HandleReceived
 - fipa_acl_envelope.h, [316](#)
 - fipa_envelope.c, [243](#)
- fipa_envelope_HandleTo
 - fipa_acl_envelope.h, [316](#)
 - fipa_envelope.c, [244](#)
- fipa_envelope_Parse
 - fipa_acl_envelope.h, [316](#)
 - fipa_envelope.c, [244](#)
- fipa_envelope_ParseAddresses
 - fipa_acl_envelope.h, [316](#)
 - fipa_envelope.c, [244](#)
- fipa_envelope_ParseAgentIdentifier
 - fipa_acl_envelope.h, [317](#)
 - fipa_envelope.c, [244](#)
- fipa_envelope_ParseResolvers
 - fipa_acl_envelope.h, [317](#)
 - fipa_envelope.c, [245](#)
- fipa_expression_Compose
 - fipa_acl.c, [231](#)
 - fipa_acl.h, [303](#)
- fipa_expression_Copy
 - fipa_acl.c, [231](#)
 - fipa_acl.h, [303](#)
- fipa_expression_Destroy
 - fipa_acl.c, [231](#)
 - fipa_acl.h, [304](#)
- fipa_expression_New
 - fipa_acl.c, [232](#)
 - fipa_acl.h, [304](#)
- fipa_expression_p
 - fipa_acl.h, [298](#)
- fipa_expression_Parse
 - fipa_acl.c, [232](#)
 - fipa_acl.h, [304](#)
- fipa_expression_s, [70](#)
 - content, [70](#)
 - type, [70](#)
- fipa_expression_s::content_u, [71](#)
 - datetime, [71](#)
 - expression, [71](#)
 - number, [71](#)
 - string, [71](#)
 - word, [71](#)
- fipa_expression_t
 - fipa_acl.h, [298](#)
- fipa_expression_type_e
 - fipa_acl.h, [299](#)
- fipa_expression_type_t
 - fipa_acl.h, [298](#)
- fipa_GetAtom
 - fipa_acl.c, [232](#)
 - fipa_acl.h, [304](#)
- fipa_GetNextWord
 - fipa_acl.c, [232](#)
 - fipa_acl.h, [304](#)
- fipa_GetWholeToken
 - fipa_acl.c, [232](#)
 - fipa_acl.h, [305](#)
- fipa_message_parameter_Parse
 - fipa_acl.c, [232](#)
 - fipa_acl.h, [305](#)
- fipa_message_string_Copy
 - fipa_acl.c, [233](#)
 - fipa_acl.h, [305](#)
- fipa_message_string_Destroy
 - fipa_acl.c, [233](#)
 - fipa_acl.h, [305](#)
- fipa_message_string_New
 - fipa_acl.c, [233](#)
 - fipa_acl.h, [305](#)
- fipa_message_string_p
 - fipa_acl.h, [298](#)
- fipa_message_string_s, [72](#)
 - message, [72](#)
 - parse, [72](#)
- fipa_message_string_t
 - fipa_acl.h, [298](#)
- fipa_message_type_Parse
 - fipa_acl.c, [233](#)
 - fipa_acl.h, [305](#)
- fipa_number_Compose
 - fipa_acl.c, [233](#)
 - fipa_acl.h, [305](#)
- fipa_number_Copy
 - fipa_acl.c, [234](#)
 - fipa_acl.h, [306](#)
- fipa_number_Destroy
 - fipa_acl.c, [234](#)
 - fipa_acl.h, [306](#)
- fipa_number_New
 - fipa_acl.c, [234](#)
 - fipa_acl.h, [306](#)

- fipa_number_p
 - fipa_acl.h, 298
- fipa_number_s, 73
 - str, 73
- fipa_number_t
 - fipa_acl.h, 298
- fipa_performative_Compose
 - fipa_acl.c, 234
 - fipa_acl.h, 306
- fipa_performative_e
 - fipa_acl.h, 299
- fipa_Reply
 - fipa_acl.c, 234
 - fipa_acl.h, 306
- fipa_string_Compose
 - fipa_acl.c, 234
 - fipa_acl.h, 306
- fipa_string_Copy
 - fipa_acl.c, 235
 - fipa_acl.h, 307
- fipa_string_Destroy
 - fipa_acl.c, 235
 - fipa_acl.h, 307
- fipa_string_New
 - fipa_acl.c, 235
 - fipa_acl.h, 307
- fipa_string_p
 - fipa_acl.h, 298
- fipa_string_Parse
 - fipa_acl.c, 235
 - fipa_acl.h, 307
- fipa_string_s, 74
 - content, 74
- fipa_string_t
 - fipa_acl.h, 298
- fipa_url_Compose
 - fipa_acl.c, 235
 - fipa_acl.h, 307
- fipa_url_Copy
 - fipa_acl.c, 235
 - fipa_acl.h, 307
- fipa_url_Destroy
 - fipa_acl.c, 235
 - fipa_acl.h, 307
- fipa_url_New
 - fipa_acl.c, 235
 - fipa_acl.h, 308
- fipa_url_p
 - fipa_acl.h, 298
- fipa_url_Parse
 - fipa_acl.c, 236
 - fipa_acl.h, 308
- fipa_url_s, 75
 - str, 75
- fipa_url_sequence_Compose
 - fipa_acl.c, 236
 - fipa_acl.h, 308
- fipa_url_sequence_Copy
 - fipa_acl.c, 236
 - fipa_acl.h, 308
- fipa_url_sequence_Destroy
 - fipa_acl.c, 236
 - fipa_acl.h, 308
- fipa_url_sequence_New
 - fipa_acl.c, 236
 - fipa_acl.h, 308
- fipa_url_sequence_p
 - fipa_acl.h, 298
- fipa_url_sequence_Parse
 - fipa_acl.c, 236
 - fipa_acl.h, 308
- fipa_url_sequence_s, 76
 - num, 76
 - urls, 76
- fipa_url_sequence_t
 - fipa_acl.h, 298
- fipa_url_t
 - fipa_acl.h, 299
- fipa_word_Compose
 - fipa_acl.c, 237
 - fipa_acl.h, 309
- fipa_word_Copy
 - fipa_acl.c, 237
 - fipa_acl.h, 309
- fipa_word_Destroy
 - fipa_acl.c, 237
 - fipa_acl.h, 309
- fipa_word_New
 - fipa_acl.c, 237
 - fipa_acl.h, 309
- fipa_word_p
 - fipa_acl.h, 299
- fipa_word_Parse
 - fipa_acl.c, 237
 - fipa_acl.h, 309
- fipa_word_s, 77
 - content, 77
- fipa_word_t
 - fipa_acl.h, 299
- foo
 - class.cxx, 524
 - foo_c, 79
 - foo_s, 82
- FOO_BLUE
 - enum.cxx, 525
- FOO_ONE
 - enum.cxx, 525
- FOO_RED

- enum.cxx, 525
- FOO_TWO
 - enum.cxx, 525
- foo_c, 78
 - ~foo_c, 78
 - bar, 79
 - barfoo, 80
 - foo, 79
 - foo_c, 78
 - foo_c, 78
 - get_bar, 78
 - get_barfoo, 79
 - get_foo, 78
 - get_global, 79
 - global, 79
 - set_bar, 79
 - set_foo, 79
 - setfoobar, 79
- foo_default_int
 - function.cxx, 526
- foo_default_string
 - function.cxx, 526
- foo_enum_e
 - enum.cxx, 525
- foo_enum_t
 - enum.cxx, 525
- foo_float_function
 - function.cxx, 526
- foo_s, 81
 - ~foo_s, 81
 - bar, 82
 - foo, 82
 - foo_s, 81
 - foo_s, 81
 - get_bar, 81
 - get_foo, 81
 - set_bar, 81
 - set_foo, 82
- foo_t
 - struct.cxx, 527
- foo_void_function
 - function.cxx, 526
- FORBIDDEN
 - mtp_http.h, 393
- FREEMEM
 - fipa_acl.c, 227
- from
 - fipa_acl_Param_s, 64
- from_address
 - message_s, 155
- FSb
 - aes.c, 663
- FT0
 - aes.c, 663
- FT1
 - aes.c, 663
- FT2
 - aes.c, 663
- FT3
 - aes.c, 663
- function.cxx
 - foo_default_int, 526
 - foo_default_string, 526
 - foo_float_function, 526
 - foo_void_function, 526
- G
 - dhm_context, 55
- g_mc_platform
 - libmc.c, 460
- GENERATOR
 - dh_genprime.c, 802
- get_bar
 - foo_c, 78
 - foo_s, 81
- get_barfoo
 - foo_c, 79
- get_foo
 - foo_c, 78
 - foo_s, 81
- get_global
 - foo_c, 79
- GET_REQUEST
 - ssl_client1.c, 809
 - ssl_client2.c, 811
- GET_THREAD_MODE
 - macros.h, 376
- GET_UINT32_BE
 - aes.c, 660
 - des.c, 691
 - sha1.c, 742
 - sha2.c, 749
- GET_UINT32_LE
 - md4.c, 719
 - md5.c, 725
- GET_UINT64_BE
 - sha4.c, 756
- GetAgentExecEngine
 - LibMC::MCAgent, 144
- GetAgentFields
 - LibMC::MCAgent, 141
- GetAgentReturnData
 - LibMC::MCAgent, 145
- GetAgentXMLString
 - LibMC::MCAgent, 142
- giant
 - mc_platform_s, 95
- giant_cond

- mc_platform_s, 95
- giant_lock
 - mc_platform_s, 95
 - syncList_s, 196
- global
 - foo_c, 79
- GONE
 - mtp_http.h, 393
- GX
 - dhm_context, 56
- GY
 - dhm_context, 56
- HaltAgency
 - LibMC::MCAgency, 113
- Halted
 - LibMC::MCAgency, 109
- handler_HELP
 - cmd_prompt.c, 216
- handler_PRINT_CONNECTLIST
 - cmd_prompt.c, 216
- handler_PRINTLIST_AGENTS
 - cmd_prompt.c, 216
- handler_PRINTLIST_MESSAGE
 - cmd_prompt.c, 216
- handler_QUIT
 - cmd_prompt.c, 216
- handler_SEND
 - cmd_prompt.c, 216
- hardclock
 - timing.c, 780
 - timing.h, 643
- hardclock_init
 - timing.c, 780
- HAVE_STRDUP
 - config.h, 483
 - vcnet/config.h, 484
 - vcnet2005/config.h, 485
- HAVE_VSNPRINTF
 - config.h, 483
- havege.c
 - _CRT_SECURE_NO_DEPRECATED, 710
 - _havege_src, 712
 - havege_fill, 711
 - havege_init, 711
 - havege_rand, 711
 - ONE_ITERATION, 710
 - SWAP, 710
 - TST1_ENTER, 710
 - TST1_LEAVE, 711
 - TST2_ENTER, 711
 - TST2_LEAVE, 711
- havege.h
 - COLLECT_SIZE, 577
 - havege_init, 577
 - havege_rand, 577
- havege_fill
 - havege.c, 711
- havege_init
 - havege.c, 711
 - havege.h, 577
- havege_rand
 - havege.c, 711
 - havege.h, 577
- havege_state, 83
 - offset, 83
 - pool, 83
 - PT1, 83
 - PT2, 83
 - WALK, 83
- hello.c
 - _CRT_SECURE_NO_DEPRECATED, 793
 - main, 793
- help
 - commands.x.h, 279
- HERE
 - df.h, 287
- home
 - agent_s, 39
- home_port
 - agent_s, 39
- host
 - mtp_http_s, 160
- host_id.h
 - host_id_Compare, 318
 - host_id_Destroy, 318
 - host_id_Initialize, 318
 - host_id_p, 318
 - host_id_t, 318
- host_id_Compare
 - host_id.h, 318
- host_id_Destroy
 - host_id.h, 318
- host_id_Initialize
 - host_id.h, 318
- host_id_p
 - host_id.h, 318
- host_id_s, 85
 - hostname, 85
 - port, 85
- host_id_t
 - host_id.h, 318
- HOST_NAME_MAX
 - libmc.c, 412
- hostName
 - agency_s, 31
 - LibMC::MCAgency::MCAgency_t, 133
- hostname

- host_id_s, [85](#)
- mc_platform_s, [91](#)
- hour
 - _x509_time, [28](#)
 - fipa_DateTime_s, [68](#)
- hr_time, [86](#)
 - opaque, [86](#)
- hs_md5
 - ssl_context, [191](#)
- hs_sha1
 - ssl_context, [192](#)
- HTTP_CONNECT
 - mtp_http.h, [393](#)
- HTTP_DELETE
 - mtp_http.h, [393](#)
- HTTP_GET
 - mtp_http.h, [392](#)
- HTTP_HEAD
 - mtp_http.h, [392](#)
- HTTP_NUM_PERFORMATIVES
 - mtp_http.h, [393](#)
- HTTP_OPTIONS
 - mtp_http.h, [393](#)
- HTTP_PERFORMATIVE_UNDEF
 - mtp_http.h, [392](#)
- HTTP_PERFORMATIVE_ZERO
 - mtp_http.h, [392](#)
- HTTP_POST
 - mtp_http.h, [392](#)
- HTTP_PUT
 - mtp_http.h, [392](#)
- HTTP_RESPONSE
 - mtp_http.h, [393](#)
- HTTP_TRACE
 - mtp_http.h, [393](#)
- http_GetExpression
 - mtp_http.c, [480](#)
 - mtp_http.h, [394](#)
- http_GetToken
 - mtp_http.c, [480](#)
 - mtp_http.h, [394](#)
- HTTP_PARSE_EXPR
 - mtp_http.c, [479](#)
- http_ParseExpression
 - mtp_http.c, [480](#)
 - mtp_http.h, [394](#)
- http_ParseHeader
 - mtp_http.c, [481](#)
 - mtp_http.h, [395](#)
- http_performative
 - mtp_http_s, [160](#)
- http_performative_e
 - mtp_http.h, [392](#)
- HTTP_RESPONSE
 - ssl_server.c, [813](#)
- http_status_code
 - mtp_http_s, [160](#)
- http_status_code_e
 - mtp_http.h, [393](#)
- http_to_hostport
 - message.c, [476](#)
 - message.h, [389](#)
- http_type
 - message_s, [155](#)
- http_version
 - mtp_http_s, [160](#)
- id
 - agent_s, [38](#)
 - barrier_node_s, [47](#)
 - LibMC::MCAgent, [145](#)
 - syncListNode_s, [197](#)
- in_ctr
 - ssl_context, [187](#)
- in_hdr
 - ssl_context, [187](#)
- in_hslen
 - ssl_context, [188](#)
- in_left
 - ssl_context, [188](#)
- in_msg
 - ssl_context, [187](#)
- in_msglen
 - ssl_context, [188](#)
- in_msgtype
 - ssl_context, [187](#)
- in_offt
 - ssl_context, [187](#)
- in_reply_to
 - fipa_acl_message_s, [63](#)
- index
 - command_s, [50](#)
- index_compare
 - mxml-index.c, [497](#)
- index_find
 - mxml-index.c, [497](#)
- index_sort
 - mxml-index.c, [497](#)
- init_agent_status
 - agent_datastate_s, [35](#)
 - agent_task_s, [43](#)
- Initialize
 - LibMC::MCAgency, [111](#)
- Initialized
 - LibMC::MCAgency, [109](#)
- INSERT
 - df.h, [287](#)
- int

- data_structures.h, 285
- integer
 - mxml_value_u, 172
- intended_receiver
 - fipa_acl_Param_s, 65
- interp_options
 - mc_platform_s, 94
- interpreter_variable_data.h
 - interpreter_variable_data_Copy, 319
 - interpreter_variable_data_Destroy, 319
 - interpreter_variable_data_Initialize, 320
 - interpreter_variable_data_-
InitializeFromAgent, 320
 - interpreter_variable_data_New, 320
 - interpreter_variable_data_p, 319
 - interpreter_variable_data_t, 319
- interpreter_variable_data_Copy
 - agent_return_data.c, 208
 - interpreter_variable_data.h, 319
- interpreter_variable_data_Destroy
 - agent_return_data.c, 208
 - interpreter_variable_data.h, 319
- interpreter_variable_data_Initialize
 - agent_return_data.c, 208
 - interpreter_variable_data.h, 320
- interpreter_variable_data_InitializeFromAgent
 - agent_return_data.c, 208
 - interpreter_variable_data.h, 320
- interpreter_variable_data_New
 - agent_return_data.c, 209
 - interpreter_variable_data.h, 320
- interpreter_variable_data_p
 - interpreter_variable_data.h, 319
- interpreter_variable_data_s, 87
 - array_dim, 87
 - array_extent, 87
 - data, 88
 - data_type, 87
 - name, 87
 - size, 87
- interpreter_variable_data_t
 - interpreter_variable_data.h, 319
- ipad
 - md2_context, 148
 - md4_context, 150
 - md5_context, 152
 - sha1_context, 179
 - sha2_context, 181
 - sha4_context, 183
- is
 - commands.x.h, 279
- is224
 - sha2_context, 182
- is384
 - sha4_context, 184
- is_utc
 - fipa_DateTime_s, 69
- isHTTP
 - message_s, 154
- issuer
 - _x509_cert, 24
- issuer_id
 - _x509_cert, 25
- issuer_raw
 - _x509_cert, 24
- iv_dec
 - ssl_context, 194
- iv_enc
 - ssl_context, 194
- ivlen
 - ssl_context, 194
- K
 - dhm_context, 56
 - sha1.c, 742
 - sha4.c, 760
- KEY_LEN
 - rsa.h, 600
- KEY_SIZE
 - rsa_genkey.c, 806
- keylen
 - ssl_context, 193
- KT0
 - aes.c, 663
- KT1
 - aes.c, 663
- KT2
 - aes.c, 664
- KT3
 - aes.c, 664
- language
 - fipa_acl_message_s, 62
- last_child
 - mxml_node_s, 169
- last_error
 - agency_s, 33
 - LibMC::MCAgency::MCAgency_t, 134
- left
 - md2_context, 149
- len
 - _x509_buf, 22
 - dhm_context, 55
 - dynstring_s, 57
 - rsa_context, 174
- LENGTH_REQUIRED
 - mtp_http.h, 393
- LHs

- des.c, [696](#)
- LibMC, [13](#)
- libmc.c
 - g_mc_platform, [460](#)
 - HOST_NAME_MAX, [412](#)
 - MC_AclAddReceiver, [412](#)
 - MC_AclAddReceiver_chdl, [412](#)
 - MC_AclAddReplyTo, [412](#)
 - MC_AclAddReplyTo_chdl, [412](#)
 - MC_AclDestroy, [412](#)
 - MC_AclDestroy_chdl, [413](#)
 - MC_AclNew, [413](#)
 - MC_AclNew_chdl, [413](#)
 - MC_AclPost, [413](#)
 - MC_AclPost_chdl, [414](#)
 - MC_AclReply, [414](#)
 - MC_AclReply_chdl, [414](#)
 - MC_AclRetrieve, [414](#)
 - MC_AclRetrieve_chdl, [415](#)
 - MC_AclSend, [415](#)
 - MC_AclSend_chdl, [415](#)
 - MC_AclSetContent, [415](#)
 - MC_AclSetContent_chdl, [416](#)
 - MC_AclSetPerformative, [416](#)
 - MC_AclSetPerformative_chdl, [416](#)
 - MC_AclSetSender, [416](#)
 - MC_AclSetSender_chdl, [416](#)
 - MC_AclWaitRetrieve, [416](#)
 - MC_AclWaitRetrieve_chdl, [417](#)
 - MC_AddAgent, [417](#)
 - MC_AddAgent_chdl, [417](#)
 - MC_AgentVariableRetrieve, [417](#)
 - MC_AgentVariableRetrieve_chdl, [419](#)
 - MC_AgentVariableSave, [419](#)
 - MC_AgentVariableSave_chdl, [420](#)
 - MC_Barrier, [421](#)
 - MC_Barrier_chdl, [421](#)
 - MC_BarrierDelete, [421](#)
 - MC_BarrierDelete_chdl, [421](#)
 - MC_BarrierInit, [421](#)
 - MC_BarrierInit_chdl, [422](#)
 - MC_CallAgentFunc, [422](#)
 - MC_CallAgentFunc_chdl, [423](#)
 - MC_ChInitializeOptions, [423](#)
 - MC_ComposeAgent, [423](#)
 - MC_ComposeAgent_chdl, [424](#)
 - MC_CondBroadcast, [424](#)
 - MC_CondBroadcast_chdl, [425](#)
 - MC_CondReset, [425](#)
 - MC_CondReset_chdl, [425](#)
 - MC_CondSignal, [425](#)
 - MC_CondSignal_chdl, [426](#)
 - MC_CondWait, [427](#)
 - MC_CondWait_chdl, [428](#)
 - MC_CopyAgent, [428](#)
 - MC_DeleteAgent, [428](#)
 - MC_DeleteAgent_chdl, [428](#)
 - MC_DeregisterService, [429](#)
 - MC_DeregisterService_chdl, [429](#)
 - MC_DestroyServiceSearchResult, [429](#)
 - MC_DestroyServiceSearchResult_chdl, [429](#)
 - MC_End, [429](#)
 - MC_End_chdl, [430](#)
 - MC_FindAgentByID, [430](#)
 - MC_FindAgentByID_chdl, [431](#)
 - MC_FindAgentByName, [431](#)
 - MC_FindAgentByName_chdl, [432](#)
 - MC_GetAgentArrivalTime, [433](#)
 - MC_GetAgentArrivalTime_chdl, [433](#)
 - MC_GetAgentExecEngine, [433](#)
 - MC_GetAgentID, [433](#)
 - MC_GetAgentID_chdl, [433](#)
 - MC_GetAgentName, [433](#)
 - MC_GetAgentName_chdl, [433](#)
 - MC_GetAgentNumTasks, [434](#)
 - MC_GetAgentNumTasks_chdl, [435](#)
 - MC_GetAgentReturnData, [435](#)
 - MC_GetAgentStatus, [438](#)
 - MC_GetAgentStatus_chdl, [438](#)
 - MC_GetAgentType, [438](#)
 - MC_GetAgentXMLString, [438](#)
 - MC_GetAgentXMLString_chdl, [438](#)
 - MC_GetAllAgents, [438](#)
 - MC_GetTimeOfDay_chdl, [439](#)
 - MC_HaltAgency, [439](#)
 - MC_HaltAgency_chdl, [439](#)
 - MC_Initialize, [439](#)
 - MC_InitializeAgencyOptions, [440](#)
 - MC_LoadAgentFromFile, [441](#)
 - MC_MainLoop, [441](#)
 - MC_MutexLock, [441](#)
 - MC_MutexLock_chdl, [443](#)
 - MC_MutexUnlock, [443](#)
 - MC_MutexUnlock_chdl, [443](#)
 - MC_PrintAgentCode, [443](#)
 - MC_PrintAgentCode_chdl, [444](#)
 - MC_RegisterService, [444](#)
 - MC_RegisterService_chdl, [445](#)
 - MC_ResetSignal, [445](#)
 - MC_ResumeAgency, [446](#)
 - MC_ResumeAgency_chdl, [446](#)
 - MC_RetrieveAgent, [446](#)
 - MC_RetrieveAgent_chdl, [446](#)
 - MC_RetrieveAgentCode, [446](#)
 - MC_RetrieveAgentCode_chdl, [447](#)
 - MC_SearchForService, [447](#)
 - MC_SearchForService_chdl, [449](#)
 - MC_SemaphorePost, [449](#)

