The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

Notwithstanding any other warranty herein, all document files and software of these suppliers are provided “as is” with all faults. Cisco and the above-named suppliers disclaim all warranties, expressed or implied, including, without limitation, those of merchantability, fitness for a particular purpose and noninfringement or arising from a course of dealing, usage, or trade practice.

In no event shall Cisco or its suppliers be liable for any indirect, special, consequential, or incidental damages, including, without limitation, lost profits or loss or damage to data arising out of the use or inability to use this manual, even if Cisco or its suppliers have been advised of the possibility of such damages.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Cisco Unified Communications Manager Managed Services Guide

© 2010 Cisco Systems, Inc. All rights reserved.
CONTENTS

Preface xxvii
    Purpose xxvii
    Audience xxvii
    Organization xxviii
    Related Documentation xxviii
    Conventions xxix
    Obtaining Documentation and Submitting a Service Request xxx
    Cisco Product Security Overview xxx

Related Documentation xxxi

CHAPTER 1 Overview 1-1
    Cisco Unified Communications Manager 1-1
    Supported Deployment Models 1-2
    Managed Services 1-3
    Cisco Unified Serviceability 1-4
        Trace Tools 1-4
        Troubleshooting Trace 1-5
        Trace Collection 1-5
    Cisco Unified Reporting 1-5
    Cisco Unified Real-Time Monitoring Tool 1-6
    Call Detail Records and Call Management Records 1-7
    Call Detail Record Analysis and Reporting 1-7
    Management Information Base 1-8

CHAPTER 2 New and Changed Information 2-1
    Cisco Unified Communications Manager, Release 8.5(1) 2-1
        Audit Log Support for Cisco Unity Connection 2-2
        Alarm Additions and Changes 2-3
        Enhanced Reason Codes 2-4
        New Perfmon Counters for Cisco SIP Normalization 2-6
        SNMP MIBs 2-13
        Supported Servers 2-13
Chapter 3 Managing and Monitoring the Health of Cisco Unified Communications Manager Systems 3-1

Overview of Supported Interfaces 3-1
Critical Processes to Monitor 3-2
Available Supported MIBs 3-11
RTMT Monitoring of Cisco Unified CM System Health 3-12
RTMT Summary View 3-12
CPU Usage 3-13
% IOwait Monitoring 3-15
Virtual Memory 3-16
Disk Usage 3-17
  Disk Name Mapping 3-18
Database Replication and Cisco Unified Communication Manager Nodes 3-20
Cisco Unified CM Process and CPU Usage 3-20
CodeYellow 3-21
RIS Data Collector PerfMonLog 3-23
Critical Service Status 3-24
Syslog Messages 3-25
RTMT Alerts as Syslog Messages and Traps 3-26
Recovery, Hardware Migration, and Backup/Restore 3-26
  Backup/Restore 3-26
Platform Monitoring 3-27
  Using SNMP MIBs 3-27
    MIBs and MCS Types 3-28
  Using Command Line Interface 3-28
  Hardware Migration 3-31
  Platform Security 3-31
    Locked-down System 3-32
    Cisco Security Agent Support 3-32
    Security Patching and Updating 3-32
    Role-Based Access Control 3-32
Software Configuration Management 3-32
  General Install/Upgrade Procedures 3-33
  Detecting Installed Release and Packages 3-33
Available Reports 3-33
  RTMT Reports 3-33
  Serviceability Reports 3-34
  Cisco Unified Reporting 3-34
General Health and Troubleshooting Tips 3-35
  Using of Onboard Agents 3-36
  Call Detail Records and Call Maintenance Records 3-36
  Perfmon Counters 3-36
  Integration with Uninterruptible Power Supplies (UPS) 3-37
  Native Hardware Out of Band Management (OOB) 3-37
  Phone Registration Status 3-37
  Historical Information Download 3-37
  Cisco CallManager Service Stops Responding 3-38
Cisco HW Conference Bridge Device 5-22
Cisco IP Manager Assistant 5-22
Cisco Lines 5-23
Cisco Locations 5-23
Cisco Media Streaming Application 5-24
Cisco Messaging Interface 5-27
Cisco MGCP BRI Device 5-28
Cisco MGCP FXO Device 5-29
Cisco MGCP FXS Device 5-29
Cisco MGCP Gateways 5-30
Cisco MGCP PRI Device 5-30
Cisco MGCP T1 CAS Device 5-31
Cisco Mobility Manager 5-32
Cisco Music On Hold (MOH) Device 5-33
Cisco MTP Device 5-34
Cisco Phones 5-34
Cisco Presence Feature 5-34
Cisco QSIG Feature 5-35
Cisco Signaling Performance 5-35
Cisco SIP 5-36
Cisco SIP Normalization 5-36
Cisco SIP Stack 5-43
Cisco SIP Station 5-51
Cisco SW Conf Bridge Device 5-52
Cisco TFTP Server 5-53
Cisco Transcode Device 5-56
Cisco Video Conference Bridge 5-57
Cisco Web Dialer 5-58
Cisco WSM Connector 5-58
PerfMon Objects and Counters for System 5-59
Cisco Tomcat Connector 5-59
Cisco Tomcat JVM 5-61
Cisco Tomcat Web Application 5-61
Database Change Notification Client 5-62
Database Change Notification Server 5-63
Database Change Notification Subscription 5-64
Database Local DSN 5-64
DB User Host Information Counters 5-64
Enterprise Replication DBSpace Monitors 5-64
Enterprise Replication Perfmon Counters 5-65
Contents

IP 5-65
Memory 5-66
Network Interface 5-67
Number of Replicates Created and State of Replication 5-68
Partition 5-69
Process 5-70
Processor 5-71
System 5-72
TCP 5-73
Thread 5-73
Cisco Intercompany Media Engine Performance Objects and Alerts 5-74
Cisco Intercompany Media Engine Server Objects 5-74
IME Configuration Manager 5-74
IME Server 5-74
IME Server System Performance 5-77
Cisco Intercompany Media Engine Server Alerts 5-78
Cisco Unified Communications Manager Server Objects 5-78
IME Client 5-79
IME Client Instance 5-80
Cisco Unified Communications Manager Server Alerts 5-80

CHAPTER 6 Cisco Unified Serviceability Alarms and CiscoLog Messages 6-1
Cisco Unified Serviceability Alarms and CiscoLog Messages 6-2
CiscoLog Format 6-2
Log File and Syslog Outputs 6-3
Standard Syslog Server Implementations 6-4
Clock Synchronization 6-4
Multipart Messages 6-4
CiscoLog Message Format 6-5
Message Length Limit 6-6
SEQNUM Field 6-6
HOST Field 6-6
TIMESTAMP Field 6-8
HEADER Field 6-10
TAGS Field 6-14
MESSAGE Field 6-17
Internationalization 6-18
Versioning 6-18
Preconfigured System Alarm Notifications 6-19
AuthenticationFailed  6-19
CiscoDRFFailure  6-20
CoreDumpFileFound  6-21
CpuPegging  6-21
CriticalServiceDown  6-22
HardwareFailure  6-22
LogFileSearchStringFound  6-23
LogPartitionHighWaterMarkExceeded  6-23
LogPartitionLowWaterMarkExceeded  6-24
LowActivePartitionAvailableDiskSpace  6-25
LowAvailableVirtualMemory  6-25
LowInactivePartitionAvailableDiskSpace  6-26
LowSwapPartitionAvailableDiskSpace  6-26
ServerDown  6-27
SparePartitionHighWaterMarkExceeded  6-28
SparePartitionLowWaterMarkExceeded  6-29
SyslogSeverityMatchFound  6-30
SyslogStringMatchFound  6-30
SystemVersionMismatched  6-31
TotalProcessesAndThreadsExceededThreshold  6-31
Preconfigured CallManager Alarm Notifications  6-32
BeginThrottlingCallListBLFSubscriptions  6-33
CallProcessingNodeCpuPegging  6-33
CDRAgentSendFileFailed  6-34
CDRFileDeliveryFailed  6-35
CDRHighWaterMarkExceeded  6-35
CDRMaximumDiskSpaceExceeded  6-36
CodeYellow  6-36
DBChangeNotifyFailure  6-37
DBReplicationFailure  6-37
DDRBlockPrevention  6-38
DDRDown  6-39
ExcessiveVoiceQualityReports  6-39
IMEDistributedCacheInactive  6-40
IMEOverQuota  6-40
IMEQualityAlert  6-41
InsufficientFallbackIdentifiers  6-42
IMEServiceStatus  6-42
InvalidCredentials  6-43
LowCallManagerHeartbeatRate  6-44
LowTFTPServerHeartbeatRate  6-44
MaliciousCallTrace  6-45
MediaListExhausted  6-45
MgcpDChannelOutOfService  6-46
NumberOfRegisteredDevicesExceeded  6-46
NumberOfRegisteredGatewaysDecreased  6-47
NumberOfRegisteredGatewaysIncreased  6-47
NumberOfRegisteredMediaDevicesDecreased  6-48
NumberOfRegisteredMediaDevicesIncreased  6-48
NumberOfRegisteredPhonesDropped  6-49
RouteListExhausted  6-49
SDLLinkOutOfService  6-50
TCPSetupToIMEFailed  6-50
TLSConnectionToIMEFailed  6-51
Emergency-Level Alarms  6-52
BDINotStarted  6-52
CallDirectorCreationError  6-52
CiscoDirSyncStartFailure  6-53
ExceptionInInitSDIConfiguration  6-53
FileWriteError  6-53
GlobalSPUtilsCreationError  6-54
HuntGroupControllerCreationError  6-54
HuntGroupCreationError  6-55
IPAddressResolveError  6-55
IPMANotStarted  6-55
LineStateSrvEngCreationError  6-56
LostConnectionToCM  6-56
NoCMEntriesInDB  6-57
NoFeatureLicense  6-57
OutOfMemory  6-58
ServiceNotInstalled  6-58
SyncDBCreationError  6-58
SysControllerCreationError  6-59
TapiLinesTableCreationError  6-59
TimerServicesCreationError  6-60
TestAlarmEmergency  6-60
WDNotStarted  6-60
Alert-Level Alarms  6-61
CertValidLessThanADay  6-61
CMIException  6-61
CMOverallInitTimeExceeded 6-62
ConfigThreadChangeNotifyServerInstanceFailed 6-63
ConfigThreadChangeNotifyServerSingleFailed 6-64
ConfigThreadChangeNotifyServerStartFailed 6-64
CiscoLicenseApproachingLimit 6-65
CiscoLicenseOverDraft 6-65
CMVersionMismatch 6-65
CreateThreadFailed 6-66
DBLException 6-67
InvalidCredentials 6-68
MemAllocFailed 6-69
NoDbConnectionAvailable 6-69
ParityConfigurationError 6-70
SerialPortOpeningError 6-70
SDIControlLayerFailed 6-71
SDLLinkOOS 6-72
SocketError 6-73
StopBitConfigurationError 6-73
TFTPServerListenSetSockOptFailed 6-74
TFTPServerListenBindFailed 6-75
TestAlarmAlert 6-75
TLSConnectionToIMEFailed 6-76
TVSServerListenBindFailed 6-76
TVSServerListenSetSockOptFailed 6-77
UnknownException 6-78
VMDNConfigurationException 6-78

Critical-Level Alarms 6-79
BChannelOOS 6-79
CallManagerFailure 6-80
CertExpiryCritical 6-81
CertValidfor7days 6-82
CDRMaximumDiskSpaceExceeded 6-82
CiscoDirSyncProcessFailToStart 6-83
CodeRedEntry 6-84
CodeYellowEntry 6-85
CoreDumpFileFound 6-85
DChannelOOS 6-86
DUPLEX_MISMATCH 6-87
ErrorChangeNotifyClientBlock 6-87
LogPartitionHighWaterMarkExceeded 6-88
<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>MaxCallsReached</td>
<td>6-89</td>
</tr>
<tr>
<td>MGCPGatewayLostComm</td>
<td>6-89</td>
</tr>
<tr>
<td>StationTCPInitError</td>
<td>6-90</td>
</tr>
<tr>
<td>TCPSetupToIMEFailed</td>
<td>6-90</td>
</tr>
<tr>
<td>TimerThreadSlowed</td>
<td>6-91</td>
</tr>
<tr>
<td>TestAlarmCritical</td>
<td>6-92</td>
</tr>
<tr>
<td>Error-Level Alarms</td>
<td>6-92</td>
</tr>
<tr>
<td>ANNDeviceRecoveryCreateFailed</td>
<td>6-92</td>
</tr>
<tr>
<td>AwaitingResponseFromPDPTimeout</td>
<td>6-93</td>
</tr>
<tr>
<td>BadCDRFileFound</td>
<td>6-94</td>
</tr>
<tr>
<td>BDIApplicationError</td>
<td>6-95</td>
</tr>
<tr>
<td>BDIOverloaded</td>
<td>6-95</td>
</tr>
<tr>
<td>CARSchedulerJobError</td>
<td>6-95</td>
</tr>
<tr>
<td>CARSchedulerJobFailed</td>
<td>6-96</td>
</tr>
<tr>
<td>CCDIPReachableTimeOut</td>
<td>6-98</td>
</tr>
<tr>
<td>CCDPSTNFailOverDurationTimeOut</td>
<td>6-98</td>
</tr>
<tr>
<td>CDRAgentSendFileFailed</td>
<td>6-99</td>
</tr>
<tr>
<td>CDRAgentSendFileFailureContinues</td>
<td>6-100</td>
</tr>
<tr>
<td>CDRFileDeliveryFailed</td>
<td>6-100</td>
</tr>
<tr>
<td>CDRFileDeliveryFailureContinues</td>
<td>6-101</td>
</tr>
<tr>
<td>CFBDeviceRecoveryCreateFailed</td>
<td>6-102</td>
</tr>
<tr>
<td>CiscoDhcpdFailure</td>
<td>6-102</td>
</tr>
<tr>
<td>CiscoDirSyncProcessFailedRetry</td>
<td>6-103</td>
</tr>
<tr>
<td>CiscoDirSyncProcessFailedNoRetry</td>
<td>6-103</td>
</tr>
<tr>
<td>CiscoDirSyncProcessConnectionFailed</td>
<td>6-104</td>
</tr>
<tr>
<td>CiscoDirSyncDBAccessFailure</td>
<td>6-104</td>
</tr>
<tr>
<td>CiscoLicenseManagerDown</td>
<td>6-105</td>
</tr>
<tr>
<td>CiscoLicenseRequestFailed</td>
<td>6-105</td>
</tr>
<tr>
<td>CiscoLicenseDataStoreError</td>
<td>6-105</td>
</tr>
<tr>
<td>CiscoLicenseInternalError</td>
<td>6-106</td>
</tr>
<tr>
<td>CiscoLicenseFileError</td>
<td>6-106</td>
</tr>
<tr>
<td>CLM_MsgIntChkError</td>
<td>6-107</td>
</tr>
<tr>
<td>CLM_UnrecognizedHost</td>
<td>6-107</td>
</tr>
<tr>
<td>ConfigItAllBuildFilesFailed</td>
<td>6-108</td>
</tr>
<tr>
<td>ConfigItAllReadConfigurationFailed</td>
<td>6-108</td>
</tr>
<tr>
<td>ConfigThreadBuildFileFailed</td>
<td>6-109</td>
</tr>
<tr>
<td>ConfigThreadCNCMGrpBuildFileFailed</td>
<td>6-109</td>
</tr>
<tr>
<td>ConfigThreadCNGrpBuildFileFailed</td>
<td>6-110</td>
</tr>
<tr>
<td>ConfigThreadReadConfigurationFailed</td>
<td>6-110</td>
</tr>
<tr>
<td>ConfigThreadUnknownExceptionCaught</td>
<td>6-111</td>
</tr>
</tbody>
</table>
DRFBackupCancelInternalError 6-141
DRFLogDirAccessFailure 6-142
DRFFailure 6-143
DRFLocalDeviceError 6-143
DuplicateLearnedPattern 6-144
EMAppInitializationFailed 6-145
EMCCFailedInLocalCluster 6-145
EMServiceConnectionError 6-146
EndPointTransientConnection 6-147
EndPointUnregistered 6-152
ErrorChangeNotifyClientTimeout 6-158
ErrorParsingDirectiveFromPDP 6-158
ErrorReadingInstalledRPMS 6-159
FailureResponseFromPDP 6-159
FailedToReadConfig 6-160
FirewallMappingFailure 6-161
ICTCallThrottlingStart 6-161
IDSEngineCritical 6-162
IDSEngineFailure 6-163
IDSReplicationFailure 6-163
InsufficientFallbackIdentifiers 6-164
InvalidIPNetPattern 6-165
InvalidPortHandle 6-165
IPMAApplicationError 6-166
IPMAOverloaded 6-166
IPMAFilteringDown 6-167
IPv6InterfaceNotInstalled 6-167
kANNDeviceRecordNotFound 6-168
kCFBDeviceRecordNotFound 6-169
kCreateAudioSourcesFailed 6-169
kCreateControlFailed 6-170
kDbConnectionFailed 6-171
kIPVMSDeviceDriverNotFound 6-171
kIpVmsMgrNoLocalHostName 6-172
kIpVmsMgrNoLocalNetworkIPAddr 6-172
kIPVMSMgrWrongDriverVersion 6-173
kMOHTFTPGoRequestFailed 6-174
kPWavMgrThreadxFailed 6-175
kReadCfgUserLocaleEnterpriseSvcParm 6-175
kRequestedANNStreamsFailed 6-176
<table>
<thead>
<tr>
<th>Alarm Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LostConnectionToSAFForwarder</td>
<td>6-176</td>
</tr>
<tr>
<td>MultipleSIPTrunksToSamePeerAndLocalPort</td>
<td>6-177</td>
</tr>
<tr>
<td>NodeNotTrusted</td>
<td>6-177</td>
</tr>
<tr>
<td>NumDevRegExceeded</td>
<td>6-178</td>
</tr>
<tr>
<td>PublishFailedOverQuota</td>
<td>6-178</td>
</tr>
<tr>
<td>ReadConfigurationUnknownException</td>
<td>6-179</td>
</tr>
<tr>
<td>ReadingFileFailure</td>
<td>6-180</td>
</tr>
<tr>
<td>RsvpNoMoreResourcesAvailable</td>
<td>6-181</td>
</tr>
<tr>
<td>RTMT_ALERT</td>
<td>6-183</td>
</tr>
<tr>
<td>RTMT-ERROR-ALERT</td>
<td>6-183</td>
</tr>
<tr>
<td>SAFForwarderError</td>
<td>6-184</td>
</tr>
<tr>
<td>SAFResponderError</td>
<td>6-187</td>
</tr>
<tr>
<td>ScheduledCollectionError</td>
<td>6-188</td>
</tr>
<tr>
<td>SerialPortGetStatusError</td>
<td>6-188</td>
</tr>
<tr>
<td>SerialPortSetStatusError</td>
<td>6-189</td>
</tr>
<tr>
<td>ServiceActivationFailed</td>
<td>6-189</td>
</tr>
<tr>
<td>ServiceDeactivationFailed</td>
<td>6-190</td>
</tr>
<tr>
<td>ServiceFailed</td>
<td>6-190</td>
</tr>
<tr>
<td>ServiceStartFailed</td>
<td>6-191</td>
</tr>
<tr>
<td>ServiceStopFailed</td>
<td>6-191</td>
</tr>
<tr>
<td>ServiceExceededMaxRestarts</td>
<td>6-191</td>
</tr>
<tr>
<td>SIPNormalizationResourceWarning</td>
<td>6-192</td>
</tr>
<tr>
<td>SIPNormalizationScriptError</td>
<td>6-193</td>
</tr>
<tr>
<td>SIPTrunkOOS</td>
<td>6-195</td>
</tr>
<tr>
<td>SparePartitionLowWaterMarkExceeded</td>
<td>6-197</td>
</tr>
<tr>
<td>SystemResourceError</td>
<td>6-198</td>
</tr>
<tr>
<td>TestAlarmError</td>
<td>6-198</td>
</tr>
<tr>
<td>ThreadPoolProxyUnknownException</td>
<td>6-199</td>
</tr>
<tr>
<td>UnableToRegisterwithCallManagerService</td>
<td>6-199</td>
</tr>
<tr>
<td>UserLoginFailed</td>
<td>6-200</td>
</tr>
<tr>
<td>WritingFileFailure</td>
<td>6-200</td>
</tr>
<tr>
<td>WDAppllicationError</td>
<td>6-201</td>
</tr>
<tr>
<td>WDOverloaded</td>
<td>6-201</td>
</tr>
<tr>
<td><strong>Warning-Level Alarms</strong></td>
<td>6-202</td>
</tr>
<tr>
<td>AnnunciatorNoMoreResourcesAvailable</td>
<td>6-202</td>
</tr>
<tr>
<td>ApplicationConnectionDropped</td>
<td>6-203</td>
</tr>
<tr>
<td>ApplicationConnectionError</td>
<td>6-204</td>
</tr>
<tr>
<td>authAdminLock</td>
<td>6-204</td>
</tr>
<tr>
<td>AuthenticationFailed</td>
<td>6-205</td>
</tr>
<tr>
<td>authFail</td>
<td>6-205</td>
</tr>
</tbody>
</table>
authHackLock  6-206
authInactiveLock  6-207
authLdapInactive  6-207
BDIStopped  6-208
CallAttemptBlockedByPolicy  6-208
CCDLearnedPatternLimitReached  6-209
CDRHWMExceeded  6-210
CertValidLessThanMonth  6-210
ConferenceNoMoreResourcesAvailable  6-211
CtiDeviceOpenFailure  6-212
CtiLineOpenFailure  6-213
CtiIncompatibleProtocolVersion  6-214
CtiMaxConnectionReached  6-215
CtiProviderCloseHeartbeatTimeout  6-216
CtiQbeFailureResponse  6-216
DaTimeOut  6-217
DeviceImageDownloadFailure  6-218
DevicePartiallyRegistered  6-220
DeviceTransientConnection  6-224
DeviceUnregistered  6-229
DigitAnalysisTimeoutAwaitingResponse  6-234
DirSyncNoSchedulesFound  6-234
DirSyncScheduledTaskTimeoutOccurred  6-235
DRFComponentDeRegistered  6-235
DRFDeRegistrationFailure  6-236
DRFDeRegisteredServer  6-237
DRFNoBackupTaken  6-237
DRFSchedulerDisabled  6-238
EMCCFailedInRemoteCluster  6-238
ErrorParsingResponseFromPDP  6-239
FailedToFillIDirectiveFromPDP  6-240
H323Stopped  6-241
InvalidSubscription  6-242
InvalidQBEMessage  6-243
IPMAManagerLogout  6-243
IPMAStopped  6-244
kANNAudioFileMissing  6-244
kANNAudioUndefinedAnnID  6-245
kANNAudioUndefinedLocale  6-245
kANNDeviceStartingDefaults  6-246
kCFBDeviceStartingDefaults 6-246
kChangeNotifyServiceCreationFailed 6-247
kChangeNotifyServiceGetEventFailed 6-248
kChangeNotifyServiceRestartFailed 6-249
kDeviceDriverError 6-250
kDeviceMgrCreateFailed 6-251
kDeviceMgrOpenReceiveFailedOutOfStreams 6-251
kDeviceMgrRegisterKeepAliveResponseError 6-252
kDeviceMgrRegisterWithCallManagerError 6-253
kDeviceMgrSocketDrvNotifyEvtCreateFailed 6-253
kDeviceMgrSocketNotifyEventCreateFailed 6-254
kDeviceMgrStartTransmissionOutOfStreams 6-255
kDeviceMgrThreadxFailed 6-255
kFixedInputCodecStreamFailed 6-256
kFixedInputCreateControlFailed 6-257
kFixedInputCreateSoundCardFailed 6-258
kFixedInputInitSoundCardFailed 6-258
kFixedInputTranscoderFailed 6-259
kGetFileNameFailed 6-260
kIPVMSMgrEventCreationFailed 6-260
kIPVMSMgrThreadxFailed 6-261
kIpVmsMgrThreadWaitFailed 6-262
kMOHMgrCreateFailed 6-262
kMOHMgrExitEventCreationFailed 6-263
kMOHMgrThreadxFailed 6-264
kMTPDeviceRecordNotFound 6-265
kRequestedCFBStreamsFailed 6-265
kRequestedMOHStreamsFailed 6-266
kRequestedMTPStreamsFailed 6-266
LogCollectionJobLimitExceeded 6-267
LDAPServerUnreachable 6-267
LogPartitionLowWaterMarkExceeded 6-268
MaliciousCall 6-268
MaxDevicesPerNodeExceeded 6-269
MaxDevicesPerProviderExceeded 6-270
MediaResourceListExhausted 6-270
MemAllocFailed 6-272
MohNoMoreResourcesAvailable 6-273
MtpNoMoreResourcesAvailable 6-274
MTPDeviceRecoveryCreateFailed 6-276
NotEnoughChans 6-277
NoCallManagerFound 6-278
PublishFailed 6-279
QRTRequest 6-279
RejectedRoutes 6-280
RouteListExhausted 6-281
ServiceStartupFailed 6-281
ServingFileWarning 6-282
SparePartitionHighWaterMarkExceeded 6-282
SSUserNotInDB 6-283
SIPStopped 6-284
SIPLineRegistrationError 6-285
SIPTrunkPartiallyISV 6-289
SoftwareLicenseNotValid 6-291
StationEventAlert 6-291
TestAlarmWarning 6-292
TotalProcessesAndThreadsExceededThresholdStart 6-292
ThreadKillingError 6-293
UnableToSetorResetMWI 6-294
UserInputFailure 6-294
UserUserPrecedenceAlarm 6-295
BeginThrottlingCallListBLFSubscriptions 6-297
kANNAudioCreateDirFailed 6-297
MOHDeviceRecoveryCreateFailed 6-298
kDeviceMgrExitEventCreationFailed 6-299
kMOHDeviceRecordNotFound 6-299
kMOHBadMulticastIP 6-300
SSODisabled 6-301
SSONullTicket 6-301
SSOServerUnreachable 6-302
WDStopped 6-302
Notice-Level Alarms 6-303
authExpired 6-303
authMustChange 6-304
BChannelISV 6-305
CallManagerOnline 6-305
CertValidityOver30Days 6-306
CodeYellowExit 6-306
credReadFailure 6-307
DbInsertValidatedDIDFailure 6-307
DChannelISV 6-308
EMAppStopped 6-309
EndPointRegistered 6-309
H323Started 6-312
ICTCallThrottlingEnd 6-313
kDeviceMgrMoreThan50SocketEvents 6-314
MGCPGatewayGainedComm 6-315
MaxCallDurationTimeout 6-315
SDLLinkISV 6-316
SIPNormalizationScriptOpened 6-318
SIPNormalizationScriptClosed 6-318
SIPNormalizationAutoResetDisabled 6-320
SIPStarted 6-321
SIPTrunkISV 6-322
SMDICmdError 6-323
SMDIMessageError 6-324
TestAlarmNotice 6-325
TotalProcessesAndThreadsExceededThresholdEnd 6-325

Informational-Level Alarms 6-326
AdministrativeEvent 6-326
AdminPassword 6-326
AuditEventGenerated 6-327
AgentOnline 6-327
AgentOffline 6-328
AuthenticationSucceeded 6-328
authSuccess 6-328
BDIStarted 6-329
BuildStat 6-329
CiscoDirSyncStarted 6-330
CiscoDirSyncProcessStarted 6-330
CiscoDirSyncProcessCompleted 6-330
CiscoDirSyncProcessStoppedManually 6-331
CiscoDirSyncProcessStoppedAuto 6-331
CLM_ConnectivityTest 6-332
CLM_IPSecCertUpdated 6-332
CLM_IPAddressChange 6-333
CLM_PeerState 6-333
credFullUpdateSuccess 6-334
credFullUpdateFailure 6-334
credReadSuccess 6-335
DRFSchedulerUpdated 6-370
EMAppStarted 6-370
EMCCUserLoggedIn 6-371
EMCCUserLoggedOut 6-371
EndPointResetInitiated 6-372
EndPointRestartInitiated 6-374
EndThrottlingCallListBLFSubscriptions 6-376
IDSEngineDebug 6-377
IDSEngineInformation 6-377
IDSReplicationInformation 6-378
IPMAInformation 6-378
IPMAStarted 6-379
ITLFileRegenerated 6-379
kANNICMPErrorNotification 6-379
kCFBICMPErrorNotification 6-380
kReadCfgIpTosMediaResourceToCmNotFound 6-381
kDeviceMgrLockoutWithCallManager 6-381
kDeviceMgrRegisterWithCallManager 6-382
kDeviceMgrThreadWaitFailed 6-383
kDeviceMgrUnregisterWithCallManager 6-384
kIPVMSSStarting 6-384
kIPVMSStopping 6-385
kMOHICMPErrorNotification 6-385
kMOHMgrThreadWaitFailed 6-386
kMOHMgrIsAudioSourceInUseThisIsNULL 6-387
kMOHRewindStreamControlNull 6-387
kMOHRewindStreamMediaPositionObjectNull 6-388
kMTPDeviceStartingDefaults 6-389
kReadCfgMOHEnabledCodecsNotFound 6-389
LoadShareDeActivateTimeout 6-390
LogFileSearchStringFound 6-390
MaxHoldDurationTimeout 6-391
PermissionDenied 6-392
PktCapServiceStarted 6-392
PktCapServiceStopped 6-392
PktCapOnDeviceStarted 6-393
PktCapOnDeviceStopped 6-393
PublicationRunCompleted 6-394
RedirectCallRequestFailed 6-394
RollBackToPre8.0Disabled 6-395
RollBackToPre8.0Enabled  6-395
RouteRemoved       6-396
SAFPublishRevoke   6-396
SAFUnknownService  6-397
SecurityEvent      6-398
ServiceActivated   6-398
ServiceDeactivated 6-399
ServiceStarted     6-399
ServiceStopped     6-400
SoftwareLicenseValid 6-400
StationAlarm       6-400
StationConnectionError 6-401
TestAlarmAppliance 6-402
TestAlarmInformational 6-402
TVSCertificateRegenerated 6-403
UserAlreadyLoggedIn 6-403
UserLoggedOut      6-404
UserLoginSuccess   6-404
WDInformation      6-404
WDStarted          6-405
Debug-Level Alarms 6-405
TestAlarmDebug     6-405
Obsolete Alarms in Cisco Unified Communications Manager Release 8.0(1)  6-406
  Obsolete Alarms in CallManager Catalog  6-407
  Obsolete Alarms in CertMonitor Alarm Catalog  6-408
  Obsolete Alarms in CMI Alarm Catalog  6-408
  Obsolete Alarms in CTI Manager Alarm Catalog  6-408
  Obsolete Alarms in DB Alarm Catalog  6-410
  Obsolete Alarms in IpVms Alarm Catalog  6-410
  Obsolete Alarms in Test Alarm Catalog  6-413

CHAPTER 7 Cisco Management Information Base  7-1
CISCO-CCM-MIB  7-1
  Revisions  7-3
  Definitions  7-14
  Textual Conventions  7-14
  Objects  7-20
  Tables  7-21
    Cisco Unified CM Group Table  7-21
Cisco Unified CM Table  7-22
Cisco Unified CM Group Mapping Table  7-25
Cisco Unified CM Region Table  7-26
Cisco Unified CM Region Pair Table  7-27
Cisco Unified CM Time Zone Table  7-29
Device Pool Table  7-30
Cisco Unified CM Product Type Table  7-32
Phone Table  7-34
Phone Failed Table  7-40
Phone Status Update Table  7-43
Enhanced Phone Extension Table with Combination Index  7-45
Gateway Table  7-47
Gateway Trunk Table  7-52
All Scalar Objects  7-53
Media Device Table  7-59
CTI Device Table  7-62
CTI Device Directory Number Table  7-66
Alarms  7-67
  Cisco Unified CM Alarm Enable  7-67
  Phone Failed Config Objects  7-67
  Phone Status Update Config Objects  7-68
  Gateway Alarm Enable  7-69
  Malicious Call Alarm Enable  7-69
Notification and Alarms  7-70
  H323 Device Table  7-76
  Voice Mail Device Table  7-85
  Voice Mail Directory Number Table  7-88
  Quality Report Alarm Configuration Information  7-89
  Sip Device Table  7-89
  Notifications Types  7-92
  MIB Conformance Statements  7-96
  Compliance Statements  7-96
Cisco Unified CM Managed Services and SNMP Traps  7-106
Cisco Unified CM Alarms to Enable  7-106
  Traps to Monitor  7-107
  Dynamic Table Objects  7-109
  Static Table Objects  7-110
  Troubleshooting  7-111
  General Tips  7-111
For Linux and Cisco Unified CM Releases 5.x, 6.x, 7.x  7-114
Windows and Cisco Unified CM version 4.x 7-115
Limitations 7-115
Frequently Asked Questions 7-116
CISCO-CCM-CAPABILITY 7-121
Revisions 7-122
Definitions 7-122
Agent Capabilities 7-122
CISCO-CDP-MIB 7-127
Revisions 7-128
Definitions 7-129
CDP Interface Group 7-129
CDP Address Cache Group 7-132
CDP Global Group 7-139
Conformance Information 7-140
Compliance Statements 7-141
Units Of Conformance 7-141
Troubleshooting 7-143
Frequently Asked Questions 7-143
CISCO-SYSLOG-MIB 7-144
Revisions 7-145
Definitions 7-145
Object Identifiers 7-145
Textual Conventions 7-145
Basic Syslog Objects 7-146
Syslog Message History Table 7-147
Notifications 7-149
Conformance Information 7-150
Compliance Statements 7-150
Units of Conformance 7-150
Troubleshooting 7-150
Trap Configuration 7-150
Frequently Asked Questions 7-151
CISCO-SYSLOG-EXT-MIB 7-152
Revisions 7-153
Definitions 7-153
Textual Conventions 7-153
Syslog Configuration Group 7-155
cseSyslogServerTable 7-156
cseSyslogMessageControlTable 7-158
Contents

Conformance 7-160
Units of Conformance 7-161

Chapter 8 Industry-Standard Management Information Base 8-1

SYSAPPL-MIB 8-1
  Revisions 8-2
  Definitions 8-2
  System Application MIB 8-2
  Textual Conventions 8-3
  Installed Application Groups 8-3
    sysApplInstallPkgTable 8-4
    sysApplInstallElmtTable 8-6
  sysApplRun Group 8-10
    sysApplRunTable 8-10
  sysApplPastRunTable 8-12
  sysApplElmtRunTable 8-14
  sysApplElmtPastRunTable 8-17
Additional Scalar Objects that Control Table Sizes 8-21
  sysApplMap Group 8-23
  Conformance Macros 8-25
Troubleshooting 8-26
  Linux and Cisco Unified CM Releases 5.x, 6.x, 7.x 8-26
  Windows and Cisco Unified CM Release 4.x 8-26
  Using Servlets in Cisco Unified CM 7.x 8-27
  Frequently Asked Questions 8-28

RFC1213-MIB (MIB-II) 8-28
  Revisions 8-29
  Definitions 8-29
  Object Identifiers 8-29
  Textual Conventions 8-29
  Groups in MIB-II 8-29
  Historical 8-30
  System Group 8-30
  Interfaces Group 8-32
    Interfaces Table 8-32
  Address Translation Group 8-37
  IP Group 8-39
    IP Address Table 8-43
    IP Routing Table 8-45
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Address Translation Table</td>
<td>8-49</td>
</tr>
<tr>
<td>Additional IP Objects</td>
<td>8-50</td>
</tr>
<tr>
<td>ICMP Group</td>
<td>8-50</td>
</tr>
<tr>
<td>TCP Group</td>
<td>8-55</td>
</tr>
<tr>
<td>TCP Connection Table</td>
<td>8-58</td>
</tr>
<tr>
<td>Additional TCP Objects</td>
<td>8-60</td>
</tr>
<tr>
<td>UDP Group</td>
<td>8-60</td>
</tr>
<tr>
<td>UDP Listener Table</td>
<td>8-61</td>
</tr>
<tr>
<td>EGP Group</td>
<td>8-62</td>
</tr>
<tr>
<td>EGP Neighbor Table</td>
<td>8-63</td>
</tr>
<tr>
<td>Additional EGP Objects</td>
<td>8-67</td>
</tr>
<tr>
<td>Transmission Group</td>
<td>8-67</td>
</tr>
<tr>
<td>SNMP Group</td>
<td>8-67</td>
</tr>
<tr>
<td>HOST-RESOURCES-MIB</td>
<td>8-73</td>
</tr>
<tr>
<td>Revisions</td>
<td>8-75</td>
</tr>
<tr>
<td>Definitions</td>
<td>8-76</td>
</tr>
<tr>
<td>Object Identifiers</td>
<td>8-76</td>
</tr>
<tr>
<td>Textual Conventions</td>
<td>8-76</td>
</tr>
<tr>
<td>Host Resources System Group</td>
<td>8-77</td>
</tr>
<tr>
<td>Host Resources Storage Group</td>
<td>8-79</td>
</tr>
<tr>
<td>Host Resources Device Group</td>
<td>8-81</td>
</tr>
<tr>
<td>File System Table</td>
<td>8-90</td>
</tr>
<tr>
<td>Host Resources Running Software Group</td>
<td>8-92</td>
</tr>
<tr>
<td>Host Resources Running Software Performance Group</td>
<td>8-95</td>
</tr>
<tr>
<td>Host Resources Installed Software Group</td>
<td>8-96</td>
</tr>
<tr>
<td>Conformance Information</td>
<td>8-98</td>
</tr>
<tr>
<td>Compliance Statements</td>
<td>8-98</td>
</tr>
<tr>
<td>Cisco Unified CM Release 6.x Feature Services</td>
<td>8-100</td>
</tr>
<tr>
<td>Cisco Unified CM Release 6.x Network Services</td>
<td>8-102</td>
</tr>
<tr>
<td>Troubleshooting</td>
<td>8-103</td>
</tr>
<tr>
<td>Frequently Asked Questions</td>
<td>8-104</td>
</tr>
<tr>
<td>IF-MIB</td>
<td>8-105</td>
</tr>
<tr>
<td>Revisions</td>
<td>8-106</td>
</tr>
<tr>
<td>Definitions</td>
<td>8-107</td>
</tr>
<tr>
<td>Objects</td>
<td>8-107</td>
</tr>
<tr>
<td>Textual Conventions</td>
<td>8-107</td>
</tr>
<tr>
<td>Interface Index</td>
<td>8-108</td>
</tr>
<tr>
<td>Interfaces Table</td>
<td>8-109</td>
</tr>
<tr>
<td>Extension to the Interface Table</td>
<td>8-115</td>
</tr>
<tr>
<td>High Capacity Counter Objects</td>
<td>8-117</td>
</tr>
</tbody>
</table>
Interface Stack Group 8-121
Generic Receive Address Table 8-123
Definition of Interface-Related Traps 8-124
Conformance Information 8-125
Compliance Statements 8-125
Units of Conformance 8-127
Deprecated Definitions - Objects 8-129
   Interface Test Table 8-129
Deprecated Definitions - Groups 8-133
Deprecated Definitions - Compliance 8-134

CHAPTER 9
Vendor-Specific Management Information Base 9-1
Vendor-Specific Management Information Base 9-1
Supported Servers in Cisco Unified CM Releases 9-1
   Cisco Unified CM Release 8.5(1)  9-2
      Inapplicable MIBs in Cisco Unified CM Release 8.5(1)  9-2
   Cisco Unified CM Release 8.0(2)  9-4
      Inapplicable MIBs in Cisco Unified CM Release 8.0(2)  9-4
   Cisco Unified CM Release 8.0(1)  9-5
      Inapplicable MIBs in Cisco Unified CM Release 8.0(1)  9-6
   Cisco Unified CM Release 7.1(2)  9-8
      Inapplicable MIBs in Cisco Unified CM Release 7.1(2)  9-9
   Cisco Unified CM Release 7.1(1)  9-9
      Inapplicable MIBs  9-10
   Cisco Unified CM Release 7.0(1)  9-11
      Unsupported Servers by MIB  9-11
   Cisco Unified CM Release 6.1(3)  9-12
      Unsupported Servers by MIB  9-13
   Cisco Unified CM Release 6.1  9-14
      Unsupported Servers by MIB  9-14
   Cisco Unified CM Release 6.0  9-15
      Unsupported Servers by MIB  9-16
IBM MIBs  9-17
   IBM Status Messages  9-18
Hewlett Packard MIBs  9-20
   HP Status Messages  9-21
Intel MIBs  9-26
   Intel Status Messages  9-26
Preface

This chapter describes the purpose, audience, organization, and conventions of this document. It contains the following sections:

- Purpose, page xxvii
- Audience, page xxvii
- Organization, page xxviii
- Related Documentation, page xxviii
- Conventions, page xxix
- Obtaining Documentation and Submitting a Service Request, page xxx
- Cisco Product Security Overview, page xxx

Purpose

This document gives an overview of Cisco Unified Communications Manager (formerly Cisco Unified CallManager), deployment models, and related Management Information Bases (MIBs). It also explains sylogs, alerts, and alarms for the managed services that Service Providers implement in their networks. This document outlines basic concepts including Simple Network Management Protocol (SNMP) and the features of Cisco Unified Serviceability including Real-Time Monitoring Tool (RTMT).

Audience

This document provides information for administrators who install, upgrade, and maintain a service provider network. You need to have an understanding of Cisco Unified Communications Manager and Cisco Unified Communications Manager Business Edition 5000. See the “Related Documentation” section on page xxviii for Cisco Unified Communications Manager documents and other related technologies.
Organization

The following table provides an outline of the chapters in this document.

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chapter 1, “Overview”</td>
<td>Describes concepts with which you need to be familiar to implement SNMP, MIBs, and serviceability features.</td>
</tr>
<tr>
<td>Chapter 2, “New and Changed Information”</td>
<td>Describes the new and changed information in Cisco Unified Communications Manager releases.</td>
</tr>
<tr>
<td>Chapter 3, “Managing and Monitoring the Health of Cisco Unified Communications Manager Systems”</td>
<td>Describes methods for managing and monitoring the Cisco Unified Communications Manager servers.</td>
</tr>
<tr>
<td>Chapter 4, “Simple Network Management Protocol”</td>
<td>Describes the versions of SNMP and provides some troubleshooting tips.</td>
</tr>
<tr>
<td>Chapter 5, “Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts”</td>
<td>Describes the Cisco Unified Real-Time Monitoring Tool, default alarms, PerfMon counters, trace collection and other tools for troubleshooting.</td>
</tr>
<tr>
<td>Chapter 7, “Cisco Management Information Base”</td>
<td>Describes Cisco MIBs and the functionality of each with troubleshooting tips.</td>
</tr>
<tr>
<td>Chapter 8, “Industry-Standard Management Information Base”</td>
<td>Describes industry-standard MIBs including the functionality of each with troubleshooting tips.</td>
</tr>
<tr>
<td>Chapter 9, “Vendor-Specific Management Information Base”</td>
<td>Describes vendor-specific MIBs including the functionality of each with troubleshooting tips.</td>
</tr>
</tbody>
</table>

Related Documentation

This section lists documents that provide information on Cisco Unified Communications Manager, Cisco Unified IP Phones, and Cisco Unified Serviceability. Find the index to the documents at http://www.cisco.com/en/US/products/sw/voicesw/ps556/prod_maintenance_guides_list.html

- Cisco Unified Communications Manager—A suite of documents that relate to the installation and configuration of Cisco Unified Communications Manager. Refer to the Cisco Unified Communications Manager Documentation Guide for a list of documents on installing and configuring Cisco Unified Communications Manager including:
  - Cisco Unified Communications Manager Administration Guide
  - Cisco Unified Communications Manager System Guide
  - Cisco Unified Communications Manager Features and Services Guide

- Cisco Unified IP Phones and Services—A suite of documents that relate to the installation and configuration of Cisco Unified IP Phones.

- Cisco Unified Serviceability—A suite of documents that relate to the maintenance of managed services within Cisco Unified Serviceability. Refer to the Cisco Unified Communications Manager Documentation Guide for a complete list of documents including:
  - Cisco Unified Serviceability Administration Guide
- Cisco Unified Communications Manager Call Detail Records Administration Guide
- Cisco Unified Communications Manager CDR Analysis and Reporting Administration Guide
- Cisco Unified Real-Time Monitoring Tool Administration Guide
- Cisco Unified Reporting Administration Guide
- Command Line Interface Reference Guide for Cisco Unified Communications Solutions
- Disaster Recovery System Administration Guide for Cisco Unified Communications Manager

### Conventions

This document uses the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong> font</td>
<td>Commands and keywords are in <strong>boldface</strong>.</td>
</tr>
<tr>
<td>italic font</td>
<td>Arguments for which you supply values are in <em>italics</em>.</td>
</tr>
<tr>
<td>[ ]</td>
<td>Elements in square brackets are optional.</td>
</tr>
<tr>
<td>{ x | y | z }</td>
<td>Alternative keywords are grouped in braces and separated by vertical bars.</td>
</tr>
<tr>
<td>[ x | y | z ]</td>
<td>Optional alternative keywords are grouped in brackets and separated by vertical bars.</td>
</tr>
<tr>
<td>string</td>
<td>A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.</td>
</tr>
<tr>
<td><strong>screen</strong> font</td>
<td>Terminal sessions and information the system displays are in <strong>screen</strong> font.</td>
</tr>
<tr>
<td><strong>boldface</strong> <strong>screen</strong> font</td>
<td>Information you must enter is in <strong>boldface</strong> <strong>screen</strong> font.</td>
</tr>
<tr>
<td>italic <strong>screen</strong> font</td>
<td>Arguments for which you supply values are in <em>italic</em>* <strong>screen</strong> font.</td>
</tr>
<tr>
<td>→</td>
<td>This pointer highlights an important line of text in an example.</td>
</tr>
<tr>
<td>^</td>
<td>The symbol ^ represents the key labeled Control—for example, the key combination ^D in a screen display means hold down the Control key while you press the D key.</td>
</tr>
<tr>
<td>&lt; &gt;</td>
<td>Nonprinting characters, such as passwords are in angle brackets.</td>
</tr>
</tbody>
</table>

Notes use the following conventions:

**Note**

Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the publication.

Timesavers use the following conventions:
Timesaver

Means the described action saves time. You can save time by performing the action described in the paragraph.

Tips use the following conventions:

Tip

Means the following are useful tips.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly What's New in Cisco Product Documentation, which also lists all new and revised Cisco technical documentation, at:


Subscribe to the What's New in Cisco Product Documentation as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS Version 2.0.

Cisco Product Security Overview

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.


If you require further assistance please contact us by sending e-mail to export@cisco.com.
Related Documentation

You can browse the related documentation for Cisco Unified Communications Manager by clicking one of the following links:

- All technical documentation for Cisco Unified Communications Manager [here](#).
- The *Cisco Unified Communications Manager Documentation Guide* for your release [here](#).

You can also use the Documentation Custom Search utility to search through the documentation for this product.

To browse the documentation for another Cisco product, search or navigate from [here](#).

*Cisco Unified Communications Manager Documentation Guide*

The *Cisco Unified Communications Manager Documentation Guide* describes the various documents that comprise the Cisco Unified Communications Manager documentation set. The guide contains hyperlinks that link directly to these documents.
Overview

This chapter gives a conceptual overview of Cisco Unified Communications Manager (Cisco Unified CM) and Cisco Unified CM Business Edition 5000, possible deployment models, Simple Network Management Protocol (SNMP) including traps, Management Information Bases (MIBs), syslogs, and alerts/alarms. It contains the following sections:

- Cisco Unified Communications Manager, page 1-1
- Supported Deployment Models, page 1-2
- Managed Services, page 1-3
- Cisco Unified Serviceability, page 1-4
- Cisco Unified Real-Time Monitoring Tool, page 1-6
- Call Detail Records and Call Management Records, page 1-7
- Call Detail Record Analysis and Reporting, page 1-7
- Management Information Base, page 1-8

Cisco Unified Communications Manager

The Cisco Unified CM serves as the software-based call-processing component of the Cisco Unified Communications family of products. A wide range of Cisco Media Convergence Servers provides high-availability server platforms for Cisco Unified Communications Manager call processing, services, and applications.

The Cisco Unified CM system extends enterprise telephony features and functions to packet telephony network devices such as IP phones, media processing devices, voice-over-IP (VoIP) gateways, and multimedia applications. Additional data, voice, and video services, such as unified messaging, multimedia conferencing, collaborative contact centers, and interactive multimedia response systems, interact through Cisco Unified CM open telephony application programming interface (API).

Cisco Unified CM provides signaling and call control services to Cisco integrated telephony applications as well as third-party applications. Cisco Unified CM performs the following primary functions—

- Call processing
- Signaling and device control
- Dial plan administration
- Phone feature administration
Supported Deployment Models

Three types of Cisco Unified CM supported deployments exist—Single site, multisite WAN with centralized call processing, and multisite WAN with distributed call processing. The following paragraphs describe each of these:

- **Single Site**—Consists of a call processing agent cluster that is located at a single site, or campus, with no telephony services that are provided over an IP WAN. An enterprise would typically deploy the single-site model over a LAN or metropolitan area network (MAN), which carries the voice traffic within the site. In this model, calls beyond the LAN or MAN use the public switched telephone network (PSTN).

- **Multisite WAN with Centralized Call Processing**—Consists of a single call processing agent cluster that provides services for many remote sites and uses the IP WAN to transport Cisco Unified Communications traffic between the sites. The IP WAN also carries call control signaling between the central site and the remote sites.

- **Multisite WAN with Distributed Call Processing**—Consists of multiple independent sites, each with its own call processing agent cluster that is connected to an IP WAN that carries voice traffic between the distributed sites.

Cisco Unified CM BE 5000 supports three main types of deployment models—Single-site, multisite WAN with centralized call processing, and multisite WAN deployment with distributed call processing. Cisco Unified CM BE 5000 is a single-platform deployment, running both Cisco Unified CM and Cisco Unity Connection on the same server. Each type is described in the following paragraphs:

- **Single-Site**—Consists of Cisco Unified CM and Cisco Unity Connection running on the same hardware platform located at a single site or campus, with no telephony services provided over an IP WAN.

- **Multisite WAN with Centralized Call Processing**—Consists of a single call processing appliance that provides services for up to 20 sites (one central site and 19 remote sites), and this model uses the IP WAN to transport IP telephony traffic between the sites. The IP WAN also carries call control signaling between the central site and the remote sites.

- **Multisite WAN with Distributed Call Processing**—Consists of independent sites, each with its own call processing agent connected to an IP WAN that carries voice traffic between the distributed sites. The multisite WAN deployment with distributed call processing enables Cisco Unified CM BE 5000 to operate with Cisco Unified CM or other Cisco Unified CM BE 5000 deployments. With this model, Cisco Unified CM BE 5000 supports the use of H.323 intercluster trunks as well as SIP trunks to interconnect with Cisco Unified CM deployments or other Cisco Unified CM BE 5000 deployments. Each site can be a single site with its own call processing agent, a centralized call processing site and all of its associated remote sites, or a legacy PBX with Voice over IP (VoIP) gateway.
Managed Services

Two general types of managed services exist:

- Basic services that provide connectivity to the network—Routing, Domain Name System (DNS), and quality of service (QoS).
- High-valued services that the Service Provider offers to its customers—Videoconferencing, mobile IP, VPNs, VoIP, and Wireless. The high-valued services use the basic services as a backbone.

The service provider may require these server types and services:

- Web server with the ability to display web pages, even during high usage hours, to meet the demands of customers. The web pages get used to pay bills, check minutes of usage in the case of a cell phone, and buy new products. The web server and application server work together to display information that the service provider customer requires.
- Dedicated application server with the ability to advise customers when a product is out of stock, when bill is past due, or when need arises to buy more minutes.
- Mail server with the ability to notify customers to confirm an order or send a receipt for purchases.
- Secure gateway with VPN with the ability to have secure communications between the service provider and its customers and suppliers.

Be aware that any one of these services is critical to the operations of a service provider. Managing these services to ensure continuous operation requires a system that monitors fault, configuration, performance and security across all of the network elements. The introduction of element-to-element synchronization and the issues of using different vendor products complicates the task.

Cisco Unified Serviceability and SNMP attempt to address some of these network management issues:

- Are infrastructure elements functioning? If not, which are failing?
- What cause the failure? For example, recent configuration changes.
- What is the impact of the failure on the network as a whole and the impact on the elements within the network?
- What is the impact of the failure on services and customers?
- How long to correct the failure?
- Are there backup facilities?
- Are there any pending failures?
- How many packets were sent and received on a particular device? How many web pages were accessed.
- How were other devices used—how often and how long?

Cisco Unified CM supports SNMP v1, v2, and v3. SNMP remotely monitors, configures, and controls networks. SNMP sends fault messages to assigned managers as SNMP trap or inform request Protocol Data Units (PDUs). For more information, see Chapter 4, “Simple Network Management Protocol.”

Cisco Unified Serviceability, a component of Cisco Unified CM Administration includes its own set of error messages and alarms. Both applications use Management Information Base (MIB) text files to manage alarms and alerts, notifications, and error messages. For more information, see Chapter 6, “Cisco Unified Serviceability Alarms and CiscoLog Messages.”
**Cisco Unified Serviceability**

Cisco Unified Serviceability, a web-based troubleshooting tool, enables the following functions:

- Saves alarms and events for troubleshooting and provides alarm definitions.
- Saves trace information to various log files for troubleshooting.
- Monitors real-time behavior of components by using the Cisco Unified Real-Time Monitoring Tool (RTMT).
- Provides feature services that you can activate, deactivate, and view through the Service Activation window.
- Provides an interface for starting and stopping feature and network services.
- Generates and archives daily reports; for example, alert summary or server statistic reports.
- Allows Cisco Unified Communications Manager to work as a managed device for SNMP remote management and troubleshooting.
- Monitors the disk usage of the log partition on a server.
- Monitors the number of threads and processes in the system; uses cache to enhance the performance.

For information about configuring service parameters, refer to the *Cisco Unified Communications Manager Administration Guide*. For information about configuring Serviceability features, refer to the *Cisco Unified Serviceability Administration Guide*.

This section contains the following topics:

- **Trace Tools**, page 1-4
- **Troubleshooting Trace**, page 1-5
- **Trace Collection**, page 1-5
- **Cisco Unified Reporting**, page 1-5

**Trace Tools**

Trace tools assist you in troubleshooting issues with your voice application. Cisco Unified Serviceability supports SDI (System Diagnostic Interface) trace, SDL (Signaling Distribution Layer) trace for Cisco CallManager and Cisco CTIManager services, and Log4J trace for Java applications.

You use the Trace Configuration window to specify the level of information that you want traced as well the type of information that you want to be included in each trace file. If the service is a call-processing application such as Cisco CallManager or Cisco CTIManager, you can configure a trace on devices such as phones and gateway.

In the Alarm Configuration window, you can direct alarms to various locations, including SDI trace log files or SDL trace log files. If you want to do so, you can configure trace for alerts in the RTMT. After you have configured information that you want to include in the trace files for the various services, you can collect and view trace files by using the trace and log central option in the RTMT.
Troubleshooting Trace

The Troubleshooting Trace Settings window allows you to choose the services in Cisco Unified Serviceability for which you want to set predetermined troubleshooting trace settings. In this window, you can choose a single service or multiple services and change the trace settings for those services to the predetermined trace settings.

If you have clusters (Cisco Unified Communications Manager only), you can choose the services on different Cisco Unified Communications Manager servers in the cluster, so the trace settings of the chosen services get changed to the predetermined trace settings. You can choose specific activated services for a single server, all activated services for the server, specific activated services for all servers in the cluster, or all activated services for all servers in the cluster. In the window, N/A displays next to inactive services.

When you open the Troubleshooting Trace Settings window after you apply troubleshooting trace settings to a service, the service that you set for troubleshooting displays as checked. In the Troubleshooting Trace Settings window, you can reset the trace settings to the original settings.

After you apply Troubleshooting Trace Setting to a service, the Trace Configuration window displays a message that troubleshooting trace is set for the given service(s). From the Related Links drop-down list box, you can choose the Troubleshooting Trace Settings option if you want to reset the settings for the service. For the given service, the Trace Configuration window displays all the settings as read-only, except for some parameters of trace output settings; for example, Maximum No. of Files. You can modify these parameters even after you apply troubleshooting trace settings.

Trace Collection

Use Trace and Log Central, an option in the RTMT, to collect, view, and zip various service traces and/or other log files. With the Trace and Log Central option, you can collect SDL/SDI traces, Application Logs, System Logs (such as Event View Application, Security, and System logs), and crash dump files.

For more information on trace collection, refer to the Cisco Unified Real-Time Monitoring Tool Administration Guide.

Cisco Unified Reporting

Cisco Unified Reporting web application, which is accessed at the Cisco Unified Communications Manager console, generates reports for troubleshooting or inspecting cluster data. This tool provides a snapshot of cluster data without requiring multiple steps to find the data. The tool design facilitates gathering data from existing sources, comparing the data, and reporting irregularities.

A report combines data from one or more sources on one or more servers into one output view. For example, you can view a report that shows the hosts file for all servers in the cluster. The application gathers information from the publisher server and each subscriber server. A report provides data for all active cluster nodes that are accessible at the time that the report is generated.

Some reports run checks to identify conditions that could impact cluster operations. Status messages indicate the outcome of every data check that is run.

Only authorized users can access the Cisco Unified Reporting application. By default, this includes administrator users in the Standard Cisco Unified CM Super Users group. As an authorized user, you can view reports, generate new reports, or download reports at the graphical user interface (GUI).
Administrator users in the Standard Cisco Unified CM Super Users group can access all administrative applications in the Cisco Unified Communications Manager Administration navigation menu, including Cisco Unified Reporting, with a single sign onto one of the applications.

Cisco Unified Reporting includes the following capabilities:
- A user interface for generating, archiving, and downloading reports
- Notification message if a report will take excessive time to generate or consume excessive CPU

Generated reports in Cisco Unified Reporting may use any of the following data sources:
- RTMT counters
- CDR CAR
- Cisco Unified CM DB
- Disk files
- Operating System API calls
- Network API calls
- Prefs (Windows registry)
- CLI
- RIS

Cisco Unified Real-Time Monitoring Tool

Real-Time Monitoring Tool (RTMT) is a client-side application that uses HTTPS and TCP to monitor system performance, device status, device discovery, CTI applications, and voice messaging ports. RTMT can connect directly to devices by using HTTPS to troubleshoot system issues. RTMT performs the following tasks:
- Monitor a set of predefined management objects that monitor the health of the system.
- Generate various alerts, in the form of e-mails, for objects when values go over/below user-configured thresholds.
- Collect and view traces in various default viewers that exist in RTMT.
- Translate Q931 messages.
- View syslog messages in SysLog Viewer.
- Work with performance-monitoring counters.

In addition to SNMP traps, RTMT can monitor and parse syslog messages that are provided by the hardware vendors, and then send these alerts to RTMT Alert Central. You can configure RTMT to notify the Cisco Unified CM system administrator if and when the alerts occur. You can configure the notifications for e-mail or Epage or both.

For more information, refer to Cisco Unified Real-Time Monitoring Tool Administration Guide.
Call Detail Records and Call Management Records

Call Detail Records (CDRs) and Call Management Records (CMRs) get used for post-processing activities such as generating billing records and network analysis. When you install your system, the system enables CDRs by default. CMRs remain disabled by default. You can enable or disable CDRs or CMRs at any time that the system is in operation.

The CDR Management (CDRM) feature, a background application, supports the following capabilities:

• Collects the CDR/CMR files from the Cisco Unified Communications Manager server or node to the CDR Repository server or node.
• Collects and maintains the CDR/CMR files on the server where you configure CAR.
• Maintains the CDR/CMR files on the CDR Repository node or CDR server.
• Allows third-party applications to retrieve CDR/CMR files on demand through a SOAP interface.
• Accepts on-demand requests for searching file names.
• Pushes CDR/CMR files from individual nodes within a cluster to the CDR Repository server or node.
• Sends CDR/CMR files to up to three customer billing servers via FTP/SFTP.
• Monitors disk usage of CDR/CMR files on the server where you configure CAR or on the CDR Repository server or node.
• Periodically deletes CDR/CMR files that were successfully delivered. You can configure the amount of storage that is used to store flat files. Predefined storage limits exist. If the storage limits are exceeded, the CDR Repository Manager deletes old files to reduce the disk usage to the preconfigured low water mark. The post-processing applications can later retrieve the buffered historical data to re-get any lost, corrupted, or missing data. The CDRM feature, which is not aware of the flat file format, does not manipulate the file contents.

CDRM includes two default services, the CDR Agent and the CDR Repository Manager, and one activate service, CDR onDemand Service.

For more information, refer to the Cisco Unified Communications Manager Call Detail Records Administration Guide.

Call Detail Record Analysis and Reporting

Cisco Unified Serviceability supports Call Detail Record (CDR) Analysis and Reporting (CAR) and is available in the Tools menu. CAR generates reports for Quality of Service (QoS), traffic, and billing information. For its primary function, CAR generates reports about the users of Cisco Unified Communications Manager and reports on system status with respect to call processing. CAR also performs CAR database management activities. You can perform these tasks in one of the following ways:

• Automatically configure the required tasks to take place.
• Manually perform the tasks by using the web interface.

CAR processes the CDRs from flat files that the CDR repository service places in the repository folder structure. CAR processes CDRs at a scheduled time and frequency. By default, CDR data loads continuously 24 hours per day and 7 days per week; however, you can set the loading time, interval, and duration as needed. In addition, the default setting loads only CDR records. CMR records do not get loaded by default.
CAR provides e-mail alerts for various events, including the following events:

- Charge Limit Notification indicates when the daily charge limit for a user exceeds the specified maximum.
- QoS Notification indicates when the percentage of good calls drops below a specified range or the percentage of poor calls exceeds a specified limit.

For more information, refer to the *Cisco Unified Communications Manager CDR Analysis and Reporting Administration Guide*.

## Management Information Base

The Management Information Base (MIB) converts object identifiers (OIDs) that are numerical strings into an ASCII text file. The OIDs identify data objects. The OID represents specific characteristics of a device or application and can have one or more object instances (variables). Managed objects, alarms, notifications, and other valuable information get identified by the OID and get listed in the MIB.

The OID gets logically represented in a tree hierarchy. The root of the tree stays unnamed and splits into three main branches—Consultative Committee for International Telegraph and Telephone (CCITT), International Organization for Standardization (ISO), and joint ISO/CCITT.

These branches and those that fall below each category have short text strings and integers to identify them. Text strings describe object names, while integers allow computer software to create compact, encoded representations of the names. For example, the Cisco MIB variable authAddr represents an object name and gets denoted by the number 5, which is listed at the end of OID 1.3.6.1.4.1.9.2.1.5.

The OID in the Internet MIB hierarchy represents the sequence of numeric labels on the nodes along a path from the root to the object. The OID 1.3.6.1.2.1 represents the Internet standard MIB. It also can get expressed as iso.org.dod.internet.mgmt.mib.

The Cisco MIB set comprises a collection of variables that are private extensions to the Internet standard MIB II and many other Internet standard MIBs. RFC 1213, *Management Information Base for Network Management of TCP/IP-based Internets—MIB-II* documents MIB II.

Cisco Unified CM and Cisco Unified CM BE 5000 support the following MIBs:

- CISCO-CCM-MIB
- CISCO-CCM-CAPABILITY
- CISCO-CDP-MIB
- CISCO-SYSLOG-MIB
- HOST-RESOURCES-MIB
- MIB-II
- SYSAPPL-MIB
- Vendor-specific MIBs

For descriptions of the supported MIBs, see the following chapters:

- Chapter 7, “Cisco Management Information Base”
- Chapter 8, “Industry-Standard Management Information Base”
- Chapter 9, “Vendor-Specific Management Information Base”
New and Changed Information

This chapter describes the new and changed information in Cisco Unified Communications Manager (Cisco Unified CM) for Release 8.0(x). It contains the following sections:

- Cisco Unified Communications Manager, Release 8.5(1), page 2-1
- Cisco Unified Communications Manager, Release 8.0(2), page 2-24
- Cisco Unified Communications Manager, Release 8.0(1), page 2-25


Cisco Unified Communications Manager, Release 8.5(1)

This section describes the new and changed information in Cisco Unified Communications Manager, Release 8.5(1). It contains the following subsections:

- Audit Log Support for Cisco Unity Connection, page 2-2
- Alarm Additions and Changes, page 2-3
- Enhanced Reason Codes, page 2-4
- New Perfmon Counters for Cisco SIP Normalization, page 2-6
- SNMP MIBs, page 2-13
- Supported Servers, page 2-13
- Serviceability - Session Manager Edition (SME), page 2-14
- Default Settings for Alarm Configuration Settings, page 2-19
- New Cisco Unity Connection Alerts, page 2-19
- Logging On to CAR, page 2-20
- Configuring the Trunk, page 2-20
- Configuring Trunk Utilization Reports, page 2-21
- Cisco Dialed Number Analyzer Server, page 2-23
Audit Log Support for Cisco Unity Connection

With audit logging, configuration changes to the Cisco Unity Connection system gets logged in separate log files for auditing.

The following components generate audit events for Cisco Unity Connection:

- Command-Line Interface, page 2-2
- Cisco Unity Connection Administration, page 2-2
- Cisco Personal Communications Assistant (Cisco PCA), page 2-2
- Cisco Unity Connection Serviceability, page 2-2
- Cisco Unity Connection Clients that Use the Representational State Transfer APIs, page 2-2

**Command-Line Interface**

All commands issued via the command-line interface are logged (for both Cisco Unified Communications Manager and Cisco Unity Connection).

**Cisco Unity Connection Administration**

Cisco Unity Connection Administration logs the following events:
- User logging (user logins and user logouts).
- All configuration changes, including but not limited to users, contacts, call management objects, networking, system settings, and telephony.
- Task management (enabling or disabling a task).
- Bulk Administration Tool (bulk creates, bulk deletes).
- Custom Keypad Map (map updates)

**Cisco Personal Communications Assistant (Cisco PCA)**

The Cisco Personal Communications Assistant client logs the following events:
- User logging (user logins and user logouts).
- All configuration changes made via the Messaging Assistant.

**Cisco Unity Connection Serviceability**

Cisco Unity Connection Serviceability logs the following events:
- User logging (user logins and user logouts).
- All configuration changes.
- Activating, deactivating, starting or stopping services.

**Cisco Unity Connection Clients that Use the Representational State Transfer APIs**

Cisco Unity Connection clients that use the Representational State Transfer (REST) APIs log the following events:
- User logging (user API authentication).
- API calls that utilize Cisco Unity Connection Provisioning Interface (CUPI).

For Cisco Unity Connection, the application administration account that was created during installation has the Audit Administrator role and can assign other administrative users to the role. You can also remove the Audit Administrator role from this account.
The Audit Administrator role in Cisco Unity Connection provides the ability to view, download and delete audit logs in Cisco Unified Cisco Unified Real-Time Monitoring Tool.

For information on roles and users in Cisco Unity Connection, refer to the User Moves, Adds, and Changes Guide for Cisco Unity Connection.

For a description of the settings that you can configure for audit log configuration, refer to the Cisco Unified Serviceability Administration Guide.

**Alarm Additions and Changes**

**OPTIONS ping**
Cisco Unified Communications Manager SIP OPTIONS allows a SIP trunk to track the status of remote destinations. The following new alarms are generated for OPTIONS ping:

- SIPTrunkISV
- SIPTrunkOOS
- SIPTrunkPartiallyISV

**SIP Normalization and Transparency**
Cisco Unified Communications Manager identifies the usage of and errors with SIP normalization scripts; that is, when the script gets opened and closed as well as when errors and resource warnings occur.

The following new alarms are generated for SIP Normalization and Transparency:

- SIPNormalizationScriptOpened
- SIPNormalizationScriptClosed
- SIPNormalizationResourceWarning
- SIPNormalizationScriptError
- SIPNormalizationAutoResetDisabled

**Single Sign On and SmartCard Authentication**

The parameters are modified for the following existing alarms:

- authLdapInactive
- authFail
- authSuccess

The following new alarms are generated for Single Sign On and SmartCard Authentication:

- LDAPServerUnreachable
- SSODisabled
- SSONullTicket
- SSOServerUnreachable
- SSOUserNotInDB

For more information on alarms, see Cisco Unified Serviceability Alarms and CiscoLog Messages, page 6-1.
## Enhanced Reason Codes

The following provides the reason codes that are added for EndPointTransientConnection alarm:

<table>
<thead>
<tr>
<th>Reason Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>maxDevRegExceeded</td>
<td>Maximum number of device registrations have been reached.</td>
</tr>
<tr>
<td>DeviceInitiatedReset</td>
<td>Indicates that the error was due to device initiated reset.</td>
</tr>
<tr>
<td>CallManagerReset</td>
<td>Indicates that the error was due to call manager reset.</td>
</tr>
<tr>
<td>DirectoryNumberMismatch</td>
<td>Indicates mismatch between the directory number that the SIP device is trying to register with and the directory number that is configured in the Cisco Unified CM for the SIP device.</td>
</tr>
<tr>
<td>DatabaseTimeout</td>
<td>Cisco Unified CM requested device configuration data from the database, but did not receive a response within 10 minutes.</td>
</tr>
<tr>
<td>RegistrationSequenceError</td>
<td>(SCCP only) A device requested configuration information from the Cisco Unified CM at an unexpected time. The Cisco Unified CM had not yet obtained the requested information. The device will automatically attempt to register again. If this alarm occurs again, manually reset the device. If this alarm continues to occur after the manual reset, there may be an internal firmware error.</td>
</tr>
<tr>
<td>InvalidCapabilities</td>
<td>(SCCP only) The Cisco Unified CM detected an error in the media capabilities reported in the StationCapabilitiesRes message by the device during registration. The device will automatically attempt to register again. If this alarm occurs again, manually reset the device. If this alarm continues to occur after the manual reset, there may be a protocol error.</td>
</tr>
<tr>
<td>CapabilityResponseTimeout</td>
<td>(SCCP only) The Cisco Unified CM timed out while waiting for the device to respond to a request to report its media capabilities. Possible causes include device power outage, network power outage, network configuration error, network delay, packet drops, and packet corruption. It is also possible to get this error if the Cisco Unified CM node is experiencing high CPU usage. Verify that the device is powered up and operating. Verify that network connectivity exists between the device and Cisco Unified CM, and verify that the CPU utilization is in the safe range.</td>
</tr>
</tbody>
</table>
| SecurityMismatch          | Unified CM detected a mismatch in the security settings of the device and/or the Cisco Unified CM. The following mismatches are detected:  
  - The device established a secure connection, yet reported that it cannot do authenticated signaling.  
  - The device did not establish a secure connection, but the security mode configured for the device indicates that it should have done so.  
  - The device established a secure connection, but the security mode configured for the device indicates that it should not have done so |
| AutoRegisterDBError       | Auto-registration of a device failed for one of the following reasons:                                  |
|                           |  - Auto-registration is not allowed for the device type.                                               |
|                           |  - An error occurred in the auto-registration stored procedure.                                        |
| DBAccessError             | Device registration failed because of an error that occurred while building the station registration profile. This usually indicates a synchronization problem with the database. |
The following provides the reason codes that are added for `EndPointUnregistered` alarm:

<table>
<thead>
<tr>
<th>Reason Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AutoRegisterDBConfigTimeout</td>
<td>(SCCP only) Unified CM timed out during auto-registration of a device. The registration profile of the device did not get inserted into the database in time. The device will automatically attempt to register again.</td>
</tr>
<tr>
<td>DeviceTypeMismatch</td>
<td>The device type reported by the device does not match the device type configured on the Unified CM.</td>
</tr>
<tr>
<td>AddressingModeMismatch</td>
<td>(SCCP only) Cisco Unified CM detected an error related to the addressing mode configured for the device. One of the following errors was detected:</td>
</tr>
<tr>
<td></td>
<td>• The device is configured to use only IPv4 addressing, but did not specify an IPv4 address.</td>
</tr>
<tr>
<td></td>
<td>• The device is configured to use only IPv6 addressing, but did not specify an IPv6 address.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Reason Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NoEntryInDatabase</td>
<td>Device not configured properly in the Cisco Unified CM database.</td>
</tr>
<tr>
<td>DatabaseConfigurationError</td>
<td>Device configuration error in the Cisco Unified CM database.</td>
</tr>
<tr>
<td>DeviceNameUnresolvable</td>
<td>The Cisco Unified CM is unable to resolve the device name to an IP Address internally.</td>
</tr>
<tr>
<td>MaxDevRegExceeded</td>
<td>Maximum number of device registrations have been reached.</td>
</tr>
<tr>
<td>InitializationError</td>
<td>Indicates that an error occurred when the Cisco Unified CM tries to initialize the device.</td>
</tr>
<tr>
<td>PowerSavePlus</td>
<td>The device powered off as a result of the Power Save Plus feature that is enabled for this device. When the device powers off, it remains unregistered from Cisco Unified CM until the Phone On Time is defined in the Product Specific Configuration for this device.</td>
</tr>
<tr>
<td>CallManagerForcedRestart</td>
<td>(SIP Only) The device did not respond to an Apply Config request and as a result, Unified CM sent a restart request to the device. The device may be offline due to a power outage or network problem. Confirm that the device is powered-up and that network connectivity exists between the device and Cisco Unified CM.</td>
</tr>
<tr>
<td>SourceIPAddrChanged</td>
<td>(SIP Only) The device has been unregistered because the IP address in the Contact header of the REGISTER message has changed. The device will be automatically re-registered. No action is necessary.</td>
</tr>
<tr>
<td>SourcePortChanged</td>
<td>(SIP Only) The device has been unregistered because the port number in the Contact header of the REGISTER message has changed. The device will be automatically re-registered. No action is necessary.</td>
</tr>
<tr>
<td>RegistrationSequenceError</td>
<td>A device requested configuration information from the Unified CM at an unexpected time. The Unified CM no longer had the requested information in memory.</td>
</tr>
<tr>
<td>InvalidCapabilities</td>
<td>(SCCP only) Cisco Unified CM detected an error in the updated media capabilities reported by the device. The device reported the capabilities in one of the StationUpdateCapabilities message variants.</td>
</tr>
</tbody>
</table>
New Perfmon Counters for Cisco SIP Normalization

The Cisco SIP Normalization performance object contains counters that allow you to monitor aspects of the normalization script, including initialization errors, runtime errors, and script status. Each device that has an associated script causes a new instance of these counters to be created.