- MC_SemaphorePost_chdl, [450](#)
- MC_SemaphoreWait, [450](#)
- MC_SemaphoreWait_chdl, [451](#)
- MC_SendAgentMigrationMessage, [451](#)
- MC_SendAgentMigrationMessage_chdl, [451](#)
- MC_SendAgentMigrationMessageFile, [452](#)
- MC_SendAgentMigrationMessageFile_chdl, [452](#)
- MC_SendCh, [452](#)
- MC_SendSteerCommand, [452](#)
- MC_SendSteerCommand_chdl, [452](#)
- MC_SetAgentStatus, [452](#)
- MC_SetAgentStatus_chdl, [453](#)
- MC_SetDefaultAgentStatus, [453](#)
- MC_SetDefaultAgentStatus_chdl, [453](#)
- MC_SetThreadOff, [453](#)
- MC_SetThreadOn, [454](#)
- MC_SetThreadsAllOff, [454](#)
- MC_SetThreadsAllOn, [454](#)
- MC_Steer, [455](#)
- MC_SteerControl, [456](#)
- MC_SyncDelete, [456](#)
- MC_SyncDelete_chdl, [457](#)
- MC_SyncInit, [457](#)
- MC_SyncInit_chdl, [457](#)
- MC_TerminateAgent, [457](#)
- MC_TerminateAgent_chdl, [458](#)
- MC_WaitAgent, [458](#)
- MC_WaitRetrieveAgent, [458](#)
- MC_WaitSignal, [458](#)
- libmc.h
 - agency_p, [327](#)
 - agency_t, [327](#)
 - agent_p, [327](#)
 - agent_t, [327](#)
 - ERR, [328](#)
 - error_code_e, [327](#)
 - error_code_t, [327](#)
 - EXPORTMC, [326](#)
 - MC_AGENT_ACTIVE, [328](#)
 - MC_AGENT_NEUTRAL, [328](#)
 - MC_AGENT_SUSPENDED, [328](#)
 - MC_ALL_SIGNALS, [329](#)
 - MC_ERR, [327](#)
 - MC_ERR_CONNECT, [327](#), [328](#)
 - MC_ERR_EMPTY, [327](#), [328](#)
 - MC_ERR_INVALID, [327](#), [328](#)
 - MC_ERR_INVALID_ARGS, [327](#), [328](#)
 - MC_ERR_MEMORY, [327](#), [328](#)
 - MC_ERR_NOT_FOUND, [327](#), [328](#)
 - MC_ERR_PARSE, [327](#), [328](#)
 - MC_ERR_SEND, [327](#), [328](#)
 - MC_EXEC_AGENT, [329](#)
 - MC_LOCAL_AGENT, [328](#)
 - MC_NO_SIGNAL, [329](#)
 - MC_NO_STATUS, [328](#)
 - MC_NONE, [328](#)
 - MC_RECV_AGENT, [329](#)
 - MC_RECV_CONNECTION, [329](#)
 - MC_RECV_MESSAGE, [329](#)
 - MC_RECV_RETURN, [329](#)
 - MC_REMOTE_AGENT, [328](#)
 - MC_RESTART, [329](#)
 - MC_RETURN_AGENT, [328](#)
 - MC_RUN, [329](#)
 - MC_STOP, [329](#)
 - MC_SUCCESS, [327](#)
 - MC_SUSPEND, [329](#)
 - MC_THREAD_ACC, [329](#)
 - MC_THREAD_AGENT, [329](#)
 - MC_THREAD_ALL, [329](#)
 - MC_THREAD_AMS, [329](#)
 - MC_THREAD_CP, [329](#)
 - MC_THREAD_DF, [329](#)
 - MC_WAIT_CH, [328](#)
 - MC_WAIT_FINISHED, [328](#)
 - MC_WAIT_MESSGSEND, [328](#)
 - MC_WARN_DUPLICATE, [327](#), [328](#)
 - MC_AclAddReceiver, [330](#)
 - MC_AclAddReplyTo, [330](#)
 - MC_AclDestroy, [330](#)
 - MC_AclNew, [330](#)
 - MC_AclPost, [330](#)
 - MC_AclReply, [331](#)
 - MC_AclRetrieve, [331](#)
 - MC_AclSend, [331](#)
 - MC_AclSetContent, [332](#)
 - MC_AclSetPerformative, [332](#)
 - MC_AclSetSender, [332](#)
 - MC_AclWaitRetrieve, [332](#)
 - MC_AddAgent, [333](#)
 - MC_AgentStatus_e, [328](#)
 - MC_AgentType_e, [328](#)
 - MC_AgentVariableRetrieve, [333](#)
 - MC_AgentVariableSave, [335](#)
 - MC_BarrierDelete, [336](#)
 - MC_BarrierInit, [336](#)
 - MC_CallAgentFunc, [337](#)
 - MC_ChInitializeOptions, [338](#)
 - MC_ComposeAgent, [338](#)
 - MC_CondBroadcast, [339](#)
 - MC_CondReset, [339](#)
 - MC_CondSignal, [340](#)
 - MC_CondWait, [341](#)
 - MC_CopyAgent, [342](#)
 - MC_DeleteAgent, [342](#)
 - MC_DestroyServiceSearchResult, [343](#)
 - MC_End, [343](#)

- MC_FindAgentByID, 344
- MC_FindAgentByName, 344
- MC_GetAgentExecEngine, 346
- MC_GetAgentID, 346
- MC_GetAgentName, 346
- MC_GetAgentNumTasks, 346
- MC_GetAgentReturnData, 347
- MC_GetAgentStatus, 350
- MC_GetAgentType, 350
- MC_GetAgentXMLString, 351
- MC_HaltAgency, 351
- MC_Initialize, 351
- MC_InitializeAgencyOptions, 352
- MC_LoadAgentFromFile, 353
- MC_MainLoop, 353
- MC_MutexLock, 353
- MC_MutexUnlock, 355
- MC_PrintAgentCode, 355
- MC_RegisterService, 355
- MC_ResetSignal, 357
- MC_ResumeAgency, 357
- MC_RetrieveAgent, 357
- MC_RetrieveAgentCode, 358
- MC_SearchForService, 358
- MC_SemaphorePost, 360
- MC_SemaphoreWait, 361
- MC_SendAgentMigrationMessage, 362
- MC_SendAgentMigrationMessageFile, 362
- MC_SetAgentStatus, 363
- MC_SetDefaultAgentStatus, 363
- MC_SetThreadOff, 363
- MC_SetThreadOn, 364
- MC_SetThreadsAllOff, 364
- MC_SetThreadsAllOn, 364
- MC_Signal_e, 328
- MC_Steer, 365
- MC_SteerCommand_e, 329
- MC_SteerCommand_t, 327
- MC_SteerControl, 366
- MC_SyncDelete, 366
- MC_SyncInit, 367
- MC_TerminateAgent, 367
- MC_ThreadIndex_e, 329
- MC_Wait, 326
- MC_WaitAgent, 367
- MC_WaitRetrieveAgent, 368
- MC_WaitSignal, 368
- MCAgency_t, 327
- MAgencyOptions_t, 327
- MCAgent_t, 327
- LibMC::MCAclMessage, 98
 - AclMsg, 103
 - aclmsg_p, 103
 - AddReceiver, 101
 - AddReplyTo, 101
 - Destroy, 102
 - FIPA_ACCEPT_PROPOSAL, 99
 - FIPA_AGREE, 99
 - FIPA_CALL_FOR_PROPOSAL, 99
 - FIPA_CANCEL, 99
 - FIPA_CONFIRM, 99
 - FIPA_DISCONFIRM, 99
 - FIPA_ERROR, 99
 - FIPA_FAILURE, 99
 - FIPA_INFORM, 99
 - FIPA_INFORM_IF, 99
 - FIPA_INFORM_REF, 99
 - FIPA_NOT_UNDERSTOOD, 99
 - FIPA_PROPOGATE, 99
 - FIPA_PROPOSE, 99
 - FIPA_PROXY, 99
 - FIPA_QUERY_IF, 99
 - FIPA_QUERY_REF, 99
 - FIPA_REFUSE, 99
 - FIPA_REJECT_PROPOSAL, 99
 - FIPA_REQUEST, 99
 - FIPA_REQUEST_WHEN, 99
 - FIPA_REQUEST_WHENEVER, 99
 - FIPA_SUBSCRIBE, 99
 - FIPA_ZERO, 99
 - MC_FipaPerformative_e, 99
 - MCAclMessage, 100
 - New, 100
 - Reply, 100
 - SetContent, 102
 - SetPerformative, 100
 - SetSender, 101
- LibMC::MCAgency, 104
 - _MC_AclAddReceiver, 129
 - _MC_AclAddReplyTo, 129
 - _MC_AclDestroy, 129
 - _MC_AclNew, 129
 - _MC_AclPost, 129
 - _MC_AclReply, 129
 - _MC_AclRetrieve, 129
 - _MC_AclSend, 129
 - _MC_AclSetContent, 129
 - _MC_AclSetPerformative, 129
 - _MC_AclSetSender, 129
 - _MC_AclWaitRetrieve, 129
 - _MC_AddAgent, 127
 - _MC_BarrierDelete, 126
 - _MC_BarrierInit, 126
 - _MC_CallAgentFunc, 129
 - _MC_ChInitializeOptions, 124
 - _MC_CondBroadcast, 125
 - _MC_CondReset, 125
 - _MC_CondSignal, 125

- [_MC_CondWait, 125](#)
- [_MC_DeleteAgent, 127](#)
- [_MC_End, 124](#)
- [_MC_FindAgentByID, 127](#)
- [_MC_FindAgentByName, 127](#)
- [_MC_GetAgentExecEngine, 129](#)
- [_MC_GetAgentID, 127](#)
- [_MC_GetAgentName, 129](#)
- [_MC_GetAgentNumTasks, 129](#)
- [_MC_GetAgentReturnData, 129](#)
- [_MC_GetAgentStatus, 129](#)
- [_MC_GetAgentType, 129](#)
- [_MC_GetAgentXMLString, 129](#)
- [_MC_HaltAgency, 124](#)
- [_MC_Initialize, 124](#)
- [_MC_InitializeAgencyOptions, 124](#)
- [_MC_MainLoop, 125](#)
- [_MC_MutexLock, 125](#)
- [_MC_MutexUnlock, 125](#)
- [_MC_PrintAgentCode, 129](#)
- [_MC_RegisterService, 127](#)
- [_MC_ResetSignal, 126](#)
- [_MC_ResumeAgency, 124](#)
- [_MC_RetrieveAgent, 127](#)
- [_MC_RetrieveAgentCode, 129](#)
- [_MC_SearchForService, 127](#)
- [_MC_SemaphorePost, 126](#)
- [_MC_SemaphoreWait, 126](#)
- [_MC_SendAgentMigrationMessage, 125](#)
- [_MC_SendAgentMigrationMessageFile, 125](#)
- [_MC_SetAgentStatus, 129](#)
- [_MC_SetDefaultAgentStatus, 124](#)
- [_MC_SetThreadOff, 124](#)
- [_MC_SetThreadOn, 124](#)
- [_MC_SetThreadsAllOff, 124](#)
- [_MC_SetThreadsAllOn, 124](#)
- [_MC_Steer, 126](#)
- [_MC_SteerControl, 126](#)
- [_MC_SyncDelete, 126](#)
- [_MC_SyncInit, 126](#)
- [_MC_TerminateAgent, 129](#)
- [_MC_WaitAgent, 125](#)
- [_MC_WaitRetrieveAgent, 125](#)
- [_MC_WaitSignal, 126](#)
- [AclSend, 123](#)
- [AddAgent, 122](#)
- [Agency, 130](#)
- [agency_p, 129](#)
- [BarrierDelete, 119](#)
- [BarrierInit, 120](#)
- [CH_REGULARCH, 109](#)
- [CH_SAFECH, 109](#)
- [ChInitializeOptions, 111](#)
- [ChShellType, 109](#)
- [CondBroadcast, 116](#)
- [CondReset, 116](#)
- [CondSignal, 116](#)
- [CondWait, 116](#)
- [End, 111](#)
- [Ended, 109](#)
- [FindAgentByID, 122](#)
- [FindAgentByName, 122](#)
- [HaltAgency, 113](#)
- [Halted, 109](#)
- [Initialize, 111](#)
- [Initialized, 109](#)
- [LoadAgentMigrationMessageFile, 115](#)
- [MainLoop, 123](#)
- [MC_ALL_SIGNALS, 110](#)
- [MC_EXEC_AGENT, 110](#)
- [MC_NO_SIGNAL, 110](#)
- [MC_RECV_AGENT, 110](#)
- [MC_RECV_CONNECTION, 110](#)
- [MC_RECV_MESSAGE, 110](#)
- [MC_RECV_RETURN, 110](#)
- [MC_RESTART, 110](#)
- [MC_RUN, 110](#)
- [MC_STOP, 110](#)
- [MC_SUSPEND, 110](#)
- [MC_THREAD_ACC, 110](#)
- [MC_THREAD_AGENT, 110](#)
- [MC_THREAD_ALL, 110](#)
- [MC_THREAD_AMS, 110](#)
- [MC_THREAD_CP, 110](#)
- [MC_THREAD_DF, 110](#)
- [MC_Signal_e, 110](#)
- [MC_SteerCommand_e, 110](#)
- [MC_ThreadIndex_e, 109](#)
- [MCAgency, 111](#)
- [MCAgencyState, 109](#)
- [mcdll, 130](#)
- [MutexLock, 117](#)
- [MutexUnlock, 117](#)
- [NoState, 109](#)
- [options, 129](#)
- [Port, 130](#)
- [port, 130](#)
- [RegisterService, 121](#)
- [ResetSignal, 118](#)
- [ResumeAgency, 113](#)
- [RetrieveAgent, 123](#)
- [Running, 109](#)
- [SearchForService, 121](#)
- [SemaphorePost, 117](#)
- [SemaphoreWait, 118](#)
- [SendAgentMigrationMessage, 115](#)
- [SendAgentMigrationMessageFile, 114](#)
- [SetDefaultAgentStatus, 114](#)

- SetThreadOff, [113](#)
- SetThreadOn, [112](#)
- SetThreadsAllOff, [112](#)
- SetThreadsAllOn, [112](#)
- State, [131](#)
- state, [130](#)
- Steer, [120](#)
- SteerControl, [120](#)
- SyncDelete, [118](#)
- SyncInit, [119](#)
- WaitAgent, [114](#)
- WaitRetrieveAgent, [114](#)
- WaitSignal, [119](#)
- LibMC::MCAgency::ChOptions_t, [132](#)
 - chhome, [132](#)
 - shelltype, [132](#)
- LibMC::MCAgency::MCAgency_t, [133](#)
 - client, [133](#)
 - default_agentstatus, [134](#)
 - enable_security, [134](#)
 - filename, [133](#)
 - hostName, [133](#)
 - last_error, [134](#)
 - mc_platform, [134](#)
 - portno, [133](#)
 - portnoc, [133](#)
 - server, [133](#)
 - stack_size, [134](#)
 - threads, [134](#)
- LibMC::MCAgency::MCAgencyOptions_t, [135](#)
 - default_agent_status, [135](#)
 - enable_security, [135](#)
 - modified, [135](#)
 - stack_size, [135](#)
 - threads, [135](#)
- LibMC::MCAGENT, [138](#)
 - AcIPost, [143](#)
 - AcIRetrieve, [143](#)
 - AcIWaitRetrieve, [143](#)
 - Agent, [146](#)
 - agent_p, [145](#)
 - AgentID, [146](#)
 - AgentName, [146](#)
 - AgentNumTasks, [147](#)
 - AgentStatus, [147](#)
 - AgentType, [147](#)
 - CallAgentFunc, [144](#)
 - DeleteAgent, [141](#)
 - GetAgentExecEngine, [144](#)
 - GetAgentFields, [141](#)
 - GetAgentReturnData, [145](#)
 - GetAgentXMLString, [142](#)
 - id, [145](#)
 - MC_AGENT_ACTIVE, [140](#)
 - MC_AGENT_NEUTRAL, [140](#)
 - MC_AGENT_SUSPENDED, [140](#)
 - MC_LOCAL_AGENT, [140](#)
 - MC_NO_STATUS, [140](#)
 - MC_NONE, [140](#)
 - MC_REMOTE_AGENT, [140](#)
 - MC_RETURN_AGENT, [140](#)
 - MC_WAIT_CH, [140](#)
 - MC_WAIT_FINISHED, [140](#)
 - MC_WAIT_MESSGSEND, [140](#)
 - MC_AgentStatus_e, [140](#)
 - MC_AgentType_e, [140](#)
 - MCAGENT, [141](#)
 - name, [145](#)
 - numTasks, [145](#)
 - operator IntPtr, [141](#)
 - operator MCAGENT, [141](#)
 - PrintAgentCode, [142](#)
 - RetrieveAgentCode, [142](#)
 - status, [146](#)
 - TerminateAgent, [142](#)
 - ToString, [141](#)
 - type, [146](#)
- LibMC::Properties, [14](#)
- LibMC::Properties::Settings, [177](#)
 - Default, [178](#)
 - defaultInstance, [178](#)
 - SettingChangingEventHandler, [177](#)
 - Settings, [177](#)
 - SettingsSavingEventHandler, [177](#)
- list
 - barrier_queue_s, [48](#)
 - syncList_s, [196](#)
- list.c
 - list_pGetSize, [461](#)
 - ListAdd, [461](#)
 - ListDelete, [461](#)
 - ListGetHead, [461](#)
 - ListInitialize, [461](#)
 - ListInsert, [462](#)
 - ListPop, [462](#)
 - ListSearch, [462](#)
 - ListTerminate, [462](#)
- list.h
 - DATA, [463](#)
 - list_p, [464](#)
 - list_t, [464](#)
 - ListAdd, [464](#)
 - ListDelete, [464](#)
 - ListGetHead, [464](#)
 - ListGetSize, [464](#)
 - ListInitialize, [464](#)
 - ListInsert, [464](#)
 - listNode_p, [464](#)

- listNode_t, [464](#)
- ListPop, [464](#)
- ListSearch, [465](#)
- ListTerminate, [465](#)
- QUEUE_TEMPLATE, [463](#)
- list_p
 - list.h, [464](#)
- list_pGetSize
 - list.c, [461](#)
- list_s, [89](#)
 - listhead, [89](#)
 - size, [89](#)
- list_t
 - list.h, [464](#)
- ListAdd
 - list.c, [461](#)
 - list.h, [464](#)
- ListDelete
 - list.c, [461](#)
 - list.h, [464](#)
- listen_Thread
 - acc.c, [201](#)
 - acc.h, [247](#)
- listen_thread_arg
 - mc_platform_s, [93](#)
- ListGetHead
 - list.c, [461](#)
 - list.h, [464](#)
- ListGetSize
 - list.h, [464](#)
- listhead
 - list_s, [89](#)
- ListInitialize
 - list.c, [461](#)
 - list.h, [464](#)
- ListInsert
 - list.c, [462](#)
 - list.h, [464](#)
- listNode_p
 - list.h, [464](#)
- listNode_s, [90](#)
 - next, [90](#)
 - node_data, [90](#)
- listNode_t
 - list.h, [464](#)
- ListPop
 - list.c, [462](#)
 - list.h, [464](#)
- ListSearch
 - list.c, [462](#)
 - list.h, [465](#)
- ListTerminate
 - list.c, [462](#)
 - list.h, [465](#)
- LoadAgentMigrationMessageFile
 - LibMC::MCAgency, [115](#)
- lock
 - agent_s, [41](#)
 - barrier_node_s, [47](#)
 - barrier_queue_s, [48](#)
 - df.h, [290](#)
 - mc_rwlock_s, [96](#)
 - syncList_s, [196](#)
 - syncListNode_s, [197](#)
- LOCKED
 - mtp_http.h, [394](#)
- m
 - arc4_context, [46](#)
- mac_dec
 - ssl_context, [195](#)
- mac_enc
 - ssl_context, [194](#)
- maclen
 - ssl_context, [194](#)
- macros.h
 - CH_DATATYPE_SIZE, [372](#)
 - CH_DATATYPE_STR_TO_VAL, [372](#)
 - CH_DATATYPE_STRING, [373](#)
 - CH_DATATYPE_VALUE_STRING, [373](#)
 - CH_STRING_DATATYPE, [374](#)
 - CHECK_NULL, [374](#)
 - COND_BROADCAST, [375](#)
 - COND_DESTROY, [375](#)
 - COND_INIT, [375](#)
 - COND_RESET, [375](#)
 - COND_SIGNAL, [375](#)
 - COND_SLEEP, [376](#)
 - COND_SLEEP_ACTION, [376](#)
 - COND_T, [376](#)
 - COND_WAIT, [376](#)
 - GET_THREAD_MODE, [376](#)
 - MUTEX_DESTROY, [376](#)
 - MUTEX_INIT, [377](#)
 - MUTEX_LOCK, [377](#)
 - MUTEX_NEW, [377](#)
 - MUTEX_T, [377](#)
 - MUTEX_UNLOCK, [378](#)
 - PTHREAD_STACK_SIZE, [378](#)
 - RWLOCK_DESTROY, [378](#)
 - RWLOCK_INIT, [378](#)
 - RWLOCK_RDLOCK, [378](#)
 - RWLOCK_RDUNLOCK, [378](#)
 - RWLOCK_T, [378](#)
 - RWLOCK_WRLOCK, [379](#)
 - RWLOCK_WRUNLOCK, [379](#)
 - SEMAPHORE_DESTROY, [379](#)
 - SEMAPHORE_INIT, [379](#)

- SEMAPHORE_POST, 379
- SEMAPHORE_T, 379
- SEMAPHORE_WAIT, 379
- SET_THREAD_OFF, 379
- SET_THREAD_ON, 379
- SIGNAL, 380
- SLEEP_QUEUE, 380
- SLEEP_RESET, 380
- STRUCT, 380
- THREAD_CANCEL, 380
- THREAD_CREATE, 380
- THREAD_JOIN, 381
- THREAD_T, 381
- WAKE_QUEUE, 381
- WARN, 381
- mail_queue
 - agent_mailbox_s, 37
- mailbox
 - agent_s, 41
- main
 - aescript2.c, 792
 - benchmark.c, 816
 - dh_client.c, 800
 - dh_genprime.c, 802
 - dh_server.c, 803
 - hello.c, 793
 - md5sum.c, 794
 - mpi_demo.c, 805
 - mxmldoc.c, 521
 - rsa_genkey.c, 806
 - rsa_sign.c, 807
 - rsa_verify.c, 808
 - selftest.c, 818
 - sha1sum.c, 796
 - sha2sum.c, 798
 - ssl_client1.c, 810
 - ssl_client2.c, 812
 - ssl_server.c, 814
 - testmxml.c, 528
- MainLoop
 - LibMC::MCAgency, 123
- major_ver
 - ssl_context, 186
- master
 - ssl_context, 193
- max_pathlen
 - _x509_cert, 25
- max_ver
 - ssl_context, 186
- MC_AGENT_ACTIVE
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_AGENT_NEUTRAL
 - libmc.h, 328
- LibMC::MCAgent, 140
- MC_AGENT_SUSPENDED
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_ALL_SIGNALS
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_ERR
 - libmc.h, 327
 - mc_error.h, 382
- MC_ERR_CONNECT
 - libmc.h, 327, 328
 - mc_error.h, 382
- MC_ERR_EMPTY
 - libmc.h, 327, 328
 - mc_error.h, 382
- MC_ERR_INVALID
 - libmc.h, 327, 328
 - mc_error.h, 382
- MC_ERR_INVALID_ARGS
 - libmc.h, 327, 328
 - mc_error.h, 382, 383
- MC_ERR_MEMORY
 - libmc.h, 327, 328
 - mc_error.h, 382, 383
- MC_ERR_NOT_FOUND
 - libmc.h, 327, 328
 - mc_error.h, 382, 383
- MC_ERR_PARSE
 - libmc.h, 327, 328
 - mc_error.h, 382
- MC_ERR_SEND
 - libmc.h, 327, 328
 - mc_error.h, 382, 383
- mc_error.h
 - ERR, 382
 - MC_ERR, 382
 - MC_ERR_CONNECT, 382
 - MC_ERR_EMPTY, 382
 - MC_ERR_INVALID, 382
 - MC_ERR_INVALID_ARGS, 382, 383
 - MC_ERR_MEMORY, 382, 383
 - MC_ERR_NOT_FOUND, 382, 383
 - MC_ERR_PARSE, 382
 - MC_ERR_SEND, 382, 383
 - MC_SUCCESS, 382
 - MC_WARN_DUPLICATE, 382, 383
- MC_EXEC_AGENT
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_LOCAL_AGENT
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_NO_SIGNAL

- libmc.h, 329
- LibMC::MCAgency, 110
- MC_NO_STATUS
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_NONE
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_RECV_AGENT
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_RECV_CONNECTION
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_RECV_MESSAGE
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_RECV_RETURN
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_REMOTE_AGENT
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_RESTART
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_RETURN_AGENT
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_RUN
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_STOP
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_SUCCESS
 - libmc.h, 327
 - mc_error.h, 382
- MC_SUSPEND
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_THREAD_ACC
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_THREAD_AGENT
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_THREAD_ALL
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_THREAD_AMS
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_THREAD_CP
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_THREAD_DF
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_WAIT_CH
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_WAIT_FINISHED
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_WAIT_MESSGSEND
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_WARN_DUPLICATE
 - libmc.h, 327, 328
 - mc_error.h, 382, 383
- MC_AclAddReceiver
 - libmc.c, 412
 - libmc.h, 330
- MC_AclAddReceiver_chdl
 - agent_lib.h, 255
 - libmc.c, 412
- MC_AclAddReplyTo
 - libmc.c, 412
 - libmc.h, 330
- MC_AclAddReplyTo_chdl
 - agent_lib.h, 255
 - libmc.c, 412
- MC_AclDestroy
 - libmc.c, 412
 - libmc.h, 330
- MC_AclDestroy_chdl
 - agent_lib.h, 255
 - libmc.c, 413
- MC_AclNew
 - libmc.c, 413
 - libmc.h, 330
- MC_AclNew_chdl
 - agent_lib.h, 255
 - libmc.c, 413
- MC_AclPost
 - libmc.c, 413
 - libmc.h, 330
- MC_AclPost_chdl
 - agent_lib.h, 255
 - libmc.c, 414
- MC_AclReply
 - libmc.c, 414
 - libmc.h, 331
- MC_AclReply_chdl
 - agent_lib.h, 255
 - libmc.c, 414
- MC_AclRetrieve

- libmc.c, 414
- libmc.h, 331
- MC_AclRetrieve_chdl
 - agent_lib.h, 256
 - libmc.c, 415
- MC_AclSend
 - libmc.c, 415
 - libmc.h, 331
- MC_AclSend_chdl
 - agent_lib.h, 256
 - libmc.c, 415
- MC_AclSetContent
 - libmc.c, 415
 - libmc.h, 332
- MC_AclSetContent_chdl
 - agent_lib.h, 256
 - libmc.c, 416
- MC_AclSetPerformative
 - libmc.c, 416
 - libmc.h, 332
- MC_AclSetPerformative_chdl
 - agent_lib.h, 256
 - libmc.c, 416
- MC_AclSetSender
 - libmc.c, 416
 - libmc.h, 332
- MC_AclSetSender_chdl
 - agent_lib.h, 256
 - libmc.c, 416
- MC_AclWaitRetrieve
 - libmc.c, 416
 - libmc.h, 332
- MC_AclWaitRetrieve_chdl
 - agent_lib.h, 256
 - libmc.c, 417
- MC_AddAgent
 - libmc.c, 417
 - libmc.h, 333
- MC_AddAgent_chdl
 - agent_lib.h, 256
 - libmc.c, 417
- MC_AgentStatus_e
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_AgentType_e
 - libmc.h, 328
 - LibMC::MCAgent, 140
- MC_AgentVariableRetrieve
 - libmc.c, 417
 - libmc.h, 333
- MC_AgentVariableRetrieve_chdl
 - agent_lib.h, 257
 - libmc.c, 419
- MC_AgentVariableSave
 - libmc.c, 419
 - libmc.h, 335
- MC_AgentVariableSave_chdl
 - agent_lib.h, 257
 - libmc.c, 420
- MC_Barrier
 - libmc.c, 421
- MC_Barrier_chdl
 - agent_lib.h, 257
 - libmc.c, 421
- MC_BarrierDelete
 - libmc.c, 421
 - libmc.h, 336
- MC_BarrierDelete_chdl
 - agent_lib.h, 257
 - libmc.c, 421
- MC_BarrierInit
 - libmc.c, 421
 - libmc.h, 336
- MC_BarrierInit_chdl
 - agent_lib.h, 257
 - libmc.c, 422
- MC_CallAgentFunc
 - libmc.c, 422
 - libmc.h, 337
- MC_CallAgentFunc_chdl
 - agent_lib.h, 257
 - libmc.c, 423
- MC_ChInitializeOptions
 - libmc.c, 423
 - libmc.h, 338
- MC_ComposeAgent
 - libmc.c, 423
 - libmc.h, 338
- MC_ComposeAgent_chdl
 - agent_lib.h, 257
 - libmc.c, 424
- MC_CondBroadcast
 - libmc.c, 424
 - libmc.h, 339
- MC_CondBroadcast_chdl
 - agent_lib.h, 258
 - libmc.c, 425
- MC_CondReset
 - libmc.c, 425
 - libmc.h, 339
- MC_CondReset_chdl
 - agent_lib.h, 258
 - libmc.c, 425
- MC_CondSignal
 - libmc.c, 425
 - libmc.h, 340
- MC_CondSignal_chdl
 - agent_lib.h, 258

- libmc.c, 426
- MC_CondWait
 - libmc.c, 427
 - libmc.h, 341
- MC_CondWait_chdl
 - agent_lib.h, 258
 - libmc.c, 428
- MC_CopyAgent
 - libmc.c, 428
 - libmc.h, 342
- MC_DeleteAgent
 - libmc.c, 428
 - libmc.h, 342
- MC_DeleteAgent_chdl
 - agent_lib.h, 258
 - libmc.c, 428
- MC_DeregisterService
 - libmc.c, 429
- MC_DeregisterService_chdl
 - agent_lib.h, 258
 - libmc.c, 429
- MC_DestroyServiceSearchResult
 - libmc.c, 429
 - libmc.h, 343
- MC_DestroyServiceSearchResult_chdl
 - agent_lib.h, 258
 - libmc.c, 429
- MC_End
 - libmc.c, 429
 - libmc.h, 343
- MC_End_chdl
 - agent_lib.h, 259
 - libmc.c, 430
- mc_error.h
 - error_code_e, 382
 - error_code_t, 382
- MC_FindAgentByID
 - libmc.c, 430
 - libmc.h, 344
- MC_FindAgentByID_chdl
 - agent_lib.h, 259
 - libmc.c, 431
- MC_FindAgentByName
 - libmc.c, 431
 - libmc.h, 344
- MC_FindAgentByName_chdl
 - agent_lib.h, 259
 - libmc.c, 432
- MC_FipaPerformative_e
 - LibMC::MCAclMessage, 99
- MC_GetAgentArrivalTime
 - libmc.c, 433
- MC_GetAgentArrivalTime_chdl
 - libmc.c, 433
- MC_GetAgentExecEngine
 - libmc.c, 433
 - libmc.h, 346
- MC_GetAgentID
 - libmc.c, 433
 - libmc.h, 346
- MC_GetAgentID_chdl
 - agent_lib.h, 259
 - libmc.c, 433
- MC_GetAgentName
 - libmc.c, 433
 - libmc.h, 346
- MC_GetAgentName_chdl
 - agent_lib.h, 259
 - libmc.c, 433
- MC_GetAgentNumTasks
 - libmc.c, 434
 - libmc.h, 346
- MC_GetAgentNumTasks_chdl
 - agent_lib.h, 259
 - libmc.c, 435
- MC_GetAgentReturnData
 - libmc.c, 435
 - libmc.h, 347
- MC_GetAgentStatus
 - libmc.c, 438
 - libmc.h, 350
- MC_GetAgentStatus_chdl
 - agent_lib.h, 259
 - libmc.c, 438
- MC_GetAgentType
 - libmc.c, 438
 - libmc.h, 350
- MC_GetAgentXMLString
 - libmc.c, 438
 - libmc.h, 351
- MC_GetAgentXMLString_chdl
 - agent_lib.h, 259
 - libmc.c, 438
- MC_GetAllAgents
 - libmc.c, 438
- MC_GetTimeOfDay_chdl
 - agent_lib.h, 260
 - libmc.c, 439
- MC_HaltAgency
 - libmc.c, 439
 - libmc.h, 351
- MC_HaltAgency_chdl
 - agent_lib.h, 260
 - libmc.c, 439
- MC_Initialize
 - libmc.c, 439
 - libmc.h, 351
- MC_InitializeAgencyOptions

- libmc.c, [440](#)
- libmc.h, [352](#)
- MC_LoadAgentFromFile
 - libmc.c, [441](#)
 - libmc.h, [353](#)
- MC_MainLoop
 - libmc.c, [441](#)
 - libmc.h, [353](#)
- MC_MutexLock
 - libmc.c, [441](#)
 - libmc.h, [353](#)
- MC_MutexLock_chdl
 - agent_lib.h, [260](#)
 - libmc.c, [443](#)
- MC_MutexUnlock
 - libmc.c, [443](#)
 - libmc.h, [355](#)
- MC_MutexUnlock_chdl
 - agent_lib.h, [260](#)
 - libmc.c, [443](#)
- mc_platform
 - agency_s, [32](#)
 - agent_s, [41](#)
 - df.h, [290](#)
 - LibMC::MCAgency::MCAgency_t, [134](#)
- mc_platform.c
 - DEFAULT_HOSTNAME_LENGTH, [466](#)
 - mc_platform_Destroy, [466](#)
 - mc_platform_Initialize, [466](#)
- mc_platform.h
 - mc_platform_Destroy, [384](#)
 - mc_platform_Initialize, [385](#)
 - mc_platform_p, [384](#)
 - mc_platform_t, [384](#)
- mc_platform_Destroy
 - mc_platform.c, [466](#)
 - mc_platform.h, [384](#)
- mc_platform_Initialize
 - mc_platform.c, [466](#)
 - mc_platform.h, [385](#)
- mc_platform_p
 - mc_platform.h, [384](#)
- mc_platform_s, [91](#)
 - acc, [93](#)
 - agent_queue, [92](#)
 - ams, [92](#)
 - asm_message_queue, [92](#)
 - barrier_queue, [93](#)
 - client_thread_arg, [93](#)
 - cmd_prompt, [93](#)
 - connection_queue, [92](#)
 - default_agentstatus, [93](#)
 - df, [92](#)
 - err, [91](#)
 - giant, [95](#)
 - giant_cond, [95](#)
 - giant_lock, [95](#)
 - hostname, [91](#)
 - interp_options, [94](#)
 - listen_thread_arg, [93](#)
 - MC_signal, [94](#)
 - MC_signal_cond, [94](#)
 - MC_signal_lock, [94](#)
 - MC_steer_command, [94](#)
 - MC_steer_cond, [95](#)
 - MC_steer_lock, [94](#)
 - MC_sync_cond, [94](#)
 - MC_sync_lock, [94](#)
 - message_queue, [92](#)
 - port, [92](#)
 - quit, [95](#)
 - quit_lock, [95](#)
 - stack_size, [93](#)
 - syncList, [93](#)
- mc_platform_t
 - mc_platform.h, [384](#)
- MC_PrintAgentCode
 - libmc.c, [443](#)
 - libmc.h, [355](#)
- MC_PrintAgentCode_chdl
 - agent_lib.h, [260](#)
 - libmc.c, [444](#)
- MC_RegisterService
 - libmc.c, [444](#)
 - libmc.h, [355](#)
- MC_RegisterService_chdl
 - agent_lib.h, [260](#)
 - libmc.c, [445](#)
- MC_ResetSignal
 - libmc.c, [445](#)
 - libmc.h, [357](#)
- MC_ResumeAgency
 - libmc.c, [446](#)
 - libmc.h, [357](#)
- MC_ResumeAgency_chdl
 - agent_lib.h, [260](#)
 - libmc.c, [446](#)
- MC_RetrieveAgent
 - libmc.c, [446](#)
 - libmc.h, [357](#)
- MC_RetrieveAgent_chdl
 - agent_lib.h, [260](#)
 - libmc.c, [446](#)
- MC_RetrieveAgentCode
 - libmc.c, [446](#)
 - libmc.h, [358](#)
- MC_RetrieveAgentCode_chdl
 - agent_lib.h, [261](#)

- libmc.c, [447](#)
- mc_rwlock.c
 - mc_rwlock_destroy, [468](#)
 - mc_rwlock_init, [468](#)
 - mc_rwlock_rdlock, [468](#)
 - mc_rwlock_rdunlock, [468](#)
 - mc_rwlock_wrlock, [468](#)
 - mc_rwlock_wrunlock, [469](#)
- mc_rwlock.h
 - mc_rwlock_destroy, [386](#)
 - mc_rwlock_init, [386](#)
 - mc_rwlock_p, [386](#)
 - mc_rwlock_rdlock, [386](#)
 - mc_rwlock_rdunlock, [387](#)
 - mc_rwlock_t, [386](#)
 - mc_rwlock_wrlock, [387](#)
 - mc_rwlock_wrunlock, [387](#)
- mc_rwlock_destroy
 - mc_rwlock.c, [468](#)
 - mc_rwlock.h, [386](#)
- mc_rwlock_init
 - mc_rwlock.c, [468](#)
 - mc_rwlock.h, [386](#)
- mc_rwlock_p
 - mc_rwlock.h, [386](#)
- mc_rwlock_rdlock
 - mc_rwlock.c, [468](#)
 - mc_rwlock.h, [386](#)
- mc_rwlock_rdunlock
 - mc_rwlock.c, [468](#)
 - mc_rwlock.h, [387](#)
- mc_rwlock_s, [96](#)
 - cond, [96](#)
 - lock, [96](#)
 - num_readers, [96](#)
 - write_flag, [96](#)
 - write_request, [96](#)
- mc_rwlock_t
 - mc_rwlock.h, [386](#)
- mc_rwlock_wrlock
 - mc_rwlock.c, [468](#)
 - mc_rwlock.h, [387](#)
- mc_rwlock_wrunlock
 - mc_rwlock.c, [469](#)
 - mc_rwlock.h, [387](#)
- MC_SearchForService
 - libmc.c, [447](#)
 - libmc.h, [358](#)
- MC_SearchForService_chdl
 - agent_lib.h, [261](#)
 - libmc.c, [449](#)
- MC_SemaphorePost
 - libmc.c, [449](#)
 - libmc.h, [360](#)
- MC_SemaphorePost_chdl
 - agent_lib.h, [261](#)
 - libmc.c, [450](#)
- MC_SemaphoreWait
 - libmc.c, [450](#)
 - libmc.h, [361](#)
- MC_SemaphoreWait_chdl
 - agent_lib.h, [261](#)
 - libmc.c, [451](#)
- MC_SendAgentMigrationMessage
 - libmc.c, [451](#)
 - libmc.h, [362](#)
- MC_SendAgentMigrationMessage_chdl
 - agent_lib.h, [261](#)
 - libmc.c, [451](#)
- MC_SendAgentMigrationMessageFile
 - libmc.c, [452](#)
 - libmc.h, [362](#)
- MC_SendAgentMigrationMessageFile_chdl
 - agent_lib.h, [261](#)
 - libmc.c, [452](#)
- MC_SendCh
 - libmc.c, [452](#)
- MC_SendSteerCommand
 - libmc.c, [452](#)
- MC_SendSteerCommand_chdl
 - agent_lib.h, [261](#)
 - libmc.c, [452](#)
- MC_SetAgentStatus
 - libmc.c, [452](#)
 - libmc.h, [363](#)
- MC_SetAgentStatus_chdl
 - agent_lib.h, [261](#)
 - libmc.c, [453](#)
- MC_SetDefaultAgentStatus
 - libmc.c, [453](#)
 - libmc.h, [363](#)
- MC_SetDefaultAgentStatus_chdl
 - agent_lib.h, [261](#)
 - libmc.c, [453](#)
- MC_SetThreadOff
 - libmc.c, [453](#)
 - libmc.h, [363](#)
- MC_SetThreadOn
 - libmc.c, [454](#)
 - libmc.h, [364](#)
- MC_SetThreadsAllOff
 - libmc.c, [454](#)
 - libmc.h, [364](#)
- MC_SetThreadsAllOn
 - libmc.c, [454](#)
 - libmc.h, [364](#)
- MC_signal
 - mc_platform_s, [94](#)

- MC_signal_cond
 - mc_platform_s, 94
- MC_Signal_e
 - libmc.h, 328
 - LibMC::MCAgency, 110
- MC_signal_lock
 - mc_platform_s, 94
- MC_Steer
 - libmc.c, 455
 - libmc.h, 365
- MC_steer_command
 - mc_platform_s, 94
- MC_steer_cond
 - mc_platform_s, 95
- MC_steer_lock
 - mc_platform_s, 94
- MC_SteerCommand_e
 - libmc.h, 329
 - LibMC::MCAgency, 110
- MC_SteerCommand_t
 - libmc.h, 327
- MC_SteerControl
 - libmc.c, 456
 - libmc.h, 366
- MC_sync_cond
 - mc_platform_s, 94
- MC_sync_lock
 - mc_platform_s, 94
- MC_SyncDelete
 - libmc.c, 456
 - libmc.h, 366
- MC_SyncDelete_chdl
 - agent_lib.h, 262
 - libmc.c, 457
- MC_SyncInit
 - libmc.c, 457
 - libmc.h, 367
- MC_SyncInit_chdl
 - agent_lib.h, 262
 - libmc.c, 457
- MC_TerminateAgent
 - libmc.c, 457
 - libmc.h, 367
- MC_TerminateAgent_chdl
 - agent_lib.h, 262
 - libmc.c, 458
- MC_ThreadIndex_e
 - libmc.h, 329
 - LibMC::MCAgency, 109
- MC_Wait
 - libmc.h, 326
- MC_WaitAgent
 - libmc.c, 458
 - libmc.h, 367
- MC_WaitRetrieveAgent
 - libmc.c, 458
 - libmc.h, 368
- MC_WaitSignal
 - libmc.c, 458
 - libmc.h, 368
- MCAclMessage
 - LibMC::MCAclMessage, 100
- MCAgency
 - LibMC::MCAgency, 111
- MCAgency_t
 - libmc.h, 327
- MCAgencyOptions_s, 136
 - default_agent_status, 136
 - enable_security, 136
 - modified, 136
 - stack_size, 136
 - threads, 136
- MCAgencyOptions_t
 - libmc.h, 327
- MCAgencyState
 - LibMC::MCAgency, 109
- MCAgent
 - LibMC::MCAgent, 141
- MCAgent_t
 - libmc.h, 327
- mcddl
 - LibMC::MCAgency, 130
- md2
 - md2.c, 714
 - md2.h, 580
- md2.c
 - _CRT_SECURE_NO_DEPRECATED, 714
 - _md2_src, 717
 - md2, 714
 - md2_file, 714
 - md2_finish, 714
 - md2_hmac, 715
 - md2_hmac_finish, 715
 - md2_hmac_starts, 715
 - md2_hmac_update, 715
 - md2_process, 716
 - md2_self_test, 716
 - md2_starts, 716
 - md2_update, 716
 - PI_SUBST, 717
- md2.h
 - md2, 580
 - md2_file, 580
 - md2_finish, 580
 - md2_hmac, 580
 - md2_hmac_finish, 581
 - md2_hmac_starts, 581
 - md2_hmac_update, 581