Table 2-1 describes the Cisco SIP Normalization counters.

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FallbackInitiated</td>
<td>The device has initiated a fallback and will automatically reregister to a higher-priority Cisco Unified CM. No action is necessary.</td>
</tr>
<tr>
<td>DeviceSwitch</td>
<td>A second instance of an endpoint with the same device name has registered and assumed control. No action is necessary.</td>
</tr>
</tbody>
</table>

Table 2-1Cisco SIP Normalization

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeviceResetAutomatically</td>
<td>This counter indicates the number of times that Cisco Unified CM automatically resets the device (SIP trunk). The device reset is based on the values that are specified in the Script Execution Error Recovery Action and System Resource Error Recovery Action fields on the SIP Normalization Script Configuration window in Cisco Unified Communications Manager Administration. When the device (SIP trunk) is reset due to script errors, the counter value increments. This count restarts when:</td>
</tr>
<tr>
<td></td>
<td>• The SIP trunk is deleted.</td>
</tr>
<tr>
<td></td>
<td>• The script on the trunk gets changed or deleted.</td>
</tr>
<tr>
<td></td>
<td>• Cisco Unified Communications Manager restarts.</td>
</tr>
<tr>
<td>DeviceResetManually</td>
<td>This counter indicates the number of times that the device (SIP trunk) is reset manually in Cisco Unified Communications Manager Administration or by other methods, such as AXL. When the device associated with a script is reset due to configuration changes, the counter value increments. The counter restarts when:</td>
</tr>
<tr>
<td></td>
<td>• The SIP trunk is deleted.</td>
</tr>
<tr>
<td></td>
<td>• The script on the trunk gets changed or deleted.</td>
</tr>
</tbody>
</table>
ErrorExecution This counter represents the number of execution errors that occurred while the script executed. Execution errors can occur while a message handler executes. Execution errors can be caused by resource errors, an argument mismatch in a function call, and so on.

When an execution error occurs, Cisco Unified CM performs the following actions:

- Automatically restores the message to the original content before applying additional error handling actions.
- Increments the value of the counter.
- Takes appropriate action based on the configuration of the Script Execution Error Recovery Action and System Resource Error Recovery Action fields in Cisco Unified Communications Manager Administration.

Check the SIPNormalizationScriptError alarm for details, including the line number in the script that failed. Correct the script problem, upload the corrected script as needed, and reset the trunk. This counter increments every time an execution error occurs. This counter provides a count from the most recent trunk reset that involved a script configuration change. (A device reset alone does not restart the count; the script configuration must also change before the reset occurs.) If the counter continues to increment after you fix the script problem, examine the script again.

ErrorInit This counter represents the number of times a script error occurred after the script successfully loaded into memory, but failed to initialize in Cisco Unified CM. A script can fail to initialize due to resource errors, an argument mismatch in a function call, the expected table was not returned, and so on.

Check the SIPNormalizationScriptError alarm for details, including the line number in the script that failed. Correct the script problem, upload the corrected script as needed, and reset the trunk. This counter increments every time an initialization error occurs. This counter provides a count from the most recent trunk reset that was accompanied by a script configuration change. (A device reset alone does not restart the count; the script configuration must also change before the reset occurs.) If the counter continues to increment after you fix the script problem, examine the script again. When the error occurs during initialization, Cisco Unified CM automatically disables the script.

ErrorInternal This counter indicates the number of internal errors that occurred while the script executed. Internal errors are very rare. If the value in this counter is higher than zero, a defect exists in the system that is not related to the script content or execution. Collect SDI traces and contact the Technical Assistance Center (TAC).
Table 2-1  Cisco SIP Normalization (continued)

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ErrorLoad</td>
<td>This counter represents the number of times a script error occurred when the script loaded into memory in Cisco Unified Communications Manager. A script can fail to load due to memory issues or syntax errors. Check the SIPNormalizationScriptError alarm for details. Check the script syntax for errors, upload the corrected script as needed, and reset the trunk. This counter increments every time a load error occurs. This counter provides a count from the most recent trunk reset that was accompanied by a script configuration change. (A device reset alone will not restart the count; the script configuration must also change before the reset occurs.) If the counter continues to increment even after you fix the script problem, examine the script again.</td>
</tr>
<tr>
<td>ErrorResource</td>
<td>This counter indicates whether the script encountered a resource error. Two kinds of resource errors exist: exceeding the value in the Memory Threshold field and exceeding the value in the Lua Instruction Threshold field. (Both fields display on the SIP Normalization Script Configuration window in Cisco Unified Communications Manager Administration.) If either condition occurs, Cisco Unified Communications Manager immediately closes the script and issues the SIPNormalizationScriptError alarm. If a resource error occurs while the script loads or initializes, the script is disabled. If a resource error occurs during execution, the configured system resource error recovery action is taken. (The setting of the System Resource Error Recovery Action field on the SIP Normalization Script Configuration window in Cisco Unified Communications Manager Administration defines this action.)</td>
</tr>
<tr>
<td>MemoryUsage</td>
<td>This counter specifies the amount of memory, in bytes, that the script consumes. This counter increases and decreases to match the amount of memory that the script uses. This count gets cleared when the script closes (because a closed script does not consume memory) and restarts when the script opens (gets enabled). A high number in this counter indicates a resource problem. Check the MemoryUsagePercentage counter and the SIPNormalizationResourceWarning alarm, which occur when the resource consumption exceeds an internally set threshold.</td>
</tr>
<tr>
<td>MemoryUsagePercentage</td>
<td>This counter specifies the percentage of the total amount of memory that the script consumes. The value in this counter is derived by dividing the value in the MemoryUsage counter by the value in the Memory Threshold field (in the SIP Normalization Script Configuration window) and multiplying the result by 100 to arrive at a percentage. This counter increases and decreases in accordance with the MemoryUsage counter. This count gets cleared when the script closes (because closed scripts do not consume memory) and restarts when the script opens (gets enabled). When this counter reaches the internally controlled resource threshold, the SIPNormalizationResourceWarning alarm is issued.</td>
</tr>
</tbody>
</table>
Table 2-1  Cisco SIP Normalization (continued)

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MessageRollback</td>
<td>This counter indicates the number of times that the system automatically rolled back a message. The system rolls back the message by using the error handling that is specified in the Script Execution Error Recovery Action field in the SIP Normalization Script Configuration window in Cisco Unified CM Administration. When an execution error occurs, Cisco Unified CM automatically restores the message to the original content before applying additional error handling actions. If error handling specifies rollback only, no further action is taken beyond rolling back to the original message before the normalization attempt. For the other possible Script Execution Error Recovery Actions, message rollback always occurs first, followed by the specified action, such as disabling the script, resetting the script automatically, or resetting the trunk automatically.</td>
</tr>
<tr>
<td>msgAddContentBody</td>
<td>This counter represents the number of times that the script added a content body to the message. If you are using the msg:addContentBody API in the script, this counter increases each time that the msg:addContentBody API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgAddHeader</td>
<td>This counter represents the number of times that the script added a SIP header to the message. If you are using the msg:addHeader API in the script, this counter increases each time that the msg:addHeader API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgAddHeaderUriParameter</td>
<td>This counter represents the number of times that the script added a SIP header URI parameter to a SIP header in the message. If you are using the msg:addHeaderUriParameter API in the script, this counter increases each time that the msg:addHeaderUriParameter API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgAddHeaderValueParameter</td>
<td>This counter represents the number of times that the script added a SIP header value parameter to a SIP header in the message. If you are using the msg:addHeaderValueParameter API in the script, this counter increases each time that the msg:addHeaderValueParameter API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgApplyNumberMask</td>
<td>This counter represents the number of times that the script applied a number mask to a SIP header in the message. If you are using the msg:applyNumberMask API in the script, this counter increases each time that the msg:applyNumberMask API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgBlock</td>
<td>This counter represents the number of times that the script blocked a message. If you are using the msg:block API in the script, this counter increases each time that the msg:block API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgConvertDiversionToHI</td>
<td>This counter represents the number of times that the script converted Diversion headers into History-Info headers in the message. If you are using the msg:convertDiversionToHI API in the script, this counter increases each time that the msg:convertDiversionToHI API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
</tbody>
</table>
### Table 2-1  Cisco SIP Normalization (continued)

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>msgConvertHIToDiversion</td>
<td>This counter represents the number of times that the script converted Diversion headers into History-Info headers in the message. If you are using the msg:convertDiversionToHI API in the script, this counter increases each time that the msg:convertDiversionToHI API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgModifyHeader</td>
<td>This counter represents the number of times that the script modified a SIP header in the message. If you are using the msg:modifyHeader API in the script, this counter increases each time that the msg:modifyHeader API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgRemoveContentBody</td>
<td>This counter represents the number of times that the script removed a content body from the message. If you are using the msg:removeContentBody API in the script, this counter increases each time that the msg:removeContentBody API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgRemoveHeader</td>
<td>This counter represents the number of times that the script removed a SIP header from the message. If you are using the msg:removeHeader API in the script, this counter increases each time that the msg:removeHeader API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgRemoveHeaderValue</td>
<td>This counter represents the number of times that the script removed a SIP header value from the message. If you are using the msg:removeHeaderValue API in the script, this counter increases each time that the msg:removeHeaderValue API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgSetRequestUri</td>
<td>This counter represents the number of times that the script modified the request URI in the message. If you are using the msg:setRequestUri API in the script, this counter increases each time that the msg:setRequestUri API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgSetResponseCode</td>
<td>This counter represents the number of times that the script modified the response code and/or response phrase in the message. If you are using the msg:setResponseCode API in the script, this counter increases each time that the msg:setResponseCode API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgSetSdp</td>
<td>This counter represents the number of times that the script set the SDP in the message. If you are using the msg:setSdp API in the script, this counter increases each time that the msg:setSdp API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>ptAddContentBody</td>
<td>This counter represents the number of times that the script added a content body to the PassThrough (pt) object. If you are using the pt:addContentBody API in the script, this counter increases each time that the pt:addContentBody API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>ptAddHeader</td>
<td>This counter represents the number of times that the script added a SIP header to the PassThrough (pt) object. If you are using the pt:addHeader API in the script, this counter increases each time that the pt:addHeader API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
</tbody>
</table>
Table 2-1  Cisco SIP Normalization (continued)

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ptAddHeaderUriParameter</td>
<td>This counter represents the number of times that the script added a SIP header URI parameter to the PassThrough (pt) object. If you are using the pt:addHeaderUriParameter API in the script, this counter increases each time that the pt:addHeaderUriParameter API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>ptAddHeaderValueParameter</td>
<td>This counter represents the number of times that the script added a SIP header value parameter to the PassThrough (pt) object. If you are using the pt:addHeaderValueParameter API in the script, this counter increases each time that the pt:addHeaderValueParameter API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>ptAddRequestUriParameter</td>
<td>This counter represents the number of times that the script added a request URI parameter to the PassThrough (pt) object. If you are using the pt:addRequestUriParameter API in the script, this counter increases each time that the pt:addRequestUriParameter API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>ScriptActive</td>
<td>This counter indicates whether the script is currently active (running on the trunk). The following values display for the counter:</td>
</tr>
<tr>
<td></td>
<td>• 0—Indicates that the script is closed (disabled).</td>
</tr>
<tr>
<td></td>
<td>• 1—Indicates that the script is open and operational.</td>
</tr>
<tr>
<td></td>
<td>To open the script that should be running on this trunk, perform the following actions:</td>
</tr>
<tr>
<td></td>
<td>1. Check for any alarms that might indicate why the script is not open.</td>
</tr>
<tr>
<td></td>
<td>2. Correct any errors.</td>
</tr>
<tr>
<td></td>
<td>3. Upload a new script if necessary.</td>
</tr>
<tr>
<td></td>
<td>4. Reset the trunk.</td>
</tr>
<tr>
<td>ScriptClosed</td>
<td>This counter indicates the number of times that Cisco Unified Communications Manager has closed the script.</td>
</tr>
<tr>
<td></td>
<td>When the script is closed, it is not enabled on this device.</td>
</tr>
<tr>
<td></td>
<td>Cisco Unified CM closes the script under one of the following conditions:</td>
</tr>
<tr>
<td></td>
<td>• The device was reset manually.</td>
</tr>
<tr>
<td></td>
<td>• The device was reset automatically (due to an error).</td>
</tr>
<tr>
<td></td>
<td>• The device was deleted.</td>
</tr>
<tr>
<td></td>
<td>This count restarts when the SIP trunk is reset after a change to the script configuration and when Cisco Unified CM restarts.</td>
</tr>
</tbody>
</table>
ScriptDisabledAutomatically

This counter indicates the number of times that the system automatically disabled the script. The values that are specified in the Script Execution Error Recovery Action and System Resource Error Recovery Action fields in the SIP Normalization Script Configuration window in Cisco Unified CM Administration determine whether the script is disabled. The script also gets disabled as a result of script error conditions that are encountered during loading and initialization. This counter provides a count from the most recent manual device reset that involved a script configuration change (a device reset alone does not restart the count; the script must also have changed before the reset occurs). This counter increments every time Cisco Unified CM automatically disables a script due to script errors.

If the number in this counter is higher than expected, perform the following actions:

• Check for SIPNormalizationScriptError alarm and SIPNormalizationAutoResetDisabled alarm.

• Check for any resource-related alarms and counters in RTMT to determine whether a resource issue is occurring.

• Check for any unexpected SIP normalization events in the SDI trace files.

ScriptOpened

This counter indicates the number of times that the Cisco Unified CM attempted to open the script. For the script to open, it must load into memory in Cisco Unified CM, initialize, and be operational. A number greater than one in this counter means that Cisco Unified CM has made more than one attempt to open the script on this SIP trunk, either for an expected reason or due to an error during loading or initialization. The error can occur due to execution errors or resource errors or invalid syntax in the script. Expect this counter to be greater than one if any of these counters increment: DeviceResetManually, DeviceResetAutomatically, or ScriptResetAutomatically. The DeviceResetManually counter increments when an expected event, such as a maintenance window on the SIP trunk, causes the script to close.

If the number in this counter is high for an unexpected reason, perform the following actions:

• Check for alarms, such as the SIPNormalizationScriptClosed, SIPNormalizationScriptError, or SIPNormalizationResourceWarning.

• Check resource-related alarms and counters in RTMT to determine whether a resource issue is occurring.

• Check for any unexpected SIP normalization events in the SDI trace files.

This count restarts when the SIP trunk resets after a script configuration change and when Cisco Unified CM restarts.
Table 2-1  Cisco SIP Normalization (continued)

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| ScriptResetAutomatically         | This counter indicates the number of times that the system automatically reset the script. The script resets based on the values that are specified in the Script Execution Error Recovery Action and System Resource Error Recovery Action fields in the SIP Normalization Script Configuration window in Cisco Unified CM Administration. This counter specifies a count of the number of automatic script resets after the last manual device reset; this counter increments every time the Cisco Unified CM automatically resets a script due to script errors. If the number in this counter is higher than expected, perform the following actions:  
  - Check for a SIPNormalizationScriptError alarm.  
  - Check for any resource-related alarms and counters in RTMT to determine whether a resource issue is occurring.  
  - Check for any unexpected SIP normalization events in the SDI trace files.                                                                 |

SNMP MIBs

The following TEXTUAL-CONVENTIONs are updated for 8.5(1) release:

- CcmDevUnregCauseCode
- CcmDevRegFailCauseCode

For more information, refer Cisco Management Information Base, page 7-1.

Supported Servers

The following IBM Server Models are supported for this release:

- MCS-7816-I3-IPC1
- MCS-7816-I4-IPC1
- MCS-7816-I5-IPC1
- MCS-7825-I3-IPC1
- MCS-7825-I4-IPC1
- MCS-7825-I5-IPC1
- MCS-7828-I3-SS1
- MCS-7828-I4-SS1
- MCS-7828-I5-SS1
- MCS-7835-I2-IPC1
- MCS-7835-I2-IPC2
- MCS-7835-I3-IPC1
- MCS-7845-I2-IPC1
- MCS-7845-I2-IPC2
- MCS-7845-I3-IPC1
The following HP Server Models are supported for this release:
- MCS-7816-H3-IPC1
- MCS-7825-H2-IPC1
- MCS-7825-H3-IPC1
- MCS-7825-H4-IPC1
- MCS-7828-H3-IPC1
- MCS-7835-H2-IPC1
- MCS-7835-H2-IPC2
- MCS-7845-H2-IPC1
- MCS-7845-H2-IPC2
- DL380G6 SW only

The following Cisco Unified Computing Systems are supported for this release:
- UCS B200 M1
- UCS C210 M1

For information on inapplicable MIBs for 8.5(1) release, refer Vendor-Specific Management Information Base, page 9-1.

Serviceability - Session Manager Edition (SME)

The Cisco Unified Communications Manager captures and logs all SIP message activities, which comprise the incoming and outgoing calls or sessions that pass through the Cisco Unified Communications Manager. The Cisco Unified Communications Manager stores the messages on a per-transaction basis in a new Call Log file, which can be downloaded through Cisco Unified Real-Time Monitoring Tool (RTMT) for post-processing activity.

RTMT users can search or trace the calls based on the following criteria:
- Calling Number/URI
- Called Number/URI
- Start Time
- Duration

RTMT downloads the Call Log file that includes the Start Time and Duration. The tool searches for the matching calls, lists the matching call records, and provides the SIP message Call Flow Diagram.

Before you Begin
Perform the following task:
- Use the enterprise parameter, Enable Call Trace Log, to enable or disable Call Tracing. For more information on configuring enterprise parameters, refer to the Cisco Unified Communications Manager Administration Guide.
- The default value for maximum number of Call Trace log files specifies 2000 and the default value for maximum Call Trace log file size specifies 2 MB.
Procedure

**Step 1**
To display information on Session Trace, from the RTMT menus, choose **CallManager > Call Process > Session Trace**.

The Session Trace screen displays.

**Step 2**
Enter the search criteria and Click **Run**.

**Note**
You can search calls based on the following criteria: Calling Number/URI, Called Number/URI, Start Time, and Duration. The search applies to the entire Unified CM cluster, not just the local node. If any node fails to collect the trace files, the system displays an error message in the bottom panel and pops up the message prompt to the user.

Click **Yes** to ignore the error and generate the table, based on the input.

**Note**
In Calling Number/URI, Called Number/URI, you can use wild character "*" to match any number of characters. For example, a search for 123* fetches numbers like "123", "123456", "123*", "1234", etc.

If you want to search for numbers with a "*" in them, use "\*". For example, to search for a Called Number like 12*45, enter 12\*45 in the search box.

If matching calls are found, the Matching Call pane displays Start Time, Calling DN, Original Called DN, Final Called DN, and Termination Cause Code. The Termination Cause Code helps to identify the failure calls, and provides the reason for the failure of the calls. The Termination Cause Code is displayed in parenthesis followed by description.

Consider the following scenario:

- If the call is in progress or if the call trace logging is turned off after the call, the Termination Cause Code column remains blank.

**Note**
If cause code description is missing or if you want more information on the Termination Cause Codes, refer the CDR cause codes in *Cisco Unified Call Details Records Administration Guide*.

**Step 3**
Select a call (a row) to trace.

By default, **Include SIP Message** check box is selected to view the associated SIP protocol messages or call transactions.

**Step 4**
To generate the SIP Message Call Flow Diagram, click **Trace Call**. If you want to stop the generation of the session information, click **Cancel** on the progress window.
The Analyze Call Diagram window displays the corresponding SIP messages in the Call Flow Diagram.

**Figure 2-1 Call Flow Diagram for Simple Call Scenario**

---

**Step 5**
Click the tabs that you want to view. The following tabs are available:

- Call Flow Diagram—Displays the corresponding SIP messages in the Call Flow Diagram.
- Log File—Displays the entire log file.
- SIP Message—Appears only when the **Include SIP Message** check box is checked. Displays the actual SIP message that gets logged into the SDI log file.

**Step 6**
The following table lists the messages that display when you move your mouse on each SIP message in the Call Flow Diagram:

<table>
<thead>
<tr>
<th>Displayed Messages</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sender</td>
<td>Displays the IP address of the originating call.</td>
</tr>
<tr>
<td>SIP Call ID</td>
<td>Displays the SIP call ID.</td>
</tr>
<tr>
<td>Message Label</td>
<td>Displays the message type for the corresponding SIP message onto which you move your mouse; for example, 200 OK, or 180 Ringing.</td>
</tr>
<tr>
<td>Receiver</td>
<td>Displays the IP address of the destination call.</td>
</tr>
<tr>
<td>Device Name</td>
<td>Displays the name of the device.</td>
</tr>
</tbody>
</table>
Chapter 2  New and Changed Information

Cisco Unified Communications Manager Managed Services Guide

OL-22523-01

2-17

Cisco Unified Communications Manager, Release 8.5(1)

Click the See message in log file link to view the subset of call logs that can be downloaded and analyzed.

Click the See SIP Message link. A new SIP Message tab displays adjacent to the Log File tab. Click the SIP message tab to display the actual SIP message that gets logged into the SDI log file.

To view the SIP messages that get logged into the SDI log file, do the following:


– set the trace level to any one of the following—State Transition, Significant, Arbitrary or Detailed.

You can view the See SIP Message link only when the Include SIP Message check box is checked.

Note

Step 7  Click Save.

The call flow diagram gets saved as index.html in the specified folder along with the SIP messages. You can email the files to the Technical Assistance Center (TAC).

You can do the following:

a. To view the online help, click Help.

b. To exit the Analyze Call Diagram screen, click Close.

c. To navigate to the previous page, click Previous Messages.

d. To navigate to the next page, click Next Messages.

Note  Previous Messages or Next Messages is enabled only when the message size exceeds a threshold.

Call Log files

The Session Manager logs the call data in new log files. These new log files are located in the following folder: /var/log/active/cm/trace/ccm/calllogs/.

The Call Log name has the following file name pattern: calllogs_ddd.ddddddd.txt.gz.
The Call Logs include the following message types:

- **Call Control**—Writes call information at call setup, split, join, and release.
  
<table>
<thead>
<tr>
<th>Timestamp</th>
<th>MessageType (CC)</th>
<th>Operation (SETUP/SPLI/JOIN/RELEASE)</th>
<th>CI for one leg (aCI)</th>
<th>CI for other leg (bCI)</th>
<th>calling DN</th>
<th>Orig Called DN</th>
<th>Final Called DN</th>
</tr>
</thead>
</table>

- **Device Layer**—Writes metadata information that relates to message from or to the device.

<table>
<thead>
<tr>
<th>Timestamp</th>
<th>MessageType (SIPL/SIPT)</th>
<th>My leg CI</th>
<th>Protocol (tcp/ucp)</th>
<th>Direction (IN/OUT)</th>
<th>local ip</th>
<th>local port</th>
<th>device name</th>
<th>device ip</th>
<th>device port</th>
<th>Correlation id</th>
<th>Message Tag</th>
<th>SIP Call ID</th>
<th>SIP method</th>
</tr>
</thead>
</table>

**Scenarios while Uninstalling RTMT**

Consider the following scenarios, while uninstalling RTMT:

1. User saves the call flow diagram files in a folder (created by the user under `user.dir` directory)—All the files in `user.dir` are deleted except the newly created folder and the `user.dir` directory.

2. User saves the call flow diagram files in a folder (created during RTMT installation) under `user.dir` directory—All the files in `user.dir` are deleted including the `user.dir` directory.

3. User directly saves the call flow diagram files under `user.dir` directory—All the files in `user.dir` are deleted except the newly created files and the `user.dir` directory.

4. User saves the call flow diagram files in a folder which is outside `user.dir` directory—`user.dir` and its contents are removed. The folder created by the user is not deleted.

**Limitations**

The following limitations apply when the Call Flow Diagram gets generated:

- Search does not show incomplete calls.
  
  **Example:**

  When the user picks up the handset and hangs up without dialing the complete DN, it will not be listed in the search results.

- The Call Flow Diagram does not show some SIP messages in the following scenarios:
  
  - Conference calls involving more than three parties.
  
  - A call leg is used to invoke a feature alone.

  **Example:**

  Phone B and Phone C are in the same pickup group.

  1. User A calls Phone B.
  2. User C lifts up the Phone C handset.
  3. User C presses the PickUp softkey to pickup the call.

  SIP messages exchanged in Step 2 are not displayed in the Call Flow Diagram.

  In these cases, a RELEASE message is logged in the call logs without a corresponding SETUP message.

**GUI Changes**

Use the enterprise parameter, Enable Call Trace Log, to enable or disable the Call Tracing. For more information on configuring enterprise parameters, refer to the *Cisco Unified Communications Manager Administration Guide*. 
Chapter 2  New and Changed Information

A new Session Trace window allows you to search for the matching calls, lists the matching call records, and provide the SIP message Call Flow Diagram. In RTMT, choose CallManager > Call Process > Session Trace.

Default Settings for Alarm Configuration Settings

The default alarm configuration settings information is included in the Cisco Unified Serviceability Administration Guide.

Table 2 describes the default alarm configuration settings.

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Default Alarm Configuration Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local Syslogs</td>
<td>Remote Syslogs</td>
</tr>
<tr>
<td>Enable Alarm</td>
<td>Checked</td>
</tr>
<tr>
<td>Alarm Event Level</td>
<td>Error</td>
</tr>
</tbody>
</table>

New Cisco Unity Connection Alerts

The following new alerts are added for Cisco Unity Connection:

- DiskConsumptionCloseToCapacityThreshold, page 2-19
- DiskConsumptionExceedsCapacityThreshold, page 2-20

DiskConsumptionCloseToCapacityThreshold

This alert is generated when the hard disk usage on the Cisco Unity Connection server reaches ten percent below the percentage limit specified on the System Settings > Advanced > Disk Capacity in Cisco Unity Connection Administration. For example, with a capacity threshold limit of 95 percent, the alert is triggered when usage reaches at least 85 percent.

Default Configuration

Table 3 lists the default configuration for the DiskConsumptionCloseToCapacityThreshold RTMT Alert.

<table>
<thead>
<tr>
<th>Table 3</th>
<th>Default Configuration for the DiskConsumptionCloseToCapacityThreshold RTMT Alert</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>Default Configuration</td>
</tr>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Error</td>
</tr>
<tr>
<td>Enable/Disable this alert on following server(s)</td>
<td>Enabled</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>DiskConsumptionCloseToCapacityThreshold event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
</tbody>
</table>
Cisco Unified Communications Manager, Release 8.5(1)

Chapter 2  New and Changed Information

DiskConsumptionExceedsCapacityThreshold

This alert is generated when the hard disk usage on the Cisco Unity Connection server meets or exceeds the percentage limit specified on the **System Settings > Advanced > Disk Capacity** in Cisco Unity Connection Administration.

Default Configuration

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Email</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

Table 3  Default Configuration for the DiskConsumptionCloseToCapacityThreshold RTMT Alert

Table 4  Default Configuration for the DiskConsumptionExceedsCapacityThreshold RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Error</td>
</tr>
<tr>
<td>Enable/Disable this alert on following server(s)</td>
<td>Enabled</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>DiskConsumptionExceedsCapacityThreshold event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable Email</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

Logging On to CAR

Before you log on to CAR, perform one of the following tasks:

- For CAR system administrators only—From Cisco Unified Serviceability, choose **Tools > CDR Analysis and Reporting**.
- For CAR users or administrators—From the web browser, enter

  https://<Server-ip/name>:8443/car/

Configuring the Trunk

Tip

Configure the trunks in CAR for existing Cisco Unified Communications Manager system trunks. After you add trunks to Cisco Unified Communications Manager Administration, configure the new trunks in CAR. When trunks are deleted from the Cisco Unified Communications Manager system, the system automatically removes the trunks (and any configuration settings that you specified) from CAR.
CAR uses the area code information to determine whether calls are local or long distance. You must provide the Number of Ports information for each trunk to enable CAR to generate the Utilization reports.

This section describes how to configure trunks in CAR.

**Procedure**

**Step 1** Choose **System > System Parameters > Trunk Configuration**. The Trunk Configuration window displays.

*Note* If you have not configured trunks in Cisco Unified Communications Manager Administration, a message displays that indicates that you have not configured trunks for the system.

*Note* CAR uses the area code information to determine whether calls are local or long distance. For 8.5(1) release, the area code is not updated for all the trunks.

**Step 2** In the Max No. of Ports field, enter the number of ports for each trunk that you want to configure. The maximum number of ports range from 1 to 1000.

*Note* CAR uses the values that were provided for the trunk when it was added in Cisco Unified Communications Manager Administration. Therefore, some trunks will already have a zero for maximum number of ports, depending on the details that were specified when the trunk was added in Cisco Unified Communications Manager Administration. CAR does not accept zero as a value for the maximum number of ports; you may be prompted to change the maximum number of ports for all trunks with a value of zero.

**Step 3** To make the changes, click the **Update** button.

You can run reports in CAR on any or all of the configured trunks.

**Configuring Trunk Utilization Reports**

Only CAR administrators generate the Trunk Utilization report. This report calculates the utilization reports for devices based on the duration of calls that passed through the devices.

You can generate this report on an hourly, daily, or monthly basis. The system calculates the utilization of a trunk for each hour in the selected date range. For example, the system calculates the utilization of a trunk between 11hrs-12hrs, using the formula, (Sum of the duration of calls that used the trunk in that hour / (total seconds in an hour * maximum number of ports in a trunk * number of days between the fromDate and toDate selected) * 100).

Similarly, to get the utilization for each day in a week, the system calculates the utilization using the formula, ((sum of the duration of calls that used the trunk in a day) / (total seconds in each day * number of each day between the fromDate and toDate selected * maximum number of ports in a trunk) * 100).

In the case of monthly utilization reports, the system calculates the utilization for each day in a month, using the formula, ((sum of the duration of calls that used the trunk in a day) / (total seconds in each day * number of each day between the fromDate and toDate selected * maximum number of ports in a trunk) * 100).
Reports generate for each trunk that is chosen.

For calculation of the trunk utilization, the system uses the port numbers from the CAR Trunk Configuration window. To find this window, choose System > System Parameters > Trunk Configuration. You cannot take port details for H.323 trunks from the Cisco Unified Communications Manager database because the H.323 port number always equals zero in the database. The user must update H.323 trunk ports information in the CAR Trunk Configuration window.

Be aware that the only port detail information that is taken from the CAR Trunk Configuration window is only for those trunks that do not have port details that are available or that show zero in the Cisco Unified Communications Manager database.

This section describes how to generate, view, or mail Trunk Utilization reports.

Procedure

**Step 1** Choose Device Reports > Trunk > Utilization.

The Trunk Utilization window displays.

**Step 2** In the Generate Reports field, choose a time as described in Table 5.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hour of Day</td>
<td>Displays the cumulative utilization for each hour in a 24-hour period for the period that you specify in Step 7.</td>
</tr>
<tr>
<td>Day of Week</td>
<td>Displays the cumulative utilization for the days of the week that occur within the period that you specify in Step 7.</td>
</tr>
<tr>
<td>Day of Month</td>
<td>Displays the cumulative utilization for the days of the month that occur within the period that you specify in Step 7.</td>
</tr>
</tbody>
</table>

**Note** The Trunk Utilization report is not generated automatically.

**Step 3** To display the list of trunks that you can include in the report in the List of Trunks box, perform one of the following tasks:

- To display all trunks in the List of Trunks box, click Trunk Types in the column on the left side of the window.

- To display trunks for a particular trunk type in the List of Trunks box, click the icon next to Trunk Types in the column on the left side of the window. The tree structure expands, and a list of trunk types displays. Choose a trunk type from the list, and the trunk name displays in the List of Trunks box.

**Note** The List of Trunks box will list up to 200 trunks that are configured for the chosen trunk type.
Chapter 2  New and Changed Information

Cisco Unified Communications Manager, Release 8.5(1)

Note You can generate the trunk utilization reports for route groups, route lists, and route patterns that are connected through trunks.

Step 4 Choose a trunk type from the list.
The trunk name displays in the List of Trunks box.

Note The List of Trunks box displays up to 200 trunks that are configured for the chosen trunk type.

Step 5 In the List of Trunks box, choose the trunks that you want to include in the report.

Note You can generate a report for up to five trunks at a time.

Step 6 To move the chosen trunk to the list of Selected Trunks box, click the down arrow.
The trunk(s) that you chose displays in the Selected Trunks box.

Step 7 If you chose Generate New Report, enter the date range of the period for which you want to see call information.

Note Ensure the date and time range does not exceed one month.

Step 8 If you want the report in CSV format, choose CSV (comma separated value) in the Report Format area.
If you want the report in PDF format, choose PDF (portable document format) in the Report Format area.

Step 9 Click the View Report button.
The report displays.

Step 10 If you want to mail the report, click the Send Report button.

Cisco Dialed Number Analyzer Server

The Cisco Dialed Number Analyzer Server service along with the Cisco Dialed Number Analyzer service supports Cisco Unified Communications Manager Dialed Number Analyzer. This service needs to be activated only on the node that is dedicated specifically for the Cisco Dialed Number Analyzer service.

This service can be activated by choosing Tools > Service Activation and choosing Tools > Dialed Number Analyzer Server, from the Serviceability UI.

Unified CM clusters only: Cisco does not recommend that you activate the service on all the servers in a cluster. Cisco recommends that you activate this service only on one of the servers of a cluster where call-processing activity is the least.

In a Cisco Unified Communications Manager Business Edition 5000 system, this service supports Cisco Unified Communications Manager only.
The following states the recommendation for the newly added service:

<table>
<thead>
<tr>
<th>Service/Servlet</th>
<th>Activation Recommendations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Dialed Number Analyzer Server</td>
<td>If you have more than one node in the cluster, activate this service on one node that is dedicated specifically for the Cisco Dialed Number Analyzer service.</td>
</tr>
</tbody>
</table>

**Cisco Unified Communications Manager, Release 8.0(2)**

The Cisco Unified Real-Time Monitoring Tool runs on both the Cisco Intercompany Media Engine server and the Cisco Unified Communications Managerservers to provide information about system and feature health.

You can install RTMT onto a client machine from only one product type—Cisco Unified Communications Manager or Cisco Intercompany Media Engine. Installing RTMT client from different product types on the same client machine is not supported.

For information on installing RTMT on a Cisco Intercompany Media Engine server, refer to the *Cisco Intercompany Media Engine Installation and Configuration Guide*.

For information on installing RTMT on a Cisco Unified Communications Manager server, refer to the *Cisco Unified Real-Time Monitoring Tool Administration Guide*.

This section describes the new and changed information in Cisco Unified Communications Manager, Release 8.0(2). It contains information on new performance objects, and alerts for both the Cisco Unified Communications Manager server and the Cisco Intercompany Media Engine server.

- Cisco Unified Communications Manager Server, page 2-24
- Cisco Intercompany Media Engine Server, page 2-25

**Cisco Unified Communications Manager Server**

**Performance Objects**

The following performance objects are available on the Cisco Unified Communications Manager server to support Cisco Intercompany Media Engine. For descriptions of the objects and related counters, refer to the *Cisco Intercompany Media Engine Installation and Configuration Guide*.

- IME Client
- IME Client Instance

**Alerts**

The following alerts are available on the Cisco Unified Communications Manager server to support Cisco Intercompany Media Engine. For descriptions and default configuration settings, refer to the *Cisco Intercompany Media Engine Installation and Configuration Guide*.

- IMEDistributedCacheInactive
- IMEOverQuota
- IMEQualityAlert
- InsufficientFallbackIdentifiers
- IMEServiceStatus
Cisco Intercompany Media Engine Server

Performance Objects
The following performance objects are available on the Cisco Intercompany Media Engine server to support the Cisco Intercompany Media Engine feature. For descriptions of the objects and related counters, refer to the Cisco Intercompany Media Engine Installation and Configuration Guide.

- IME Configuration Manager
- IME Server
- IME Server System Performance

Alerts
The following alerts are available on the Cisco Intercompany Media Engine server to support the Cisco Intercompany Media Engine feature. For descriptions and default configuration settings, refer to the Cisco Intercompany Media Engine Installation and Configuration Guide.

- BannedFromNetwork
- IMEDistributedCacheCertificateExpiring
- IMEDistributedCacheFailure
- IMESdlLinkOutOfService
- InvalidCertificate
- InvalidCredentials
- MessageOfTheDay
- SWUpdateRequired
- TicketPasswordChanged
- ValidationsPendingExceeded
- CriticalAuditEventGenerated

Cisco Unified Communications Manager, Release 8.0(1)

This section describes the new and changed information in Cisco Unified Communications Manager, Release 8.0(1). It contains the following subsections:

- Cisco Unified Serviceability, page 2-26
- Cisco Unified Real-Time Monitoring Tool, page 2-47
- Cisco Unified CDR Analysis and Reporting, page 2-51
- Cisco Unified Call Detail Records, page 2-52
- Cisco Unified Reporting, page 2-55
Cisco Unified Serviceability

This section contains the following subsections:

- New Alarm Additions and Changes, page 2-26
- Obsolete Alarms, page 2-43

New Alarm Additions and Changes

The following lists the new alarm catalogs that are added:

- Audit Log Catalog—The following new Audit Log alarms are added in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdministrativeEvent</td>
<td>Failed to write into the primary file path. Audit Event is generated by this application. Severity level is Informational.</td>
</tr>
<tr>
<td>CriticalEvent</td>
<td>Failed to write into the primary file path. Audit Event is generated by this application. Severity level is Informational.</td>
</tr>
<tr>
<td>SecurityEvent</td>
<td>Failed to write into the primary file path. Audit Event is generated by this application. Severity level is Informational.</td>
</tr>
</tbody>
</table>

- EM Alarm Catalog—The following new EM alarms are added in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMAppInitializationFailed</td>
<td>EM Application not started. Error occurred while starting application. Severity level is Error.</td>
</tr>
<tr>
<td>EMAppStarted</td>
<td>EM Application started successfully. Severity level is Informational.</td>
</tr>
<tr>
<td>EMAppStopped</td>
<td>EM Application started. Application is shutting down gracefully because of an unloaded from Tomcat. Severity level is Notice.</td>
</tr>
<tr>
<td>EMCCFailedInLocalCluster</td>
<td>EMCC login failure occurred due to one of the following conditions:</td>
</tr>
<tr>
<td></td>
<td>- Devices are incompatible with EMCC.</td>
</tr>
<tr>
<td></td>
<td>- Unable to retrieve remote cluster information.</td>
</tr>
<tr>
<td></td>
<td>- EMCC is restricted by the local cluster.</td>
</tr>
<tr>
<td></td>
<td>- EMCC is restricted by the local cluster.</td>
</tr>
</tbody>
</table>
### Chapter 2  New and Changed Information

#### Cisco Unified Communications Manager, Release 8.0(1)

**TVS Alarm Catalog**—The following new TVS alarms are added in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMCCFailedInRemoteCluster</td>
<td>There was an EMCC login failure at a remote Unified CM. EMCC login could fail due to the following reasons:</td>
</tr>
<tr>
<td></td>
<td>• User does not exist in any of the configured remote cluster.</td>
</tr>
<tr>
<td></td>
<td>• User is not enabled for EMCC.</td>
</tr>
<tr>
<td></td>
<td>• No free EMCC base device.</td>
</tr>
<tr>
<td></td>
<td>• EMCC access was prevented by remote cluster.</td>
</tr>
<tr>
<td></td>
<td>• Untrusted certificate received from the remote end while trying to establish a connection.</td>
</tr>
<tr>
<td>EMCCUserLoggedIn</td>
<td>EMCC login was successful. Severity level is Informational(6).</td>
</tr>
<tr>
<td>EMCCUserLoggedOut</td>
<td>EMCC logout was successful. Severity level is Informational(6).</td>
</tr>
<tr>
<td>EMServiceConnectionError</td>
<td>EM Service not reachable. EM Service might be down in one or more nodes in the cluster. Severity level is Error.</td>
</tr>
<tr>
<td>NodeNotTrusted</td>
<td>Untrusted Node was contacted. Severity level is Error.</td>
</tr>
<tr>
<td>UserInputFailure</td>
<td>EMCC login failure due to invalid user input due to invalid user credentials or the credentials have expired. Severity level is Warning(4).</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConfigThreadChangeNotifyServerSingleFailed</td>
<td>Failed to allocate resources to handle configuration change notification from database.</td>
</tr>
<tr>
<td>ConfigThreadReadConfigurationFailed</td>
<td>Failed to retrieve enterprise parameter values from database at TVS service startup.</td>
</tr>
<tr>
<td>DefaultDurationInCacheModified</td>
<td>Default value of a Certificate duration in cache is modified in the Service Parameter page.</td>
</tr>
<tr>
<td>ITLFileRegenerated</td>
<td>New ITL File has been generated.</td>
</tr>
<tr>
<td>RollBackToPre8.0Disabled</td>
<td>Roll Back to Pre 8.0 has been disabled in the Enterprise Parameter page.</td>
</tr>
<tr>
<td>SDIControlLayerFailed</td>
<td>Failed to update trace logging or alarm subsystem for new settings.</td>
</tr>
<tr>
<td>TVSCertificateRegenerated</td>
<td>TVS Server certificate has been regenerated.</td>
</tr>
<tr>
<td>TVSServerListenBindFailed</td>
<td>Fail to connect to the network port through which file requests are received.</td>
</tr>
<tr>
<td>TVSServerListenSetSockOptFailed</td>
<td>Failed to increase the size of the network buffer for receiving file requests.</td>
</tr>
</tbody>
</table>

**Call Manager Catalog**—The following new Call Manager alarms are added in the alarm catalog:
<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMVersionMismatch</td>
<td>One or more Unified CM nodes in a cluster are running different Cisco CallManager versions.</td>
</tr>
<tr>
<td>ConflictingDataIE</td>
<td>A call has been rejected because the incoming PRI/BRI Setup message had an invalid IE.</td>
</tr>
<tr>
<td>DbInfoCorrupt</td>
<td>Database information returned is corrupt. Database configuration error was encountered.</td>
</tr>
<tr>
<td>DbInfoError</td>
<td>Error in the database information retrieved. Database configuration error was encountered.</td>
</tr>
<tr>
<td>DbInfoTimeout</td>
<td>Database Information request timed out. Timeout was encountered while trying to read database configuration.</td>
</tr>
<tr>
<td>DbInsertValidatedDIDFailure</td>
<td>The Insertion of an IME provided E.164 DID has failed. A failure occurred attempting to insert a Cisco Unified Active Link learned DID.</td>
</tr>
<tr>
<td>EndPointRegistered</td>
<td>This alarm occurs when a device is successfully registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>EndPointResetInitiated</td>
<td>This alarm occurs when a device is reset via the Reset button in Cisco Unified CM Administration.</td>
</tr>
<tr>
<td>EndPointRestartInitiated</td>
<td>Device restart initiated or Apply Config initiated on the specified device.</td>
</tr>
<tr>
<td>EndPointTransientConnection</td>
<td>Endpoint transient connection attempt.</td>
</tr>
<tr>
<td>EndPointUnregistered</td>
<td>An endpoint that has previously registered with Cisco Unified Communications Manager has unregistered.</td>
</tr>
<tr>
<td>FirewallMappingFailure</td>
<td>Firewall unreachable.</td>
</tr>
<tr>
<td>IMEQualityAlertEntry</td>
<td>IME call quality problem.</td>
</tr>
<tr>
<td>IMEQualityAlertExit</td>
<td>IME call quality problem cleared.</td>
</tr>
<tr>
<td>IMEDistributedCacheInactive</td>
<td>Inactive IME distributed cache.</td>
</tr>
<tr>
<td>IMEOverQuota</td>
<td>Each IME server has a fixed quota on the total number of DIDs it can write into the IME distributed cache.</td>
</tr>
<tr>
<td>InsufficientFallbackIdentifiers</td>
<td>Cannot allocate fallback identifier.</td>
</tr>
<tr>
<td>InvalidSubscription</td>
<td>A message has been received from an IME server that contains a subscription identifier that is not handled by this node.</td>
</tr>
<tr>
<td>RouteRemoved</td>
<td>Route is removed automatically.</td>
</tr>
<tr>
<td>InvalidCredentials</td>
<td>Credential Failure to IME server.</td>
</tr>
<tr>
<td>PublicationRunCompleted</td>
<td>Completion of publication of published DID patterns.</td>
</tr>
<tr>
<td>PublishFailed</td>
<td>Unified CM attempted to store a number into the IME distributed cache, but the attempt failed. This is typically due to a transient problem in the IME distributed cache.</td>
</tr>
<tr>
<td>PublishFailedOverQuota</td>
<td>Each IME server has a fixed quota on the total number of DIDs it can write into the IME distributed cache.</td>
</tr>
<tr>
<td>RejectedRoutes</td>
<td>Rejected route due to Untrusted status.</td>
</tr>
<tr>
<td>TCPSetupToIMEFailed</td>
<td>Connection Failure to IME server.</td>
</tr>
</tbody>
</table>
Chapter 2      New and Changed Information

Cisco Unified Communications Manager, Release 8.0(1)

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLSConnectionToIMEFailed</td>
<td>TLS Failure to IME service.</td>
</tr>
</tbody>
</table>

### New SAF and CCD Alarms

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LostConnectionToSAFForwarder</td>
<td>Connection to the SAF Forwarder has been lost.</td>
</tr>
<tr>
<td>SAFForwarderError</td>
<td>SAF Forwarder error response sent to Unified CM.</td>
</tr>
<tr>
<td>SAFUnknownService</td>
<td>Unified CM does not recognize the service ID in a publish revoke or withdraw message.</td>
</tr>
<tr>
<td>SAFPublishRevoke</td>
<td>A CLI command revoked the publish action for the specified service or subservice ID.</td>
</tr>
<tr>
<td>SAFResponderError</td>
<td>This is raised when SAF forwarder doesn't know the transaction ID within SAF response from this Cisco Unified CM.</td>
</tr>
<tr>
<td>DuplicateLearnedPattern</td>
<td>This alarm occurs when CCD requesting service received a duplicate Hosted DN.</td>
</tr>
<tr>
<td>CCDPReachableTimeOut</td>
<td>CCD Requesting Service IP Reachable Duration times out.</td>
</tr>
<tr>
<td>CCDPSTNFailOverDurationTimeOut</td>
<td>The internal limit on PSTN failover has expired.</td>
</tr>
</tbody>
</table>

### New Alarms in External Call Control

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectionFailureToPDP</td>
<td>A connection request from Unified CM to the policy decision point (PDP) failed.</td>
</tr>
<tr>
<td>ConnectionToPDPInService</td>
<td>A connection was successfully established between Cisco Unified Communications Manager (Unified CM) and the policy decision point (PDP).</td>
</tr>
<tr>
<td>AwaitingResponseFromPDPTimeout</td>
<td>Cisco Unified Communication Manager timed out waiting for the routing response from the policy decision point.</td>
</tr>
<tr>
<td>ErrorParsingResponseFromPDP</td>
<td>Cisco Unified Communications Manager failed to parse one or multiple optional elements or attributes in the call routing response from the policy decision point.</td>
</tr>
<tr>
<td>ErrorParsingDirectiveFromPDP</td>
<td>Cisco Unified Communications Manager (Unified CM) failed to parse the call routing directive or the diversion destination in the call routing response from the policy decision point (PDP).</td>
</tr>
<tr>
<td>FailureResponseFromPDP</td>
<td>The policy decision point (PDP) returned a 4xx (client) or 5xx (server) status code in the HTTP response.</td>
</tr>
<tr>
<td>CallAttemptBlockedByPolicy</td>
<td>A call was attempted but blocked or rejected by the policy decision point (PDP).</td>
</tr>
<tr>
<td>FailedToFulfillDirectiveFromPDP</td>
<td>Cisco Unified Communications Manager cannot fulfill the call routing directive returned by the PDP.</td>
</tr>
<tr>
<td>DigitAnalysisTimeoutAwaitingResponse</td>
<td>Cisco Unified Communications Manager sent a routing request to the policy decision point but the request timed out without a response.</td>
</tr>
</tbody>
</table>

### Changed Alarms in Call Manager Catalog

The following CallManager alarms are changed in the alarm catalog:
### Alarm Names

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Alarm Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>AnnunciatorNoMoreResourcesAvailable</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>BChannelISV</td>
<td>Severity changed from Informational to Notice.</td>
</tr>
<tr>
<td>BChannelOOS</td>
<td>Severity changed from Error to Critical.</td>
</tr>
<tr>
<td>BeginThrottlingCallListBLFSubscriptions</td>
<td>Severity level is Warning.</td>
</tr>
<tr>
<td>CMINitializationStateTime</td>
<td>Severity level is Informational.</td>
</tr>
<tr>
<td>CMOverallInitTimeExceeded</td>
<td>Severity changed from Error to Alert.</td>
</tr>
<tr>
<td>CMTotalInitializationStateTime</td>
<td>Severity level is Informational.</td>
</tr>
<tr>
<td>CallManagerFailure</td>
<td>Severity changed from Error to Critical; Enum Definitions are updated.</td>
</tr>
<tr>
<td>CallManagerOnline</td>
<td>Severity level is Notice.</td>
</tr>
<tr>
<td>CodeRedEntry</td>
<td>Severity changed from Error to Critical.</td>
</tr>
<tr>
<td>CodeYellowEntry</td>
<td>Severity changed from Error to Critical.</td>
</tr>
<tr>
<td>CodeYellowExit</td>
<td>Severity changed from Error to Notice.</td>
</tr>
<tr>
<td>ConferenceNoMoreResourcesAvailable</td>
<td>Changed severity level from Error to Warning.</td>
</tr>
<tr>
<td>ConnectionFailure</td>
<td>Severity level is Error (3).</td>
</tr>
<tr>
<td>DBLException</td>
<td>Severity changed from Error to Alert.</td>
</tr>
<tr>
<td>DChannelISV</td>
<td>Severity changed from Informational to Notice.</td>
</tr>
<tr>
<td>DChannelOOS</td>
<td>Severity changed from Error to Critical.</td>
</tr>
<tr>
<td>DaTimeOut</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>DatabaseDefaultsRead</td>
<td>Severity changed from Notice to Informational.</td>
</tr>
<tr>
<td>DeviceApplyConfigInitiated</td>
<td>Severity level is Informational.</td>
</tr>
<tr>
<td>DeviceCloseMaxEventsExceeded</td>
<td>Severity level is Error (3).</td>
</tr>
<tr>
<td>DeviceDnInformation</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>DeviceInitTimeout</td>
<td>Severity level is Error (3).</td>
</tr>
<tr>
<td>DevicePartiallyRegistered</td>
<td>Following information is updated:</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for performance monitor object type</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for DeviceType</td>
</tr>
<tr>
<td>DeviceRegistered</td>
<td>Following information is updated:</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for Performance Monitor ObjType</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for Device type</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for IPAddrAttributes</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for IPV6AddrAttributes</td>
</tr>
<tr>
<td>DeviceResetInitiated</td>
<td>• Enum Definitions for DeviceType are updated.</td>
</tr>
<tr>
<td></td>
<td>• Parameters added: Product type [String]</td>
</tr>
</tbody>
</table>

Cisco Unified Communications Manager Managed Services Guide

2-30

OL-22523-01
<table>
<thead>
<tr>
<th>Alarm Names</th>
<th>Alarm Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeviceRestartInitiated</td>
<td>• Enum Definitions for DeviceType are updated.</td>
</tr>
<tr>
<td></td>
<td>• Parameters added: Product type [String]</td>
</tr>
<tr>
<td>DeviceTransientConnection</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following information is updated:</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for DeviceType</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for IPAddrAttributes</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for IPV6AddrAttributes</td>
</tr>
<tr>
<td>DeviceTypeMismatch</td>
<td>Following information is updated:</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for DBDeviceType</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for DeviceType</td>
</tr>
<tr>
<td>DeviceUnregistered</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following information is updated:</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for DeviceType</td>
</tr>
<tr>
<td></td>
<td>– Enum Definition</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for IPAddrAttributes</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for IPV6AddrAttributes</td>
</tr>
<tr>
<td>EndThrottlingCallListBLFSubscriptions</td>
<td>Severity changed from Warning to Informational.</td>
</tr>
<tr>
<td>H323Started</td>
<td>• Severity changed from Informational to Notice.</td>
</tr>
<tr>
<td></td>
<td>• Following information is updated:</td>
</tr>
<tr>
<td></td>
<td>– Parameters</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for DeviceType</td>
</tr>
<tr>
<td>H323Stopped</td>
<td>Following information is updated:</td>
</tr>
<tr>
<td></td>
<td>• Parameters</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for DeviceType</td>
</tr>
<tr>
<td>ICTCallThrottlingEnd</td>
<td>Severity changed from Error to Notice.</td>
</tr>
<tr>
<td>ICTCallThrottlingStart</td>
<td>Severity level is Error (3).</td>
</tr>
<tr>
<td>MGCPGatewayGainedComm</td>
<td>Severity changed from Informational to Notice.</td>
</tr>
<tr>
<td>MaliciousCall</td>
<td>Severity changed from Informational to Warning.</td>
</tr>
</tbody>
</table>
### Alarm Names

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Alarm Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MaxCallDurationTimeout</td>
<td>• Severity changed from Informational to Notice.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>– Originating Device name(String)</td>
</tr>
<tr>
<td></td>
<td>– Destination Device name(String)</td>
</tr>
<tr>
<td></td>
<td>– Call start time(UInt)</td>
</tr>
<tr>
<td></td>
<td>– Call stop time(UInt)</td>
</tr>
<tr>
<td></td>
<td>– Calling Party Number(String)</td>
</tr>
<tr>
<td></td>
<td>– Called Party Number(String)</td>
</tr>
<tr>
<td>MaxCallsReached</td>
<td>Severity changed from Error to Critical.</td>
</tr>
<tr>
<td>MaxHoldDurationTimeout</td>
<td>Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>• Originating Device Name(String)</td>
</tr>
<tr>
<td></td>
<td>• Destination Device Name(String)</td>
</tr>
<tr>
<td></td>
<td>• Hold start time(UInt)</td>
</tr>
<tr>
<td></td>
<td>• Hold stop time(UInt)</td>
</tr>
<tr>
<td></td>
<td>• Calling Party Number(String)</td>
</tr>
<tr>
<td></td>
<td>• Called Party Number(String)</td>
</tr>
<tr>
<td>MediaResourceListExhausted</td>
<td>Enum Definitions for MediaResourceType is updated.</td>
</tr>
<tr>
<td>MohNoMoreResourcesAvailable</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>MtpNoMoreResourcesAvailable</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Media Resource List Name parameter added.</td>
</tr>
<tr>
<td>MultipleSIPTrunksToSamePeerAndLocalPort</td>
<td>Severity level is Error.</td>
</tr>
<tr>
<td>NoFeatureLicense</td>
<td>Severity changed from Error to Emergency.</td>
</tr>
<tr>
<td>NotEnoughChans</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Device Name(String) is the only parameter.</td>
</tr>
<tr>
<td>NumDevRegExceeded</td>
<td>Severity level is Error (3).</td>
</tr>
<tr>
<td>PktCapOnDeviceStarted</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>PktCapOnDeviceStopped</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>PktCapServiceStarted</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>PktCapServiceStopped</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>RouteListExhausted</td>
<td>Severity level is Warning.</td>
</tr>
<tr>
<td>RsvpNoMoreResourcesAvailable</td>
<td>Media Resource List Name(String) parameter is added.</td>
</tr>
<tr>
<td>SDLLinkISV</td>
<td>Severity changed from Informational to Notice.</td>
</tr>
<tr>
<td>SDLLinkOOS</td>
<td>Severity changed from Error to Alert.</td>
</tr>
</tbody>
</table>

---

### Alarm Names

- MaxCallDurationTimeout: Severity changed from Informational to Notice. Following parameters added:
  - Originating Device name(String)
  - Destination Device name(String)
  - Call start time(UInt)
  - Call stop time(UInt)
  - Calling Party Number(String)
  - Called Party Number(String)

- MaxCallsReached: Severity changed from Error to Critical.

- MaxHoldDurationTimeout: Following parameters added:
  - Originating Device Name(String)
  - Destination Device Name(String)
  - Hold start time(UInt)
  - Hold stop time(UInt)
  - Calling Party Number(String)
  - Called Party Number(String)

- MediaResourceListExhausted: Enum Definitions for MediaResourceType is updated.

- MohNoMoreResourcesAvailable: Severity changed from Error to Warning.


- MultipleSIPTrunksToSamePeerAndLocalPort: Severity level is Error.

- NoFeatureLicense: Severity changed from Error to Emergency.

- NotEnoughChans: Severity changed from Error to Warning. Device Name(String) is the only parameter.

- NumDevRegExceeded: Severity level is Error (3).

- PktCapOnDeviceStarted: Severity level is Informational (6).

- PktCapOnDeviceStopped: Severity level is Informational (6).

- PktCapServiceStarted: Severity level is Informational (6).

- PktCapServiceStopped: Severity level is Informational (6).

- RouteListExhausted: Severity level is Warning.

- RsvpNoMoreResourcesAvailable: Media Resource List Name(String) parameter is added.

- SDLLinkISV: Severity changed from Informational to Notice.

- SDLLinkOOS: Severity changed from Error to Alert.
<table>
<thead>
<tr>
<th>Alarm Names</th>
<th>Alarm Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIPSLineRegistrationError</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for DeviceType are updated.</td>
</tr>
<tr>
<td></td>
<td>• Enum Reasons table is updated.</td>
</tr>
<tr>
<td>SIPSStarted</td>
<td>• Severity changed from Informational to Notice.</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for InTransportType and</td>
</tr>
<tr>
<td></td>
<td>OutTransportType are updated.</td>
</tr>
<tr>
<td>SIPSStopped</td>
<td>Enum Definitions for InTransportType and</td>
</tr>
<tr>
<td></td>
<td>OutTransportType are updated.</td>
</tr>
<tr>
<td>StationAlarm</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>StationConnectionError</td>
<td>• Reason Code[Enum] parameter added.</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for Reason Code table added.</td>
</tr>
<tr>
<td>StationEventAlert</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>StationTCPInitError</td>
<td>• Severity changed from Error to Critical.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters are removed:</td>
</tr>
<tr>
<td></td>
<td>– Error Number [String]</td>
</tr>
<tr>
<td></td>
<td>– ErrorCode [Int]</td>
</tr>
<tr>
<td>TimerThreadSlowed</td>
<td>Severity changed from Warning to Critical.</td>
</tr>
<tr>
<td>UserUserPrecedenceAlarm</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Enum definitions updated.</td>
</tr>
</tbody>
</table>

- **CDRRep Alarm Catalog**—The following CDRRep alarms are updated in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Alarm Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDRAgentSendFileFailed</td>
<td>Changed Data Collector Routing List element to Alert Manager.</td>
</tr>
<tr>
<td>CDRAgentSendFileFailureContinues</td>
<td>Severity level is Error (3).</td>
</tr>
<tr>
<td>CDRFileDeliveryFailed</td>
<td>Changed Data Collector Routing List element to Alert Manager.</td>
</tr>
<tr>
<td>CDRFileDeliveryFailureContinues</td>
<td>Severity level is Error (3).</td>
</tr>
<tr>
<td>CDRHWMExceeded</td>
<td>Changed Data Collector Routing List element to Alert Manager.</td>
</tr>
<tr>
<td>CDRMaximumDiskSpaceExceeded</td>
<td>Facility and sub-facility changed. Added Routing List and changed Data Collector to Alert Manager.</td>
</tr>
</tbody>
</table>

- **Certificate Monitor Alarm Catalog**—The following new Certificate Monitor alarms are added in the alarm catalog:
### CMI Alarm Catalog

- **CertValidLessThanADay**
  Certificate is about to expire in less than 24 hours or has expired.

- **CertValidFor7Days**
  Alarm indicates that the certificate has expired or expires in less than seven days.

- **CertValidityOver30Days**
  Alarm indicates that the certificate expiry is approaching but the expiry date is more than 30 days.

- **CertValidLessThanMonth**
  Alarm indicates that the certificate will expire in 30 days or less.

- **CMIException**
  Error while reading the database.

- **CMIServiceStatus**
  CMI service is running and working properly.

- **DBLException**
  Unable to connect to the database.

- **InvalidPortHandle**
  The handle for the opened serial port is invalid.

- **MemAllocFailed**
  CMI tried to allocate memory and failed.

- **ParityConfigurationError**
  The CMI service parameter, Parity, has an invalid configuration.

- **ReadingFileFailure**
  CMI failed to read SMDI messages from the serial port.

- **SMDICmdError**
  CMI receives an invalid incoming SMDI message.

- **SMDIMessageError**
  SMDI message contains invalid DN.

- **SerialPortGetStatusError**
  When CMI tries to get the status of serial port, the operating system returns an error.

- **SerialPortOpeningError**
  When CMI tries to open the serial port, the operating system returns an error.

- **SerialPortSetStatusError**
  When CMI tries to set the status of serial port, the operating system returns an error.

- **StopBitConfigurationError**
  The Cisco Messaging Interface service parameter, Stop Bits, has an invalid configuration.

- **ThreadKillingError**
  An error occurred when CMI tried to stop the CMI service.

- **UnknownException**
  Unknown error while connecting to database.

- **VMDNConfigurationError**
  The Voice Mail DN for CMI is invalid.

- **WritingFileFailure**
  CMI failed to write SMDI messages to the serial port.

### CTI Manager Alarm Catalog

- The following new CTI Manager alarms are added in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PowerDown</td>
<td>Alarm indicates that the power is down.</td>
</tr>
<tr>
<td>PowerSuspended</td>
<td>Alarm indicates that the power is suspended.</td>
</tr>
<tr>
<td>PowerError</td>
<td>Error occurred when power signals are not received.</td>
</tr>
<tr>
<td>PowerDownTimeout</td>
<td>Alarm is triggered if power is down for more than the configured threshold.</td>
</tr>
<tr>
<td>PowerSuspendedTimeout</td>
<td>Alarm is triggered if power is suspended for more than the configured period.</td>
</tr>
<tr>
<td>PowerOff</td>
<td>Alarm indicates that the power is off.</td>
</tr>
<tr>
<td>PowerOn</td>
<td>Alarm indicates that the power is on.</td>
</tr>
</tbody>
</table>

- **PowerDown**
  Alarm indicates that the power is down.

- **PowerSuspended**
  Alarm indicates that the power is suspended.

- **PowerError**
  Error occurred when power signals are not received.

- **PowerDownTimeout**
  Alarm is triggered if power is down for more than the configured threshold.

- **PowerSuspendedTimeout**
  Alarm is triggered if power is suspended for more than the configured period.

- **PowerOff**
  Alarm indicates that the power is off.

- **PowerOn**
  Alarm indicates that the power is on.
<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ApplicationConnectionDropped</td>
<td>Application has dropped the connection to CTIManager.</td>
</tr>
<tr>
<td>ApplicationConnectionError</td>
<td>CTIManager is unable to allow connections from Applications.</td>
</tr>
<tr>
<td>Ct Device Closed</td>
<td>Application closed a device.</td>
</tr>
<tr>
<td>Ct Device In Service</td>
<td>Device is back in service.</td>
</tr>
<tr>
<td>Ct Device Open Failure</td>
<td>Application is unable to open the device.</td>
</tr>
<tr>
<td>Ct Device Opened</td>
<td>Application opened a device.</td>
</tr>
<tr>
<td>Ct Device Out of Service</td>
<td>Device is out of service.</td>
</tr>
<tr>
<td>Ct Line Closed</td>
<td>Application closed the line.</td>
</tr>
<tr>
<td>Ct Line In Service</td>
<td>Line is back in service.</td>
</tr>
<tr>
<td>Ct Line Open Failure</td>
<td>Application is unable to open the line.</td>
</tr>
<tr>
<td>Ct Line Opened</td>
<td>Application opened the line.</td>
</tr>
<tr>
<td>Ct Line Opened</td>
<td>Line is out of service.</td>
</tr>
<tr>
<td>Ct Max Connection Reached</td>
<td>Maximum number of CTI connections has been reached, no new connection will be accepted unless an existing connection is closed.</td>
</tr>
<tr>
<td>Ct Provider Close Heartbeat Timeout</td>
<td>CTI heartbeat timeout occurred causing CTIManager to close the application connection.</td>
</tr>
<tr>
<td>Ct Provider Closed</td>
<td>CTI application closed the provider. The IP address is shown in either IPv4 or IPv6 format depending on the IP addressing mode of the application.</td>
</tr>
<tr>
<td>Ct Provider Open Failure</td>
<td>CTI application is unable to open the provider. The IP address is shown in either IPv4 or IPv6 format depending on the IP addressing mode of the application.</td>
</tr>
<tr>
<td>Ct Provider Opened</td>
<td>CTI Application opened the provider successfully. The IP address is shown in either IPv4 or IPv6 format depending on the IP addressing mode of the Application.</td>
</tr>
<tr>
<td>Ct Qbe Failure Response</td>
<td>The requested operation from the application could not be performed because of a normal or abnormal condition.</td>
</tr>
<tr>
<td>Invalid QBE Message</td>
<td>QBE PDU from application is invalid.</td>
</tr>
<tr>
<td>Max Devices Per Node Exceeded</td>
<td>An application has opened more devices than the limit set in the CTIManager service parameter, Maximum Devices Per Node.</td>
</tr>
<tr>
<td>Max Devices Per Provider Exceeded</td>
<td>An application has opened more devices than the limit set in the CTIManager service parameter, Maximum Devices Per Provider.</td>
</tr>
<tr>
<td>Redirect Call Request Failed</td>
<td>CTIManager is unable to redirect a call.</td>
</tr>
</tbody>
</table>
Cisco Unified Communications Manager, Release 8.0(1)

Chapter 2  New and Changed Information

• DB Alarm Catalog—The following DB alarms are updated in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UnableToRegisterwithCallManagerService</td>
<td>CTI cannot communicate with Cisco CallManager service to register supplementary service features.</td>
</tr>
<tr>
<td>UnableToSetorResetMWI</td>
<td>An error occurred when setting the message waiting indication (MWI) lamp.</td>
</tr>
</tbody>
</table>

• DRF Alarm Catalog—The following new DRF alarms are added in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Alarm Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRFBackupCompleted</td>
<td>DRF backup completed successfully.</td>
</tr>
<tr>
<td>DRFLocalDeviceError</td>
<td>DRF unable to access local device.</td>
</tr>
<tr>
<td>DRFNoBackupTaken</td>
<td>A valid backup of the current system was not found after an Upgrade, Migration, or Fresh Install.</td>
</tr>
<tr>
<td>DRFRestoreCompleted</td>
<td>DRF restore completed successfully.</td>
</tr>
</tbody>
</table>

• IMS Alarm Catalog—The following IMS alarms are updated in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Alarm Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdminPassword</td>
<td>Severity level is Informational.</td>
</tr>
<tr>
<td>authAdminLock</td>
<td>Severity level is Warning (4).</td>
</tr>
<tr>
<td>authExpired</td>
<td>Added Routing List element and updated the parameter list.</td>
</tr>
<tr>
<td>authFail</td>
<td>Changed severity level from Notice to Warning.</td>
</tr>
<tr>
<td>authHackLock</td>
<td>Updated the parameter list.</td>
</tr>
<tr>
<td>authInactiveLock</td>
<td>Updated the parameter list.</td>
</tr>
<tr>
<td>authLdapInactive</td>
<td>Severity level is Warning (4).</td>
</tr>
<tr>
<td>authMustChange</td>
<td>• Parameter list is updated.</td>
</tr>
<tr>
<td></td>
<td>• Routing List element is added.</td>
</tr>
<tr>
<td>authSuccess</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>credFullUpdateFailure</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>credFullUpdateSuccess</td>
<td>Severity level is Informational (6).</td>
</tr>
</tbody>
</table>
### Chapter 2  New and Changed Information

#### Cisco Unified Communications Manager, Release 8.0(1)

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Alarm Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>credReadFailure</td>
<td>Changed severity level to Notice from Informational. Updated parameter list and added Routing List element.</td>
</tr>
<tr>
<td>credReadSuccess</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>credUpdateFailure</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>credUpdateSuccess</td>
<td>Severity level is Informational (6).</td>
</tr>
</tbody>
</table>

- IpVms Alarm Catalog—The following new IpVms alarm is added:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Alarm Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>kANNAudioFileMissing</td>
<td>Announcement file not found. The announciator was unable to access an announcement audio file. This may be caused by not uploading a custom announcement to each server in the cluster or a locale has not been installed on the server.</td>
</tr>
</tbody>
</table>

**Changed Alarms in IpVms Alarm Catalog**
The following IpVms alarms are changed:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Alarm Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANNDeviceRecoveryCreateFailed</td>
<td>Added Routing List elements and Parameters.</td>
</tr>
<tr>
<td>CFBDeviceRecoveryCreateFailed</td>
<td>Added Routing List elements and Parameters.</td>
</tr>
<tr>
<td>MOHDeviceRecoveryCreateFailed</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>MTPDeviceRecoveryCreateFailed</td>
<td>Changed severity level from Error to Warning and added existing Routing List elements and Parameters.</td>
</tr>
<tr>
<td>SoftwareLicenseNotValid</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>SoftwareLicenseValid</td>
<td>Severity—Informational.</td>
</tr>
<tr>
<td>kANNAudioCreateDirFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Parameter list updated.</td>
</tr>
<tr>
<td>kANNAudioUndefinedAnnID</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Parameter list removed.</td>
</tr>
<tr>
<td>kANNAudioUndefinedLocale</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Parameter list is updated.</td>
</tr>
<tr>
<td>kANNDeviceRecordNotFound</td>
<td>Severity changed from Warning to Error.</td>
</tr>
<tr>
<td>kANNDeviceStartingDefaults</td>
<td>• Severity changed from Informational to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Parameter list added.</td>
</tr>
<tr>
<td>kANNICMPErrorNotification</td>
<td>Parameter list updated.</td>
</tr>
<tr>
<td>kCFBDeviceRecordNotFound</td>
<td>Severity changed from Informational to Error.</td>
</tr>
</tbody>
</table>

---
<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Alarm Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>kCFBDeviceStartingDefaults</td>
<td>• Severity changed from Informational to Warning.</td>
</tr>
<tr>
<td></td>
<td>• New parameters added:</td>
</tr>
<tr>
<td></td>
<td>– Parameter Name(String)</td>
</tr>
<tr>
<td></td>
<td>– Value Used(String)</td>
</tr>
<tr>
<td>kCFBICMPErrorNotification</td>
<td>Following parameters are removed:</td>
</tr>
<tr>
<td></td>
<td>Call ID [ULong] Party ID [ULong] IP Port [ULong]</td>
</tr>
<tr>
<td>kChangeNotifyServiceCreationFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>– OS Error Code(Int)</td>
</tr>
<tr>
<td></td>
<td>– OS Error Description(String)</td>
</tr>
<tr>
<td>kChangeNotifyServiceGetEventFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>– OS Error Code(Int)</td>
</tr>
<tr>
<td></td>
<td>– OS Error Description(String)</td>
</tr>
<tr>
<td>kChangeNotifyServiceRestartFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>– OS Error Code(Int)</td>
</tr>
<tr>
<td></td>
<td>– OS Error Description(String)</td>
</tr>
<tr>
<td>kCreateAudioSourcesFailed</td>
<td>Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>– OS Error Code(Int)</td>
</tr>
<tr>
<td></td>
<td>– OS Error Description(String)</td>
</tr>
<tr>
<td>kCreateControlFailed</td>
<td>Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>– OS Error Code(Int)</td>
</tr>
<tr>
<td></td>
<td>– OS Error Description(String)</td>
</tr>
<tr>
<td>kDeviceDriverError</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kDeviceMgrCreateFailed</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kDeviceMgrExitEventCreationFailed</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kDeviceMgrLockoutWithCallManager</td>
<td>Severity changed from Error to Informational.</td>
</tr>
<tr>
<td>kDeviceMgrMoreThan50SocketEvents</td>
<td>Severity changed from Informational to Notice.</td>
</tr>
<tr>
<td>kDeviceMgrOpenReceiveFailedOutOfStreams</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kDeviceMgrRegisterKeepAliveResponseError</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kDeviceMgrRegisterWithCallManager</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>kDeviceMgrRegisterWithCallManagerError</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kDeviceMgrSocketDrvNotifyEvtCreateFailed</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kDeviceMgrSocketDrvNotifyEvtCreateFailed</td>
<td>Severity changed to Warning from Error.</td>
</tr>
<tr>
<td>Alarm Name</td>
<td>Alarm Changes</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td>kDeviceMgrStartTransmissionOutOfStreams</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kDeviceMgrThreadWaitFailed</td>
<td>• Severity changed from Error to Informational.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>– OS Error Code [Int]</td>
</tr>
<tr>
<td></td>
<td>– OS Error Description [String]</td>
</tr>
<tr>
<td>kDeviceMgrThreadxFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>– OS Error Code [Int]</td>
</tr>
<tr>
<td></td>
<td>– OS Error Description [String]</td>
</tr>
<tr>
<td>kDeviceMgrUnregisterWithCallManager</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>kFixedInputCodecStreamFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters removed:</td>
</tr>
<tr>
<td></td>
<td>– Audio Source ID [ULong]</td>
</tr>
<tr>
<td></td>
<td>– System error code [ULong]</td>
</tr>
<tr>
<td>kFixedInputCreateControlFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Audio Source ID [ULong] parameter is removed.</td>
</tr>
<tr>
<td>kFixedInputCreateSoundCardFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Audio Source ID [ULong] parameter is removed.</td>
</tr>
<tr>
<td>kFixedInputInitSoundCardFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters are removed:</td>
</tr>
<tr>
<td></td>
<td>– Audio Source ID [ULong]</td>
</tr>
<tr>
<td></td>
<td>– System error code [ULong]</td>
</tr>
<tr>
<td>kFixedInputTranscoderFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters are removed:</td>
</tr>
<tr>
<td></td>
<td>– Audio Source ID [ULong]</td>
</tr>
<tr>
<td></td>
<td>– System error code [ULong]</td>
</tr>
<tr>
<td>kGetFileNameFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Audio Source ID [ULong] parameter is removed.</td>
</tr>
<tr>
<td>kIPVMSDeviceDriverNotFound</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
<tr>
<td>kIPVMSMgrEventCreationFailed</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kIPVMSMgrThreadxFailed</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kIPVMSMgrWrongDriverVersion</td>
<td>Following parameters are removed:</td>
</tr>
<tr>
<td></td>
<td>• Found [ULong]</td>
</tr>
<tr>
<td></td>
<td>• Need [ULong]</td>
</tr>
<tr>
<td>Alarm Name</td>
<td>Alarm Changes</td>
</tr>
<tr>
<td>-----------------------------------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>kIPVMSStarting</td>
<td>ProcessID [ULong] parameter is removed.</td>
</tr>
<tr>
<td>kIPVMSStopping</td>
<td>ProcessID [ULong] parameter is removed.</td>
</tr>
<tr>
<td>kIpVmsMgrNoLocalHostName</td>
<td>Severity level is Error (3).</td>
</tr>
<tr>
<td>kIpVmsMgrNoLocalNetworkIPAddr</td>
<td>Severity level is Error (3).</td>
</tr>
<tr>
<td>kIpVmsMgrThreadWaitFailed</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kMOHBadMulticastIP</td>
<td>Severity changed from Warning to Error. Following parameters are removed:</td>
</tr>
<tr>
<td></td>
<td>• Audio Source ID [ULong]</td>
</tr>
<tr>
<td></td>
<td>• Call/Conference ID [ULong]</td>
</tr>
<tr>
<td></td>
<td>• Multicast IP Port [ULong]</td>
</tr>
<tr>
<td>kMOHDeviceRecordNotFound</td>
<td>Severity changed from Informational to Warning.</td>
</tr>
<tr>
<td>kMOHICMPErrorNotification</td>
<td>Following parameters are removed:</td>
</tr>
<tr>
<td></td>
<td>• Call ID [ULong] Party ID [ULong] IP Port [ULong]</td>
</tr>
<tr>
<td>kMOHMgrCreateFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• OS Error Description(String) parameter is added.</td>
</tr>
<tr>
<td>kMOHMgrExitEventCreationFailed</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kMOHMgrIsAudioSourceInUseThisIsNull</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>kMOHMgrThreadWaitFailed</td>
<td>• Severity changed from Error to Informational.</td>
</tr>
<tr>
<td></td>
<td>• OS Error Description(String) parameter is added.</td>
</tr>
<tr>
<td>kMOHMgrThreadxFailed</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• OS Error Description(String) parameter is added.</td>
</tr>
<tr>
<td>kMOHRewindStreamControlNull</td>
<td>• Severity changed from Error to Informational.</td>
</tr>
<tr>
<td></td>
<td>• Audio Source ID [ULong] parameter is removed.</td>
</tr>
<tr>
<td>kMOHRewindStreamMediaPositionObjectNull</td>
<td>• Severity changed from Error to Informational.</td>
</tr>
<tr>
<td></td>
<td>• Audio Source ID [ULong] parameter is removed.</td>
</tr>
<tr>
<td>kMOHTFTPGoRequestFailed</td>
<td>Following parameters added:</td>
</tr>
<tr>
<td>kMTPDeviceRecordNotFound</td>
<td>Severity changed from Informational to Warning.</td>
</tr>
<tr>
<td>kMTPDeviceStartingDefaults</td>
<td>MTP Run Flag(String) parameter is added.</td>
</tr>
<tr>
<td>kPWavMgrThreadxFailed</td>
<td>Severity level is Error (3).</td>
</tr>
<tr>
<td>kReadCfgIpTosMediaResourceToCmNotFound</td>
<td>Severity level is Informational (6).</td>
</tr>
</tbody>
</table>
JavaApplications Alarm Catalog—The following new JavaApplications alarms are added in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>kReadCfgMOHEnabledCodecsNotFound</td>
<td>Severity level is Informational (6).</td>
</tr>
<tr>
<td>kReadCfgUserLocaleEnterpriseSvcParm</td>
<td>Severity level is Error (3).</td>
</tr>
</tbody>
</table>

JavaApplications Alarm Catalog—The following JavaApplication alarms are updated in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>kRequestedANNStreamsFailed</td>
<td>Following parameters are removed: Requested streams [ULong] Allocated streams [ULong]</td>
</tr>
<tr>
<td>kRequestedCFBStreamsFailed</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kRequestedMOHStreamsFailed</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>kRequestedMTPStreamsFailed</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

Login Alarm Catalog—The following Login alarm is updated:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthenticationFailed</td>
<td>Severity Changed from Error to Warning.</td>
</tr>
</tbody>
</table>

LpmTct Alarm catalog—The following LpmTct alarms are changed in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CoreDumpFileFound</td>
<td>Severity level is Critical.</td>
</tr>
<tr>
<td>LogCollectionJobLimitExceeded</td>
<td>Severity changed from Informational to Warning.</td>
</tr>
<tr>
<td>LogFileSearchStringFound</td>
<td>Severity level is Informational.</td>
</tr>
<tr>
<td>LogPartitionHighWaterMarkExceeded</td>
<td>Severity changed from Error to Critical.</td>
</tr>
<tr>
<td>LogPartitionLowWaterMarkExceeded</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>
Cisco Unified Communications Manager Managed Services Guide

Chapter 2  New and Changed Information

Cisco Unified Communications Manager, Release 8.0(1)

- RTMT Alarm Catalog—The following new RTMT alarms are added in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SparePartitionHighWaterMarkExceeded</td>
<td>Severity changed from Error to Warning. Note: Spare Partition is not used for Intercompany Media Engine server. So this alert will not be triggered for Intercompany Media Engine</td>
</tr>
<tr>
<td>SparePartitionLowWaterMarkExceeded</td>
<td>Severity level is Error (3). Note: Spare Partition is not used for Intercompany Media Engine server. So this alert will not be triggered for Intercompany Media Engine</td>
</tr>
</tbody>
</table>

- SystemAccess Alarm catalog—The following System Access Alarms are updated in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TotalProcessesAndThreadsExceeded</td>
<td>Severity changed from Informational to Notice.</td>
</tr>
</tbody>
</table>

- TFTP Alarm catalog—The following TFTP Alarms are updated in the alarm catalog:

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNFFBuffWriteToFileopenfailed</td>
<td>Severity changed from Informational to Error.</td>
</tr>
<tr>
<td>CNFFBuffWriteToFilewritefailed</td>
<td>Severity changed from Informational to Error.</td>
</tr>
<tr>
<td>ConfigItAllBuildFilesFailed</td>
<td>Severity changed from Informational to Error.</td>
</tr>
<tr>
<td>ConfigItAllReadConfigurationFailed</td>
<td>Severity changed from Informational to Error.</td>
</tr>
<tr>
<td>ConfigThreadBuildFileFailed</td>
<td>Severity changed from Informational to Error.</td>
</tr>
<tr>
<td>ConfigThreadCNCMGrpBuildFileFailed</td>
<td>Severity changed from Informational to Error.</td>
</tr>
<tr>
<td>ConfigThreadChangeNotifyServerInstanceFailed</td>
<td>Severity changed from Error to Alert.</td>
</tr>
<tr>
<td>ConfigThreadChangeNotifyServerSingleFailed</td>
<td>Severity changed from Error to Alert.</td>
</tr>
<tr>
<td>ConfigThreadChangeNotifyServerStartFailed</td>
<td>Severity changed from Error to Alert.</td>
</tr>
<tr>
<td>ConfigThreadReadConfigurationFailed</td>
<td>Severity changed from Informational to Error.</td>
</tr>
<tr>
<td>CreateThreadFailed</td>
<td>Severity changed from Error to Alert.</td>
</tr>
<tr>
<td>NoCallManagerFound</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td>SDIControlLayerFailed</td>
<td>Severity changed from Critical to Alert.</td>
</tr>
</tbody>
</table>
For more information on alarms, see Cisco Unified Serviceability Alarms and CiscoLog Messages, page 6-1.