- md2_self_test, [582](#)
 - md2_starts, [582](#)
 - md2_update, [582](#)
- md2_context, [148](#)
 - buffer, [148](#)
 - cksum, [148](#)
 - ipad, [148](#)
 - left, [149](#)
 - opad, [148](#)
 - state, [148](#)
- md2_file
 - md2.c, [714](#)
 - md2.h, [580](#)
- md2_finish
 - md2.c, [714](#)
 - md2.h, [580](#)
- md2_hmac
 - md2.c, [715](#)
 - md2.h, [580](#)
- md2_hmac_finish
 - md2.c, [715](#)
 - md2.h, [581](#)
- md2_hmac_starts
 - md2.c, [715](#)
 - md2.h, [581](#)
- md2_hmac_update
 - md2.c, [715](#)
 - md2.h, [581](#)
- md2_process
 - md2.c, [716](#)
- md2_self_test
 - md2.c, [716](#)
 - md2.h, [582](#)
- md2_starts
 - md2.c, [716](#)
 - md2.h, [582](#)
- md2_update
 - md2.c, [716](#)
 - md2.h, [582](#)
- md4
 - md4.c, [720](#)
 - md4.h, [584](#)
- md4.c
 - _CRT_SECURE_NO_DEPRECATED, [719](#)
 - _md4_src, [723](#)
 - F, [719](#)
 - GET_UINT32_LE, [719](#)
 - md4, [720](#)
 - md4_file, [720](#)
 - md4_finish, [720](#)
 - md4_hmac, [721](#)
 - md4_hmac_finish, [721](#)
 - md4_hmac_starts, [721](#)
 - md4_hmac_update, [722](#)
 - md4_padding, [723](#)
 - md4_process, [722](#)
 - md4_self_test, [722](#)
 - md4_starts, [722](#)
 - md4_update, [722](#)
 - P, [719](#)
 - PUT_UINT32_LE, [719](#)
 - S, [720](#)
- md4.h
 - md4, [584](#)
 - md4_file, [584](#)
 - md4_finish, [584](#)
 - md4_hmac, [584](#)
 - md4_hmac_finish, [585](#)
 - md4_hmac_starts, [585](#)
 - md4_hmac_update, [585](#)
 - md4_self_test, [586](#)
 - md4_starts, [586](#)
 - md4_update, [586](#)
- md4_context, [150](#)
 - buffer, [150](#)
 - ipad, [150](#)
 - opad, [150](#)
 - state, [150](#)
 - total, [150](#)
- md4_file
 - md4.c, [720](#)
 - md4.h, [584](#)
- md4_finish
 - md4.c, [720](#)
 - md4.h, [584](#)
- md4_hmac
 - md4.c, [721](#)
 - md4.h, [584](#)
- md4_hmac_finish
 - md4.c, [721](#)
 - md4.h, [585](#)
- md4_hmac_starts
 - md4.c, [721](#)
 - md4.h, [585](#)
- md4_hmac_update
 - md4.c, [722](#)
 - md4.h, [585](#)
- md4_padding
 - md4.c, [723](#)
- md4_process
 - md4.c, [722](#)
- md4_self_test
 - md4.c, [722](#)
 - md4.h, [586](#)
- md4_starts
 - md4.c, [722](#)
 - md4.h, [586](#)
- md4_update

- md4.c, [722](#)
- md4.h, [586](#)
- md5
 - md5.c, [726](#)
 - md5.h, [588](#)
- md5.c
 - _CRT_SECURE_NO_DEPRECATED, [725](#)
 - _md5_src, [729](#)
 - F, [725](#)
 - GET_UINT32_LE, [725](#)
 - md5, [726](#)
 - md5_file, [726](#)
 - md5_finish, [726](#)
 - md5_hmac, [727](#)
 - md5_hmac_finish, [727](#)
 - md5_hmac_starts, [727](#)
 - md5_hmac_update, [728](#)
 - md5_padding, [729](#)
 - md5_process, [728](#)
 - md5_self_test, [728](#)
 - md5_starts, [728](#)
 - md5_update, [728](#)
 - P, [725](#)
 - PUT_UINT32_LE, [725](#)
 - S, [726](#)
- md5.h
 - md5, [588](#)
 - md5_file, [588](#)
 - md5_finish, [588](#)
 - md5_hmac, [588](#)
 - md5_hmac_finish, [589](#)
 - md5_hmac_starts, [589](#)
 - md5_hmac_update, [589](#)
 - md5_self_test, [590](#)
 - md5_starts, [590](#)
 - md5_update, [590](#)
- md5_check
 - md5sum.c, [794](#)
- md5_context, [152](#)
 - buffer, [152](#)
 - ipad, [152](#)
 - opad, [152](#)
 - state, [152](#)
 - total, [152](#)
- md5_file
 - md5.c, [726](#)
 - md5.h, [588](#)
- md5_finish
 - md5.c, [726](#)
 - md5.h, [588](#)
- md5_hmac
 - md5.c, [727](#)
 - md5.h, [588](#)
- md5_hmac_finish
 - md5.c, [727](#)
 - md5.h, [589](#)
- md5_hmac_update
 - md5.c, [728](#)
 - md5.h, [589](#)
- md5_padding
 - md5.c, [729](#)
- md5_print
 - md5sum.c, [794](#)
- md5_process
 - md5.c, [728](#)
- md5_self_test
 - md5.c, [728](#)
 - md5.h, [590](#)
- md5_starts
 - md5.c, [728](#)
 - md5.h, [590](#)
- md5_update
 - md5.c, [728](#)
 - md5.h, [590](#)
- md5_wrapper
 - md5sum.c, [794](#)
- md5sum.c
 - _CRT_SECURE_NO_DEPRECATED, [794](#)
 - main, [794](#)
 - md5_check, [794](#)
 - md5_print, [794](#)
 - md5_wrapper, [794](#)
- message
 - dynstring_s, [57](#)
 - fipa_message_string_s, [72](#)
- message.c
 - http_to_hostport, [476](#)
 - message_Copy, [476](#)
 - message_Destroy, [476](#)
 - message_InitializeFromAgent, [476](#)
 - message_InitializeFromConnection, [476](#)
 - message_InitializeFromString, [477](#)
 - message_New, [477](#)
 - message_Send, [477](#)
 - SOCKET_INPUT_SIZE, [476](#)
- message.h
 - AGENT_UPDATE, [389](#)
 - CANCEL, [389](#)
 - ENCRYPTED_DATA, [389](#)
 - ENCRYPTION_INITIALIZE, [389](#)
 - FIPA_ACL, [389](#)
 - http_to_hostport, [389](#)
 - message_Copy, [389](#)
 - message_Destroy, [389](#)
 - message_InitializeFromAgent, [390](#)

- message_InitializeFromConnection, 390
- message_InitializeFromString, 390
- message_New, 390
- message_p, 388
- message_Send, 390
- message_t, 388
- message_type_e, 389
- message_type_t, 389
- MOBILE_AGENT, 389
- N_UNDRSTD, 389
- NUM_MESSAGE_TYPE, 389
- QUER_IF, 389
- QUER_REF, 389
- RELAY, 389
- REQUEST, 389
- REQUEST_ENCRYPTION_INITIALIZE, 389
- RETURN_MSG, 389
- SUBSCRIBE, 389
- message_body
 - message_s, 155
- message_Copy
 - message.c, 476
 - message.h, 389
- message_Destroy
 - message.c, 476
 - message.h, 389
- message_id
 - message_s, 154
- message_InitializeFromAgent
 - message.c, 476
 - message.h, 390
- message_InitializeFromConnection
 - message.c, 476
 - message.h, 390
- message_InitializeFromString
 - message.c, 477
 - message.h, 390
- message_New
 - message.c, 477
 - message.h, 390
- message_p
 - message.h, 388
- message_parts
 - mtp_http_s, 162
- message_queue
 - mc_platform_s, 92
- message_queue.c
 - message_queue_SendOutgoing, 478
- message_queue_SendOutgoing
 - message_queue.c, 478
- message_s, 154
 - addr, 154
 - agent_xml_flag, 156
 - connect_id, 154
 - from_address, 155
 - http_type, 155
 - isHTTP, 154
 - message_body, 155
 - message_id, 154
 - message_type, 155
 - target, 156
 - to_address, 156
 - update_name, 155
 - update_num, 155
 - xml_payload, 155
 - xml_root, 155
- message_Send
 - message.c, 477
 - message.h, 390
- message_t
 - message.h, 388
- message_type
 - message_s, 155
- message_type_e
 - message.h, 389
- message_type_t
 - message.h, 389
- message_xml_parse
 - xml_parser.c, 836
 - xml_parser.h, 404
- message_xml_parse__message
 - xml_parser.c, 836
 - xml_parser.h, 404
- METHOD_NOT_ALLOWED
 - mtp_http.h, 393
- millisecond
 - fipa_DateTime_s, 69
- mime_version
 - mtp_http_s, 162
- min
 - _x509_time, 28
- minlen
 - ssl_context, 193
- minor_ver
 - ssl_context, 186
- minute
 - fipa_DateTime_s, 68
- MISC
 - df.h, 287
- MOBILE_AGENT
 - message.h, 389
- MODE_DECRYPT
 - aescript2.c, 791
- MODE_ENCRYPT
 - aescript2.c, 791
- modified

- LibMC::MCAgency::MCAgencyOptions_t, 135
- MCAgencyOptions_s, 136
- mon
 - _x509_time, 28
- month
 - fipa_DateTime_s, 68
- mpi, 157
 - n, 157
 - p, 157
 - s, 157
- mpi_add_abs
 - bignum.c, 673
 - bignum.h, 551
- mpi_add_int
 - bignum.c, 673
 - bignum.h, 551
- mpi_add_mpi
 - bignum.c, 673
 - bignum.h, 552
- mpi_cmp_abs
 - bignum.c, 673
 - bignum.h, 552
- mpi_cmp_int
 - bignum.c, 674
 - bignum.h, 552
- mpi_cmp_mpi
 - bignum.c, 674
 - bignum.h, 552
- mpi_copy
 - bignum.c, 674
 - bignum.h, 553
- mpi_demo.c
 - _CRT_SECURE_NO_DEPRECATED, 805
 - main, 805
- mpi_div_int
 - bignum.c, 674
 - bignum.h, 553
- mpi_div_mpi
 - bignum.c, 675
 - bignum.h, 553
- mpi_exp_mod
 - bignum.c, 675
 - bignum.h, 554
- mpi_free
 - bignum.c, 675
 - bignum.h, 554
- mpi_gcd
 - bignum.c, 676
 - bignum.h, 554
- mpi_gen_prime
 - bignum.c, 676
 - bignum.h, 555
- mpi_get_digit
 - bignum.c, 676
- mpi_grow
 - bignum.c, 677
 - bignum.h, 555
- mpi_init
 - bignum.c, 677
 - bignum.h, 555
- mpi_inv_mod
 - bignum.c, 677
 - bignum.h, 555
- mpi_is_prime
 - bignum.c, 677
 - bignum.h, 556
- mpi_lsb
 - bignum.c, 678
 - bignum.h, 556
- mpi_lset
 - bignum.c, 678
 - bignum.h, 556
- mpi_mod_int
 - bignum.c, 678
 - bignum.h, 556
- mpi_mod_mpi
 - bignum.c, 678
 - bignum.h, 557
- mpi_montg_init
 - bignum.c, 678
- mpi_montmul
 - bignum.c, 679
- mpi_montred
 - bignum.c, 679
- mpi_msb
 - bignum.c, 679
 - bignum.h, 557
- mpi_mul_hlp
 - bignum.c, 679
- mpi_mul_int
 - bignum.c, 679
 - bignum.h, 557
- mpi_mul_mpi
 - bignum.c, 679
 - bignum.h, 557
- mpi_read_binary
 - bignum.c, 680
 - bignum.h, 558
- mpi_read_file
 - bignum.c, 680
 - bignum.h, 558
- mpi_read_string
 - bignum.c, 680
 - bignum.h, 558
- mpi_self_test
 - bignum.c, 681
 - bignum.h, 559

- mpi_shift_l
 - bignum.c, 681
 - bignum.h, 559
- mpi_shift_r
 - bignum.c, 681
 - bignum.h, 559
- mpi_sub_abs
 - bignum.c, 682
 - bignum.h, 560
- mpi_sub_hlp
 - bignum.c, 682
- mpi_sub_int
 - bignum.c, 682
 - bignum.h, 560
- mpi_sub_mpi
 - bignum.c, 682
 - bignum.h, 560
- mpi_swap
 - bignum.c, 682
 - bignum.h, 560
- mpi_write_binary
 - bignum.c, 683
 - bignum.h, 560
- mpi_write_file
 - bignum.c, 683
 - bignum.h, 561
- mpi_write_hlp
 - bignum.c, 683
- mpi_write_string
 - bignum.c, 684
 - bignum.h, 561
- mtp_http.h
 - ACCEPTED, 393
 - BAD_REQUEST, 393
 - CONFLICT, 393
 - CONTINUE, 393
 - CREATED, 393
 - EXPECTATION_FAILED, 394
 - FAILED_DEPENDANCY, 394
 - FORBIDDEN, 393
 - GONE, 393
 - HTTP_CONNECT, 393
 - HTTP_DELETE, 393
 - HTTP_GET, 392
 - HTTP_HEAD, 392
 - HTTP_NUM_PERFORMATIVES, 393
 - HTTP_OPTIONS, 393
 - HTTP_PERFORMATIVE_UNDEF, 392
 - HTTP_PERFORMATIVE_ZERO, 392
 - HTTP_POST, 392
 - HTTP_PUT, 392
 - HTTP_RESPONSE, 393
 - HTTP_TRACE, 393
 - LENGTH_REQUIRED, 393
 - LOCKED, 394
 - METHOD_NOT_ALLOWED, 393
 - MULTI_STATUS, 393
 - NO_CONTENT, 393
 - NON_AUTHORITATIVE_INFORMATION, 393
 - NOT_ACCEPTABLE, 393
 - NOT_FOUND, 393
 - OK, 393
 - PARTIAL_CONTENT, 393
 - PAYMENT_REQUIRED, 393
 - PRECONDITION_FAILED, 393
 - PROCESSING, 393
 - PROXY_AUTHENTICATION_REQUIRED, 393
 - REQUEST_TIMEOUT, 393
 - REQUEST_URI_TOO_LONG, 393
 - REQUESTED_RANGE_NOT_SATISFIABLE, 393
 - REQUEST_ENTITY_TOO_LARGE, 393
 - RESET_CONTENT, 393
 - RETRY_WITH, 394
 - SWITCHING_PROTOCOLS, 393
 - UNAUTHORIZED, 393
 - UNORDERED_COLLECTION, 394
 - UNPROCESSABLE_ENTITY, 394
 - UNSUPPORTED_MEDIA_TYPE, 393
 - UPGRADE_REQUIRED, 394
- mtp_http.c
 - http_GetExpression, 480
 - http_GetToken, 480
 - HTTP_PARSE_EXPR, 479
 - http_ParseExpression, 480
 - http_ParseHeader, 481
 - mtp_http_ComposeMessage, 481
 - mtp_http_CreateMessage, 481
 - mtp_http_Destroy, 481
 - mtp_http_InitializeFromConnection, 482
 - mtp_http_New, 482
 - mtp_http_Parse, 482
 - mtp_http_ParseResponse, 482
 - SAFE_FREE, 480
- mtp_http.h
 - http_GetExpression, 394
 - http_GetToken, 394
 - http_ParseExpression, 394
 - http_ParseHeader, 395
 - http_performative_e, 392
 - http_status_code_e, 393
 - mtp_http_ComposeMessage, 395
 - mtp_http_content_t, 392
 - mtp_http_CreateMessage, 395
 - mtp_http_Destroy, 395
 - mtp_http_InitializeFromConnection, 395

- mtp_http_New, 395
- mtp_http_p, 392
- mtp_http_Parse, 396
- mtp_http_t, 392
- SOCKET_INPUT_SIZE, 392
- mtp_http_ComposeMessage
 - mtp_http.c, 481
 - mtp_http.h, 395
- mtp_http_content_s, 159
 - content_type, 159
 - data, 159
- mtp_http_content_t
 - mtp_http.h, 392
- mtp_http_CreateMessage
 - mtp_http.c, 481
 - mtp_http.h, 395
- mtp_http_Destroy
 - mtp_http.c, 481
 - mtp_http.h, 395
- mtp_http_InitializeFromConnection
 - mtp_http.c, 482
 - mtp_http.h, 395
- mtp_http_New
 - mtp_http.c, 482
 - mtp_http.h, 395
- mtp_http_p
 - mtp_http.h, 392
- mtp_http_Parse
 - mtp_http.c, 482
 - mtp_http.h, 396
- mtp_http_ParseResponse
 - mtp_http.c, 482
- mtp_http_s, 160
 - accept_ranges, 161
 - boundary, 162
 - cache_control, 162
 - connection, 161
 - content, 162
 - content_length, 161
 - content_type, 161
 - date, 161
 - host, 160
 - http_performative, 160
 - http_status_code, 160
 - http_version, 160
 - message_parts, 162
 - mime_version, 162
 - response_code, 162
 - response_string, 162
 - return_code, 161
 - server, 161
 - target, 161
 - user_agent, 162
- mtp_http_t
 - mtp_http.h, 392
- MUL
 - aes.c, 660
- MULADDC_CORE
 - bn_asm.h, 563
- MULADDC_INIT
 - bn_asm.h, 563
- MULADDC_STOP
 - bn_asm.h, 563
- MULTI_STATUS
 - mtp_http.h, 393
- MUTEX_DESTROY
 - macros.h, 376
- MUTEX_INIT
 - macros.h, 377
- MUTEX_LOCK
 - macros.h, 377
- MUTEX_NEW
 - macros.h, 377
- MUTEX_T
 - macros.h, 377
- MUTEX_UNLOCK
 - macros.h, 378
- MutexLock
 - LibMC::MCAgency, 117
- MutexUnlock
 - LibMC::MCAgency, 117
- mxml-attr.c
 - mxmlElementGetAttr, 487
 - mxmlElementSetAttr, 487
- mxml-entity.c
 - callbacks, 489
 - default_callback, 488
 - mxmlEntityAddCallback, 488
 - mxmlEntityGetName, 488
 - mxmlEntityGetValue, 488
 - mxmlEntityRemoveCallback, 488
 - num_callbacks, 489
- mxml-file.c
 - ENCODE_UTF16BE, 491
 - ENCODE_UTF16LE, 491
 - ENCODE_UTF8, 491
 - mxml_add_char, 491
 - mxml_bad_char, 491
 - mxml_custom_load_cb, 496
 - mxml_custom_save_cb, 496
 - mxml_error_cb, 496
 - mxml_fd_getc, 492
 - mxml_fd_putc, 492
 - mxml_fd_read, 492
 - mxml_fd_write, 492
 - mxml_fdbuf_t, 491
 - mxml_file_getc, 492
 - mxml_file_putc, 492

- [mxml_get_entity, 492](#)
 - [mxml_load_data, 493](#)
 - [mxml_parse_element, 493](#)
 - [mxml_string_getc, 493](#)
 - [mxml_string_putc, 493](#)
 - [mxml_write_name, 493](#)
 - [mxml_write_node, 493](#)
 - [mxml_write_string, 494](#)
 - [mxml_write_ws, 494](#)
 - [mxmlLoadFd, 494](#)
 - [mxmlLoadFile, 494](#)
 - [mxmlLoadString, 494](#)
 - [mxmlSaveAllocString, 495](#)
 - [mxmlSaveFd, 495](#)
 - [mxmlSaveFile, 495](#)
 - [mxmlSaveString, 495](#)
 - [mxmlSetCustomHandlers, 495](#)
 - [mxmlSetErrorCallback, 495](#)
- [mxml-index.c](#)
 - [index_compare, 497](#)
 - [index_find, 497](#)
 - [index_sort, 497](#)
 - [mxmlIndexDelete, 497](#)
 - [mxmlIndexEnum, 498](#)
 - [mxmlIndexFind, 498](#)
 - [mxmlIndexNew, 498](#)
 - [mxmlIndexReset, 498](#)
- [mxml-node.c](#)
 - [mxml_new, 499](#)
 - [mxmlAdd, 499](#)
 - [mxmlDelete, 499](#)
 - [mxmlNewCustom, 500](#)
 - [mxmlNewElement, 500](#)
 - [mxmlNewInteger, 500](#)
 - [mxmlNewOpaque, 500](#)
 - [mxmlNewReal, 500](#)
 - [mxmlNewText, 501](#)
 - [mxmlNewTextf, 501](#)
 - [mxmlRemove, 501](#)
- [mxml-private.c](#)
 - [mxml_error, 502](#)
 - [mxml_error_cb, 503](#)
 - [mxml_integer_cb, 502](#)
 - [mxml_opaque_cb, 502](#)
 - [mxml_real_cb, 502](#)
- [mxml-search.c](#)
 - [mxmlFindElement, 504](#)
 - [mxmlWalkNext, 504](#)
 - [mxmlWalkPrev, 504](#)
- [mxml-set.c](#)
 - [mxmlSetCustom, 505](#)
 - [mxmlSetElement, 505](#)
 - [mxmlSetInteger, 505](#)
 - [mxmlSetOpaque, 505](#)
 - [mxmlSetReal, 505](#)
 - [mxmlSetText, 505](#)
 - [mxmlSetTextf, 506](#)
- [mxml-string.c](#)
 - [mxml_strdupf, 507](#)
 - [mxml_vsnprintf, 507](#)
- [mxml.h](#)
 - [MXML_CUSTOM, 513](#)
 - [MXML_ELEMENT, 513](#)
 - [MXML_INTEGER, 513](#)
 - [MXML_OPAQUE, 513](#)
 - [MXML_REAL, 513](#)
 - [MXML_TEXT, 513](#)
 - [MXML_ADD_AFTER, 510](#)
 - [MXML_ADD_BEFORE, 510](#)
 - [MXML_ADD_TO_PARENT, 510](#)
 - [mxml_attr_t, 512](#)
 - [mxml_custom_load_cb_t, 512](#)
 - [mxml_custom_save_cb_t, 512](#)
 - [mxml_custom_t, 512](#)
 - [MXML_DESCEND, 510](#)
 - [MXML_DESCEND_FIRST, 510](#)
 - [mxml_element_t, 513](#)
 - [mxml_error, 513](#)
 - [mxml_index_t, 513](#)
 - [MXML_INTEGER_CALLBACK, 511](#)
 - [mxml_integer_cb, 513](#)
 - [MXML_NO_CALLBACK, 511](#)
 - [MXML_NO_DESCEND, 511](#)
 - [MXML_NO_PARENT, 511](#)
 - [mxml_node_t, 513](#)
 - [MXML_OPAQUE_CALLBACK, 511](#)
 - [mxml_opaque_cb, 513](#)
 - [MXML_REAL_CALLBACK, 511](#)
 - [mxml_real_cb, 513](#)
 - [MXML_TAB, 511](#)
 - [MXML_TEXT_CALLBACK, 512](#)
 - [mxml_text_t, 513](#)
 - [mxml_type_e, 513](#)
 - [mxml_type_t, 513](#)
 - [mxml_value_t, 513](#)
 - [MXML_WRAP, 512](#)
 - [MXML_WS_AFTER_CLOSE, 512](#)
 - [MXML_WS_AFTER_OPEN, 512](#)
 - [MXML_WS_BEFORE_CLOSE, 512](#)
 - [MXML_WS_BEFORE_OPEN, 512](#)
 - [mxmlAdd, 514](#)
 - [mxmlDelete, 514](#)
 - [mxmlElementGetAttr, 514](#)
 - [mxmlElementSetAttr, 514](#)
 - [mxmlEntityAddCallback, 515](#)
 - [mxmlEntityGetName, 515](#)
 - [mxmlEntityGetValue, 515](#)
 - [mxmlEntityRemoveCallback, 515](#)

- mxmxmlFindElement, 515
- mxmxmlIndexDelete, 515
- mxmxmlIndexEnum, 515
- mxmxmlIndexFind, 516
- mxmxmlIndexNew, 516
- mxmxmlIndexReset, 516
- mxmxmlLoadFd, 516
- mxmxmlLoadFile, 516
- mxmxmlLoadString, 516
- mxmxmlNewCustom, 516
- mxmxmlNewElement, 516
- mxmxmlNewInteger, 517
- mxmxmlNewOpaque, 517
- mxmxmlNewReal, 517
- mxmxmlNewText, 517
- mxmxmlNewTextf, 517
- mxmxmlRemove, 518
- mxmxmlSaveAllocString, 518
- mxmxmlSaveFd, 518
- mxmxmlSaveFile, 518
- mxmxmlSaveString, 518
- mxmxmlSetCustom, 518
- mxmxmlSetCustomHandlers, 518
- mxmxmlSetElement, 518
- mxmxmlSetErrorCallback, 518
- mxmxmlSetInteger, 518
- mxmxmlSetOpaque, 519
- mxmxmlSetReal, 519
- mxmxmlSetText, 519
- mxmxmlSetTextf, 519
- mxmxmlWalkNext, 519
- mxmxmlWalkPrev, 519
- MXXML_CUSTOM
 - mxmxml.h, 513
- MXXML_ELEMENT
 - mxmxml.h, 513
- MXXML_INTEGER
 - mxmxml.h, 513
- MXXML_OPAQUE
 - mxmxml.h, 513
- MXXML_REAL
 - mxmxml.h, 513
- MXXML_TEXT
 - mxmxml.h, 513
- MXXML_ADD_AFTER
 - mxmxml.h, 510
- MXXML_ADD_BEFORE
 - mxmxml.h, 510
- mxmxml_add_char
 - mxmxml-file.c, 491
- MXXML_ADD_TO_PARENT
 - mxmxml.h, 510
- mxmxml_attr_s, 163
 - name, 163
 - value, 163
- mxmxml_attr_t
 - mxmxml.h, 512
- mxmxml_bad_char
 - mxmxml-file.c, 491
- mxmxml_custom_load_cb
 - mxmxml-file.c, 496
- mxmxml_custom_load_cb_t
 - mxmxml.h, 512
- mxmxml_custom_s, 164
 - data, 164
 - destroy, 164
- mxmxml_custom_save_cb
 - mxmxml-file.c, 496
- mxmxml_custom_save_cb_t
 - mxmxml.h, 512
- mxmxml_custom_t
 - mxmxml.h, 512
- MXXML_DESCEND
 - mxmxml.h, 510
- MXXML_DESCEND_FIRST
 - mxmxml.h, 510
- mxmxml_element_t
 - mxmxml.h, 513
- mxmxml_error
 - mxmxml-private.c, 502
 - mxmxml.h, 513
- mxmxml_error_cb
 - mxmxml-file.c, 496
 - mxmxml-private.c, 503
- mxmxml_fd_getc
 - mxmxml-file.c, 492
- mxmxml_fd_putc
 - mxmxml-file.c, 492
- mxmxml_fd_read
 - mxmxml-file.c, 492
- mxmxml_fd_write
 - mxmxml-file.c, 492
- mxmxml_fdbuf_s, 165
 - buffer, 165
 - current, 165
 - end, 165
 - fd, 165
- mxmxml_fdbuf_t
 - mxmxml-file.c, 491
- mxmxml_file_getc
 - mxmxml-file.c, 492
- mxmxml_file_putc
 - mxmxml-file.c, 492
- mxmxml_get_entity
 - mxmxml-file.c, 492
- mxmxml_index_s, 166
 - alloc_nodes, 166
 - attr, 166

- cur_node, [166](#)
- nodes, [166](#)
- num_nodes, [166](#)
- mxml_index_t
 - mxml.h, [513](#)
- MXML_INTEGER_CALLBACK
 - mxml.h, [511](#)
- mxml_integer_cb
 - mxml-private.c, [502](#)
 - mxml.h, [513](#)
- mxml_load_data
 - mxml-file.c, [493](#)
- mxml_new
 - mxml-node.c, [499](#)
- MXML_NO_CALLBACK
 - mxml.h, [511](#)
- MXML_NO_DESCEND
 - mxml.h, [511](#)
- MXML_NO_PARENT
 - mxml.h, [511](#)
- mxml_node_s, [168](#)
 - child, [168](#)
 - last_child, [169](#)
 - next, [168](#)
 - parent, [168](#)
 - prev, [168](#)
 - type, [168](#)
 - value, [169](#)
- mxml_node_t
 - mxml.h, [513](#)
- MXML_OPAQUE_CALLBACK
 - mxml.h, [511](#)
- mxml_opaque_cb
 - mxml-private.c, [502](#)
 - mxml.h, [513](#)
- mxml_parse_element
 - mxml-file.c, [493](#)
- MXML_REAL_CALLBACK
 - mxml.h, [511](#)
- mxml_real_cb
 - mxml-private.c, [502](#)
 - mxml.h, [513](#)
- mxml_strdup
 - config.h, [483](#)
 - mxml-string.c, [507](#)
 - vcnet/config.h, [484](#)
 - vcnet2005/config.h, [485](#)
- mxml_string_getc
 - mxml-file.c, [493](#)
- mxml_string_putc
 - mxml-file.c, [493](#)
- MXML_TAB
 - mxml.h, [511](#)
- MXML_TEXT_CALLBACK
 - mxml.h, [512](#)
- mxml_text_s, [170](#)
 - string, [170](#)
 - whitespace, [170](#)
- mxml_text_t
 - mxml.h, [513](#)
- mxml_type_e
 - mxml.h, [513](#)
- mxml_type_t
 - mxml.h, [513](#)
- mxml_value_s, [171](#)
 - attrs, [171](#)
 - name, [171](#)
 - num_attrs, [171](#)
- mxml_value_t
 - mxml.h, [513](#)
- mxml_value_u, [172](#)
 - custom, [173](#)
 - element, [172](#)
 - integer, [172](#)
 - opaque, [172](#)
 - real, [172](#)
 - text, [172](#)
- MXML_VERSION
 - config.h, [483](#)
 - vcnet/config.h, [484](#)
 - vcnet2005/config.h, [485](#)
- mxml_vsnprintf
 - mxml-string.c, [507](#)
 - vcnet/config.h, [484](#)
 - vcnet2005/config.h, [485](#)
- MXML_WRAP
 - mxml.h, [512](#)
- mxml_write_name
 - mxml-file.c, [493](#)
- mxml_write_node
 - mxml-file.c, [493](#)
- mxml_write_string
 - mxml-file.c, [494](#)
- mxml_write_ws
 - mxml-file.c, [494](#)
- MXML_WS_AFTER_CLOSE
 - mxml.h, [512](#)
- MXML_WS_AFTER_OPEN
 - mxml.h, [512](#)
- MXML_WS_BEFORE_CLOSE
 - mxml.h, [512](#)
- MXML_WS_BEFORE_OPEN
 - mxml.h, [512](#)
- mxmlAdd
 - mxml-node.c, [499](#)
 - mxml.h, [514](#)
- mxmlDelete
 - mxml-node.c, [499](#)

- mxmml.h, 514
- mxmmlDoc.c
 - add_variable, 521
 - main, 521
 - safe_strepy, 521
 - scan_file, 521
 - sort_node, 522
 - STATE_C_COMMENT, 520
 - STATE_CHARACTER, 520
 - STATE_CXX_COMMENT, 520
 - STATE_IDENTIFIER, 520
 - STATE_NONE, 521
 - STATE_PREPROCESSOR, 521
 - STATE_STRING, 521
 - update_comment, 522
 - write_documentation, 522
 - write_element, 522
 - write_string, 522
 - ws_cb, 523
- mxmmlElementGetAttr
 - mxmml-attr.c, 487
 - mxmml.h, 514
- mxmmlElementSetAttr
 - mxmml-attr.c, 487
 - mxmml.h, 514
- mxmmlEntityAddCallback
 - mxmml-entity.c, 488
 - mxmml.h, 515
- mxmmlEntityGetName
 - mxmml-entity.c, 488
 - mxmml.h, 515
- mxmmlEntityGetValue
 - mxmml-entity.c, 488
 - mxmml.h, 515
- mxmmlEntityRemoveCallback
 - mxmml-entity.c, 488
 - mxmml.h, 515
- mxmmlFindElement
 - mxmml-search.c, 504
 - mxmml.h, 515
- mxmmlIndexDelete
 - mxmml-index.c, 497
 - mxmml.h, 515
- mxmmlIndexEnum
 - mxmml-index.c, 498
 - mxmml.h, 515
- mxmmlIndexFind
 - mxmml-index.c, 498
 - mxmml.h, 516
- mxmmlIndexNew
 - mxmml-index.c, 498
 - mxmml.h, 516
- mxmmlIndexReset
 - mxmml-index.c, 498
- mxmml.h, 516
- mxmmlLoadFd
 - mxmml-file.c, 494
 - mxmml.h, 516
- mxmmlLoadFile
 - mxmml-file.c, 494
 - mxmml.h, 516
- mxmmlLoadString
 - mxmml-file.c, 494
 - mxmml.h, 516
- mxmmlNewCustom
 - mxmml-node.c, 500
 - mxmml.h, 516
- mxmmlNewElement
 - mxmml-node.c, 500
 - mxmml.h, 516
- mxmmlNewInteger
 - mxmml-node.c, 500
 - mxmml.h, 517
- mxmmlNewOpaque
 - mxmml-node.c, 500
 - mxmml.h, 517
- mxmmlNewReal
 - mxmml-node.c, 500
 - mxmml.h, 517
- mxmmlNewText
 - mxmml-node.c, 501
 - mxmml.h, 517
- mxmmlNewTextf
 - mxmml-node.c, 501
 - mxmml.h, 517
- mxmmlRemove
 - mxmml-node.c, 501
 - mxmml.h, 518
- mxmmlSaveAllocString
 - mxmml-file.c, 495
 - mxmml.h, 518
- mxmmlSaveFd
 - mxmml-file.c, 495
 - mxmml.h, 518
- mxmmlSaveFile
 - mxmml-file.c, 495
 - mxmml.h, 518
- mxmmlSaveString
 - mxmml-file.c, 495
 - mxmml.h, 518
- mxmmlSetCustom
 - mxmml-set.c, 505
 - mxmml.h, 518
- mxmmlSetCustomHandlers
 - mxmml-file.c, 495
 - mxmml.h, 518
- mxmmlSetElement
 - mxmml-set.c, 505

- mxml.h, 518
- mxmlSetErrorCallback
 - mxml-file.c, 495
 - mxml.h, 518
- mxmlSetInteger
 - mxml-set.c, 505
 - mxml.h, 518
- mxmlSetOpaque
 - mxml-set.c, 505
 - mxml.h, 519
- mxmlSetReal
 - mxml-set.c, 505
 - mxml.h, 519
- mxmlSetText
 - mxml-set.c, 505
 - mxml.h, 519
- mxmlSetTextf
 - mxml-set.c, 506
 - mxml.h, 519
- mxmlWalkNext
 - mxml-search.c, 504
 - mxml.h, 519
- mxmlWalkPrev
 - mxml-search.c, 504
 - mxml.h, 519
- my_preferred_ciphers
 - ssl_server.c, 814
- myrand
 - benchmark.c, 817
- N
 - rsa_context, 174
- n
 - mpi, 157
- N_UNDRSTD
 - message.h, 389
- name
 - agent_s, 38
 - fipa_agent_identifier_s, 66
 - interpreter_variable_data_s, 87
 - LibMC::MCAgent, 145
 - mxml_attr_s, 163
 - mxml_value_s, 171
- nb_zero
 - ssl_context, 188
- net.c
 - _CRT_SECURE_NO_DEPRECATED, 731
 - net_accept, 731
 - net_bind, 731
 - net_close, 731
 - net_connect, 731
 - net_is_blocking, 732
 - net_recv, 732
 - net_send, 732
- net_set_block, 732
- net_set_nonblock, 732
- net_usleep, 733
- net.h
 - ERR_NET_ACCEPT_FAILED, 593
 - ERR_NET_BIND_FAILED, 593
 - ERR_NET_CONN_RESET, 593
 - ERR_NET_CONNECT_FAILED, 593
 - ERR_NET_LISTEN_FAILED, 593
 - ERR_NET_RECV_FAILED, 593
 - ERR_NET_SEND_FAILED, 593
 - ERR_NET_SOCKET_FAILED, 593
 - ERR_NET_UNKNOWN_HOST, 593
 - ERR_NET_WOULD_BLOCK, 594
 - net_accept, 594
 - net_bind, 594
 - net_close, 594
 - net_connect, 594
 - net_recv, 595
 - net_send, 595
 - net_set_nonblock, 595
 - net_usleep, 595
- net_accept
 - net.c, 731
 - net.h, 594
- net_bind
 - net.c, 731
 - net.h, 594
- net_close
 - net.c, 731
 - net.h, 594
- net_connect
 - net.c, 731
 - net.h, 594
- net_is_blocking
 - net.c, 732
- net_recv
 - net.c, 732
 - net.h, 595
- net_send
 - net.c, 732
 - net.h, 595
- net_set_block
 - net.c, 732
- net_set_nonblock
 - net.c, 732
 - net.h, 595
- net_usleep
 - net.c, 733
 - net.h, 595
- New
 - LibMC::MCAclMessage, 100
- next
 - _x509_cert, 25

- [_x509_name](#), 27
 - [listNode_s](#), 90
 - [mxml_node_s](#), 168
- NO_CONTENT
 - [mtp_http.h](#), 393
- node
 - [df.h](#), 290
- node_data
 - [listNode_s](#), 90
- nodes
 - [mxml_index_s](#), 166
- NON_AUTHORITATIVE_INFORMATION
 - [mtp_http.h](#), 393
- NoState
 - [LibMC::MCAgency](#), 109
- NOT_ACCEPTABLE
 - [mtp_http.h](#), 393
- NOT_FOUND
 - [mtp_http.h](#), 393
- nr
 - [aes_context](#), 30
- num
 - [fipa_agent_identifier_set_s](#), 67
 - [fipa_url_sequence_s](#), 76
- NUM_MESSAGE_TYPE
 - [message.h](#), 389
- num_args
 - [command_s](#), 50
- num_attrs
 - [mxml_value_s](#), 171
- num_callbacks
 - [mxml-entity.c](#), 489
- num_entries
 - [df.h](#), 290
- num_nodes
 - [mxml_index_s](#), 166
- num_params
 - [fipa_acl_envelope_s](#), 60
- num_readers
 - [mc_rwlock_s](#), 96
- num_registered
 - [barrier_node_s](#), 47
- num_saved_variables
 - [agent_task_s](#), 44
- num_waiting
 - [barrier_node_s](#), 47
- number
 - [fipa_expression_s::content_u](#), 71
- number_of_elements
 - [agent_task_s](#), 43
- number_of_tasks
 - [agent_datastate_s](#), 35
- numTasks
 - [LibMC::MCAgent](#), 145
- O_BINARY
 - [testmxml.c](#), 528
- offset
 - [havege_state](#), 83
- oid
 - [_x509_name](#), 27
- OID_PKCS1
 - [x509.h](#), 653
- OID_PKCS1_RSA
 - [x509.h](#), 653
- OID_PKCS9
 - [x509.h](#), 653
- OID_X520
 - [x509.h](#), 653
- OK
 - [mtp_http.h](#), 393
- ONE_ITERATION
 - [havege.c](#), 710
- ontology
 - [fipa_acl_message_s](#), 62
- opad
 - [md2_context](#), 148
 - [md4_context](#), 150
 - [md5_context](#), 152
 - [sha1_context](#), 179
 - [sha2_context](#), 181
 - [sha4_context](#), 183
- opaque
 - [hr_time](#), 86
 - [mxml_value_u](#), 172
- operator IntPtr
 - [LibMC::MCAgent](#), 141
- operator MCAgent
 - [LibMC::MCAgent](#), 141
- options
 - [LibMC::MCAgency](#), 129
- orphan
 - [agent_s](#), 39
- out_ctr
 - [ssl_context](#), 188
- out_hdr
 - [ssl_context](#), 188
- out_left
 - [ssl_context](#), 189
- out_msg
 - [ssl_context](#), 188
- out_msglen
 - [ssl_context](#), 189
- out_msgtype
 - [ssl_context](#), 189
- out_uoff
 - [ssl_context](#), 189
- own_cert
 - [ssl_context](#), 190

- own_key
 - ssl_context, 189
- owner
 - agent_s, 39
- P
 - dhm_context, 55
 - md4.c, 719
 - md5.c, 725
 - rsa_context, 175
 - sha1.c, 742
 - sha2.c, 749
 - sha4.c, 756
- p
 - _x509_buf, 22
 - mpi, 157
- PACKAGE_STRING
 - winconfig.h, 826
- PACKAGE_VERSION
 - winconfig.h, 826
- params
 - fipa_acl_envelope_s, 60
- parent
 - mxml_node_s, 168
- parse
 - fipa_message_string_s, 72
- PARTIAL_CONTENT
 - mtp_http.h, 393
- payload_encoding
 - fipa_acl_Param_s, 65
- payload_length
 - fipa_acl_Param_s, 64
- PAYMENT_REQUIRED
 - mtp_http.h, 393
- peer_cert
 - ssl_context, 190
- peer_cn
 - ssl_context, 190
- performative
 - fipa_acl_message_s, 61
- persistent
 - agent_datastate_s, 35
 - agent_task_s, 43
- PI_SUBST
 - md2.c, 717
- pk_oid
 - _x509_cert, 24
- PKCS9_EMAIL
 - x509.h, 653
- PLAINTEXT
 - dh_server.c, 803
- pmslen
 - ssl_context, 192
- pool
 - havege_state, 83
- Port
 - LibMC::MCAgency, 130
- port
 - acc.h, 248
 - host_id_s, 85
 - LibMC::MCAgency, 130
 - mc_platform_s, 92
- portno
 - agency_s, 31
 - LibMC::MCAgency::MCAgency_t, 133
- portnoc
 - agency_s, 32
 - LibMC::MCAgency::MCAgency_t, 133
- PRECONDITION_FAILED
 - mtp_http.h, 393
- premaster
 - ssl_context, 192
- prev
 - mxml_node_s, 168
- print_connectlist
 - commands.x.h, 279
- PrintAgentCode
 - LibMC::MCAgent, 142
- printlist_agents
 - commands.x.h, 280
- process_command
 - cmd_prompt.c, 217
 - cmd_prompt.h, 276
- PROCESSING
 - mtp_http.h, 393
- protocol
 - fipa_acl_message_s, 62
- PROXY_AUTHENTICATION_REQUIRED
 - mtp_http.h, 393
- PT1
 - havege_state, 83
- PT2
 - havege_state, 83
- PTHREAD_STACK_SIZE
 - macros.h, 378
- PUT_UINT32_BE
 - aes.c, 660
 - des.c, 691
 - sha1.c, 743
 - sha2.c, 749
- PUT_UINT32_LE
 - md4.c, 719
 - md5.c, 725
- PUT_UINT64_BE
 - sha4.c, 756
- Q
 - rsa_context, 175