Obsoleted Alarms

The following alarms are obsoleted in this release:

**Call Manager Catalog**
- ConferenceCreated
- ConferenceDeleted
- CtiCallAcceptTimeout
- CtiStaleCallHandle
- DatabaseAuditInfo_074
- DatabaseDeviceNoDirNum
- DatabaseInternalDataError_06e
- DatabaseInternalDataError_06f
- DatabaseInternalDataError_070
- DatabaseInternalDataError_071
- DatabaseInternalDataError_072
- DatabaseInternalDataError_073
- DatabaseInternalDataError_075
- DnTimeout
- GatewayAlarm
- H323AddressResolutionError
- H323CallFailureAlarm
- MWIParamMisMatch
- NoConnection
- OutOfDnForAutoRegistration
- PktCapDownloadFailed
- PktCapDownloadOK
- PktCapLoginFailed
- PktCapLoginOK
- Redirection
- SIP IPPortConflict
- ThrottlingSampleActivity
- TotalCodeYellowEntry

**CertMonitor Alarm Catalog**
- CertExpired
- CertExpiryApproaching
• CertExpiryDebug
• CertExpiryError

CMI Alarm Catalog
• CCMConnectionError
• CMIDebugAlarm
• CMIStarted
• CMIStopped
• COMException
• ConfigParaNotFound
• DisconnectionToCCM
• WSAStartupFailed

CTI Manager Alarm Catalog
• kCtiDeviceOpenFailAccessDenied
• kCtiDirectoryLoginFailure
• kCtiEnvProcDevListRegTimeout
• kCtiExistingCallNotifyArrayOverflow
• kCtiIllegalEnumHandle
• kCtiIllegalFilterSize
• kCtiIllegalQbeHeader
• kCtiInvalidQbeSizeAndOffsets
• kCtiLineCallInfoResArrayOverflow
• kCtiLineOpenFailAccessDenied
• kCtiMYTCPSendError
• kCtiMytcpErrSocketBroken
• kCtiNewCallNotifyArrayOverflow
• kCtiNullTcpHandle
• kCtiProviderOpenInvalidUserNameSize
• kCtiQbeLengthMisMatch
• kCtiQbeMessageTooLong
• kCtiSdlErrorvException
• kCtiSsRegisterManagerErr
• kCtiTcpInitError
• kCtiUnknownConnectionHandle

DB Alarm Catalog
• ErrorChangeNotifyReconcile
IpVms Alarm Catalog

- kANNAudioComException
- kANNAudioOpenFailed
- kANNAudioTftpFileMissing
- kANNAudioTftpMgrCreate
- kANNAudioTftpMgrStartFailed
- kANNAudioThreadException
- kANNAudioThreadWaitFailed
- kANNAudioThreadxFailed
- kANNAudioXmlLoadFailed
- kANNAudioXmlSyntax
- kAddIpVmsRenderFailed
- kCfgListComException
- kCfgListDblException
- kCfgListUnknownException
- kCreateGraphManagerFailed
- kDeviceMgrThreadException
- kDownloadMOHFileFailed
- kFixedInputAddAudioCaptureDeviceFailed
- kFixedInputAddG711AlawIpVmsRenderFailed
- kFixedInputAddG711UlawIpVmsRenderFailed
- kFixedInputAddG729IpVmsRenderFailed
- kFixedInputAddMOHEncoderFailed
- kFixedInputAddWideBandIpVmsRenderFailed
- kFixedInputAudioCapMOHEncoderConnFailed
- kFixedInputAudioCaptureCreateFailed
- kFixedInputClassEnumeratorCreateFailed
- kFixedInputCreateGraphManagerFailed
- kFixedInputFindAudioCaptureDeviceFailed
- kFixedInputGetEventNotificationFailed
- kFixedInputGetNameFailed
- kFixedInputGetG711AlawIpVmsRendInfFailed
- kFixedInputGetG711AlawIpVmsRenderFailed
- kFixedInputGetG711UlawIpVmsRendInfFailed
- kFixedInputGetG711UlawIpVmsRenderFailed
- kFixedInputGetG729IpVmsRendInfFailed
- kFixedInputGetG729IpVmsRenderFailed
- kFixedInputGetMOHEncoderFailed
- kFixedInputGetMediaControlFailed
- kFixedInputGetMediaPositionFailed
- kFixedInputGetWideBandIpVmsRendInfFailed
- kFixedInputGetWideBandIpVmsRenderFailed
- kFixedInputMOHEncG711A lawRenderConnFailed
- kFixedInputMOHEncG711UlawRenderConnFailed
- kFixedInputMOHEncG729RenderConnFailed
- kFixedInputMOHEncWidebandRenderConnFailed
- kFixedInputSetNotifyWindowFailed
- kGetEventNotificationFailed
- kGetIpVmsRenderFailed
- kGetIpVmsRenderInterfaceFailed
- kGetMediaControlFailed
- kGetMediaPositionFailed
- kMOHFilterNotifyError
- kMOHMgrThreadCreateWindowExFailed
- kMOHPlayStreamControlNull
- kMOHPlayStreamMediaControlObjectNull
- kMOHThreadException
- kMTPICMPErrorNotification
- kPWavMgrExitEventCreateFailed
- kPWavMgrThreadException
- kReadCfgANNComException
- kReadCfgANNDbIException
- kReadCfgANNListComException
- kReadCfgANNListDbIException
- kReadCfgANNListUnknownException
- kReadCfgANNUnknownException
- kReadCfgCFBComException
- kReadCfgCFBDblException
- kReadCfgCFBListComException
- kReadCfgCFBListDbIException
- kReadCfgCFBListUnknownException
- kReadCfgCFBUne wnownException
- kReadCfgDbIGetChgNotifyFailed
- kReadCfgDbIGetNodeNameFailed
- kReadCfgEnterpriseComException
- kReadCfgEnterpriseDbIException
New Perfmon Counters

New perfmon counters are added for the following objects:
• Cisco CallManager External Call Control—This feature provides information about the counters that are added to support the External Call Control feature. Table 2-1 contains information about the External Call Control counters.

Table 2-6 Cisco CallManager External Call Control

<table>
<thead>
<tr>
<th>Counts</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cisco CallManager Object</strong></td>
<td></td>
</tr>
<tr>
<td>ExternalCallControlEnabledCallsAttempted</td>
<td>This counter specifies the total number of calls to devices that have the External Call Control feature enabled. This is a cumulative count of all calls to intercept-enabled patterns or DNs since the last restart of the Cisco CallManager service.</td>
</tr>
<tr>
<td>ExternalCallControlEnabledCallsCompleted</td>
<td>This counter specifies the total number of calls that were connected to a device that had the External Call Control feature enabled. This is a cumulative count of all calls to intercept-enabled patterns or DNs since the last restart of the Cisco CallManager service.</td>
</tr>
<tr>
<td>ExternalCallControlEnabledFailureTreatmentApplied</td>
<td>This counter specifies the total number of calls that were cleared or routed based on failure treatments (such as Allow or Deny) that are defined in the External Call Control profile.</td>
</tr>
<tr>
<td><strong>External Call Control Objects</strong></td>
<td></td>
</tr>
<tr>
<td>PDP Servers Total</td>
<td>This counter defines the total number of PDP servers in all External Call Control Profiles configured in Cisco Unified CM Administration. This counter increments when a new PDP server is added and decrements when a PDP server is removed.</td>
</tr>
<tr>
<td>PDP Servers In Service</td>
<td>This counter defines the total number of in-service (active) PDP servers.</td>
</tr>
<tr>
<td>PDP Servers Out Of Service</td>
<td>This counter defines the total number of times that PDP servers have transitioned from in-service to out-of-service. This is a cumulative count of out-of-service PDP servers since the last restart of the Cisco CallManager service.</td>
</tr>
<tr>
<td>Connections Active To PDP Server</td>
<td>This counter specifies the total number of connections that Cisco Unified Communications Manager has established (currently active) with PDP servers.</td>
</tr>
<tr>
<td>Connections Lost To PDP Server</td>
<td>This counter specifies the total number of times that active connections between Cisco Unified Communications Manager and the PDP servers were disconnected. This is a cumulative count since the last restart of the Cisco CallManager service.</td>
</tr>
</tbody>
</table>

• Cisco CallManager SAF—The Cisco SAF Client object provides information about SAF counters that are specific to each node. Table 2-7 describes Cisco SAF Client object counters.
Table 2-7  Cisco CallManager SAF Client Object

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAFConnectionsSucceeded (range from 0 to 2)</td>
<td>Total number of SAF client connections currently active on this Unified CM node.</td>
</tr>
<tr>
<td>SAFFConnectionsFailed (range from 0 to 2)</td>
<td>Total number of SAF client connections that failed on the Unified CM node. A failed connection is a connection that did not register with the SAF Forwarder.</td>
</tr>
</tbody>
</table>

A Cisco Unified CM node restart causes a counter reset.

- Cisco Extension Mobility—The Cisco Extension Mobility object provides information about the extension mobility application. Table 2-8 contains information about the newly added Cisco Extension Mobility counters.

Table 2-8  Cisco Extension Mobility Application

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Number of EMCC Messages</td>
<td>This represents the total number of messages related to EMCC Requests that came from remote clusters.</td>
</tr>
<tr>
<td>Number of Remote Devices</td>
<td>This represents the total number of devices from other clusters that are currently using a EMCC Base Device (EMCC Logged in).</td>
</tr>
<tr>
<td>Number of Unknown Remote Users</td>
<td>This represents the total number of users who were not found in any of the remote cluster during inter-cluster extension mobility login.</td>
</tr>
<tr>
<td>Active Inter-cluster Sessions</td>
<td>This represents the total number of inter cluster Extension Mobility requests that are currently in progress.</td>
</tr>
<tr>
<td>Total Number of Remote Users</td>
<td>This represents the total number of users from other cluster who use a local device of this cluster and have logged into a remote cluster.</td>
</tr>
<tr>
<td>EMCC Check User Requests Handled</td>
<td>This represents the total number of EMCC check user requests that came from remote clusters.</td>
</tr>
</tbody>
</table>

- Cisco Feature Control Policy—The Cisco Feature Control feature provides information about the two new counters for TFTP. Table 2-9 contains information about the newly added Cisco Feature Control Policy feature counters.

Table 2-9  Cisco Feature Control Policy

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BuildFeaturePolicyCount</td>
<td>Indicates the number of built FCP files</td>
</tr>
<tr>
<td>FeaturePolicyChangeNotifications</td>
<td>Indicates the number of sent FCP change notifications</td>
</tr>
</tbody>
</table>

- Cisco IME Server—The Cisco IME Server provides information about the Performance Object and Counters for IME.
The following contains the Performance Object for Cisco IME Server:

VAPStatus (range from 0 to 2)—This flag indicates the overall health of the connection to the IME servers for a particular IME service. If 1, it means that Unified CM has successfully established a connection to its primary and, if configured, backup servers for the IME service. 2 = Unhealthy. 0 = Unknown.

- The following contains the Performance Counters for Cisco IME Server. Table 2-10 contains information about the Performance Counters for Cisco IME Server.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PublishedRoutes</td>
<td>Total number of DIDs published successfully into the DHT across all IME services. It is a dynamic measurement, and as such, gives you an indication of your own provisioned usage in addition to a sense of how successful the system has been in storing them into the network.</td>
</tr>
<tr>
<td>RejectedRoutes</td>
<td>Number of learned routes which were rejected because the number or domain were blacklisted by the administrator. This provides an indication of the number of 'missed opportunities' - cases where a VoIP call could happen in the future, but will not due to the blocked validation.</td>
</tr>
<tr>
<td>LearnedRoutes</td>
<td>Total number of distinct phone numbers which have been learned by IME and are present as routes in Unified CM's routing tables. If this number grows too large, it may exceed the per-cluster limit, and require additional clusters for scale.</td>
</tr>
<tr>
<td>UniqueDomains</td>
<td>Number of unique domain names of peer enterprises discovered by IME. It is an indicator of overall usage of the system.</td>
</tr>
<tr>
<td>FailedB2BLinkSetups</td>
<td>Total number of call attempts for which a IME route was available, but which were set up through the PSTN due to a failure to connect to the target over the IP network.</td>
</tr>
<tr>
<td>B2BLinkCallsAttempted</td>
<td>Number of calls initiated by UCM through IME. This includes calls that are accepted, as well as busy, no-answer and failed calls. The metric is strictly on initiation.</td>
</tr>
<tr>
<td>B2BLinkCallsSetup</td>
<td>Number of IME calls successfully placed by Unified CM and answered by the remote party, resulting in an IP call.</td>
</tr>
<tr>
<td>FailedFallbackCalls</td>
<td>Total number of failed fallback attempts.</td>
</tr>
<tr>
<td>e164 DIDs Learned</td>
<td>Number of DIDs learned from the IME server.</td>
</tr>
<tr>
<td>B2BLinkCallsAccepted</td>
<td>Number of IME calls successfully received by UCM and answered by the called party, resulting in an IP call.</td>
</tr>
<tr>
<td>B2BLinkCallsReceived</td>
<td>Number of calls received by Unified CM through IME. This includes calls that are accepted, as well as busy, no-answer and failed calls. The metric is strictly on initiation.</td>
</tr>
</tbody>
</table>

For more information, see the Cisco Unified Real-Time Monitoring Tool Administration Guide.
Cisco Unified CDR Analysis and Reporting

The functionality of Call Detail Records (CDR) Analysis and Reporting (CAR) is primarily to generate reports on Unified CM users and system status with respect to call processing records that are loaded to CAR database. CAR also does some CAR database management activities. CAR automatically schedules required tasks to take place or you can manually perform the tasks by using the web interface.

This section contains the following subsections:

- New Cisco CAR DB Alarms, page 2-51
- New CAR Object and Counters, page 2-51
- Hunt/CTI Integration for CAR Reporting, page 2-52
- CAR and CDRM Alarm Interface, page 2-52
- System-Wide Call Tracking End-to-End Call Trace, page 2-52

New Cisco CAR DB Alarms

New alarms for the CAR DB instance separation get added in this release. A new thread of [CARIDSAlarm] gets created in the existing CAR Scheduler Service to receive the IDS alarms. There are four new categories and alarms with information specific to the IDS based on the class IDs.

The following new alarms support the CAR database instance:

- **CARIDSEngineDebug**—Indicates debug events from CAR IDS database engine. This alarm provides low-level debugging information from CAR IDS database engine. System administrator can disregard this alarm. Severity level is Debug(7).
- **CARIDSEngineInformation**—No error has occurred but some routine event completed in CAR IDS database engine. Severity level is Informational(6).
- **CARIDSEngineCritical**—This alarm does not compromise data or prevent the use of the system but does required attention. Severity level is Critical(2).
- **CARIDSEngineFailure**—Combined alarm for emergency and error situations. Something unexpected occurred that might compromise data or access to data or cause CAR IDS to fail. Severity level is Error(3).

Note

For any alarms with severity levels at or higher than Critical, an alert gets automatically generated.

For more information, see *Cisco Unified CDR Analysis and Reporting Guide*.

New CAR Object and Counters

The new CAR counters monitor the CAR database space and shared memory usage. The following CAR counters for the Cisco CAR DB object get supported:

- **RootDBSpaceUsed**—Percentage of Root DB space consumed. The root DB space gets used by the IDS system tables in the CAR IDS instance.
- **CARDBSpaceUsed**—Percentage of CAR DB space consumed. The CAR DB space gets used by the CAR database.
- **CARTempDBSpaceUsed**—Percentage of CAR temporary DB space consumed. The CAR temporary DB space gets used by temporary tables in the CAR IDS instance and used by CAR applications.
• FreeSharedMemory—Total free and shared memory expressed in kilobytes (KB). Shared memory gets used by the database system and all database applications in the CAR IDS instance.

• UsedSharedMemory—Total used and shared memory expressed in kilobytes (KB). Shared memory gets used by the database system and all database applications in the CAR IDS instance.

There are no performance counters that monitor the CAR IDS processes individually because the counters get automatically added for each new process. The counters get implemented with a new thread/job of the CAR IDS performance in the existing CAR Scheduler service by using Java API (JNI based statsUpdate()).

Hunt/CTI Integration for CAR Reporting

CAR supports hunt groups and contains the following new reports:

• Hunt Pilot Summary
• Hunt Pilot Detailed Report

Hunt Pilot Summary

Only CAR administrators generate the Hunt Pilot Summary Report. The CDR Hunt Pilot Call Summary report displays the call details for the specified hunt pilot. This report displays only an overview of the calls for the hunt pilots and hunt member information is not included. The CAR administrator can generate report for a maximum of five hunt pilot DNs.

Hunt Pilot Detailed Report

Only CAR administrators generate the Hunt Pilot Detailed Call Report. This report displays call details for a hunt pilot number or a hunt member DN.

CAR and CDRM Alarm Interface

CAR and CDRM allow the alarm interface to raise alerts. The alarm interface can generate Syslog events, SNMP traps, and e-mail notifications by using RIS/Collector/Alert Manager. CAR allows the performance interface to poll serviceability counters and to be monitored in Cisco Unified Real Time Monitoring Tool.

System-Wide Call Tracking End-to-End Call Trace

The End-to-End Call Trace feature facilitates tracing calls that traverse multiple Cisco voice products, such as Unified CM, Cisco IOS Gateways, and other products.

There are four new CDR fields added: CAR Loader, schema, CDR export, CDR search reports and migration.

For more information about System-Wide Call Tracking (SCT), see End-to-End Call Trace, page 2-53.

Cisco Unified Call Detail Records

This feature traces calls that traverse multiple Cisco voice products by using the call records collected from each platform generated for the same call.

This section contains information on the following topics:
• End-to-End Call Trace, page 2-53
• Remote Destination to Number Mapping and CDRs, page 2-53
• New CDR Fields to Support Call Control Discovery, page 2-53
• New CDR Fields to Support External Call Control, page 2-53
• New CDR Support for iSAC Codec, page 2-55
• New CDR Fields for Hunt List Support, page 2-55

End-to-End Call Trace

To support the End-to-End call trace, following new fields have been added in the CDR search reports:

- IncomingProtocolID
- IncomingProtocolCallRef
- OutgoingProtocolID
- OutgoingProtocolCallRef

Remote Destination to Number Mapping and CDRs

For an outgoing call to mobile users, the called party information in the CDR gets recorded based on the “Log Mobile Number in CDR” service parameter. The default equals False. If the service parameter is False, the enterprise number of the mobile user gets recorded in the CDR as the called party number. If the service parameter equals True, the mobile number gets recorded in CDR as the called party number.

New CDR Fields to Support Call Control Discovery

New codes display for the call control discovery feature, as described in Table 2-11. (For more information on call control discovery, see Cisco Unified CDR Guide.)

<table>
<thead>
<tr>
<th>Value</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>464</td>
<td>Redirect Reason Code</td>
<td>Indicates that the call is redirected to a PSTN failover number</td>
</tr>
<tr>
<td>131</td>
<td>Call Termination Code</td>
<td>Call Control Discovery PSTN Failover (Cisco specific)</td>
</tr>
<tr>
<td>29</td>
<td>OnBehalfof Code</td>
<td>CCDRequestingService</td>
</tr>
</tbody>
</table>

New CDR Fields to Support External Call Control

Table 2-12 describes the new CDR fields for the external call control feature. Use Table 2-12 in conjunction with the Table 2-13, which describes the routing reason values that are specific to external call control. (For more information on external call control, see Cisco Unified CDR Guide.)
Table 2-12  CDR Fields for External Call Control

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Range of Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentRoutingReason</td>
<td>Positive Integer</td>
<td>This field, which is used with the external call control feature, displays the reason why the call was intercepted for the current call. For a list of reasons, see Table 2-13. Default value is 0.</td>
</tr>
<tr>
<td>origRoutingReason</td>
<td>Positive Integer</td>
<td>This field, which is used with the external call control feature, displays the reason why the call was intercepted for the first time. For a list of reasons, see Table 2-13. Default value is 0.</td>
</tr>
<tr>
<td>lastRedirectingRoutingReason</td>
<td>Positive Integer</td>
<td>This field, which is used with the external call control feature, displays why the call was intercepted for the last time. For a list of reasons, see Table 2-13. Default - Empty string.</td>
</tr>
</tbody>
</table>

Table 2-13 includes the reasons that can display for the currentRoutingReason, origRoutingReason, or lastRedirectingRoutingReason fields.

Table 2-13  Routing Reason Values for External Call Control

<table>
<thead>
<tr>
<th>Value that Displays in the Field</th>
<th>Reason</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>PDPDecision_NONE</td>
<td>This value indicates that the route server did not return a routing directive to the Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>1</td>
<td>PDPDecision_Allow_Fulfilled</td>
<td>This value indicates that Cisco Unified Communications Manager allowed a call.</td>
</tr>
<tr>
<td>2</td>
<td>PDPDecision_Allow_Unfulfilled</td>
<td>This value indicates that Cisco Unified Communications Manager disallowed a call.</td>
</tr>
<tr>
<td>3</td>
<td>PDPDecision_Divert_Fulfilled</td>
<td>This value indicates that Cisco Unified Communications Manager diverted the call.</td>
</tr>
<tr>
<td>4</td>
<td>PDPDecision_Divert_Unfulfilled</td>
<td>This value indicates that Cisco Unified Communications Manager was not able to divert the call.</td>
</tr>
<tr>
<td>5</td>
<td>PDPDecision_Forward_Fulfilled</td>
<td>This value indicates that Cisco Unified Communications Manager forwarded the call.</td>
</tr>
</tbody>
</table>
CAR supports the new fields from the loader, CDR export, and CDR search reports on display and migration.

**New CDR Support for iSAC Codec**

The codec fields can now support the iSAC (Media_Payload_ISAC) with the value of 89.

**New CDR Fields for Hunt List Support**

Table 2-14 describes the new CDRs for the hunt list support (see Cisco Unified CDR Guide for more information).

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Range of Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>huntPilotDN</td>
<td>Text String</td>
<td>This field indicates the hunt pilot DN through which the call is routed. Default - Empty string.</td>
</tr>
<tr>
<td>huntPilotPartition</td>
<td>Text String</td>
<td>This field indicates the partition for the hunt pilot DN. Default - Empty string.</td>
</tr>
<tr>
<td>huntPilotDN</td>
<td>Text String</td>
<td>This field indicates the hunt pilot DN through which the call is routed. Default - Empty string.</td>
</tr>
</tbody>
</table>

**Cisco Unified Reporting**

There are no updates for Cisco Unified Reporting Guide in the Release 8.0(1).
## MIB Updates for 8.0(1)

Table 2-15 lists the deprecated and replaced MIBs.

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deprecated</td>
<td>CcmDevFailCauseCode; Added CcmDevRegFailCauseCode and CcmDevUnregCauseCode</td>
</tr>
<tr>
<td>Deprecated</td>
<td>ccmPhoneStatusReason; Added ccmPhoneUnregReason and ccmPhoneRegFailReason in ccmPhoneTable</td>
</tr>
<tr>
<td>Deprecated</td>
<td>ccmPhoneFailCauseCode; Added ccmPhoneFailedRegFailReason in ccmPhoneFailedTable</td>
</tr>
<tr>
<td>Deprecated</td>
<td>ccmPhoneStatusUpdateReason; Added ccmPhoneStatusUnregReason and ccmPhoneStatusRegFailReason in ccmPhoneStatusUpdateTable</td>
</tr>
<tr>
<td>Deprecated</td>
<td>ccmGatewayStatusReason; Added ccmGatewayUnregReason and ccmGatewayRegFailReason in ccmGatewayTable.</td>
</tr>
<tr>
<td>Deprecated</td>
<td>ccmMediaDeviceStatusReason; Added ccmMediaDeviceUnregReason and ccmMediaDeviceRegFailReason in ccmMediaDeviceTable.</td>
</tr>
<tr>
<td>Deprecated</td>
<td>ccmCTIDeviceStatusReason; Added ccmCTIDeviceUnregReason and ccmCTIDeviceRegFailReason in ccmCTIDeviceTable</td>
</tr>
<tr>
<td>Deprecated</td>
<td>ccmH323DevStatusReason; Added ccmH323DevUnregReason and ccmH323DevRegFailReason in ccmH323DeviceTable.</td>
</tr>
<tr>
<td>Deprecated</td>
<td>ccmVMailDevStatusReason; Added ccmVMailDevUnregReason and ccmVMailDevRegFailReason in ccmVoiceMailDeviceTable.</td>
</tr>
<tr>
<td>Deprecated</td>
<td>ccmGatewayFailCauseCode; Added ccmGatewayRegFailCauseCode in ccmNotificationsInfo.</td>
</tr>
<tr>
<td>Deprecated the following Notification Type</td>
<td>ccmGatewayFailed and added ccmGatewayFailedReason.</td>
</tr>
</tbody>
</table>
### Table 2-15  Updated MIBs (continued)

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deprecated following MODULE-COMPLIANCE</td>
<td>ciscoCcmMIBComplianceRev6; Added ciscoCcmMIBComplianceRev7.</td>
</tr>
<tr>
<td>Obsoleted following OBJECT_GROUPS</td>
<td>ccmInfoGroupRev3, ccmH323DeviceInfoGroupRev1</td>
</tr>
</tbody>
</table>
Managing and Monitoring the Health of Cisco Unified Communications Manager Systems

This chapter describes how to manage and monitor the health of Cisco Unified Communications Manager (Cisco Unified CM) systems. It contains the following sections:

- **Overview of Supported Interfaces, page 3-1**
- **Critical Processes to Monitor, page 3-2**
- **Available Supported MIBs, page 3-11**
- **RTMT Monitoring of Cisco Unified CM System Health, page 3-12**
- **Recovery, Hardware Migration, and Backup/Restore, page 3-26**
- **Platform Monitoring, page 3-27**
- **Software Configuration Management, page 3-32**
- **Available Reports, page 3-33**
- **General Health and Troubleshooting Tips, page 3-35**
- **Related Documentation, page 3-44**

**Note**
Serviceability APIs (AXL/SOAP) that are used for serviceability queries and Administrative XML (AXL) that are used as a provisioning read and write APIs are not covered in this document.

**Overview of Supported Interfaces**

The following interfaces are supported on Cisco Unified CM servers:

- **SNMP MIB/Trap**—Supports polling and traps by using select MIBs from Cisco and the native platforms.
- **SSH Secure Shell Client**—Replaces telnet and ftp clients by using a more secure protocol. This application encrypts the entire network session and can use public-key authentication.
- **Local and Remote Syslog**—Contains types of platform and Cisco Unified CM application events, alerts, and alarms are written to syslog servers.
- **HTTPS**—Displays the following web pages by using HTTPS—Cisco Unified CM Administration, Cisco Unified Serviceability, Disaster Recovery System, and Unified OS Administration.
Critical Processes to Monitor

Table 3-1 describes the critical processes that require monitoring. Be aware of following items while monitoring the processes:

• Any of the services, process names, or process sets could change at any time with newer Cisco Unified CM releases without notice.

• HOST-RESOURCES-MIB could be deprecated in any future Cisco Unified CM release.
Chapter 3  Managing and Monitoring the Health of Cisco Unified Communications Manager Systems

Critical Processes to Monitor

- Whether a process is auto-restarted or the maximum number of restarts could change for any newer Cisco Unified CM releases without notice.
- Process names represent value shown in HOST-RESOURCES-MIB::hrSWRUNName.
- Any processes not included in this list are transient or not critical for system operation. Those processes should be ignored and they can change without notice.
- Services Cisco CallManager through Cisco CDR Agent can be monitored by using SYSAPPL-MIB.

Table 3-1  Critical Services to Monitor

<table>
<thead>
<tr>
<th>Service</th>
<th>Stop</th>
<th>Start Restart Instruction</th>
<th>Process Name</th>
<th>Auto Restart</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco CallManager</td>
<td>Stop</td>
<td>Start Restart Instruction</td>
<td>ccm</td>
<td>3</td>
<td>The Cisco CallManager service provides software-only call processing as well as signaling and call control functionality for Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>Cisco TFTP</td>
<td>Stop</td>
<td>Start Restart Instruction</td>
<td>ctftp</td>
<td>3</td>
<td>The Cisco Trivial File Transfer Protocol (TFTP) builds and serves files that are consistent with the trivial file transfer protocol, a simplified version of FTP. Cisco TFTP serves embedded component executables, ringer files, and device configuration files.</td>
</tr>
<tr>
<td>Cisco IP Voice Media Streaming App</td>
<td>Stop</td>
<td>Start Restart Instruction</td>
<td>ipvmsd</td>
<td>3</td>
<td>The Cisco IP Voice Media Streaming Application service provides voice media streaming functionality for the Cisco Unified CallManager for use with MTP, conferencing, music on hold (MOH), and announcator. The Cisco IP Voice Media Streaming Application relays messages from the Cisco Unified CallManager to the IP voice media streaming driver, which handles RTP streaming.</td>
</tr>
<tr>
<td>Cisco CTIManager</td>
<td>Stop</td>
<td>Start Restart Instruction</td>
<td>CTI Manager</td>
<td>3</td>
<td>The CTI Manager contains the CTI components that interface with applications. With CTI Manager, applications can access resources and functionality of all Cisco Unified CallManagers in the cluster and have improved failover capability. Although one or more CTI Managers can be active in a cluster, only one CTI Manager can exist on an individual server. An application (JTAPI/TAPI) can have simultaneous connections to multiple CTI Managers; however, an application can only use one connection at a time to open a device with media termination.</td>
</tr>
</tbody>
</table>
### Critical Services to Monitor (continued)

<table>
<thead>
<tr>
<th>Service</th>
<th>Stop</th>
<th>Start</th>
<th>Process Name</th>
<th>Auto Restart</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco DHCP Monitor Service</td>
<td></td>
<td></td>
<td>DHCP Monitor</td>
<td>3</td>
<td>Cisco DHCP Monitor Service monitors IP address changes for IP phones in the database tables. When a change is detected, it modifies the /etc/dhcpd.conf file and restarts the DHCPD daemon.</td>
</tr>
<tr>
<td>Cisco CallManager SNMP Service</td>
<td></td>
<td></td>
<td>ccmAgt</td>
<td>3</td>
<td>This service provides SNMP access to provisioning and statistics information that is available for Cisco Unified CallManager.</td>
</tr>
<tr>
<td>Cisco CTL Provider Service Status</td>
<td></td>
<td></td>
<td>CTL Provider</td>
<td>3</td>
<td>The Cisco CTL Provider service, which runs with local system account privileges, works with the Cisco CTL Provider Utility, a client-side plug-in, to change the security mode for the cluster from nonsecure to mixed mode. When you install the plug-in, the Cisco CTL Provider service retrieves a list of all Cisco Unified CallManager and Cisco TFTP servers in the cluster for the CTL file, which contains a list of security tokens and servers in the cluster.</td>
</tr>
<tr>
<td>Cisco Certificate Authority Proxy Function</td>
<td></td>
<td></td>
<td>capf</td>
<td>3</td>
<td>Working in conjunction with the CAPF application, the Cisco Certificate Authority Proxy Function (CAPF) service can perform the following tasks, depending on your configuration—(1) Issue locally significant certificates to supported Cisco Unified IP Phone models. (2) Using SCEP, request certificates from third-party certificate authorities on behalf of supported Cisco Unified IP Phone models. (3) Upgrade existing certificates on the phones. (4) Retrieve phone certificates for troubleshooting. (5) Delete locally significant certificates on the phone.</td>
</tr>
<tr>
<td>Cisco DirSync</td>
<td></td>
<td></td>
<td>CCM DirSync</td>
<td>3</td>
<td>Unlike Windows versions of Cisco Unified CallManager, Cisco Unified CallManager does not contain an embedded directory. Because of this change, the Cisco Unified CallManager database stores all user information. If you use an integrated corporate directory, for example, Microsoft Active Directory or Netscape/iPlanet Directory, with Cisco Unified CallManager, the Cisco DirSync service migrates the user data to the Cisco Unified CallManager database. The Cisco DirSync service does not synchronize the passwords from the corporate directory.</td>
</tr>
</tbody>
</table>
### Critical Processes to Monitor (continued)

<table>
<thead>
<tr>
<th>Service</th>
<th>Stop</th>
<th>Start</th>
<th>Process Name</th>
<th>Auto Restart</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Messaging Interface</td>
<td></td>
<td></td>
<td>cmi</td>
<td>3</td>
<td>The Cisco Messaging Interface allows you to connect a simplified message desk interface (SMDI)-compliant external voice-messaging system with the Cisco Unified CallManager. The CMI service provides the communication between the voice-messaging system and Cisco Unified CallManager. The SMDI defines a way for a phone system to provide a voice-messaging system with the information that is needed to intelligently process incoming calls.</td>
</tr>
<tr>
<td>Cisco CallManager Attendant Console Server</td>
<td></td>
<td></td>
<td>acserver</td>
<td>3</td>
<td>The Cisco CallManager Attendant Console Server service provides centralized services for Cisco Unified CallManager Attendant Console clients and pilot points. For Attendant Console clients, this service provides call-control functionality, line state information for any accessible line within the Cisco Unified CallManager domain, and caching of directory information. For pilot points, this service provides automatic redirection to directory numbers that are listed in hunt groups and failover during a Cisco Unified CallManager failure.</td>
</tr>
<tr>
<td>Cisco Extended Functions</td>
<td></td>
<td></td>
<td>cef</td>
<td>3</td>
<td>The Cisco Extended Functions service provides support for some Cisco Unified CallManager features, including Quality Report Tool (QRT).</td>
</tr>
<tr>
<td>Cisco Bulk Provisioning Service</td>
<td></td>
<td></td>
<td>BPS</td>
<td>3</td>
<td>You can activate the Cisco Bulk Provisioning Service only on the first node. If you use the Cisco Unified Bulk Administration Tool (BAT) to administer phones and users, you must activate this service.</td>
</tr>
<tr>
<td>Cisco TAPS Service</td>
<td></td>
<td></td>
<td>TAPS</td>
<td>3</td>
<td>The Cisco TAPS Service supports the Cisco Unified CallManager Auto-Registered Phone Tool, which allows a user to upload a customized configuration on an autoregistered phone after a user responds to Interactive Voice Response (IVR) prompts.</td>
</tr>
</tbody>
</table>
### Critical Processes to Monitor (continued)

<table>
<thead>
<tr>
<th>Service</th>
<th>Stop</th>
<th>Start</th>
<th>Process Name</th>
<th>Auto Restart</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco CAR Scheduler</td>
<td></td>
<td></td>
<td>carschlr</td>
<td></td>
<td>The Cisco CAR Scheduler service allows you to schedule CAR-related tasks; for example, you can schedule report generation or CDR file loading into the CAR database. This service starts automatically.</td>
</tr>
<tr>
<td>Cisco AMC Service</td>
<td></td>
<td></td>
<td>amc</td>
<td>3</td>
<td>Used for the real-time monitoring tool (RTMT), this service, Alert Manager and Collector service, existed as a component of the Cisco RIS Data Collector service in previous Windows releases of Cisco Unified CallManager. This service allows RTMT to retrieve real-time information that exists on nodes in the cluster.</td>
</tr>
<tr>
<td>Cisco Trace Collection Service</td>
<td></td>
<td></td>
<td>tracecoll</td>
<td>3</td>
<td>The Cisco Trace Collection Service, along with the Cisco Trace Collection Servlet, supports trace collection and allows users to view traces by using the RTMT client. After Cisco Unified CallManager installation, this service starts automatically. If you stop this service on a server, you cannot collect or view traces on that server.</td>
</tr>
<tr>
<td>A Cisco DB Replicator</td>
<td></td>
<td></td>
<td>dblrpc</td>
<td>3</td>
<td>The A Cisco DB Replicator service ensures database configuration and data synchronization between the first and subsequent nodes in the cluster.</td>
</tr>
<tr>
<td>Cisco Tomcat</td>
<td></td>
<td></td>
<td>tomcat</td>
<td>3</td>
<td>The Cisco Tomcat service supports the web server.</td>
</tr>
<tr>
<td>SNMP Master Agent</td>
<td></td>
<td></td>
<td>snmpdm</td>
<td>3</td>
<td>This service, which acts as the agent protocol engine, provides authentication, authorization, access control, and privacy functions that relate to SNMP requests.</td>
</tr>
<tr>
<td>MIB2 Agent</td>
<td></td>
<td></td>
<td>mib2agt</td>
<td>3</td>
<td>This service provides SNMP access to variables that are defined in RFC 1213, which read and write variables; for example, system, interfaces, IP, and so on.</td>
</tr>
<tr>
<td>Host Resources Agent</td>
<td></td>
<td></td>
<td>hostagt</td>
<td>3</td>
<td>This service provides SNMP access to host information, such as storage resources, process tables, device information, and installed software base.</td>
</tr>
</tbody>
</table>
### Critical Services to Monitor (continued)

<table>
<thead>
<tr>
<th>Service</th>
<th>Stop</th>
<th>Start</th>
<th>Process Name</th>
<th>Auto Restart</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Native Agent Adapter</td>
<td></td>
<td></td>
<td>naaagt</td>
<td>3</td>
<td>This service allows you to forward SNMP requests to another SNMP agent that runs on the system.</td>
</tr>
<tr>
<td>System Application Agent</td>
<td></td>
<td></td>
<td>sappagt</td>
<td>3</td>
<td>This service provides SNMP access to the applications that are installed and executing on the system. This implements the SYSAPPL-MIB.</td>
</tr>
<tr>
<td>Cisco CDP Agent</td>
<td></td>
<td></td>
<td>cdpAggt</td>
<td>3</td>
<td>This service uses the Cisco Discovery Protocol to provide SNMP access to network connectivity information on the Cisco Unified CallManager node.</td>
</tr>
<tr>
<td>Cisco Syslog Agent</td>
<td></td>
<td></td>
<td>Cisco Syslog SubA</td>
<td>3</td>
<td>This service supports gathering of syslog messages that various Cisco Unified CallManager components generate.</td>
</tr>
<tr>
<td>Cisco License Manager</td>
<td></td>
<td></td>
<td>Cisco License Mgr</td>
<td>3</td>
<td>Cisco License Manager keeps track of the licenses that a customer purchases and uses. It controls licenses checkins and checkouts, and it takes responsibility for issuing and reclaiming licenses. Cisco License Manager manages the Cisco Unified CallManager application and the number of IP phone unit licenses. When the number of phones exceeds the number of licenses, it issues alarms to notify the administrator. This service runs on all the nodes, but the service on the first node has the responsibility for issuing and reclaiming licenses.</td>
</tr>
<tr>
<td>Cisco Certificate Expiry Monitor</td>
<td></td>
<td></td>
<td>certM</td>
<td>3</td>
<td>This service periodically checks the expiration status of certificates that Cisco Unified CallManager generates and sends notification when a certificate gets close to its expiration date.</td>
</tr>
<tr>
<td>Cisco Database Layer Monitor</td>
<td></td>
<td></td>
<td>dbmon</td>
<td>3</td>
<td>The Cisco Database Layer Monitor service monitors aspects of the database layer. This server takes responsibility for change notification and monitoring.</td>
</tr>
<tr>
<td>Cisco Log Partition Monitoring Tool</td>
<td></td>
<td></td>
<td>Lpm Tool</td>
<td>3</td>
<td>The Cisco Log Partition Monitoring Tool service supports the Log Partition Monitoring feature, which monitors the disk usage of the log partition on a server (or all servers in the cluster) by using configured thresholds and a polling interval.</td>
</tr>
</tbody>
</table>
### Critical Processes to Monitor

<table>
<thead>
<tr>
<th>Service</th>
<th>Process Name</th>
<th>Auto Restart</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco CDP</td>
<td>cdpd</td>
<td>6</td>
<td>Cisco CDP advertises Cisco Unified CallManager to other applications, so the application, for example, SNMP or CiscoWorks2000, can perform network management tasks for Cisco Unified CallManager.</td>
</tr>
<tr>
<td>Cisco RIS Data Collector</td>
<td>RisDC</td>
<td>3</td>
<td>The Real-time Information Server (RIS) maintains real-time Cisco Unified CallManager information such as device registration status, performance counter statistics, critical alarms generated, and so on. The Cisco RIS Data Collector service provides an interface for applications, such as Real-Time Monitoring Tool (RTMT), SOAP applications, Cisco Unified CallManager Administration and AlertMgrCollector (AMC) to retrieve the information that is stored in all RIS nodes in the cluster.</td>
</tr>
<tr>
<td>Cisco DRF Master</td>
<td>CiscoDRFMaster</td>
<td>3</td>
<td>The Cisco DRF Master Agent service supports the DRF Master Agent, which works with the graphical user interface (GUI) or command line interface (CLI) to schedule backups, perform restorations, view dependencies, check status of jobs, and cancel jobs, if necessary. The Cisco DRF Master Agent also provides the storage medium for the backup and restoration process.</td>
</tr>
<tr>
<td>Cisco DRF Local</td>
<td>CiscoDRFLocal</td>
<td>3</td>
<td>The Cisco DRF Local service supports the Cisco DRF Local Agent, which acts as the workhorse for the DRF Master Agent. Components on a node register with the Cisco DRF Local Agent to use the disaster recovery framework. The Cisco DRF Local Agent executes commands that it receives from the Cisco DRF Master Agent. Cisco DRF Local Agent sends the status, logs, and command results to the Cisco DRF Master Agent.</td>
</tr>
<tr>
<td>Cisco CDR Repository Manager</td>
<td>cdrrep</td>
<td>3</td>
<td>You can start and stop the Cisco CDR Repository Manager service only on the first node, which contains the Cisco Unified CallManager database. This service starts automatically.</td>
</tr>
</tbody>
</table>
Table 3-1 Critical Services to Monitor (continued)

<table>
<thead>
<tr>
<th>Service</th>
<th>Stop</th>
<th>Start</th>
<th>Process Name</th>
<th>Auto Restart</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco CDR Agent</td>
<td></td>
<td></td>
<td>cdragent</td>
<td>3</td>
<td>The Cisco CDR Agent service transfers CDR and CMR files that are generated by Cisco Unified CallManager from the local host to the CDR repository node, where the CDR Repository Manager service runs over a SFTP connection. For this service to work, activate the Cisco CallManager service on the first node and ensure that it is running.</td>
</tr>
<tr>
<td>SSH Service Status</td>
<td>CLI</td>
<td>util</td>
<td>ssdh</td>
<td>3</td>
<td>—</td>
</tr>
<tr>
<td>Syslog Service Status</td>
<td>Auto-</td>
<td>restart</td>
<td>syslogd</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>SNMP Service Status</td>
<td>CLI</td>
<td>snmp</td>
<td>hardware-agent restart **</td>
<td>—</td>
<td>IBM—snmpd, slp_srvreg cimlistener, cimserver, dirsnmpd, &quot;&quot;java...com.tivoli.twg.agent.TWGAgent&quot;&quot; **** HP</td>
</tr>
<tr>
<td>DRF Restoral Condition</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No API to monitor status of DRF Restoral Condition.</td>
</tr>
<tr>
<td>IBM Director Agent SNMP</td>
<td>IBM process covered by SNMP Service</td>
<td>cim listenerd</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>IBM Director Agent SNMP</td>
<td>IBM process covered by SNMP Service</td>
<td>cim serverd</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>dirsnpmd</td>
<td>IBM process covered by SNMP Service</td>
<td>dir snmpd</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Cmaeventd</td>
<td>HP process covered by SNMP Service</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Cmafcad</td>
<td>HP process covered by SNMP Service</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Cmahealthd</td>
<td>HP process covered by SNMP Service</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Cmahostd</td>
<td>HP process covered by SNMP Service</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Cmaidad</td>
<td>HP process covered by SNMP Service</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>
### Critical Processes to Monitor

<table>
<thead>
<tr>
<th>Service</th>
<th>Stop</th>
<th>Start</th>
<th>Process Name</th>
<th>Auto Restart Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cmaided</td>
<td></td>
<td></td>
<td>HP process covered by</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SNMP Service</td>
<td></td>
</tr>
<tr>
<td>Cmanicd</td>
<td></td>
<td></td>
<td>HP process covered by</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SNMP Service</td>
<td></td>
</tr>
<tr>
<td>Cmaperred</td>
<td></td>
<td></td>
<td>HP process covered by</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SNMP Service</td>
<td></td>
</tr>
<tr>
<td>Cmaperfd</td>
<td></td>
<td></td>
<td>HP process covered by</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SNMP Service</td>
<td></td>
</tr>
<tr>
<td>Cmasm2d</td>
<td></td>
<td></td>
<td>HP process covered by</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SNMP Service</td>
<td></td>
</tr>
<tr>
<td>Cmastdeqd</td>
<td></td>
<td></td>
<td>HP process covered by</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SNMP Service</td>
<td></td>
</tr>
<tr>
<td>Cmathreshd</td>
<td></td>
<td></td>
<td>HP process covered by</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SNMP Service</td>
<td></td>
</tr>
<tr>
<td>hpsm</td>
<td></td>
<td></td>
<td>HP process covered by</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SNMP Service</td>
<td></td>
</tr>
<tr>
<td>hpsmxld</td>
<td></td>
<td></td>
<td>hpsmxld</td>
<td></td>
</tr>
<tr>
<td>snmpsa-ah</td>
<td></td>
<td></td>
<td>INTEL process covered</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>by SNMP Service</td>
<td></td>
</tr>
<tr>
<td>Cisco Security</td>
<td></td>
<td></td>
<td>Auto-restart being</td>
<td></td>
</tr>
<tr>
<td>Agent Service</td>
<td></td>
<td></td>
<td>addressed by Cisco.</td>
<td></td>
</tr>
<tr>
<td>Status</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ciscosec</td>
<td></td>
<td></td>
<td>Indefinite</td>
<td></td>
</tr>
<tr>
<td>Cisco Electronic</td>
<td>enStart</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Notification</td>
<td></td>
<td></td>
<td>Serviceability/Tools</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&gt; Control Center -</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Network Services</td>
<td></td>
</tr>
<tr>
<td>Time Synchronization Service</td>
<td>ntpd</td>
<td></td>
<td>Auto-restarts</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>according to ‘init’</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>rules (10 if</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>instantaneous failure,</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>otherwise higher).”</td>
<td></td>
</tr>
<tr>
<td>Service Manager</td>
<td>CLI</td>
<td>servM</td>
<td>Auto-restarts</td>
<td></td>
</tr>
<tr>
<td></td>
<td>utils</td>
<td></td>
<td>according to ‘init’</td>
<td></td>
</tr>
<tr>
<td></td>
<td>restart</td>
<td></td>
<td>rules (10 if</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Service Manager</td>
<td></td>
<td>instantaneous failure,</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>otherwise higher).</td>
<td></td>
</tr>
</tbody>
</table>
Table 3-1 Critical Services to Monitor (continued)

<table>
<thead>
<tr>
<th>Service</th>
<th>Stop</th>
<th>Start</th>
<th>Process Name</th>
<th>Auto Restart</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Racoon DB</td>
<td>N/A</td>
<td></td>
<td>racoon</td>
<td>—</td>
<td>Internet Key Exchange (IKE) daemon for automatically keying IPsec connections. Auto-restarts according to ‘init’ rules (10 if instantaneous failure, otherwise higher).</td>
</tr>
<tr>
<td>IP Sec Manager</td>
<td>—</td>
<td></td>
<td>ipsec_mgr</td>
<td>—</td>
<td>Auto-restarts according to ‘init’ rules (10 if instantaneous failure, otherwise higher).</td>
</tr>
</tbody>
</table>

SysLog Test Cases For Cisco Unified CM

| MGCPGateway LostComm | Natively supported alarm—GUI Serviceability/Alarm/Catalog, CallManager, MGCPGatewayLostComm/Find |
| SDLLinkOOS          | Natively supported alarm—GUI Serviceability/Alarm/Catalog, CallManager, SDLLinkOOS/Find” |

SNMP Trap Test Cases

<table>
<thead>
<tr>
<th>Service</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ccmGatewayFailed Event</td>
<td>CCM-MIB::ccmGatewayFailed</td>
</tr>
<tr>
<td>IBMPSG PowerSupplyEvent</td>
<td>IBM-SYSTEM-POWER-MIB; pull cord on IBM MCS-7835 &amp; MCS-7845 servers with redundant power supply to invoke.</td>
</tr>
</tbody>
</table>

Available Supported MIBs

The following MIBs can be reviewed and used for monitoring system health:

- Cisco MIBs (Chapter 7, “Cisco Management Information Base”)
  - CISCO-CCM-MIB, page 7-1
  - CISCO-CCM-CAPABILITY, page 7-121
  - CISCO-CDP-MIB, page 7-127
  - CISCO-SYSLOG-MIB, page 7-144
  - CISCO-SYSLOG-EXT-MIB, page 7-152
- Industry-Standard MIBs (Chapter 8, “Industry-Standard Management Information Base”)
  - SYSAPPL-MIB, page 8-1
The following topics are described in this section:

- RTMT Summary View, page 3-12
- CPU Usage, page 3-13
- % IOwait Monitoring, page 3-15
- Virtual Memory, page 3-16
- Disk Usage, page 3-17
- Database Replication and Cisco Unified Communication Manager Nodes, page 3-20
- Cisco Unified CM Process and CPU Usage, page 3-20
- CodeYellow, page 3-21
- RIS Data Collector PerfMonLog, page 3-23
- Critical Service Status, page 3-24
- Syslog Messages, page 3-25
- RTMT Alerts as Syslog Messages and Traps, page 3-26

RTMT Summary View

The RTMT summary view displays the overall health of the system, which should be monitored daily, including:

- CPU utilization level
- Memory utilization level
- Phone registration status
- Call in progress
- Gateway status

If CPU and memory utilization levels exceed the 70 percent mark, then the Cisco Unified CM publisher and subscribers that are participating in call processing could be overloaded. Key indicators of system health and performance issues are:

- System Time, User Time, IOWait, soft irq, irq
- CPU Pegging Alerts
- Process using most CPU
- High % iowait
- High % iowait due to common partition
- Process responsible for Disk IO
- CodeYellow
If you do not want the RTMT client running on your workstation or PC all the time, you can configure a threshold for each alert that is of interest to you and how you want to be notified. Then you can close the RTMT client on your workstation or PC.

The RTMT backend, AMC service, which is up and running as soon as the Cisco Unified CM server is up and running, collects and processes all the information needed, and notifies you according to how you configured the notification.

RTMT CPU and memory page reports CPU usage in terms of the following:

- %System—CPU utilization percentage that occurred while executing at the system level (kernel)
- %User—CPU utilization percentage that occurred while executing at the user level (application).
- %IOWait—CPU percentage of time of idle waiting for outstanding disk I/O request.
- %SoftIRQ—Percentage of time the processor is executing deferred IRQ processing (for example, processing of network packets).
- %IRQ—Percentage of time that the processor is executing the interrupt request which is assigned to devices for interrupt or sending a signal to the computer when it is finished processing.

### CPU Usage

High CPU utilization can impact the call processing by creating delays or interruptions in the service. It could affect the end user service. Sometimes high CPU utilization is indicative of a memory leak. RIS DataCollector PerfMonLog when enabled tracks CPU usage.

![Note](image)

Cisco recommends that RIS DataCollector PerfMonLog be enabled.

Table 3-2 shows CPU usage guidelines.

<table>
<thead>
<tr>
<th>Usage</th>
<th>MCS-7835</th>
<th>MCS-7845</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total CPU usage—Processor (_Total)</td>
<td>% CPU Time</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt; 68% is good</td>
<td>&lt; 68% is good</td>
</tr>
<tr>
<td></td>
<td>68–70% triggers a warning</td>
<td>68–70% triggers a warning</td>
</tr>
<tr>
<td></td>
<td>&gt; 80% is bad</td>
<td>&gt; 80% is bad</td>
</tr>
<tr>
<td>Process ccm CPU</td>
<td>&lt; 44%</td>
<td>&lt; 22%</td>
</tr>
<tr>
<td>IOWAIT—Processor (_Total) \IOwait Percentage</td>
<td>&lt; 10% is good</td>
<td>&lt; 10% is good</td>
</tr>
<tr>
<td>CallManager Service Virtual Memory size</td>
<td>&lt; 2.1 GB</td>
<td>&lt; 2.1 GB</td>
</tr>
</tbody>
</table>

You can also monitor CPU usage by using APIs. Using the SOAP API, you can monitor the following perfmon counters:

- Under Processor object—% CPU Time, System Percentage, User Percentage, IOWait Percentage, Softirq Percentage, IRQ Percentage
- Under Process object—% CPU Time
Using the SNMP interface, you can monitor the following perfmon counters:

- Host Resource MIB—hrProcessorLoad, hrSWRunPerfCPU
- CPQHOST-MIB—cpqHoCpuUtilMin, cpqHoCpuUtilFiveMin

If you see high CPU usage, identify which process is causing it. If %system and/or %user is high enough to generate CPUPegging alert, check the alert message to see the processes that are using the most CPU. You can go to the RTMT Process page, sort by %CPU to identify high CPU processes.

Figure 3-2 shows the CPU usage.

Figure 3-2 Cisco Unified Serviceability CPU Usage

For analysis, RIS Data Collector PerfMonLog tracks processes %CPU usage at system level. RTMT monitors CPU usage and when CPU usage is above a threshold, RTMT generates CallProcessingNodeCPUPegging alert. Figure 3-3 shows the alert status.

Figure 3-3 RTMT Alert Central with Alert Status

Monitor the “In Safe Range” column often. If it is marked “No,” then the condition is not corrected. For example, if In Safe Range column displays No for CallProcessingNodeCPUPegging, then it means the CPU usage on that node is above the threshold and requires attention.

In addition to CallProcessingNodeCPUPegging, high CPU usage potentially causes the following alerts to trigger:

- CodeYellow
- CodeRed
- CoreDumpFileFound
- CriticalServiceDown
- LowCallManagerHeartbeatRate
- LowTFTPServerHeartbeatRate
• LowAttendantConsoleHeartRate
When a service crashes, the corresponding trace files may have been overwritten. Cisco TAC needs the trace files to troubleshoot the crash. In the case of CoreDumpFileFound, CodeYellow, and CriticalServiceDown, the Enable Trace Download option should be enabled to assist Cisco TAC.

% IOwait Monitoring

High %IOwait indicates high disk input/output (I/O) activities. Consider the following high IOwait conditions:

• Heavy memory swapping—Check %CPU Time for Swap Partition to see if there is high level of memory swapping activity. One potential cause of high memory swapping is memory leak.
• DB activity—Database accesses Active Partition. If %CPU Time for Active Partition is high, then most likely there are a lot of DB activities.
• Common (or Log) Partition in the trace and log files storage location—Check the following:
  – Trace Log Center to see if there is any trace collection activity going on. If call processing is impacted (ie, CodeYellow), then consider adjusting trace collection schedule. If zip option is used, please turning it off.
  – Trace setting at the detailed level because Cisco Unified CM generates a lot of trace. If high %iowait and/or Cisco Unified CM is in CodeYellow state, and Cisco Unified CM service trace setting is at Detailed, please chance trace setting to “Error” to reduce the trace writing.

You can use RTMT to identify processes that are responsible for high %IOwait:

• If %IOwait is high enough to cause CPU Pegging alert, check the alert message to check processes waiting for disk IO.
• Go to RTMT Process page, sort by Status. Check for processes in Uninterruptible Disk Sleep state
• Download RIS Data Collector PerfMonLog file to examine the process status for longer period of time.

Figure 3-4 shows an example of RTMT Process window sorted by Status. Check for processes in Uninterruptible Disk Sleep state. The FTP process is in the Uninterruptible Disk Sleep state.

Figure 3-4  FTP Process in Uninterruptible Disk Sleep State
Virtual Memory

Virtual memory consists of physical memory (RAM) and swap memory (Disk). The RTMT CPU and Memory window has system level memory usage information as the following:

- **Total**—Total amount of physical memory
- **Free**—Amount of free memory
- **Shared**—Amount of shared memory used
- **Buffers**—Amount of memory used for buffering purpose
- **Cached**—Amount of cached memory
- **Used**—Calculated as Total – Free – Buffers – Cached + Shared
- **Total Swap**—Total amount of swap space
- **Used Swap**—Amount of swap space in use on the system.
- **Free Swap**—Amount of free swap space available on the system

**Note**
Using SOAP APIs, you can query memory information for the following perfmon counters:

- Under Memory object—% Mem Used, % VM Used, Total Kbytes, Total Swap Kbytes, Total VM Kbytes, Used Kbytes, Used Swap Kbytes, Used VM Kbytes
- Under Process object—VmSize, VmData, VmRSS, % Memory Usage

Using SNMP, you can query the following perfmon counters:

- Host Resource MIB—hrStorageSize, hrStorageUsed, hrStorageAllocationUnits, hrStorageDescr, hrStorageType, hrMemorySize

**Note**
You can download some historical information by using RTMT Trace Log Central. The Cisco AMC Service PerfMonLog is enabled by default. The Cisco AMC Service PerfMonLog is deprecated in Cisco Unified CM Release 6.0 because Cisco RIS Data Collector PerfMonLog was introduced. The Cisco RIS Data Collector PerfMonLog disabled by default in Cisco Unified CM Release 5.x and enabled by default in Cisco Unified CM Release 6.0.

**Note**
Perfmon Virtual Memory refers to Total (Physical + Swap) memory whereas Host Resource MIB Virtual Memory refers to Swap memory only.

The RTMT Process window displays process level memory usage information as follows:

- **VmSize**—Total virtual memory used by the process
- **VmRSS**—Resident Set currently in physical memory used by the process including Code, Data and Stack
- **VmData**—Virtual memory usage of heap by the process
- **Page Fault Count**—Represents the number of major page faults that a process encountered that required the data to be loaded into physical memory

**Figure 3-5** shows RTMT Process window. You can sort VmSize by clicking on VmSize tab. Then you can identify which process consumes more memory.
Possible memory leak causes can be from the VmSize continuously increasing.

When a process leaks memory, the system administrator should report it to Cisco and include trace files. RIS Data Collector PerfMonLog collects the data and it contains historical information on memory usage.

### Disk Usage

There are four disks or partitions in the Cisco Unified CM hard drive:

- **Common partition (log partition)**—Contains the trace/log files
- **Active partition**—Contains files (binaries, libraries and config files) of active OS and the Cisco Unified CM release
- **Inactive partition**—Contains files for alternative Cisco Unified CM release (for example, an older version that was upgraded from or newer version recently upgraded to but the server has not been toggled to this release).
- **Swap partition**—Used for swap space

Using SOAP APIs, you can get partition information for the following perfmon counters:

- Under Partition object—Total Mbytes, Used Mbytes, Queue Length, Write Bytes Per Sec, Read Bytes Per Sec

Using the SNMP MIB, you can query the following information:

- Host Resource MIB—hrStorageSize, hrStorageUsed hrStorageAllocationUnits, hrStorageDescr, hrStorageType

You can download the following historical information by using RTMT Trace and Log Central:

- Cisco AMC Service PerfMonLog // enabled by default. Deprecated in Cisco Unified CM 6.0, because Cisco RIS Data Collector PerfMonLog is introduced
- Cisco RIS Data Collector PerfMonLog // disabled by default in Cisco Unified CM 5.x; enabled by default in Cisco Unified CM 6.0

Figure 3-6 shows disk usage by partition in RTMT.
Disk Name Mapping

Perfmon instance names as shown in RTMT and SOAP are:

- Active
- Inactive
- Common
- Boot
- Swap
- SharedMemory

Names shown in Host Resource MIB hrStorage description are:

- /partB
- /common
- /grub
- Virtual Memory
- /dev/shm

The partition alerts are as follows:

- LogPartitionLowWaterMarkExceeded— Occurs when the percentage of used disk space in the log partition has exceeded the configured low water mark. This alert should be considered as early warning for an administrator to clean up disk space. You can use RMT Trace/Log Central to collect trace/log files and then delete these trace/log files from the server. In addition to manually clean up the traces/log files, the system administrator should also adjust the number of trace files to be kept to avoid hitting low water mark again.

Figure 3-6  Disk Usage by Partition
• LogPartitionHighWaterMarkExceeded—Occurs when the percentage of used disk space in the log partition has exceeded the configured high water mark. When this alert is generated, Log Partition Monitoring (LPM) utility starts to delete files in Log Partition until the Log Partition is down to the low water mark to avoid running out of disk space. Since LPM may delete some files that you want to keep, you need to act upon receiving LogPartitionLowWaterMarkExceed alert.

• LowActivePartitionAvailableDiskSpace—Occurs when the percentage of available disk space of the Active Partition is lower than the configured value. Please use the default threshold that Cisco recommends. At default threshold, this alert should never be generated. If this alert occurs, a system administrator can adjust the threshold as temporary workaround but Cisco TAC should look into this. One place to look is /tmp using remote access. We have seen cases where large files are left there by 3rd party software.

• LowInactivePartitionAvailableDiskSpace—Occurs when the percentage of available disk space of the InActive Partition is lower than the configured value. Please use the default threshold that Cisco recommends. At default threshold, this alert should never be generated. If this alert occurs, a system administrator can adjust the threshold as temporary workaround but Cisco TAC should look into this.

Table 3-3 shows a comparison of disk-related perfmon counters between Cisco Unified CM Release 4.x and Cisco Unified CM Release 5.x.

<table>
<thead>
<tr>
<th>Logical Disk</th>
<th>% Disk Time</th>
<th>Partition</th>
<th>% CPU Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk Read Bytes/sec</td>
<td>Read Kbytes Per Sec</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Disk Write Bytes/sec</td>
<td>Write Kbytes Per Sec</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Current Disk Queue Length</td>
<td>Queue Length</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Free Megabytes</td>
<td>Used Mbytes</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Total Mbytes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>% Free Space</td>
<td>% Used</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Database Replication and Cisco Unified Communication Manager Nodes

You can use RTMT Database Summary to monitor your database activities as shown in Figure 3-7. For example, click CallManager > Service > Database Summary.

Figure 3-7 Database Summary in RTMT

![Database Summary in RTMT](image)

Cisco Unified CM Process and CPU Usage

The Cisco Unified CM process is labeled “ccm.” Table 3-4 contains general guidelines for the Cisco Unified CM Process and CPU usage.

Table 3-4 Cisco Unified CM Process and CPU Usage

<table>
<thead>
<tr>
<th>CPU usage Process (ccm)% CPU Time</th>
<th>MCS-7835 Server</th>
<th>MCS-7845 Server</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 44% is good</td>
<td>&lt; 22% is good</td>
<td></td>
</tr>
<tr>
<td>44-52% triggers a warning</td>
<td>22-36% triggers a warning</td>
<td></td>
</tr>
<tr>
<td>&gt; 60% is bad</td>
<td>&gt; 30% is bad</td>
<td></td>
</tr>
</tbody>
</table>
The MCS-7845 server has more processors and a lower threshold for CPU usage because the ccm process is a multithreaded application. But the main router thread does the bulk of call processing. A single thread can run only on one processor at any given time even when there are multiple processors available. That means ccm main router thread can run out of CPU resource even when there are idle processors.

With hyper-threading, the MCS-7845 server has 4 virtual processors. So on the server where the main router thread is running at full blast to do call processing, it is possible three other processors are near idle. In this situation UC Manager can get into Code Yellow state even when total CPU usage is 25 to 30 percent. (Similarly MCS-7835 server with two virtual processors, UC Manager could get into Code Yellow state at around 50 to 60 percent of CPU usage.

Use the following to query perfmon counters:

- **SOAP APIs:**
  - Perfmon counters
  - Device information
  - DB access
  - CDR access

- **SNMP:**
  - CISCO-CCM-MIB—ccmPhoneTable, ccmGatewayTable, etc.
  - Download historical information by using RTMT Trace/Log Central.
  - Cisco AMC Service PerfMonLog is enabled by default. This was deprecated in Cisco Unified CM Release 6.0 because Cisco RIS Data Collector PerfMonLog was introduced.
  - Cisco RIS Data Collector PerfMonLog was disabled by default in Cisco Unified CM Release 5.x and enabled by default in Cisco Unified CM Release 6.0.

### CodeYellow

CodeYellow state occurs when the ccm process is so overloaded that it cannot process incoming calls anymore. In this case, Cisco Unified CM initiates call throttling. This does not mean that one processor CPU usage is at 100 percent and the remaining processors are operating at 0 percent in RTMT.

Since the main thread can run on processor A for 1/10th of a second and processor B on the next 2/10th of a second, etc., the CPU usage shown in RTMT would be more balanced. By default RTMT shows average CPU usage for a 30-second duration.

You can configure the CodeYellow alert so that once it occurs, the trace files can be downloaded for troubleshooting purposes.

The AverageExpectedDelay counter represents the current average expected delay for handling any incoming message. If the value is above the value specified in “Code Yellow Entry Latency” service parameter, CodeYellow alarm is generated. This counter is one of key indicator of call processing performance issue.

If you see CodeYellow, but the total CPU usage is only 25 percent, it is because Cisco Unified CM needs one processor for call processing. When no processor resource is available, CodeYellow may occur even when the total CPU usage is only around 25 to 30 percent in a 4-virtual processor server. Similarly on a 2 processor server, CodeYellow is possible around 50 percent of total CPU usage.
Other perfmon counters should be monitored are:

- Cisco CallManager\CallsActive, CallsAttempted, EncryptedCallsActive, AuthenticatedCallsActive, VideoCallsActive
- Cisco CallManager\RegisteredHardwarePhones, RegisteredMGCPGateway
- Cisco CallManager\T1ChannelsActive, FXOPortsActive, MTPResourceActive, MOHMulticastResourceActive
- Cisco Locations\BandwidthAvailable
- Cisco CallManager System Performance\AverageExpectedDelay
- CodeYellow
- DBReplicationFailure
- LowCallManagerHeartbeat
- ExcessiveVoiceQualityReports
- MaliciousCallTrace
- CDRFileDeliveryFailure/CDRAgentSendFileFailed
- Critical Service Down
- CoreDumpFileFound

Figure 3-8 displays the RTMT performance window.