- QP
 - rsa_context, 176
- QUER_IF
 - message.h, 389
- QUER_REF
 - message.h, 389
- QUEUE_TEMPLATE
 - list.h, 463
- quit
 - mc_platform_s, 95
- quit_lock
 - mc_platform_s, 95
- R
 - sha1.c, 743
 - sha2.c, 750
- randbytes
 - ssl_context, 192
- raw
 - _x509_cert, 23
- RCON
 - aes.c, 664
- read_fd
 - ssl_context, 187
- real
 - mxml_value_u, 172
- received
 - fipa_acl_Param_s, 65
- received_by
 - fipa_acl_envelope_Received_s, 58
- received_date
 - fipa_acl_envelope_Received_s, 58
- received_from
 - fipa_acl_envelope_Received_s, 58
- received_id
 - fipa_acl_envelope_Received_s, 58
- received_via
 - fipa_acl_envelope_Received_s, 58
- receiver
 - fipa_acl_message_s, 61
- receiver_num
 - fipa_acl_message_s, 61
- register
 - df_request.x.h, 292
- RegisterService
 - LibMC::MCAgency, 121
- RELAY
 - message.h, 389
- remote_hostname
 - connection_s, 51
- Remove
 - data_structures.h, 285
- RemoveID
 - data_structures.h, 285
- RemoveName
 - data_structures.h, 285
- Reply
 - LibMC::MCAclMessage, 100
- reply_by
 - fipa_acl_message_s, 63
- reply_to
 - fipa_acl_message_s, 62
- reply_with
 - fipa_acl_message_s, 63
- REQUEST
 - df.c, 221
 - df.h, 287
 - df_request.x.h, 292
 - message.h, 389
- REQUEST_ENCRYPTION_INITIALIZE
 - message.h, 389
- REQUEST_TIMEOUT
 - mtp_http.h, 393
- REQUEST_URI_TOO_LONG
 - mtp_http.h, 393
- request_handler_DEREGISTER
 - df.c, 224
- request_handler_REGISTER
 - df.c, 224
- request_handler_SEARCH
 - df.c, 224
- request_handler_SUBSCRIBE
 - df.c, 224
- request_list
 - df.h, 290
- REQUESTED_RANGE_NOT_SATISFIABLE
 - mtp_http.h, 393
- REQUEST_ENTITY_TOO_LARGE
 - mtp_http.h, 393
- RESET_CONTENT
 - mtp_http.h, 393
- ResetSignal
 - LibMC::MCAgency, 118
- resolvers
 - fipa_agent_identifier_s, 66
- response_code
 - mtp_http_s, 162
- response_string
 - mtp_http_s, 162
- ResumeAgency
 - LibMC::MCAgency, 113
- resumed
 - ssl_context, 191
- retain_order
 - fipa_agent_identifier_set_s, 67
- RetrieveAgent
 - LibMC::MCAgency, 123
- RetrieveAgentCode

- LibMC::MCAgent, 142
- RETRY_WITH
 - mtp_http.h, 394
- RETURN_MSG
 - message.h, 389
- return_code
 - mtp_http_s, 161
- return_data
 - agent_datastate_s, 35
 - agent_s, 40
- RHs
 - des.c, 696
- RN
 - rsa_context, 176
- rng_d
 - ssl_context, 192
- rng_f
 - ssl_context, 192
- ROTR
 - sha2.c, 750
 - sha4.c, 757
- ROTR8
 - aes.c, 660
- RP
 - dhm_context, 56
 - rsa_context, 176
- RQ
 - rsa_context, 176
- rsa
 - _x509_cert, 24
- rsa.c
 - _CRT_SECURE_NO_DEPRECATED, 735
 - rsa_check_privkey, 735
 - rsa_check_pubkey, 735
 - rsa_free, 735
 - rsa_gen_key, 736
 - rsa_pkcs1_decrypt, 736
 - rsa_pkcs1_encrypt, 736
 - rsa_pkcs1_sign, 737
 - rsa_pkcs1_verify, 737
 - rsa_private, 738
 - rsa_public, 738
 - rsa_read_private, 739
 - rsa_read_public, 739
 - rsa_self_test, 739
 - rsa_write_private, 740
 - rsa_write_public, 740
- rsa.h
 - ASN1_HASH_MD5, 599
 - ASN1_HASH_SHA1, 599
 - ERR_RSA_BAD_INPUT_DATA, 599
 - ERR_RSA_INVALID_PADDING, 599
 - ERR_RSA_KEY_CHK_FAILED, 599
 - ERR_RSA_KEY_GEN_FAILED, 599
 - ERR_RSA_KEY_RD_FAILED, 599
 - ERR_RSA_KEY_WR_FAILED, 599
 - ERR_RSA_PRIVATE_FAILED, 600
 - ERR_RSA_PUBLIC_FAILED, 600
 - ERR_RSA_VERIFY_FAILED, 600
 - KEY_LEN, 600
 - rsa_check_privkey, 603
 - rsa_check_pubkey, 603
 - RSA_CRYPT, 600
 - RSA_D, 600
 - RSA_DP, 600
 - RSA_DQ, 601
 - RSA_E, 601
 - rsa_free, 603
 - rsa_gen_key, 603
 - RSA_MD2, 601
 - RSA_MD4, 601
 - RSA_MD5, 601
 - RSA_N, 601
 - RSA_P, 602
 - rsa_pkcs1_decrypt, 604
 - rsa_pkcs1_encrypt, 604
 - rsa_pkcs1_sign, 605
 - rsa_pkcs1_verify, 605
 - rsa_private, 606
 - rsa_public, 606
 - RSA_Q, 602
 - RSA_QP, 602
 - RSA_RAW, 602
 - rsa_read_private, 606
 - rsa_read_public, 607
 - rsa_self_test, 607
 - RSA_SHA1, 602
 - RSA_SIGN, 603
 - rsa_write_private, 607
 - rsa_write_public, 608
- rsa_check_privkey
 - dsa.c, 706
 - rsa.c, 735
 - rsa.h, 603
- rsa_check_pubkey
 - dsa.c, 706
 - rsa.c, 735
 - rsa.h, 603
- rsa_context, 174
 - D, 175
 - DP, 175
 - DQ, 175
 - E, 175
 - len, 174
 - N, 174
 - P, 175
 - Q, 175
 - QP, 176

- RN, 176
- RP, 176
- RQ, 176
- ver, 174
- RSA_CRYPT
 - rsa.h, 600
- RSA_D
 - rsa.h, 600
- RSA_DP
 - rsa.h, 600
- RSA_DQ
 - rsa.h, 601
- RSA_E
 - rsa.h, 601
- rsa_free
 - dsa.c, 706
 - rsa.c, 735
 - rsa.h, 603
- rsa_gen_key
 - dsa.c, 706
 - rsa.c, 736
 - rsa.h, 603
- rsa_genkey.c
 - _CRT_SECURE_NO_DEPRECATED, 806
 - EXPONENT, 806
 - KEY_SIZE, 806
 - main, 806
- RSA_MD2
 - rsa.h, 601
- RSA_MD4
 - rsa.h, 601
- RSA_MD5
 - rsa.h, 601
- RSA_N
 - rsa.h, 601
- RSA_P
 - rsa.h, 602
- rsa_pkcs1_decrypt
 - dsa.c, 707
 - rsa.c, 736
 - rsa.h, 604
- rsa_pkcs1_encrypt
 - dsa.c, 707
 - rsa.c, 736
 - rsa.h, 604
- rsa_pkcs1_sign
 - dsa.c, 707
 - rsa.c, 737
 - rsa.h, 605
- rsa_pkcs1_verify
 - dsa.c, 708
 - rsa.c, 737
 - rsa.h, 605
- rsa_private
 - dsa.c, 708
 - rsa.c, 738
 - rsa.h, 606
- rsa_public
 - dsa.c, 709
 - rsa.c, 738
 - rsa.h, 606
- RSA_Q
 - rsa.h, 602
- RSA_QP
 - rsa.h, 602
- RSA_RAW
 - rsa.h, 602
- rsa_read_private
 - rsa.c, 739
 - rsa.h, 606
- rsa_read_public
 - rsa.c, 739
 - rsa.h, 607
- rsa_self_test
 - dsa.c, 709
 - rsa.c, 739
 - rsa.h, 607
- RSA_SHA1
 - rsa.h, 602
- RSA_SIGN
 - rsa.h, 603
- rsa_sign.c
 - _CRT_SECURE_NO_DEPRECATED, 807
 - main, 807
- rsa_verify.c
 - _CRT_SECURE_NO_DEPRECATED, 808
 - main, 808
- rsa_write_private
 - rsa.c, 740
 - rsa.h, 607
- rsa_write_public
 - rsa.c, 740
 - rsa.h, 608
- RSb
 - aes.c, 664
- RT0
 - aes.c, 664
- RT1
 - aes.c, 664
- RT2
 - aes.c, 664
- RT3
 - aes.c, 664
- run_lock
 - agent_s, 40
- Running
 - LibMC::MCAgency, 109
- RWLOCK_DESTROY

- macros.h, [378](#)
- RWLOCK_INIT
 - macros.h, [378](#)
- RWLOCK_RDLOCK
 - macros.h, [378](#)
- RWLOCK_RDUNLOCK
 - macros.h, [378](#)
- RWLOCK_T
 - macros.h, [378](#)
- RWLOCK_WRLOCK
 - macros.h, [379](#)
- RWLOCK_WRUNLOCK
 - macros.h, [379](#)
- S
 - md4.c, [720](#)
 - md5.c, [726](#)
 - sha1.c, [743](#)
- s
 - mpi, [157](#)
- S0
 - sha2.c, [750](#)
 - sha4.c, [757](#)
- S1
 - sha2.c, [750](#)
 - sha4.c, [757](#)
- S2
 - sha2.c, [750](#)
 - sha4.c, [757](#)
- S3
 - sha2.c, [750](#)
 - sha4.c, [757](#)
- SAFE_FREE
 - mtp_http.c, [480](#)
- safe_strcpy
 - mxmldoc.c, [521](#)
- saved_variables
 - agent_task_s, [44](#)
- SB1
 - des.c, [696](#)
- SB2
 - des.c, [697](#)
- SB3
 - des.c, [697](#)
- SB4
 - des.c, [698](#)
- SB5
 - des.c, [698](#)
- SB6
 - des.c, [698](#)
- SB7
 - des.c, [699](#)
- SB8
 - des.c, [699](#)
- scan_file
 - mxmldoc.c, [521](#)
- scanf_argv
 - aescript2.c, [792](#)
- Search
 - data_structures.h, [285](#)
- search
 - df_request.x.h, [292](#)
- search_results
 - df.h, [291](#)
- search_string
 - df.h, [291](#)
- SearchForService
 - LibMC::MCAgency, [121](#)
- SearchName
 - data_structures.h, [285](#)
- sec
 - _x509_time, [28](#)
- second
 - fipa_DateTime_s, [68](#)
- selftest.c
 - _CRT_SECURE_NO_DEPRECATED, [818](#)
 - main, [818](#)
- sem
 - syncListNode_s, [197](#)
- SEMAPHORE_DESTROY
 - macros.h, [379](#)
- SEMAPHORE_INIT
 - macros.h, [379](#)
- SEMAPHORE_POST
 - macros.h, [379](#)
- SEMAPHORE_T
 - macros.h, [379](#)
- SEMAPHORE_WAIT
 - macros.h, [379](#)
- SemaphorePost
 - LibMC::MCAgency, [117](#)
- SemaphoreWait
 - LibMC::MCAgency, [118](#)
- send
 - commands.x.h, [280](#)
- SendAgentMigrationMessage
 - LibMC::MCAgency, [115](#)
- SendAgentMigrationMessageFile
 - LibMC::MCAgency, [114](#)
- sender
 - fipa_acl_message_s, [61](#)
- serial
 - _x509_cert, [23](#)
- server
 - agency_s, [31](#)
 - LibMC::MCAgency::MCAgency_t, [133](#)
 - mtp_http_s, [161](#)
- server_fd

- acc.h, [248](#)
- SERVER_NAME
 - dh_client.c, [800](#)
 - ssl_client1.c, [809](#)
 - ssl_client2.c, [811](#)
- server_name
 - agent_task_s, [44](#)
- SERVER_PORT
 - dh_client.c, [800](#)
 - dh_server.c, [803](#)
 - ssl_client1.c, [809](#)
 - ssl_client2.c, [811](#)
- serverfd
 - connection_s, [51](#)
- service_list
 - df.h, [291](#)
- service_types_e
 - df.h, [287](#)
- ssid
 - ssl_context, [191](#)
- session_table
 - ssl_server.c, [814](#)
- set_alarm
 - timing.c, [780](#)
 - timing.h, [643](#)
- set_bar
 - foo_c, [79](#)
 - foo_s, [81](#)
- set_foo
 - foo_c, [79](#)
 - foo_s, [82](#)
- set_foobar
 - foo_c, [79](#)
- SET_THREAD_OFF
 - macros.h, [379](#)
- SET_THREAD_ON
 - macros.h, [379](#)
- set_timer
 - timing.c, [780](#)
 - timing.h, [644](#)
- SetContent
 - LibMC::MCAclMessage, [102](#)
- SetDefaultAgentStatus
 - LibMC::MCAgency, [114](#)
- SetPerformative
 - LibMC::MCAclMessage, [100](#)
- SetSender
 - LibMC::MCAclMessage, [101](#)
- SetThreadOff
 - LibMC::MCAgency, [113](#)
- SetThreadOn
 - LibMC::MCAgency, [112](#)
- SetThreadsAllOff
 - LibMC::MCAgency, [112](#)
- SetThreadsAllOn
 - LibMC::MCAgency, [112](#)
- SettingChangingEventHandler
 - LibMC::Properties::Settings, [177](#)
- Settings
 - LibMC::Properties::Settings, [177](#)
- SettingsSavingEventHandler
 - LibMC::Properties::Settings, [177](#)
- sha1
 - sha1.c, [743](#)
 - sha1.h, [610](#)
- sha1.c
 - _CRT_SECURE_NO_DEPRECATED, [742](#)
 - _sha1_src, [746](#)
 - F, [742](#)
 - GET_UINT32_BE, [742](#)
 - K, [742](#)
 - P, [742](#)
 - PUT_UINT32_BE, [743](#)
 - R, [743](#)
 - S, [743](#)
 - sha1, [743](#)
 - sha1_file, [743](#)
 - sha1_finish, [744](#)
 - sha1_hmac, [744](#)
 - sha1_hmac_finish, [744](#)
 - sha1_hmac_starts, [745](#)
 - sha1_hmac_update, [745](#)
 - sha1_padding, [746](#)
 - sha1_process, [745](#)
 - sha1_self_test, [745](#)
 - sha1_starts, [746](#)
 - sha1_update, [746](#)
- sha1.h
 - sha1, [610](#)
 - sha1_file, [610](#)
 - sha1_finish, [610](#)
 - sha1_hmac, [610](#)
 - sha1_hmac_finish, [611](#)
 - sha1_hmac_starts, [611](#)
 - sha1_hmac_update, [611](#)
 - sha1_self_test, [612](#)
 - sha1_starts, [612](#)
 - sha1_update, [612](#)
- sha1_check
 - sha1sum.c, [796](#)
- sha1_context, [179](#)
 - buffer, [179](#)
 - ipad, [179](#)
 - opad, [179](#)
 - state, [179](#)
 - total, [179](#)
- sha1_file
 - sha1.c, [743](#)

- sha1.h, 610
- sha1_finish
 - sha1.c, 744
 - sha1.h, 610
- sha1_hmac
 - sha1.c, 744
 - sha1.h, 610
- sha1_hmac_finish
 - sha1.c, 744
 - sha1.h, 611
- sha1_hmac_starts
 - sha1.c, 745
 - sha1.h, 611
- sha1_hmac_update
 - sha1.c, 745
 - sha1.h, 611
- sha1_padding
 - sha1.c, 746
- sha1_print
 - sha1sum.c, 796
- sha1_process
 - sha1.c, 745
- sha1_self_test
 - sha1.c, 745
 - sha1.h, 612
- sha1_starts
 - sha1.c, 746
 - sha1.h, 612
- sha1_update
 - sha1.c, 746
 - sha1.h, 612
- sha1_wrapper
 - sha1sum.c, 796
- sha1sum.c
 - _CRT_SECURE_NO_DEPRECATED, 796
 - main, 796
 - sha1_check, 796
 - sha1_print, 796
 - sha1_wrapper, 796
- sha2
 - sha2.c, 750
 - sha2.h, 615
- sha2.c
 - _CRT_SECURE_NO_DEPRECATED, 749
 - _sha2_src, 754
 - F0, 749
 - F1, 749
 - GET_UINT32_BE, 749
 - P, 749
 - PUT_UINT32_BE, 749
 - R, 750
 - ROTR, 750
 - S0, 750
 - S1, 750
 - S2, 750
 - S3, 750
 - sha2, 750
 - sha2_file, 751
 - sha2_finish, 751
 - sha2_hmac, 751
 - sha2_hmac_finish, 752
 - sha2_hmac_starts, 752
 - sha2_hmac_update, 752
 - sha2_padding, 754
 - sha2_process, 752
 - sha2_self_test, 753
 - sha2_starts, 753
 - sha2_update, 753
 - SHR, 750
- sha2.h
 - sha2, 615
 - sha2_file, 615
 - sha2_finish, 615
 - sha2_hmac, 615
 - sha2_hmac_finish, 616
 - sha2_hmac_starts, 616
 - sha2_hmac_update, 616
 - sha2_self_test, 617
 - sha2_starts, 617
 - sha2_update, 617
- sha2_check
 - sha2sum.c, 798
- sha2_context, 181
 - buffer, 181
 - ipad, 181
 - is224, 182
 - opad, 181
 - state, 181
 - total, 181
- sha2_file
 - sha2.c, 751
 - sha2.h, 615
- sha2_finish
 - sha2.c, 751
 - sha2.h, 615
- sha2_hmac
 - sha2.c, 751
 - sha2.h, 615
- sha2_hmac_finish
 - sha2.c, 752
 - sha2.h, 616
- sha2_hmac_starts
 - sha2.c, 752
 - sha2.h, 616
- sha2_hmac_update
 - sha2.c, 752
 - sha2.h, 616
- sha2_padding

- sha2.c, 754
- sha2_print
 - sha2sum.c, 798
- sha2_process
 - sha2.c, 752
- sha2_self_test
 - sha2.c, 753
 - sha2.h, 617
- sha2_starts
 - sha2.c, 753
 - sha2.h, 617
- sha2_update
 - sha2.c, 753
 - sha2.h, 617
- sha2_wrapper
 - sha2sum.c, 798
- sha2sum.c
 - _CRT_SECURE_NO_DEPRECATED, 798
 - main, 798
 - sha2_check, 798
 - sha2_print, 798
 - sha2_wrapper, 798
- sha4
 - sha4.c, 757
 - sha4.h, 620
- sha4.c
 - _CRT_SECURE_NO_DEPRECATED, 756
 - _sha4_src, 760
 - F0, 756
 - F1, 756
 - GET_UINT64_BE, 756
 - K, 760
 - P, 756
 - PUT_UINT64_BE, 756
 - ROTR, 757
 - S0, 757
 - S1, 757
 - S2, 757
 - S3, 757
 - sha4, 757
 - sha4_file, 757
 - sha4_finish, 758
 - sha4_hmac, 758
 - sha4_hmac_finish, 758
 - sha4_hmac_starts, 759
 - sha4_hmac_update, 759
 - sha4_padding, 760
 - sha4_process, 759
 - sha4_self_test, 759
 - sha4_starts, 760
 - sha4_update, 760
 - SHR, 757
- sha4.h
 - sha4, 620
- sha4_file, 620
- sha4_finish, 620
- sha4_hmac, 621
- sha4_hmac_finish, 621
- sha4_hmac_starts, 621
- sha4_hmac_update, 622
- sha4_self_test, 622
- sha4_starts, 622
- sha4_update, 622
- uint64, 620
- UL64, 620
- sha4_context, 183
 - buffer, 183
 - ipad, 183
 - is384, 184
 - opad, 183
 - state, 183
 - total, 183
- sha4_file
 - sha4.c, 757
 - sha4.h, 620
- sha4_finish
 - sha4.c, 758
 - sha4.h, 620
- sha4_hmac
 - sha4.c, 758
 - sha4.h, 621
- sha4_hmac_finish
 - sha4.c, 758
 - sha4.h, 621
- sha4_hmac_starts
 - sha4.c, 759
 - sha4.h, 621
- sha4_hmac_update
 - sha4.c, 759
 - sha4.h, 622
- sha4_padding
 - sha4.c, 760
- sha4_process
 - sha4.c, 759
- sha4_self_test
 - sha4.c, 759
 - sha4.h, 622
- sha4_starts
 - sha4.c, 760
 - sha4.h, 622
- sha4_update
 - sha4.c, 760
 - sha4.h, 622
- shelltype
 - LibMC::MCAgency::ChOptions_t, 132
- SHR
 - sha2.c, 750
 - sha4.c, 757