Figure 3-8  RTMT Performance of Stand Alone Clusters

Note

In general, Cisco Unified CM Release 4.x perfmon counters have been preserved by using the same names and representing the same values.
RIS Data Collector PerfMonLog

In Cisco Unified CM Release 5.x, the RIS Data Collector PerfMonLog file is not enabled by default. It is recommended that RIS Data Collector PerfMonLog is enabled to assist in troubleshooting. It tracks CPU, memory, disk, and the network. If you enable RIS Data Collector PerfMonLog, then you can disable AMC PerfMonLog. In Cisco Unified CM Release 6.x, RIS Data Collector PerfMonLog replaced AMC PerfMonLog.

Note

With RIS Data Collector PerfMonLog enabled, the impact on the CPU is small, around 1%.

Use RTMT Trace and Log Center to download Cisco RIS Data Collector PerfMonLog files for the time period that you are interested in. Open the log file using Windows Perfmon Viewer (or RTMT Perfmon viewer), then add Performance counters of interest such as:

- CPU usage > Processor or Process % CPU
- Memory usage > Memory %VM Used
- Disk usage > Partition % Used
- Call Processing > Cisco CallManager CallsActive

Figure 3-9 shows the output of the Windows Perfmon Viewer.

Figure 3-9  Windows Perfmon Viewer
Critical Service Status

The RTMT Critical Service window provides current status of all critical services as shown in Figure 3-10.

**Figure 3-10  Critical Service Window in RTMT**

CriticalServiceDown alert is generated when any of service is down. By default, RTMT back-end service checks for the status every 30 seconds. It is possible if the service goes down and comes back up within that period, the CriticalServiceDown alert may not be generated.

CriticalServiceDown alert monitors only those services listed in RTMT Critical Services page. If you suspect if service got restarted without generating Core files, check the RTMT Critical Service page has elapsed time and Check RIS Troubleshooting perfmon log files and see if PID for service (process) is changed.

The following CLI can be used to check the logs of Service Manager:
- `file get activelog platform/servm_startup.log`
- `file get activelog platform/log/servm*.log`

The following CLI can be used to duplicate certain RTMT functions:
- `admin:utils service`
- `show perf`
- `show risdb`
CoreDumpFileFound alert is generated when RTMT backend service detects new Core Dump file. Both CriticalServiceDown and CoreDumpFileFound alert can be configured to download corresponding trace files for troubleshooting purpose. This helps to preserve trace files at the time of a crash.

**Syslog Messages**

Syslog messages can be viewed using RTMT syslog viewer as shown in Figure 3-11.

*Figure 3-11  Syslog Viewer*

To send syslog traps to a remote server for the CISCO-SYSLOG-MIB follow these steps:

**Step 1**  Setup Trap (Notification) destination in Cisco Unified Serviceability SNMP window.

**Step 2**  Enable trap generation in CISCO-SYSLOG-MIB.

**Step 3**  Set the appropriate SysLog level in CISCO-SYSLOG-MIB.

If syslog traps are not being generated for some Cisco Unified CM service alarms, check the RTMT syslog viewer to see if the alarms are shown there. If not, adjust alarm configuration setting to send alarms to local syslog.

Syslogs generated due to hardware failures have an event severity of 4 or higher and contain one of the following patterns:

- *cma*[????]:*
- *cma*[????]:*
• *cma*[^?????]:*
• *hp*[^????]:*
• *hp*[^????]:*
• *hp*[^????]:*

You can search for the above patterns to find hardware failure events in syslog.

For information on alarm configuration, refer to the Alarm Configuration section of the Cisco Unified Serviceability Administration Guide at


**RTMT Alerts as Syslog Messages and Traps**

RTMT alerts can be sent to a remote syslog server. To send to a local and remote syslog server, configure the AMC alarm in Cisco Unified Serviceability. Figure 3-12 shows the window.

**Figure 3-12  Local and Remote Syslog Configuration**

Recovery, Hardware Migration, and Backup/Restore

The following topics are described in this section:

- Backup/Restore, page 3-26
- Platform Monitoring, page 3-27

**Backup/Restore**

Cisco provides the following backup/restore utilities:

- Cisco Unified CM Release 4.x uses the Backup and Restore System (BARS) application
- Cisco Unified CM Release 5.x uses the Disaster Recovery Framework (DRF)
- Cisco Unified CM Release 6.x uses the Disaster Recovery System (DRS), essentially a renaming of DRF above
These tools support writing backup files to (or reading restore files from) a local tape drive, or a file on a network location. BARS uses Windows shares and DRF/DRS use SFTP to access the network location. If a third-party backup solution is desired, BARS/DRF/DRS can write to a network location for the third-party backup solution to pick up.

DRF/DRS perform a cluster-wide backup, meaning data from all nodes is backed up, but restores are only to the node(s) that need it.

For more details, including what is configured to be included in the backup or what files are created, refer to the following documents depending on release:

- Disaster Recovery Administration Guide
- Cisco IP Telephony Disaster Recovery Administration Guide
- Cisco IP Telephony Backup and Restore System (BARS) Administration Guide

It is recommended to take a fresh backup every time an install, upgrade or options install is done to the appliance, whether or not configuration data changes were made.

If a catastrophic hardware failure occurs and the hardware must be replaced, reinstall Cisco Unified CM on the new hardware, then perform a restore from your backup.

---

**Note**

Drive pull/swap is not supported as a fast recovery solution for the appliance.


---

**Platform Monitoring**

This section describes hardware-layer monitoring for system component temperature, fan status, power supply status, RAID and disk status, network status, and operational status. CPU status/utilization and Memory status/utilization are covered in another section. It contains the following subsections:

- **Using SNMP MIBs**, page 3-27
- **Using Command Line Interface**, page 3-28
- **Hardware Migration**, page 3-31
- **Platform Security**, page 3-31

---

**Using SNMP MIBs**

Cisco Unified CM hardware servers are monitored by using SNMP MIBs. The following MIBs are supported:

- **Vendor-Specific MIBs** (Chapter 9, “Vendor-Specific Management Information Base”)
  - IBM-SYSTEM-LMSENSOR
  - IBM-SYSTEM-POWER
  - IBM-SYSTEM-RAID
  - IBM-SYSTEM-xxx-MIB
Chapter 3      Managing and Monitoring the Health of Cisco Unified Communications Manager Systems

Platform Monitoring

- CPQ-xxx-MIB (HP)
- CPQHEALTH (HP)
- INTEL-SERVER-BASEBOARD6 (Introduced in Cisco Unified CM Release 7.1[2])

You configure SNMP in the network management applications to receive SNMP traps, notifications, and informs listed in the MIBs. Specific MIB support varies by Cisco Unified CM release and hardware vendor.

MIBs and MCS Types

There are no specific OIDs available to directly give the MCS type. In the case of Linux appliances, the value of sysObjectID can be mapped to the server types. For instance sysobjectID returns 1.3.6.1.4.1.9.1.583 for a HP-7825 server.

In the case of Windows, there are no such specific values returned for server types except for OID does identify the server as a Windows server. Refer to http://www.oidview.com/mibs/9/CISCO-PRODUCTS-MIB.html for list of sysObjectIDs assigned to different hardware.

For Media Convergence Server (MCS) MIBs supported by Cisco Unified CM releases, go to this URL—http://www.cisco.com/en/US/docs/voice_ip_comm/cucm/compat/cmmibcmp.xls.

Using Command Line Interface

System BIOS is viewable during the server boot sequence. The following commands are useful to view details about hardware, BIOS, RAID, and firmware. These items are included as part of the Cisco Unified CM image and do not need to be managed separately as in Cisco Unified CM Release 4.x, but may need to be inspected during diagnostic activity.

```
show hardware
show environment [fans | power-supply | temperature]
show tech all
utils create report hardware
```

You can also use the admin:utils fior status CLI to isolate which process causes high IOwait. Other available options to use with the admin:utils fior command are—enable, disable, start, stop, list, top. For example, at the command prompt type admin:utils fior list. This displays:

```
2007-05-31 Counters Reset
Time  Process  PID  State  Bytes Read  Bytes Written
17:02:45  rpmq  31206  Done  14173728  0
17:04:51  java  31147  Done  310724  3582
17:04:56  snmpget  31365  Done  989543  0
17:10:22  top  12516  Done  7983360  0
17:21:17  java  31485  Done  313202  2209
17:44:34  java  1194  Done  92483  0
17:44:51  java  1231  Done  192291  0
17:45:09  cdpd  6145  Done  0  2430100
17:45:25  java  1319  Done  192291  0
17:45:31  java  1330  Done  192291  0
```
Use admin:utils for top CLI for output sorted by top disk users. This displays:

Sort by Bytes Written

<table>
<thead>
<tr>
<th>Process</th>
<th>PID</th>
<th>Bytes Read</th>
<th>Read Rate</th>
<th>Bytes Written</th>
<th>Write Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linuxzip</td>
<td>19556</td>
<td>61019083</td>
<td>15254771</td>
<td>12325229</td>
<td>3081307</td>
</tr>
<tr>
<td>Linuxzip</td>
<td>19553</td>
<td>58341109</td>
<td>11668622</td>
<td>9860680</td>
<td>1972136</td>
</tr>
<tr>
<td>Linuxzip</td>
<td>19544</td>
<td>55679597</td>
<td>11135919</td>
<td>7390382</td>
<td>1478076</td>
</tr>
<tr>
<td>installld</td>
<td>28786</td>
<td>3764719</td>
<td>83660</td>
<td>6847693</td>
<td>152171</td>
</tr>
<tr>
<td>Linuxzip</td>
<td>20150</td>
<td>18963498</td>
<td>6321166</td>
<td>6672927</td>
<td>2224309</td>
</tr>
<tr>
<td>Linuxzip</td>
<td>20148</td>
<td>53597311</td>
<td>17865770</td>
<td>5943560</td>
<td>1981187</td>
</tr>
<tr>
<td>Linuxzip</td>
<td>19968</td>
<td>9643296</td>
<td>4821648</td>
<td>5438963</td>
<td>2719482</td>
</tr>
<tr>
<td>Linuxzip</td>
<td>19965</td>
<td>53107868</td>
<td>10621574</td>
<td>5222659</td>
<td>1044532</td>
</tr>
<tr>
<td>Linuxzip</td>
<td>19542</td>
<td>53014605</td>
<td>13253651</td>
<td>4922147</td>
<td>1230537</td>
</tr>
<tr>
<td>mv</td>
<td>5048</td>
<td>3458525</td>
<td>3458525</td>
<td>3454941</td>
<td>3454941</td>
</tr>
</tbody>
</table>

Other commands that are available are as follows:
- admin:utils diagnose list
- admin:utils diagnose test
- admin:utils diagnose module <moduleName>
- admin:utils diagnose fix
- admin:utils create report hardware
- admin:utils iostat

**admin:utils diagnose list CLI**
Displays all available diagnostic tests as follows:

Available diagnostics modules
- disk_space
  - Check available disk space as well as any unusual disk usage
- service_manager
  - Check if service manager is running
- tomcat
  - Check if Tomcat is deadlocked or not running

**admin:utils diagnose test CLI**
Executes each diagnostic test. It will not attempt to repair anything. This displays:

Starting diagnostic test(s)
------------------------------
<table>
<thead>
<tr>
<th>Test</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>test - disk_space</td>
<td>Passed</td>
</tr>
<tr>
<td>test - service_manager</td>
<td>Passed</td>
</tr>
</tbody>
</table>
Chapter 3      Managing and Monitoring the Health of Cisco Unified Communications Manager Systems

Platform Monitoring

test - tomcat              —Passed
Diagnostics Completed

admin:utils diagnose module <moduleName> CLI
Executes a single diagnostic test and attempt to fix the problem. You can also use admin:utils diagnose fix CLI to run all of the diagnostic tests at once. For example, admin:utils diagnose module tomcat displays:

Starting diagnostic test(s)
===========================
test - tomcat              —Passed
Diagnostics Completed

admin:utils diagnose fix CLI
Execute all diagnostic tests, and if possible, attempt to repair the system. This displays:

Starting diagnostic test(s)
===========================
test - disk_space          —Passed
test - service_manager     —Passed
test - tomcat              —Passed
Diagnostics Completed

admin:utils create report hardware CLI
Creates a system report containing disk array, remote console, diagnostic, and environmental data. No parameters are required. This displays:

***   W A R N I N G   ***
This process can take several minutes as the disk array, remote console, system diagnostics and environmental systems are probed for their current values.
Continue? Press y or Y to continue, any other key to cancel request.
Continuing with System Report request...
Collecting Disk Array Data...SmartArray Equipped server detected...Done
Collecting Remote Console Data...Done
Collecting Model Specific System Diagnostic Information...Done
Collecting Environmental Data...Done
Collecting Remote Console System Log Data...Done
Creating single compressed system report...Done
System report written to SystemReport-20070730020505.tgz
To retrieve diagnostics use CLI command:
file get activelog platform/log/SystemReport-20070730020505.tgz

admin:utils iostat CLI
Provides the iostat output for the given number of iterations and interval. Displays the interval in seconds between two iostat readings and the number of iostat iterations to be performed. This displays:

Executing command... Please be patient
Tue Oct  9 12:47:09 IST 2007
Linux 2.4.21-47.ELsmp (csevdir60)
10/09/2007 Time—12:47:09 PM
avg-cpu %user %nice %sys %iowait %idle
3.61 0.02 3.40 0.51 92.47
Device rrqm/s wrqm/s r/s w/s rsec/s wsec/s rkB/s wkB/s avgrq-sz avgqu-sz await svctm
sda 3.10 19.78 0.34 7.49 27.52 218.37 13.76 109.19 31.39 0.05 5.78 0.73
sda1 0.38 4.91 0.14 0.64 4.21 44.40 2.10 22.20 62.10 0.02 26.63 1.62
Platform Monitoring

The following CLI can be used to monitor and manage intracluster connections:

- admin:utils dbreplication status
- admin:utils dbreplication repair all/nodename
- admin:utils dbreplication reset all/nodename
- admin:utils dbreplication stop
- admin:utils dbreplication dropadmindb
- admin:utils dbreplication setrepltimeout
- show tech dbstateinfo
- show tech dbinuse
- show tech notify
- run sql <query>

Hardware Migration

Customers may wish to migrate their Cisco Unified CM to more powerful hardware, either to prepare for upgrading to a later Cisco Unified CM release that does not support the older hardware, or just to leverage capabilities only available in the more powerful hardware, such as increases in capacity/performance or RAID. The procedure is to backup from the old hardware, install the same Cisco Unified CM release to the new hardware, then restore on the new hardware.

Migrating to more powerful hardware may require a migration SKU to cover royalties Cisco owes to third-parties. If you are considering this, have your account team check the Guide to Cisco Unified CM Upgrades and Server Migrations, which is a supplement to the Cisco Unified CM Ordering Guide.

Platform Security

The following topics are covered in this section:

- Locked-down System, page 3-32
- Cisco Security Agent Support, page 3-32
- Security Patching and Updating, page 3-32
- Role-Based Access Control, page 3-32
Locked-down System

For security, Cisco Security Agent is included along with a built-in firewall controlling connectivity among all cluster nodes, via IP tables and sensitive ports defined by the application. No AntiVirus application is installed on the appliance. The native OS used by the appliance is also hardened to minimize attack surface and vulnerabilities; fewer than 200 of the thousands of available packages are used to eliminate unused software and the corresponding vulnerabilities.

No “on-box” e-mail clients or Web browsers are supported, all unnecessary logins have been removed or disabled, and all software is provided by Cisco and digitally signed to ensure it is authorized by Cisco. The GUI, CLI, and API interfaces that Cisco provides are the only methods to administer the system, and authentication is required for users to interact with them. It also useful to note that appliances of this sort are less frequently targets of malware than Microsoft Windows or other systems with open-system access to the native OS, so significantly fewer patches need to be applied to the base OS.

Cisco Unified CM regulates its TCP/UDP port usage. See the “Cisco Unified Communications Manager TCP and UDP Port Usage” document for each Cisco Unified CM release for the specific list.

Cisco Security Agent Support

The Appliance supports the “headless” or unmanaged Cisco Security Agent. A future release will add support for the event monitoring features of Cisco Security Agent Management Center, but not for policy edits and distribution.

Security Patching and Updating

The Appliance’s software image contains all security updates and patches made to firmware, drivers, native OS, database and Cisco Unified CM application components. Customers who keep current with Cisco maintenance releases are automatically covered for security updates. For more details, refer to the Application Note “Appliance Security Update Process for Cisco Unified Communications Manager” (C27-412838-00), available on request from your Cisco account team.

Role-Based Access Control

Cisco Unified CM uses Multi-Layer Admin (MLA) for RBAC control over authorization to Cisco Unified CM configuration.

Software Configuration Management

The Cisco Unified CM server uses a bundled image including all components needed for the system in a single set of DVDs or software downloads. Unlike Cisco Unified CM Release 4.x in which there were up to 6 different components to manage for a total of 18 updates per year on average to stay current, the server has 2 components with an average of 5 updates per year to stay current.

It is recommended that you keep your system current with the latest maintenance release for a major/minor feature release. Major and minor release install files are available on DVD media kits or on Product Upgrade Tool at http://www.cisco.com/cisco/software/navigator.html.

Rebuilds, upgrade files for minor and maintenance releases, and Cisco option files and tools are available as software downloads from Software Center at http://www.cisco.com/kobayashi/sw-center/sw-voice.shtml.
Customers wishing to receive automatic e-mail notification of availability of new files on Software Center should subscribe to the e-mail notification tool on that site. Engineering “special” releases are only available to customers by using Cisco Technical Assistance Center.

The following topics are described in this section:

- General Install/Upgrade Procedures, page 3-33
- Detecting Installed Release and Packages, page 3-33

**General Install/Upgrade Procedures**

Unattended first-time installs can be performed by using the Cisco Unified Communications Answer File Generator at [http://www.cisco.com/web/cuc_afg/index.html](http://www.cisco.com/web/cuc_afg/index.html). For other details, see the online help and the document Installing Cisco Unified Communications Manager.

For upgrades and from the list, find the appropriate release for your upgrade in the following index: [http://www.cisco.com/en/US/products/sw/voicesw/ps556/prod_installation_guides_list.html](http://www.cisco.com/en/US/products/sw/voicesw/ps556/prod_installation_guides_list.html)

**Detecting Installed Release and Packages**

You have several methods to display the installed release and packages that are:

- `show version [active | inactive]` and `show packages active` commands
- Cisco Unified Operations Manager
- Unified OS Administration
- Cisco Unified Communications Manager
- SNMP

A third-party NMS can query the Cisco Unified CM release by using the following SNMP OID:

- `.iso.org.dod.internet.private.enterprises.cisco.ciscoMgmt.ciscoCcmMIB.ciscoCcmMIBObjects.ccmGeneralInfo.ccm Table.ccmEntry.ccmVersion`

The Cisco Unified CM licensing web page displays the uploaded license file release, which may or may not be an exact match for what is installed on the system.

**Available Reports**

This section contains the following subsections:

- RTMT Reports, page 3-33
- Serviceability Reports, page 3-34
- Cisco Unified Reporting, page 3-34

**RTMT Reports**

RTMT has a number of pre-can screens for information such as Summary, Call Activity, Device Status, Server Status, Service Status, and Alert Status. RTMT “Summary” pre-can screen shows a summary view of Cisco Unified CM system health. It shows CPU, Memory, Registered Phones, CallsInProgress,
and ActiveGateway ports & channels. This should be one of the first thing you want to check each day to make sure CPU & memory usage are within normal range for your cluster and all phones are registered properly.

Phone Summary and Device Summary pre-can screens provide more detailed information about phone and gateway status. If there are a number of devices that fail to register, then you can use the Admin Find/List page or RTMT device search to get further information regarding the problem devices. Critical Services pre-can screen displays the current running/activation status of key services. You can access all the pre-can screens by simply clicking the corresponding icons on the left.

**Serviceability Reports**

The Cisco Serviceability Reporter service generates daily reports in Cisco Unified CallManager Serviceability Web Page. Each report provides a summary that comprises different charts that display the statistics for that particular report. Reporter generates reports once a day on the basis of logged information, such as:

- Device Statistics Report
- Server Statistics Report
- Service Statistics Report
- Call Activities Report
- Alert Summary Report
- Performance Protection Report

For detailed information about each report, go to [http://www.cisco.com/en/US/docs/voice_ip_comm/cucm/service/5_0_2/ccmsrvs/sssrvrep.html#wp1033420](http://www.cisco.com/en/US/docs/voice_ip_comm/cucm/service/5_0_2/ccmsrvs/sssrvrep.html#wp1033420)

**Cisco Unified Reporting**

Cisco Unified Reporting is accessed at the Cisco Unified CM Administration console and generates reports for troubleshooting or inspecting cluster data. It provides cluster data without requiring multiple steps to find the data. The tool design facilitates gathering data from existing sources, comparing the data, and reporting irregularities. Figure 3-13 displays the available reports. Refer to the Cisco Unified CM Administration Guide for further detailed information.
General Health and Troubleshooting Tips

This section contains the following subsections:

- Using of Onboard Agents, page 3-36
- Call Detail Records and Call Maintenance Records, page 3-36
- Perfmon Counters, page 3-36
- Integration with Uninterruptible Power Supplies (UPS), page 3-37
- Native Hardware Out of Band Management (OOB), page 3-37
- Phone Registration Status, page 3-37
- Historical Information Download, page 3-37
- Cisco CallManager Service Stops Responding, page 3-38
- Database Replication Fails Between the Publisher and the Subscriber, page 3-39
- Database Replication Does Not Occur on Lost Node, page 3-41
- Database Tables Out of Sync Do Not Trigger Alert, page 3-42
- Reset Database Replication When Reverting to Prior Release, page 3-43
Using of Onboard Agents

Onboard agents are third-party software clients, agents or daemons installed on-box, including but not limited to:

- Anti-virus clients
- Uninterruptible Power Supply monitoring agents
- Management agents

Certain types of onboard agents are supported in Cisco Unified CM Release 4.x. The appliance used by Cisco Unified CM Release 5.0 and later releases does not support installation of onboard agents, rather it exposes APIs for third-party integration.


Call Detail Records and Call Maintenance Records

CDR and CMRs are used for a variety of uses including billing, chargeback, administrative oversight and diagnostics. In addition to a canned application for managing CDR/CMR, Cisco Unified CM Release 4.x supported various means of direct database access for external systems to access the CDR/CMR data. Cisco Unified CM Release 5.0 and later releases use SFTP to push formatted files off Cisco Unified CM to the requesting application.

When CDR is activated, a CPU utilization increase of 2% is typical, 4% if both CDR and CMR are activated.

Perfmon Counters

Table 3-5 lists some equivalent perfmon counters between Cisco Unified CM Release 4.x and Release 5.x and later.

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 4.x Perfmon Counters</th>
<th>Cisco Unified CM Release 5.x Perfmon Counters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process % Privileged Time</td>
<td>Process % Privileged Time</td>
</tr>
<tr>
<td>% Processor Time</td>
<td>Process % Privileged Time</td>
</tr>
<tr>
<td>Processor % UserTime</td>
<td>Processor % UserTime</td>
</tr>
<tr>
<td>% Privileged Time</td>
<td>% Privileged Time</td>
</tr>
<tr>
<td>% Idle Time</td>
<td>% Idle Time</td>
</tr>
<tr>
<td>% Processor Time</td>
<td>% Processor Time</td>
</tr>
</tbody>
</table>
Integration with Uninterruptible Power Supplies (UPS)

As of Cisco Unified CM Release 6.0(1a) and later, the server supports integration with certain models of APC UPS for certain MCS 7800 models. Previous server releases rely on an external script monitoring the UPS and issuing the Cisco CLI for graceful shutdown. See the release notes for Cisco Unified CM 6.0(1b) for more details at http://www.cisco.com/en/US/docs/voice_ip_comm/cucm/rel_notes/6_0_1/cucm_rel_note-601b.html.

Note
Native hardware out-of-band management such as HP iLO or IBM RSA II cannot be used for graceful shutdown of Cisco Unified CM.

Native Hardware Out of Band Management (OOB)

The supported features of HP iLO and IBM RSA II are enabled for the following areas:

- CPU status/utilization
- Memory status/utilization
- System components temperatures
- Fan status
- Power Supply status
- RAID & disk status
- Network status including NIC
- Operational status, including instrumentation of system/kernel status and data dumps following major system issues, indicating nature/type of the operational problem and degree of severity.

Support of these interfaces on the server includes the following capabilities (specific feature names vary by hardware vendor):

- Remote console (to access boot screens and the Cisco CLI)
- Remote power management

Phone Registration Status

Phone registration status needs to be monitored for sudden changes. If the registration status changes slightly and readjusts quickly over a short time frame, then it could be indicative of phone move, add, or change. A sudden smaller drop in phone registration counter can be indicative of a localized outage, for instance an access switch or a WAN circuit outage or malfunction. A significant drop in registered phone level needs immediate attention by the administrator. This counter especially needs to be monitored before and after the upgrades to ensure the system is restored completely.

Historical Information Download

You can also download some historical information using RTMT Trace Log Center or SOAP APIs, such as:

- Cisco AMC Service PerfMonLog is enabled by default but deprecated in Cisco Unified CM Release 6.0 because Cisco RIS Data Collector PerfMonLog is introduced.
• Cisco RIS Data Collector PerfMonLog is disabled by default in Cisco Unified CM Release 5.x and enabled by default in Cisco Unified CM Release 6.0.

Cisco CallManager Service Stops Responding

When the Cisco CallManager service stops responding, the following message displays in the System Event log:

The Cisco CallManager service terminated unexpectedly.
It has done this 1 time. The following corrective action will be taken in 60000 ms. Restart the service.

Other messages you may see in this situation:

Timeout 3000 milliseconds waiting for Cisco CallManager service to connect.

The Cisco Communications Manager failed to start due to the following error:
The service did not respond to the start or control request in a timely fashion.

At this time when devices such as the Cisco Unified IP Phones and gateways, unregister from the Cisco Unified Communications Manager, users receive delayed dial tone, and/or the Cisco Unified Communications Manager server freezes due to high CPU usage. For event log messages that are not included here, view the Cisco Unified Communications Manager Event Logs.

Possible Cause
The Cisco CallManager service can stop responding because the service does not have enough resources such as CPU or memory to function. Generally, the CPU utilization in the server is 100 percent at that time.

Recommended Action
Depending on what type of interruption you experience, you will need to gather different data that will help determine the root cause of the interruption.
Use the following procedure if a lack of resources interruption occurs.

Procedure

---

Step 1 Collect Cisco CallManager traces 15 minutes before and after the interruption.
Step 2 Collect SDL traces 15 minutes before and after the interruption.
Step 3 Collect perfmon traces if available.
Step 4 If the traces are not available, start collecting the perfmon traces and track memory and CPU usage for each process that is running on the server. These will help in the event of another lack of resources interruption.
Database Replication Fails Between the Publisher and the Subscriber

Replicating the database represents a core function of Cisco Unified Communications Manager clusters. The server with the master copy of the database acts as the publisher (first node), while the servers that replicate the database comprise subscribers (subsequent nodes).

Tip
Before you install Cisco Unified Communications Manager on the subscriber server, you must add the subscriber to the Server Configuration window in Cisco Unified Communications Manager Administration to ensure that the subscriber replicates the database that exists on the publisher database server. After you add the subscriber server to the Server Configuration window and then install Cisco Unified Communications Manager on the subscriber, the subscriber receives a copy of the database that exists on the publisher server.

Changes that are made on the publisher server are not reflected on phones that are registered with the subscriber server.

Possible Cause
Replication fails between the publisher and subscriber servers.

Recommended Action
Verify and, if necessary, repair database replication, as described in the following procedure:

Procedure

Step 1 Verify database replication. You can use the CLI, Cisco Unified Reporting, or RTMT to verify database replication.

- To verify using the CLI, see Step 2.
- To verify using Cisco Unified Reporting, see Step 3.
- To verify using RTMT, see Step 4.

Step 2 To verify database replication using the CLI, access the CLI and issue the following command to check replication on each node. You will need to run this CLI command on each node to check its replication status. Also, after a subscriber is installed, depending on the number of subscribers, it may take a considerable amount of time to achieve a status of 2.

```
admin: show perf query class "Number of Replicates Created and State of Replication"
``` 

- Perf class (Number of Replicates Created and State of Replication)
  has instances and values:
  ReplicateCount  -> Number of Replicates Created   = 344
  ReplicateCount  -> Replicate_State                = 2

Be aware that the Replicate_State object shows a value of 2 in this case. The following list shows the possible values for Replicate_State:

- 0—This value indicates that replication did not start. Either no subsequent nodes (subscribers) exist, or the Cisco Database Layer Monitor service is not running and has not been running since the subscriber was installed.
- 1—This value indicates that replicates have been created, but their count is incorrect.
- 2—This value indicates that replication is good.
Chapter 3  Managing and Monitoring the Health of Cisco Unified Communications Manager Systems

General Health and Troubleshooting Tips

- 3—This value indicates that replication is bad in the cluster.
- 4—This value indicates that replication setup did not succeed.

Step 3: To verify database replication using Cisco Unified Reporting, perform the following tasks:

a. From the Navigation drop-down list box in the upper, right corner in Cisco Unified Communications Manager Administration, choose Cisco Unified Reporting.

b. After Cisco Unified Reporting displays, click System Reports.

c. Generate and view the Cisco Unified CM Database Status report, which provides debugging information for database replication.

Once you have generated the report, open it and look at the Cisco Unified CM Database Status. It gives the RTMT replication counters for all servers in the cluster. All servers should have a replicate state of 2, and all servers should have the same number of replicates created.

If you see any servers whose replicate states are not equal to 2 in the above status check, inspect the “Replication Server List” on this report. It shows which servers are connected and communicating with each node. Each server should show itself as local (in its list) and the other servers as active connected. If you see any servers as dropped, it usually means there is a communication problem between the nodes.

d. If you want to do so, generate and view the Cisco Unified CM Database Status report, which provides a snapshot of the health of the Cisco Unified Communications Manager database.

Step 4: To verify database replication using RTMT, perform the following tasks:

a. Open the Cisco Unified Real-Time Monitoring Tool (RTMT).

b. Click the CallManager tab.

c. Click Database Summary. The Replication Status pane displays.

The following list shows the possible values for the Replication Status pane:

- 0—This value indicates that replication has not started. Either no subsequent nodes (subscribers) exist, or the Cisco Database Layer Monitor service is not running and has not been running since the subscriber was installed.

- 1—This value indicates that replicates have been created, but their count is incorrect.

- 2—This value indicates that replication is good.

- 3—This value indicates that replication is bad in the cluster.

- 4—This value indicates that replication setup did not succeed.

d. To view the Replicate State performance monitoring counter, choose System > Performance > Open Performance Monitoring. Double-click the publisher database server (first node) to expand the performance monitors. Click Number of Replicates Created and State of Replication. Double-click Replicate State. Click ReplicateCount from the Object Instances window and click Add.

Tip: To view the definition of the counter, right click the counter name and choose Counter Description.

Step 5: If all the servers have a good RTMT status, but you suspect the databases are not in sync, you can run the CLI command `utils dbreplication status` (If any of the servers showed an RTMT status of 4, proceed to Step 6).
This status command can be run on all servers by using `utils dbreplication status all` or on one subscriber by using `utils dbreplication status <hostname>`.

The status report will tell you if any tables are suspect. If there are suspect tables, you will want to do a replication repair CLI command to sync the data from the publisher server to the subscriber servers.

The replication repair can be done on all subscriber servers (using the `all` parameter) or on just one subscriber server by using the following: `utils dbreplication repair usage:utils dbreplication repair [nodename]|all`.

After running the replication repair, which can take several minutes, you can run another status command to verify that all tables are now in sync. If tables are in sync after running the repair, you are successful in fixing replication.

---

**Note**

Only do **Step 6** if one of the servers showed an RTMT status of 4, or had a status of 0 for more than four hours.

---

**Step 6**

Generate and view the **Cisco Unified CM Database Status** report, which provides debugging information for database replication. For each subscriber server that has a bad RTMT status, check that the hosts, rhosts, sqlhosts, and services files have the appropriate information.

Generate and view the **Cisco Unified CM Cluster Overview** report. Verify that the subscriber servers have the same version, verify that connectivity is good, and verify that time delay is within tolerances.

If the preceding conditions are acceptable, do the following to reset replication on that subscriber server:

a. At the subscriber server, perform the CLI command `utils dbreplication stop`.
   Do this for all subscriber servers that have an RTMT value of 4.

b. At the publisher server, perform the CLI command `utils dbreplication stop`.

c. At the publisher server, perform the CLI command `utils dbreplication reset <hostname>` where `<hostname>` is the hostname of the subscriber server that needs to be reset. If all subscriber servers need to be reset, use command `utils dbreplication reset all`.

---

**Database Replication Does Not Occur on Lost Node**

Database replication does not occur when connectivity is restored on lost node recovery. You can verify the state of replication by using the methods given in the topic **Database Replication Fails Between the Publisher and the Subscriber**, page 3-39. Only use the following procedure if you have already tried to reset replication on the node, and have been unsuccessful.

**Possible Cause**

The CDR check remains stuck in a loop, due to a delete on device table.

**Recommended Action**

---

**Step 1**

Run `utils dbreplication stop` on the affected subscribers. You can run them all at once.

**Step 2**

Wait until **Step 1** completes, then, run `utils dbreplication stop` on the affected publisher server.
Run `utils dbreplication clusterreset` from the affected publisher server. When you run the command, the log name gets listed in the log file. Watch this file to monitor the process status. The path to the follows:
/var/log/active/cm/trace/dbl/sdi

**Step 3**  From the affected publisher, run `utils dbreplication reset all`.

**Step 4**  Stop and restart all the services on all the subscriber servers [or restart/reboot all the systems (subscriber servers)] in the cluster to get the service changes. Do this only after `utils dbreplication status` shows Status 2.

---

**Database Tables Out of Sync Do Not Trigger Alert**

Out of sync means that two servers in the cluster do not contain the same information in a specific database table.

On Cisco Unified Communications Manager Version 6.x or later, the symptoms include unexpected call processing behaviors. Calls do not get routed or handled as expected. The symptoms may occur on either the publisher or on the subscriber servers.

On Cisco Unified Communications Manager Version 5.x, the symptoms include unexpected call processing behaviors. Calls do not get routed or handled as expected but only when the publisher server is offline. If you see these symptoms, you can run the `utils dbreplication status` command “Out of sync” displays. If “Out of sync” does not display, this is not the problem.

**Possible Cause**

Database tables remain out of sync between nodes. Replication alerts only indicate failure in the replication process and do not indicate when database tables are out of sync. Normally, if replication is working, tables should remain in sync. Instances can occur in which replication appears to be working, but database tables are “Out of sync”.

**Recommended Action**

**Step 1**  Reset cluster replication by using CLI commands. Ensure servers in the cluster are online with full IP connectivity for this to work. Confirm that all servers in the cluster are online by using platform CLI and Cisco Unified Reporting.

**Step 2**  If the servers are in Replication State 2, use the `utils dbreplication repair server name` command on the publisher server.

If the servers are not in Replication State 2, use the `utils dbreplication stop` command on all subscriber servers.

Then, use the `utils dbreplication stop` and then `utils dbreplication reset all` commands on the publisher server.
Reset Database Replication When Reverting to Prior Release

If you revert the servers in a cluster to run an older product release, you must manually reset database replication within the cluster. To reset database replication after you revert all the cluster servers to the older product release, use the `utils dbreplication reset` command all on the publisher server.

When you switch versions by using Cisco Unified Communications Operating System Administration or the CLI, you get a message reminding you about the requirement to reset database replication if you are reverting to an older product release.

Useful Commands and Utilities

This section provides a quick reference for commands and utilities to help you troubleshoot a Cisco Unified Communications Manager server with root access disabled.

Table 3-6 provides a summary of the CLI commands and GUI selections that you can use to gather information troubleshooting various system problems.

<table>
<thead>
<tr>
<th>Information</th>
<th>Linux Command</th>
<th>Serviceability GUI Tool</th>
<th>CLI commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU usage</td>
<td>top</td>
<td>RTMT</td>
<td>Processor CPU usage: show perf query class Processor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Go to View tab and select Server &gt; CPU and Memory</td>
<td>Process CPU Usage for all processes: show perf query counter Process “% CPU Time”</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Individual process counter details (including CPU usage) show perf query instance &lt;Process task_name&gt;</td>
</tr>
<tr>
<td>Process state</td>
<td>ps</td>
<td>RTMT</td>
<td>show perf query counter Process “Process Status”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Go to View tab and select Server &gt; Process</td>
<td></td>
</tr>
<tr>
<td>Disk usage</td>
<td>df/du</td>
<td>RTMT</td>
<td>show perf query counter Partition “% Used”</td>
</tr>
<tr>
<td>Memory</td>
<td>free</td>
<td>RTMT</td>
<td>or show perf query class Partition</td>
</tr>
<tr>
<td>Network status</td>
<td>netstats</td>
<td>RTMT</td>
<td>show perf query class Memory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Go to View tab and select Server &gt; CPU and Memory</td>
<td>show network status</td>
</tr>
</tbody>
</table>
**Related Documentation**

It supplements but does not replace the existing documentation including the following:

  - *Cisco Unified Communications Manager Serviceability Administration Guide*
  - *Cisco Unified Communications Manager Serviceability System Guide*
  - *Changing the IP Address and Hostname for Cisco Unified Communications Manager 5.x, 6.x, and 7.x Servers*
  - *Cisco Unified Communications Real-Time Monitoring Tool Administration Guide*
  - *Cisco Unified Communications Operating System Administration Guide*
  - *Disaster Recovery System Administration Guide*
  - *Replacing a Single Server or Cluster for Cisco Unified Communications Manager*
  - *Upgrading to Cisco Unified Communications Manager*
  - *Installing Cisco Security Agent for Cisco Unified Communications Manager*

For documentation for CDR/CMR, see the following documents:

- For Cisco Unified CM Release 8.0(1)
- For Cisco Unified CM Release 6.1(1)
- For Cisco Unified CM Release 6.0(1)
- For Cisco Unified CM Release 5.1(3)
- For Cisco Unified CM Release 5.0(4)
Simple Network Management Protocol

This chapter gives an overview of Simple Network Management Protocol (SNMP). It contains the following sections:

- Overview, page 4-1
- SNMP Versioning, page 4-2
- SNMP and Cisco Unified CM Basics, page 4-3
- SNMP Basic Commands, page 4-3
- SNMP Community Strings and Users, page 4-4
- SNMP and Cisco MIBs, page 4-4
- SNMP Traps and Informs, page 4-5
- SNMP Trace Configuration, page 4-5
- SNMP Tips, page 4-5
- SNMP Troubleshooting, page 4-6

Overview

Simple Network Management Protocol (SNMP), an application layer protocol, facilitates the exchange of management information among network devices, such as nodes and routers. It comprises part of the TCP/IP suite. System administrators can remotely manage network performance, find and solve network problems, and plan for network growth by using SNMP.

Instead of defining a large set of commands, SNMP places all operations in a *get-request*, *get-next-request*, *get-bulk-request*, and *set-request* format. For example, an SNMP manager can get a value from an SNMP agent or store a value in that SNMP agent. The SNMP manager can comprise part of a network management system (NMS), and the SNMP agent can reside on a networking device such as a router.

SNMP comprises of three parts—SNMP manager, SNMP agent, and MIBs. You can compile the Cisco MIB with your network management software.

The NMS uses the Cisco MIB variables to set device variables and to poll devices on the internetwork for specific information. The results of a poll can get graphed and analyzed to help you troubleshoot internetwork problems, increase network performance, verify the configuration of devices, and monitor traffic loads.
The SNMP agent gathers data from the MIB, which is the repository for information about device parameters and network data. The SNMP agent also can send traps (notifications) of certain events, to the SNMP manager. The Cisco host //ftp.cisco.com makes available the Cisco trap file, “mib.traps,” which documents the format of Cisco traps.

The SNMP manager uses information in the MIB to perform the operations as described:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>get-request</td>
<td>Retrieve a value from a specific variable.</td>
</tr>
<tr>
<td>get-next-request</td>
<td>Retrieve the value following the named variable. Often used to retrieve variables from within a table. With this operation, an SNMP manager does not need to know the exact variable name. A sequential search gets performed to find the needed variable from within the MIB.</td>
</tr>
<tr>
<td>get-response</td>
<td>Reply to get-request, get-next-request, get-bulk-request, and set-request that an NMS sent.</td>
</tr>
<tr>
<td>get-bulk-request</td>
<td>Fills the get-response with up to max-repetition number of get-next interactions, similar to get-next-request.</td>
</tr>
<tr>
<td>set-request</td>
<td>Store a value in a specific variable.</td>
</tr>
<tr>
<td>traps</td>
<td>Sent by an SNMP agent to an SNMP manager to indicate that some event occurred.</td>
</tr>
</tbody>
</table>

**SNMP Versioning**

Three versions of SNMP exist: version 1 (SNMPv1), version 2 (SNMPv2), and version 3 (SNMPv3). SNMPv1 represents the initial implementation of SNMP that functions within the specifications of the Structure of Management Information (SMI) and operates over protocols, such as User Datagram Protocol (UDP) and IP.

The SNMPv1 SMI defines highly structured MIB tables that are used to group objects that contain multiple variables. Tables contain zero or more rows, which are indexed, so SNMP can retrieve or alter an entire row with a supported command.

With SNMPv1, the NMS issues a request, and managed devices return responses. Agents use the Trap operation to asynchronously inform the NMS of a significant event.

As with SNMPv1, SNMPv2c functions within the specifications of SMI. MIB modules contain definitions of interrelated managed objects. Be aware that the operations that are used in SNMPv1 are similar to those that are used in SNMPv2. The SNMPv2 trap operation, for example, serves the same function as that used in SNMPv1, but it uses a different message format and replaces the SNMPv1 trap.

The Inform operation in SNMPv2c enables one NMS to send trap information to another NMS and to receive a response from the NMS.

SNMPv3 provides the following security features:

- Authentication—Verifying that the request comes from a genuine source.
- Privacy—Encrypting data.
- Authorization—Verifying that the user allows the requested operation.
- Access control—Verifying that the user has access to the objects that are requested.
SNMPv3 prevents packets from being exposed on the network. Instead of using community strings like SNMP v1 and v2, SNMP v3 uses SNMP users, as described in the “SNMP Community Strings and Users” section on page 4-4.

### SNMP and Cisco Unified CM Basics

A network that uses SNMP requires three key components—managed devices, agents, and network management software (NMS).

- **Managed devices**—Devices that contain SNMP agents and reside on a network. Managed devices collect and store information and make it available by using SNMP.
  - The first node in the Cisco Unified CM cluster acts as the managed device. In Cisco Unified CM, the server on which Cisco Unified CM is installed acts as the managed device.

- **Agents**—Software modules that contain local knowledge of management information and translates it into a form that is compatible with SNMP.
  - Cisco Unified CM uses a master agent and subagent components to support SNMP. The master agent acts as the agent protocol engine and performs the authentication, authorization, access control, and privacy functions that relate to SNMP requests. It contains a few Management Information Base (MIB) variables. The master agent also connects and disconnects subagents after the subagent completes necessary tasks.
  - Cisco Unified CM uses a subagent to interact with the local Cisco Unified CM only. The Cisco Unified CM subagents send trap and information messages to the SNMP Master Agent, and the SNMP Master Agent communicates with the SNMP trap receiver (notification destination).

- **NMS**—SNMP management application that runs on a PC and provides the bulk of the processing and memory resources that are required for network management. It executes applications that monitor and control managed devices. Cisco Unified Communications Manager works with the following NMS:
  - CiscoWorks2000
  - HP OpenView
  - Third-party applications that support SNMP and Cisco Unified Communications Manager SNMP interfaces

### SNMP Basic Commands

Managed devices get monitored and controlled by using four basic SNMP commands: read, write, trap, and traversal operations.

- **NMS uses the read command to monitor managed devices.** The NMS examines different variables that are maintained by managed devices.

- **NMS uses the write command to control managed devices.** The NMS changes the values of variables stored within managed devices.

- **Managed devices use the trap command to asynchronously report events to the NMS.** When certain types of events occur, a managed device sends a trap to the NMS.

- **NMS uses traversal operations to determine which variables a managed device supports and to sequentially gather information in variable tables, such as a routing table.**
SNMP Community Strings and Users

Although SNMP community strings provide no security, the strings authenticate access to MIB objects and function as embedded passwords. You configure SNMP community strings for SNMP v1 and v2c only.

SNMP v3 does not use community strings. It uses SNMP users that serve the same purpose as community strings but provide security because encryption or authentication is configured.

No default community string or user exists.

SNMP and Cisco MIBs

You can access the Cisco MIB variables by using SNMP which facilitates the exchange of management information between network devices. The SNMP system comprises three parts: SNMP manager, SNMP agent, and MIB.

Instead of defining a large set of commands, SNMP places all operations in a get-request, get-next-request, get-bulk-request, and set-request format. For example, an SNMP manager can get a value from an SNMP agent or store a value in that SNMP agent. The SNMP manager can comprise part of a network management system (NMS), and the SNMP agent can reside on a networking device such as a router. You can compile the Cisco MIB with your network management software. If SNMP is configured on a router, the SNMP agent can respond to MIB-related queries that are being sent by the NMS.

The NMS uses the Cisco MIB variables to set device variables and to poll devices on the internetwork for specific information. The results of a poll can get graphed and analyzed to help you troubleshoot internetwork problems, increase network performance, verify the configuration of devices, monitor traffic loads, and more.

The SNMP agent gathers data from the MIB, which provides the repository for information about device parameters and network data. The SNMP agent also can send traps (notifications) of certain events, to the SNMP manager. The Cisco host //ftp.cisco.com makes available the Cisco trap file, “mib.traps,” which documents the format of Cisco traps.

The SNMP manager uses information in the MIB to perform the operations as described:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>get-request</td>
<td>Retrieve a value from a specific variable.</td>
</tr>
<tr>
<td>get-next-request</td>
<td>Retrieve the value following the named variable. Often used to retrieve</td>
</tr>
<tr>
<td></td>
<td>variables from within a table. With this operation, an SNMP manager does not</td>
</tr>
<tr>
<td></td>
<td>need to know the exact variable name. A sequential search is performed to</td>
</tr>
<tr>
<td></td>
<td>find the needed variable from within the MIB.</td>
</tr>
<tr>
<td>get-response</td>
<td>The reply to a get-request, get-next-request, get-bulk-request, and set-request sent by an NMS.</td>
</tr>
<tr>
<td>get-bulk-request</td>
<td>Similar to get-next-request, but fills the get-response with up to</td>
</tr>
<tr>
<td></td>
<td>max-repetition number of get-next interactions.</td>
</tr>
<tr>
<td>set-request</td>
<td>Store a value in a specific variable.</td>
</tr>
<tr>
<td>traps</td>
<td>An unsolicited message sent by an SNMP agent to an SNMP manager indicating</td>
</tr>
<tr>
<td></td>
<td>that some event has occurred.</td>
</tr>
</tbody>
</table>
SNMP Traps and Informs

An SNMP agent sends notifications in the form of traps or informs to identify important system events. Traps do not receive acknowledgments from the destination whereas informs do receive acknowledgments.

Note
Cisco Unity Connection does not support SNMP traps.

For all notifications, the system sends traps immediately if the corresponding trap flags are enabled. In the case of the syslog agent, the Cisco Unified CM alarms and system-level log messages get sent to syslog daemon for logging. Also, some standard third-party applications send the log messages to syslog daemon for logging. These log messages get bagged locally in the syslog files and also get converted into SNMP traps/notifications.

The following list contains Cisco Unified CM SNMP trap and inform messages that are sent to a configured trap destination:
- Cisco Unified CM failed
- Phone failed
- Phones status update
- Gateway failed
- Media resource list exhausted
- Route list exhausted
- Gateway layer 2 change
- Quality report
- Malicious call
- Syslog message generated

SNMP Trace Configuration

For Cisco Unified CM, you can configure traces for the SNMP agent in the Trace Configuration window in Cisco Unified Serviceability by choosing the Cisco Unified CM SNMP Service in the Performance and Monitoring Services service group. A default setting exists for all the agents. For Cisco CDP Agent and Cisco Syslog Agent, you use the command line interface (CLI) to changetrace settings, as described in the Command Line Interface Reference Guide for Cisco Unified Solutions.

SNMP Tips

Refer to the CISCO-CCM-CAPABILITY-MIB at http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2&mibName=CISCO-CCM-CAPABILITY or “CISCO-CCM-CAPABILITY” section on page 7-121. As stated in the CISCO-CCM-CAPABILITY-MIB, ccmPhoneDevicePoolIndex does not get supported, so it returns a 0. The Callmanager device registration alarm currently does not contain the device pool information.
If Cisco CallManager SNMP service is not running, only the following tables in the MIB respond:

- ccmGroupTable
- ccmRegionTable
- ccmRegionPairTable
- ccmDevicePoolTable
- ccmProductTypeTable
- ccmQualityReportAlarmConfigInfo
- ccmGlobalInfo

To get Cisco CallManager SNMP service running, activate and start the service in Cisco Unified Serviceability. Query the following tables in the SYSAPPL-MIB:

- SysApplInstallPkgTable to get an inventory of Cisco Unified Communications Manager applications that are installed on the system.
- SysApplRunTable to get an inventory of Cisco Unified Communications Manager applications that are running on the system.

**Note**
Cisco Unified Communications Manager uses the following Web application services and servlets:
Cisco CallManager Admin, Cisco CallManager Cisco IP Phone Services, Cisco CallManager Personal Directory, Cisco CallManager Serviceability, Cisco CallManager Serviceability RTMT, Cisco Extension Mobility, Cisco Extension Mobility Application, Cisco RTMT Reporter Servlet, Cisco Tomcat Stats Servlet, Cisco Trace Collection Servlet, Cisco AXL Web Service, Cisco Unified Mobile Voice Access Service, Cisco Extension Mobility, Cisco IP Manager Assistant, Cisco Web Dialer Service, Cisco CAR Web Service, and Cisco Dialed Number Analyzer.

### SNMP Troubleshooting

In general ensure that all the feature and network services are running and verify that the community string or SNMP user is properly configured on the Cisco Unified CM system. You configure the SNMP community string or user by choosing **SNMP > V1/V2 > Community String** or **SNMP > V3 > User** in Cisco Unified Serviceability.

Other tips are as follows:

- **Cannot poll any MIBs from the system**—This condition means that the community string or the SNMP user is not configured on the system or they do not match with what is configured on the system. Check the configuration and reconfigure if necessary.

**Note**
By default, no community string or user is configured on the system.

- **Cannot receive any notifications from the system**—This condition means that the notification destination is not configured correctly on the system. Verify that you configured the notification destination properly in the Notification Destination (V1/V2c or V3) Configuration window.

- **Cannot receive SNMP traps from Cisco Unified Communications Manager node**—Verify that you configured the following MIB Object IDentifiers (OIDs) that relate to phone registration/deregistration/failure to the following values (the default for both values equals 0):
- `ccmPhoneFailedAlarmInterval (1.3.6.1.4.1.9.9.156.1.9.2)` set to 30-3600. You can use this CLI command: `snmpset -c <community string> -v2c <transmitter ipaddress> 1.3.6.1.4.1.9.9.156.1.9.2.0 i <value>`

- `ccmPhoneStatusUpdateAlarmInterval (1.3.6.1.4.1.9.9.156.1.9.4)` set to 30-3600. You can use this CLI command: `snmpset -c <community string> -v2c <transmitter ipaddress> 1.3.6.1.4.1.9.9.156.1.9.4.0 i <value>`

  Verify that you configured the notification destination properly in the Notification Destination (V1/V2c or V3) Configuration window.

  Verify that you configured the community string/user privileges correctly, including Notify permissions, in the Community String (V1/V2c) or User (V3) Configuration window.

Because System Application Agent cannot show services that are activated and deactivated or monitor Web App services or servlets, use this approach to monitor system health and service status for Cisco Unified Communications Manager applications:

- Use the Serviceability API `getservicestatus` to provide complete status information, including activation status, for both Web applications and non-Web applications. See the AXL Serviceability API Guide for more details.

- Check service status with this CLI command: `utils service list`

- Monitor the servM-generated messages with Syslog (see the following example):

  ```
  Mar 18 16:40:52 ciscart26 local7 6 : 92: Mar 18 11:10:52.630 UTC :
  %CCM_SERVICEMANAGER-SERVICEMANAGER-6-ServiceActivated: Service Activated. Service
  Name:Cisco CallManager SNMP Service App ID:Cisco Service Manager Cluster ID: Node
  ID:ciscart26
  ```

If an SNMP request specifies multiple OIDs and the variables are pointing to empty tables, you may get a NO_SUCH_NAME (for SNMP V1) or GENERIC ERROR (for SNMP V2c or V3) due to a timeout problem. A timeout can occur as a result of throttling enhancements to protect the Cisco Unified Communications Manager processing engine.

You can retrieve the count of entries in CCMH323DeviceTable and ccmSIPDeviceTable by using scalar objects, so the SNMP Manager (the client) can avoid unnecessary `get/getnext` operations on these tables when no entries exist. As an SNMP developer, you can use the following workaround for this problem:

- Use the available scalar variables (1.3.6.1.4.1.9.9.156.1.5) to determine table size before accessing the table or perform the `get` operation on the desired table; then, query the non-empty tables.

- Reduce the number of variables that are queried in a single request; for example, for empty tables, if the management application has the timeout set to 3 seconds, specify only 1 OID. (For non-empty tables, it takes 1 second to retrieve one row of data.)

- Increase the response timeout.

- Reduce the number of retries.

- Avoid using `getbulk` SNMP API. The `getbulk` API retrieves the number of records that is specified by MaxRepetitions, so even if the next object goes outside the table or MIB, it gets those objects. Empty tables cause even more delay. Use `getbulk` API for non-empty tables with a known number of records. In these circumstances, set MaxRepetitions to 5 seconds to require a response within 5 seconds.

- Structure SNMP queries to adapt to existing limits.

- Avoid performing multiple `getbulks` to walk the PhoneTable periodically in case a large number of phones are registered to Cisco CallManager. You can use the `ccmPhoneNumberStatusUpdateTable`, which updates whenever there is a Phone update, to decide whether to walk the PhoneTable.
For more information about MIBs and troubleshooting, refer to the following chapters:

- Chapter 7, “Cisco Management Information Base”
- Chapter 8, “Industry-Standard Management Information Base”
- Chapter 9, “Vendor-Specific Management Information Base”

**SNMP/R MIBs**

When SNMP/R binaries spike the CPU, collect the following logs and information for analysis:

- Note the processes that are experiencing high CPU usage.
- Check to see if any SNMP polling is occurring and get the polling interval of the application.
- Note the SNMP versions by using the `show packages active snmp` command.
- Execute the `show process using-most cpu` command and collect the output.
- Collect the Perfmon logs by executing the `file get activelog /cm/log/ris/csv/` command.
- Collect the traces for SNMP Master Agent, and other binaries experiencing high CPU.
- Send the above information to Support for further troubleshooting.

When the SNMP Master Agent does not start, check to see if port 161 is open. If the port is open, collect the SNMP Master Agent traces for further analysis.

When migrating from Windows to Linux Cisco Unified CM, the `ccmH323DevRmtCM1InetAddress` has been defined as OctetString in Cisco Unified CM Release 5.x and later. So, the IP Address displays as Hexadecimal instead of the dotted decimal format as displayed in Cisco Unified CM Release 4.x.
CHAPTER 5

Cisco Unified Real-Time Monitoring Tool
Tracing, PerfMon Counters, and Alerts

This chapter briefly describes the Cisco Unified Communications Real-Time Monitoring Tool (RTMT) tracing capabilities, perfmon objects and counters, and alerts. It contains the following sections:

- Cisco Unified Real-Time Monitoring, page 5-1
- Performance Monitoring in RTMT, page 5-2
- Cisco Intercompany Media Engine Performance Objects and Alerts, page 5-74

Cisco Unified Real-Time Monitoring

The RTMT runs as a client-side application and uses HTTPS and TCP to monitor system performance, device status, device discovery, CTI applications, and voice messaging ports. RTMT can connect directly to devices by using HTTPS to troubleshoot system issues. Cisco Unified RTMT performs the following tasks:

- Monitor a set of predefined management objects that monitor the health of the system.
- Generate various alerts, in the form of e-mails, for objects when values go over/below user-configured thresholds.
- Collect and view traces in various default viewers that exist in RTMT.
- Translate Q931 messages.
- View syslog messages in SysLog Viewer.
- Work with performance-monitoring counters.

In addition to SNMP traps, Cisco Unified RTMT can monitor and parse syslog messages that are provided by the hardware vendors, and then send these alerts to RTMT Alert Central. You can configure RTMT to notify the Cisco Unified CM system administrator when the alerts occur. The notifications can occur by using e-mail or Epage or both.

Note

Be aware the RTMT is best used for a single cluster. For large and enterprise networks that have multiple clusters deployed, Cisco recommends using Cisco Unified Operations Manager. For details about Cisco Unified Operations Manager, go to http://www.cisco.com/en/US/products/ps6535/index.htm.
Performance Monitoring in RTMT

Cisco Unified Communications Manager updates performance counters (called PerfMon counters). The counters contain simple, useful information about the system and devices on the system, such as number of registered phones, number of active calls, number of available conference bridge resources, and voice messaging port usage.

You can monitor the performance of the components of the system and the components for the application on the system by choosing the counters for any object. The counters for each object display when the folder expands.

For Cisco Unified Communications Manager, the Cisco CallManager object contains most of the Cisco Unified Communications Manager performance counters, and these counters have only one instance. The instance-based counters that belong to the other objects can have zero or more instances. For example, if two phones are registered to Cisco Unified Communications Manager, two instances of each counter that belong to the Cisco phones object exist.

You can log perfmon counters locally on the computer and use the performance log viewer in RTMT to display the perfmon CSV log files that you collected or the Real-time Information Server Data Collection (RISDC) perfmon logs.

RTMT provides alert notifications for troubleshooting performance. It also periodically polls performance counters to display data for that counter. Performance monitoring allows you to perform the following tasks:

- Monitor performance counters including all the Cisco Unified Communications Manager servers in a cluster (if applicable), TFTP servers, and database servers.
- Continuously monitor a set of preconfigured objects and receive notification in the form of an email message.
- Associate counter threshold settings to alert notification. An email or popup message provides notification to the administrator.
- Save and restore settings, such as counters that get monitored, threshold settings, and alert notifications, for customized troubleshooting tasks.
- Display up to six perfmon counters in one chart for performance comparisons.

This section contains the following subsections:

- PerfMon Alert Notifications, page 5-2
- PerfMon Objects and Counters for Cisco Unified Communications Manager, page 5-5
- PerfMon Objects and Counters for System, page 5-59

PerfMon Alert Notifications

The alert notifications keep you updated on system and Cisco Unified Communications Manager issues. You can use the parameters that are already contained in RTMT or configure your own. Table 5-1 lists the available settings and describes each. The Threshold, Value Calculated As, Duration, Frequency, and Schedule panes of RTMT contain the settings.
Table 5-1  Counter Alert Configuration Parameters

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Threshold Pane</strong></td>
<td></td>
</tr>
<tr>
<td>Trigger alert when Over and Under conditions get met</td>
<td>Check the box and enter the value that applies.</td>
</tr>
<tr>
<td></td>
<td>• Over—Check this box to configure a maximum threshold that</td>
</tr>
<tr>
<td></td>
<td>must be met before an alert notification is activated. In the</td>
</tr>
<tr>
<td></td>
<td>Over value field, enter a value. For example, enter a value that</td>
</tr>
<tr>
<td></td>
<td>equals the number of calls in progress.</td>
</tr>
<tr>
<td></td>
<td>• Under—Check this box to configure a minimum threshold that</td>
</tr>
<tr>
<td></td>
<td>must be met before an alert notification is activated. In the</td>
</tr>
<tr>
<td></td>
<td>Under value field, enter a value. For example, enter a value that</td>
</tr>
<tr>
<td></td>
<td>equals the number of calls in progress.</td>
</tr>
<tr>
<td><strong>Tip</strong></td>
<td>Use these boxes in conjunction with the Frequency and Schedule configuration parameters.</td>
</tr>
</tbody>
</table>

| **Value Calculated As Pane**                 |                                                                             |
| Absolute, Delta, Delta Percentage            | Click the radio button that applies.                                        |
|                                               | • Absolute—Choose Absolute to display the data at its current status. These |
|                                               |   counter values are cumulative.                                            |
|                                               | • Delta—Choose Delta to display the difference between the                  |
|                                               |   current counter value and the previous counter value.                    |
|                                               | • Delta Percentage—Choose Delta Percentage to display the                  |
|                                               |   counter performance changes in percentage.                               |

| **Duration Pane**                            |                                                                             |
| Trigger alert only when value constantly...; Trigger alert immediately | Trigger alert only when value constantly...—If you want the alert          |
|                                               |   notification only when the value is constantly below or over threshold   |
|                                               |   for a desired number of seconds, click this radio button and enter       |
|                                               |   seconds after which you want the alert to be sent.                      |
|                                               | • Trigger alert immediately—If you want the alert notification to          |
|                                               |   be sent immediately, click this radio button.                           |
Note

If you require an e-mail notifications, check the Enable E-mail box.

You can also use data sampling in RTMT. The perfmon counters that display in the RTMT Perfmon Monitoring pane have green dots that represent samples of data over time. You can configure the number of samples to collect and the number of data points to show in the chart. Table 5-2 lists and describes the parameters.

Table 5-2 Data Sample Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absolute</td>
<td>Because some counter values are accumulative, choose Absolute to display the data at its current status.</td>
</tr>
<tr>
<td>Delta</td>
<td>Choose Delta to display the difference between the current counter value and the previous counter value.</td>
</tr>
<tr>
<td>Delta Percentage</td>
<td>Choose Delta Percentage to display the counter performance changes in percentage.</td>
</tr>
</tbody>
</table>
PerfMon Objects and Counters for Cisco Unified Communications Manager

This section provides information on Cisco Unified Communications Manager PerfMon objects and counters.

Cisco Analog Access

The Cisco Analog Access object provides information about registered Cisco Analog Access gateways. Table 5-3 contains information about Cisco Analog Access counters.

Table 5-3  Cisco Analog Access

<table>
<thead>
<tr>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OutboundBusyAttempts</td>
</tr>
<tr>
<td>This counter represents the total number of times that</td>
</tr>
<tr>
<td>Cisco Unified Communications Manager attempts a call</td>
</tr>
<tr>
<td>through the analog access gateway when all ports were</td>
</tr>
<tr>
<td>busy.</td>
</tr>
<tr>
<td>PortsActive</td>
</tr>
<tr>
<td>This counter represents the number of ports that are</td>
</tr>
<tr>
<td>currently in use (active). A port appears active when</td>
</tr>
<tr>
<td>a call is in progress on that port.</td>
</tr>
<tr>
<td>PortsOutOfService</td>
</tr>
<tr>
<td>This counter represents the number of ports that are</td>
</tr>
<tr>
<td>currently out of service. Counter applies only to</td>
</tr>
<tr>
<td>loop-start and ground-start trunks.</td>
</tr>
</tbody>
</table>

Cisco Annunciator Device

The Cisco Annunciator Device object provides information about registered Cisco annunciator devices. Table 5-4 contains information about Cisco Annunciator counters.

Table 5-4  Cisco Annunciator Device

<table>
<thead>
<tr>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OutOfResources</td>
</tr>
<tr>
<td>This counter represents the total number of times that</td>
</tr>
<tr>
<td>Cisco Unified Communications Manager attempted to</td>
</tr>
<tr>
<td>allocate an annunciator resource from an annunciator</td>
</tr>
<tr>
<td>device and failed; for example, because all resources</td>
</tr>
<tr>
<td>were already in use.</td>
</tr>
<tr>
<td>ResourceActive</td>
</tr>
<tr>
<td>This counter represents the total number of annunciator</td>
</tr>
<tr>
<td>resources that are currently active (in use) for a</td>
</tr>
<tr>
<td>annunciator device.</td>
</tr>
<tr>
<td>ResourceAvailable</td>
</tr>
<tr>
<td>This counter represents the total number of resources</td>
</tr>
<tr>
<td>that are not active and are still available to be used</td>
</tr>
<tr>
<td>at the current time for the annunciator device.</td>
</tr>
<tr>
<td>ResourceTotal</td>
</tr>
<tr>
<td>This counter represents the total number of annunciator</td>
</tr>
<tr>
<td>resources that are configured for an annunciator device.</td>
</tr>
</tbody>
</table>

Cisco CallManager

The Cisco CallManager object provides information about calls, applications, and devices that are registered with the Cisco Unified Communications Manager. Table 5-5 contains information about Cisco CallManager counters.
### Table 5-5: Cisco CallManager

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AnnunciatorOutOfResources</td>
<td>This counter represents the total number of times that Cisco Unified Communications Manager attempted to allocate an annunciator resource from those that are registered to a Cisco Unified Communications Manager when none were available.</td>
</tr>
<tr>
<td>AnnunciatorResourceActive</td>
<td>This counter represents the total number of annunciator resources that are currently in use on all annunciator devices that are registered with a Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>AnnunciatorResourceAvailable</td>
<td>This counter represents the total number of annunciator resources that are not active and are currently available.</td>
</tr>
<tr>
<td>AnnunciatorResourceTotal</td>
<td>This counter represents the total number of annunciator resources that are provided by all annunciator devices that are currently registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>AuthenticatedCallsActive</td>
<td>This counter represents the number of authenticated calls that are currently active (in use) on Cisco Unified Communications Manager. An authenticated call designates one in which all the endpoints that are participating in the call are authenticated. An authenticated phone uses the Transport Layer Security (TLS) authenticated Skinny protocol signaling with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>AuthenticatedCallsCompleted</td>
<td>This counter represents the number of authenticated calls that connected and subsequently disconnected through Cisco Unified Communications Manager. An authenticated call designates one in which all the endpoints that are participating in the call are authenticated. An authenticated phone uses the TLS authenticated Skinny protocol signaling with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>AuthenticatedPartiallyRegisteredPhone</td>
<td>This counter represents the number of partially registered, authenticated SIP phones.</td>
</tr>
<tr>
<td>AuthenticatedRegisteredPhones</td>
<td>This counter represents the total number of authenticated phones that are registered to Cisco Unified Communications Manager. An authenticated phone uses the TLS authenticated Skinny protocol signaling with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>BRIChannelsActive</td>
<td>This counter represents the number of BRI voice channels that are currently in an active call on this Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>BRISpansInService</td>
<td>This counter represents the number of BRI spans that are currently available for use.</td>
</tr>
<tr>
<td>CallManagerHeartBeat</td>
<td>This counter represents the heartbeat of Cisco Unified Communications Manager. This incremental count indicates that Cisco Unified Communications Manager is up and running. If the count does not increment, that indicates that Cisco Unified Communications Manager is down.</td>
</tr>
<tr>
<td>CallsActive</td>
<td>This counter represents the number of voice or video streaming connections that are currently in use (active); in other words, the number of calls that actually have a voice path that is connected on Cisco Unified Communications Manager.</td>
</tr>
</tbody>
</table>
### Table 5-5Cisco CallManager (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsAttempted</td>
<td>This counter represents the total number of attempted calls. An attempted call occurs any time that a phone goes off hook and back on hook, regardless of whether any digits were dialed, or whether it connected to a destination. The system considers some call attempts during feature operations (such as transfer and conference) to be attempted calls.</td>
</tr>
<tr>
<td>CallsCompleted</td>
<td>This counter represents the number of calls that were actually connected (a voice path or video stream was established) through Cisco Unified Communications Manager. This number increases when the call terminates.</td>
</tr>
<tr>
<td>CallsInProgress</td>
<td>This counter represents the number of voice or video calls that are currently in progress on Cisco Unified Communications Manager, including all active calls. When a phone that is registered with Skinny Client Control Protocol (SCCP) goes off hook, the CallsInProgress progress counter increments until it goes back on hook. For Cisco Unified IP Phones 7902, 7905, 7912, 7940, and 7960 that register with SIP, the CallsInProgress counter increments when the dial softkey is pressed. For all other phones that are running SIP, the CallsInProgress counter increments when the first digit is pressed. When all voice or video calls that are in progress are connected, the number of CallsInProgress represents the number of CallsActive. The counter decreases by one when a phone goes back on hook.</td>
</tr>
<tr>
<td>CM_MediaTermPointsRequestsThrottled</td>
<td>This counter represents the total number of media termination point (MTP) resource requests that have been denied due to throttling (a resource from this MTP was not allocated because, as specified by the Cisco CallManager service parameter, MTP and Transcoder Resource Throttling Percentage, the MTP was being utilized beyond the configured throttle percentage). This counter increments each time a request for an MTP on this Cisco Unified Communications Manager (Cisco Unified CM) node is requested and denied due to MTP throttling and reflects a running total since the start of the Cisco CallManager service.</td>
</tr>
<tr>
<td>CM_TranscoderRequestsThrottled</td>
<td>This counter represents the total number of transcoder resource requests that have been denied due to throttling (a resource from this transcoder was not allocated because, as specified by the Cisco CallManager service parameter MTP and Transcoder Resource Throttling Percentage, the transcoder was being utilized beyond the configured throttle percentage). This counter increments each time a request for a transcoder on this Cisco Unified Communications Manager (Cisco Unified CM) node is requested and denied due to transcoder throttling and reflects a running total since the start of the Cisco CallManager service.</td>
</tr>
<tr>
<td>EncryptedCallsActive</td>
<td>This counter represents the number of encrypted calls that are currently active (in use) on this Cisco Unified Communications Manager. An encrypted call represents one in which all the endpoints that are participating in the call are encrypted.</td>
</tr>
<tr>
<td>EncryptedCallsCompleted</td>
<td>This counter represents the number of encrypted calls that were connected and subsequently disconnected through this Cisco Unified Communications Manager. An encrypted call represents one in which all the endpoints that are participating in the call are encrypted.</td>
</tr>
</tbody>
</table>
### Table 5-5  Cisco CallManager (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EncryptedPartiallyRegisteredPhones</td>
<td>This counter represents the number of partially registered, encrypted SIP phones.</td>
</tr>
<tr>
<td>EncryptedRegisteredPhones</td>
<td>This counter represents the total number of encrypted phones that are registered on this Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>FXOPortsActive</td>
<td>This counter represents the number of FXO ports that are currently in use (active) on a Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>FXOPortsInService</td>
<td>This counter represents the number of FXO ports that are currently available for use in the system.</td>
</tr>
<tr>
<td>FXSPortsActive</td>
<td>This counter represents the number of FXS ports that are currently in use (active) on a Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>FXSPortsInService</td>
<td>This counter represents the number of FXS ports that are currently available for use in the system.</td>
</tr>
<tr>
<td>HuntListsInService</td>
<td>This counter represents the number of hunt lists that are currently in service on Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>HWConferenceActive</td>
<td>This counter represents the total number of hardware conference resources that are provided by all hardware conference bridge devices that are currently registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>HWConferenceCompleted</td>
<td>This counter represents the total number of conferences that used a hardware conference bridge (hardware-based conference devices such as Cisco Catalyst 6000, Cisco Catalyst 4000, Cisco VG200, Cisco series 26xx and 36xx) that is allocated from Cisco Unified Communications Manager and that have completed, which means that the conference bridge has been allocated and released. A conference activates when the first call connects to the bridge. The conference completes when the last call disconnects from the bridge.</td>
</tr>
<tr>
<td>HWConferenceOutOfResources</td>
<td>This counter represents the total number of times that Cisco Unified Communications Manager attempted to allocate a hardware conference resource from those that are registered to a Cisco Unified Communications Manager when none was available.</td>
</tr>
<tr>
<td>HWConferenceResourceActive</td>
<td>This counter represents the total number of conference resources that are in use on all hardware conference devices (such as Cisco Catalyst 6000, Catalyst 4000, Cisco VG200, Cisco series 26xx and 36xx) that are registered with Cisco Unified Communications Manager. System considers conference to be active when one or more calls are connected to a bridge.</td>
</tr>
<tr>
<td>HWConferenceResourceAvailable</td>
<td>This counter represents the number of hardware conference resources that are not in use and that are available to be allocated on all hardware conference devices (such as Cisco Catalyst 6000, Cisco Catalyst 4000, Cisco VG200, Cisco series 26xx and 36xx) that are allocated from Cisco Unified Communications Manager and that have been completed, which means that the conference bridge has been allocated and released. A conference activates when the first call connects to the bridge. The conference completes when the last call disconnects from the bridge.</td>
</tr>
<tr>
<td>HWConferenceResourceTotal</td>
<td>This counter represents the number of active conferences on all hardware conference devices that are registered with Cisco Unified Communications Manager.</td>
</tr>
</tbody>
</table>
Table 5-5  Cisco CallManager (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InitializationState</td>
<td>This counter represents the current initialization state of Cisco Unified Communications Manager. Cisco Unified Communications Manager includes the following initialization state values: 1-Database; 2-Regions; 3-Locations; 4-QoS Policy; 5-Time Of Day; 6-AAR Neighborhoods; 7-Digit Analysis; 8-Route Plan; 9-Call Control; 10-RSVP Session Manager; 11-Supplementary Services; 12-Directory; 13-SDL Link; 14-Device; 100-Initialization Complete. Not all states display when this counter is used. This does not indicate that an error occurred; it simply indicates that the state(s) initialized and completed within the refresh period of the performance monitor.</td>
</tr>
<tr>
<td>LocationOutOfResources</td>
<td>This counter represents the total number of times that a call through Locations failed due to the lack of bandwidth.</td>
</tr>
<tr>
<td>MOHMulticastResourceActive</td>
<td>This counter represents the total number of multicast MOH resources that are currently in use (active) on all MOH servers that are registered with a Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>MOHMulticastResourceAvailable</td>
<td>This counter represents the total number of active multicast MOH connections that are not being used on all MOH servers that are registered with a Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>MOHOutOfResources</td>
<td>This counter represents the total number of times that the Media Resource Manager attempted to allocate a MOH resource when all available resources on all MOH servers that are registered with a Cisco Unified Communications Manager were already active.</td>
</tr>
<tr>
<td>MOHTotalMulticastResources</td>
<td>This counter represents the total number of multicast MOH resources or connections that are provided by all MOH servers that are currently registered with a Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>MOHTotalUnicastResources</td>
<td>This counter represents the total number of unicast MOH resources or streams that are provided by all MOH servers that are currently registered with Cisco Unified Communications Manager. Each MOH unicast resource uses one stream.</td>
</tr>
<tr>
<td>MOHUnicastResourceActive</td>
<td>This counter represents the total number of unicast MOH resources that are currently in use (active) on all MOH servers that are registered with Cisco Unified Communications Manager. Each MOH unicast resource uses one stream.</td>
</tr>
<tr>
<td>MOHUnicastResourceAvailable</td>
<td>This counter represents the total number of unicast MOH resources that are currently available on all MOH servers that are registered with Cisco Unified Communications Manager. Each MOH unicast resource uses one stream.</td>
</tr>
<tr>
<td>MTPOutOfResources</td>
<td>This counter represents the total number of times that Cisco Unified Communications Manager attempted but failed to allocate an MTP resource from one MTP device that is registered with Cisco Unified Communications Manager. This also means that no transcoders were available to act as MTPs.</td>
</tr>
<tr>
<td>MTPResourceActive</td>
<td>This counter represents the total number of MTP resources that are currently in use (active) on all MTP devices that are registered with a Cisco Unified Communications Manager. Each MTP resource uses two streams. An MTP in use represents one MTP resource that has been allocated for use in a call.</td>
</tr>
</tbody>
</table>
### Table 5-5  Cisco CallManager (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTPResourceAvailable</td>
<td>This counter represents the total number of MTP resources that are not in use and are available to be allocated on all MTP devices that are registered with Cisco Unified Communications Manager. Each MTP resource uses two streams. An MTP in use represents one MTP resource that has been allocated for use in a call.</td>
</tr>
<tr>
<td>MTPResourceTotal</td>
<td>This counter represents the total number of media termination point (MTP) resources that are provided by all MTP devices that are currently registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>MTP_RequestsThrottled</td>
<td>This counter represents the total number of media termination point (MTP) resource requests that have been denied due to throttling (a resource from this MTP was not allocated because, as specified by the Cisco CallManager service parameter MTP and Transcoder Resource Throttling Percentage, the MTP was being utilized beyond the configured throttle percentage). This counter increments each time a resource is requested from this MTP and is denied due to throttling. This counter reflects a running total since the MTP device registered with the Cisco CallManager service.</td>
</tr>
<tr>
<td>PartiallyRegisteredPhone</td>
<td>This counter represents the number of partially registered phones that are running SIP.</td>
</tr>
<tr>
<td>PRIChannelsActive</td>
<td>This counter represents the number of PRI voice channels that are in an active call on a Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>PRISpansInService</td>
<td>This counter represents the number of PRI spans that are currently available for use.</td>
</tr>
<tr>
<td>RegisteredAnalogAccess</td>
<td>This counter represents the number of registered Cisco analog access gateways that are registered with system. The count does not include the number of Cisco analog access ports.</td>
</tr>
<tr>
<td>RegisteredHardwarePhones</td>
<td>This counter represents the number of Cisco hardware IP phones (for example, Cisco Unified IP Phones 7960, 7940, 7910, and so on.) that are currently registered in the system.</td>
</tr>
<tr>
<td>RegisteredMGCPGateway</td>
<td>This counter represents the number of MGCP gateways that are currently registered in the system.</td>
</tr>
<tr>
<td>RegisteredOtherStationDevices</td>
<td>This counter represents the number of station devices other than Cisco hardware IP phones that are currently registered in the system (for example, Cisco IP SoftPhone, CTI port, CTI route point, Cisco voice-mail port).</td>
</tr>
<tr>
<td>SIPLineServerAuthorizationChallenges</td>
<td>This counter represents the number of authentication challenges for incoming SIP requests that the Cisco Unified Communications Manager server issued to phones that are running SIP. An authentication challenge occurs when a phone that is running SIP with Digest Authentication enabled sends a SIP line request to Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>SIPLineServerAuthorizationFailures</td>
<td>This counter represents the number of authentication challenge failures for incoming SIP requests from SIP phones to the Cisco Unified Communications Manager server. An authentication failure occurs when a SIP phone with Digest Authentication enabled sends a SIP line request with bad credentials to Cisco Unified Communications Manager.</td>
</tr>
</tbody>
</table>
## Table 5-5  Cisco CallManager (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIPTrunkAuthorization</td>
<td>This counter represents the number of application-level authorization checks for incoming SIP requests that Cisco Unified Communications Manager has issued to SIP trunks. An application-level authorization check occurs when Cisco Unified Communications Manager compares an incoming SIP request to the application-level settings on the SIP Trunk Security Profile Configuration window in Cisco Unified Communications Manager Administration.</td>
</tr>
<tr>
<td>SIPTrunkAuthorizationFailures</td>
<td>This counter represents the number of application-level authorization failures for incoming SIP requests that have occurred on Cisco Unified Communications Manager SIP trunks. An application-level authorization failure occurs when Cisco Unified Communications Manager compares an incoming SIP request to the application-level authorization settings on the SIP Trunk Security Profile Configuration window in Cisco Unified Communications Manager Administration and finds that authorization for one or more of the SIP features on that window is not allowed.</td>
</tr>
<tr>
<td>SIPTrunkServerAuthenticationChallenges</td>
<td>This counter represents the number of authentication challenges for incoming SIP requests that Cisco Unified Communications Manager issued to SIP trunks. An authentication challenge occurs when a SIP trunk with Digest Authentication enabled sends a SIP request to Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>SIPTrunkServerAuthenticationFailures</td>
<td>This counter represents the number of authentication challenge failures that occurred for incoming SIP requests from SIP trunks to Cisco Unified Communications Manager. An authentication failure occurs when a SIP trunk with Digest Authentication enabled sends a SIP request with bad credentials to Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>SWConferenceActive</td>
<td>This counter represents the number of active conferences on all software conference devices that are registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>SWConferenceCompleted</td>
<td>This counter represents the total number of conferences that used a software conference bridge that was allocated from a Cisco Unified Communications Manager and that have been completed, which means that the conference bridge has been allocated and released. A conference activates when the first call connects to the bridge. The conference completes when the last call disconnects from the bridge.</td>
</tr>
<tr>
<td>SWConferenceOutOfResources</td>
<td>This counter represents the total number of times that Cisco Unified Communications Manager attempted to allocate a software conference resource from those that are registered to Cisco Unified Communications Manager when none was available. Counter includes failed attempts to add a new participant to an existing conference.</td>
</tr>
<tr>
<td>SWConferenceResourceActive</td>
<td>This counter represents the total number of conference resources that are in use on all software conference devices that are registered with Cisco Unified Communications Manager. The system considers a conference to be active when one or more calls connect to a bridge. One resource equals one stream.</td>
</tr>
<tr>
<td>SWConferenceResourceAvailable</td>
<td>This counter represents the number of new software-based conferences that can be started at the same time, for Cisco Unified Communications Manager. You must have a minimum of three streams available for each new conference. One resource equals one stream.</td>
</tr>
</tbody>
</table>
### Table 5-5  Cisco CallManager (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SWConferenceResourceTotal</td>
<td>This counter represents the total number of software conference resources that are provided by all software conference bridge devices that are currently registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>SystemCallsAttempted</td>
<td>This counter represents the total number of server-originated calls and attempted calls to the Unity message waiting indicator (MWI).</td>
</tr>
<tr>
<td>T1ChannelsActive</td>
<td>This counter represents the number of T1 CAS voice channels that are in an active call on a Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>T1SpansInService</td>
<td>This counter represents the number of T1 CAS spans that are currently available for use.</td>
</tr>
<tr>
<td>TLSConnectedSIPTrunks</td>
<td>This counter represents the number of SIP trunks that are configured and connected via Transport Layer Security (TLS).</td>
</tr>
<tr>
<td>TLSConnectedWSM</td>
<td>This counter represents the number of WSM Connectors that are configured and connected to Motorola WSM via Transport Layer Security (TLS).</td>
</tr>
<tr>
<td>TranscoderOutOfResources</td>
<td>This counter represents the total number of times that Cisco Unified Communications Manager attempted to allocate a transcoder resource from a transcoder device that is registered to a Cisco Unified Communications Manager when none was available.</td>
</tr>
<tr>
<td>TranscoderResourceActive</td>
<td>This counter represents the total number of transcoders that are in use on all transcoder devices that are registered with Cisco Unified Communications Manager. A transcoder in use represents one transcoder resource that has been allocated for use in a call. Each transcoder resource uses two streams.</td>
</tr>
<tr>
<td>TranscoderResourceAvailable</td>
<td>This counter represents the total number of transcoders that are not in use and that are available to be allocated on all transcoder devices that are registered with Cisco Unified Communications Manager. Each transcoder resource uses two streams.</td>
</tr>
<tr>
<td>TranscoderResourceTotal</td>
<td>This counter represents the total number of transcoder resources that are provided by all transcoder devices that are currently registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>VCBCConferenceActive</td>
<td>This counter represents the total number of active video conferences on all video conference bridge devices that are registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>VCBCConferenceAvailable</td>
<td>This counter represents the total number of new video conferences on all video conference bridge devices that are registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>VCBCConferenceCompleted</td>
<td>This counter represents the total number of video conferences that used a video conference bridge that are allocated from Cisco Unified Communications Manager and that have been completed, which means that the conference bridge has been allocated and released. A conference activates when the first call connects to the bridge. The conference completes when the last call disconnects from the bridge.</td>
</tr>
<tr>
<td>VCBCConferenceTotal</td>
<td>This counter represents the total number of video conferences that are supported on all video conference bridge devices that are registered with Cisco Unified Communications Manager.</td>
</tr>
</tbody>
</table>
Cisco CallManager External Call Control

The Cisco CallManager External Call Control feature provides information about the counters that are added to support the External Call Control feature. Table 5-6 contains information about the External Call Control counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VCBOutOfConferences</td>
<td>This counter represents the total number of times that Cisco Unified Communications Manager attempted to allocate a video conference resource from those that are registered to Cisco Unified Communications Manager when none was available.</td>
</tr>
<tr>
<td>VCBOutOfResources</td>
<td>This counter represents the total number of failed new video conference requests. A conference request can fail because, for example, the configured number of conferences is already in use.</td>
</tr>
<tr>
<td>VCBResourceActive</td>
<td>This counter represents the total number of video conference resources that are currently in use on all video conference devices that are registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>VCBResourceAvailable</td>
<td>This counter represents the total number of video conference resources that are not active and are currently available.</td>
</tr>
<tr>
<td>VCBResourceTotal</td>
<td>This counter represents the total number of video conference resources that are provided by all video conference bridge devices that are currently registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>VideoCallsActive</td>
<td>This counter represents the number of active video calls with active video streaming connections on all video conference bridge devices that are registered with Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>VideoCallsCompleted</td>
<td>This counter represents the number of video calls that were actually connected with video streams and then released.</td>
</tr>
<tr>
<td>VideoOutOfResources</td>
<td>This counter represents the total number of times that Cisco Unified Communications Manager attempted to allocate a video-streaming resource from one of the video conference bridge devices that is registered to Cisco Unified Communications Manager when none was available.</td>
</tr>
<tr>
<td>XCODE_RequestsThrottled</td>
<td>This counter represents the total number of transcoder resource requests that have been denied due to throttling (a resource from this transcoder was not allocated because, as specified by the Cisco CallManager service parameter MTP and Transcoder Resource Throttling Percentage, the transcoder was being utilized beyond the configured throttle percentage). This counter increments each time a resource is requested from this transcoder and is denied due to throttling. This counter reflects a running total since the transcoder device registered with the Cisco CallManager service.</td>
</tr>
</tbody>
</table>
Table 5-6  Cisco CallManager External Call Control

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Unified Communication Manager (Cisco CallManager) Object</td>
<td></td>
</tr>
<tr>
<td>ExternalCallControlEnabledCallsAttempted</td>
<td>This counter specifies the total number of calls to devices that have the External Call Control feature enabled. This is a cumulative count of all calls to intercept-enabled patterns or DNs since the last restart of the Cisco CallManager service.</td>
</tr>
<tr>
<td>ExternalCallControlEnabledCallsCompleted</td>
<td>This counter specifies the total number of calls that were connected to a device that had the External Call Control feature enabled. This is a cumulative count of all calls to intercept-enabled patterns or DNs since the last restart of the Cisco CallManager service.</td>
</tr>
<tr>
<td>ExternalCallControlEnabledFailureTreatmentApplied</td>
<td>This counter specifies the total number of calls that were cleared or routed based on failure treatments (such as Allow or Deny) that are defined in the External Call Control profile.</td>
</tr>
<tr>
<td><strong>External Call Control Objects</strong></td>
<td></td>
</tr>
<tr>
<td>PDP Servers Total</td>
<td>This counter defines the total number of PDP servers in all External Call Control Profiles configured in Cisco Unified CM Administration. This counter increments when a new PDP server is added and decrements when a PDP server is removed.</td>
</tr>
<tr>
<td>PDP Servers In Service</td>
<td>This counter defines the total number of in-service (active) PDP servers.</td>
</tr>
<tr>
<td>PDP Servers Out Of Service</td>
<td>This counter defines the total number of times that PDP servers have transitioned from in-service to out-of-service. This is a cumulative count of out-of-service PDP servers since the last restart of the Cisco CallManager service.</td>
</tr>
<tr>
<td>Connections Active To PDP Server</td>
<td>This counter specifies the total number of connections that Cisco Unified Communications Manager has established (currently active) with PDP servers.</td>
</tr>
<tr>
<td>Connections Lost To PDP Server</td>
<td>This counter specifies the total number of times that active connections between Cisco Unified Communications Manager and the PDP servers were disconnected. This is a cumulative count since the last restart of the Cisco CallManager service.</td>
</tr>
</tbody>
</table>

Cisco CallManager SAF

The Cisco SAF Client object provides information about SAF counters that are specific to each node. Table 5-7 contains information about Cisco SAF Client object counters.

Table 5-7  Cisco CallManager SAF Client Object

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAF Connections Succeeded (range from 0 to 2)</td>
<td>Total number of SAF client connections currently active on this Unified CM node.</td>
</tr>
<tr>
<td>SAF Connections Failed (range from 0 to 2)</td>
<td>Total number of SAF client connections that failed on the Unified CM node. A failed connection is a connection that did not register with the SAF Forwarder.</td>
</tr>
</tbody>
</table>

Note: A Cisco Unified CM node restart causes a counter reset.

See Real-Time Monitoring Tool Guide for more information.
# Cisco CallManager System Performance

The Cisco CallManager System Performance object provides system performance information about Cisco Unified Communications Manager. Table 5-8 contains information about Cisco CallManager system performance counters.

## Table 5-8  Cisco CallManager System Performance

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AverageExpectedDelay</td>
<td>This counter represents the current average expected delay before any incoming message gets handled.</td>
</tr>
<tr>
<td>CallsRejectedDueToICTThrottling</td>
<td>This counter represents the total number of calls that were rejected since the start of Cisco CallManager service due to Intercluster Trunk (ICT) call throttling. When the threshold limit of 140 calls per 5 seconds is met, the ICT will start throttling (rejecting) new calls. One cause for ICT call throttling occurs when calls across an ICT enter a route loop condition.</td>
</tr>
<tr>
<td>CallThrottlingGenericCounter3</td>
<td>This counter represents a generic counter that is used for call-throttling purpose.</td>
</tr>
<tr>
<td>CodeRedEntryExit</td>
<td>This counter indicates whether Cisco Unified Communications Manager has entered or exited a Code state (call-throttling mode). Valid values include 0 (Exit) and 1 (Entry).</td>
</tr>
<tr>
<td>CodeYellowEntryExit</td>
<td>This counter indicates whether Cisco Unified Communications Manager has entered or exited a Code Yellow state (call-throttling mode). Valid values include 0 (Exit) and 1 (Entry).</td>
</tr>
<tr>
<td>EngineeringCounter1</td>
<td>Do not use this counter unless directed by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>EngineeringCounter2</td>
<td>Do not use this counter unless directed by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>EngineeringCounter3</td>
<td>Do not use this counter unless directed by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>EngineeringCounter4</td>
<td>Do not use this counter unless directed by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>EngineeringCounter5</td>
<td>Do not use this counter unless directed by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>EngineeringCounter6</td>
<td>Do not use this counter unless directed by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>EngineeringCounter7</td>
<td>Do not use this counter unless directed by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>EngineeringCounter8</td>
<td>Do not use this counter unless directed by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>QueueSignalsPresent 1-High</td>
<td>This counter indicates the number of high-priority signals in the Cisco Unified Communications Manager queue. High-priority signals include timeout events, internal Cisco Unified Communications Manager keepalives, certain gatekeeper events, and internal process creation, among other events. A large number of high-priority events will cause degraded performance on Cisco Unified Communications Manager and result in slow call connection or loss of dial tone. Use this counter in conjunction with the QueueSignalsProcessed 1-High counter to determine the processing delay on Cisco Unified Communications Manager.</td>
</tr>
</tbody>
</table>
### Performance Monitoring in RTMT

Queue Signals Present 2-Normal

This counter indicates the number of normal-priority signals in the Cisco Unified Communications Manager queue. Normal-priority signals include call-processing functions, key presses, on-hook and off-hook notifications, among other events. A large number of normal-priority events will cause degraded performance on Cisco Unified Communications Manager, sometimes resulting in delayed dial tone, slow call connection, or loss of dial tone. Use this counter in conjunction with the Queue Signals Processed 2-Normal counter to determine the call-processing delay on Cisco Unified Communications Manager. Remember that high-priority signals must complete before normal-priority signals begin to process, so check the high-priority counters as well to get an accurate picture of the potential delay.

Queue Signals Present 3-Low

This counter indicates the number of low-priority signals in the Cisco Unified Communications Manager queue. Low-priority signals include station device registration (except the initial station registration request message), among other events. A large number of signals in this queue could result in delayed device registration, among other events.

Queue Signals Present 4-Lowest

This counter indicates the number of lowest priority signals in the Cisco Unified Communications Manager queue. Lowest priority signals include the initial station registration request message during device registration, among other events. A large number of signals in this queue could result in delayed device registration, among other events.

Queue Signals Processed 1-High

This counter indicates the number of high-priority signals that Cisco Unified Communications Manager processes for each 1-second interval. Use this counter in conjunction with the Queue Signals Present 1-High counter to determine the processing delay on this queue.

Queue Signals Processed 2-Normal

This counter indicates the number of normal-priority signals that Cisco Unified Communications Manager processes for each 1-second interval. Use this counter in conjunction with the Queue Signals Present 2-Normal counter to determine the processing delay on this queue. Remember that high-priority signals get processed before normal-priority signals.

Queue Signals Processed 3-Low

This counter indicates the number of low-priority signals that Cisco Unified Communications Manager processes for each 1-second interval. Use this counter in conjunction with the Queue Signals Present 3-Low counter to determine the processing delay on this queue. The number of signals processed gives an indication of how much device registration activity is being processed in this time interval.

Queue Signals Processed 4-Lowest

This counter indicates the number of lowest priority signals that Cisco Unified Communications Manager processes for each 1-second interval. Use this counter in conjunction with the Queue Signals Present 4-Lowest counter to determine the processing delay on this queue. The number of signals that are processed gives an indication of how many devices began the Cisco Unified Communications Manager registration process in this time interval.

Queue Signals Processed Total

This counter provides a sum total of all queue signals that Cisco Unified Communications Manager processes for each 1-second interval for all queue levels: high, normal, low, and lowest.

### Table 5-8 Cisco CallManager System Performance (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QueueSignalsPresent 2-Normal</td>
<td>This counter indicates the number of normal-priority signals in the Cisco Unified Communications Manager queue. Normal-priority signals include call-processing functions, key presses, on-hook and off-hook notifications, among other events. A large number of normal-priority events will cause degraded performance on Cisco Unified Communications Manager, sometimes resulting in delayed dial tone, slow call connection, or loss of dial tone. Use this counter in conjunction with the Queue Signals Processed 2-Normal counter to determine the call-processing delay on Cisco Unified Communications Manager. Remember that high-priority signals must complete before normal-priority signals begin to process, so check the high-priority counters as well to get an accurate picture of the potential delay.</td>
</tr>
<tr>
<td>QueueSignalsPresent 3-Low</td>
<td>This counter indicates the number of low-priority signals in the Cisco Unified Communications Manager queue. Low-priority signals include station device registration (except the initial station registration request message), among other events. A large number of signals in this queue could result in delayed device registration, among other events.</td>
</tr>
<tr>
<td>QueueSignalsPresent 4-Lowest</td>
<td>This counter indicates the number of lowest priority signals in the Cisco Unified Communications Manager queue. Lowest priority signals include the initial station registration request message during device registration, among other events. A large number of signals in this queue could result in delayed device registration, among other events.</td>
</tr>
<tr>
<td>QueueSignalsProcessed 1-High</td>
<td>This counter indicates the number of high-priority signals that Cisco Unified Communications Manager processes for each 1-second interval. Use this counter in conjunction with the Queue Signals Present 1-High counter to determine the processing delay on this queue.</td>
</tr>
<tr>
<td>QueueSignalsProcessed 2-Normal</td>
<td>This counter indicates the number of normal-priority signals that Cisco Unified Communications Manager processes for each 1-second interval. Use this counter in conjunction with the Queue Signals Present 2-Normal counter to determine the processing delay on this queue. Remember that high-priority signals get processed before normal-priority signals.</td>
</tr>
<tr>
<td>QueueSignalsProcessed 3-Low</td>
<td>This counter indicates the number of low-priority signals that Cisco Unified Communications Manager processes for each 1-second interval. Use this counter in conjunction with the Queue Signals Present 3-Low counter to determine the processing delay on this queue. The number of signals processed gives an indication of how much device registration activity is being processed in this time interval.</td>
</tr>
<tr>
<td>QueueSignalsProcessed 4-Lowest</td>
<td>This counter indicates the number of lowest priority signals that Cisco Unified Communications Manager processes for each 1-second interval. Use this counter in conjunction with the Queue Signals Present 4-Lowest counter to determine the processing delay on this queue. The number of signals that are processed gives an indication of how many devices began the Cisco Unified Communications Manager registration process in this time interval.</td>
</tr>
<tr>
<td>QueueSignalsProcessed Total</td>
<td>This counter provides a sum total of all queue signals that Cisco Unified Communications Manager processes for each 1-second interval for all queue levels: high, normal, low, and lowest.</td>
</tr>
</tbody>
</table>
Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

Performance Monitoring in RTMT

Table 5-8  Cisco CallManager System Performance (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkinnyDevicesThrottled</td>
<td>This counter represents the total number of Skinny devices that are being throttled. A Skinny device gets throttled (asked to shut down and reregister) when the total number of events that the Skinny device generated exceeds the configured maximum threshold value (default value specifies 2000 events) within a 5-second interval.</td>
</tr>
<tr>
<td>ThrottlingSampleActivity</td>
<td>This counter indicates how many samples, out of the configured sample size, have non-zero averageExpectedDelay values. This counter gets reset when any sample has an averageExpectedDelay value of zero. This process repeats for each batch of samples. A batch represents the configured sample size.</td>
</tr>
<tr>
<td>TotalCodeYellowEntry</td>
<td>This counter indicates the number of times that Cisco Unified Communications Manager call processing enters the code yellow state. This counter remains cumulative from the start of the Cisco Unified Communications Manager process.</td>
</tr>
</tbody>
</table>

Cisco CTIManager

The Cisco CTI Manager object provides information about Cisco CTI Manager. Table 5-9 contains information about Cisco CTIManager counters.

Table 5-9  Cisco CTI Manager

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CcmLinkActive</td>
<td>This counter represents the total number of active Cisco Unified Communications Manager links. CTI Manager maintains links to all active servers in a cluster, if applicable.</td>
</tr>
<tr>
<td>CTIConnectionActive</td>
<td>This counter represents the total number of CTI clients that are currently connected to the CTIManager. This counter increases by one when a new connection is established and decreases by one when a connection is released. The CTIManager service parameter MaxCTIConnections determines the maximum number of active connections.</td>
</tr>
<tr>
<td>DevicesOpen</td>
<td>This counter represents the total number of devices that are configured in Cisco Unified Communications Manager that CTI applications control and/or monitor. Devices include hardware IP phones, CTI ports, CTI route points, and so on.</td>
</tr>
<tr>
<td>LinesOpen</td>
<td>This counter represents the total number of lines that are configured in Cisco Unified Communications Manager that control and/or monitor CTI applications.</td>
</tr>
<tr>
<td>QbeVersion</td>
<td>This counter represents the version number of the Quick Buffer Encoding (QBE) interface that the CTIManager uses.</td>
</tr>
</tbody>
</table>

Cisco Dual-Mode Mobility

The Cisco Dual-Mode Mobility object provides information about the dual-mode mobility application on Cisco Unified Communications Manager. Table 5-10 contains information about Cisco Dual-Mode Mobility counters.
## Chapter 5  Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

### Performance Monitoring in RTMT

**Table 5-10  Cisco Dual-Mode Mobility**

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsAnchored</td>
<td>This counter represents the number of calls that are placed or received on dual-mode phones that are anchored in Cisco Unified Communications Manager. The counter increments when a call is received from or placed to a dual-mode phone. The counter increments twice if a dual-mode phone calls another dual-mode phone.</td>
</tr>
<tr>
<td>DMMSRegistered</td>
<td>This counter represents the number of Dual-mode Mobile Station (DMMS) subscribers that are registered in the wireless LAN (WLAN).</td>
</tr>
<tr>
<td>FollowMeAborted</td>
<td>This counter represents the number of failed follow-me operations.</td>
</tr>
<tr>
<td>FollowMeAttempted</td>
<td>This counter represents the number of follow-me operations that Cisco Unified Communications Manager attempted. The counter increments when a SIP 302 - Moved Temporarily message is received from the Wireless Service Manager (WSM) and Cisco Unified Communications Manager redirects the call to the DMMS in WLAN.</td>
</tr>
<tr>
<td>FollowMeCompleted</td>
<td>This counter represents the number of follow-me operations that were successfully completed. The counter increments when the DMMS in WLAN answers the call and the media (voice path) successfully gets established with the calling device.</td>
</tr>
<tr>
<td>FollowMeInProgress</td>
<td>This counter represents the number of follow-me operations that are currently in progress. The counter increments when a follow-me is attempted, and it decrements when the follow-me operation aborts or completes.</td>
</tr>
<tr>
<td>H1HandOutAttempted</td>
<td>This counter represents the number of H1 hand-out operations that dual-mode phones attempt. The counter increments when Cisco Unified Communications Manager processes a call to the H1 number from a DMMS.</td>
</tr>
<tr>
<td>H1HandOutCompleted</td>
<td>This counter represents the number of successfully completed H1 hand-out operations. The counter increments when the DMMS in WLAN successfully reestablishes a media (voice path).</td>
</tr>
<tr>
<td>H2HandOutCompleted</td>
<td>This counter represents the number of successfully completed H2 hand-out operations. The counter increments when the DMMS in WLAN successfully reestablishes a media (voice path).</td>
</tr>
<tr>
<td>H2HandOutsAttempted</td>
<td>This counter represents the number of H2 hand-out operations that dual-mode phones attempt. The counter increments when Cisco Unified Communications Manager receives a call to the H2 number from a DMMS.</td>
</tr>
<tr>
<td>HandInAborted</td>
<td>This counter represents the number of hand-in operations that failed.</td>
</tr>
<tr>
<td>HandInAttempted</td>
<td>This counter represents the number of hand-in operations that dual-mode phones attempt.</td>
</tr>
<tr>
<td>HandInCompleted</td>
<td>This counter represents the number of successfully completed hand-in operations. The counter increments when the DMMS in WLAN successfully reestablishes a media (voice path).</td>
</tr>
<tr>
<td>HandInInProgress</td>
<td>This counter represents the number of hand-in operations that are currently in progress. The counter increments when a hand-in is attempted, and the counter decrements when the hand-in is aborted or completed.</td>
</tr>
</tbody>
</table>
Cisco Extension Mobility

The Cisco Extension Mobility object provides information about the extension mobility application. Table 5-11 contains information about Cisco Extension Mobility counters.

Table 5-11 Cisco Extension Mobility Application

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequestsHandled</td>
<td>This counter represents the total number of HTTP requests that the extension mobility application handled since the last restart of the Cisco CallManager service. A typical login would constitute two HTTP requests: one to query the initial login state of the device and another to log in the user on a device. Similarly, a typical logout also results in two HTTP requests.</td>
</tr>
<tr>
<td>RequestsInProgress</td>
<td>This counter represents the number of HTTP requests that the extension mobility application currently is handling. A typical login would constitute two HTTP requests: one to query the initial login state of the device and another to log in the user on a device. Similarly, a typical logout also results in two HTTP requests.</td>
</tr>
<tr>
<td>RequestsThrottled</td>
<td>This counter represents the total number of Login/Logout Requests that failed due to throttling.</td>
</tr>
<tr>
<td>LoginsSuccessful</td>
<td>This counter represents the total number of successful login requests that were completed through Extension Mobility Service.</td>
</tr>
<tr>
<td>LogoutsSuccessful</td>
<td>This counter represents the total number of successful logout requests that were completed through Extension Mobility Service</td>
</tr>
<tr>
<td>Total Login/LogoutRequestsAttempted</td>
<td>This counter represents the total number of Login and Logout requests that were attempted through this Extension Mobility Service. This number includes both successful and unsuccessful attempts.</td>
</tr>
<tr>
<td>Total Number of EMCC Messages</td>
<td>This represents the total number of messages related to EMCC Requests that came from remote clusters.</td>
</tr>
<tr>
<td>Number of Remote Devices</td>
<td>This represents the total number of devices from other clusters that are currently using a EMCC Base Device (EMCC Logged in).</td>
</tr>
<tr>
<td>Number of Unknown Remote Users</td>
<td>This represents the total number of users who were not found in any of the remote cluster during inter-cluster extension mobility login.</td>
</tr>
<tr>
<td>Active Inter-cluster Sessions</td>
<td>This represents the total number of inter cluster Extension Mobility requests that are currently in progress.</td>
</tr>
<tr>
<td>Total Number of Remote Users</td>
<td>This represents the total number of users from other cluster who use a local device of this cluster and have logged into a remote cluster.</td>
</tr>
<tr>
<td>EMCC Check User Requests Handled</td>
<td>This represents the total number of EMCC check user requests that came from remote clusters.</td>
</tr>
</tbody>
</table>
Cisco Feature Control Policy

The Cisco Feature Control Policy feature provides information about the two new counters for TFTP. Table 5-12 contains information about Cisco Feature Control Policy feature counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BuildFeaturePolicyCount</td>
<td>Indicates the number of built FCP files</td>
</tr>
<tr>
<td>FeaturePolicyChangeNotifications</td>
<td>Indicates the number of sent FCP change notifications</td>
</tr>
</tbody>
</table>

Cisco Gatekeeper

The Cisco Gatekeeper object provides information about registered Cisco gatekeeper devices. Table 5-13 contains information about Cisco gatekeeper device counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACFsReceived</td>
<td>This counter represents the total number of RAS Admission Confirm messages that are received from the configured gatekeeper and its alternate gatekeepers.</td>
</tr>
<tr>
<td>ARQsAttempted</td>
<td>This counter represents the total number of RAS Admission Request messages that are attempted by using the configured gatekeeper and its alternate gatekeepers.</td>
</tr>
<tr>
<td>RasRetries</td>
<td>This counter represents the number of retries due to loss or delay of all RAS acknowledgement messages on the configured gatekeeper and its alternate gatekeepers.</td>
</tr>
<tr>
<td>VideoOutOfResources</td>
<td>This counter represents the total number of video-stream requests to the configured gatekeeper or its alternate gatekeepers that failed, most likely due to lack of bandwidth.</td>
</tr>
</tbody>
</table>

Cisco H.323

The Cisco H.323 object provides information about registered Cisco H.323 devices. Table 5-14 contains information about Cisco H.323 device counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsActive</td>
<td>This counter represents the number of streaming connections that are currently active (in use) on the configured H.323 device; in other words, the number of calls that actually have a voice path that is connected.</td>
</tr>
<tr>
<td>CallsAttempted</td>
<td>This counter represents the total number of calls that have been attempted on a device, including both successful and unsuccessful call attempts.</td>
</tr>
<tr>
<td>CallsCompleted</td>
<td>This counter represents the total number of successful calls that were made from a device.</td>
</tr>
<tr>
<td>CallsInProgress</td>
<td>This counter represents the number of calls that are currently in progress on a device.</td>
</tr>
</tbody>
</table>
Performance Monitoring in RTMT

Cisco Hunt Lists

The Cisco Hunt Lists object provides information about the hunt lists that are defined in Cisco Unified Communications Manager Administration. Table 5-15 contains information about Cisco hunt list counters.

### Table 5-15  Cisco Hunt Lists

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsAbandoned</td>
<td>This counter represents the number of abandoned calls that occurred through a hunt list. An abandoned call represents one in which a caller hangs up before the call is answered.</td>
</tr>
<tr>
<td>CallsActive</td>
<td>This counter represents the number of calls that are currently active (in use) that occurred through a hunt list. An active call represents one that gets distributed and answered, and to which a voice path connects.</td>
</tr>
<tr>
<td>CallsBusyAttempts</td>
<td>This counter represents the number of times that calls through a hunt list were attempted when all members of the line and/or route groups were busy.</td>
</tr>
<tr>
<td>CallsInProgress</td>
<td>This counter represents the number of calls that are currently in progress through a hunt list. A call in progress represents one that the call distributor is attempting to extend to a member of a line or route group and that has not yet been answered. Examples of a hunt list member include a line, a station device, a trunk device, or a port/channel of a trunk device.</td>
</tr>
<tr>
<td>CallsRingNoAnswer</td>
<td>This counter represents the total number of calls through a hunt list that rang but that called parties did not answer.</td>
</tr>
</tbody>
</table>
Chapter 5  Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

Table 5-15  Cisco Hunt Lists (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HuntListInService</td>
<td>This counter specifies whether the particular hunt list is currently in service. A value of 0 indicates that the hunt list is out of service; a value of 1 indicates that the hunt list is in service. Reasons that a hunt list could be out of service include the list is not running on a primary Cisco Unified Communications Manager based on its Cisco Unified Communications Manager Group or the hunt list has been disabled in Cisco Unified Communications Manager Administration.</td>
</tr>
<tr>
<td>MembersAvailable</td>
<td>This counter represents the total number of available or idle members of line and route groups that belong to an in-service hunt list. An available member currently handles a call and will accept a new call. An idle member does not handle any call and will accept a new call. A hunt list member can comprise a route group, line group, or a combination. A member of a line group represents a directory number of a line on an IP phone or a voice-mail port. A member of a route group represents a station gateway, a trunk gateway, or port/channel of a trunk gateway.</td>
</tr>
</tbody>
</table>

Cisco HW Conference Bridge Device

The Cisco HW Conference Bridge Device object provides information about registered Cisco hardware conference bridge devices. Table 5-16 contains information about Cisco hardware conference bridge device counters.

Table 5-16  Cisco HW Conference Bridge Device

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HWConferenceActive</td>
<td>This counter represents the number of conferences that are currently active (in use) on a HW conference bridge device. One resource represents one stream.</td>
</tr>
<tr>
<td>HWConferenceCompleted</td>
<td>This counter represents the total number of conferences that have been allocated and released on a HW conference device. A conference starts when the first call connects to the bridge. The conference completes when the last call disconnects from the bridge.</td>
</tr>
<tr>
<td>OutOfResources</td>
<td>This counter represents the total number of times that an attempt was made to allocate a conference resource from a HW conference device and failed, for example, because all resources were already in use.</td>
</tr>
<tr>
<td>ResourceActive</td>
<td>This counter represents the number of resources that are currently in use (active) for this HW conference device. One resource represents one stream.</td>
</tr>
<tr>
<td>ResourceAvailable</td>
<td>This counter represents the total number of resources that are not active and are still available to be used now for a HW conference device. One resource represents one stream.</td>
</tr>
<tr>
<td>ResourceTotal</td>
<td>This counter represents the total number of resources for a HW conference bridge device. This counter equals the sum of the counters ResourceAvailable and ResourceActive. One resource represents one stream.</td>
</tr>
</tbody>
</table>

Cisco IP Manager Assistant

The Cisco IP Manager Assistant (IPMA) Service object provides information about the Cisco Unified Communications Manager Assistant application. Table 5-17 contains information on Cisco IPMA counters.
Table 5-17  Cisco IP Manager Assistant Service

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AssistantsActive</td>
<td>This counter represents the number of assistant consoles that are currently active.</td>
</tr>
<tr>
<td>LinesOpen</td>
<td>An active assistant console exists when an assistant is logged in from the assistant</td>
</tr>
<tr>
<td></td>
<td>console desktop application.</td>
</tr>
<tr>
<td>ManagersActive</td>
<td>This counter represents the current number of managers that the Cisco IPMA is</td>
</tr>
<tr>
<td></td>
<td>servicing.</td>
</tr>
<tr>
<td>SessionsCurrent</td>
<td>This counter represents the total number of managers assistants that are currently</td>
</tr>
<tr>
<td></td>
<td>using the Cisco Unified Communications Manager Assistant application. Each manager</td>
</tr>
<tr>
<td></td>
<td>and each assistant constitute an active session; so, for one</td>
</tr>
<tr>
<td></td>
<td>manager/assistant pair, this counter would reflect two sessions.</td>
</tr>
</tbody>
</table>

Cisco Lines

The Cisco Lines object represents the number of Cisco lines (directory numbers) that can dial and connect to a device. Lines represent all directory numbers that terminate on an endpoint. The directory number that is assigned to it identifies the line. The Cisco Lines object does not include directory numbers that include wildcards such as a pattern for a Digital or Analog Access gateway.

The Active counter represents the state of the line, either active or not active. A zero indicates that the line is not in use. When the number is greater than zero, this indicates that the line is active, and the number represents the number of calls that are currently in progress on that line. If more than one call is active, this indicates that the call is on hold either because of being placed on hold specifically (user hold) or because of a network hold operation (for example, a transfer is in progress, and it is on transfer hold). This applies to all directory numbers that are assigned to any device.

Cisco Locations

The Cisco Location object provides information about locations that are defined in Cisco Unified Communications Manager. Table 5-18 contains information on Cisco location counters.

Table 5-18  Cisco Locations

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BandwidthAvailable</td>
<td>This counter represents the current bandwidth in a given location. A value of 0</td>
</tr>
<tr>
<td></td>
<td>indicates that no bandwidth is available.</td>
</tr>
<tr>
<td>BandwidthMaximum</td>
<td>This counter represents the maximum bandwidth that is available in a given location.</td>
</tr>
<tr>
<td></td>
<td>A value of 0 indicates that infinite bandwidth is available.</td>
</tr>
<tr>
<td>CallsInProgress</td>
<td>This counter represents the number of calls that are currently in progress on a</td>
</tr>
<tr>
<td></td>
<td>particular Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>OutOfResources</td>
<td>This counter represents the total number of times that a call on a particular Cisco</td>
</tr>
<tr>
<td></td>
<td>Unified Communications Manager through the location failed due to lack of bandwidth.</td>
</tr>
</tbody>
</table>
### Table 5-18  Cisco Locations (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RSVP AudioReservationErrorCounts</td>
<td>This counter represents the number of RSVP reservation errors in the audio stream.</td>
</tr>
<tr>
<td>RSVP MandatoryConnectionsInProgress</td>
<td>This counter represents the number of connections with mandatory RSVP that are in progress.</td>
</tr>
<tr>
<td>RSVP OptionalConnectionsInProgress</td>
<td>This counter represents the number of connections with optional RSVP that are in progress.</td>
</tr>
<tr>
<td>RSVP TotalCallsFailed</td>
<td>This counter represents the total number of failed calls due to a RSVP reservation failure.</td>
</tr>
<tr>
<td>RSVP VideoCallsFailed</td>
<td>This counter represents the number of video calls that failed due to a RSVP reservation failure.</td>
</tr>
<tr>
<td>RSVP VideoReservationErrorCounts</td>
<td>This counter represents the number of RSVP reservation errors in the video stream.</td>
</tr>
<tr>
<td>VideoBandwidthAvailable</td>
<td>This counter represents the bandwidth that is currently available for video in the location where the person who initiated the video conference resides. A value of 0 indicates that no bandwidth is available.</td>
</tr>
<tr>
<td>VideoBandwidthMaximum</td>
<td>This counter represents the maximum bandwidth that is available for video in the location where the person who initiated the video conference resides. A value of 0 indicates that no bandwidth is allocated for video.</td>
</tr>
<tr>
<td>VideoOutOfResources</td>
<td>This counter represents the total number of failed video-stream requests (most likely due to lack of bandwidth) in the location where the person who initiated the video conference resides.</td>
</tr>
</tbody>
</table>

### Cisco Media Streaming Application

The Cisco IP Voice Media Streaming Application object provides information about the registered MTPs, MOH servers, conference bridge servers, and annunciators. Table 5-19 contains information on Cisco IP Voice Media Streaming Application counters.

**Note**

One object exists for each Cisco Unified Communications Manager in the Cisco Unified Communications Manager group that is associated with the device pool that the annunciator device is configured to use.
### Table 5-19  Cisco Media Streaming Application

<table>
<thead>
<tr>
<th>Counter</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANNConnectionsLost</td>
<td>This counter represents the total number of times since the last restart of the Cisco IP Voice Media Streaming Application that a Cisco Unified Communications Manager connection was lost.</td>
</tr>
<tr>
<td>ANNConnectionState</td>
<td>For each Cisco Unified Communications Manager that is associated with an annunciator, this counter represents the current registration state to Cisco Unified Communications Manager; 0 indicates no registration to Cisco Unified Communications Manager; 1 indicates registration to the primary Cisco Unified Communications Manager; 2 indicates connection to the secondary Cisco Unified Communications Manager (connected to Cisco Unified Communications Manager but not registered until the primary Cisco Unified Communications Manager connection fails).</td>
</tr>
<tr>
<td>ANNConnectionsTotal</td>
<td>This counter represents the total number of annunciator instances that have been started since the Cisco IP Voice Media Streaming Application service started.</td>
</tr>
<tr>
<td>ANNIInstancesActive</td>
<td>This counter represents the number of actively playing (currently in use) announcements.</td>
</tr>
<tr>
<td>ANNSActive</td>
<td>This counter represents the total number of currently active simplex (one direction) streams for all connections. Each stream direction counts as one stream. One internal stream provides the audio input and another output stream to the endpoint device.</td>
</tr>
<tr>
<td>ANNSAvailable</td>
<td>This counter represents the remaining number of streams that are allocated for the annunciator device that are available for use. This counter starts as 2 multiplied by the number of configured connections (defined in the Cisco IP Voice Media Streaming App service parameter for the Annunciator, Call Count) and is reduced by one for each active stream that started.</td>
</tr>
<tr>
<td>ANNSTotal</td>
<td>This counter represents the total number of simplex (one direction) streams that connected to the annunciator device since the Cisco IP Voice Media Streaming Application service started.</td>
</tr>
<tr>
<td>CFBConferencesActive</td>
<td>This counter represents the number of active (currently in use) conferences.</td>
</tr>
<tr>
<td>CFBConferencesTotal</td>
<td>This counter represents the total number of conferences that started since the Cisco IP Voice Media Streaming Application service started.</td>
</tr>
<tr>
<td>CFBConnectionsLost</td>
<td>This counter represents the total number of times since the last restart of the Cisco IP Voice Media Streaming Application that a Cisco Unified Communications Manager connection was lost.</td>
</tr>
<tr>
<td>CFBConnectionState</td>
<td>For each Cisco Unified Communications Manager that is associated with a SW Conference Bridge, this counter represents the current registration state to Cisco Unified Communications Manager; 0 indicates no registration to Cisco Unified Communications Manager; 1 indicates registration to the primary Cisco Unified Communications Manager; 2 indicates connection to the secondary Cisco Unified Communications Manager (connected to Cisco Unified Communications Manager but not registered until the primary Cisco Unified Communications Manager connection fails).</td>
</tr>
<tr>
<td>CFBStreamsActive</td>
<td>This counter represents the total number of currently active simplex (one direction) streams for all conferences. Each stream direction counts as one stream. In a three-party conference, the number of active streams equals 6.</td>
</tr>
</tbody>
</table>
### Table 5-19  Cisco Media Streaming Application (continued)

<table>
<thead>
<tr>
<th>Counter</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFBStreamsAvailable</td>
<td>This counter represents the remaining number of streams that are allocated for the conference bridge that are available for use. This counter starts as 2 multiplied by the number of configured connections (defined in the Cisco IP Voice Media Streaming App service parameter for Conference Bridge, Call Count) and is reduced by one for each active stream that started.</td>
</tr>
<tr>
<td>CFBStreamsTotal</td>
<td>This counter represents the total number of simplex (one direction) streams that connected to the conference bridge since the Cisco IP Voice Media Streaming Application service started.</td>
</tr>
</tbody>
</table>
| MOHAudioSourcesActive    | This counter represents the number of active (currently in use) audio sources for this MOH server. Be aware that some of these audio sources may not be actively streaming audio data if no devices are listening. The exception exists for multicast audio sources, which will always be streaming audio.  
When an audio source is in use, even after the listener has disconnected, this counter will always have one input stream for each configured MOH codec. For unicast streams, the stream may exist in a suspended state where no audio data is received until a device connects to listen to the stream. Each MOH multicast resource uses one stream for each audio source and codec combination. For example, if the default audio source is configured for multicast, G.711 mu-law and wideband codecs, then two streams get used (default audio source + G.711 mu-law and default audio source + wideband). |
| MOHConnectionsLost       | This counter represents the total number of times since the last restart of the Cisco IP Voice Media Streaming Application that a Cisco Unified Communications Manager connection was lost.                                                                                                                                                                                                                           |
| MOHConnectionState       | For each Cisco Unified Communications Manager that is associated with an MOH, this counter represents the current registration state to Cisco Unified Communications Manager; 0 indicates no registration to Cisco Unified Communications Manager; 1 indicates registration to the primary Cisco Unified Communications Manager; 2 indicates connection to the secondary Cisco Unified Communications Manager (connected to Cisco Unified Communications Manager but not registered until the primary Cisco Unified Communications Manager connection fails). |
| MOHStreamsActive         | This counter represents the total number of active (currently in use) simplex (one direction) streams for all connections. One output stream exists for each device that is listening to a unicast audio source, and one input stream exists for each active audio source, multiplied by the number of MOH codecs.  
When an audio source has been used once, it will always have one input stream for each configured MOH codec. For unicast streams, the stream may exist in a suspended state where no audio data is received until a device connects to listen to the stream. Each MOH multicast resource uses one stream for each audio source and codec combination. For example, if the default audio source is configured for multicast, G.711 mu-law and wideband codecs, then two streams get used (default audio source + G.711 mu-law and default audio source + wideband). |
Table 5-19  Cisco Media Streaming Application (continued)

<table>
<thead>
<tr>
<th>Counter</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOHStreamsAvailable</td>
<td>This counter represents the remaining number of streams that are allocated for the MOH device that are available for use. This counter starts as 408 plus the number of configured half-duplex unicast connections and is reduced by 1 for each active stream that started. The counter gets reduced by 2 for each multicast audio source, multiplied by the number of MOH codecs that are configured. The counter gets reduced by 1 for each unicast audio source, multiplied by the number of MOH codecs that are configured.</td>
</tr>
<tr>
<td>MOHStreamsTotal</td>
<td>This counter represents the total number of simplex (one direction) streams that have connected to the MOH server since the Cisco IP Voice Media Streaming Application service started.</td>
</tr>
<tr>
<td>MTPConnectionsLost</td>
<td>This counter represents the total number of times since the last restart of the Cisco IP Voice Streaming Application that a Cisco Unified Communications Manager connection was lost.</td>
</tr>
<tr>
<td>MTPConnectionState</td>
<td>For each Cisco Unified Communications Manager that is associated with an MTP, this counter represents the current registration state to Cisco Unified Communications Manager; 0 indicates no registration to Cisco Unified Communications Manager; 1 indicates registration to the primary Cisco Unified Communications Manager; 2 indicates connection to the secondary Cisco Unified Communications Manager (connected to Cisco Unified Communications Manager but not registered until the primary Cisco Unified Communications Manager connection fails).</td>
</tr>
<tr>
<td>MTPConnectionsTotal</td>
<td>This counter represents the total number of MTP instances that have been started since the Cisco IP Voice Media Streaming Application service started.</td>
</tr>
<tr>
<td>MTPInstancesActive</td>
<td>This counter represents the number of active (currently in use) instances of MTP.</td>
</tr>
<tr>
<td>MTPStreamsActive</td>
<td>This counter represents the total number of currently active simplex (one direction) streams for all connections. Each stream direction counts as one stream.</td>
</tr>
<tr>
<td>MTPStreamsAvailable</td>
<td>This counter represents the remaining number of streams that are allocated for the MTP device that are available for use. This counter starts as 2 multiplied by the number of configured connections (defined in the Cisco IP Voice Media Streaming App service parameter for MTP, Call Count) and is reduced by one for each active stream that started.</td>
</tr>
<tr>
<td>MTPStreamsTotal</td>
<td>This counter represents the total number of simplex (one direction) streams that connected to the MTP device since the Cisco IP Voice Media Streaming Application service started.</td>
</tr>
</tbody>
</table>

Cisco Messaging Interface

The Cisco Messaging Interface object provides information about the Cisco Messaging Interface (CMI) service. Table 5-20 contains information on Cisco Messaging Interface (CMI) counters.
Table 5-20  Cisco Messaging Interface

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HeartBeat</td>
<td>This counter represents the heartbeat of the CMI service. This incremental count</td>
</tr>
<tr>
<td></td>
<td>indicates that the CMI service is up and running. If the count does not increase</td>
</tr>
<tr>
<td></td>
<td>(increment), this means that the CMI service is down.</td>
</tr>
<tr>
<td>SMDIMessageCountInbound</td>
<td>This counter represents the running count of inbound SMDI messages since the last</td>
</tr>
<tr>
<td></td>
<td>restart of the CMI service.</td>
</tr>
<tr>
<td>SMDIMessageCountInbound24Hour</td>
<td>This counter represents the rolling count of inbound SMDI messages in the last</td>
</tr>
<tr>
<td></td>
<td>24 hours.</td>
</tr>
<tr>
<td>SMDIMessageCountOutbound</td>
<td>This counter represents the running count of outbound SMDI messages since the last</td>
</tr>
<tr>
<td></td>
<td>restart of the CMI service.</td>
</tr>
<tr>
<td>SMDIMessageCountOutbound24Hour</td>
<td>This counter represents the rolling count of outbound SMDI messages in the last</td>
</tr>
<tr>
<td></td>
<td>24 hours.</td>
</tr>
<tr>
<td>StartTime</td>
<td>This counter represents the time in milliseconds when the CMI service started.</td>
</tr>
<tr>
<td></td>
<td>The real-time clock in the computer, which simply acts as a reference point that</td>
</tr>
<tr>
<td></td>
<td>indicates the current time and the time that has elapsed, in milliseconds, since the</td>
</tr>
<tr>
<td></td>
<td>service started, provides the basis for this time. The reference point specifies</td>
</tr>
<tr>
<td></td>
<td>midnight, January 1, 1970.</td>
</tr>
</tbody>
</table>

Cisco MGCP BRI Device

The Cisco Media Gateway Control Protocol (MGCP) Foreign Exchange Office (FXO) Device object provides information about registered Cisco MGCP BRI devices. Table 5-21 contains information on Cisco MGCP BRI device counters.

Table 5-21  Cisco MGCP BRI Device

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsCompleted</td>
<td>This counter represents the total number of successful calls that were made from</td>
</tr>
<tr>
<td></td>
<td>this MGCP Basic Rate Interface (BRI) device</td>
</tr>
<tr>
<td>Channel 1 Status</td>
<td>This counter represents the status of the indicated B-Channel that is associated</td>
</tr>
<tr>
<td></td>
<td>with the MGCP BRI device. Possible values: 0 (Unknown) indicates the status of the</td>
</tr>
<tr>
<td></td>
<td>channel could not be determined; 1 (Out of service) indicates that this channel is</td>
</tr>
<tr>
<td></td>
<td>not available for use; 2 (Idle) indicates that this channel has no active call and</td>
</tr>
<tr>
<td></td>
<td>is ready for use; 3 (Busy) indicates an active call on this channel; 4 (Reserved)</td>
</tr>
<tr>
<td></td>
<td>indicates that this channel has been reserved for use as a D-channel or for use as</td>
</tr>
<tr>
<td></td>
<td>a Synch-Channel for BRI.</td>
</tr>
<tr>
<td>Channel 2 Status</td>
<td>This counter represents the status of the indicated B-Channel that is associated</td>
</tr>
<tr>
<td></td>
<td>with the MGCP BRI device. Possible values: 0 (Unknown) indicates the status of the</td>
</tr>
<tr>
<td></td>
<td>channel could not be determined; 1 (Out of service) indicates that this channel is</td>
</tr>
<tr>
<td></td>
<td>not available for use; 2 (Idle) indicates that this channel has no active call and</td>
</tr>
<tr>
<td></td>
<td>is ready for use; 3 (Busy) indicates an active call on this channel; 4 (Reserved)</td>
</tr>
<tr>
<td></td>
<td>indicates that this channel has been reserved for use as a D-channel or for use as</td>
</tr>
<tr>
<td></td>
<td>a Synch-Channel for BRI.</td>
</tr>
</tbody>
</table>
Table 5-21  Cisco MGCP BRI Device (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DatalinkInService</td>
<td>This counter represents the state of the Data Link (D-Channel) on the corresponding digital access gateway. This value will get set to 1 (one) if the Data Link is up (in service) or 0 (zero) if the Data Link is down (out of service).</td>
</tr>
<tr>
<td>OutboundBusyAttempts</td>
<td>This counter represents the total number of times that a call through this MGCP BRI device was attempted when no voice channels were available.</td>
</tr>
</tbody>
</table>

Cisco MGCP FXO Device

The Cisco Media Gateway Control Protocol (MGCP) Foreign Exchange Office (FXO) Device object provides information about registered Cisco MGCP FXO devices. Table 5-22 contains information on Cisco MGCP FXO device counters.

Table 5-22  Cisco MGCP FXO Device

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsCompleted</td>
<td>This counter represents the total number of successful calls that were made from the port on an MGCP FXO device.</td>
</tr>
<tr>
<td>OutboundBusyAttempts</td>
<td>This counter represents the total number of times that a call through the port on this MGCP FXO device was attempted when no voice channels were available.</td>
</tr>
<tr>
<td>PortStatus</td>
<td>This counter represents the status of the FXO port that is associated with this MGCP FXO device.</td>
</tr>
</tbody>
</table>

Cisco MGCP FXS Device

The Cisco MGCP Foreign Exchange Station (FXS) Device object provides information about registered Cisco MGCP FXS devices. One instance of this object gets created for each port on a Cisco Catalyst 6000 24 port FXS Analog Interface Module gateway. For example, a fully configured Catalyst 6000 Analog Interface Module would represent 24 separate instances of this object. Table 5-23 contains information on Cisco MGCP FXS device counters.

Table 5-23  Cisco MGCP FXS Device

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsCompleted</td>
<td>This counter represents the total number of successful calls that were made from this port on the MGCP FXS device.</td>
</tr>
<tr>
<td>OutboundBusyAttempts</td>
<td>This counter represents the total number of times that a call through this port on the MGCP FXS device was attempted when no voice channels were available.</td>
</tr>
<tr>
<td>PortStatus</td>
<td>This counter represents the status of the FXS port that is associated with a MGCP FXS device.</td>
</tr>
</tbody>
</table>
Cisco MGCP Gateways

The Cisco MGCP Gateways object provides information about registered MGCP gateways. Table 5-24 contains information on Cisco MGCP gateway counters.

### Table 5-24  Cisco MGCP Gateways

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BRIChannelsActive</td>
<td>This counter represents the number of BRI voice channels that are currently active in a call in the gateway.</td>
</tr>
<tr>
<td>BRISpansInService</td>
<td>This counter represents the number of BRI spans that are currently available for use in the gateway.</td>
</tr>
<tr>
<td>FXOPortsActive</td>
<td>This counter represents the number of FXO ports that are currently active in a call in the gateway.</td>
</tr>
<tr>
<td>FXOPortsInService</td>
<td>This counter represents the number of FXO ports that are currently available for use in the gateway.</td>
</tr>
<tr>
<td>FXSPortsActive</td>
<td>This counter represents the number of FXS ports that are currently active in a call in the gateway.</td>
</tr>
<tr>
<td>FXSPortsInService</td>
<td>This counter represents the number of FXS ports that are currently available for use in the gateway.</td>
</tr>
<tr>
<td>PRIChannelsActive</td>
<td>This counter represents the number of PRI voice channels that are currently active in a call in the gateway.</td>
</tr>
<tr>
<td>PRISpansInService</td>
<td>This counter represents the number of PRI spans that are currently available for use in the gateway.</td>
</tr>
<tr>
<td>T1ChannelsActive</td>
<td>This counter represents the number of T1 CAS voice channels that are currently active in a call in the gateway.</td>
</tr>
<tr>
<td>T1SpansInService</td>
<td>This counter represents the number of T1 CAS spans that are currently available for use in the gateway.</td>
</tr>
</tbody>
</table>

Cisco MGCP PRI Device

The Cisco MGCP Primary Rate Interface (PRI) Device object provides information about registered Cisco MGCP PRI devices. Table 5-25 contains information on Cisco MGCP PRI device counters.

### Table 5-25  Cisco MGCP PRI Device

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsActive</td>
<td>This counter represents the number of calls that are currently active (in use) on this MGCP PRI device.</td>
</tr>
<tr>
<td>CallsCompleted</td>
<td>This counter represents the total number of successful calls that were made from this MGCP PRI device.</td>
</tr>
</tbody>
</table>
Table 5-25  
Cisco MGCP PRI Device (continued)  

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channel 1 Status through Channel 15 Status (consecutively numbered)</td>
<td>This counter represents the status of the indicated B-Channel that is associated with a MGCP PRI device. Possible values: 0 (Unknown) indicates that the status of the channel could not be determined; 1 (Out of service) indicates that this channel is not available for use; 2 (Idle) indicates that this channel has no active call and is ready for use; 3 (Busy) indicates that an active call exists on this channel; 4 (Reserved) indicates that this channel has been reserved for use as a D-Channel or for use as a Synch-Channel for E-1.</td>
</tr>
<tr>
<td>Channel 16 Status</td>
<td>This counter represents the status of the indicated B-Channel that is associated with a MGCP PRI Device. Possible values: 0-Unknown, 1-Out of service, 2-Idle, 3-Busy, 4-Reserved, for an E1 PRI Interface, this channel is reserved for use as a D-Channel.</td>
</tr>
<tr>
<td>Channel 17 Status through Channel 31 Status (consecutively numbered)</td>
<td>This counter represents the status of the indicated B-Channel that is associated with the MGCP PRI Device. 0-Unknown, 1-Out of service, 2-Idle, 3-Busy, 4-Reserved.</td>
</tr>
<tr>
<td>DatalinkInService</td>
<td>This counter represents the state of the Data Link (D-Channel) on the corresponding digital access gateway. This value will get set to 1 (one) if the Data Link is up (in service) or 0 (zero) if the Data Link is down (out of service).</td>
</tr>
<tr>
<td>OutboundBusyAttempts</td>
<td>This counter represents the total number of times that a call through an MGCP PRI device was attempted when no voice channels were available.</td>
</tr>
</tbody>
</table>

Cisco MGCP T1 CAS Device  

The Cisco MGCP T1 Channel Associated Signaling (CAS) Device object provides information about registered Cisco MGCP T1 CAS devices. Table 5-26 contains information on Cisco MGCP T1 CAS device counters.

Table 5-26  
Cisco MGCP T1 CAS Device  

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsActive</td>
<td>This counter represents the number of calls that are currently active (in use) on this MGCP T1 CAS device.</td>
</tr>
<tr>
<td>CallsCompleted</td>
<td>This counter represents the total number of successful calls that were made from this MGCP T1 CAS device.</td>
</tr>
<tr>
<td>Channel 1 Status through Channel 24 Status (consecutively numbered)</td>
<td>This counter represents the status of the indicated B-Channel that is associated with an MGCP T1 CAS device. Possible values: 0 (Unknown) indicates the status of the channel could not be determined; 1 (Out of service) indicates that this channel is not available for use; 2 (Idle) indicates that this channel has no active call and is ready for use; 3 (Busy) indicates that an active call exists on this channel; 4 (Reserved) indicates that this channel has been reserved for use as a D-Channel or for use as a Synch-Channel for E-1.</td>
</tr>
<tr>
<td>OutboundBusyAttempts</td>
<td>This counter represents the total number of times that a call through the MGCP T1 CAS device was attempted when no voice channels were available.</td>
</tr>
</tbody>
</table>
Cisco Mobility Manager

The Cisco Mobility Manager object provides information on registered Cisco Unified Mobility Manager devices. Table 5-27 contains information on Cisco Unified Mobility Manager device counters.

Table 5-27 Cisco Mobility Manager

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MobileCallsAnchored</td>
<td>This counter represents the total number of paths that are associated with single-mode/dual-mode phone call that is currently anchored on a Cisco Unified Communications Manager. Call anchoring occurs when a call enters an enterprise gateway and connects to a mobility application that then uses redirection to send the call back out an enterprise gateway. For example, this counter increments twice for a dual-mode phone-to-dual-mode phone call: once for the originating call and once for the terminating call. When the call terminates, this counter decrements accordingly.</td>
</tr>
<tr>
<td>MobilityHandinsAborted</td>
<td>This counter represents the total number of aborted handins.</td>
</tr>
<tr>
<td>MobileHandinsCompleted</td>
<td>This counter represents the total number of handins that were completed by dual-mode phones. A completed handin occurs when the call successfully connects in the enterprise network and the phone moves from WAN to WLAN.</td>
</tr>
<tr>
<td>MobilityHandinsFailed</td>
<td>This counter represents the total number of handins (calls on mobile devices that move from cellular to the wireless network) that failed.</td>
</tr>
<tr>
<td>MobilityHandoutsAborted</td>
<td>This counter represents the total number of aborted handouts.</td>
</tr>
<tr>
<td>MobileHandoutsCompleted</td>
<td>This counter represents the total number of handouts (calls on mobile devices that move from the enterprise WLAN network to the cellular network) that were completed. A completed handout occurs when the call successfully connects.</td>
</tr>
<tr>
<td>MobileHandoutsFailed</td>
<td>This counter represents the total number of handouts (calls on mobile devices that move from cellular to the wireless network) that failed.</td>
</tr>
<tr>
<td>MobilityFollowMeCallsAttempted</td>
<td>This counter represents the total number of follow-me calls that were attempted.</td>
</tr>
<tr>
<td>MobilityFollowMeCallsIgnoredDueToAnswerTooSoon</td>
<td>This counter represents the total number of follow-me calls that were ignored before the AnswerTooSoon timer went off.</td>
</tr>
<tr>
<td>MobilityIVRCallsAttempted</td>
<td>This counter represents the total number of attempted IVR calls.</td>
</tr>
<tr>
<td>MobilityIVRCallsFailed</td>
<td>This counter represents the total number of failed IVR calls.</td>
</tr>
<tr>
<td>MobilityIVRCallsSucceeded</td>
<td>This counter represents the total number of successful IVR calls.</td>
</tr>
<tr>
<td>MobilitySCCPDualModeRegistered</td>
<td>This counter represents the total number of dual-mode SCCP devices that are registered.</td>
</tr>
<tr>
<td>MobilitySIPDualModeRegistered</td>
<td>This counter represents the total number of dual-mode SIP devices that are registered.</td>
</tr>
</tbody>
</table>
Cisco Music On Hold (MOH) Device

The Cisco Music On Hold (MOH) Device object provides information about registered Cisco MOH devices. Table 5-28 contains information on Cisco MOH device counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOHHighestActiveResources</td>
<td>This counter represents the largest number of simultaneously active MOH connections for an MOH server. This number includes both multicast and unicast connections.</td>
</tr>
<tr>
<td>MOHMulticastResourceActive</td>
<td>This counter represents the number of currently active multicast connections to multicast addresses that are served by an MOH server. Each MOH multicast resource uses one stream for each audio source and codec combination. For example, if the default audio source is configured for multicast, G.711 mu-law and wideband codecs, two streams get used (default audio source + G.711 mu-law and default audio source + wideband).</td>
</tr>
<tr>
<td>MOHMulticastResourceAvailable</td>
<td>This counter represents the number of multicast MOH connections to multicast addresses that are served by an MOH server that are not active and are still available to be used now for the MOH server. Each MOH multicast resource uses one stream for each audio source and codec combination. For example, if the default audio source is configured for multicast, G.711 mu-law and wideband codecs, two streams get used (default audio source + G.711 mu-law and default audio source + wideband).</td>
</tr>
<tr>
<td>MOHOutOfResources</td>
<td>This counter represents the total number of times that the Media Resource Manager attempted to allocate an MOH resource when all available resources on all MOH servers that are registered with a Cisco Unified Communications Manager were already active.</td>
</tr>
<tr>
<td>MOHTotalMulticastResources</td>
<td>This counter represents the total number of multicast MOH connections that are allowed to multicast addresses that are served by an MOH server. Each MOH multicast resource uses one stream for each audio source and codec combination. For example, if the default audio source is configured for multicast, G.711 mu-law and wideband codecs, two streams get used (default audio source + G.711 mu-law and default audio source + wideband).</td>
</tr>
<tr>
<td>MOHTotalUnicastResources</td>
<td>This counter represents the total number of unicast MOH connections that are allowed by an MOH server. Each MOH unicast resource uses one stream.</td>
</tr>
<tr>
<td>MOHUnicastResourceActive</td>
<td>This counter represents the number of active unicast MOH connections to an MOH server. Each MOH unicast resource uses one stream.</td>
</tr>
<tr>
<td>MOHUnicastResourceAvailable</td>
<td>This counter represents the number of unicast MOH connections that are not active and are still available to be used now for an MOH server. Each MOH unicast resource uses one stream.</td>
</tr>
</tbody>
</table>
Cisco MTP Device

The Cisco Media Termination Point (MTP) Device object provides information about registered Cisco MTP devices. Table 5-29 contains information on Cisco MTP device counters.

Table 5-29  Cisco MTP Device

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OutOfResources</td>
<td>This counter represents the total number of times that an attempt was made to allocate an MTP resource from an MTP device and failed; for example, because all resources were already in use.</td>
</tr>
<tr>
<td>ResourceActive</td>
<td>This counter represents the number of MTP resources that are currently in use (active) for an MTP device. Each MTP resource uses two streams. An MTP in use represents one MTP resource that has been allocated for use in a call.</td>
</tr>
<tr>
<td>ResourceAvailable</td>
<td>This counter represents the total number of MTP resources that are not active and are still available to be used now for an MTP device. Each MTP resource uses two streams. An MTP in use represents one MTP resource that has been allocated for use in a call.</td>
</tr>
<tr>
<td>ResourceTotal</td>
<td>This counter represents the total number of MTP resources that an MTP device provides. This counter equals the sum of the counters ResourceAvailable and ResourceActive.</td>
</tr>
</tbody>
</table>

Cisco Phones

The Cisco Phones object provides information about the number of registered Cisco Unified IP Phones, including both hardware-based and other station devices.

The CallsAttempted counter represents the number of calls that have been attempted from this phone. This number increases each time that the phone goes off hook and on hook.

Cisco Presence Feature

The Cisco Presence object provides information about presence subscriptions, such as statistics that are related to the speed dial or call list Busy Lamp Field (BLF) subscriptions. Table 5-30 contains information on Cisco Presence feature.

Table 5-30  Cisco Presence

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActiveCallListAndTrunkSubscriptions</td>
<td>This counter represents the active presence subscriptions for the call list feature as well as presence subscriptions through SIP trunk.</td>
</tr>
<tr>
<td>ActiveSubscriptions</td>
<td>This counter represents all active incoming and outgoing presence subscriptions.</td>
</tr>
<tr>
<td>CallListAndTrunkSubscriptionsThrottled</td>
<td>This counter represents the cumulative number of rejected call list and trunk side presence subscriptions due to throttling for the call list feature.</td>
</tr>
<tr>
<td>IncomingLineSideSubscriptions</td>
<td>This counter represents the cumulative number of presence subscriptions that were received on the line side.</td>
</tr>
</tbody>
</table>
Table 5-30  Cisco Presence

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IncomingTrunkSideSubscriptions</td>
<td>This counter represents the cumulative number of presence subscriptions that were received on the trunk side.</td>
</tr>
<tr>
<td>OutgoingTrunkSideSubscriptions</td>
<td>This counter represents the cumulative number of presence subscriptions that were sent on the trunk side.</td>
</tr>
</tbody>
</table>

Cisco QSIG Feature

The Cisco QSIG Feature object provides information regarding the operation of various QSIG features, such as call diversion and path replacement. Table 5-31 contains information on the Cisco QSIG feature counters.

Table 5-31  Cisco QSIG Feature

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallForwardByRerouteCompleted</td>
<td>This counter represents the number of successful calls that has been forwarded by rerouting. Call forward by rerouting enables the path for a forwarded call to be optimized (minimizes the number of B-Channels in use) from the originator perspective. This counter gets reset when the Cisco CallManager service parameter Call Forward by Reroute Enabled is enabled or disabled, or when the Cisco CallManager service restarts.</td>
</tr>
<tr>
<td>PathReplacementCompleted</td>
<td>This counter represents the number of successful path replacements that have occurred. Path replacement in a QSIG network optimizes the path between two edge PINX (PBXs) that are involved in a call. This counter resets when the Cisco CallManager service parameter Path Replacement Enabled is enabled or disabled, or when the Cisco CallManager service restarts.</td>
</tr>
</tbody>
</table>

Cisco Signaling Performance

The Cisco Signaling Performance object provides call-signaling data on transport communications on Cisco Unified Communications Manager. Table 5-32 contains information on the Cisco Signaling Performance counter.

Table 5-32  Cisco Signaling Performance

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDPAllotsThrottled</td>
<td>This counter represents the total number of incoming UDP packets that were throttled (dropped) because they exceeded the threshold for the number of incoming packets per second that is allowed from a single IP address. Configure the threshold via the SIP Station UDP Port Throttle Threshold and SIP Trunk UDP Port Throttle Threshold service parameters in Cisco Unified Communications Manager Administration. This counter increments for every throttledUDP packet that was received since the last restart of the Cisco CallManager Service.</td>
</tr>
</tbody>
</table>
Cisco SIP

The Cisco Session Initiation Protocol (SIP) object provides information about configured SIP devices. Table 5-33 contains information on the Cisco SIP counters.

### Table 5-33 Cisco SIP

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsActive</td>
<td>This counter represents the number of calls that are currently active (in use) on this SIP device.</td>
</tr>
<tr>
<td>CallsAttempted</td>
<td>This counter represents the number of calls that have been attempted on this SIP device, including both successful and unsuccessful call attempts.</td>
</tr>
<tr>
<td>CallsCompleted</td>
<td>This counter represents the number of calls that were actually connected (a voice path was established) from a SIP device. This number increases when the call terminates.</td>
</tr>
<tr>
<td>CallsInProgress</td>
<td>This counter represents the number of calls that are currently in progress on a SIP device, including all active calls. When all calls that are in progress are connected, the number of CallsInProgress equals the number of CallsActive.</td>
</tr>
<tr>
<td>VideoCallsActive</td>
<td>This counter represents the number of video calls with streaming video connections that are currently active (in use) on this SIP device.</td>
</tr>
<tr>
<td>VideoCallsCompleted</td>
<td>This counter represents the number of video calls that were actually connected with video streams for this SIP device. This number increments when the call terminates.</td>
</tr>
</tbody>
</table>

**Cisco SIP Normalization**

The Cisco SIP Normalization performance object contains counters that allow you to monitor aspects of the normalization script, including initialization errors, runtime errors, and script status. Each device that has an associated script causes a new instance of these counters to be created. Table 5-34 provides the Cisco SIP Normalization counters.