- sidlen
 - ssl_context, [191](#)
- sidtable
 - ssl_context, [191](#)
- sig
 - _x509_cert, [25](#)
- sig_oid1
 - _x509_cert, [24](#)
- sig_oid2
 - _x509_cert, [25](#)
- sighandler
 - timing.c, [780](#)
- sign
 - fipa_DateTime_s, [69](#)
- SIGNAL
 - macros.h, [380](#)
- signalled
 - syncListNode_s, [197](#)
- size
 - barrier_queue_s, [48](#)
 - df.h, [291](#)
 - dynstring_s, [57](#)
 - interpreter_variable_data_s, [87](#)
 - list_s, [89](#)
 - syncList_s, [196](#)
- size_of_element_array
 - agent_task_s, [43](#)
- SLEEP_QUEUE
 - macros.h, [380](#)
- SLEEP_RESET
 - macros.h, [380](#)
- small_prime
 - bignum.c, [684](#)
- SOCKET_INPUT_SIZE
 - message.c, [476](#)
 - mtp_http.h, [392](#)
- SOME
 - df.h, [287](#)
- sort_node
 - mxmldoc.c, [522](#)
- split_string
 - cmd_prompt.c, [217](#)
 - cmd_prompt.h, [276](#)
- ssl.h
 - ERR_SSL_BAD_HS_CERTIFICATE, [627](#)
 - ERR_SSL_BAD_HS_CERTIFICATE_-
REQUEST, [627](#)
 - ERR_SSL_BAD_HS_CERTIFICATE_-
VERIFY, [627](#)
 - ERR_SSL_BAD_HS_CHANGE_CIPHER_-
SPEC, [627](#)
 - ERR_SSL_BAD_HS_CLIENT_HELLO, [627](#)
 - ERR_SSL_BAD_HS_CLIENT_KEY_-
EXCHANGE, [628](#)
 - ERR_SSL_BAD_HS_FINISHED, [628](#)
 - ERR_SSL_BAD_HS_SERVER_HELLO, [628](#)
 - ERR_SSL_BAD_HS_SERVER_HELLO_-
DONE, [628](#)
 - ERR_SSL_BAD_HS_SERVER_KEY_-
EXCHANGE, [628](#)
 - ERR_SSL_CA_CHAIN_REQUIRED, [628](#)
 - ERR_SSL_CERTIFICATE_REQUIRED, [628](#)
 - ERR_SSL_CERTIFICATE_TOO_LARGE,
[628](#)
 - ERR_SSL_FATAL_ALERT_MESSAGE, [628](#)
 - ERR_SSL_FEATURE_UNAVAILABLE, [629](#)
 - ERR_SSL_INVALID_MAC, [629](#)
 - ERR_SSL_INVALID_MODULUS_SIZE, [629](#)
 - ERR_SSL_INVALID_RECORD, [629](#)
 - ERR_SSL_NO_CIPHER_CHOSEN, [629](#)
 - ERR_SSL_NO_CLIENT_CERTIFICATE,
[629](#)
 - ERR_SSL_NO_SESSION_FOUND, [629](#)
 - ERR_SSL_PEER_CLOSE_NOTIFY, [629](#)
 - ERR_SSL_PEER_VERIFY_FAILED, [629](#)
 - ERR_SSL_PRIVATE_KEY_REQUIRED, [630](#)
 - ERR_SSL_UNEXPECTED_MESSAGE, [630](#)
 - ERR_SSL_UNKNOWN_CIPHER, [630](#)
 - SSL3_EDH_RSA_DES_168_SHA, [630](#)
 - SSL3_RSA_DES_168_SHA, [630](#)
 - SSL3_RSA_RC4_128_MD5, [630](#)
 - SSL3_RSA_RC4_128_SHA, [630](#)
 - SSL_CERTIFICATE_REQUEST, [635](#)
 - SSL_CERTIFICATE_VERIFY, [635](#)
 - SSL_CLIENT_CERTIFICATE, [635](#)
 - SSL_CLIENT_CHANGE_CIPHER_SPEC,
[635](#)
 - SSL_CLIENT_FINISHED, [635](#)
 - SSL_CLIENT_HELLO, [634](#)
 - SSL_CLIENT_KEY_EXCHANGE, [635](#)
 - SSL_HANDSHAKE_OVER, [635](#)
 - SSL_HELLO_REQUEST, [634](#)
 - SSL_SERVER_CERTIFICATE, [634](#)
 - SSL_SERVER_CHANGE_CIPHER_SPEC,
[635](#)
 - SSL_SERVER_FINISHED, [635](#)
 - SSL_SERVER_HELLO, [634](#)
 - SSL_SERVER_HELLO_DONE, [635](#)
 - SSL_SERVER_KEY_EXCHANGE, [635](#)
 - SSL_ALERT_CLOSE_NOTIFY, [630](#)
 - SSL_ALERT_FATAL, [630](#)
 - SSL_ALERT_NO_CERTIFICATE, [631](#)
 - SSL_ALERT_WARNING, [631](#)
 - SSL_BUFFER_LEN, [631](#)
 - ssl_calc_verify, [635](#)
 - ssl_client_start, [635](#)
 - ssl_close_notify, [635](#)
 - SSL_COMPRESS_NULL, [631](#)

- ssl_default_ciphers, [641](#)
- ssl_derive_keys, [636](#)
- SSL_EXPIRATION_TIME, [631](#)
- ssl_flush_output, [636](#)
- ssl_free, [636](#)
- ssl_get_cipher_name, [636](#)
- ssl_get_verify_result, [636](#)
- ssl_handshake, [636](#)
- SSL_HS_CERTIFICATE, [631](#)
- SSL_HS_CERTIFICATE_REQUEST, [631](#)
- SSL_HS_CERTIFICATE_VERIFY, [631](#)
- SSL_HS_CLIENT_HELLO, [631](#)
- SSL_HS_CLIENT_KEY_EXCHANGE, [632](#)
- SSL_HS_FINISHED, [632](#)
- SSL_HS_HELLO_REQUEST, [632](#)
- SSL_HS_SERVER_HELLO, [632](#)
- SSL_HS_SERVER_HELLO_DONE, [632](#)
- SSL_HS_SERVER_KEY_EXCHANGE, [632](#)
- ssl_init, [637](#)
- SSL_IS_CLIENT, [632](#)
- SSL_IS_SERVER, [632](#)
- SSL_MAX_CONTENT_LEN, [632](#)
- SSL_MSG_ALERT, [633](#)
- SSL_MSG_APPLICATION_DATA, [633](#)
- SSL_MSG_CHANGE_CIPHER_SPEC, [633](#)
- SSL_MSG_HANDSHAKE, [633](#)
- ssl_parse_certificate, [637](#)
- ssl_parse_change_cipher_spec, [637](#)
- ssl_parse_finished, [637](#)
- ssl_read, [637](#)
- ssl_read_record, [638](#)
- ssl_server_start, [638](#)
- SSL_SESSION_TBL_LEN, [633](#)
- ssl_set_authmode, [638](#)
- ssl_set_ca_chain, [639](#)
- ssl_set_ciphlist, [639](#)
- ssl_set_dhm_vals, [639](#)
- ssl_set_endpoint, [639](#)
- ssl_set_io_files, [639](#)
- ssl_set_rng_func, [639](#)
- ssl_set_rsa_cert, [639](#)
- ssl_set_sidtable, [640](#)
- ssl_states, [634](#)
- SSL_VERIFY_NONE, [633](#)
- SSL_VERIFY_OPTIONAL, [633](#)
- SSL_VERIFY_REQUIRED, [633](#)
- ssl_write, [640](#)
- ssl_write_certificate, [640](#)
- ssl_write_change_cipher_spec, [640](#)
- ssl_write_finished, [640](#)
- ssl_write_record, [641](#)
- SSLV3_MAJOR_VERSION, [634](#)
- SSLV3_MINOR_VERSION, [634](#)
- TLS10_MINOR_VERSION, [634](#)
- TLS11_MINOR_VERSION, [634](#)
- TLS1_EDH_RSA_AES_256_SHA, [634](#)
- TLS1_RSA_AES_256_SHA, [634](#)
- SSL3_EDH_RSA_DES_168_SHA
 - ssl.h, [630](#)
- SSL3_RSA_DES_168_SHA
 - ssl.h, [630](#)
- SSL3_RSA_RC4_128_MD5
 - ssl.h, [630](#)
- SSL3_RSA_RC4_128_SHA
 - ssl.h, [630](#)
- SSL_CERTIFICATE_REQUEST
 - ssl.h, [635](#)
- SSL_CERTIFICATE_VERIFY
 - ssl.h, [635](#)
- SSL_CLIENT_CERTIFICATE
 - ssl.h, [635](#)
- SSL_CLIENT_CHANGE_CIPHER_SPEC
 - ssl.h, [635](#)
- SSL_CLIENT_FINISHED
 - ssl.h, [635](#)
- SSL_CLIENT_HELLO
 - ssl.h, [634](#)
- SSL_CLIENT_KEY_EXCHANGE
 - ssl.h, [635](#)
- SSL_HANDSHAKE_OVER
 - ssl.h, [635](#)
- SSL_HELLO_REQUEST
 - ssl.h, [634](#)
- SSL_SERVER_CERTIFICATE
 - ssl.h, [634](#)
- SSL_SERVER_CHANGE_CIPHER_SPEC
 - ssl.h, [635](#)
- SSL_SERVER_FINISHED
 - ssl.h, [635](#)
- SSL_SERVER_HELLO
 - ssl.h, [634](#)
- SSL_SERVER_HELLO_DONE
 - ssl.h, [635](#)
- SSL_SERVER_KEY_EXCHANGE
 - ssl.h, [635](#)
- SSL_ALERT_CLOSE_NOTIFY
 - ssl.h, [630](#)
- SSL_ALERT_FATAL
 - ssl.h, [630](#)
- SSL_ALERT_NO_CERTIFICATE
 - ssl.h, [631](#)
- SSL_ALERT_WARNING
 - ssl.h, [631](#)
- SSL_BUFFER_LEN
 - ssl.h, [631](#)
- ssl_calc_finished
 - ssl_tls.c, [771](#)
- ssl_calc_verify

- ssl.h, 635
- ssl_tls.c, 771
- ssl_cli.c
 - _CRT_SECURE_NO_DEPRECATED, 762
 - _ssl_cli_src, 764
 - ssl_client_start, 762
 - ssl_parse_certificate_request, 763
 - ssl_parse_server_hello, 763
 - ssl_parse_server_hello_done, 763
 - ssl_parse_server_key_exchange, 763
 - ssl_write_certificate_verify, 763
 - ssl_write_client_hello, 764
 - ssl_write_client_key_exchange, 764
- ssl_client1.c
 - _CRT_SECURE_NO_DEPRECATED, 809
 - GET_REQUEST, 809
 - main, 810
 - SERVER_NAME, 809
 - SERVER_PORT, 809
- ssl_client2.c
 - _CRT_SECURE_NO_DEPRECATED, 811
 - GET_REQUEST, 811
 - main, 812
 - SERVER_NAME, 811
 - SERVER_PORT, 811
- ssl_client_start
 - ssl.h, 635
 - ssl_cli.c, 762
- ssl_close_notify
 - ssl.h, 635
 - ssl_tls.c, 771
- SSL_COMPRESS_NULL
 - ssl.h, 631
- ssl_context, 185
 - authmode, 190
 - ca_chain, 190
 - cipher, 193
 - cipherlist, 193
 - client_auth, 190
 - ctx_dec, 194
 - ctx_enc, 194
 - ctxlen, 193
 - dhm_ctx, 192
 - endpoint, 190
 - hs_md5, 191
 - hs_sha1, 192
 - in_ctr, 187
 - in_hdr, 187
 - in_hhlen, 188
 - in_left, 188
 - in_msg, 187
 - in_msglen, 188
 - in_msgtype, 187
 - in_offt, 187
 - iv_dec, 194
 - iv_enc, 194
 - ivlen, 194
 - keylen, 193
 - mac_dec, 195
 - mac_enc, 194
 - maclen, 194
 - major_ver, 186
 - master, 193
 - max_ver, 186
 - minlen, 193
 - minor_ver, 186
 - nb_zero, 188
 - out_ctr, 188
 - out_hdr, 188
 - out_left, 189
 - out_msg, 188
 - out_msglen, 189
 - out_msgtype, 189
 - out_uoff, 189
 - own_cert, 190
 - own_key, 189
 - peer_cert, 190
 - peer_cn, 190
 - pmslen, 192
 - premaster, 192
 - randbytes, 192
 - read_fd, 187
 - resumed, 191
 - rng_d, 192
 - rng_f, 192
 - sessid, 191
 - sidlen, 191
 - sidtable, 191
 - state, 186
 - verify_result, 191
 - write_fd, 189
- ssl_decrypt_buf
 - ssl_tls.c, 771
- ssl_default_ciphers
 - ssl.h, 641
 - ssl_tls.c, 777
- ssl_derive_keys
 - ssl.h, 636
 - ssl_tls.c, 771
- ssl_encrypt_buf
 - ssl_tls.c, 772
- SSL_EXPIRATION_TIME
 - ssl.h, 631
- ssl_flush_output
 - ssl.h, 636
 - ssl_tls.c, 772
- ssl_free
 - ssl.h, 636

- ssl_tls.c, 772
- ssl_get_cipher_name
 - ssl.h, 636
 - ssl_tls.c, 772
- ssl_get_session
 - ssl_srv.c, 765
- ssl_get_verify_result
 - ssl.h, 636
 - ssl_tls.c, 772
- ssl_handshake
 - ssl.h, 636
 - ssl_tls.c, 773
- SSL_HS_CERTIFICATE
 - ssl.h, 631
- SSL_HS_CERTIFICATE_REQUEST
 - ssl.h, 631
- SSL_HS_CERTIFICATE_VERIFY
 - ssl.h, 631
- SSL_HS_CLIENT_HELLO
 - ssl.h, 631
- SSL_HS_CLIENT_KEY_EXCHANGE
 - ssl.h, 632
- SSL_HS_FINISHED
 - ssl.h, 632
- SSL_HS_HELLO_REQUEST
 - ssl.h, 632
- SSL_HS_SERVER_HELLO
 - ssl.h, 632
- SSL_HS_SERVER_HELLO_DONE
 - ssl.h, 632
- SSL_HS_SERVER_KEY_EXCHANGE
 - ssl.h, 632
- ssl_init
 - ssl.h, 637
 - ssl_tls.c, 773
- SSL_IS_CLIENT
 - ssl.h, 632
- SSL_IS_SERVER
 - ssl.h, 632
- ssl_mac_md5
 - ssl_tls.c, 773
- ssl_mac_shal
 - ssl_tls.c, 773
- SSL_MAX_CONTENT_LEN
 - ssl.h, 632
- SSL_MSG_ALERT
 - ssl.h, 633
- SSL_MSG_APPLICATION_DATA
 - ssl.h, 633
- SSL_MSG_CHANGE_CIPHER_SPEC
 - ssl.h, 633
- SSL_MSG_HANDSHAKE
 - ssl.h, 633
- ssl_parse_certificate
 - ssl.h, 637
 - ssl_tls.c, 773
- ssl_parse_certificate_request
 - ssl_cli.c, 763
- ssl_parse_certificate_verify
 - ssl_srv.c, 765
- ssl_parse_change_cipher_spec
 - ssl.h, 637
 - ssl_tls.c, 774
- ssl_parse_client_hello
 - ssl_srv.c, 766
- ssl_parse_client_key_exchange
 - ssl_srv.c, 766
- ssl_parse_finished
 - ssl.h, 637
 - ssl_tls.c, 774
- ssl_parse_server_hello
 - ssl_cli.c, 763
- ssl_parse_server_hello_done
 - ssl_cli.c, 763
- ssl_parse_server_key_exchange
 - ssl_cli.c, 763
- ssl_read
 - ssl.h, 637
 - ssl_tls.c, 774
- ssl_read_record
 - ssl.h, 638
 - ssl_tls.c, 774
- ssl_server.c
 - _CRT_SECURE_NO_DEPRECATED, 813
 - dhm_G, 814
 - dhm_P, 814
 - HTTP_RESPONSE, 813
 - main, 814
 - my_preferred_ciphers, 814
 - session_table, 814
- ssl_server_start
 - ssl.h, 638
 - ssl_srv.c, 766
- SSL_SESSION_TBL_LEN
 - ssl.h, 633
- ssl_set_authmode
 - ssl.h, 638
 - ssl_tls.c, 774
- ssl_set_ca_chain
 - ssl.h, 639
 - ssl_tls.c, 775
- ssl_set_ciphlist
 - ssl.h, 639
 - ssl_tls.c, 775
- ssl_set_dhm_vals
 - ssl.h, 639
 - ssl_tls.c, 775
- ssl_set_endpoint

- ssl.h, 639
- ssl_tls.c, 775
- ssl_set_io_files
 - ssl.h, 639
 - ssl_tls.c, 775
- ssl_set_rng_func
 - ssl.h, 639
 - ssl_tls.c, 775
- ssl_set_rsa_cert
 - ssl.h, 639
 - ssl_tls.c, 776
- ssl_set_session
 - ssl_srv.c, 766
- ssl_set_sidtable
 - ssl.h, 640
 - ssl_tls.c, 776
- ssl_srv.c
 - _CRT_SECURE_NO_DEPRECATED, 765
 - _ssl_srv_src, 768
 - ssl_get_session, 765
 - ssl_parse_certificate_verify, 765
 - ssl_parse_client_hello, 766
 - ssl_parse_client_key_exchange, 766
 - ssl_server_start, 766
 - ssl_set_session, 766
 - ssl_write_certificate_request, 767
 - ssl_write_server_hello, 767
 - ssl_write_server_hello_done, 767
 - ssl_write_server_key_exchange, 767
- ssl_states
 - ssl.h, 634
- ssl_tls.c
 - _CRT_SECURE_NO_DEPRECATED, 771
 - _ssl_tls_src, 777
 - ssl_calc_finished, 771
 - ssl_calc_verify, 771
 - ssl_close_notify, 771
 - ssl_decrypt_buf, 771
 - ssl_default_ciphers, 777
 - ssl_derive_keys, 771
 - ssl_encrypt_buf, 772
 - ssl_flush_output, 772
 - ssl_free, 772
 - ssl_get_cipher_name, 772
 - ssl_get_verify_result, 772
 - ssl_handshake, 773
 - ssl_init, 773
 - ssl_mac_md5, 773
 - ssl_mac_sha1, 773
 - ssl_parse_certificate, 773
 - ssl_parse_change_cipher_spec, 774
 - ssl_parse_finished, 774
 - ssl_read, 774
 - ssl_read_record, 774
 - ssl_set_authmode, 774
 - ssl_set_ca_chain, 775
 - ssl_set_ciphlist, 775
 - ssl_set_dhm_vals, 775
 - ssl_set_endpoint, 775
 - ssl_set_io_files, 775
 - ssl_set_rng_func, 775
 - ssl_set_rsa_cert, 776
 - ssl_set_sidtable, 776
 - ssl_write, 776
 - ssl_write_certificate, 776
 - ssl_write_change_cipher_spec, 776
 - ssl_write_finished, 777
 - ssl_write_record, 777
 - tls1_prf, 777
- SSL_VERIFY_NONE
 - ssl.h, 633
- SSL_VERIFY_OPTIONAL
 - ssl.h, 633
- SSL_VERIFY_REQUIRED
 - ssl.h, 633
- ssl_write
 - ssl.h, 640
 - ssl_tls.c, 776
- ssl_write_certificate
 - ssl.h, 640
 - ssl_tls.c, 776
- ssl_write_certificate_request
 - ssl_srv.c, 767
- ssl_write_certificate_verify
 - ssl_cli.c, 763
- ssl_write_change_cipher_spec
 - ssl.h, 640
 - ssl_tls.c, 776
- ssl_write_client_hello
 - ssl_cli.c, 764
- ssl_write_client_key_exchange
 - ssl_cli.c, 764
- ssl_write_finished
 - ssl.h, 640
 - ssl_tls.c, 777
- ssl_write_record
 - ssl.h, 641
 - ssl_tls.c, 777
- ssl_write_server_hello
 - ssl_srv.c, 767
- ssl_write_server_hello_done
 - ssl_srv.c, 767
- ssl_write_server_key_exchange
 - ssl_srv.c, 767
- SSLV3_MAJOR_VERSION
 - ssl.h, 634
- SSLV3_MINOR_VERSION
 - ssl.h, 634

- stack_size
 - agency_s, [32](#)
 - LibMC::MCAgency::MCAgency_t, [134](#)
 - LibMC::MCAgency::MCAgencyOptions_t, [135](#)
 - mc_platform_s, [93](#)
 - MCAgencyOptions_s, [136](#)
- start
 - _hr_time, [21](#)
- State
 - LibMC::MCAgency, [131](#)
- state
 - LibMC::MCAgency, [130](#)
 - md2_context, [148](#)
 - md4_context, [150](#)
 - md5_context, [152](#)
 - sha1_context, [179](#)
 - sha2_context, [181](#)
 - sha4_context, [183](#)
 - ssl_context, [186](#)
- STATE_C_COMMENT
 - mxmldoc.c, [520](#)
- STATE_CHARACTER
 - mxmldoc.c, [520](#)
- STATE_CXX_COMMENT
 - mxmldoc.c, [520](#)
- STATE_IDENTIFIER
 - mxmldoc.c, [520](#)
- STATE_NONE
 - mxmldoc.c, [521](#)
- STATE_PREPROCESSOR
 - mxmldoc.c, [521](#)
- STATE_STRING
 - mxmldoc.c, [521](#)
- status
 - LibMC::MCAgent, [146](#)
- Steer
 - LibMC::MCAgency, [120](#)
- SteerControl
 - LibMC::MCAgency, [120](#)
- str
 - fipa_number_s, [73](#)
 - fipa_url_s, [75](#)
- string
 - fipa_expression_s::content_u, [71](#)
 - mxmld_text_s, [170](#)
- strtok_r
 - winconfig.h, [826](#)
- STRUCT
 - acc.h, [247](#)
 - ams.h, [268](#)
 - df.h, [289](#), [290](#)
 - macros.h, [380](#)
 - xml_parser.h, [405](#)
- struct.cxx
 - foo_t, [527](#)
- subject
 - _x509_cert, [24](#)
- subject_id
 - _x509_cert, [25](#)
- subject_raw
 - _x509_cert, [24](#)
- SUBSCRIBE
 - message.h, [389](#)
- SWAP
 - havege.c, [710](#)
- SWITCHING_PROTOCOLS
 - mtp_http.h, [393](#)
- sync_list.c
 - syncListAddNode, [470](#)
 - syncListDelete, [470](#)
 - syncListFind, [470](#)
 - syncListInit, [470](#)
 - syncListNew, [471](#)
 - syncListNodeDestroy, [471](#)
 - syncListNodeInit, [471](#)
 - syncListNodeNew, [471](#)
 - syncListRemove, [471](#)
- sync_list.h
 - syncList_p, [472](#)
 - syncList_t, [472](#)
 - syncListAddNode, [473](#)
 - syncListDelete, [473](#)
 - syncListFind, [473](#)
 - syncListGet, [473](#)
 - syncListInit, [473](#)
 - syncListNew, [473](#)
 - syncListNode_p, [472](#)
 - syncListNode_t, [472](#)
 - syncListNodeDestroy, [473](#)
 - syncListNodeInit, [474](#)
 - syncListNodeNew, [474](#)
 - syncListRemove, [474](#)
- SyncDelete
 - LibMC::MCAgency, [118](#)
- SyncInit
 - LibMC::MCAgency, [119](#)
- syncList
 - mc_platform_s, [93](#)
- syncList_p
 - sync_list.h, [472](#)
- syncList_s, [196](#)
 - giant_lock, [196](#)
 - list, [196](#)
 - lock, [196](#)
 - size, [196](#)
- syncList_t
 - sync_list.h, [472](#)