### Table 5-34 Cisco SIP Normalization

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeviceResetAutomatically</td>
<td>This counter indicates the number of times that Cisco Unified CM automatically resets the device (SIP trunk). The device reset is based on the values that are specified in the Script Execution Error Recovery Action and System Resource Error Recovery Action fields on the SIP Normalization Script Configuration window in Cisco Unified Communications Manager Administration. When the device (SIP trunk) is reset due to script errors, the counter value increments. This count restarts when the device is reset manually.</td>
</tr>
</tbody>
</table>
Chapter 5 Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

Performance Monitoring in RTMT

Table 5-34 Cisco SIP Normalization (continued)

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| DeviceResetManually | This counter indicates the number of times that the device (SIP trunk) is reset manually in Cisco Unified Communications Manager Administration or by other methods, such as AXL. When the device associated with a script is reset due to configuration changes, the counter value increments. The counter restarts when:   
  • The SIP trunk is deleted.   
  • The script on the trunk gets changed or deleted.   
  • Cisco Unified Communications Manager restarts. |
| ErrorExecution      | This counter represents the number of execution errors that occurred while the script executed. Execution errors can occur while a message handler executes. Execution errors can be caused by resource errors, an argument mismatch in a function call, and so on.   
  When an execution error occurs, Cisco Unified CM performs the following actions:   
  • Automatically restores the message to the original content before applying additional error handling actions.   
  • Increments the value of the counter.   
  • Takes appropriate action based on the configuration of the Script Execution Error Recovery Action and System Resource Error Recovery Action fields in Cisco Unified Communications Manager Administration.   
  Check the SIPNormalizationScriptError alarm for details, including the line number in the script that failed. Correct the script problem, upload the corrected script as needed, and reset the trunk. This counter increments every time an execution error occurs. This counter provides a count from the most recent trunk reset that involved a script configuration change. (A device reset alone does not restart the count; the script configuration must also change before the reset occurs.) If the counter continues to increment after you fix the script problem, examine the script again. |
| ErrorInit           | This counter represents the number of times a script error occurred after the script successfully loaded into memory, but failed to initialize in Cisco Unified CM. A script can fail to initialize due to resource errors, an argument mismatch in a function call, the expected table was not returned, and so on.   
  Check the SIPNormalizationScriptError alarm for details, including the line number in the script that failed. Correct the script problem, upload the corrected script as needed, and reset the trunk. This counter increments every time an initialization error occurs. This counter provides a count from the most recent trunk reset that was accompanied by a script configuration change. (A device reset alone does not restart the count; the script configuration must also change before the reset occurs.) If the counter continues to increment after you fix the script problem, examine the script again. When the error occurs during initialization, Cisco Unified CM automatically disables the script. |
| ErrorInternal       | This counter indicates the number of internal errors that occurred while the script executed. Internal errors are very rare. If the value in this counter is higher than zero, a defect exists in the system that is not related to the script content or execution. Collect SDI traces and contact the Technical Assistance Center (TAC). |
Performance Monitoring in RTMT

Chapter 5    Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

Table 5-34    Cisco SIP Normalization (continued)

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ErrorLoad</td>
<td>This counter represents the number of times a script error occurred when the script loaded into memory in Cisco Unified Communications Manager. A script can fail to load due to memory issues or syntax errors. Check the SIPNormalizationScriptError alarm for details. Check the script syntax for errors, upload the corrected script as needed, and reset the trunk. This counter increments every time a load error occurs. This counter provides a count from the most recent trunk reset that was accompanied by a script configuration change. (A device reset alone will not restart the count; the script configuration must also change before the reset occurs.) If the counter continues to increment even after you fix the script problem, examine the script again.</td>
</tr>
<tr>
<td>ErrorResource</td>
<td>This counter indicates whether the script encountered a resource error. Two kinds of resource errors exist: exceeding the value in the Memory Threshold field and exceeding the value in the Lua Instruction Threshold field. (Both fields display on the SIP Normalization Script Configuration window in Cisco Unified Communications Manager Administration.) If either condition occurs, Cisco Unified Communications Manager immediately closes the script and issues the SIPNormalizationScriptError alarm. If a resource error occurs while the script loads or initializes, the script is disabled. If a resource error occurs during execution, the configured system resource error recovery action is taken. (The setting of the System Resource Error Recovery Action field on the SIP Normalization Script Configuration window in Cisco Unified Communications Manager Administration defines this action.)</td>
</tr>
<tr>
<td>MemoryUsage</td>
<td>This counter specifies the amount of memory, in bytes, that the script consumes. This counter increases and decreases to match the amount of memory that the script uses. This count gets cleared when the script closes (because a closed script does not consume memory) and restarts when the script opens (gets enabled). A high number in this counter indicates a resource problem. Check the MemoryUsagePercentage counter and the SIPNormalizationResourceWarning alarm, which occur when the resource consumption exceeds an internally set threshold.</td>
</tr>
<tr>
<td>MemoryUsagePercentage</td>
<td>This counter specifies the percentage of the total amount of memory that the script consumes. The value in this counter is derived by dividing the value in the MemoryUsage counter by the value in the Memory Threshold field (in the SIP Normalization Script Configuration window) and multiplying the result by 100 to arrive at a percentage. This counter increases and decreases in accordance with the MemoryUsage counter. This count gets cleared when the script closes (because closed scripts do not consume memory) and restarts when the script opens (gets enabled). When this counter reaches the internally controlled resource threshold, the SIPNormalizationResourceWarning alarm is issued.</td>
</tr>
</tbody>
</table>
Table 5-34  Cisco SIP Normalization (continued)

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MessageRollback</td>
<td>This counter indicates the number of times that the system automatically rolled back a message. The system rolls back the message by using the error handling that is specified in the Script Execution Error Recovery Action field in the SIP Normalization Script Configuration window in Cisco Unified CM Administration. When an execution error occurs, Cisco Unified CM automatically restores the message to the original content before applying additional error handling actions. If error handling specifies rollback only, no further action is taken beyond rolling back to the original message before the normalization attempt. For the other possible Script Execution Error Recovery Actions, message rollback always occurs first, followed by the specified action, such as disabling the script, resetting the script automatically, or resetting the trunk automatically.</td>
</tr>
<tr>
<td>msgAddContentBody</td>
<td>This counter represents the number of times that the script added a content body to the message. If you are using the msg:addContentBody API in the script, this counter increases each time that the msg:addContentBody API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgAddHeader</td>
<td>This counter represents the number of times that the script added a SIP header to the message. If you are using the msg:addHeader API in the script, this counter increases each time that the msg:addHeader API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgAddHeaderUriParameter</td>
<td>This counter represents the number of times that the script added a SIP header URI parameter to a SIP header in the message. If you are using the msg:addHeaderUriParameter API in the script, this counter increases each time that the msg:addHeaderUriParameter API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgAddHeaderValueParameter</td>
<td>This counter represents the number of times that the script added a SIP header value parameter to a SIP header in the message. If you are using the msg:addHeaderValueParameter API in the script, this counter increases each time that the msg:addHeaderValueParameter API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgApplyNumberMask</td>
<td>This counter represents the number of times that the script applied a number mask to a SIP header in the message. If you are using the msg:applyNumberMask API in the script, this counter increases each time that the msg:applyNumberMask API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgBlock</td>
<td>This counter represents the number of times that the script blocked a message. If you are using the msg:block API in the script, this counter increases each time that the msg:block API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgConvertDiversionToHI</td>
<td>This counter represents the number of times that the script converted Diversion headers into History-Info headers in the message. If you are using the msg:convertDiversionToHI API in the script, this counter increases each time that the msg:convertDiversionToHI API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
</tbody>
</table>
### Table 5-34  Cisco SIP Normalization (continued)

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>msgConvertHIToDiversion</td>
<td>This counter represents the number of times that the script converted Diversion headers into History-Info headers in the message. If you are using the msg:convertDiversionToHI API in the script, this counter increases each time that the msg:convertDiversionToHI API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgModifyHeader</td>
<td>This counter represents the number of times that the script modified a SIP header in the message. If you are using the msg:modifyHeader API in the script, this counter increases each time that the msg:modifyHeader API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgRemoveContentBody</td>
<td>This counter represents the number of times that the script removed a content body from the message. If you are using the msg:removeContentBody API in the script, this counter increases each time that the msg:removeContentBody API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgRemoveHeader</td>
<td>This counter represents the number of times that the script removed a SIP header from the message. If you are using the msg:removeHeader API in the script, this counter increases each time that the msg:removeHeader API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgRemoveHeaderValue</td>
<td>This counter represents the number of times that the script removed a SIP header value from the message. If you are using the msg:removeHeaderValue API in the script, this counter increases each time that the msg:removeHeaderValue API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgSetRequestUri</td>
<td>This counter represents the number of times that the script modified the request URI in the message. If you are using the msg:setRequestUri API in the script, this counter increases each time that the msg:setRequestUri API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgSetResponseCode</td>
<td>This counter represents the number of times that the script modified the response code and/or response phrase in the message. If you are using the msg:setResponseCode API in the script, this counter increases each time that the msg:setResponseCode API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>msgSetSdp</td>
<td>This counter represents the number of times that the script set the SDP in the message. If you are using the msg:setSdp API in the script, this counter increases each time that the msg:setSdp API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>ptAddContentBody</td>
<td>This counter represents the number of times that the script added a content body to the PassThrough (pt) object. If you are using the pt:addContentBody API in the script, this counter increases each time that the pt:addContentBody API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>ptAddHeader</td>
<td>This counter represents the number of times that the script added a SIP header to the PassThrough (pt) object. If you are using the pt:addHeader API in the script, this counter increases each time that the pt:addHeader API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
</tbody>
</table>
Table 5-34  
Cisco SIP Normalization (continued)

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ptAddHeaderUriParameter</td>
<td>This counter represents the number of times that the script added a SIP header URI parameter to the PassThrough (pt) object. If you are using the pt:addHeaderUriParameter API in the script, this counter increases each time that the pt:addHeaderUriParameter API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>ptAddHeaderValueParameter</td>
<td>This counter represents the number of times that the script added a SIP header value parameter to the PassThrough (pt) object. If you are using the pt:addHeaderValueParameter API in the script, this counter increases each time that the pt:addHeaderValueParameter API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
<tr>
<td>ptAddRequestUriParameter</td>
<td>This counter represents the number of times that the script added a request URI parameter to the PassThrough (pt) object. If you are using the pt:addRequestUriParameter API in the script, this counter increases each time that the pt:addRequestUriParameter API executes successfully. If the counter behavior is not as expected, examine the script logic for errors.</td>
</tr>
</tbody>
</table>
| ScriptActive                  | This counter indicates whether the script is currently active (running on the trunk). The following values display for the counter:  
• 0—Indicates that the script is closed (disabled).  
• 1—Indicates that the script is open and operational.  
To open the script that should be running on this trunk, perform the following actions:  
1. Check for any alarms that might indicate why the script is not open.  
2. Correct any errors.  
3. Upload a new script if necessary.  
4. Reset the trunk. |
| ScriptClosed                  | This counter indicates the number of times that Cisco Unified Communications Manager has closed the script.  
When the script is closed, it is not enabled on this device.  
Cisco Unified CM closes the script under one of the following conditions:  
• The device was reset manually.  
• The device was reset automatically (due to an error).  
• The device was deleted.  
This count restarts when the SIP trunk is reset after a change to the script configuration and when Cisco Unified CM restarts. |
Performance Monitoring in RTMT

Table 5-34  Cisco SIP Normalization (continued)

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| ScriptDisabledAutomatically | This counter indicates the number of times that the system automatically disabled the script. The values that are specified in the Script Execution Error Recovery Action and System Resource Error Recovery Action fields in the SIP Normalization Script Configuration window in Cisco Unified CM Administration determine whether the script is disabled. The script also gets disabled as a result of script error conditions that are encountered during loading and initialization. This counter provides a count from the most recent manual device reset that involved a script configuration change (a device reset alone does not restart the count; the script must also have changed before the reset occurs). This counter increments every time Cisco Unified CM automatically disables a script due to script errors. If the number in this counter is higher than expected, perform the following actions:  
  • Check for SIPNormalizationScriptError alarm and SIPNormalizationAutoResetDisabled alarm.  
  • Check for any resource-related alarms and counters in RTMT to determine whether a resource issue is occurring.  
  • Check for any unexpected SIP normalization events in the SDI trace files. |
| ScriptOpened                | This counter indicates the number of times that the Cisco Unified CM attempted to open the script. For the script to open, it must load into memory in Cisco Unified CM, initialize, and be operational. A number greater than one in this counter means that Cisco Unified CM has made more than one attempt to open the script on this SIP trunk, either for an expected reason or due to an error during loading or initialization. The error can occur due to execution errors or resource errors or invalid syntax in the script. Expect this counter to be greater than one if any of these counters increment: DeviceResetManually, DeviceResetAutomatically, or ScriptResetAutomatically. The DeviceResetManually counter increments when an expected event, such as a maintenance window on the SIP trunk, causes the script to close. If the number in this counter is high for an unexpected reason, perform the following actions:  
  • Check for alarms, such as the SIPNormalizationScriptClosed, SIPNormalizationScriptError, or SIPNormalizationResourceWarning.  
  • Check resource-related alarms and counters in RTMT to determine whether a resource issue is occurring.  
  • Check for any unexpected SIP normalization events in the SDI trace files.  
  This count restarts when the SIP trunk resets after a script configuration change and when Cisco Unified CM restarts. |
Cisco SIP Stack

The Cisco SIP Stack object provides information about Session Initiation Protocol (SIP) stack statistics that are generated or used by SIP devices such as SIP Proxy, SIP Redirect Server, SIP Registrar, and SIP User Agent. Table 5-35 contains information on Cisco SIP Stack counters.

<table>
<thead>
<tr>
<th>Counter Name</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AckIns</td>
<td>This counter represents the total number of ACK requests that the SIP device received.</td>
</tr>
<tr>
<td>AckOuts</td>
<td>This counter represents the total number of ACK requests that the SIP device sent.</td>
</tr>
<tr>
<td>ByeIns</td>
<td>This counter represents the total number of BYE requests that the SIP device received. This number includes retransmission.</td>
</tr>
<tr>
<td>ByeOuts</td>
<td>This counter represents the total number of BYE requests that the SIP device sent. This number includes retransmission.</td>
</tr>
<tr>
<td>CancelIns</td>
<td>This counter represents the total number of CANCEL requests that the SIP device received. This number includes retransmission.</td>
</tr>
<tr>
<td>CancelOuts</td>
<td>This counter represents the total number of CANCEL requests that the SIP device sent. This number includes retransmission.</td>
</tr>
<tr>
<td>CCBsAllocated</td>
<td>This counter represents the number of Call Control Blocks (CCB) that are currently in use by the SIP stack. Each active SIP dialog uses one CCB.</td>
</tr>
<tr>
<td>GlobalFailedClassIns</td>
<td>This counter represents the total number of 6xx class SIP responses that the SIP device received. This number includes retransmission. This class of responses indicates that a SIP device, that is providing a client function, received a failure response message. Generally, the responses indicate that a server had definitive information on a particular called party and not just the particular instance in the Request-URI.</td>
</tr>
<tr>
<td>Counter</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>GlobalFailedClassOuts</td>
<td>This counter represents the total number of 6xx class SIP responses that the SIP device sent. This number includes retransmission. This class of responses indicates that a SIP device, that is providing a server function, received a failure response message. Generally, the responses indicate that a server had definitive information on a particular called party and not just the particular instance in the Request-URI.</td>
</tr>
<tr>
<td>InfoClassIns</td>
<td>This counter represents the total number of 1xx class SIP responses that the SIP device received. This includes retransmission. This class of responses provides information on the progress of a SIP request.</td>
</tr>
<tr>
<td>InfoClassOuts</td>
<td>This counter represents the total number of 1xx class SIP responses that the SIP device sent. This includes retransmission. This class of responses provides information on the progress of processing a SIP request.</td>
</tr>
<tr>
<td>InfoIns</td>
<td>This counter represents the total number of INFO requests that the SIP device has received. This number includes retransmission.</td>
</tr>
<tr>
<td>InfoOuts</td>
<td>This counter represents the total number of INFO requests that the SIP device has sent. This number includes retransmission.</td>
</tr>
<tr>
<td>InviteIns</td>
<td>This counter represents the total number of INVITE requests that the SIP device received. This number includes retransmission.</td>
</tr>
<tr>
<td>InviteOuts</td>
<td>This counter represents the total number of INVITE requests that the SIP device sent. This number includes retransmission.</td>
</tr>
<tr>
<td>NotifyIns</td>
<td>This counter represents the total number of NOTIFY requests that the SIP device received. This number includes retransmission.</td>
</tr>
<tr>
<td>NotifyOuts</td>
<td>This counter represents the total number of NOTIFY requests that the SIP device sent. This number includes retransmission.</td>
</tr>
<tr>
<td>OptionsIns</td>
<td>This counter represents the total number of OPTIONS requests that the SIP device received. This number includes retransmission.</td>
</tr>
<tr>
<td>OptionsOuts</td>
<td>This counter represents the total number of OPTIONS requests that the SIP device sent. This number includes retransmission.</td>
</tr>
<tr>
<td>PRAckIns</td>
<td>This counter represents the total number of PRACK requests that the SIP device received. This number includes retransmission.</td>
</tr>
<tr>
<td>PRAckOuts</td>
<td>This counter represents the total number of PRACK requests that the SIP device sent. This number includes retransmission.</td>
</tr>
<tr>
<td>PublishIns</td>
<td>This counter represents the total number of PUBLISH requests that the SIP device received. This number includes retransmissions.</td>
</tr>
<tr>
<td>PublishOuts</td>
<td>This counter represents the total number of PUBLISH requests that the SIP device sent. This number includes retransmission.</td>
</tr>
<tr>
<td>RedirClassIns</td>
<td>This counter represents the total number of 3xx class SIP responses that the SIP device received. This number includes retransmission. This class of responses provides information about redirections to addresses where the callee may be reachable.</td>
</tr>
</tbody>
</table>
### Counters

**RedirClassOuts**
This counter represents the total number of 3xx class SIP responses that the SIP device sent. This number includes retransmission. This class of responses provides information about redirections to addresses where the callee may be reachable.

**ReferIns**
This counter represents the total number of REFER requests that the SIP device received. This number includes retransmission.

**ReferOuts**
This counter represents the total number of REFER requests that the SIP device sent. This number includes retransmission.

**RegisterIns**
This counter represents the total number of REGISTER requests that the SIP device received. This number includes retransmission.

**RegisterOuts**
This counter represents the total number of REGISTER requests that the SIP device sent. This number includes retransmission.

**RequestsFailedClassIns**
This counter represents the total number of 4xx class SIP responses that the SIP device received. This number includes retransmission. This class of responses indicates a request failure by a SIP device that is providing a client function.

**RequestsFailedClassOuts**
This counter represents the total number of 4xx class SIP responses that the SIP device sent. This number includes retransmission. This class of responses indicates a request failure by a SIP device that is providing a server function.

**RetryByes**
This counter represents the total number of BYE retries that the SIP device sent. To determine the number of first BYE attempts, subtract the value of this counter from the value of the sipStatsByeOuts counter.

**RetryCancels**
This counter represents the total number of CANCEL retries that the SIP device sent. To determine the number of first CANCEL attempts, subtract the value of this counter from the value of the sipStatsCancelOuts counter.

**RetryInfo**
This counter represents the total number of INFO retries that the SIP device sent. To determine the number of first INFO attempts, subtract the value of this counter from the value of the sipStatsInfoOuts counter.

**RetryInvites**
This counter represents the total number of INVITE retries that the SIP device sent. To determine the number of first INVITE attempts, subtract the value of this counter from the value of the sipStatsInviteOuts counter.

**RetryNotify**
This counter represents the total number of NOTIFY retries that the SIP device sent. To determine the number of first NOTIFY attempts, subtract the value of this counter from the value of the sipStatsNotifyOuts counter.

**RetryPRAck**
This counter represents the total number of PRACK retries that the SIP device sent. To determine the number of first PRACK attempts, subtract the value of this counter from the value of the sipStatsPRAckOuts counter.

**RetryPublish**
This counter represents the total number of PUBLISH retries that the SIP device sent. To determine the number of first PUBLISH attempts, subtract the value of this counter from the value of the sipStatsPublishOuts counter.

**RetryRefer**
This counter represents the total number of REFER retries that the SIP device sent. To determine the number of first REFER attempts, subtract the value of this counter from the value of the sipStatsReferOuts counter.
### Table 5-35  Cisco SIP Stack (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RetryRegisters</td>
<td>This counter represents the total number of REGISTER retries that the SIP device sent. To determine the number of first REGISTER attempts, subtract the value of this counter from the value of the sipStatsRegisterOuts counter.</td>
</tr>
<tr>
<td>RetryRel1xx</td>
<td>This counter represents the total number of Reliable 1xx retries that the SIP device sent.</td>
</tr>
<tr>
<td>RetryRequestsOut</td>
<td>This counter represents the total number of Request retries that the SIP device sent.</td>
</tr>
<tr>
<td>RetryResponsesFinal</td>
<td>This counter represents the total number of Final Response retries that the SIP device sent.</td>
</tr>
<tr>
<td>RetryResponsesNonFinal</td>
<td>This counter represents the total number of non-Final Response retries that the SIP device sent.</td>
</tr>
<tr>
<td>RetrySubscribe</td>
<td>This counter represents the total number of SUBSCRIBE retries that the SIP device sent. To determine the number of first SUBSCRIBE attempts, subtract the value of this counter from the value of the sipStatsSubscribeOuts counter.</td>
</tr>
<tr>
<td>RetryUpdate</td>
<td>This counter represents the total number of UPDATE retries that the SIP device sent. To determine the number of first UPDATE attempts, subtract the value of this counter from the value of the sipStatsUpdateOuts counter.</td>
</tr>
<tr>
<td>SCBsAllocated</td>
<td>This counter represents the number of Subscription Control Blocks (SCB) that are currently in use by the SIP stack. Each subscription uses one SCB.</td>
</tr>
<tr>
<td>ServerFailedClassIns</td>
<td>This counter represents the total number of 5xx class SIP responses that the SIP device received. This number includes retransmission. This class of responses indicates that failure responses were received by a SIP device that is providing a client function.</td>
</tr>
<tr>
<td>ServerFailedClassOuts</td>
<td>This counter represents the total number of 5xx class SIP responses that the SIP device sent. This number includes retransmission. This class of responses indicates that failure responses were received by a SIP device that is providing a server function.</td>
</tr>
<tr>
<td>SIPGenericCounter1</td>
<td>Do not use this counter unless directed to do so by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>SIPGenericCounter2</td>
<td>Do not use this counter unless directed to do so by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>SIPGenericCounter3</td>
<td>Do not use this counter unless directed to do so by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>SIPGenericCounter4</td>
<td>Do not use this counter unless directed to do so by a Cisco Engineering Special build. Cisco uses information in this counter for diagnostic purposes.</td>
</tr>
<tr>
<td>SIPHandlerSDLQueueSignalsPresent</td>
<td>This counter represents the number of SDL signals that are currently on the four SDL priority queues of the SIPHandler component. The SIPHandler component contains the SIP stack.</td>
</tr>
</tbody>
</table>
Table 5-35  Cisco SIP Stack (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
</table>
| StatusCode1xxIns  | This counter represents the total number of 1xx response messages, including retransmission, that the SIP device received. This count includes the following 1xx responses:  
  • 100 Trying  
  • 180 Ringing  
  • 181 Call is being forwarded  
  • 182 Queued  
  • 183 Session Progress |
| StatusCode1xxOuts | This counter represents the total number of 1xx response messages, including retransmission, that the SIP device sent. This count includes the following 1xx responses:  
  • 100 Trying  
  • 180 Ringing  
  • 181 Call is being forwarded  
  • 182 Queued  
  • 183 Session Progress |
| StatusCode2xxIns  | This counter represents the total number of 2xx response messages, including retransmission, that the SIP device received. This count includes the following 2xx responses:  
  • 200 OK  
  • 202 Success Accepted |
| StatusCode2xxOuts | This counter represents the total number of 2xx response messages, including retransmission, that the SIP device sent. This count includes the following 2xx responses:  
  • 200 OK  
  • 202 Success Accepted |
| StatusCode3xxIns  | This counter represents the total number of 3xx response messages, including retransmission, that the SIP device received. This count includes the following 3xx responses:  
  • 300 Multiple Choices  
  • 301 Moved Permanently  
  • 302 Moved Temporarily  
  • 303 Incompatible Bandwidth Units  
  • 305 Use Proxy  
  • 380 Alternative Service |
| StatusCode302Outs | This counter represents the total number of 302 Moved Temporarily response messages, including retransmission, that the SIP device sent. |
Table 5-35  Cisco SIP Stack (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
</table>
| StatusCode4xxIns | This counter represents the total number of 4xx response messages, including retransmission, that the SIP device received. This count includes the following 4xx responses:  
|                | • 400 Bad Request                                                                    |
|                | • 401 Unauthorized                                                                    |
|                | • 402 Payment Required                                                                |
|                | • 403 Forbidden                                                                       |
|                | • 404 Not Found                                                                       |
|                | • 405 Method Not Allowed                                                              |
|                | • 406 Not Acceptable                                                                  |
|                | • 407 Proxy Authentication Required                                                  |
|                | • 408 Request Timeout                                                                 |
|                | • 409 Conflict                                                                        |
|                | • 410 Gone                                                                            |
|                | • 413 Request Entity Too Large                                                        |
|                | • 414 Request-URI Too Long                                                           |
|                | • 415 Unsupported Media Type                                                          |
|                | • 416 Unsupported URI Scheme                                                          |
|                | • 417 Unknown Resource Priority                                                       |
|                | • 420 Bad Extension                                                                   |
|                | • 422 Session Expires Value Too Small                                                  |
|                | • 423 Interval Too Brief                                                               |
|                | • 480 Temporarily Unavailable                                                        |
|                | • 481 Call/Transaction Does Not Exist                                                 |
|                | • 482 Loop Detected                                                                   |
|                | • 483 Too Many Hops                                                                   |
|                | • 484 Address Incomplete                                                              |
|                | • 485 Ambiguous                                                                       |
|                | • 486 Busy Here                                                                       |
|                | • 487 Request Terminated                                                              |
|                | • 488 Not Acceptable Here                                                             |
|                | • 489 Bad Subscription Event                                                          |
|                | • 491 Request Pending                                                                  |
Table 5-35  Cisco SIP Stack (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StatusCode4xxOuts</td>
<td>This counter represents the total number of 4xx response messages, including retransmission, that the SIP device sent. This count includes the following 4xx responses:</td>
</tr>
<tr>
<td></td>
<td>• 400 Bad Request</td>
</tr>
<tr>
<td></td>
<td>• 401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>• 402 Payment Required</td>
</tr>
<tr>
<td></td>
<td>• 403 Forbidden</td>
</tr>
<tr>
<td></td>
<td>• 404 Not Found</td>
</tr>
<tr>
<td></td>
<td>• 405 Method Not Allowed</td>
</tr>
<tr>
<td></td>
<td>• 406 Not Acceptable</td>
</tr>
<tr>
<td></td>
<td>• 407 Proxy Authentication Required</td>
</tr>
<tr>
<td></td>
<td>• 408 Request Timeout</td>
</tr>
<tr>
<td></td>
<td>• 409 Conflict</td>
</tr>
<tr>
<td></td>
<td>• 410 Gone</td>
</tr>
<tr>
<td></td>
<td>• 413 Request Entity Too Large</td>
</tr>
<tr>
<td></td>
<td>• 414 Request-URI Too Long</td>
</tr>
<tr>
<td></td>
<td>• 415 Unsupported Media Type</td>
</tr>
<tr>
<td></td>
<td>• 416 Unsupported URI Scheme</td>
</tr>
<tr>
<td></td>
<td>• 417 Unknown Resource Priority</td>
</tr>
<tr>
<td></td>
<td>• 420 Bad Extension</td>
</tr>
<tr>
<td></td>
<td>• 422 Session Expires Value Too Small</td>
</tr>
<tr>
<td></td>
<td>• 423 Interval Too Brief</td>
</tr>
<tr>
<td></td>
<td>• 480 Temporarily Unavailable</td>
</tr>
<tr>
<td></td>
<td>• 481 Call/Transaction Does Not Exist</td>
</tr>
<tr>
<td></td>
<td>• 482 Loop Detected</td>
</tr>
<tr>
<td></td>
<td>• 483 Too Many Hops</td>
</tr>
<tr>
<td></td>
<td>• 484 Address Incomplete</td>
</tr>
<tr>
<td></td>
<td>• 485 Ambiguous</td>
</tr>
<tr>
<td></td>
<td>• 486 Busy Here</td>
</tr>
<tr>
<td></td>
<td>• 487 Request Terminated</td>
</tr>
<tr>
<td></td>
<td>• 488 Not Acceptable Here</td>
</tr>
<tr>
<td></td>
<td>• 489 Bad Subscription Event</td>
</tr>
<tr>
<td></td>
<td>• 491 Request Pending</td>
</tr>
</tbody>
</table>
## Performance Monitoring in RTMT

### Table 5-35  Cisco SIP Stack (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
</table>
| StatusCode5xxIns | This counter represents the total number of 5xx response messages, including retransmission, that the SIP device received. This count includes the following 5xx responses:  
• 500 Server Internal Error  
• 501 Not Implemented  
• 502 Bad Gateway  
• 503 Service Unavailable  
• 504 Server Timeout  
• 505 Version Not Supported  
• 580 Precondition Failed |
| StatusCode5xxOuts | This counter represents the total number of 5xx response messages, including retransmission, that the SIP device sent. This count includes the following 5xx responses:  
• 500 Server Internal Error  
• 501 Not Implemented  
• 502 Bad Gateway  
• 503 Service Unavailable  
• 504 Server Timeout  
• 505 Version Not Supported  
• 580 Precondition Failed |
| StatusCode6xxIns | This counter represents the total number of 6xx response messages, including retransmission, that the SIP device received. This count includes the following 6xx responses:  
• 600 Busy Everywhere  
• 603 Decline  
• 604 Does Not Exist Anywhere  
• 606 Not Acceptable |
| StatusCode6xxOuts | This counter represents the total number of 6xx response messages, including retransmission, that the SIP device sent. This count includes the following 6xx responses:  
• 600 Busy Everywhere  
• 603 Decline  
• 604 Does Not Exist Anywhere  
• 606 Not Acceptable |
| SubscribeIns   | This counter represents the total number of SUBSCRIBE requests that the SIP device received. This number includes retransmission. |
| SubscribeOuts  | This counter represents the total number of SUBSCRIBE requests that the SIP device sent. This number includes retransmission. |
Cisco SIP Station

The Cisco SIP Station object provides information about SIP line-side devices. Table 5-36 contains information on the Cisco SIP Station counters.

Table 5-36 Cisco SIP Station

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConfigMismatchesPersistent</td>
<td>This counter represents the number of times that a phone that is running SIP was persistently unable to register due to a configuration version mismatch between the TFTP server and Cisco Unified Communications Manager since the last restart of the Cisco Unified Communications Manager. This counter increments each time that Cisco Unified Communications Manager cannot resolve the mismatch and manual intervention is required (such as a configuration update or device reset).</td>
</tr>
<tr>
<td>ConfigMismatchesTemporary</td>
<td>This counter represents the number of times that a phone that is running SIP was temporarily unable to register due to a configuration version mismatch between the TFTP server and Cisco Unified Communications Manager since the last restart of the Cisco CallManager service. This counter increments each time Cisco Unified Communications Manager can resolve the mismatch automatically.</td>
</tr>
<tr>
<td>DBTimeouts</td>
<td>This counter represents the number of new registrations that failed because a timeout occurred while the system was attempting to retrieve the device configuration from the database.</td>
</tr>
</tbody>
</table>
Chapter 5  Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

Table 5-36  Cisco SIP Station (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NewRegAccepted</td>
<td>This counter represents the total number of new REGISTRATION requests that have been removed from the NewRegistration queue and processed since the last restart of the Cisco CallManager service.</td>
</tr>
<tr>
<td>NewRegQueueSize</td>
<td>This counter represents the number of REGISTRATION requests that are currently on the NewRegistration queue. The system places REGISTRATION requests that are received from devices that are not currently registered on this queue before they are processed.</td>
</tr>
<tr>
<td>NewRegRejected</td>
<td>This counter represents the total number of new REGISTRATION requests that were rejected with a 486 Busy Here response and not placed on the NewRegistration queue since the last restart of the Cisco CallManager service. The system rejects REGISTRATION requests if the NewRegistration queue exceeds a programmed size.</td>
</tr>
<tr>
<td>TokensAccepted</td>
<td>This counter represents the total number of token requests that have been granted since the last Cisco Communications Manager restart. Cisco Unified Communications Manager grants tokens as long as the number of outstanding tokens remains below the number that is specified in the Cisco CallManager service parameter Maximum Phone Fallback Queue Depth.</td>
</tr>
<tr>
<td>TokensOutstanding</td>
<td>This counter represents the number of devices that have been granted a token but have not yet registered. The system requires that devices that are reconnecting to a higher priority Cisco Unified Communications Manager server be granted a token before registering. Tokens protect Cisco Unified Communications Manager from being overloaded with registration requests when it comes back online after a failover situation.</td>
</tr>
<tr>
<td>TokensRejected</td>
<td>This counter represents the total number of token requests that have been rejected since the last Cisco Unified Communications Manager restart. Cisco Unified Communications Manager will reject token request if the number of outstanding tokens is greater than the number that is specified in the Cisco CallManager service parameter Maximum Phone Fallback Queue Depth.</td>
</tr>
</tbody>
</table>

**Cisco SW Conf Bridge Device**

The Cisco SW Conference Bridge Device object provides information about registered Cisco software conference bridge devices. **Table 5-37** contains information on the Cisco software conference bridge device counters.

Table 5-37  Cisco SW Conf Bridge Device

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OutOfResources</td>
<td>This counter represents the total number of times that an attempt was made to allocate a conference resource from a SW conference device and failed because all resources were already in use.</td>
</tr>
<tr>
<td>ResourceActive</td>
<td>This counter represents the number of resources that are currently in use (active) for a SW conference device. One resource represents one stream.</td>
</tr>
<tr>
<td>ResourceAvailable</td>
<td>This counter represents the total number of resources that are not active and are still available to be used now for a SW conference device. One resource represents one stream.</td>
</tr>
</tbody>
</table>
Table 5-37  Cisco SW Conf Bridge Device (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourceTotal</td>
<td>This counter represents the total number of conference resources that a SW conference device provides. One resource represents one stream. This counter equals the sum of the ResourceAvailable and ResourceActive counters.</td>
</tr>
<tr>
<td>SWConferenceActive</td>
<td>This counter represents the number of software-based conferences that are currently active (in use) on a SW conference device.</td>
</tr>
<tr>
<td>SWConferenceCompleted</td>
<td>This counter represents the total number of conferences that have been allocated and released on a SW conference device. A conference starts when the first call connects to the bridge. The conference completes when the last call disconnects from the bridge.</td>
</tr>
</tbody>
</table>

Cisco TFTP Server

The Cisco Trivial File Transfer Protocol (TFTP) Server object provides information about the Cisco TFTP server. Table 5-38 contains information on Cisco TFTP server counters.

Table 5-38  Cisco TFTP Server

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BuildAbortCount</td>
<td>This counter represents the number of times that the build process aborted when it received a Build all request. This counter increases when building of device/unit/softkey/dial rules gets aborted as a result of group level change notifications.</td>
</tr>
<tr>
<td>BuildCount</td>
<td>This counter represents the number of times since the TFTP service started that the TFTP server has built all the configuration files in response to a database change notification that affects all devices. This counter increases by one every time the TFTP server performs a new build of all the configuration files.</td>
</tr>
<tr>
<td>BuildDeviceCount</td>
<td>This counter represents the number of devices that were processed in the last build of all the configuration files. This counter also updates while processing device change notifications. The counter increases when a new device is added and decreases when an existing device is deleted.</td>
</tr>
<tr>
<td>BuildDialruleCount</td>
<td>This counter represents the number of dial rules that were processed in the last build of the configuration files. This counter also updates while processing dial rule change notifications. The counter increases when a new dial rule is added and decreases when an existing dial rule is deleted.</td>
</tr>
<tr>
<td>BuildDuration</td>
<td>This counter represents the time in seconds that it took to build the last configuration files.</td>
</tr>
<tr>
<td>BuildSignCount</td>
<td>This counter represents the number of security-enabled phone devices for which the configuration file was digitally signed with the Cisco Unified Communications Manager server key in the last build of all the configuration files. This counter also updates while processing security-enabled phone device change notifications.</td>
</tr>
<tr>
<td>BuildSoftKeyCount</td>
<td>This counter represents the number of softkeys that were processed in the last build of the configuration files. This counter increments when a new softkey is added and decrements when an existing softkey is deleted.</td>
</tr>
</tbody>
</table>
### Performance Monitoring in RTMT

**Chapter 5 / Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts**

#### Table 5-38  Cisco TFTP Server (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BuildUnitCount</td>
<td>This counter represents the number of gateways that were processed in the last build of all the configuration files. This counter also updates while processing unit change notifications. The counter increases when a new gateway is added and decreases when an existing gateway is deleted.</td>
</tr>
<tr>
<td>ChangeNotifications</td>
<td>This counter represents the total number of all the Cisco Unified Communications Manager database change notifications that the TFTP server received. Each time that a device configuration is updated in Cisco Unified Communications Manager Administration, the TFTP server gets sent a database change notification to rebuild the XML file for the updated device.</td>
</tr>
<tr>
<td>DeviceChangeNotifications</td>
<td>This counter represents the number of times that the TFTP server received database change notification to create, update, or delete configuration files for devices.</td>
</tr>
<tr>
<td>DialruleChangeNotifications</td>
<td>This counter represents the number of times that the TFTP server received database change notification to create, update, or delete configuration files for dial rules.</td>
</tr>
<tr>
<td>EncryptCount</td>
<td>This counter represents the number of configuration files that were encrypted. This counter gets updated each time a configuration file is successfully encrypted.</td>
</tr>
<tr>
<td>GKFoundCount</td>
<td>This counter represents the number of GK files that were found in the cache. This counter gets updated each time a GK file is found in the cache.</td>
</tr>
<tr>
<td>GKNotFoundCount</td>
<td>This counter represents the number of GK files that were not found in the cache. This counter gets updated each time a request to get a GK file results in the cache not finding it.</td>
</tr>
<tr>
<td>HeartBeat</td>
<td>This counter represents the heartbeat of the TFTP server. This incremental count indicates that the TFTP server is up and running. If the count does not increase, this means that the TFTP server is down.</td>
</tr>
<tr>
<td>HttpConnectRequests</td>
<td>This counter represents the number of clients that are currently requesting the HTTP GET file request.</td>
</tr>
<tr>
<td>HttpRequests</td>
<td>This counter represents the total number of file requests (such as requests for XML configuration files, phone firmware files, audio files, and so on.) that the HTTP server handled. This counter represents the sum total of the following counters since the HTTP service started: RequestsProcessed, RequestsNotFound, RequestsOverflow, RequestsAborted, and RequestsInProgress.</td>
</tr>
<tr>
<td>HttpRequestsAborted</td>
<td>This counter represents the total number of HTTP requests that the HTTP server canceled (aborted) unexpectedly. Requests could get aborted if the requesting device cannot be reached (for instance, the device lost power) or if the file transfer was interrupted due to network connectivity problems.</td>
</tr>
<tr>
<td>HttpRequestsNotFound</td>
<td>This counter represents the total number of HTTP requests where the requested file was not found. When the HTTP server does not find the requested file, a message gets sent to the requesting device.</td>
</tr>
<tr>
<td>HttpRequestsOverflow</td>
<td>This counter represents the total number of HTTP requests that were rejected when the maximum number of allowable client connections was reached. The requests may have arrived while the TFTP server was building the configuration files or because of some other resource limitation. The Cisco TFTP advanced service parameter, Maximum Serving Count, sets the maximum number of allowable connections.</td>
</tr>
</tbody>
</table>
### Table 5-38 Cisco TFTP Server (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpRequestProcessed</td>
<td>This counter represents the total number of HTTP requests that the HTTP server successfully processed.</td>
</tr>
<tr>
<td>HttpServedFromDisk</td>
<td>This counter represents the number of requests that the HTTP server completed with the files that are on disk and not cached in memory.</td>
</tr>
<tr>
<td>LDFoundCount</td>
<td>This counter represents the number of LD files that were found in the cache. This counter gets updated each time that a LD file is found in cache memory.</td>
</tr>
<tr>
<td>LDNotFoundCount</td>
<td>This counter represents the number of LD files that were not found in cache memory. This counter gets updated each time that a request to get an LD file results in the cache not finding it.</td>
</tr>
<tr>
<td>MaxServingCount</td>
<td>This counter represents the maximum number of client connections that the TFTP can serve simultaneously. The Cisco TFTP advanced service parameter, Maximum Serving Count, sets this value.</td>
</tr>
<tr>
<td>Requests</td>
<td>This counter represents the total number of file requests (such as requests for XML configuration files, phone firmware files, audio files, and so on.) that the TFTP server handles. This counter represents the sum total of the following counters since the TFTP service started: RequestsProcessed, RequestsNotFound, RequestsOverflow, RequestsAborted, and RequestsInProgress.</td>
</tr>
<tr>
<td>RequestsAborted</td>
<td>This counter represents the total number of TFTP requests that the TFTP server canceled (aborted) unexpectedly. Requests could get aborted if the requesting device cannot be reached (for instance, the device lost power) or if the file transfer was interrupted due to network connectivity problems.</td>
</tr>
<tr>
<td>RequestsInProgress</td>
<td>This counter represents the number of file requests that the TFTP server currently is processing. This counter increases for each new file request and decreases for each file request that completes. This counter indicates the current load of the TFTP server.</td>
</tr>
<tr>
<td>RequestsNotFound</td>
<td>This counter represents the total number of TFTP requests for which the requested file was not found. When the TFTP server does not find the requested file, a message gets sent to the requesting device. If this counter increments in a cluster that is configured as secure, this event usually indicates an error condition. If, however, the cluster is configured as non-secure, it is normal for the CTL file to be absent (not found), which results in a message being sent to the requesting device and a corresponding increment in this counter. For non-secure clusters, this normal occurrence does not represent an error condition.</td>
</tr>
<tr>
<td>RequestsOverflow</td>
<td>This counter represents the total number of TFTP requests that were rejected because the maximum number of allowable client connections was exceeded, because requests arrived while the TFTP server was building the configuration files, or because of some other resource limitation. The Cisco TFTP advanced service parameter, Maximum Serving Count, sets the maximum number of allowable connections.</td>
</tr>
<tr>
<td>RequestsProcessed</td>
<td>This counter represents the total number of TFTP requests that the TFTP server successfully processed.</td>
</tr>
</tbody>
</table>
Table 5-38  Cisco TFTP Server (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SegmentsAckowed</td>
<td>This counter represents the total number of data segments that the client devices acknowledged. Files get sent to the requesting device in data segments of 512 bytes, and for each 512-byte segment, the device sends the TFTP server an acknowledgment message. Each additional data segment gets sent upon receipt of the acknowledgment for the previous data segment until the complete file successfully gets transmitted to the requesting device.</td>
</tr>
<tr>
<td>SegmentsFromDisk</td>
<td>This counter represents the number of data segments that the TFTP server reads from the files on disk, while serving files.</td>
</tr>
<tr>
<td>SegmentSent</td>
<td>This counter represents the total number of data segments that the TFTP server sent. Files get sent to the requesting device in data segments of 512 bytes.</td>
</tr>
<tr>
<td>SEFFoundCount</td>
<td>This counter represents the number of SEP files that were successfully found in the cache. This counter gets updated each time a SEP file is found in the cache.</td>
</tr>
<tr>
<td>SEPNotFoundCount</td>
<td>This counter represents the number of SEP files that were not found in the cache. This counter gets updated each time a request to get a SEP file produces a not found in cache memory result.</td>
</tr>
<tr>
<td>SIPFoundCount</td>
<td>This counter represents the number of SIP files that were successfully found in the cache. This counter gets updated each time a SIP file is found in the cache.</td>
</tr>
<tr>
<td>SIPNotFoundCount</td>
<td>This counter represents the number of SIP files that were not found in the cache. This counter gets updated each time a request to get a SIP file produces a not found in cache memory result.</td>
</tr>
<tr>
<td>SoftkeyChangeNotifications</td>
<td>This counter represents the number of times that the TFTP server received database change notification to create, update, or delete configuration files for softkeys.</td>
</tr>
<tr>
<td>UnitChangeNotifications</td>
<td>This counter represents the number of times that the TFTP server received database change notification to create, update, or delete gateway-related configuration files.</td>
</tr>
</tbody>
</table>

Cisco Transcode Device

The Cisco Transcode Device object provides information about registered Cisco transcoding devices. Table 5-39 contains information on Cisco transcoder device counters.

Table 5-39  Cisco Transcode Device

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OutOfResources</td>
<td>This counter represents the total number of times that an attempt was made to allocate a transcoder resource from a transcoding device and failed; for example, because all resources were already in use.</td>
</tr>
<tr>
<td>ResourceActive</td>
<td>This counter represents the number of transcoder resources that are currently in use (active) for a transcoding device. Each transcoder resource uses two streams.</td>
</tr>
</tbody>
</table>
Cisco Video Conference Bridge

The Cisco Video Conference Bridge object provides information about registered Cisco video conference bridge devices. Table 5-40 contains information on Cisco video conference bridge device counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConferencesActive</td>
<td>This counter represents the total number of video conferences that are currently active (in use) on a video conference bridge device. The system specifies a conference as active when the first call connects to the bridge.</td>
</tr>
<tr>
<td>ConferencesAvailable</td>
<td>This counter represents the number of video conferences that are not active and are still available on a video conference device.</td>
</tr>
<tr>
<td>ConferencesCompleted</td>
<td>This counter represents the total number of video conferences that have been allocated and released on a video conference device. A conference starts when the first call connects to the bridge. The conference completes when the last call disconnects from the bridge.</td>
</tr>
<tr>
<td>ConferencesTotal</td>
<td>This counter represents the total number of video conferences that are configured for a video conference device.</td>
</tr>
<tr>
<td>OutOfConferences</td>
<td>This counter represents the total number of times that an attempt was made to initiate a video conference from a video conference device and failed because the device already had the maximum number of active conferences that is allowed (as specified by the TotalConferences counter).</td>
</tr>
<tr>
<td>OutOfResources</td>
<td>This counter represents the total number of times that an attempt was made to allocate a conference resource from a video conference device and failed, for example, because all resources were already in use.</td>
</tr>
<tr>
<td>ResourceActive</td>
<td>This counter represents the total number of resources that are currently active (in use) on a video conference bridge device. One resource gets used per participant.</td>
</tr>
<tr>
<td>ResourceAvailable</td>
<td>This counter represents the total number of resources that are not active and are still available on a device to handle additional participants for a video conference bridge device.</td>
</tr>
<tr>
<td>ResourceTotal</td>
<td>This counter represents the total number of resources that are configured on a video conference bridge device. One resource gets used per participant.</td>
</tr>
</tbody>
</table>

Table 5-40 Cisco Video Conference Bridge

Table 5-39 Cisco Transcode Device (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourceAvailable</td>
<td>This counter represents the total number of resources that are not active and are still available to be used now for a transcoder device. Each transcoder resource uses two streams.</td>
</tr>
<tr>
<td>ResourceTotal</td>
<td>This counter represents the total number of transcoder resources that a transcoder device provided. This counter equals the sum of the ResourceActive and ResourceAvailable counters.</td>
</tr>
</tbody>
</table>
Cisco Web Dialer

The Cisco Web Dialer object provides information about the Cisco Web Dialer application and the Redirector servlet. Table 5-41 contains information on the Cisco Web Dialer counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsCompleted</td>
<td>This counter represents the number of Make Call and End Call requests that the Cisco Web Dialer application successfully completed.</td>
</tr>
<tr>
<td>CallsFailed</td>
<td>This counter represents the number of Make Call and End Call requests that were unsuccessful.</td>
</tr>
<tr>
<td>RedirectorSessionsHandled</td>
<td>This counter represents the total number of HTTP sessions that the Redirector servlet handled since the last service startup.</td>
</tr>
<tr>
<td>RedirectorSessionsInProgress</td>
<td>This counter represents the number of HTTP sessions that are currently being serviced by the Redirector servlet.</td>
</tr>
<tr>
<td>RequestsCompleted</td>
<td>This counter represents the number of Make Call and End Call requests that the Web Dialer servlet successfully completed.</td>
</tr>
<tr>
<td>RequestsFailed</td>
<td>This counter represents the number of Make Call and End Call requests that failed.</td>
</tr>
<tr>
<td>SessionsHandled</td>
<td>This counter represents the total number of CTI sessions that the Cisco Web Dialer servlet handled since the last service startup.</td>
</tr>
<tr>
<td>SessionsInProgress</td>
<td>This counter represents the number of CTI sessions that the Cisco Web Dialer servlet is currently servicing.</td>
</tr>
</tbody>
</table>

Cisco WSM Connector

The WSM object provides information on WSMConnectors that are configured on Cisco Unified Communications Manager. Each WSMConnector represents a physical Motorola WSM device. Table 5-42 contains information on the Cisco WSM Connector counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsActive</td>
<td>This counter represents the number of calls that are currently active (in use) on the WSMConnector device.</td>
</tr>
<tr>
<td>CallsAttempted</td>
<td>This counter represents the number of calls that have been attempted on the WSMConnector device, including both successful and unsuccessful call attempts.</td>
</tr>
<tr>
<td>CallsCompleted</td>
<td>This counter represents the number of calls that are connected (a voice path was established) through the WSMConnector device. The counter increments when the call terminates.</td>
</tr>
<tr>
<td>CallsInProgress</td>
<td>This counter represents the number of calls that are currently in progress on the WSMConnector device. This includes all active calls. When the number of CallsInProgress equals the number of CallsActive, this indicates that all calls are connected.</td>
</tr>
<tr>
<td>DMMSRegistered</td>
<td>This counter represents the number of DMMS subscribers that are registered to the WSM.</td>
</tr>
</tbody>
</table>
PerfMon Objects and Counters for System

This section provides information on Cisco Unified Communications Manager System PerfMon objects and counters.

Cisco Tomcat Connector

The Tomcat Hypertext Transport Protocol (HTTP)/HTTP Secure (HTTPS) Connector object provides information about Tomcat connectors. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related web pages are accessed. The Secure Socket Layer (SSL) status of the URLs for web applications provides the basis for the instance name for each Tomcat HTTP Connector. For example, https://<IP Address>:8443 for SSL or http://<IP Address>:8080 for non-SSL. Table 5-43 contains information on the Tomcat HTTP connector counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Errors</td>
<td>This counter represents the total number of HTTP errors (for example, 401 Unauthorized) that the connector encountered. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides the basis for the instance name for each Tomcat HTTP connector. For example, https://&lt;IP Address&gt;:8443 for SSL or http://&lt;IP Address&gt;:8080 for non-SSL.</td>
</tr>
<tr>
<td>MBytesReceived</td>
<td>This counter represents the amount of data that the connector received. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides the basis for the instance name for each Tomcat HTTP connector. For example, https://&lt;IP Address&gt;:8443 for SSL or http://&lt;IP Address&gt;:8080 for non-SSL.</td>
</tr>
<tr>
<td>MBytesSent</td>
<td>This counter represents the amount of data that the connector sent. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides the basis for the instance name for each Tomcat HTTP connector. For example, https://&lt;IP Address&gt;:8443 for SSL or http://&lt;IP Address&gt;:8080 for non-SSL.</td>
</tr>
</tbody>
</table>
### Table 5-43  Cisco Tomcat Connector (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requests</td>
<td>This counter represents the total number of request that the connector handled. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides basis for the instance name for each Tomcat HTTP connector. For example, https://&lt;IP Address&gt;:8443 for SSL or http://&lt;IP Address&gt;:8080 for non-SSL.</td>
</tr>
<tr>
<td>ThreadsTotal</td>
<td>This counter represents the current total number of request processing threads, including available and in-use threads, for the connector. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides basis for the instance name for each Tomcat HTTP connector. For example, https://&lt;IP Address&gt;:8443 for SSL or http://&lt;IP Address&gt;:8080 for non-SSL.</td>
</tr>
<tr>
<td>ThreadsMax</td>
<td>This counter represents the maximum number of request processing threads for the connector. Each incoming request on a Cisco Unified Communications Manager related window requires a thread for the duration of that request. If more simultaneous requests are received than the currently available request processing threads can handle, additional threads will get created up to the configured maximum shown in this counter. If still more simultaneous requests are received, they accumulate within the server socket that the connector created, up to an internally specified maximum number. Any further simultaneous requests will receive connection refused messages until resources are available to process them. A Tomcat HTTP connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when Cisco Unified Communications Manager related windows are accessed. The Secure Socket Layer (SSL) status of the URLs for the web application provides the basis for the instance name for each Tomcat HTTP connector. For example, https://&lt;IP Address&gt;:8443 for SSL or http://&lt;IP Address&gt;:8080 for non-SSL.</td>
</tr>
<tr>
<td>ThreadsBusy</td>
<td>This counter represents the current number of busy/in-use request processing threads for the connector. A Tomcat Connector represents an endpoint that receives requests and sends responses. The connector handles HTTP/HTTPS requests and sends HTTP/HTTPS responses that occur when web pages that are related to Cisco Unified Communications Manager are accessed. The Secure Sockets Layer (SSL) status of the URLs for the web application provides the basis for the instance name for each Tomcat connector. For example, https://&lt;IP Address&gt;:8443 for SSL or http://&lt;IP Address&gt;:8080 for non-SSL.</td>
</tr>
</tbody>
</table>
Cisco Tomcat JVM

The Cisco Tomcat Java Virtual Machine (JVM) object provides information about the Tomcat JVM, which represents, among other things, a pool of common resource memory that Cisco Unified Communications Manager related web applications such as Cisco Unified Communications Manager Administration, Cisco Unified Serviceability, Cisco Unity Connection Administration, and more use. Table 5-44 contains information on the Tomcat JVM counters.

Table 5-44  Tomcat JVM

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KBytesMemoryFree</td>
<td>This counter represents the amount of free dynamic memory block (heap memory) in the Tomcat Java Virtual Machine. The dynamic memory block stores all objects that Tomcat and its web applications, such as Cisco Unified Communications Manager Administration, Cisco Unified Serviceability, and Cisco Unity Connection Administration create. When the amount of free dynamic memory is low, more memory gets automatically allocated, and total memory size (represented by the KbytesMemoryTotal counter) increases but only up to the maximum (represented by the KbytesMemoryMax counter). You can determine the amount of memory in use by subtracting KBytesMemoryFree from KbytesMemoryTotal.</td>
</tr>
<tr>
<td>KBytesMemoryMax</td>
<td>This counter represents the amount of free dynamic memory block (heap memory) in the Tomcat Java Virtual Machine. The dynamic memory block stores all objects that Tomcat and its web applications, such as Cisco Unified Communications Manager Administration, Cisco Unified Serviceability, and Cisco Unity Connection Administration, create.</td>
</tr>
<tr>
<td>KBytesMemoryTotal</td>
<td>This counter represents the current total dynamic memory block size, including free and in-use memory, of Tomcat Java Virtual Machine. The dynamic memory block stores all objects that Tomcat and its web applications, such as Cisco Unified Communications Manager Administration, Cisco Unified Serviceability, and Cisco Unity Connection Administration, create.</td>
</tr>
</tbody>
</table>

Cisco Tomcat Web Application

The Cisco Tomcat Web Application object provides information about how to run Cisco Unified Communications Manager web applications. The URLs for the web application provide basis for the instance name for each Tomcat Web Application. For example, Cisco Unified Communications Manager Administration (https://<IP Address>:8443/ccmadmin) gets identified by ccmadmin, Cisco Unified Serviceability gets identified by ccmservice, Cisco Unified Communications Manager User Options gets identified by ccmuser, Cisco Unity Connection Administration (https://<IP Address>:8443/cuadmin) gets identified by cuadmin, and URLs that do not have an extension, such as https://<IP Address>:8443 or http://<IP Address>:8080, get identified by _root. Table 5-45 contains information on the Tomcat Web Application counters.
Chapter 5  Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

**Table 5-45  Tomcat Web Application**

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Errors</td>
<td>This counter represents the total number of HTTP errors (for example, 401 Unauthorized) that a Cisco Unified Communications Manager related web application encountered. The URLs for the web application provide the basis instance name for each Tomcat Web Application. For example, Cisco Unified Communications Manager Administration (https://&lt;IP Address&gt;:8443/ccmadmin) gets identified by ccmadmin, Cisco Unified Serviceability gets identified by ccmservice, Cisco Unified Communications Manager User Options gets identified by ccmuser, Cisco Unity Connection Administration (https://&lt;IP Address&gt;:8443/cuadmin) gets identified by cuadmin, and URLs that do not have an extension, such as https://&lt;IP Address&gt;:8443 or http://&lt;IP Address&gt;:8080), get identified by _root.</td>
</tr>
<tr>
<td>Requests</td>
<td>This counter represents the total number of requests that the web application handles. Each time that a web application is accessed, its Requests counter increments accordingly. The URLs for the web application provide the basis instance name for each Tomcat Web Application. For example, Cisco Unified Communications Manager Administration (https://&lt;IP Address&gt;:8443/ccmadmin) gets identified by ccmadmin, Cisco Unified Serviceability gets identified by ccmservice, Cisco Unified Communications Manager User Options gets identified by ccmuser, Cisco Unity Connection Administration (https://&lt;IP Address&gt;:8443/cuadmin) gets identified by cuadmin, and URLs that do not have an extension, such as https://&lt;IP Address&gt;:8443 or http://&lt;IP Address&gt;:8080), get identified by _root.</td>
</tr>
<tr>
<td>SessionsActive</td>
<td>This counter represents the number of sessions that the web application currently has active (in use). The URLs for the web application provide the basis instance name for each Tomcat Web Application. For example, Cisco Unified Communications Manager Administration (https://&lt;IP Address&gt;:8443/ccmadmin) gets identified by ccmadmin, Cisco Unified Serviceability gets identified by ccmservice, Cisco Unified Communications Manager User Options gets identified by ccmuser, Cisco Unity Connection Administration (https://&lt;IP Address&gt;:8443/cuadmin) gets identified by cuadmin, and URLs that do not have an extension, such as https://&lt;IP Address&gt;:8443 or http://&lt;IP Address&gt;:8080), get identified by _root.</td>
</tr>
</tbody>
</table>

**Database Change Notification Client**

The Database Change Notification Client object provides information on change notification clients. Table 5-46 contains information on the Database Change Notification Client counters.

**Table 5-46  Database Change Notification Client**

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>MessagesProcessed</td>
<td>This counter represents the number of database change notifications that have been processed. This counter refreshes every 15 seconds.</td>
</tr>
<tr>
<td>MessagesProcessing</td>
<td>This counter represents the number of change notification messages that are currently being processed or are waiting to be processed in the change notification queue for this client. This counter refreshes every 15 seconds.</td>
</tr>
</tbody>
</table>
Table 5-46  Database Change Notification Client (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>QueueHeadPointer</td>
<td>This counter represents the head pointer to the change notification queue. The head pointer acts as the starting point in the change notification queue. To determine the number of notifications in the queue, subtract the head pointer value from the tail pointer value. By default, this counter refreshes every 15 seconds.</td>
</tr>
<tr>
<td>QueueMax</td>
<td>This counter represents the largest number of change notification messages that will be processed for this client. This counter remains cumulative since the last restart of the Cisco Database Layer Monitor service.</td>
</tr>
<tr>
<td>QueueTailPointer</td>
<td>This counter represents the tail pointer to the change notification queue. The tail pointer represents the ending point in the change notification queue. To determine the number of notifications in the queue, subtract the head pointer value from the tail pointer value. By default, this counter refreshes every 15 seconds</td>
</tr>
<tr>
<td>TablesSubscribed</td>
<td>This counter represents the number of tables in which this client has subscribed.</td>
</tr>
</tbody>
</table>

Database Change Notification Server

The Database Change Notification Server object provides information on different change-notification-related statistics. Table 5-47 contains information on the Database Change Notification Server counters.

Table 5-47  Database Change Notification Server

<table>
<thead>
<tr>
<th>Counter</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clients</td>
<td>This counter represents the number of change notification clients (services/servlets) that have subscribed for change notification.</td>
</tr>
<tr>
<td>Queue Delay</td>
<td>This counter provides the number of seconds that the change notification process has messages to process but is not processing them. This condition is true if:</td>
</tr>
<tr>
<td></td>
<td>• Either Change Notification Requests Queued in Database (QueuedRequestsInDB) and Change Notification Requests Queued in Memory (QueuedRequestsInMemory) are non-zero, or</td>
</tr>
<tr>
<td></td>
<td>• The Latest Change Notification Messages Processed count is not changing. This condition gets checked every 15 seconds.</td>
</tr>
<tr>
<td>QueuedRequestsInDB</td>
<td>This counter represents the number of change notification records that are in the DBCNQueue (Database Change Notification Queue) table via direct TCP/IP connection (not queued in shared memory). This counter refreshes every 15 seconds.</td>
</tr>
<tr>
<td>QueuedRequestsInMemory</td>
<td>This counter represents the number of change notification requests that are queued in shared memory.</td>
</tr>
</tbody>
</table>
Database Change Notification Subscription

The Database Change Notification Subscription object displays the names of tables where the client will receive Change Notifications.

The SubscribedTable object displays the table with the service or servlet that will receive change notifications. Because the counter does not increment, this display occurs for informational purposes only.

Database Local DSN

The Database Local Data Source Name (DSN) object and LocalDSN counter provide the DSN information for the local machine. Table 5-48 contains information on the Database local DSN.

Table 5-48 Database Local Data Source Name

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>CcmDbSpace_Used</td>
<td>This counter represents the amount of Ccm DbSpace that is being consumed.</td>
</tr>
<tr>
<td>CcmtempDbSpace_Used</td>
<td>This counter represents the amount of Ccmtemp DbSpace that is being consumed.</td>
</tr>
<tr>
<td>CNDbSpace_Used</td>
<td>This counter represents the percentage of CN dbspace consumed.</td>
</tr>
<tr>
<td>LocalDSN</td>
<td>This counter represents the data source name (DSN) that is being referenced from the local machine.</td>
</tr>
<tr>
<td>SharedMemory_Free</td>
<td>This counter represents total shared memory that is free.</td>
</tr>
<tr>
<td>SharedMemory_Used</td>
<td>This counter total shared memory that is used.</td>
</tr>
<tr>
<td>RootDbSpace_Used</td>
<td>This counter represents the amount of RootDbSpace that is being consumed.</td>
</tr>
</tbody>
</table>

DB User Host Information Counters

The DB User Host Information object provides information on DB User Host. The DB:User:Host Instance object displays the number of connections that are present for each instance of DB:User:Host.

Enterprise Replication DBSpace Monitors

The enterprise replication DBSpace monitors object displays the usage of various ER DbSpaces. Table 5-49 contains information on the enterprise replication DB monitors.

Table 5-49 Enterprise Replication DBSpace Monitors

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERDbSpace_Used</td>
<td>This counter represents the amount of enterprise replication DbSpace that was consumed.</td>
</tr>
<tr>
<td>ERSBDDbSpace_Used</td>
<td>This counter represents the amount of ERDbSpace that was consumed.</td>
</tr>
</tbody>
</table>
Enterprise Replication Perfmon Counters

The Enterprise Replication Perfmon Counter object provides information on the various replication counters. The ServerName:ReplicationQueueDepth counter displays the server name followed by the replication queue depth.

IP

The IP object provides information on the IP statistics on your system. Table 5-50 contains information on the IP counters.

Table 5-50 IP

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frag Creates</td>
<td>This counter represents the number of IP datagrams fragments that have been generated at this entity.</td>
</tr>
<tr>
<td>Frag Fails</td>
<td>This counter represents the number of IP datagrams that were discarded at this entity because the datagrams could not be fragmented, such as datagrams where the Do not Fragment flag was set.</td>
</tr>
<tr>
<td>Frag OKs</td>
<td>This counter represents the number of IP datagrams that were successfully fragmented at this entity.</td>
</tr>
<tr>
<td>In Delivers</td>
<td>This counter represents the number of input datagrams that were delivered to IP user protocols. This includes Internet Control Message Protocol (ICMP).</td>
</tr>
<tr>
<td>In Discards</td>
<td>This counter represents the number of discarded input IP datagrams when no problems were encountered. Lack of buffer space provides one possible reason. This counter does not include any datagrams that were discarded while they were awaiting reassembly.</td>
</tr>
<tr>
<td>In HdrErrors</td>
<td>This counter represents the number of discarded input datagrams that had header errors. This includes bad checksums, version number mismatch, other format errors, time-to-live exceeded, and other errors that were discovered in processing IP options.</td>
</tr>
<tr>
<td>In Receives</td>
<td>This counter represents the number of input datagrams that were received from all network interfaces. This counter includes datagrams that were received with errors.</td>
</tr>
<tr>
<td>In UnknownProtos</td>
<td>This counter represents the number of locally addressed datagrams that were received successfully but discarded because of an unknown or unsupported protocol.</td>
</tr>
<tr>
<td>InOut Requests</td>
<td>This counter represents the number of incoming IP datagrams that were received and the number of outgoing IP datagrams that were sent.</td>
</tr>
<tr>
<td>Out Discards</td>
<td>This counter represents the number of output IP datagrams that were not transmitted and were discarded. Lack of buffer space provides one possible reason.</td>
</tr>
<tr>
<td>Out Requests</td>
<td>This counter represents the total number of IP datagrams that local IP protocols, including ICMP, supply to IP in requests transmission. This counter does not include any datagrams that were counted in ForwDatagrams.</td>
</tr>
</tbody>
</table>
Performance Monitoring in RTMT

Table 5-50  IP (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reasm Fails</td>
<td>This counter represents the number of IP reassembly failures that the IP reassembly algorithm detected, including time outs, errors, and so on. This counter does not represent the discarded IP fragments because some algorithms, such as the algorithm in RFC 815, can lose track of the number of fragments because it combines them as they are received.</td>
</tr>
<tr>
<td>Reasm OKs</td>
<td>This counter represents the number of IP datagrams that were successfully reassembled.</td>
</tr>
<tr>
<td>Reasm Reqds</td>
<td>This counter represents the number of IP fragments that were received that required reassembly at this entity.</td>
</tr>
</tbody>
</table>

Table 5-51  Memory

The memory object provides information about the usage of physical memory and swap memory on the server. Table 5-51 contains information on memory counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>% Mem Used</td>
<td>This counter displays the system physical memory utilization as a percentage. The value of this counter equals (Total KBytes - Free KBytes - Buffers KBytes - Cached KBytes + Shared KBytes) / Total KBytes, which also corresponds to the Used KBytes/Total KBytes.</td>
</tr>
<tr>
<td>% Page Usage</td>
<td>This counter represents the percentage of active pages.</td>
</tr>
<tr>
<td>% VM Used</td>
<td>This counter displays the system virtual memory utilization as a percentage. The value of this counter equals (Total KBytes - Free KBytes - Buffers KBytes - Cached KBytes + Shared KBytes + Used Swap KBytes) / (Total KBytes + Total Swap KBytes), which also corresponds to Used VM KBytes/Total VM KBytes.</td>
</tr>
<tr>
<td>Buffers KBytes</td>
<td>This counter represents the capacity of buffers in your system in kilobytes.</td>
</tr>
<tr>
<td>Cached KBytes</td>
<td>This counter represents the amount of cached memory in kilobytes.</td>
</tr>
<tr>
<td>Free KBytes</td>
<td>This counter represents the total amount of memory that is available in your system in kilobytes.</td>
</tr>
<tr>
<td>Free Swap KBytes</td>
<td>This counter represents the amount of free swap space that is available in your system in kilobytes.</td>
</tr>
<tr>
<td>Faults Per Sec</td>
<td>This counter represents the number of page faults (both major and minor) that the system made per second (post 2.5 kernels only). This does not necessarily represent a count of page faults that generate I/O because some page faults can get resolved without I/O.</td>
</tr>
<tr>
<td>Low Total</td>
<td>This counter represents the total low (non-paged) memory for kernel.</td>
</tr>
<tr>
<td>Low Free</td>
<td>This counter represents the total free low (non-paged) memory for kernel.</td>
</tr>
<tr>
<td>Major Faults Per Sec</td>
<td>This counter represents the number of major faults that the system has made per second that have required loading a memory page from disk (post 2.5 kernels only).</td>
</tr>
<tr>
<td>Pages</td>
<td>This counter represents the number of pages that the system paged in from the disk plus the number of pages that the system paged out to the disk.</td>
</tr>
</tbody>
</table>
Table 5-51 Memory (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pages Input</td>
<td>This counter represents the number of pages that the system paged in from the disk.</td>
</tr>
<tr>
<td>Pages Input Per Sec</td>
<td>This counter represents the total number of kilobytes that the system paged in from the disk per second.</td>
</tr>
<tr>
<td>Pages Output</td>
<td>This counter represents the number of pages that the system paged out to the disk.</td>
</tr>
<tr>
<td>Pages Output Per Sec</td>
<td>This counter represents the total number of kilobytes that the system paged out to the disk per second.</td>
</tr>
<tr>
<td>Shared KBytes</td>
<td>This counter represents the amount of shared memory in your system in kilobytes.</td>
</tr>
<tr>
<td>Total KBytes</td>
<td>This counter represents the total amount of memory in your system in kilobytes.</td>
</tr>
<tr>
<td>Total Swap KBytes</td>
<td>This counter represents the total amount of swap space in your system in kilobytes.</td>
</tr>
<tr>
<td>Total VM KBytes</td>
<td>This counter represents the total amount of system physical and memory and swap space (Total Kbytes + Total Swap Kbytes) that is in use in your system in kilobytes.</td>
</tr>
<tr>
<td>Used KBytes</td>
<td>This counter represents the amount of system physical memory that is in use in kilobytes. The value of the Used KBytes counter equals Total KBytes minus Free KBytes minus Buffers KBytes minus Cached KBytes plus Shared KBytes. In a Linux environment, the Used KBytes value that displays in the top or free command output equals the difference of Total KBytes and Free KBytes and also includes the sum of Buffers KBytes and Cached KBytes.</td>
</tr>
<tr>
<td>Used Swap KBytes</td>
<td>This counter represents the amount of swap space that is in use on your system in kilobytes.</td>
</tr>
<tr>
<td>Used VM KBytes</td>
<td>This counter represents the system physical memory and the amount of swap space that is in use on your system in kilobytes. The value equals Total KBytes - Free KBytes - Buffers KBytes - Cached KBytes + Shared KBytes + Used Swap KBytes. This corresponds to Used Mem KBytes + Used Swap KBytes.</td>
</tr>
</tbody>
</table>

Network Interface

The network interface object provides information about the network interfaces on the system. Table 5-52 contains information on network interface counters.

Table 5-52 Network Interface

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rx Bytes</td>
<td>This counter represents the number of bytes, including framing characters, that were received on the interface.</td>
</tr>
<tr>
<td>Rx Dropped</td>
<td>This counter represents the number of inbound packets that were chosen to be discarded even though no errors had been detected. This prevents the packet from being delivered to a higher layer protocol. Discarding packets to free up buffer space provides one reason.</td>
</tr>
</tbody>
</table>
Performance Monitoring in RTMT

Table 5-52  Network Interface (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rx Errors</td>
<td>This counter represents the number of inbound packets (packet-oriented interfaces) and the number of inbound transmission units (character-oriented or fixed-length interfaces) that contained errors that prevented them from being deliverable to a higher layer protocol.</td>
</tr>
<tr>
<td>Rx Multicast</td>
<td>This counter represents the number of multicast packets that were received on this interface.</td>
</tr>
<tr>
<td>Rx Packets</td>
<td>This counter represents the number of packets that this sublayer delivered to a higher sublayer. This does not include the packets that were addressed to a multicast or broadcast address at this sublayer.</td>
</tr>
<tr>
<td>Total Bytes</td>
<td>This counter represents the total number of received (Rx) bytes and transmitted (Tx) bytes.</td>
</tr>
<tr>
<td>Total Packets</td>
<td>This counter represents the total number of Rx packets and Tx packets.</td>
</tr>
<tr>
<td>Tx Bytes</td>
<td>This counter represents the total number of octets, including framing characters, that were transmitted out from the interface.</td>
</tr>
<tr>
<td>Tx Dropped</td>
<td>This counter represents the number of outbound packets that were chosen to be discarded even though no errors were detected. This action prevents the packet from being delivered to a higher layer protocol. Discarding a packet to free up buffer space represents one reason.</td>
</tr>
<tr>
<td>Tx Errors</td>
<td>This counter represents the number of outbound packets (packet-oriented interfaces) and the number of outbound transmission units (character-oriented or fixed-length interfaces) that could not be transmitted because of errors.</td>
</tr>
<tr>
<td>Tx Packets</td>
<td>This counter represents the total number of packets that the higher level protocols requested for transmission, including those that were discarded or not sent. This does not include packets that were addressed to a multicast or broadcast address at this sublayer.</td>
</tr>
<tr>
<td>Tx QueueLen</td>
<td>This counter represents the length of the output packet queue (in packets).</td>
</tr>
</tbody>
</table>

Number of Replicates Created and State of Replication

The Number of Replicates Created and State of Replication object provides real-time replication information for the system. Table 5-53 contains information on replication counters.
Partition

The partition object provides information about the file system and its usage in the system. Table 5-54 contains information on partition counters. Be aware that these counters are available for the spare partition, if present.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>% CPU Time</td>
<td>This counter represents the percentage of CPU time that is dedicated to handling I/O requests that were issued to the disk. This counter is no longer valid when the counter value equals -1.</td>
</tr>
<tr>
<td>% Used</td>
<td>This counter represents the percentage of disk space that is in use on this file system.</td>
</tr>
<tr>
<td>% Wait in Read</td>
<td>Not Used. The Await Read Time counter replaces this counter. This counter is no longer valid when the counter value equals -1.</td>
</tr>
<tr>
<td>% Wait in Write</td>
<td>Not Used. The Await Write Time counter replaces this counter. This counter is no longer valid when the counter value equals -1.</td>
</tr>
<tr>
<td>Await Read Time</td>
<td>This counter represents the average time, measured in milliseconds, for Read requests that are issued to the device to be served. This counter is no longer valid when the counter value equals -1.</td>
</tr>
<tr>
<td>Await Time</td>
<td>This counter represents the average time, measured in milliseconds, for I/O requests that were issued to the device to be served. This includes the time that the requests spent in queue and the time that was spent servicing them. This counter is no longer valid when the counter value equals -1.</td>
</tr>
</tbody>
</table>
Chapter 5  Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

Performance Monitoring in RTMT

Table 5-54  Partition (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Await Write Time</td>
<td>This counter represents the average time, measured in milliseconds, for write requests that are issued to the device to be served. This counter is no longer valid when the counter value equals -1.</td>
</tr>
<tr>
<td>Queue Length</td>
<td>This counter represents the average queue length for the requests that were issued to the disk. This counter is no longer valid when the counter value equals -1.</td>
</tr>
<tr>
<td>Read Bytes Per Sec</td>
<td>This counter represents the amount of data in bytes per second that was read from the disk.</td>
</tr>
<tr>
<td>Total Mbytes</td>
<td>This counter represents the amount of total disk space in megabytes that is on this file system.</td>
</tr>
<tr>
<td>Used Mbytes</td>
<td>This counter represents the amount of disk space in megabytes that is in use on this file system.</td>
</tr>
<tr>
<td>Write Bytes Per Sec</td>
<td>This counter represents the amount of data that was written to the disk in bytes per second.</td>
</tr>
</tbody>
</table>

Process

The process object provides information about the processes that are running on the system. Table 5-55 contains information on process counters.

Table 5-55  Process

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>% CPU Time</td>
<td>This counter, which is expressed as a percentage of total CPU time, represents the tasks share of the elapsed CPU time since the last update.</td>
</tr>
<tr>
<td>% MemoryUsage</td>
<td>This counter represents the percentage of physical memory that a task is currently using.</td>
</tr>
<tr>
<td>Data Stack Size</td>
<td>This counter represents the stack size for task memory status.</td>
</tr>
<tr>
<td>Nice</td>
<td>This counter represents the nice value of the task. A negative nice value indicates that the process has a higher priority while a positive nice value indicates that the process has a lower priority. If the nice value equals zero, do not adjust the priority when you are determining the dispatchability of a task.</td>
</tr>
<tr>
<td>Page Fault Count</td>
<td>This counter represents the number of major page faults that a task encountered that required the data to be loaded into memory.</td>
</tr>
<tr>
<td>PID</td>
<td>This counter displays the task-unique process ID. The ID periodically wraps, but the value will never equal zero.</td>
</tr>
</tbody>
</table>
Performance Monitoring in RTMT

Table 5-55  Process (continued)

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Status</td>
<td>This counter displays the process status:</td>
</tr>
<tr>
<td></td>
<td>• 0—Running</td>
</tr>
<tr>
<td></td>
<td>• 1—Sleeping</td>
</tr>
<tr>
<td></td>
<td>• 2—Uninterruptible disk sleep</td>
</tr>
<tr>
<td></td>
<td>• 3—Zombie</td>
</tr>
<tr>
<td></td>
<td>• 4—Stopped</td>
</tr>
<tr>
<td></td>
<td>• 5—Paging</td>
</tr>
<tr>
<td></td>
<td>• 6—Unknown</td>
</tr>
<tr>
<td>Shared Memory Size</td>
<td>This counter displays the amount of shared memory (KB) that a task is using.</td>
</tr>
<tr>
<td></td>
<td>Other processes could potentially share the same memory.</td>
</tr>
<tr>
<td>STime</td>
<td>This counter displays the system time (STime), measured in jiffies, that this</td>
</tr>
<tr>
<td></td>
<td>process has scheduled in kernel mode. A jiffy corresponds to a unit of CPU time</td>
</tr>
<tr>
<td></td>
<td>and gets used as a base of measurement. One second comprises 100 jiffies.</td>
</tr>
<tr>
<td>Thread Count</td>
<td>This counter displays the number of threads that are currently grouped with a task.</td>
</tr>
<tr>
<td></td>
<td>A negative value (-1) indicates that this counter is currently not available. This</td>
</tr>
<tr>
<td></td>
<td>happens when thread statistics (which includes all performance counters in the</td>
</tr>
<tr>
<td></td>
<td>Thread object as well as the Thread Count counter in the Process object) are</td>
</tr>
<tr>
<td></td>
<td>turned off because the system total processes and threads exceeded the default</td>
</tr>
<tr>
<td></td>
<td>threshold value.</td>
</tr>
<tr>
<td>Total CPU Time Used</td>
<td>This counter displays the total CPU time in jiffies that the task used in user mode</td>
</tr>
<tr>
<td></td>
<td>and kernel mode since the start of the task. A jiffy corresponds to a unit of CPU</td>
</tr>
<tr>
<td></td>
<td>time and gets used as a base of measurement. One second comprises 100 jiffies.</td>
</tr>
<tr>
<td>UTime</td>
<td>This counter displays the time, measured in jiffies, that a task has scheduled in</td>
</tr>
<tr>
<td></td>
<td>user mode.</td>
</tr>
<tr>
<td>VmData</td>
<td>This counter displays the virtual memory usage of the heap for the task in kilobytes</td>
</tr>
<tr>
<td></td>
<td>(KB).</td>
</tr>
<tr>
<td>VmRSS</td>
<td>This counter displays the virtual memory (Vm) resident set size (RSS) that is</td>
</tr>
<tr>
<td></td>
<td>currently in physical memory in kilobytes (KB). This includes the code, data, and</td>
</tr>
<tr>
<td></td>
<td>stack.</td>
</tr>
<tr>
<td>VmSize</td>
<td>This counter displays the total virtual memory usage for a task in kilobytes (KB).</td>
</tr>
<tr>
<td></td>
<td>It includes all code, data, shared libraries, and pages that have been swapped out:</td>
</tr>
<tr>
<td></td>
<td>Virtual Image = swapped size + resident size.</td>
</tr>
<tr>
<td>Wchan</td>
<td>This counter displays the channel (system call) in which the process is waiting.</td>
</tr>
</tbody>
</table>

Processor

The processor object provides information on different processor time usage in percentages. Table 5-56 contains information on processor counters.
The System object provides information on file descriptors on your system. Table 5-57 contains information on system counters.

Table 5-57  System

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allocated FDs</td>
<td>This counter represents the total number of allocated file descriptors.</td>
</tr>
<tr>
<td>Being Used FDs</td>
<td>This counter represents the number of file descriptors that are currently in use in the system.</td>
</tr>
<tr>
<td>Freed FDs</td>
<td>This counter represents the total number of allocated file descriptors on the system that are freed.</td>
</tr>
<tr>
<td>Max FDs</td>
<td>This counter represents the maximum number of file descriptors that are allowed on the system.</td>
</tr>
<tr>
<td>Total CPU Time</td>
<td>This counter represents the total time in jiffies that the system has been up and running.</td>
</tr>
<tr>
<td>Total Processes</td>
<td>This counter represents the total number of processes on the system.</td>
</tr>
<tr>
<td>Total Threads</td>
<td>This counter represents the total number of threads on the system.</td>
</tr>
</tbody>
</table>
TCP

The TCP object provides information on the TCP statistics on your system. Table 5-58 contains information on the TCP counters.

### Table 5-58 TCP

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Opens</td>
<td>This counter displays the number of times that the TCP connections made a direct transition to the SYN-SENT state from the CLOSED state.</td>
</tr>
<tr>
<td>Attempt Fails</td>
<td>This counter displays the number of times that the TCP connections have made a direct transition to the CLOSED state from either the SYN-RCVD state or the SYN-RCVD state, plus the number of times TCP connections have made a direct transition to the LISTEN state from the SYS-RCVD state.</td>
</tr>
<tr>
<td>Curr Estab</td>
<td>This counter displays the number of TCP connections where the current state is either ESTABLISHED or CLOSE-WAIT.</td>
</tr>
<tr>
<td>Estab Resets</td>
<td>This counter displays the number of times that the TCP connections have made a direct transition to the CLOSED state from either the ESTABLISHED state or the CLOSE-WAIT state.</td>
</tr>
<tr>
<td>In Segs</td>
<td>This counter displays the total number of segments that were received, including those received in error. This count only includes segments that are received on currently established connections.</td>
</tr>
<tr>
<td>InOut Segs</td>
<td>This counter displays the total number of segments that were sent and the total number of segments that were received.</td>
</tr>
<tr>
<td>Out Segs</td>
<td>This counter displays the total number of segments that were sent. This count only includes segments that are sent on currently established connections, but excludes retransmitted octets.</td>
</tr>
<tr>
<td>Passive Opens</td>
<td>This counter displays the number of times that TCP connections have made a direct transition to the SYN-RCVD state from the LISTEN state.</td>
</tr>
<tr>
<td>RetransSegs</td>
<td>This counter displays the total number of segments that were retransmitted because the segment contains one or more previously transmitted octets.</td>
</tr>
</tbody>
</table>

Thread

The Thread object provides a list of running threads on your system. Table 5-59 contains information on the Thread counters.

### Table 5-59 Thread

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>% CPU Time</td>
<td>This counter displays the thread share of the elapsed CPU time since the last update. This counter expresses the share as a percentage of the total CPU time.</td>
</tr>
<tr>
<td>PID</td>
<td>This counter displays the threads leader process ID.</td>
</tr>
</tbody>
</table>
Cisco Intercompany Media Engine Performance Objects and Alerts

This section provides information on new performance objects, and alerts for both the Cisco Unified Communications Manager server and the Cisco Intercompany Media Engine server.

This section contains the following information:

Cisco Intercompany Media Engine Server Objects

Performance Objects

The following performance objects are available on the Cisco Intercompany Media Engine server to support the Cisco Intercompany Media Engine feature.

- IME Configuration Manager, page 5-74
- IME Server, page 5-74
- IME Server System Performance, page 5-77

IME Configuration Manager

The IME Configuration Manager object provides information about the IME distributed cache certificate. Table 5-60 contains information on the Cisco IME configuration counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DaysUntilCertExpiry</td>
<td>This counter indicates the number of days that remain until the IME distributed cache certificate expires. You must replace the certificate before it expires. When the value of this counter falls below 14, an alert gets generated once every day until the value exceeds 14.</td>
</tr>
</tbody>
</table>

IME Server

The IME Server object provides information about the Cisco IME server. Table 5-61 contains information on the Cisco IME Server counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlockedValidationOrigTLSLimit</td>
<td>This counter indicates the total number of blocked validations that occurred because the TLSValidationThreshold was reached.</td>
</tr>
<tr>
<td>BlockedValidationTermTLSLimit</td>
<td>This counter indicates the total number of blocked validations that occurred because the TLSValidationThreshold was reached.</td>
</tr>
<tr>
<td>ClientsRegistered</td>
<td>This counter indicates the number of Cisco IME clients that are currently connected to the Cisco IME server.</td>
</tr>
</tbody>
</table>
Cisco Intercompany Media Engine Performance Objects and Alerts

Chapter 5    Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

Chapter 5      Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

Table 5-61  IME Server Counters

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMEDistributedCacheHealth</td>
<td>The counter indicates the health of the IME distributed cache. The following values may display:</td>
</tr>
<tr>
<td></td>
<td>• 0 (red)—Warns that the IME distributed cache is not functioning properly; for example, the Cisco IME cannot resolve issues after the network has been partitioned. In this case, validation attempts might fail. For example, the Cisco IME service is not connected to the network and is unable to reach the bootstrap servers. An alert gets generated once every hour until the value changes from red status.</td>
</tr>
<tr>
<td></td>
<td>• 1 (yellow)—Indicates that the Cisco IME network is experiencing minor issues, such as connectivity between bootstrap servers or other Cisco IME network issues. (Check the Cisco IME alarms to determine network issues.)</td>
</tr>
<tr>
<td></td>
<td>• 2 (green)—Indicates that the Cisco IME is functioning normally and is considered healthy.</td>
</tr>
<tr>
<td>IMEDistributedCacheNodeCount</td>
<td>The counter is an integer that indicates an approximation of the total number of nodes in the IME distributed cache. Since each physical Cisco IME server hosts multiple nodes, this counter does not directly indicate the number of physical Cisco IME servers that participate in the IME distributed cache. This counter can provide an indication of the health of the IME distributed cache; for example, a problem may exist with the IME distributed cache if an expected value displays on one day (for example, 300), but then on the next day, the value drops dramatically (for example, to 10 or 2).</td>
</tr>
<tr>
<td>IMEDistributedCacheQuota</td>
<td>Indicates the number of individual DIDs that can be written into the IME Distributed Cache, by Cisco Unified CMs attached to this IME server. This number is determined by the overall configuration of the IME Distributed Cache, and the IME license installed on the IME server.</td>
</tr>
<tr>
<td>IMEDistributedCacheQuotaUsed</td>
<td>Indicates the total number of unique DID numbers that have been configured, to be published via enrolled patterns for Intercompany Media Services, by Cisco Unified CMs currently attached to this IME server.</td>
</tr>
<tr>
<td>IMEDistributedCacheReads</td>
<td>This counter indicates the total number of reads that the Cisco IME server has attempted into the IME distributed cache. This number serves as an indicator of whether the Cisco IME server is functional; that is, whether the server is interacting with other nodes.</td>
</tr>
<tr>
<td>IMEDistributedCacheStoredData</td>
<td>This counter indicates the amount of IME distributed cache storage, measured in bytes, that this Cisco IME server provides.</td>
</tr>
<tr>
<td>IMEDistributedCacheStores</td>
<td>This counter indicates the total number of stores (published numbers) that the Cisco IME server has attempted into the IME distributed cache. This number serves as an indicator of whether the Cisco IME server is functional.</td>
</tr>
<tr>
<td>InternetBandwidthRecv</td>
<td>This counter measures the amount of downlink Internet bandwidth, in Kbits/s, that the Cisco IME server is consuming.</td>
</tr>
<tr>
<td>InternetBandwidthSend</td>
<td>This counter measures the amount of uplink Internet bandwidth that the Cisco IME server in Kbits/s is consuming.</td>
</tr>
</tbody>
</table>
### Chapter 5      Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

### Cisco Intercompany Media Engine Performance Objects and Alerts

#### TerminatingVCRs
This counter indicates the total Cisco IME voice call records (VCRs) that are stored on the Cisco IME server after receiving calls. You can use these records for validating learned routes.

#### ValidationAttempts
This counter indicates the total number of attempts that the Cisco IME server has made at performing a validation because the dialed number was found in the Cisco IME network. This counter provides an overall indication of system usage.

#### ValidationsAwaitingConfirmation
This counter indicates the total number of destination phone numbers that have been validated, but that are awaiting further calls to improve the security of the system. If you use a higher level of security for learning new routes, the Cisco IME server requires multiple successful validations for a route before that route is available for calls over IP. This counter tracks the number of successful validations that have not resulted in available IP routes.

#### ValidationsPending
This counter, which is an integer, indicates the number of scheduled validation attempts to retrieve a learned route. This value indicates the backlog of work for the Cisco IME service on the Cisco IME server.

An alert gets generated when the value rises either above the high watermark or falls below the low watermark. After the high watermark is reached, an alert gets sent immediately and then once an hour until the value falls below the high watermark. When the high watermark is reached, the Cisco IME service cannot clear the backlog of work prior to the expiration of data; this situation causes records to drop, and validation may not occur. To reduce the workload, add more Cisco IME servers that can share the workload.

#### ValidationsBlocked
This counter indicates the number of times that the Cisco IME service rejected a validation attempt because the calling party was not trusted; that is, the party was on a blacklist or not on a whitelist. This value provides an indication of the number of cases where a VoIP calls cannot happen in the future because of the blocked validation.

### Table 5-61 IME Server

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TerminatingVCRs</td>
<td>This counter indicates the total Cisco IME voice call records (VCRs) that are stored on the Cisco IME server after receiving calls. You can use these records for validating learned routes.</td>
</tr>
<tr>
<td>ValidationAttempts</td>
<td>This counter indicates the total number of attempts that the Cisco IME server has made at performing a validation because the dialed number was found in the Cisco IME network. This counter provides an overall indication of system usage.</td>
</tr>
<tr>
<td>ValidationsAwaitingConfirmation</td>
<td>This counter indicates the total number of destination phone numbers that have been validated, but that are awaiting further calls to improve the security of the system. If you use a higher level of security for learning new routes, the Cisco IME server requires multiple successful validations for a route before that route is available for calls over IP. This counter tracks the number of successful validations that have not resulted in available IP routes.</td>
</tr>
<tr>
<td>ValidationsPending</td>
<td>This counter, which is an integer, indicates the number of scheduled validation attempts to retrieve a learned route. This value indicates the backlog of work for the Cisco IME service on the Cisco IME server. An alert gets generated when the value rises either above the high watermark or falls below the low watermark. After the high watermark is reached, an alert gets sent immediately and then once an hour until the value falls below the high watermark. When the high watermark is reached, the Cisco IME service cannot clear the backlog of work prior to the expiration of data; this situation causes records to drop, and validation may not occur. To reduce the workload, add more Cisco IME servers that can share the workload.</td>
</tr>
<tr>
<td>ValidationsBlocked</td>
<td>This counter indicates the number of times that the Cisco IME service rejected a validation attempt because the calling party was not trusted; that is, the party was on a blacklist or not on a whitelist. This value provides an indication of the number of cases where a VoIP calls cannot happen in the future because of the blocked validation.</td>
</tr>
</tbody>
</table>
**IME Server System Performance**

The Cisco IME System Performance object provides information about performance on the Cisco IME server. Table 5-62 contains information on the Cisco IME server system performance counters.

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QueueSignalsPresent 1-High</td>
<td>This counter indicates the number of high-priority signals in the queue on the Cisco IME server. High-priority signals include timeout events, internal KeepAlive messages, internal process creation, and so on. A large number of high-priority events causes degraded performance of the Cisco IME service and results in slower or failed validations. Use this counter in conjunction with the QueueSignalsProcessed 1-High counter to determine the processing delay on the Cisco IME server.</td>
</tr>
<tr>
<td>QueueSignalsPresent 2-Normal</td>
<td>This counter indicates the number of normal-priority signals in the queue on the Cisco IME server. Normal-priority signals include call validations, IME distributed cache operations such as stores and reads, and so on. A large number of normal-priority events causes degraded performance of the Cisco IME service and may result in slower or failed validations or disruption to IME distributed cache connectivity. Use this counter in conjunction with the QueueSignalsProcessed 2-Normal counter to determine the processing delay on the Cisco IME server. Since high-priority signal must complete before normal priority signals begin to process, check the high-priority counters to accurately understand why a delay occurs.</td>
</tr>
<tr>
<td>QueueSignalsPresent 3-Low</td>
<td>This counter indicates the number of low-priority signals in the queue on the Cisco IME server. Low-priority signals include IME distributed cache signaling and other events. A large number of signals in this queue may disrupt IME distributed cache connectivity or other events.</td>
</tr>
<tr>
<td>QueueSignalsPresent 4-Lowest</td>
<td>This counter indicates the number of lowest-priority signals in the queue on the Cisco IME server. A large number of signals in this queue may disrupt IME distributed cache connectivity and other events.</td>
</tr>
<tr>
<td>QueueSignalsProcessed 1-High</td>
<td>This counter indicates the number of high-priority signals that the Cisco IME service processes for each one-second interval. Use this counter in conjunction with the QueueSignalsPresent 1-High counter to determine the processing delay for this queue.</td>
</tr>
<tr>
<td>QueueSignalsProcessed 2-Normal</td>
<td>This counter indicates the number of normal-priority signals that the Cisco IME service processes for each one-second interval. Use this counter in conjunction with the QueueSignalsPresent 1-High counter to determine the processing delay for this queue. High-priority signals are processed before normal-priority signals.</td>
</tr>
<tr>
<td>QueueSignalsProcessed 3-Low</td>
<td>This counter indicates the number of low-priority signals that the Cisco IME service processes for each one-second interval. Use this counter in conjunction with the QueueSignalsPresent 3-Low counter to determine the processing delay for this queue.</td>
</tr>
</tbody>
</table>
Chapter 5      Cisco Unified Real-Time Monitoring Tool Tracing, PerfMon Counters, and Alerts

Cisco Intercompany Media Engine Performance Objects and Alerts

The following alerts are available on the Cisco Intercompany Media Engine server to support the Cisco Intercompany Media Engine feature. For descriptions and default configuration settings, refer to the Cisco Intercompany Media Engine Installation and Configuration Guide.

- BannedFromNetwork
- IMEDistributedCacheCertificateExpiring
- IMEDistributedCacheFailure
- IMESdlLinkOutOfService
- InvalidCertificate
- InvalidCredentials
- MessageOfTheDay
- SWUpdateRequired
- TicketPasswordChanged
- ValidationsPendingExceeded
- CriticalAuditEventGenerated

Cisco Unified Communications Manager Server Objects

The following performance objects are available on the Cisco Unified Communications Manager server to support Cisco Intercompany Media Engine.

- IME Client, page 5-79
- IME Client Instance, page 5-80
### IME Client

The IME Client object provides information about the Cisco IME client on the Cisco Unified Communications Manager server. It contains information on the Cisco IME client counters.

#### Table 5-63  Cisco IME Client

<table>
<thead>
<tr>
<th>Counters</th>
<th>Counter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallsAccepted</td>
<td>This counter indicates the number of Cisco IME calls that the Cisco Unified Communications Manager received successfully and that the called party answered, resulting in an IP call.</td>
</tr>
<tr>
<td>CallsAttempted</td>
<td>This counter indicates the number of calls that the Cisco Unified Communications Manager received through Cisco IME. This number includes accepted calls, failed calls, and busy, no-answer calls. The counter increments each time that Cisco Unified Communications Manager receives a call through Cisco IME.</td>
</tr>
<tr>
<td>CallsReceived</td>
<td>This counter indicates the number of calls that Cisco Unified Communications Manager receives through Cisco IME. This number includes accepted calls, failed calls, and busy, no-answer calls. The counter increments on call initiation.</td>
</tr>
<tr>
<td>CallsSetup</td>
<td>This counter indicates the number of Cisco IME calls that Cisco Unified Communications Manager placed successfully and that the remote party answered, resulting in an IP call.</td>
</tr>
<tr>
<td>DomainsUnique</td>
<td>This counter indicates the number of unique domain names of peer enterprises that the Cisco IME client discovered. The counter serves as an indicator of overall system usage.</td>
</tr>
<tr>
<td>FallbackCallsFailed</td>
<td>This counter indicates the total number of failed fallback attempts.</td>
</tr>
<tr>
<td>FallbackCallsSuccessful</td>
<td>This counter indicates the total number of Cisco IME calls that have fallen back to the PSTN mid-call due to a quality problem. The counter includes calls initiated and calls received by this Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>IMESetupsFailed</td>
<td>This counter indicates the total number of call attempts for which a Cisco IME route was available but that were set up through the PSTN due to a failure to connect to the target over the IP network.</td>
</tr>
<tr>
<td>RoutesLearned</td>
<td>This counter indicates the total number of distinct phone numbers that the Cisco IME has learned and that are present as routes in the Cisco Unified Communications Manager routing tables. If this number grows too large, the server may exceed the per-cluster limit, and you may need to add additional servers to your cluster.</td>
</tr>
<tr>
<td>RoutesPublished</td>
<td>This counter indicates the total number of DIDs that were published successfully into the IME distributed cache across all Cisco IME client instances. The counter provides a dynamic measurement that gives you an indication of your own provisioned usage and a sense of how successful the system has been in storing the DIDs in the network.</td>
</tr>
<tr>
<td>RoutesRejected</td>
<td>This counter indicates the number of learned routes that were rejected because the administrator blacklisted the number or domain. This counter provides an indication of the number of cases where a VoIP call cannot happen in the future because of the blocked validation.</td>
</tr>
<tr>
<td>VCRUploadRequests</td>
<td>This counter indicates the number of voice call record (VCR) upload requests that the Cisco Unified Communications Manager has sent to the Cisco IME server to be stored in the IME distributed cache.</td>
</tr>
</tbody>
</table>
IME Client Instance

The IME Client Instance object provides information about the Cisco IME client instance on the Cisco Unified Communications Manager server. Table 5-64 contains information on the Cisco IME client instance counters.

Table 5-64 IME Client Instance

<table>
<thead>
<tr>
<th>Counter Description</th>
<th>Counter</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMEServiceStatus</td>
<td>This counter indicates the overall health of the connection to the Cisco IME services for a particular Cisco IME client instance (Cisco Unified Communications Manager). The following values may display for the counter: 0—Indicates an unknown state (which may mean that the Cisco IME service is not active). If the value specifies 0, an alert gets generated once per hour while the connection remains in the unknown state. 1—Indicates a healthy state; that is, the Cisco IME service is active, and the Cisco Unified Communications Manager has successfully established a connection to its primary and backup servers for the Cisco IME client instance, if configured. 2—Indicates an unhealthy state; that is, the Cisco IME service is active, but the Cisco Unified Communications Manager has not successfully established a connection to its primary and backup servers for the Cisco IME client instance, if configured.</td>
</tr>
</tbody>
</table>

Cisco Unified Communications Manager Server Alerts

The following alerts are available on the Cisco Unified Communications Manager server to support Cisco Intercompany Media Engine. For descriptions and default configuration settings, refer to the Cisco Intercompany Media Engine Installation and Configuration Guide.

- IMEDistributedCacheInactive
- IMEOverQuota
- IMEQualityAlert
- InsufficientFallbackIdentifiers
- IMEServiceStatus
- InvalidCredentials
- TCSPSetupToIMEFailed
- TLSConnectionToIMEFailed
This chapter describes the Cisco Unified Serviceability alarms and error messages and CiscoLog message format. Network alarms tracked by Cisco Unified Serviceability for Cisco Unified Communications Manager generate the error messages.

A History table lists Cisco Unified Serviceability error messages that have been added, changed, or removed beginning in Cisco Unified Communications Manager Release 7.0(1).

This chapter contains the following sections:
- Cisco Unified Serviceability Alarms and CiscoLog Messages, page 6-2
- Preconfigured System Alarm Notifications, page 6-19
- Preconfigured CallManager Alarm Notifications, page 6-32
- Emergency-Level Alarms, page 6-52
- Alert-Level Alarms, page 6-61
- Critical-Level Alarms, page 6-79
- Error-Level Alarms, page 6-92
- Warning-Level Alarms, page 6-202
- Notice-Level Alarms, page 6-303
- Informational-Level Alarms, page 6-326
- Debug-Level Alarms, page 6-405
- Obsolete Alarms in Cisco Unified Communications Manager Release 8.0(1), page 6-406
Cisco Unified Serviceability Alarms and CiscoLog Messages

Cisco Unified Serviceability alarms provide information on runtime status and the state of the system, so you can troubleshoot problems that are associated with your system. The alarm or error message information includes the application name, machine name, and recommended action and other critical information to help you troubleshoot.

You configure the alarm interface to send alarm information to multiple locations, and each location can have its own alarm event level (from debug to emergency). You can direct alarms to the Syslog Viewer (local syslog), SNMP traps, Syslog file (remote syslog), SDI trace log file, SDL trace log file (for Cisco Unified CM and CTIManager services only), or to all destinations.

You use the Trace and Log Central option in the Cisco Unified Real-Time Monitoring Tool (RTMT) to collect alarms that get sent to an SDI or SDL trace log file. To view the alarm information sent to the local syslog, use the SysLog Viewer in RTMT.

Note All the alarms are logged based on their severity and settings of alarm event level. For information on viewing the alarm configuration settings, refer to the Cisco Unified Serviceability Administration Guide.

CiscoLog Format

CiscoLog, a specification for unified logging in Cisco software applications, gets used in the Cisco Unified RTMT. It defines the message format when messages are logged into file or by using the syslog protocol. The output that is provided by Cisco software applications gets used for auditing, fault-management, and troubleshooting of the services that are provided by these applications.

Be aware that CiscoLog message format is compatible with one of the message formats that is produced by Cisco IOS Release 12.3 by using the syslog protocol when Cisco IOS Software is configured with the following commands:

- **service sequence-numbers**—A default sequence number that is produced by Cisco IOS. An additional sequence number can also be enabled with this command. This command forces sequence numbers to be shown in terminal output, but results in two sequence numbers in the syslog output. CiscoLog standardizes on a format with just one sequence number. Thus, the compliant Cisco IOS Software configuration occurs when the second number is disabled by using the **no service sequence-numbers** command.

- **logging origin-id hostname**—The CiscoLog HOST field remains consistent with that produced by the Cisco IOS Release 12.3 when configured with this command. This command does not get documented in the Cisco IOS Software documentation but is available in Cisco IOS Release 12.3. CiscoLog stays compatible with the results that Cisco IOS Software produces in this field.

- **service timestamps log datetime localtime msec show-timezone year**—The CiscoLog TIMESTAMP field remains consistent with the timestamp format produced by Cisco IOS Release 12.3 when configured with this command.

Note CiscoLog uses the same field delimiters as Cisco IOS Software Release 12.3.

The following topics are described in this section:

- Log File and Syslog Outputs, page 6-3
- Standard Syslog Server Implementations, page 6-4
Log File and Syslog Outputs

When CiscoLog messages are written directly into a log file by an application, each message is on a separate line. The line separator should be a standard line separator used on a given platform. On Windows, the line separator must be the sequence of carriage return and line feed characters (ASCII decimal values 13 and 10 often designated as “\r\n” in programming languages). On Solaris and Linux, the line separator is a single line feed character (ASCII decimal value 10 and in programming languages typically “\n”). Two line separators must never appear one after another, for example, you cannot have “\r\n\r\n” on Windows, but “\r\n” is fine because these two characters are a single line separator.

In practical terms, this means that applications should be careful when appending data to an existing log. In some cases an initial line break is required and in others not. For example, if application crashes when writing CiscoLog message, but before it wrote a line break to file, then when the application starts up, it should print an initial line break before printing the next message. An application can determine if an initial line break is necessary during startup by checking the last character sequence in the log file that will be used for appending.

CiscoLog message format is identical for messages written directly to a log file or those generated by using the syslog protocol with two minor exceptions. When CiscoLog messages are written directly into to a file they must be appended with line separators. When CiscoLog messages are sent by using the syslog protocol then the syslog RFC 3164 protocol PRI header must be prepended to each CiscoLog message.

The syslog PRI field encodes syslog message severity and syslog facility. The severity encoded in the PRI field must match the value of the CiscoLog SEVERITY field. Any syslog facility can be used regardless of the content of the message. Typically, a given application is configured to send all its messages to a single syslog facility (usually RFC 3164 facilities local 0 through local 7). Refer to RFC 3164 for details about how to encode the PRI field. Below is an example of a CiscoLog message with the syslog protocol PRI field <165> which encodes the severity level of notice (5) and facility value local4.

```
<165>11: host.cisco.com: Jun 13 2003 12:11:52.454 UTC: %BACC-5-CONFIG: Configured from console by vty0 [10.0.0.0]
```

Messages as shown in the example above can be sent to UDP port 514 if using RFC 3164 logging mechanism.

Syslog RFC 3164 provides additional guidelines for message content formatting beyond the PRI field. However, RFC 3164 is purely information (not on IETF standards track) and actually allows messages in any format to be generated to the syslog UDP port 514 (see section 4.2 of RFC 3164). The RFC provides observation about content structure often encountered in implementations, but does not dictate or recommend its use. CiscoLog format does not follow these observations due to practical limitations of the format defined in the RFC. For example, the time stamp is specified without a year, time zone or milliseconds while the hostname can only be provided without the domain name.

CiscoLog messages must remain unaltered when relayed. The PRI field is not part of a CiscoLog message, but rather a protocol header. It can be stripped or replaced if necessary. Additional headers or footers can be added to and stripped from the CiscoLog message for transport purposes.
Standard Syslog Server Implementations

Standard syslog server implementations can be configured to forward received log messages or to store the messages locally. Most syslog server implementations strip the PRI field from the received messages and prefix additional information to the message before storage. This additional information typically includes two extra fields: the local time stamp and the host identifier (IP or DNS name) of the server, which generated or relayed the message.

The following example of a CiscoLog message shown the output after being logged by the Solaris 8 syslog server:

```
Jun 13 12:12:09 host.cisco.com 11: host.cisco.com: Jun 13 2003 12:11:52.454 UTC:
%BACC-5-CONFIG: Configured from console by vty0 [10.0.0.0]
```

There is no standard that defines how syslog servers must store messages. Implementations vary greatly. CiscoLog only addresses the format in which messages are sent to the syslog server, not how they are stored by the server that receives them. Specifically, the format and presence of any additional header fields in syslog log files is outside of the scope of this specification.

Note

The CiscoLog specification recommends that the syslog server implementation store CiscoLog messages in exactly the same format as it receives them only stripping the PRI field and without any extra headers. This would provide an identical storage format for CiscoLog messages written directly to the log file by an application or logged through syslog protocol.

Clock Synchronization

It is important that the clocks of all hosts of a distributed application be synchronized with one authoritative clock. This can be accomplished by using protocols such as NTP. Clock synchronization is recommended because the time stamps in log messages are required in order to be able to reconstruct the correct sequence of events based on messages originating from multiple processes or multiple hosts. Clock drifts can still occur, but ongoing synchronization should reduce this issue to a minimum.

Multipart Messages

ASCII control characters are not permitted in any of the fields of CiscoLog message format. Control characters include characters such as line feed, form feed and carriage returns. This means that multi-line messages are not allowed unless to allow:

- Better presentation (for example, a stack trace)
- Fragmenting messages which exceed 800 octet limit

Multi-part CiscoLog message consists of a set of multiple valid CiscoLog messages. Messages are grouped together using a special tag key “part”, which identifies the part number and the sequence number of the original message.

All messages which are part of a multi-part message must have a “part” tag as well as identical values for the HOST, TIMESTAMP, APPNAME, SEVERITY fields and other TAG values. However, the sequence number of each message has to be incremented as usual.

Example of a multi-part message:

```
16: host.cisco.com: Jun 13 2003 23:11:52.468 UTC: %BACC-3-UNEXPECTED_EXCEPTION:
%[pname.orig=rdu][part=16.1/3]: Null pointer exception
17: host.cisco.com: Jun 13 2003 23:11:52.468 UTC: %BACC-3-UNEXPECTED_EXCEPTION:
%[pname.orig=rdu][part=16.2/3]: com.cisco.Source:123
```
In this example, the first message has part number 1 and its sequence number, 16, embedded in the part tag. Subsequent messages embed the sequence number of the first message part and provide their own part number. The trailing “/3” in each part tag value means that the message consists of three parts.

CiscoLog Message Format

The CiscoLog message format follows:

```
<SEQNUM>: <HOST>: <TIMESTAMP>: %<HEADER>: [TAGS: ]<MESSAGE>
```

All fields get separated by a single colon character (ASCII decimal value 58) and a single space character (ASCII decimal value 32). The HEADER field is also preceded by a percent character (ASCII decimal value 37).

The TIMESTAMP, HEADER and TAGS fields have internal formatting. Below is a complete format with details for TIMESTAMP and HEADER fields:

```
<SEQNUM>: <HOST>: [ACCURACY]<MONTH> <DAY> <YEAR>
<HOUR>:<MINUTES>:<SECONDS>.<MILLISECONDS> <TIMEZONE>:
%<APPNAME>-<SEVERITY>-<MSGNAME>: [TAGS: ]<MESSAGE>
```

All fields except for ACCURACY and TAGS are required.

The following example shows a CiscoLog message:

```
11: host.cisco.com: Jun 13 2003 23:11:52.454 UTC: %BACC-5-CONFIG: Configured from console by vty0 [10.10.10.0]
```

The following example shows the optional TAGS and ACCURACY fields in a CiscoLog message:

```
```

The values of the specific fields in the above example are as follows:

- SEQNUM – “12”
- HOST – “host.cisco.com”
- ACCURACY – “*”
- MONTH - “Jun”
- DAY – “13”
- YEAR – “2003”
- HOUR – “23”
- MINUTES – “11”
- SECONDS – “52”
- MILLISECONDS – “454”
- TIMEZONE – “UTC”
- APPNAME – “BACC”
- SEVERITY – “4”
- MSGNAME – “BAD_REQUEST”
• TAGS – “%[pname.orig=rdt][comp=parser][mac=1,6,aa:bb:cc:11:22:33][txn=mytxn123]”
• MESSAGE – “Bad request received from device [1,6,aa:bb:cc:11:22:33]. Header missing.”

Message Length Limit

The maximum length of a complete CiscoLog message must not exceed 800 octets. The term octet is used for 8-bit data type instead of byte because byte is not 8 bits on some platforms. The words “character” and “octet” are not synonyms in parts of this specification because in places where internationalization is supported a single character may need to be represented with multiple octets. This limit is dictated by RFC 3164. The limit of 1024 octets reserves some extra space for syslog forwarding headers and/or fields that may be formalized in later specifications.

When CiscoLog message includes the syslog PRI field, then the combined CiscoLog messages and PRI field length must not exceed 805 octets.

SEQNUM Field

The SEQNUM field contains a sequence number, which can be used to order messages in the time sequence order when multiple messages are produced with the same timestamp by the same process. The sequence number begins at 0 for the first message fired by a process since the last startup and is incremented by 1 for every subsequent logging message originated by the same process. Every time the application process is restarted, its sequence number is reset back to 0. The sequence number of each message must be in the exact order in which messages are fired/logged by the application.

This may mean that in a multi-threaded application there must be some kind of synchronization to ensure this and another consideration may have to be made for Java applications that have some native (C) code in JNI. If log messages originate in both native and Java parts of the same process, the implementation needs to be synchronized to use the same sequence number counter across the two process parts and to fire messages in the order of sequence numbers.

The maximum numeric value of the SEQNUM field is 4,294,967,295 at which point the counter must be reset back to 0. The maximum positive value of a 32-bit unsigned integer as used in Cisco IOS. Cisco IOS uses ulong for the sequence number counter and ulong is a 32-bit unsigned integer on all current Cisco IOS platforms including mips, ppc, and 68k.

Sequence numbers are process specific. If application architecture has multiple application processes on a single host, which share a single logging daemon, the sequence number still has to be process-specific. Thus, each process has its own sequence number which it increments.

Sequence numbers also help detect lost messages. Therefore, sequence numbers cannot be skipped. In other words, a message must be produced for every number in the sequence order.

HOST Field

The HOST field identifies the system originating the message with a Fully Qualified DNS Name (FQDN), hostname or an IPv4/IPv6 address. If the FQDN or hostname is known, one of the two has to appear in the HOST field. It is expected that in most deployments the hostname is sufficient. However, if a deployment spans multiple domains, then using FQDNs is recommend. If an application is expected to be deployed in both scenarios, then it is recommended that the application default to the FQDNs, but make it a configurable option.

If neither FQDN nor hostname can be identified, then the IP address of the host must be used. If the IP address cannot be identified, then a constant “0.0.0.0” (without quotes) must appear in place of the HOST field.
Note

With regards to the compliance with Cisco IOS format. Cisco IOS Release 12.3 supports producing hostname, IP address, or any user-defined string in the HOST field. If it is configured to provide a hostname and it is not set on the device, it will use a string such as “Router.”

The length of the HOST field must not exceed 255 octets.

FQDN & Hostname

If multiple FQDNs or hostnames are known for a given system, applications must use the primary FQDN/hostname or an arbitrary one if no primary is designated. However, applications must use the same HOST field value until some relevant configuration change takes place. In other words, the FQDN/hostname value should not arbitrarily change from message to message if system is configured with multiple FQDNs/hostnames.

Only printable US ASCII characters (those with decimal values 32-126) and foreign language characters are allowed in the HOST field when encoding an FQDN or hostname. The appropriate character set and encoding for HOST should be compliant with RFC 1123 / STD-3.

The acceptable character set per these standards includes US ASCII letters, numbers, dash and dot separator characters (although not starting or ending with a dash). The reason that these are only recommendations of adhering to these standards is that, in practice, many hosts do not follow the convention and use characters such as underscore in the hostname. However, the HOST field cannot contain a character sequence of “: ” (colon and space) as this sequence is used as a field delimiter in the CiscoLog format.

Foreign language characters outside of the printable US ASCII characters have to be encoded according to internationalization rules.

Use of non-printable (control) ASCII characters is not allowed in the HOST field. Control characters include characters with ASCII decimal values 0-31 and 127. If an application provides a CiscoLog-compliant library with a host string, which includes one or more control characters, the logging library must do the following. If the horizontal tab character (ASCII decimal value 9) is encountered, it must be replaced with one or more space characters (ASCII decimal value 32). Eight spaces per tab are recommended because this is a convention on most Unix and Windows platforms. Other control characters must each be replaced with a question mark character (ASCII decimal value 63).

While DNS is letter-case agnostic, CiscoLog places an additional recommendation of using only lower-case characters in the HOST field for ease of readability. The use of the trailing dot at the end of the FQDN is optional. The following examples are valid HOST fields:

- host123
- host-123
- host123.cisco.com
- host123.cisco.com.

IP Addresses

The IP address value used in the HOST field can be either an IPv4 or IPv6 address. If a device has multiple IP addresses, the primary IP address of the device must be used regardless of the interface through which the CiscoLog message is sent to syslog server. If no primary IP address is designated, a fixed/static IP address is preferred to a dynamically assigned one. If multiple static IP addresses exist, any one can be used, but it must be used consistently in all messages until a relevant configuration event occurs on the system.
IPv4 Address—IPv4 address should be represented in dot notation “x.x.x.x”, where x is a decimal value from 0 to 255 encoded as ASCII text. If an IP address is unknown, “0.0.0.0” (without quotes) must be used as a place holder. Examples of valid IPv4 addresses are 0.0.0.0 and 212.1.122.11.

Below is an example of a message with an IPv4 address in the HOST field:

```
11: 212.1.122.11: Jun 13 2003 23:11:52.454 UTC: %BACC-3-BAD_REQUEST: Bad request received from device [1.2.3.4]. Missing header.
```

Below is an example of a CiscoLog message when FQDN, hostname or IP are all unknown:

```
11: 0.0.0.0: Jun 13 2003 23:11:52.454 UTC: %BACC-3-BAD_REQUEST: Bad request received from device [1.2.3.4]. Missing header.
```

IPv6 Address—IPv6 address representation must follow conventions outlined in RFC 3513, sections 2.2.1, 2.2.2 and 2.2.3. Specifically, all three conventions are supported. Both lower-case and upper-case letters can be used in the IPv6 address, but the lower-case letters are recommended. If an IP address is unknown, “0.0.0.0” (without quotes) should be used as the IP address. Examples of valid IPv6 addresses:

- 1080:0:0:800:ba98:3210:11aa:12dd (full notation)
- 1080::800:ba98:3210:11aa:12dd (use of “::” convention)
- 0:0:0:0:0:13.1.68.3 (last 4 octets expanded as in IPv4)
- 0.0.0.0 (unknown FQDN, hostname and IP address)

Below is an example of a message with an IPv6 address in the HOST field:

```
```

TIMESTAMP Field

The TIMESTAMP field provides date with year, time with milliseconds and a time zone identifier in the following format:

```
[ACCURACY]<MONTH> <DAY> <YEAR> <HOUR>:<MINUTES>:<SECONDS>.<MILLISECONDS> <TIMEZONE>
```

Below are several examples of valid time stamps:

```
Jun 13 2003 23:11:52.454 UTC
Jun 3 2003 23:11:52.454 UTC
Jun 22 2003 05:11:52.525 -0300
*Feb 14 2003 01:02:03.005 EST
```

In some cases, it is possible that a device may not have the knowledge of the date and/or time due to hardware or software limitations. In such circumstances, the following string must be produced in the TIMESTAMP field: “--- 00 0000 00:00:00.000 ---”. Below is an example of a CiscoLog message from a device which has no knowledge of date and/or time:

```
11: host.domain.com: --- 00 0000 00:00:00.000 ---: %BACC-3-BAD_REQUEST: Bad request received from device [1.2.3.4]. Missing header.
```

Devices which are not aware of their clock, may choose to provide an uptime as a relative measure of time. If device is capable of providing uptime, it is recommended that it does so in substitution for unavailable time stamp. If uptime is provided it must be provided with a standard uptime tag as outlined in the CiscoLog Standard Tags specification.

Table 6-1 details each field specification.
### Table 6-1 TIMESTAMP Field Specifications

<table>
<thead>
<tr>
<th>Field</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCURACY</td>
<td>This is an optional field. If present, it must be either a single asterisk character (ASCII decimal value 42), or a single dot character (ASCII decimal value 46). No separator character is used after this field. This field indicates the status of clock synchronization. Cisco IOS uses a special convention for time prefixes to indicate the accuracy of the time stamp. If dot character appears before the date, it means that the local time was synchronized at some point via NTP, but currently no NTP servers are available. The asterisk character in front of the date means that the local time is not authoritative, i.e. NTP servers are not setup. CiscoLog supports the use of this convention, but does not require it. If an application is integrated with NTP client software, and knows that its time is out of sync, then it can optionally prefix the message with asterisk character. However, because applications may choose not to use this scheme, the lack of &quot;.&quot; or &quot;*&quot; in CiscoLog messages should not be interpreted to mean that the local time is synchronized.</td>
</tr>
<tr>
<td>MONTH</td>
<td>Must be one of the following three-character month designations followed by a single space (ASCII decimal value 32) as a delimiter character: Jan, Feb, Mar, Apr, May, Jun, Jul, Sep, Oct, Nov or Dec.</td>
</tr>
<tr>
<td>DAY</td>
<td>Must consist of two characters. If day is a single digit, it must be prefixed with a single space character. The acceptable range of values is from 1 to 31. The day value must be followed by a single space as a delimiter character.</td>
</tr>
<tr>
<td>YEAR</td>
<td>Must consist of exactly 4 digit characters followed by a space as a delimiter character.</td>
</tr>
<tr>
<td>HOUR</td>
<td>Must consist of exactly two number characters. The hour value is based on a 24-hour clock. Values range from 00 to 23. If hour value is a single digit, it must be prefixed with a single zero character. The hour value must be followed by a single colon as a delimiter character.</td>
</tr>
<tr>
<td>MINUTES</td>
<td>Must consist of exactly two number characters. Values range from 00 to 59. If minute value is a single digit, it must be prefixed with a single zero character. The minutes value must be followed by a single colon as a delimiter character.</td>
</tr>
<tr>
<td>SECONDS</td>
<td>Must consist of exactly two number characters. Values range from 00 to 59. If seconds value is a single digit, it must be prefixed with a single zero character. The seconds value must be followed by a period as a delimiter character.</td>
</tr>
<tr>
<td>MILLISECONDS</td>
<td>Must consist of exactly 3 digit characters. Values range from 000 to 999. If milliseconds value is less than 3 digits in length it must be prefixed with extra zeros to make it a 3-character field. The milliseconds value is followed by a space as a delimiter character.</td>
</tr>
</tbody>
</table>
The HEADER field has the following format:

`<APPNAME>-<SEVERITY>-<MSGNAME>`

A single dash character (ASCII decimal value 45) serves a separator for the three fields.

### APPNAME Field

The APPNAME field in the HEADER defines the name of the application producing the message. Cisco IOS uses FACILITY in place of APPNAME that names the logical component producing the message. Cisco IOS 12.3 defines approximately 287 facilities for 3950 messages. Example of some easily

### TIMESTAMP Field Specifications (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIMEZONE</td>
<td>Must consist of at least one, but no more than 7 characters in the following ASCII decimal value range: 32-126. The value must not include a combination of colon-space-percent of characters – “%:“ (ASCII decimal values 58, 32, 37) – as this character combination is reserved as a field delimiter that follows the time stamp. There is no standard set of acronyms for time zones. A list of common time zone acronyms and corresponding time offsets from UTC is provided in the UTC specification. Uppercase letters are recommended for time zone acronym values. CiscoLog recommends the use of time offset instead of time zone identifier in this field. The offset, if provided, must follow the following format “-hhmm” or “+hhmm” to indicate hour and minute offset from UTC. In this format time zone field must always contain 5 characters, with the last 4 characters being constrained to numbers only. Unlike a textual time zone identifier, this format provides a specific time offset from universal standard time. Cisco IOS Release 12.3 supports any 7-character string as a time zone identifier, so it can be configured in a way which is compatible with this recommendation. Multiple messages may and sometimes must be produced with exactly the same time stamp. This can happen naturally on a non-preemptive operating system or may need to be deliberately induced as in the case of multi-part messages. Sequence numbers then become helpful for establishing message order. Time stamp should always be accurate to the millisecond unless it can significantly hinder performance of the application. In either case, applications must always provide the administrator with an option to output messages with exact time stamp in milliseconds. If an application uses time stamp with accuracy to the second (instead of a millisecond), it must put the last known milliseconds value or 000 in place of the milliseconds. Whatever convention is chosen by the application, it should be followed consistently.</td>
</tr>
</tbody>
</table>

---

1. Neither Cisco IOS nor CiscoLog define a standard set of time zone acronyms because there is no single established standard.
recognizable facilities: AAAA, SYS, ATM, BGP, CRYPTO, ETHERNET, FTPSERVER, CONFIG_1, IP, ISDN, RADIUS, SNMP, SYS, TCP, UBR7200, X25. A complete list of defined facilities is available in Cisco IOS documentation at http://.

Outside of the Cisco IOS, there can be multiple applications on the same host originating log messages. Therefore, it is necessary that APPNAME field identify the specific application. Additional source identifiers are available in the HOST field as well as various standard TAGS field values (pname, pid, comp, etc).

The APPNAME field must consist of at least two uppercase letters or digits and may include underscore characters. More precisely, the acceptable character set is limited to characters with the following ASCII decimal values: 48-57 (numbers), 65-90 (upper-case letters) and 95 (underscore).

The length of the APPNAME field must not exceed 24 characters.

Application names cannot conflict with other Cisco software applications and with Cisco IOS facilities.

On the Solaris platform, it is recommended (not required) that the application name values used in the APPNAME field be consistent with those used for the application installation package name, only in upper case and without the CSCO prefix. For example, an application registering as “CSCObacc” on Solaris should use “BACC” as the value of the APPNAME field.

Some applications may choose to specify a version as part of the APPNAME field. This is acceptable and may be useful in cases where the meaning of certain messages is redefined from one release to another. For example, an APPNAME value could be “BACC_2_5” for BACC version 2.5. The use the version within an application name is optional and may be introduced by applications in any release.

**SEVERITY Field**

The SEVERITY field is a numeric value from 0 to 7, providing eight different severities. The severities defined below match Cisco IOS severity levels. They are also standard syslog severities.

It is important that messages use the correct severity. An error in a certain component may be severe as far as the component is concerned, but if the overall application handles it gracefully, then the severity may be lower for the application as a whole. Table 6-2 lists guidelines that should be followed in determining the severity of a message.

**Table 6-2 Name and Severity Level and Descriptions in Error Messages**

<table>
<thead>
<tr>
<th>Name/Severity Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emergency (0)</td>
<td>System or service is unusable. Examples:</td>
</tr>
<tr>
<td></td>
<td>• Service repeatedly fails to startup</td>
</tr>
<tr>
<td></td>
<td>• System ran out of disk space while disk space is essential for this system to operate</td>
</tr>
<tr>
<td></td>
<td>• Application requires root privileges to run but does not have them</td>
</tr>
<tr>
<td>Alert (1)</td>
<td>Action must be taken immediately. Examples:</td>
</tr>
<tr>
<td></td>
<td>• Application is about to run out of licenses</td>
</tr>
<tr>
<td></td>
<td>• Application is about to run out of disk space</td>
</tr>
<tr>
<td></td>
<td>• Too many unauthorized access attempts detected</td>
</tr>
<tr>
<td></td>
<td>• Denial of service attack is detected</td>
</tr>
</tbody>
</table>
Critical (2)  Critical condition. Similar to alert, but not necessarily requiring an immediate action. Examples:
  • Received an invalid authentication request
  • Service crashed due to an error that could not be handled, like an out of memory condition, (provided it has a watchdog process to restart it, it does not necessarily require immediate action)
  • Unexpected code error that could not be handled

Error (3)  An error condition, which does not necessarily impact the ability of the service to continue to function. Examples:
  • Problem parsing/processing a particular request which does not prevent the application from handling other requests
  • Unexpected, but handled code exception

Warning (4)  A warning about some bad condition, which is not necessarily an error. Examples:
  • Lost network connection to some resource
  • Timed out waiting for a response

Notice (5)  Notifications about system-level conditions, which are not error conditions. Examples:
  • Configuration was updated (not audit level information)
  • Process has started
  • Process is shutting down gracefully on request

<table>
<thead>
<tr>
<th>Name/Severity Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical (2)</td>
<td>Critical condition. Similar to alert, but not necessarily requiring an immediate action. Examples:</td>
</tr>
<tr>
<td></td>
<td>• Received an invalid authentication request</td>
</tr>
<tr>
<td></td>
<td>• Service crashed due to an error that could not be handled, like an out of memory condition, (provided it has a watchdog process to restart it, it does not necessarily require immediate action)</td>
</tr>
<tr>
<td></td>
<td>• Unexpected code error that could not be handled</td>
</tr>
<tr>
<td>Error (3)</td>
<td>An error condition, which does not necessarily impact the ability of the service to continue to function. Examples:</td>
</tr>
<tr>
<td></td>
<td>• Problem parsing/processing a particular request which does not prevent the application from handling other requests</td>
</tr>
<tr>
<td></td>
<td>• Unexpected, but handled code exception</td>
</tr>
<tr>
<td>Warning (4)</td>
<td>A warning about some bad condition, which is not necessarily an error. Examples:</td>
</tr>
<tr>
<td></td>
<td>• Lost network connection to some resource</td>
</tr>
<tr>
<td></td>
<td>• Timed out waiting for a response</td>
</tr>
<tr>
<td>Notice (5)</td>
<td>Notifications about system-level conditions, which are not error conditions. Examples:</td>
</tr>
<tr>
<td></td>
<td>• Configuration was updated (not audit level information)</td>
</tr>
<tr>
<td></td>
<td>• Process has started</td>
</tr>
<tr>
<td></td>
<td>• Process is shutting down gracefully on request</td>
</tr>
</tbody>
</table>
If an application uses a default severity level to determine which messages should be logged, then it is recommended that this level be set at 5 (notice). This ensures that all messages of severity 5 or higher are logged by default.

**MSGNAME Field**

The MSGNAME field of the HEADER uniquely identifies the message within the context of a given APPNAME. A fixed severity and logical meaning is associated with a specific MSGNAME within a specific APPNAME. In other words, the same message name cannot appear with different severity or a completely different logical meaning for the same APPNAME value even if the message is originated by a different process.

Message names are only unique within a given application (a given APPNAME value) unless the message is one of the standard messages. Thus, applications interpreting CiscoLog messages should be careful not to assume that a message with a given name has the same meaning for all applications that may use this message name. Indeed, if the message is not one of the standard messages, it may have a different severity and meaning in a different application.

The MSGNAME field must consist of at least two characters. Acceptable characters are limited to the following ASCII decimal values: 48-57 (numbers), 65-90 (upper-case letters) and 95 (underscore). While IOS allows lower-case letters as well, the vast majority of IOS messages use only the upper-case letters. In order to be consistent with established conventions we opted to restrict the character set to upper-case letters, numbers and underscore characters.

---

**Table 6-2 Name and Severity Level and Descriptions in Error Messages (continued)**

<table>
<thead>
<tr>
<th>Name/Severity Level</th>
<th>Description</th>
</tr>
</thead>
</table>
| Informational (6)   | Informational messages are distinguished from notification in that they provide information for internal flows of the application or per-request information instead of system-wide notifications. Informational messages are used for troubleshooting by users who are familiar with the basic flows of the application. Examples:  
  - Request received  
  - Request was parsed successfully  
  - Request being processed  
  - Response sent back  
  - Acknowledgement received  
  - Detailed audit information |
| Debug (7)           | Debugging messages are similar to informational messages, but provide more detail and require the user to have better knowledge of system internal processing. These messages are typically reserved for very advanced users or Cisco technical support. Examples:  
  - Complete details for a request packet  
  - Internal state machine state changes  
  - Internal profiling statistics  
  - Internal events |
Both numeric-only or alphanumeric message names are acceptable. However, per IOS convention, it is recommended that a user-friendly alphanumeric label be preferred to a numeric-only label. For example, “NO_MEMORY” message name is preferred to a “341234” identifier.

A special tag mid is defined in the CiscoLog Standard Tags specification for identifying a numeric id corresponding to a message name. This tag can be used to provide a numeric message is in addition to the MSGNAME. When this tag is used, a given MSGNAME must always correspond to a single message id value. CiscoLog defines mid tag values for each standard message.

The length of the MSGNAME field must not exceed 30 characters, but most message names should be more concise. MSGNAME value may not conflict with the names defined in this standard.

A separate message name must be defined for each logically different message. In other words, while the message text for a given message name can vary by virtue of some substitutable parameters, logically different messages must have different message names.

The following is an example of correct use of message name:

11: host.cisco.com: Jun 13 2003 23:11:52.454 UTC: %BACC-4-CONNECTION_LOST: %[pname.orig=rdm]: Server lost connection to host [1.1.1.1]
12: host.cisco.com: Jun 13 2003 23:11:52.458 UTC: %BACC-4-CONNECTION_LOST: %[pname.orig=rdm]: Server lost connection to host [2.2.2.2]

Notice that while the IP address of the host changes, it is still logically the same type of message. The following is an example of an INCORRECT use of the message name:

15: host.cisco.com: Jun 13 2003 23:11:52.455 UTC: %BACC-4-CONNECTION: %[pname.orig=rdm]: Server lost connection to host [2.2.2.2]
16: host.cisco.com: Jun 13 2003 23:11:52.468 UTC: %BACC-4-CONNECTION: %[pname.orig=rdm]: Server re-established connection to host [2.2.2.2]

The use of a single message name for two different events in the above example is wrong and unacceptable. This is referred to as a “catch-all” message name and they must be avoided. Another extreme example is defining a message named “ERROR” and providing all error log messages under the same message name. This defeats the purpose of having the message name field, which is to enable external filtering of messages or easily trigger actions.

The only exception to the “no-catch-all” rule is when message cannot be identified ahead of time with anything better than a generic description or the users will not benefit from distinguishing the various subtypes of the message.

Although some applications may choose to do so, there is generally no need to define a separate message name for all debugging messages because debugging messages are not intended for automated filtering and action triggering based on message name. The sheer number of debugging messages and the highly dynamic nature of what is produced in them makes it very hard to define separate messages.

This specification proposes establishing a mailing list that could be used by groups for consulting purposes when in doubt about how to define certain messages. Currently, the mailing list alias used for this purpose is “cmn-logging”.

**TAGS Field**

The TAGS field is optional in the message format. It provides a standard mechanism for applications to provide structured content in the form of key-value pairs which can be used to categorize or filter a set of messages externally.

Tags can be used to identify virtual logging channels. A set of messages flagged with the same tag can later be grouped together. For example, an application may flag messages belonging to a particular thread by supplying the corresponding tag. This would then allow filtering and viewing messages based on threads.
Virtual logging channels can also be established across multiple applications. For example, if all applications could tag requests from a device with device id (mac, ip, etc), then it would be easy to filter all messages related to that device even though it communicates with multiple components.

Each application may define its own set of supported tags. A single tag consists of key and value pair separated by the equals sign and surrounded by square bracket characters as in the following format: [KEY=VALUE]. This is an example of a valid tag key-value pair [ip=123.23.22.22].

The TAGS field is prefixed with a percent character (ASCII decimal value 37) and ends with a sequence of colon and space characters (ASCII decimal values 58 and 32). When multiple tags are assembled together, no characters should appear between the tags as separators. The following example has a complete CiscoLog message with four tags:

```
12: host.cisco.com: Jun 13 2003 23:11:52.454 UTC: %BACC-4-BAD_REQUEST:
```

If TAGS field is missing, the percent character prefix and the trailing colon and space must be omitted. Thus, when the TAGS field is missing, the HEADER and MESSAGE fields must be separated by just a single colon and a space which follows the HEADER field. For example:

```
```

Multiple tags with the same tag key can be provided in the same message. This essentially provides the capability for handling multi-valued keys. Below is an example of a message produced from a device which has two IP addresses where the application chose to provide both IP addresses in the TAGS field as well as the process name:

```
12: host.cisco.com: Jun 13 2003 23:11:52.454 UTC: %BACC-4-BAD_REQUEST:
  %[pname.orig=rdu][ip.orig=1.1.1.1][ip.orig=1.1.1.2]: Bad request received from device [1,6,aa:bb:cc:11:22:33]. Missing header.
```

Any number of tags can be provided in a given message. The only limit is the overall length limit of the CiscoLog message of 800 octets.

If multiple tags are present, it is recommended that they appear in the alphanumeric order of the keys. This insures that tags are always produced in the same order. However, a different order may be chosen by an application if the order of tags is used to communicate some semantic value.

**Tag Keys**

Tag key must contain at least one character. The characters are limited to ASCII characters with decimal values 48-57 (numbers), 65-90 (upper-case letters), 95 (underscore), 97-122 (lower case letters). Use of lower-case letters is recommended. There is no strict limit on tag key length, although a general message limit of 800 octets applies and dictates that one should attempt to define short tag key names.

**Tag Semantic Extensions**

In some cases, a tag can have a standard value syntax, but different meaning depending on the content in which it is used. Tag semantic extensions are used to differentiate the contextual meaning of tags. The semantic extension tags are created by appending the tag key with a single dot character (ASCII decimal value 46) and a text string consisting of characters from a proper character set.

For example, an “ip” tag defines syntax for an IP address representation, but no semantic value. An “ip” tag found in a CiscoLog message generally means only that this IP address is somehow related to the message. In some cases, such vague association is sufficient. However, sometimes, communicating semantic value could be useful.
A message may have two IP address tags associated with it, for example, from and to IP addresses. In this case, using tags “ip.from” and “ip.to” would communicate both the syntax of the tags and some semantic value. Another example, is a standard tag “ip.orig”, which specifies the IP address of the host which originated the message. The following is an example of all three tags appearing together:

[ip.from=1.1.1.1][ip.to=2.2.2.2][ip.orig=123.12.111.1]

Multiple levels of semantic extension tags are allowed with each extension providing meaning that is more specific. For example, tag key “ip.to.primary” is valid and could mean the primary IP address of the destination host.

The semantic value is much harder to standardize than the syntax because there can an infinite number of meanings for a given value depending on the context. Thus, it is anticipated that defining tag semantics extensions will be largely application specific.

Tag Values
Tag values may contain zero or more characters. The empty (zero characters) value is interpreted as unknown or undetermined value. The value must only include printable US ASCII characters (those in the ASCII decimal value range 32-126) and foreign language characters.

There is a restriction on the use of three characters: “[”, “]” and “\”. The bracket characters (ASCII decimal values 91 & 93) must be escaped with a back slash character (ASCII decimal value 92). This helps to avoid confusion with the brackets that signify the start/end of the tag. Thus, when the tag value needs to represent characters “[“ or “]”, a sequence of “\[“ or “\]” is used instead respectively. When the escape character itself needs to be represented in the tag value, then instead of the “\” character a sequence of “\\” is used.

Use of non-printable (control) ASCII characters is not allowed in the TAG value field. Control characters include characters with ASCII decimal values 0-31 and 127. If application provides to a CiscoLog-compliant library a tag value string, which includes one or more control characters, the logging library must do the following. If the horizontal tab character (ASCII decimal value 9) is encountered, it must be replaced with one or more space characters (ASCII decimal value 32). Eight spaces per tab are recommended because this is a convention on most Unix and Windows platforms. Other control characters must each be replaced with a question mark character (ASCII decimal value 63). Technically, we only need to require escaping a closing bracket. However, requiring escaping both open and closing brackets simplifies parser code and provides for a more consistent display in raw form.

There is no strict limit on tag value length; although a general message length limit of 800 octets applies and dictates that one must be conservative.

Tag Guidelines
The TAGS field is optional in the CiscoLog message format. Tags do not replace substitutable parameters in the message body. Tags merely provide an additional way to identify and categorize messages.

Since tags are optional, they can be enabled or disabled by the application/user as required. There is no requirement for the same message to always be produced with the same set of tags. If the application supports a given tag, it does not necessarily mean that it must always produce it. This can be configurable. Indeed, it is recommended that applications provide the administrator with at least limited control over which tags get produces.

Application developers have a choice as to what information to make available in the tags and what in the message body. In some cases, the information may be duplicated between the two. This is acceptable.
The general guideline is to put all required information in the message body and make appropriate information available via tags. In other words, the message should provide sufficient meaning even when all tags are disabled. Tags merely provide additional useful information and a way to present it in a standard, easily filtered, form.

The following are two valid examples of a message where both the message and the message tags contain a MAC address. Example with tags disabled:

```
```

In the above example, the MAC address appears as part of the message field – it is not a tag. In the following example, the tags are enabled. Even though MAC address is duplicated between the tag and the message, it is acceptable.

```
```

**Process Identification Tag**

One of the standard tags, `pname.orig`, is used to identify the logical process name which originates the message. Any application that seeks to provide originating process information must do so using the “`pname.orig`” tag.

This tag is extremely valuable in addition to information in the `APPNAME` field because some applications consist of multiple processes, each of which may originate logging messages. It is recommended that any application which consists of multiple processes always provide the “`pname.orig`” tag.

**MESSAGE Field**

The `MESSAGE` field provides a descriptive message about the logging event. This field may consist of one or more characters. The character set is limited to printable US ASCII characters (ASCII decimal values 32-126) and foreign language characters.

Use of non-printable (control) ASCII characters is not permitted in the MESSAGE field. Control characters include characters with ASCII decimal values 0-31 and 127. If application provides a CiscoLog-compliant library with message string, which includes one or more control characters, the logging library must do the following. If the horizontal tab character (ASCII decimal value 9) is encountered, it must be replaced with one or more space characters (ASCII decimal value 32). Eight spaces per tab are recommended because this is a convention on most Unix and Windows platforms. Other control characters must each be replaced with a question mark character (ASCII decimal value 63).

The maximum length of the `MESSAGE` field is constrained only by the maximum length of the entire message. The maximum length of the CiscoLog message must not exceed 800 octets. Another practical limitation is a potentially highly variable length of the `TAGS` field.

Message text may contain substitutable parameters, which provide necessary details about the message. For example, the IP address in the following example is a substitutable parameter.

```
```

It is recommended (but not required) that substitutable parameters be surrounded by bracket characters “`[“ and `”]` as in the above example. It is further recommended that the message text and values of substitutable parameters do not include bracket characters. When it is not possible to avoid brackets...
characters in the values of substitutable parameters, it is recommended that the value at least does not
include unbalances brackets (like an opening bracket without a closing one). When these
recommendations are followed, it would be possible to programmatically extract substitutable parameter
values out of a CiscoLog message. However, this recommendation is not a strict requirement.

Message text should be spell-checked. Editorial review is recommended. This includes all messages that
can be seen by the customers, even debugging messages.

If the first word of the message is an English word, the first letter should be capitalized. Single sentence
messages do not require a period at the end.

### Internationalization

Foreign language characters are defined as characters with ASCII decimal values 0-126. Foreign
language characters are supported in the HOST field, the value part of the TAGS field and the
MESSAGE field.

Foreign language characters must be encoded using the Unicode standard UTF-8. UTF-8 provides
encoding for any language without requiring the application to know local encoding/decoding rules for
a particular language. In fact, the application encoding the message does not even need to know the
language of the message. UTF-8 can encode any Unicode character.

UTF-8 encodes US ASCII characters exactly as they would normally be encoded in a 7-bit ASCII
convention. This means that applications interpreting CiscoLog messages can assume that entire
messages are encoded in UTF-8. On the other hand, applications producing CiscoLog messages can
encode the entire message using US-ASCII 7-bit convention if they are known not to support foreign
languages in their products.

Since UTF-8 can encode characters in any language, it is possible to mix and match languages. For
example, it is anticipated that a one use-case would be the inclusion of just some parameters in foreign
language in an otherwise English message. For example, an English message about user authentication
could have a username in Japanese. Similarly, any number of languages can be combined in a CiscoLog
message.

In order to take advantage of messages, which include a foreign language, a log viewer capable of
interpreting UTF-8 would be necessary. Most likely, the log viewer would also require that the
appropriate language fonts be installed on a given system. In a US-ASCII only editor, the user will see
garbage for non-US-ASCII characters encoded in UTF-8, but should be able to see all US-ASCII text.

Internationalization support can be readily used with CiscoLog messages written to a local file. Syslog
RFC 3164, however, does not currently define foreign language support. Thus, in order to take
advantage of internationalization with a syslog server, one would need to use a server implementation,
which was tested to correctly relay or store all 8-bits of each octet unchanged. This would ensure that
UTF-8 encoded parts of the message retain all their information when foreign languages are used.

In UTF-8, a single character is encoded with one or more octets. The CiscoLog message length limit is
specified as 800 octets. Developers must be aware that with foreign languages, the 800-octet length limit
may mean fewer than 800 characters. When a message is split into a multi-part message using guidelines
provided in *Multipart Messages*, page 6-4, octets belonging to a single character must never be split into
separate lines.

### Versioning

CiscoLog does not provide any versioning information in the message format. Extensions to the format
must be made within the restrictions of the format. CiscoLog message formats provides for extensions
by way of defining additional tags.
If applications require changes to existing messages, the value of APPNAME can redefine message within the new space. For example, the application version can be appended to the application name as BACC_2_5 for BACC 2.5.

Preconfigured System Alarm Notifications

The following list contains the preconfigured system alerts in RTMT. Refer to the Real-Time Monitoring Tool Administration Guide for information on configuration.

- AuthenticationFailed, page 6-19
- CiscoDRFFailure, page 6-20
- CoreDumpFileFound, page 6-21
- CpuPegging, page 6-21
- CriticalServiceDown, page 6-22
- HardwareFailure, page 6-22
- LogFileSearchStringFound, page 6-23
- LogPartitionHighWaterMarkExceeded, page 6-23
- LogPartitionLowWaterMarkExceeded, page 6-24
- LowActivePartitionAvailableDiskSpace, page 6-25
- LowAvailableVirtualMemory, page 6-25
- LowInactivePartitionAvailableDiskSpace, page 6-26
- LowSwapPartitionAvailableDiskSpace, page 6-26
- ServerDown, page 6-27
- SparePartitionHighWaterMarkExceeded, page 6-28
- SparePartitionLowWaterMarkExceeded, page 6-29
- SyslogSeverityMatchFound, page 6-30
- SyslogStringMatchFound, page 6-30
- SystemVersionMismatched, page 6-31
- TotalProcessesAndThreadsExceededThreshold, page 6-31

AuthenticationFailed

Authentication validates the user ID and password that are submitted during log in. An alarm gets raised when an invalid user ID and/or the password gets used.

Table 6-3 provides the default configuration for the AuthenticationFailed RTMT Alert.

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
</tbody>
</table>
Cisco DRFFailure

This alert occurs when the DRF backup or restore process encounters errors. Table 6-4 provides the default configuration for the CiscoDRFFailure RTMT Alert.

### Table 6-4  Default Configuration for the CiscoDRFFailure RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met: CiscoDRFFailure event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>
CoreDumpFileFound

This alert occurs when the CoreDumpFileFound event gets generated. This indicates that a core dump file exists in the system.

Table 6-5 provides the default configuration for the CoreDumpFileFound RTMT Alert.

Table 6-5 Default Configuration for the CoreDumpFileFound RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>CoreDumpFileFound event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Trace download Parameters</td>
<td>Not Selected</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

CpuPegging

CPU usage gets monitored based on configured thresholds. If the usage goes above the configured threshold, this alert gets generated.

Table 6-6 provides the default configuration for the CpuPegging RTMT Alert.

Table 6-6 Default Configuration for the CpuPegging RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>99%</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert only when value remains constantly below or over threshold for 60 seconds</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger up to 3 alerts within 30 minutes</td>
</tr>
</tbody>
</table>
CriticalServiceDown

The CriticalServiceDown alert gets generated when the service status equals down (not for other states). Table 6-7 provides the default configuration for the CriticalServiceDown RTMT Alert.

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

HardwareFailure

This alert occurs when a hardware failure event (disk drive failure, power supply failure, and others) triggers. Table 6-8 provides the default configuration for the HardwareFailure RTMT Alert.

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
</tbody>
</table>
Preconfigured System Alarm Notifications

**LogFileSearchStringFound**

This alert occurs when the LogFileSearchStringFound event gets generated. This indicates that the search string was found in the log file.

Table 6-9 provides the default configuration for the LogFileSearchStringFound RTMT Alert.

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Warning</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>LogFileSearchStringFound event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**LogPartitionHighWaterMarkExceeded**

This alert occurs when the percentage of used disk space in the log partition exceeds the configured high water mark. When this alert gets generated, LPM deletes files in the log partition (down to low water mark) to avoid running out of disk space.
**Preconfigured System Alarm Notifications**

**Note**  
LPM may delete files that you want to keep. You should act immediately when you receive the LogPartitionHighWaterMarkExceeded alert.

Table 6-10 provides the default configuration for the LogPartitionHighWaterMarkExceeded RTMT Alert.

**Table 6-10 Default Configuration for the LogPartitionHighWaterMarkExceeded RTMT Alert**

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Log Partition Used Disk Space Exceeds High Water Mark (95%)</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**LogPartitionLowWaterMarkExceeded**

This alert occurs when the LogPartitionLowWaterMarkExceeded event gets generated. This indicates that the percentage of used disk space in the log partition exceeded the configured low water mark.

**Note**  
Be aware that this alert is an early warning. The administrator should start freeing up disk space. Using RTMT/TLC, you can collect trace/log files and delete them from the server. The administrator should adjust the number of trace files that are kept to avoid hitting the low water mark again.

Table 6-11 provides the default configuration for the LogPartitionLowWaterMarkExceeded RTMT Alert.

**Table 6-11 Default Configuration for the LogPartitionLowWaterMarkExceeded RTMT Alert**

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
</tbody>
</table>
Table 6-11  Default Configuration for the LogPartitionLowWaterMarkExceeded RTMT Alert (continued)

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Log Partition Used Disk Space Exceeds Low Water Mark (90%)</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

LowActivePartitionAvailableDiskSpace

This alert occurs when the percentage of available disk space on the active partition is lower than the configured value.

Table 6-12 provides the default configuration for the LowActivePartitionAvailableDiskSpace RTMT Alert.

Table 6-12  Default Configuration for the LowActivePartitionAvailableDiskSpace RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Active Partition available disspace below (4%)</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger up to 3 alerts within 30 minutes</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

LowAvailableVirtualMemory

RTMT monitors virtual memory usage. When memory runs low, a LowAvailableVirtualMemory alert gets generated.