- syncListAddNode
 - sync_list.c, [470](#)
 - sync_list.h, [473](#)
- syncListDelete
 - sync_list.c, [470](#)
 - sync_list.h, [473](#)
- syncListFind
 - sync_list.c, [470](#)
 - sync_list.h, [473](#)
- syncListGet
 - sync_list.h, [473](#)
- syncListInit
 - sync_list.c, [470](#)
 - sync_list.h, [473](#)
- syncListNew
 - sync_list.c, [471](#)
 - sync_list.h, [473](#)
- syncListNode_p
 - sync_list.h, [472](#)
- syncListNode_s, [197](#)
 - cond, [197](#)
 - id, [197](#)
 - lock, [197](#)
 - sem, [197](#)
 - signalled, [197](#)
- syncListNode_t
 - sync_list.h, [472](#)
- syncListNodeDestroy
 - sync_list.c, [471](#)
 - sync_list.h, [473](#)
- syncListNodeInit
 - sync_list.c, [471](#)
 - sync_list.h, [474](#)
- syncListNodeNew
 - sync_list.c, [471](#)
 - sync_list.h, [474](#)
- syncListRemove
 - sync_list.c, [471](#)
 - sync_list.h, [474](#)
- System, [15](#)
- System::Collections::Generic, [16](#)
- System::Reflection, [17](#)
- System::Runtime::CompilerServices, [18](#)
- System::Runtime::InteropServices, [19](#)
- System::Text, [20](#)
- t_dbl
 - bignum.h, [551](#)
- t_int
 - bignum.h, [551](#)
- tag
 - _x509_buf, [22](#)
- target
 - message_s, [156](#)
- mtp_http_s, [161](#)
- task_completed
 - agent_task_s, [43](#)
- task_progress
 - agent_datastate_s, [35](#)
- tasks
 - agent_datastate_s, [34](#)
- tbs
 - _x509_cert, [23](#)
- TerminateAgent
 - LibMC::MCAgent, [142](#)
- test_ca_crt
 - certs.c, [685](#)
 - certs.h, [565](#)
- test_ca_key
 - certs.c, [685](#)
 - certs.h, [565](#)
- test_ca_pwd
 - certs.c, [685](#)
 - certs.h, [565](#)
- test_cli_crt
 - certs.c, [686](#)
 - certs.h, [565](#)
- test_cli_key
 - certs.c, [686](#)
 - certs.h, [565](#)
- test_srv_crt
 - certs.c, [687](#)
 - certs.h, [565](#)
- test_srv_key
 - certs.c, [687](#)
 - certs.h, [566](#)
- testmxml.c
 - main, [528](#)
 - O_BINARY, [528](#)
 - type_cb, [528](#)
 - whitespace_cb, [529](#)
- text
 - mxml_value_u, [172](#)
- thread
 - cmd_prompt_s, [49](#)
 - df.h, [291](#)
- THREAD_CANCEL
 - macros.h, [380](#)
- THREAD_CREATE
 - macros.h, [380](#)
- THREAD_JOIN
 - macros.h, [381](#)
- THREAD_T
 - macros.h, [381](#)
- threads
 - agency_s, [32](#)
 - LibMC::MCAgency::MCAgency_t, [134](#)

- LibMC::MCAgency::MCAgencyOptions_t, 135
- MCAgencyOptions_s, 136
- timing.c
 - _CRT_SECURE_NO_DEPRECATED, 779
 - _timing_src, 780
 - alarmed, 780
 - hardclock, 780
 - hardclock_init, 780
 - set_alarm, 780
 - set_timer, 780
 - sighandler, 780
 - tv_init, 781
- timing.h
 - alarmed, 644
 - hardclock, 643
 - set_alarm, 643
 - set_timer, 644
- TLS10_MINOR_VERSION
 - ssl.h, 634
- TLS11_MINOR_VERSION
 - ssl.h, 634
- TLS1_EDH_RSA_AES_256_SHA
 - ssl.h, 634
- tls1_prf
 - ssl_tls.c, 777
- TLS1_RSA_AES_256_SHA
 - ssl.h, 634
- to
 - fipa_acl_Param_s, 64
- to_address
 - message_s, 156
- ToString
 - LibMC::MCAgent, 141
- total
 - md4_context, 150
 - md5_context, 152
 - sha1_context, 179
 - sha2_context, 181
 - sha4_context, 183
- TST1_ENTER
 - havege.c, 710
- TST1_LEAVE
 - havege.c, 711
- TST2_ENTER
 - havege.c, 711
- TST2_LEAVE
 - havege.c, 711
- tv_init
 - timing.c, 781
- type
 - fipa_expression_s, 70
 - LibMC::MCAgent, 146
 - mxml_node_s, 168
 - type_cb
 - testmxml.c, 528
 - TYPES
 - df.h, 287
 - uint32
 - aes.c, 661
 - uint64
 - sha4.h, 620
 - uint8
 - aes.c, 661
 - UL64
 - sha4.h, 620
 - UNAUTHORIZED
 - mtp_http.h, 393
 - UNORDERED_COLLECTION
 - mtp_http.h, 394
 - UNPROCESSABLE_ENTITY
 - mtp_http.h, 394
 - UNSUPPORTED_MEDIA_TYPE
 - mtp_http.h, 393
 - update_comment
 - mxmldoc.c, 522
 - update_name
 - message_s, 155
 - update_num
 - message_s, 155
 - UPGRADE_REQUIRED
 - mtp_http.h, 394
 - urls
 - fipa_url_sequence_s, 76
 - USAGE
 - aescrypt2.c, 791
 - user_agent
 - mtp_http_s, 162
 - v3_ext
 - _x509_cert, 25
 - val
 - _x509_name, 27
 - valid_from
 - _x509_cert, 24
 - valid_to
 - _x509_cert, 24
 - value
 - mxml_attr_s, 163
 - mxml_node_s, 169
 - var_name
 - agent_task_s, 44
 - vcnet/config.h
 - HAVE_STRDUP, 484
 - mxml_strdupf, 484
 - MXML_VERSION, 484
 - mxml_vsnprintf, 484

- [vsnprintf](#), [484](#)
- [vcnet2005/config.h](#)
 - [HAVE_STRDUP](#), [485](#)
 - [mxml_strdup](#), [485](#)
 - [MXML_VERSION](#), [485](#)
 - [mxml_vsnprintf](#), [485](#)
 - [vsnprintf](#), [485](#)
- [ver](#)
 - [rsa_context](#), [174](#)
- [verify_result](#)
 - [ssl_context](#), [191](#)
- [version](#)
 - [_x509_cert](#), [23](#)
- [vsnprintf](#)
 - [vcnet/config.h](#), [484](#)
 - [vcnet2005/config.h](#), [485](#)
- [WaitAgent](#)
 - [LibMC::MCAgency](#), [114](#)
- [waiting](#)
 - [df.h](#), [291](#)
- [waiting_cond](#)
 - [df.h](#), [291](#)
- [waiting_lock](#)
 - [df.h](#), [291](#)
- [WaitRetrieveAgent](#)
 - [LibMC::MCAgency](#), [114](#)
- [WaitSignal](#)
 - [LibMC::MCAgency](#), [119](#)
- [WAKE_QUEUE](#)
 - [macros.h](#), [381](#)
- [WALK](#)
 - [havege_state](#), [83](#)
- [WARN](#)
 - [macros.h](#), [381](#)
- [whitespace](#)
 - [mxml_text_s](#), [170](#)
- [whitespace_cb](#)
 - [testmxml.c](#), [529](#)
 - [xml_helper.c](#), [831](#)
 - [xml_helper.h](#), [400](#)
- [winconfig.h](#)
 - [PACKAGE_STRING](#), [826](#)
 - [PACKAGE_VERSION](#), [826](#)
 - [strtok_r](#), [826](#)
- [word](#)
 - [fipa_expression_s::content_u](#), [71](#)
- [write_documentation](#)
 - [mxmldoc.c](#), [522](#)
- [write_element](#)
 - [mxmldoc.c](#), [522](#)
- [write_fd](#)
 - [ssl_context](#), [189](#)
- [write_flag](#)
 - [mc_rwlock_s](#), [96](#)
- [write_request](#)
 - [mc_rwlock_s](#), [96](#)
- [write_string](#)
 - [mxmldoc.c](#), [522](#)
- [ws_cb](#)
 - [mxmldoc.c](#), [523](#)
- [X](#)
 - [dhm_context](#), [55](#)
- [x](#)
 - [arc4_context](#), [46](#)
- [x509.h](#)
 - [ASN1_BIT_STRING](#), [647](#)
 - [ASN1_BMP_STRING](#), [647](#)
 - [ASN1_BOOLEAN](#), [647](#)
 - [ASN1_CONSTRUCTED](#), [647](#)
 - [ASN1_CONTEXT_SPECIFIC](#), [647](#)
 - [ASN1_IA5_STRING](#), [648](#)
 - [ASN1_INTEGER](#), [648](#)
 - [ASN1_NULL](#), [648](#)
 - [ASN1_OCTET_STRING](#), [648](#)
 - [ASN1_OID](#), [648](#)
 - [ASN1_PRIMITIVE](#), [648](#)
 - [ASN1_PRINTABLE_STRING](#), [648](#)
 - [ASN1_SEQUENCE](#), [648](#)
 - [ASN1_SET](#), [649](#)
 - [ASN1_T61_STRING](#), [649](#)
 - [ASN1_UNIVERSAL_STRING](#), [649](#)
 - [ASN1_UTC_TIME](#), [649](#)
 - [ASN1_UTF8_STRING](#), [649](#)
 - [BADCERT_CN_MISMATCH](#), [649](#)
 - [BADCERT_HAS_EXPIRED](#), [649](#)
 - [BADCERT_NOT_TRUSTED](#), [649](#)
 - [ERR_ASN1_INVALID_DATA](#), [649](#)
 - [ERR_ASN1_INVALID_LENGTH](#), [650](#)
 - [ERR_ASN1_LENGTH_MISMATCH](#), [650](#)
 - [ERR_ASN1_OUT_OF_DATA](#), [650](#)
 - [ERR_ASN1_UNEXPECTED_TAG](#), [650](#)
 - [ERR_X509_CERT_INVALID_ALG](#), [650](#)
 - [ERR_X509_CERT_INVALID_DATE](#), [650](#)
 - [ERR_X509_CERT_INVALID_-
EXTENSIONS](#), [650](#)
 - [ERR_X509_CERT_INVALID_FORMAT](#), [650](#)
 - [ERR_X509_CERT_INVALID_NAME](#), [651](#)
 - [ERR_X509_CERT_INVALID_PEM](#), [651](#)
 - [ERR_X509_CERT_INVALID_PUBKEY](#), [651](#)
 - [ERR_X509_CERT_INVALID_SERIAL](#), [651](#)
 - [ERR_X509_CERT_INVALID_SIGNATURE](#),
[651](#)
 - [ERR_X509_CERT_INVALID_VERSION](#),
[651](#)
 - [ERR_X509_CERT_SIG_MISMATCH](#), [651](#)

- ERR_X509_CERT_UNKNOWN_PK_ALG, 651
- ERR_X509_CERT_UNKNOWN_SIG_ALG, 651
- ERR_X509_CERT_UNKNOWN_VERSION, 652
- ERR_X509_FEATURE_UNAVAILABLE, 652
- ERR_X509_KEY_INVALID_ENC_IV, 652
- ERR_X509_KEY_INVALID_FORMAT, 652
- ERR_X509_KEY_INVALID_PEM, 652
- ERR_X509_KEY_INVALID_VERSION, 652
- ERR_X509_KEY_PASSWORD_- MISMATCH, 652
- ERR_X509_KEY_PASSWORD_- REQUIRED, 652
- ERR_X509_KEY_UNKNOWN_ENC_ALG, 652
- ERR_X509_SIG_VERIFY_FAILED, 653
- OID_PKCS1, 653
- OID_PKCS1_RSA, 653
- OID_PKCS9, 653
- OID_X520, 653
- PKCS9_EMAIL, 653
- x509_add_certs, 654
- x509_buf, 654
- x509_cert, 654
- x509_cert_info, 655
- x509_dn_gets, 655
- x509_free_cert, 655
- x509_is_cert_expired, 655
- x509_name, 654
- x509_parse_key, 656
- x509_read_crtfile, 656
- x509_read_keyfile, 656
- x509_self_test, 657
- x509_time, 654
- x509_verify_cert, 657
- X520_COMMON_NAME, 653
- X520_COUNTRY, 653
- X520_LOCALITY, 653
- X520_ORG_UNIT, 654
- X520_ORGANIZATION, 654
- X520_STATE, 654
- x509_add_certs
 - x509.h, 654
 - x509read.c, 784
- x509_buf
 - x509.h, 654
- x509_cert
 - x509.h, 654
- x509_cert_info
 - x509.h, 655
 - x509read.c, 785
- x509_des3_decrypt
 - x509read.c, 785
- x509_des3_getiv
 - x509read.c, 785
- x509_dn_gets
 - x509.h, 655
 - x509read.c, 785
- x509_free_cert
 - x509.h, 655
 - x509read.c, 785
- x509_get_alg
 - x509read.c, 786
- x509_get_dates
 - x509read.c, 786
- x509_get_ext
 - x509read.c, 786
- x509_get_name
 - x509read.c, 786
- x509_get_pubkey
 - x509read.c, 786
- x509_get_serial
 - x509read.c, 787
- x509_get_sig
 - x509read.c, 787
- x509_get_uid
 - x509read.c, 787
- x509_get_version
 - x509read.c, 787
- x509_hash
 - x509read.c, 787
- x509_is_cert_expired
 - x509.h, 655
 - x509read.c, 788
- x509_name
 - x509.h, 654
- x509_parse_key
 - x509.h, 656
 - x509read.c, 788
- x509_read_crtfile
 - x509.h, 656
 - x509read.c, 788
- x509_read_keyfile
 - x509.h, 656
 - x509read.c, 789
- x509_self_test
 - x509.h, 657
 - x509read.c, 789
- x509_time
 - x509.h, 654
- x509_verify_cert
 - x509.h, 657
 - x509read.c, 789
- x509read.c
 - _CRT_SECURE_NO_DEPRECATED, 783

- [_x509_read_src, 790](#)
 - [asn1_get_bool, 783](#)
 - [asn1_get_int, 783](#)
 - [asn1_get_len, 784](#)
 - [asn1_get_mpi, 784](#)
 - [asn1_get_tag, 784](#)
 - [x509_add_certs, 784](#)
 - [x509_cert_info, 785](#)
 - [x509_des3_decrypt, 785](#)
 - [x509_des3_getiv, 785](#)
 - [x509_dn_gets, 785](#)
 - [x509_free_cert, 785](#)
 - [x509_get_alg, 786](#)
 - [x509_get_dates, 786](#)
 - [x509_get_ext, 786](#)
 - [x509_get_name, 786](#)
 - [x509_get_pubkey, 786](#)
 - [x509_get_serial, 787](#)
 - [x509_get_sig, 787](#)
 - [x509_get_uid, 787](#)
 - [x509_get_version, 787](#)
 - [x509_hash, 787](#)
 - [x509_is_cert_expired, 788](#)
 - [x509_parse_key, 788](#)
 - [x509_read_crtfile, 788](#)
 - [x509_read_keyfile, 789](#)
 - [x509_self_test, 789](#)
 - [x509_verify_cert, 789](#)
- [X520_COMMON_NAME](#)
 - [x509.h, 653](#)
- [X520_COUNTRY](#)
 - [x509.h, 653](#)
- [X520_LOCALITY](#)
 - [x509.h, 653](#)
- [X520_ORG_UNIT](#)
 - [x509.h, 654](#)
- [X520_ORGANIZATION](#)
 - [x509.h, 654](#)
- [X520_STATE](#)
 - [x509.h, 654](#)
- [xml_agent_root](#)
 - [agent_datastate_s, 35](#)
- [xml_compose.c](#)
 - [agent_xml_compose, 827](#)
 - [agent_xml_compose__agent_code, 827](#)
 - [agent_xml_compose__agent_data, 827](#)
 - [agent_xml_compose__create_row_nodes, 828](#)
 - [agent_xml_compose__data, 828](#)
 - [agent_xml_compose__gaf_message, 828](#)
 - [agent_xml_compose__home, 828](#)
 - [agent_xml_compose__message, 828](#)
 - [agent_xml_compose__mobile_agent, 829](#)
 - [agent_xml_compose__name, 829](#)
 - [agent_xml_compose__owner, 829](#)
 - [agent_xml_compose__row, 829](#)
 - [agent_xml_compose__task, 829](#)
 - [agent_xml_compose__tasks, 829](#)
- [xml_compose.h](#)
 - [agent_xml_compose, 397](#)
 - [agent_xml_compose__agent_code, 397](#)
 - [agent_xml_compose__agent_data, 397](#)
 - [agent_xml_compose__create_row_nodes, 397](#)
 - [agent_xml_compose__data, 398](#)
 - [agent_xml_compose__gaf_message, 398](#)
 - [agent_xml_compose__home, 398](#)
 - [agent_xml_compose__message, 398](#)
 - [agent_xml_compose__mobile_agent, 398](#)
 - [agent_xml_compose__name, 399](#)
 - [agent_xml_compose__owner, 399](#)
 - [agent_xml_compose__row, 399](#)
 - [agent_xml_compose__task, 399](#)
 - [agent_xml_compose__tasks, 399](#)
- [xml_find_sibling](#)
 - [xml_helper.c, 831](#)
 - [xml_helper.h, 400](#)
- [xml_get_cdata](#)
 - [xml_helper.c, 831](#)
 - [xml_helper.h, 400](#)
- [xml_get_child](#)
 - [xml_helper.c, 831](#)
 - [xml_helper.h, 400](#)
- [xml_get_deep_child](#)
 - [xml_helper.c, 832](#)
 - [xml_helper.h, 400](#)
- [xml_get_element_name](#)
 - [xml_helper.c, 832](#)
 - [xml_helper.h, 401](#)
- [xml_get_next_element](#)
 - [xml_helper.c, 832](#)
 - [xml_helper.h, 401](#)
- [xml_get_text](#)
 - [xml_helper.c, 832](#)
 - [xml_helper.h, 401](#)
- [xml_helper.c](#)
 - [whitespace_cb, 831](#)
 - [xml_find_sibling, 831](#)
 - [xml_get_cdata, 831](#)
 - [xml_get_child, 831](#)
 - [xml_get_deep_child, 832](#)
 - [xml_get_element_name, 832](#)
 - [xml_get_next_element, 832](#)
 - [xml_get_text, 832](#)
 - [xml_new_cdata, 832](#)
- [xml_helper.h](#)
 - [whitespace_cb, 400](#)
 - [xml_find_sibling, 400](#)
 - [xml_get_cdata, 400](#)
 - [xml_get_child, 400](#)

- xml_get_deep_child, 400
- xml_get_element_name, 401
- xml_get_next_element, 401
- xml_get_text, 401
- xml_new_cdata, 401
- xml_new_cdata
 - xml_helper.c, 832
 - xml_helper.h, 401
- xml_parser.c
 - _XOPEN_SOURCE, 833
 - agent_return_xml_parse, 834
 - agent_xml_parse, 834
 - agent_xml_parse__agent_code, 834
 - agent_xml_parse__agent_data, 834
 - agent_xml_parse__data, 834
 - agent_xml_parse__fill_row_data, 834
 - agent_xml_parse__home, 835
 - agent_xml_parse__mobile_agent, 835
 - agent_xml_parse__name, 835
 - agent_xml_parse__owner, 835
 - agent_xml_parse__row, 835
 - agent_xml_parse__task, 835
 - agent_xml_parse__tasks, 836
 - message_xml_parse, 836
 - message_xml_parse__message, 836
- xml_parser.h
 - agent_xml_parse, 402
 - agent_xml_parse__agent_code, 402
 - agent_xml_parse__agent_data, 402
 - agent_xml_parse__data, 403
 - agent_xml_parse__fill_row_data, 403
 - agent_xml_parse__home, 403
 - agent_xml_parse__mobile_agent, 403
 - agent_xml_parse__name, 403
 - agent_xml_parse__owner, 404
 - agent_xml_parse__row, 404
 - agent_xml_parse__task, 404
 - agent_xml_parse__tasks, 404
 - message_xml_parse, 404
 - message_xml_parse__message, 404
 - STRUCT, 405
- xml_payload
 - message_s, 155
- xml_root
 - agent_datastate_s, 35
 - message_s, 155
- XTIME
 - aes.c, 661
- xyssl_ca_cert
 - certs.c, 688
 - certs.h, 566
- y
 - arc4_context, 46
- year
 - _x509_time, 28
 - fipa_DateTime_s, 68
- ZERO
 - df.h, 287