Table 6-13 provides the default configuration for the LowAvailableVirtualMemory RTMT Alert.
Preconfigured System Alarm Notifications

Chapter 6 Cisco Unified Serviceability Alarms and CiscoLog Messages

Table 6-13 Default Configuration for the LowAvailableVirtualMemory RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>the following servers</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Available virtual memory below (30%)</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger up to 3 alerts within 30 minutes</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

LowInactivePartitionAvailableDiskSpace

This alert occurs when the percentage of available disk space of the inactive partition equals less than the configured value.

Table 6-14 provides the default configuration for the LowInactivePartitionAvailableDiskSpace RTMT Alert.

Table 6-14 Default Configuration for the LowInactivePartitionAvailableDiskSpace RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>the following servers</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Inactive Partition available disk space below (4%)</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger up to 3 alerts within 30 minutes</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

LowSwapPartitionAvailableDiskSpace

This alert indicates that the available disk space on the swap partition is low.
Note: The swap partition makes up part of virtual memory, so low available swap partition disk space means low virtual memory as well.

Table 6-15 provides the default configuration for the LowSwapPartitionAvailableDiskSpace RTMT Alert.

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold Trigger alert when following condition met:</td>
<td>Swap Partition available disk space below (105)</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger up to 3 alerts within 30 minutes</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

ServerDown

This alert occurs when a remote node cannot be reached.

Note: Cisco Unified CM clusters only—The ServerDown alert gets generated when the currently “active” AMC (primary AMC or the backup AMC, if the primary is not available) cannot reach another server in a cluster. This alert identifies network connectivity issues in addition to a server down condition.

Table 6-16 provides the default configuration for the ServerDown RTMT Alert.

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
</tbody>
</table>
Preconfigured System Alarm Notifications

**SparePartitionHighWaterMarkExceeded**

This alert occurs when the SparePartitionHighWaterMarkExceeded event gets generated. It indicates that the percentage of used disk space in the spare partition exceeds the configured high water mark. Some core file or log files are purged until the percentage of used disk space in the spare partition is below the configured low water mark. Check if the configured high water mark for used disk space in the spare partition is too low.

Cisco Log Partition Monitoring Tool (LPM) starts purging trace log files in the spare partition and keeps deleting trace log files in the spare partition until spare partition disk usage is just below the low water mark.

Name of the service generating this alarm is Cisco Log Partition Monitoring Tool.

Check if the configured high water mark for used disk space in the spare partition is too low; if it is, change the high water mark setting to a higher value. Also examine each application trace log files under spare partition and delete those trace log files that are too old or too big.

---

**Note**

Spare Partition is not used for Intercompany Media Engine server. So this alert will not be triggered for Intercompany Media Engine.

---

Table 6-17 provides the default configuration for the SparePartitionHighWaterMarkExceeded RTMT Alert.

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Spare Partition Used Disk Space Exceeds High Water Mark (95%)</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
</tbody>
</table>
SparePartitionLowWaterMarkExceeded

This alert occurs when the SparePartitionLowWaterMarkExceeded event gets generated. It indicates that the percentage of used disk space in the spare partition has exceeded the configured low water mark threshold. There are files to be purged by Cisco Log Partition Monitoring Tool (LPM). If the spare partition disk usage keeps increasing until it exceeded the configured high water mark, Cisco LPM starts purging the trace log files in the spare partition. Cisco LPM sends the alarm periodically if the spare partition disk usage has not changed.

Name of the service generating this alarm is Cisco Log Partition Monitoring Tool.

Check if the configured low water mark for used disk space in the spare partition is too low; if, change the low/high water mark settings to the higher values. Also examine each application trace log files under spare partition and clean up those trace log files that are too old or too big before the used disk space exceeds the high water mark.

**Note**

Spare Partition is not used for Intercompany Media Engine server. So this alert will not be triggered for Intercompany Media Engine.

Table 6-18 provides the default configuration for the default configuration for the SparePartitionLowWaterMarkExceeded RTMT Alert.

### Table 6-18 Default Configuration for the SparePartitionLowWaterMarkExceeded RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>the following servers</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Spare Partition Used Disk Space Exceeds Low Water Mark (90%)</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>
SyslogSeverityMatchFound

This alert occurs when the SyslogSeverityMatchFound event gets generated. This indicates that a syslog message with the matching severity level exists.

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>following servers</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>SyslogSeverityMatchFound event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Syslog Severity Parameters</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

SyslogStringMatchFound

This alert occurs when the SyslogStringMatchFound event gets generated. The alert indicates that a syslog message with the matching search string exists.

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>following servers</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>SyslogStringMatchFound event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Syslog Alert Parameters</td>
<td>(Text box for search string)</td>
</tr>
</tbody>
</table>
### Table 6-20  Default Configuration for the SyslogStringMatchFound RTMT Alert (continued)

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

### SystemVersionMismatched

This alert occurs when a mismatch in system version exists.

### Table 6-21  Default Configuration for the SystemVersionMismatched RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met: SystemVersionMismatched occurred</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger up to 1 alert within 60 minutes</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

### TotalProcessesAndThreadsExceededThreshold

This alert occurs when the TotalProcessesAndThreadsExceededThreshold event gets generated. The alert indicates that the current total number of processes and threads exceeds the maximum number of tasks that are configured for the Cisco RIS Data Collector Service Parameter. This situation could indicate that a process is leaking or that a process has thread leaking.

### Table 6-22  Default Configuration for the TotalProcessesAndThreadsExceededThreshold RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met: TotalProcessesAndThreadsExceededThreshold event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
</tbody>
</table>
Preconfigured CallManager Alarm Notifications

The following list comprises the preconfigured CallManager alerts in RTMT. Refer to the Real-Time Monitoring Tool Administration Guide for information on configuration.

- BeginThrottlingCallListBLFSubscriptions, page 6-33
- CallProcessingNodeCpuPegging, page 6-33
- CDRAgentSendFileFailed, page 6-34
- CDRFileDeliveryFailed, page 6-35
- CDRHighWaterMarkExceeded, page 6-35
- CDRMaximumDiskSpaceExceeded, page 6-36
- CodeYellow, page 6-36
- DBChangeNotifyFailure, page 6-37
- DBReplicationFailure, page 6-37
- DDRBlockPrevention, page 6-38
- DDRDown, page 6-39
- ExcessiveVoiceQualityReports, page 6-39
- IMEDistributedCacheInactive, page 6-40
- IMEOverQuota, page 6-40
- IMEQualityAlert, page 6-41
- InsufficientFallbackIdentifiers, page 6-42
- IMEServiceStatus, page 6-42
- InvalidCredentials, page 6-43
- LowCallManagerHeartbeatRate, page 6-44
- LowTFTPServerHeartbeatRate, page 6-44
- MaliciousCallTrace, page 6-45
- MediaListExhausted, page 6-45
- MgcpDChannelOutOfService, page 6-46
- NumberOfRegisteredDevicesExceeded, page 6-46
- NumberOfRegisteredGatewaysDecreased, page 6-47
- NumberOfRegisteredGatewaysIncreased, page 6-47
- NumberOfRegisteredMediaDevicesDecreased, page 6-48

### Table 6-22  Default Configuration for the TotalProcessesAndThreadsExceededThreshold RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

Preconfigured CallManager Alarm Notifications

The following list comprises the preconfigured CallManager alerts in RTMT. Refer to the Real-Time Monitoring Tool Administration Guide for information on configuration.

- BeginThrottlingCallListBLFSubscriptions, page 6-33
- CallProcessingNodeCpuPegging, page 6-33
- CDRAgentSendFileFailed, page 6-34
- CDRFileDeliveryFailed, page 6-35
- CDRHighWaterMarkExceeded, page 6-35
- CDRMaximumDiskSpaceExceeded, page 6-36
- CodeYellow, page 6-36
- DBChangeNotifyFailure, page 6-37
- DBReplicationFailure, page 6-37
- DDRBlockPrevention, page 6-38
- DDRDown, page 6-39
- ExcessiveVoiceQualityReports, page 6-39
- IMEDistributedCacheInactive, page 6-40
- IMEOverQuota, page 6-40
- IMEQualityAlert, page 6-41
- InsufficientFallbackIdentifiers, page 6-42
- IMEServiceStatus, page 6-42
- InvalidCredentials, page 6-43
- LowCallManagerHeartbeatRate, page 6-44
- LowTFTPServerHeartbeatRate, page 6-44
- MaliciousCallTrace, page 6-45
- MediaListExhausted, page 6-45
- MgcpDChannelOutOfService, page 6-46
- NumberOfRegisteredDevicesExceeded, page 6-46
- NumberOfRegisteredGatewaysDecreased, page 6-47
- NumberOfRegisteredGatewaysIncreased, page 6-47
- NumberOfRegisteredMediaDevicesDecreased, page 6-48
• NumberOfRegisteredMediaDevicesIncreased, page 6-48
• NumberOfRegisteredPhonesDropped, page 6-49
• RouteListExhausted, page 6-49
• SDLLinkOutOfService, page 6-50
• TCPSetupToIMEFailed, page 6-50
• TLSConnectionToIMEFailed, page 6-51

**BeginThrottlingCallListBLFSubscriptions**

This alert occurs when the BeginThrottlingCallListBLFSubscriptions event gets generated. This indicates that the Cisco Unified Communications Manager initiated a throttling of the CallList BLF Subscriptions to prevent a system overload.

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>BeginThrottlingCallListBLFSubscriptions event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**CallProcessingNodeCpuPegging**

This alert occurs when the percentage of CPU load on a call processing server exceeds the configured percentage for the configured time.

*Note*  
If the administrator takes no action, high CPU pegging can lead to a crash, especially in CallManager service. CoreDumpFound and CriticalServiceDown alerts might also get issued.

The CallProcessingNodeCpuPegging alert gives you time to work proactively to avoid a Cisco Unified Communications Manager crash.
Table 6-24  Default Configuration for the CallProcessingNodeCpuPegging RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>on the following servers</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Processor load over (90%)</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert only when value constantly below or over threshold for 60 seconds</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger up to 3 alerts within 30 minutes</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

CDRAgentSendFileFailed

This alert gets raised when the CDR Agent cannot send CDR files from a Cisco Unified Communications Manager node to a CDR repository node within the Cisco Unified Communications Manager cluster.

Table 6-25  Default Configuration for the CDRAgentSendFileFailed RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>on the following servers</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>CDRAgentSendFileFailed event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>


**CDRFileDeliveryFailed**

This alert gets raised when(s) FTP delivery of CDR files to the outside billing server fails.

*Table 6-26  Default Configuration for the CDRFileDeliveryFailed RTMT Alert*

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>CDRFileDeliveryFailed event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**CDRHighWaterMarkExceeded**

This alert gets raised when the high water mark for CDR files gets exceeded. It also indicates that some successfully delivered CDR files got deleted.

*Table 6-27  Default Configuration for the CDRHighWaterMarkExceeded RTMT Alert*

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>CDRHighWaterMarkExceeded event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>
Preconfigured CallManager Alarm Notifications

Chapter 6      Cisco Unified Serviceability Alarms and CiscoLog Messages

CDRMaximumDiskSpaceExceeded

This alarm gets raised when the CDR files disk usage exceeds the maximum disk allocation. It also indicates that some undeliverable files got deleted.

Table 6-28  Default Configuration for the CDRMaximumDiskSpaceExceeded RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>CDRMaximumDiskSpaceExceeded event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

CodeYellow

The AverageExpectedDelay counter represents the current average expected delay to handle any incoming message. If the value exceeds the value that is specified in CodeYellow Entry Latency service parameter, the CodeYellow alarm gets generated. You can configure the CodeYellow alert to download trace files for troubleshooting purposes.

Table 6-29  Default Configuration for the CodeYellow RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Cisco CallManager CodeYellowEntry event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Trace Download Parameters</td>
<td>Enable Trace Download not selected</td>
</tr>
</tbody>
</table>
Table 6-29  Default Configuration for the CodeYellow RTMT Alert (continued)

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**DBChangeNotifyFailure**

This alert occurs when the Cisco Database Notification Service experiences problems and might stop. This condition indicates change notification requests that are queued in the database got stuck and changes made to the system will not take effect. Ensure that the Cisco Database Layer Monitor is running on the node where the alert exists. If it is, restart the service. If that does not return this alert to safe range, collect the output of `show tech notify` and `show tech dbstateinfo` and contact TAC for information about how to proceed.

Table 6-30  Default Configuration for the DBChangeNotifyFailure RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>DBChangeNotify queue delay over 2 minutes</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger up to 1 alert within 30 minutes</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**DBReplicationFailure**

This alarm indicates a failure in IDS replication and requires database administrator intervention.

*Note* Be aware that DBReplicationFailure is based on the replication status perfmon counter (instead of DBReplicationFailure alarm as was previously the case). This alert gets triggered whenever the corresponding replication status perfmon counter specifies a value of 3 (Bad Replication) or 4 (Replication Setup Not Successful).
Preconfigured CallManager Alarm Notifications

**Table 6-31 Default Configuration for the DBReplicationFailure RTMT Alert**

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>following servers</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>DBReplicationFailure occurred</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger up to 1 alert within 60 minutes</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**DDRBlockPrevention**

This alert gets triggered when the IDSReplicationFailure alarm with alarm number 31 occurs, which invokes a proactive procedure to avoid denial of service. This procedure does not impact call processing; you can ignore replication alarms during this process.

The procedure takes up to 60 minutes to finish. Check that RTMT replication status equals 2 on each node to make sure that the procedure is complete. Do not perform a system reboot during this process.

**Table 6-32 Default Configuration for the DDRBlockPrevention RTMT Alert**

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>following servers</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>IDSReplicationFailure alarm with alarm number 31 generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger up to 1 alert within 60 minutes</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>
**DDRDown**

This alert gets triggered when the IDSReplicationFailure alarm with alarm number 32 occurs. An auto recover procedure runs in the background, and no action is needed.

The procedure takes about 15 minutes to finish. Check that RTMT replication status equals 2 on each node to make sure the procedure is complete.

<table>
<thead>
<tr>
<th>Table 6-33 Default Configuration for the DDRDown RTMT Alert</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Value</strong></td>
</tr>
<tr>
<td>Enable Alert</td>
</tr>
<tr>
<td>Severity</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
</tr>
<tr>
<td>Threshold</td>
</tr>
<tr>
<td>Duration</td>
</tr>
<tr>
<td>Frequency</td>
</tr>
<tr>
<td>Schedule</td>
</tr>
<tr>
<td>Enable E-mail</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
</tr>
</tbody>
</table>

**ExcessiveVoiceQualityReports**

This alert gets generated when the number of QRT problems that are reported during the configured time interval exceed the configured value. The default threshold specifies 0 within 60 minutes.

<table>
<thead>
<tr>
<th>Table 6-34 Default Configuration for the ExcessiveVoiceQualityReports RTMT Alert</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Value</strong></td>
</tr>
<tr>
<td>Enable Alert</td>
</tr>
<tr>
<td>Severity</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
</tr>
<tr>
<td>Threshold</td>
</tr>
<tr>
<td>Duration</td>
</tr>
<tr>
<td>Frequency</td>
</tr>
<tr>
<td>Schedule</td>
</tr>
<tr>
<td>Enable E-mail</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
</tr>
</tbody>
</table>
Preconfigured CallManager Alarm Notifications

Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Preconfigured CallManager Alarm Notifications

IMEDistributedCacheInactive

This alarm gets generated when a Cisco Unified Communications Manager attempts to connect to the Cisco IME server, but the IME distributed cache is not currently active.

Ensure that the certificate for the Cisco IME server is provisioned and that the IME distributed cache has been activated via the CLI.

Default Configuration

Table 6-35 Default Configuration for the IMEDistributedCacheInactive RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Error</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Inactive IME Distributed Cache</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

IMEOverQuota

This alert indicates that the Cisco Unified Communications Manager servers that use this Cisco IME service have exceed the quota for published direct inward dialing numbers (DIDs) to the IME distributed cache. The alert includes the name of the Cisco IME server as well as the current and target quota values.

Ensure that you have correctly provisioned the DID prefixes on all of the Cisco Unified Communications Manager servers that use this Cisco IME service.

If you have provisioned the prefixes correctly, you have exceeded the capacity of your Cisco IME service, and you need to configure another service and divide the DID prefixes across the Cisco IME client instances (Cisco Unified Communications Managers) on different Cisco IME services.

Default Configuration

Table 6-36 Default Configuration for the IMEOverQuota Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Alert</td>
</tr>
</tbody>
</table>
This alert gets generated when Cisco Unified Communications Manager determines that a substantial number of Cisco IME calls fail back to PSTN or fail to be set up due to IP network quality problems. Two types of events trigger this alert:

A large number of the currently active Cisco IME calls have all requested fallback or have fallen back to the PSTN.

A large number of the recent call attempts have gone to the PSTN and not been made over IP.

When you receive this alert, check your IP connectivity. If no problems exist with the IP connectivity, you may need to review the CDRs, CMRs, and logs from the firewalls to determine why calls have fallen back to the PSTN or have not been made over IP.

**Default Configuration**

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met: Cisco IME link quality problem</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**IMEQualityAlert**

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Error</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met: Cisco IME link quality problem</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>
InsufficientFallbackIdentifiers

This alert gets generated when too many Cisco IME calls that are currently in progress use the same fallback DID and no more DTMF digit sequences exist to allocate to a new Cisco IME call that Cisco Unified Communications Manager is processing. The new call continues, but the call cannot fallback to the PSTN if voice-quality deteriorates.

If this alert gets generated, note the fallback profile that associates with this call. Check that profile in Cisco Unified Communications Manager Administration, and examine the current setting for the “Fallback Number of Correlation DTMF Digits” field. Increase the value of that field by one, and check whether the new value eliminates these alerts. In general, this parameter should be large enough so that the number of simultaneous Cisco IME calls that are made to enrolled numbers that associate with that profile is always substantially less than 10 raised to the power of this number. For example, if you always have fewer than 10,000 simultaneous Cisco IME calls for the patterns that associate with this fallback profile, setting this value to 5 (10 to the power of 5 equals 100,000) should keep Cisco Unified Communications Manager from generating this alert.

However, increasing this value results in a small increase in the amount of time it takes to perform the fallback. As such, you should set the “Fallback Number of Correlation DTMF Digits” field to a value just large enough to prevent this alert from getting generated.

Instead of increasing the value of the DTMF digits field, you can add another fallback profile with a different fallback DID and associate that fallback profile with a smaller number of enrolled patterns. If you use this method, you can use a smaller number of digits.

Default Configuration

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Error</td>
</tr>
<tr>
<td>Enable/Disable this alert on</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>the following servers</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Cannot allocate fallback identifier</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger up to 1 alerts within one minute</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

IMEServiceStatus

This alert indicates the overall health of the connection to the Cisco IME services for a particular Cisco IME client instance (Cisco Unified Communications Manager). The alert indicates the following states:

- 0—Unknown. Likely indicates that the Cisco IME service has not been activated.
• 1—Healthy. Indicates that the Cisco Unified Communications Manager has successfully established a connection to its primary and backup servers for the Cisco IME client instance, if configured.

• 2—Unhealthy. Indicates that the Cisco IME has been activated but has not successfully completed handshake procedures with the Cisco IME server. Note that this counter reflects the handshake status of both the primary and the secondary IME servers.

Default Configuration

<table>
<thead>
<tr>
<th>Table 6-39 Default Configuration for the IMEServiceStatus Alert</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Value</strong></td>
</tr>
<tr>
<td>Enable Alert</td>
</tr>
<tr>
<td>Severity</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
</tr>
<tr>
<td>Threshold</td>
</tr>
<tr>
<td>Duration</td>
</tr>
<tr>
<td>Frequency</td>
</tr>
<tr>
<td>Schedule</td>
</tr>
<tr>
<td>Enable E-mail</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
</tr>
</tbody>
</table>

InvalidCredentials

The alert indicates that the Cisco Unified Communications Manager cannot connect to the Cisco IME server because the username and/or password configured on Cisco Unified Communications Manager do not match those configured on the Cisco IME server.

The alert includes the username and password that were used to connect to the Cisco IME server as well as the IP address and name of the target Cisco IME server. To resolve this alert, log into the Cisco IME server and check that the configured username and password match the username and password that are configured in Cisco Unified Communications Manager.

Default Configuration

<table>
<thead>
<tr>
<th>Table 6-40 Default Configuration for the InvalidCredentials Alert</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Value</strong></td>
</tr>
<tr>
<td>Enable Alert</td>
</tr>
<tr>
<td>Severity</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
</tr>
</tbody>
</table>
Preconfigured CallManager Alarm Notifications

Chapter 6    Cisco Unified Serviceability Alarms and CiscoLog Messages

Table 6-40    Default Configuration for the InvalidCredentials Alert (continued)

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Credential Failure to Cisco IME server</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

LowCallManagerHeartbeatRate

This alert occurs when the CallManager heartbeat rate equals less than the configured value.

Table 6-41    Default Configuration for the LowCallManagerHeartbeatRate RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Warning</td>
</tr>
<tr>
<td>Enable/Disable this alert on</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>the following servers</td>
<td></td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>CallManager Server heartbeat rate below 24 beats per minute.</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

LowTFTPServerHeartbeatRate

This alert occurs when TFTP server heartbeat rate equals less than the configured value.

Table 6-42    Default Configuration for the LowTFTPServerHeartbeatRate RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Warning</td>
</tr>
<tr>
<td>Enable/Disable this alert on</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>the following servers</td>
<td></td>
</tr>
</tbody>
</table>
Preconfigured CallManager Alarm Notifications

Table 6-42  Default Configuration for the LowTFTPServerHeartbeatRate RTMT Alert (continued)

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>TFTP server heartbeat rate below 24 beats per minute</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

MaliciousCallTrace

This indicates that a malicious call exists in Cisco Unified Communications Manager. The malicious call identification (MCID) feature gets invoked.

Table 6-43  Default Configuration for the MaliciousCallTrace RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Malicious call trace generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

MediaListExhausted

This alert occurs when the number of MediaListExhausted events exceeds the configured threshold during the configured time interval. This indicates that all available media resources that are defined in the media list are busy. The default specifies 0 within 60 minutes.

Table 6-44  Default Configuration for the MediaListExhausted RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Warning</td>
</tr>
</tbody>
</table>
Table 6-44  Default Configuration for the MediaListExhausted RTMT Alert (continued)

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Number of MediaListExhausted events exceeds 0 times within the last 60 minutes</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

MgcpDChannelOutOfService

This alert gets triggered when the MGCP D-Channel remains out of service.

Table 6-45  Default Configuration for the MgcpDChannelOutOfService RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>MGCP D-Channel is out-of-service</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

NumberOfRegisteredDevicesExceeded

This alert occurs when the NumberOfRegisteredDevicesExceeded event gets generated.
Table 6-46  Default Configuration for the NumberOfRegisteredDevicesExceeded RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>NumberOfRegisteredDevicesExceeded event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**NumberOfRegisteredGatewaysDecreased**

This alert occurs when the number of registered gateways in a cluster decreases between consecutive polls.

Table 6-47  Default Configuration for the NumberOfRegisteredGatewaysDecreased RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Number of registered gateway decreased</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**NumberOfRegisteredGatewaysIncreased**

This alert occurs when the number of registered gateways in the cluster increased between consecutive polls.
Preconfigured CallManager Alarm Notifications

Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Table 6-48  Default Configuration for the NumberOfRegisteredGatewaysIncreased RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met: Number of registered gateways increased</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**NumberOfRegisteredMediaDevicesDecreased**

This alert occurs when the number of registered media devices in a cluster decreases between consecutive polls.

Table 6-49  Default Configuration for the NumberOfRegisteredMediaDevicesDecreased RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met: Number of registered media devices decreased</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**NumberOfRegisteredMediaDevicesIncreased**

This alert occurs when the number of registered media devices in a cluster increases between consecutive polls.

Table 6-50  Default Configuration for the NumberOfRegisteredMediaDevicesIncreased RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
</tbody>
</table>
Table 6-50  Default Configuration for the NumberOfRegisteredMediaDevicesIncreased RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Number of registered media devices increased</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**NumberOfRegisteredPhonesDropped**

This alert occurs when the number of registered phones in a cluster drops more than the configured percentage between consecutive polls.

Table 6-51  Default Configuration for the NumberOfRegisteredPhonesDropped RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Number of registered phones in the cluster drops (10%)</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

**RouteListExhausted**

An available route could not be found in the indicated route list.

Table 6-52  Default Configuration for the RouteListExhausted RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Warning</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
</tbody>
</table>
Preconfigured CallManager Alarm Notifications

Chapter 6    Cisco Unified Serviceability Alarms and CiscoLog Messages

Table 6-52   Default Configuration for the RouteListExhausted RTMT Alert (continued)

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Number of RouteListExhausted exceeds 0 times within the last 60 minutes</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

SDLLinkOutOfService

This alert occurs when the SDLLinkOutOfService event gets generated. This event indicates that the local Cisco Unified Communications Manager cannot communicate with the remote Cisco Unified Communications Manager. This event usually indicates network errors or a nonrunning, remote Cisco Unified Communications Manager.

Table 6-53   Default Configuration for the SDLLinkOutOfService RTMT Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>SDLLinkOutOfService event generated</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

TCPSetupToIMEFailed

This alert occurs when Cisco Unified Communications Manager cannot establish a TCP connection to a Cisco IME server. This alert typically occurs when the IP address and port of the Cisco IME server are misconfigured in Cisco Unified Communications Manager Administration or when an Intranet connectivity problem exists and prevents the connection from being set up.

Ensure that the IP address and port of the Cisco IME server in the alert are valid. If the problem persists, test the connectivity between the Cisco Unified Communications Manager servers and the Cisco IME server.
Table 6-54  Default Configuration for the TCPSetupToIMEFailed Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Critical</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>Connection Failure to Cisco IME server</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>

### TLSConnectionToIMEFailed

This alert occurs when a TLS connection to the Cisco IME service could not be established because the certificate presented by the Cisco IME service has expired or is not in the Cisco Unified Communications Manager CTL.

Ensure that the Cisco IME service certificate has been configured into the Cisco Unified Communications Manager.

**Default Configuration**

Table 6-55  Default Configuration for the TLSConnectionToIMEFailed Alert

<table>
<thead>
<tr>
<th>Value</th>
<th>Default Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Alert</td>
<td>Selected</td>
</tr>
<tr>
<td>Severity</td>
<td>Alert</td>
</tr>
<tr>
<td>Enable/Disable this alert on the following servers</td>
<td>Enabled on listed servers</td>
</tr>
<tr>
<td>Threshold</td>
<td>Trigger alert when following condition met:</td>
</tr>
<tr>
<td></td>
<td>TLS Failure to Cisco IME service</td>
</tr>
<tr>
<td>Duration</td>
<td>Trigger alert immediately</td>
</tr>
<tr>
<td>Frequency</td>
<td>Trigger alert on every poll</td>
</tr>
<tr>
<td>Schedule</td>
<td>24 hours daily</td>
</tr>
<tr>
<td>Enable E-mail</td>
<td>Selected</td>
</tr>
<tr>
<td>Trigger Alert Action</td>
<td>Default</td>
</tr>
</tbody>
</table>
Emergency-Level Alarms

The emergency-level alarm equals zero (0) and means that your system or service is unusable. These alarms generally indicate platform failures. Examples follow:

- Service repeatedly fails to startup
- System ran out of disk space while disk space is essential for this system to operate
- System ran out of memory
- Motherboard failure occurred

This level is not suitable for events associated with an individual end point.

BDINotStarted

BDI application not started because of an error.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Emergency (0)

Parameters
Reason [String]

Recommended Action
See application logs for error.

CallDirectorCreationError

There was an error during the CallDirector creation.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Emergency (0)

Parameters
None
Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Emergency-Level Alarms

CiscoDirSyncStartFailure
Cisco DirSync application failed to start successfully. Error occurred while starting application

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Emergency (0)

Recommended Action
See application logs for error, may require restarting the application.

ExceptionInInitSDIConfiguration
Exception occurred in InitSDIConfiguration function.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Emergency (0)

Parameters
None

Recommended Action
None

FileWriteError
Failed to write into the primary file path.

Cisco Unified Serviceability Alarm Definition Catalog
System/Generic
Emergency-Level Alarms

Severity
Emergency (0)

Parameters
Primary File Path(String)

Recommended Action
Ensure that the primary file path is valid and the corresponding drive has sufficient disk space. Also, make sure that the path has security permissions similar to default log file path.

GlobalSPUtilsCreationError

There was an error during the GlobalSPUtils creation.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Emergency (0)

Parameters
None

Recommended Action
None

HuntGroupControllerCreationError

There was an error during the HuntGroupController creation.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Emergency (0)

Parameters
None

Recommended Action
None
HuntGroupCreationError

There was an error during the Hunt Group creation.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Emergency (0)

Parameters
None

Recommended Action
None

IPAddressResolveError

The host IP address was not resolved.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Emergency (0)

Parameters
HostName [String]

Recommended Action
None

IPMANotStarted

IPMA application not started because of an error.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications
### LineStateSrvEngCreationError

There was an error during the LineStateSrvEng creation.

**Severity**
Emergency (0)

**Parameters**
Servlet Name [String] Reason [String]

**Recommended Action**
See application logs for error.

### LostConnectionToCM

TCD connection to CallManager was lost.

**Severity**
Emergency (0)

**Parameters**
None

**Recommended Action**
None
**NoCMEntriesInDB**

There are no CallManager entries in the database.

**Facility/Sub-Facility**
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

**Severity**
Emergency (0)

**Parameters**
None

**Recommended Action**
None

**NoFeatureLicense**

No feature license found. Cisco Unified Communications Manager (Unified CM) requires a license to function. Also, Unified CM licenses are version-specific so be certain that the license is for the version you are trying to run. You can run a license unit report in Cisco Unified CM Administration (System > Licensing > License Unit Report).

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Emergency.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

**Severity**
Emergency

**Recommended Action**
Request license generation for Cisco Unified Communications Manager SWFEATURE for your version of Unified CM and upload the license in Cisco Unified CM Administration (System > Licensing > License File Upload).
OutOfMemory

The process has requested memory from the operating system, and there was not enough memory available.

Cisco Unified Serviceability Alarm Definition Catalog
System/Generic

Severity
Emergency (0)

Parameters
None

Recommended Action
None

ServiceNotInstalled

An executable is trying to start but cannot because it is not configured as a service in the service control manager. The service is %s. Service is not installed.

Cisco Unified Serviceability Alarm Definition Catalog
System/Generic

Severity
Emergency (0)

Parameters
Service (String)

Recommended Action
Reinstall the service.

SyncDBCreationError

There was an error during the SyncDB creation in SysController.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Emergency (0)
Parameters
None

Recommended Action
None

SysControllerCreationError
There was an error during the SysController creation.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Emergency (0)

Parameters
None

Recommended Action
None

TapiLinesTableCreationError
There was an error during the TapiLinesTable creation.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Emergency (0)

Parameters
None

Recommended Action
None
**TimerServicesCreationError**

There was an error during the TimerServices creation.

**Facility/Sub-Facility**
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

**Severity**
Emergency (0)

**Parameters**
None

**Recommended Action**
None

**TestAlarmEmergency**

Testing emergency alarm.

**Facility/Sub-Facility**
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
System/Test

**Severity**
Emergency (0)

**Recommended Action**
None

**WDNotStarted**

Failed to startup WebDialer application because of an error.

**Facility/Sub-Facility**
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

**Severity**
Emergency (0)
Parameters
Servlet Name [String] Reason [String]

Recommended Action
See application logs for error.

Alert-Level Alarms
The alert-level alarm equals 1 and action must take place immediately. A system error occurred and will not recover without manual intervention. Examples follow:

- Application is about to run out of licenses
- Application is about to run out of disk space
- Application is almost out of memory
- 100% CPU occurs for long period of time

Be aware that this level is not suitable for events that are associated with an individual end point.

CertValidLessThanADay
Certificate is about to expire in less than 24 hours or has expired.

Cisco Unified Serviceability Alarm Definition Catalog
System/CertMonitorAlarmCatalog

Severity
Alert(1)

Routing List
Event Log
Sys Log

Parameters
Message(String)

Recommended Action
Regenerate the certificate that is about to expire by accessing the Cisco Unified Operating System and go to Certificate Management. If the certificate is issued by a CA, generate a CSR, submit the CSR to CA, obtain a fresh certificate from CA, and upload it to Cisco Unified CM.

CMIEException
Error while reading the database.
This alarm is always associated with other alarms, which are triggered due to configuring CMI service parameter with invalid values or due to invalid handle value returned by the serial port.
Cisco Unified Serviceability Alarm Definition Catalog
CMIAAlarmCatalog/CMI

Severity
ALERT

Routing List
Event Log
SDI

Parameter(s)
CMI Exception(String)

Recommended Action
Refer to the associated alarm for further information.

CMOverallInitTimeExceeded

Initialization of the Cisco Unified Communications Manager system has taken longer than allowed by the value specified in the System Initialization Timer service parameter; as a result, the system will automatically restart now to attempt initialization again. Initialization may have failed due to a database error, or due to a large amount of new devices added to the system, or any number of other potential causes. The required time to initialize Cisco Unified Communications Manager has exceeded the time allowed by the Cisco CallManager service parameter, System Initialization Timer. This could be due to an increase in system size.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCMIEException.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager
Severity
Alert

Parameters
Cisco Unified Communications Manager Overall Initialization Time (in minutes) [Int]

Recommended Action
Try increasing the value of the Cisco CallManager service parameter, System Initialization Timer, in the Service Parameters Configuration window in Cisco Unified CM Administration. Use RTMT to discover the number of devices and number of users in the system and evaluate whether the numbers seem accurate. Try increasing the value of the Cisco CallManager service parameter, System Initialization Timer, in the Service Parameters Configuration window in Cisco Unified CM Administration. If increasing the time in the System Initialization Timer service parameter does not correct this issue, contact the Cisco Technical Assistance Center (TAC).

**ConfigThreadChangeNotifyServerInstanceFailed**

Failed to allocate resources to handle configuration change notification from database. This usually indicates a lack of memory when there is a system issue such as running out of resources.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kConfigThreadChangeNotifyServerInstanceFailed</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Alert.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_TFTP-TFTP

**Cisco Unified Serviceability Alarm Definition Catalog**
System/TFTP

**Severity**
Alert

**Recommended Action**
Use RTMT to monitor the system memory resources and consumption and correct any system issues that might be contributing to a reduced amount of system resources.
**ConfigThreadChangeNotifyServerSingleFailed**

Failed to allocate resources to handle configuration change notification from database. This usually indicates a lack of memory when there is a system issue such as running out of resources.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kConfigThreadChangeNotifyServerSingleFailed.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Alert.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_TFTP-TFTP

**Cisco Unified Serviceability Alarm Definition Catalog**

System/TFTP

**Severity**

Alert

**Recommended Action**

Use RTMT to monitor the system memory resources and consumption and correct any system issues that might be contributing to a reduced amount of system resources.

**ConfigThreadChangeNotifyServerStartFailed**

Failed to start listening to configuration change notification from database. This usually indicates a lack of memory when there is a system issue such as running out of resources.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kConfigThreadChangeNotifyServerStartFailed.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Alert.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_TFTP-TFTP

**Cisco Unified Serviceability Alarm Definition Catalog**

System/TFTP

**Severity**

ALERT
Recommended Action
Use RTMT to monitor the system memory resources and consumption and correct any system issues that might be contributing to a reduced amount of system resources.

CiscoLicenseApproachingLimit

License units consumption approaching its authorized limit.

Facility/Sub-Facility
CCM_JAVA_APPS_TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Alert (1)

Parameters
Reason [String]

Recommended Action
None

CiscoLicenseOverDraft

Overdraft licenses in use.

Facility/Sub-Facility
CCM_JAVA_APPS_TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Alert (1)

Parameters
Reason [String]

Recommended Action
None

CMVersionMismatch

One or more Unified CM nodes in a cluster are running different Cisco CallManager versions.
This alarm indicates that the local Unified CM is unable to establish communication with the remote Unified CM due to a software version mismatch. This is generally a normal occurrence when you are upgrading a Unified CM node.

**Cisco Unified Serviceability Alarm Definition Catalog**

**CallManager/CallManager**

**Severity**

ALERT

**Routing List**

SDL
SDI
Sys Log
Event Log
Data Collector

**Parameter(s)**

- Remote Application Link Protocol Version(String)
- Local Application Link Protocol Version(String)
- Remote Node ID(UInt)
- Remote Application ID(Enum)
- Remote Application Version(String)

**Enum Definitions -Remote Application ID**

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>CallManager</td>
</tr>
<tr>
<td>200</td>
<td>CTIManager</td>
</tr>
</tbody>
</table>

**Recommended Action**

The alarm details include the versions of the local and remote Unified CM nodes. Compare the versions and upgrade a node if necessary.

**CreateThreadFailed**

Failed to create a new thread. See Reason string for where it failed. This usually happens when there are system issues such as running out of memory resources.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kCreateThreadFailed.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Alert.</td>
</tr>
</tbody>
</table>
### DBLEException

An error occurred while performing database activities. A severe database layer interface error occurred. Possible causes for this include the database being unreachable or down or a DNS error.

#### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Alert.</td>
</tr>
</tbody>
</table>

#### Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

#### Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

#### Severity

Alert

#### Parameters

ErrorCode [Int] ExceptionString [String]

#### Recommended Action

Review the System Reports provided in the Cisco Unified Reporting tool, specifically the Cisco Unified CM Database Status report, for any anomalous activity. Check network connectivity to the server that is running the database. If your system uses DNS, check the DNS configuration for any errors.
InvalidCredentials

Credential Failure to IME server.
The connection to the IME server could not be completed, because the username and/or password configured on Unified CM do not match those configured on the IME server.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>New Alarm for this release.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
ALERT

Recommended Action
The alarm will include the username and password which were used to connect to the IME server, along with the IP address of the target IME server and its name. Log into the IME server and check that the username and password configured there match those configured in Unified CM.

Routing List
SDL
SDI
Sys Log
Event Log
Alert Manager

Parameter(s)
User name(String)
IP address(String)
Server name(String)
MemAllocFailed

Memory allocation failed.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kMemAllocFailed.</td>
</tr>
<tr>
<td></td>
<td>Severity changed to Alert.</td>
</tr>
<tr>
<td></td>
<td>Recommended action changed.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_SUMI-CMI

**Cisco Unified Serviceability Alarm Definition Catalog**

System/Service Manager

**Severity**

Alert

**Parameters**

Memory Allocation Failure(String)

**Recommended Action**

1. Check the syslog for the system error number.
2. If the Alert is seen repeatedly, restart Service Manager.
3. If the problem still persists, reboot the Cisco Unified CM node.

NoDbConnectionAvailable

No database connection available. Database layer could not find any working database connection.

**Facility/Sub-Facility**

CCM_DB_LAYER-DB

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DB

**Severity**

Alert (1)
### Recommended Action

In Cisco Unified Serviceability, enable Detailed level traces in the Trace Configuration window for the Cisco Database Layer Monitor service. Check network connectivity and operation of SQL Server services.

### ParityConfigurationExceptionError

The CMI service parameter, Parity, has an invalid configuration.

An invalid parity has been configured for the serial port that CMI uses to connect to the voice messaging system. It is possible that the parity value has been updated via AXL or a CLI command where validation of the value was not performed. For this reason, it is best to set this value in the Service Parameter Configuration window in Cisco Unified CM Administration and the value can be validated against the accepted range of values for this field.

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>New name changed from kParityConfigurationExceptionError.</td>
</tr>
</tbody>
</table>

### SerialPortOpeningError

When CMI tries to open the serial port, the operating system returns an error.

For a system running CMI, the serial port through which the voice messaging system is connected is always USB0, and that value is configured in the Cisco Messaging Interface service parameter, Serial Port. It is possible that the Serial Port value has been updated via AXL or a CLI command where validation of the value was not performed. CMI triggers this alarm if the value in the Serial Port service parameter is anything other than USB0.
Serial Port Opening Error(String)

Recommended Action
Ensure that USB0 is configured in the Cisco Messaging Interface service parameter Serial Port. Also, physically confirm that the cable is firmly connected to the USB0 port.

**SDIControlLayerFailed**

Failed to update trace logging or alarm subsystem for new settings. This usually indicates a lack of system resources or a failure in database access by the trace logging or alarm subsystem.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>New name changed from kSerialPortOpeningError.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Name changed from kSDIControlLayerFailed.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TFTP_TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP

Severity
Alert
Parameters
Error [Int] Reason [String]

Recommended Action
In Cisco Unified Serviceability, enable Detailed level traces in the Trace Configuration window for TFTP and Cisco Database Layer Monitor services. Also, use RTMT to look for errors that may have occurred around the time of the alarm. Ensure that the database server is running, and that the Cisco Database Layer Monitor service is running without problems. If this alarm persists, contact the Cisco Technical Assistance Center (TAC) with TFTP service and database trace files.

SDLLinkOOS
SDL link to remote application out of service. This alarm indicates that the local Unified CM has lost communication with the remote Unified CM. This alarm usually indicates that a node has gone out of service (whether intentionally for maintenance or to install a new load for example; or unintentionally due to a service failure or connectivity failure).

History
<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Alert.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Alert

Parameters

Enum Definitions for LocalApplicationID and RemoteApplicationID

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>CallManager</td>
</tr>
<tr>
<td>200</td>
<td>CTI</td>
</tr>
</tbody>
</table>
Recommended Action

In the Cisco Unified Reporting tool, run a CM Cluster Overview report and check to see if all servers can talk to the Publisher. Also check for any alarms that might have indicated a CallManager failure and take appropriate action for the indicated failure. If the node was taken out of service intentionally, bring the node back into service.

**SocketError**

Failed to open network connection for receiving file requests. This usually happens when the IP address that the TFTP service uses to open the network connection is invalid.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kSocketError.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_TFTP-TFTP

**Cisco Unified Serviceability Alarm Definition Catalog**

System/TFTP

**Severity**

Alert (1)

**Parameters**

Error [Int] Reason [String]

**Recommended Action**

Verify that the TFTP service parameter, TFTP IP Address, accurately specifies the IP address of the NIC card to use for serving files via TFTP. See the help for the (advanced) TFTP IP Address service parameter for more information. If the problem persists, go to Cisco Unified Serviceability and enable Detailed level traces in the Trace Configuration window for the TFTP service and contact the Cisco Technical Assistance Center (TAC).

**StopBitConfigurationError**

The Cisco Messaging Interface service parameter, Stop Bits, has an invalid configuration.

An invalid stop bit has been configured for the serial port that CMI uses to connect to the voice messaging system. It is possible that the Stop Bits value has been updated via AXL or a CLI command where validation of the value was not performed. For this reason, it is best to set this value in the Service Parameter Configuration window in Cisco Unified CM Administration and the value can be validated against the accepted range of values for this field.
### Alert-Level Alarms

#### Cisco Unified Serviceability Alarm Definition Catalog

**CMIAlarmCatalog/CMI**

**Severity**

ALERT

**Routing List**

Event Log

SDI

**Parameter(s)**

Illegal Stop Bit(String)

**Recommended Action**

Verify that the Cisco Messaging Interface service parameter Stop Bits is set to a valid (allowable) value.

---

**TFTPServerListenSetSockOptFailed**

Failed to increase the size of the network buffer for receiving file requests. This usually indicates a lack of memory when there is a system issue such as running out of resources.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>New name changed from kStopBitConfigurationError.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_TFTP-TFTP

**Cisco Unified Serviceability Alarm Definition Catalog**

System/TFTP

**Severity**

Alert (1)

**Parameters**

Error [Int] IPAddress [String] Port [Int]
Recommended Action
Use RTMT to monitor the system memory resources and consumption and correct any system issues that might be contributing to a reduced amount of system resources.

**TFTPServerListenBindFailed**

Fail to connect to the network port through which file requests are received. This usually happens if the network port is being used by other applications on the system or if the port was not closed properly in the last execution of TFTP server.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kTFTPServerListenBindFailed.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_TFTP-TFTP

**Cisco Unified Serviceability Alarm Definition Catalog**
System/TFTP

**Severity**
Alert (1)

**Parameters**
Error [Int] IPAddress [String] Port [Int]

**Recommended Action**
Verify that the port is not in use by other application. After stopping the TFTP server, at the command line interface (CLI) on the TFTP server, execute the following command—show network status listen. If the port number specified in this alarm is shown in this CLI command output, the port is being used. Restart the Cisco Unified Communications Manager system, which may help to release the port. If the problem persists, go to Cisco Unified Serviceability and enable Detailed level traces in the Trace configuration window for the TFTP service and contact the Cisco Technical Assistance Center (TAC).

**TestAlarmAlert**

Testing alert alarm.

**Facility/Sub-Facility**
CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Test
Alert-Level Alarms

Severity
Alert (1)

Recommended Action
None

TLSConnectionToIMEFailed

TLS Failure to IME service.
A TLS connection to the IME server could not be established because of a problem with the certificate presented by the IME server. (For example, not in the Unified CM CTL, or is in the CTL but has expired).

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>New Alarm for this release.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity
ALERT

Recommended Action
Check to see that the certificate of the IME server is configured properly in the Unified CM.

Routing List
SDL
SDI
Sys Log
Event Log
Alert Manager

Parameter(s)
SSLErrorCode(UInt)
SSLErrorText(String)

TVSServerListenBindFailed

Fail to connect to the network port through which file requests are received. This usually happens if the network port is being used by other applications on the system or if the port was not closed properly in the last execution of TVS server.
Cisco Unified Serviceability Alarm Catalog
System/TVS

Severity
ALERT

Routing List
SDI
Event Log
Data Collector
Sys Log

Parameter(s)
nError(Int)
IPAddress(String)
Port(Int)

Recommended Action
Verify that the port is not in use by other application. After stopping the TVS server, at the command line interface (CLI) on the TVS server, execute the following command: show network status listen. If the port number specified in this alarm is shown in this CLI command output, the port is being used. Restart the Cisco Unified Communications Manager system, which may help to release the port. If the problem persists, go to Cisco Unified Serviceability and enable Detailed level traces in the Trace Configuration window for the TVS service and contact the Cisco Technical Assistance Center (TAC).

TVSServerListenSetSockOptFailed

Failed to increase the size of the network buffer for receiving file requests. This usually indicates a lack of memory when there is a system issue such as running out of resources.

Cisco Unified Serviceability Alarm Catalog
System/TVS

Severity
ALERT

Routing List
SDI
Event Log
Data Collector
Sys Log

Parameter(s)
nError(Int)
IPAddress(String)
Port(Int)
Recommended Action
Use RTMT to monitor the system memory resources and consumption and correct any system issues that might be contributing to a reduced amount of system resources.

UnknownException
Unknown error while connecting to database.
When CMI service is started, it tries to read CMI service parameters from DB. During this, if there is an unknown error, CMI triggers this alarm.

Cisco Unified Serviceability Alarm Definition Catalog
CMIAlarmCatalog/CMI

Severity
ALERT

Routing List
Event Log
SDI

Recommended Action
Report to Customer Service representative.

VMDNConfigurationError
The Voice Mail DN for CMI is invalid.
CMI cannot register with Cisco Unified Communications Manager because of an invalid Voice Mail DN. This alarm occurs because the Cisco Messaging Interface service parameter, Voice Mail DN, is empty or has invalid characters other than digits (0-9). It is possible that the Voice Mail DN value has been updated via AXL or a CLI command where validation of the value was not performed. For this reason, it is best to set this value in the Service Parameter Configuration window in Cisco Unified CM Administration and the value can be validated against the accepted range of values for this field.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kVMDNConfigurationError.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CMIAlarmCatalog/CMI

Severity
ALERT
Critical-Level Alarms

The critical-level alarm equals 2 and action may need to be taken immediately; auto-recovery is expected, but monitor the condition.

This alarm acts similar to the alert-level alarm but not necessarily requiring an immediate action. A system-affecting service had a failure but recovered without intervention. Examples follow:

- Service crashed due to an error that could not be handled but a watchdog process exists that will restart the service. The crash does not necessarily require immediate action. Examples are:
  - Out of memory conditions
  - Uninitialized variables
  - Memory scribblers
- Unexpected code error occurred that could not be handled but for which the system automatically restarts.

BChannelOOS

The B-channel is out of service. The B-channel indicated by this alarm has gone out of service. Some of the more common reasons for a B-channel to go out of service include are as follows:

- Taking the channel out of service intentionally to perform maintenance on either the near- or far-end
- MGCP gateway returns an error code 501 or 510 for a MGCP command sent from Cisco Unified Communications Manager (Cisco Unified CM)
- MGCP gateway does not respond to an MGCP command sent by Cisco Unified CM three times
- Speed and duplex mismatch exists on the Ethernet port between Cisco Unified CM and the MGCP gateway.

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed severity level from Error to Critical.</td>
</tr>
</tbody>
</table>
Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Critical

Routing List
SDL
SDI
Sys Log
Event Log

Parameters
Unique channel Id [String] Device Name. [String] Reason. [Enum]Channel Id. [UInt]

Enum Definitions
• 0—None Defined

Recommended Action
Check the Cisco Unified CM advanced service parameter, Change B-channel Maintenance Status to determine if the B-channel has been taken out of service intentionally; Check the Q.931 trace for PRI SERVICE message to determine whether a PSTN provider has taken the B-channel out of service; Reset the MGCP gateway; Check the speed and duplex settings on the Ethernet port.

CallManagerFailure

Indicates an internal failure in the Cisco Unified Communications system. The service should restart in an attempt to clear the failure.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Critical.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Critical
Critical-Level Alarms

Parameters
Additional Text [Optional] [String] Host name of hosting node. [String] IP address of hosting node. [String] Reason code. [Enum]

Enum Definitions

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Unknown—Unified CM has failed for an unknown reason.</td>
</tr>
<tr>
<td>2</td>
<td>HeartBeatStopped—An internal heart beat has stopped after the preceding heart beat interval.</td>
</tr>
<tr>
<td>3</td>
<td>RouterThreadDied—An internal thread has failed.</td>
</tr>
<tr>
<td>4</td>
<td>TimerThreadDied—An internal thread has failed.</td>
</tr>
<tr>
<td>5</td>
<td>CriticalThreadDied—An internal thread has failed.</td>
</tr>
</tbody>
</table>

Recommended Action
Monitor for other alarms and restart the Cisco CallManager service, if necessary. Collect the existing trace files in case the alarm persists.

CISCO-CCM-MIB
Part of ccmCallManagerAlarmEnable. See CISCO-CCM-MIB, page 7-1 in Chapter 7, “Cisco Management Information Base.”

CertExpiryCritical
Certificate is about to expire in less than 7 days. Regenerate or reimport certificate. Name of the service generating this alarm is Cisco Certificate Expiry Monitor. The alarms are generated when any certificate generated by the system or uploaded into the system expires. Cisco Unified CM uses certificates for Tomcat (Web Server), CallManager, IPSEc and Directory. Refer Security guide for more details on various certificates. When a certificate generated by Cisco Unified CM, the default validity of the self-signed certificate is for 5 years. In case of Certificates signed by a CA, the validity is dependent on the Expiry date set by CA while issuing the certificate. Once a certificate is about to expire “Cisco Certificate Expiry Monitor” service generates alarms. The severity of the alarm is dependent on how much time is left for the certificate to expire.

The impact to system operation depends on the which certificate expired. This information is contained in the alarm. If Tomcat certificate expired, while connecting to Cisco Unified CM web pages, browser will throw an error stating certificate has expired. One can still ignore the warning and continue to connect to Cisco Unified CM pages.

In case of Directory-trust, if Directory trust certificate uploaded to Cisco Unified CM expires, Cisco Unified CM may not be able to establish SSL connection with external LDAP server. The overall impact is that SSL connection between Cisco Unified CM and other external Servers will fail.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
</tbody>
</table>
Critical-Level Alarms

Facility/Sub-Facility
/CERT

Cisco Unified Serviceability Alarm Definition Catalog
System/Cert Monitor

Severity
Critical (2)

Parameters
None

Recommended Action
Login to CUOS page. Go to Security->Certificate Management and regenerate the certificate that has expired (based on the information in alarm). This will generate a new self-signed certificate with a new expiry date. In case the certificate is signed by a CA, Generate a new CSR, send it to the CA, get the certificate signed by CA and upload the new certificate.

CertValidfor7days

Alarm indicates that the certificate has expired or expires in less than seven days.

Cisco Unified Serviceability Alarm Definition Catalog
System/CertMonitorAlarmCatalog

Severity
Critical(2)

Routing List
Event Log
Sys Log

Parameters
Message(String)

Recommended Action
Regenerate the certificate that is about to expire by accessing the Cisco Unified Operating System and go to Certificate Management. If the certificate is issued by a CA, generate a CSR, submit the CSR to CA, obtain a fresh certificate from CA, and upload it to Cisco Unified CM.

CDRMaximumDiskSpaceExceeded

The CDR files disk usage exceeded maximum disk allocation. Some undeliverable files may have been deleted to bring disk usage down. The CDR files disk usage has exceeded the maximum allocated disk space. CDRM may have deleted some CDR files that have not been sent to the outside billing servers.
yet, in order to bring the disk usage down to below High Water Mark. The decision whether to delete undeliverable files or not depends on how deletionDisable flag is configured at CDRM Configuration page. E-mail alert will be sent to the admin.

### Critical-Level Alarms

**Facility/Sub-Facility**

CDRREP

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CDR Management

**Severity**

Critical (2)

**Routing List**

Event Log

Sys Log

Alert Manager

**Parameters**

DiskUsageInMB [String]

**Recommended Action**

1. Check if there are too many undeliverable CDR files accumulated due to some condition.
2. Check network link status.
3. Check if billing server is alive.
4. Check if (s)FTP Server on the billing server is running and accepting request.
5. Check if CDRM Configuration for billing servers is correct - under serviceability->tools.
6. Check if CDR files maximum disk allocation is too low - under serviceability->tools.
7. Check CDR Repository Manager trace under /var/log/active/cm/trace/cdrrep/log4j.

### CiscoDirSyncProcessFailToStart

LDAPSsync process failed to start on particular sync agreement.

**Facility/Sub-Facility**

CCM_JAVA_APPS-TOMCATAPPLICATIONS
**Critical-Level Alarms**

**Cisco Unified Serviceability Alarm Definition Catalog**

*System/Java Applications*

**Severity**
Critical (2)

**Parameters**
AgreementId [String]

**Recommended Action**
See application logs for error

---

**CodeRedEntry**

Unified CM has entered Code Red condition and will restart.

Unified CM has been in Code Yellow state for an extended period and is unlikely to recover on its own. The Cisco CallManager service automatically restarts in an attempt to clear the condition that is causing the Code Yellow state. The amount of time that the system will remain in Code Yellow state is configurable in the Code Yellow Duration service parameter. If the duration of this parameter is set to 99999, Code Red condition will never occur.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Critical.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**

*CallManager/CallManager*

**Severity**
Critical

**Parameters**

**Recommended Action**
You should have attempted the steps in the recommended actions defined in the CodeYellowEntry alarm. If you have not, try those after the system is online. There is no other action for Code Red because the only action is to restart which is performed for you automatically.
CodeYellowEntry

CallManager has initiated call throttling due to unacceptably high delay in handling incoming calls.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Critical.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity

Critical

Parameters


Recommended Action

Memory problems or high CPU usage are generally at the root of a Code Yellow state. A bad disk could also be the cause. Also, trace level settings can consume tremendous amounts of CPU (especially when the Enable SDL TCP Event Trace checkbox is enabled on the SDL Trace Configuration window in Cisco Unified Serviceability). Check these areas to try to correct the Code Yellow condition. You can also determine the level of fragmentation on the hard disk by issuing the File Fragmentation command from the CLI for the trace directories. Monitor the situation and collect existing trace files. If the CodeYellowExit alarm is not issued in a reasonable amount of time as deemed by your organization, or if the system is frequently entering Code Yellow state, contact TAC and supply the trace information you have collected.

CoreDumpFileFound

The new core dump files have been found in the system. One of the component has crashed and generated a core dump. Use admin cli or RTMT to fetch the backtrace.

Facility/Sub-Facility

CCM_TCT-LPMTCT

Cisco Unified Serviceability Alarm Definition Catalog

System/LpmTct

Severity

Critical (2)
Critical-Level Alarms

Parameters

Recommended Action
This serious internal error should be investigated by the Cisco Technical Assistance Center (TAC). Before contacting TAC, Login to cli on CCM serve and run "active analyze core file name" to generate the backtrace of the core dump. The core file name is listed in the alert details. After the analyze command is executed, collect the backtrace using cli command "file get activelog analyze" or "Collect Traces" option from RTMT. Send these backtraces to Cisco TAC for further analysis.

DChannelOOS

The D-channel is out of service. D-channel indicated by this alarm has gone out of service. Common reasons for a D-channel going out of service include losing T1/E1/BRI cable connectivity; losing the gateway data link (Layer 2) due to an internal or external problem; or gateway reset.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Critical.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Critical

Parameters
Channel Id. [UInt] Unique channel Id [String] Device Name. [String] Device IP address [String] Reason. [Enum]

Enum Definitions
• 0—None Defined

Recommended Action
Check the connection of the T1/E1/BRI cable; reset the gateway to restore Layer 2 connectivity; investigate whether the gateway reset was intentional. If the reset was not intentional, take steps to restrict access to the Gateway Configuration window in Cisco Unified Communications Manager Administration and the gateway terminal.
DUPLEX_MISMATCH

This alarm is generated by Cisco CDP whenever there is a duplex mismatch between local interface and switch interface.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>Added DUPLEX_MISMATCH to the CDPAlarmCatalog.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CDP/CDP

Cisco Unified Serviceability Alarm Definition Catalog
System/CDP

Severity
Critical (2)

Parameters
Switch Duplex Settings(String)
Local Interface Duplex Settings(String)

Recommended Action
Ensure that duplex settings are set to auto or full on local interface as well as switch interface.

ErrorChangeNotifyClientBlock

A change notification client is busy (blocked). If the change notification client continues to be blocked for 10 minutes, the system automatically clears the block and change notification should resume successfully. Changes made to the database are not being consumed by one of the recipients. This does not always represent an issue. However, if the change notification client continues to be blocked for 10 minutes, the system automatically clears the block for all clients except the blocked one, which means that change notifications should resume successfully for all other clients. To clear the blocked client, you must restart the server.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed severity level to Critical from Error.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_DB_LAYER-DB
Critical-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
System/DB

Severity
Critical (2)

Recommended Action
At the command line interface (CLI) on the database server, execute the following command:

```bash
show tech notify
```

The CLI command output will provide information about the block. Use Cisco Unified Serviceability to restart the server that was indicated in the alarm. You may also want to gather traces to examine them for anomalous activity during the time that client was blocked. In Cisco Unified Serviceability, enable Detailed level traces in the Trace Configuration window for the Cisco Database Layer Monitor service. Also, use RTMT to look for a change that may have occurred around the time of the alarm.

LogPartitionHighWaterMarkExceeded

The percentage of used disk space in the log partition has exceeded the configured high water mark. Some of the core file and / or trace files will be purged until the percentage of used disk space in the log partition gets below the configured low water mark.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Critical.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TCT-LPMTCT

Cisco Unified Serviceability Alarm Definition Catalog
System/LpmTct

Severity
Critical

Parameters
UsedDiskSpace [String] MessageString [Optional]. [String]

Recommended Action
Login into RTMT and check the configured threshold value for LogPartitionHighWaterMarkExceeded alert in Alert Central. If the configured value is set to a lower than the default threshold value unintentionally, change the value to default.

If you continue to receive this alert for half an hour after receiving the 1st alert, check for the disk usage for Common partition under "Disk Usage" tab in RTMT. If the disk usage shown under that tab is higher than configured value in LogPartitionLowWaterMarkExceeded alert configuration, contact Cisco TAC to troubleshoot the cause of high disk usage in Common partition.
MaxCallsReached

The maximum number of simultaneous connections in a Cisco Unified Communications Manager (Unified CM) node has been reached. This is an internally-set value and when it is exceeded, Unified CM starts throttling calls to keep the number of calls below the internal threshold.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Critical.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity

Critical

Parameters

Description [Int]

Recommended Action

In the Real-Time Monitoring Tool, check the CallsActive counter in the Cisco CallManager object for an unusually high number of calls. Internal mechanisms will attempt to correct this condition. If this alarm continues to occur, collect existing SDL and CCM trace files and check to be sure that CM Services trace collection in Cisco Unified CM Serviceability is set to Detailed level.

MGCPGatewayLostComm

The MGCP gateway is no longer in communication with Cisco Unified Communications Manager (Cisco Unified CM). This could occur because Cisco Unified CM receives an MGCP unregister signal from the gateways such as RSIP graceful/forced; Cisco Unified CM doesn't receive the MGCP KeepAlive signal from the gateway; the MGCP gateway doesn't response to an MGCP command sent by Cisco Unified CM three times; a speed and duplex mismatch exists on the Ethernet port between Cisco Unified CM and the MGCP gateway; the gateway has reset.

Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity

Critical (2)
Parameters
Device Name [String]

Recommended Action
Reset the MGCP gateway in an attempt to restore communication with Cisco Unified CM; check the speed and duplex settings on the Ethernet port. In the case of an unwanted reset of the gateway which caused communication to be lost, take precautions to ensure that no unauthorized personnel resets the gateway from Cisco Unified CM Administration or via the gateway terminal.

CISCO-CCM-MIB
See Chapter 7, “Cisco Management Information Base.”

StationTCPInitError
An error during initialization was encountered.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
</table>
| 8.0(1)                              | • Severity changed from Error to Critical.  
|                                     | • Following parameters are removed:  
|                                     | – Error Number [String]  
|                                     | – ErrorCode [Int] |

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Critical

Recommended Action
Verify the Cisco Unified Communications Manager IP address is configured and is not configured as the loop back address for the IP version. If the IP settings are correct, collect SDL and SDI traces and contact TAC.

TCPSetupToIMEFailed
Connection Failure to IME server.

This alarm occurs when Unified CM is unable to establish a TCP connection to an IME server. It typically occurs when the IP address and port of the IME server are misconfigured or an Intranet connectivity problem is preventing the connection from being set up.
Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity
CRITICAL_ALARM

Recommended Action
Check to make sure that the IP address and port of the IME server - which are present in the alarm - are valid. If so, this may be due to a network connectivity problem. Test the connectivity between the Unified CM servers and the IME server.

Routing List
SDL
SDI
Sys Log
Event Log
Alert Manager

Parameter(s)
IP address(String)
Port number(UInt)

TimerThreadSlowed

Verification of the Cisco Unified Communications Manager (Unified CM) internal timing mechanism has slowed beyond acceptable limits. This generally indicates an increased load on the system or an internal anomaly.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Warning to Critical.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Critical
Recommended Action
If this alarm occurs at the same general day or time, or if it occurs with increasing frequency, collect all system performance data in Real-Time Monitoring Tool as well as all trace information for the 30 minutes prior to the time that this alarm occurred and contact Cisco Technical Assistance Center (TAC).

TestAlarmCritical

Testing critical alarm.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
System/Test

Severity
Critical (2)

Recommended Action
None

Error-Level Alarms

The error-level alarm is 3 and you should investigate important devices or subsystems and determine if immediate action is needed. Errors that do not necessarily impact the ability of the service to continue to function and do not create a system outage. More related to device or subsystems.

An example would be a device or subsystem failing for an unexpected reason.

ANNDeviceRecoveryCreateFailed

ANN device recovery create failure. The ANN device recovery class create failed, possibly due to lack of memory. If the error code is non-zero it may help determine the cause of the error. The announcement device will not be available.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Added Routing List elements and Parameters.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS
AwaitingResponseFromPDPTimeout

Cisco Unified Communication Manager timed out waiting for the routing response from the policy decision point. Cisco Unified Communications Manager (Unified CM) did not receive a call routing response from the policy decision point (PDP) within the time specified by the Cisco CallManager service parameter, Call Intercept Routing Request Timer, or on the Call Intercept Profile Configuration window in Cisco Unified CM Administration.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
ERROR_ALARM

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Policy Decision Point(String)

Recommended Action
Check whether the PDP is in service and working normally. Verify that the PDP is not overloaded; if it is, take appropriate action to reduce the load on the PDP by following some or all of these recommendations:
Consider adding more PDPs and provisioning Unified CM with additional call intercept profiles and call intercept trigger points in the various configuration pages under the Call Routing menu in Cisco Unified CM Administration.

Provision a pair of policy servers per call-intercept profile to enable load balancing.

OR

Verify that the PDP server in your deployment meets or exceed the hardware requirements specified in the documentation for Cisco Enterprise Policy Manager (CEPM) or the third-party PDP solution you have deployed. If necessary, increase the value in the Cisco CallManager service parameter, Call Intercept Routing Request Timer or the value in the Call Intercept Profile for this PDP.

**BadCDRFileFound**

Bad CDR or CMR flat file found during CDR Load to CAR database. The file could be corrupted. However, CAR loader is able to skip the bad records and load the good ones to CAR database. The name of the service generating this alarm is CAR Loader (DailyCdrLoad) job. Part of Cisco CAR Scheduler service.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Existing parameters added.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_CAR_SCH-CAR

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CDR Rep

**Severity**

Error (3)

**Parameters**

File Name(String)
First Bad Record Cause(String)
File Summary(String)

**Recommended Action**

Find the bad file from the cdr_repository folders, and check its problematic record based on the information given by the cause and summary. Collect the associated SDI and SDL traces for the bad records found in this file as soon as possible. Collect and check the CAR Scheduler traces for more details.
BDIApplicationError

BDI Facility/Sub-Facility error.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Reason [String]

Recommended Action
See application logs for details

BDIOverloaded

BDI Facility/Sub-Facility overloaded.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Reason [String]

Recommended Action
See application logs for details.

CARSchedulerJobError

CAR scheduled job failed. A normal CAR scheduled job failed such as the pre-generated Daily/Weekly/Monthly/Monthly-Bill reports jobs. The particular CAR scheduler job that fails cannot be run properly. This does not cause any significant impact on CAR functions. For pre-generated CAR report, this would result failure to run on a particular report, which leads to missing of CAR report.
Error-Level Alarms

Facility/Sub-Facility
CCM_CAR_SCH-CAR

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CAR

Severity
Error (3)

Parameters
Job Name(String)
Job Failure Cause(String)
Job Failure Detail(String)

Recommended Action
1. Check the status of Cisco CAR Scheduler service.
2. Check the Event Log from CAR page.
3. Check the contents in tbl_system_preferences table.
4. Check the number of records in tbl_billing_data, tbl_billing_error, and tbl_error_id_map tables.
5. Check if the scheduled job configuration is correct from CAR page.
6. Collect and check the CAR Scheduler traces for more details.

CARSchedulerJobFailed

Critical CAR scheduled job failed. The jobs are PopulateSchedules, DailyCdrLoad, TaskMonitor, or DatabaseMaintenance. The particular CAR scheduler job that failed cannot be run properly. This can cause significant impact on CAR functions.

• If PopulateSchedules job fails, CAR scheduler cannot schedule jobs to run for the day; this would result some/all of CAR scheduler jobs cannot start.
• If DailyCdrLoad job fails, CAR loader would not be able to load CDR/CMR records from CDR/CMR flat files into CAR database; this would result records found upon running CAR reports, and causes accumulation of CDR/CMR flat files unprocessed.
• If TaskMonitor job fails, CAR scheduler will not be able to perform the daily DB IDS memory cleanup task; this would result higher DB shared memory usage.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Existing parameters added.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
</tbody>
</table>
• If DatabaseMaintenance job fails, CAR scheduler will not be able to perform the daily optimized database maintenance Update statistics procedures; this would result CAR database not optimized for its operations.

Name of the service generating this alarm is CAR Scheduler service.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Routing list changed from Data Collector to Alert Manager and existing parameters added.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CAR

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CAR Alarm Catalog

Severity

Error

Routing List

Event Log
Sys Log
Alert manager

Parameters

Job Name(String)
Job Failure Cause(String)
Job Failure Detail(String)

Recommended Action

1. Check the status of Cisco CAR DB service.
2. Check the status of Cisco CAR Scheduler service.
3. Check the Event Log from CAR page.
4. Check the contents in tbl_system_preferences table.
5. Check the number of records in tbl_billing_data, tbl_billing_error, and tbl_error_id_map tables.
6. Check if the scheduled job configuration is correct from CAR page.
7. Collect and check the CAR Scheduler traces for more details.
**CCDIPReachableTimeOut**

CCD Requesting Service IP Reachable Duration times out.

The CCD requesting service detected that it can no longer reach the learned patterns through IP. All learned patterns from this forward will be marked as unreachable (via IP) and to allow calls to learned patterns to continue to be routed until IP becomes reachable again, all calls to learned patterns will be routed through the PSTN. Calls can be routed through the PSTN for a certain period of time before PSTN failover times out.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

**Severity**
ERROR

**Routing List**
SDL
SDI
Sys Log
Event Log

**Recommended Action**
Check IP connectivity and resolve any TCP or IP problems in the network.

**CCDPSTNFailOverDurationTimeOut**

The internal limit on PSTN failover has expired.

When learned patterns are not reachable through IP, Unified CM routes calls through the PSTN instead. Calls can be routed through PSTN for an internally-controlled duration. When this alarm occurs, the PSTN failover duration has expired and calls to learned patterns cannot be routed. All learned patterns will be purged from Unified CM.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

**Severity**
ERROR

**Routing List**
SDL
SDI
Sys Log
Event Log
Recommended Action

Troubleshoot your network to get IP connectivity restored. After IP connectivity is restored, Unified CM will automatically relearn Hosted DN patterns and calls to learned patterns will proceed through IP.

CDRAgentSendFileFailed

CDR Agent cannot send CDR files from CCM node to CDR Repository node within the CCM cluster because of timeout or other reasons. E-mail alert will be sent to the admin.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed Data Collector Routing List element to Alert Manager.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CDRREP

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CDR Rep

Severity

Error (3)

Routing List

Event Log
Sys Log
Alert Manager

Parameters

CDRRepositoryNodeAddress [String]
CDRAgentNodeAddress [String]

Recommended Action

1. Check network link status.
2. Check if CDR Repository node (first node in the cluster) is alive.
3. Check if CDR Repository Manager is activated on the first node.
4. Check CDRM Configuration under serviceability->tools.
5. Check CDR Agent trace on the specific node where error occurred.
6. Check CDR Repository Manager trace.
7. Check if the Publisher is being upgraded. If the CDRAgentSendFileFailureContinues alarm is no longer present, the condition is corrected.
CDRAgentSendFileFailureContinues

CDR Agent cannot send CDR files from CCM node to CDR Repository node on retries. CDR Agent cannot send CDR files on retries after the initial failure from CCM node to CDR Repository node within the cluster.

**Facility/Sub-Facility**
CCM_CDR_REP-CDRREP

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/CDR Rep

**Severity**
Error

**Routing List**
Event Log
Sys Log
Data Collector

**Parameters**
CDRRepositoryNodeAddress [String]
CDRAgentNodeAddress [String]

**Recommended Action**
1. Check network link status.
2. Check if CDR Repository node (first node in the cluster) is alive.
3. Check if CDR Repository Manager is activated on the first node.
4. Check CDRM Configuration under serviceability->tools.
5. Check CDR Agent trace on the specific node where error occurred.
6. Check CDR Repository Manager trace.
7. Check if the Publisher is being upgraded.

CDRFileDeliveryFailed

FTP delivery of CDR files to the Billing Server outside of the cluster failed because of timeout or other reasons. E-mail alert will be sent to the admin.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed Data Collector Routing List element to Alert Manager.</td>
</tr>
</tbody>
</table>
Facility/Sub-Facility
CDRManagement/CDRREP

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CDR Rep

Severity
Error (3)

Routing List
Event Log
Sys Log
Alert Manager

Parameters
BillingServerAddress [String]

Recommended Action
1. Check network link status.
2. Check if billing server is alive.
3. Check if (s)FTP Server on the billing server is running and accepting request.
4. Check if CDRM Configuration is correct under Serviceability > Tools.
5. Check CDR Repository Manager trace.

CDRFileDeliveryFailureContinues

(s)FTP delivery of CDR files failed on retries to the Billing Server outside of the cluster failed on retries after the initial failure.

Facility/Sub-Facility
CCM_CDR_REP-CDRREP

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CDR Rep

Severity
Error (3)

Routing List
Event Log
Sys Log
Data Collector

Parameters
BillingServerAddress [String]
Error-Level Alarms

Recommended Action
1. Check network link status.
2. Check if billing server is alive.
3. Check if (s)FTP Server on the billing server is running and accepting request.
4. Check if CDRM Configuration is correct - under Serviceability>tools.
5. Check CDR Repository Manager trace.

CFBDeviceRecoveryCreateFailed

The CFB device startup failed, possibly due to lack of memory. If the error code is non-zero it may help determine the cause of the error. The conference bridge device will not be available.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Added Routing List elements and Parameters.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM.MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Error (3)

Routing List
SDI
Event Log
Sys Log

Parameter(s)
OS Error Code(Int)
OS Error Description(String)

Recommended Action
Restart the Cisco IP Voice Media Streaming App service or restart server.

CiscoDhcpdFailure

DHCP Daemon stopped running. DHCP Daemon cannot be brought up due to configuration error or crash.
Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Reason [String]

Recommended Action
Check application log for errors and correct the configuration. May require restarting the application if nothing found during the previous steps.

CiscoDirSyncProcessFailedRetry

LDAPSync process failed on particular sync agreement.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
AgreementId [String] Reason [String]

Recommended Action
The sync process will automatic retry. See application logs for details.

CiscoDirSyncProcessFailedNoRetry

LDAPSync process failed on particular sync agreement

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)
Parameters
AgreementId [String] Reason [String]

Recommended Action
See application logs for details, the application will try to sync again in the next scheduled time.

**CiscoDirSyncProcessConnectionFailed**

LDAPSync process failed to connect to LDAP server.

**Facility/Sub-Facility**
CCM_JAVA_APPS-TOMCATAPPLICATIONS

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Java Applications

**Severity**
Error (3)

**Parameters**
AgreementId [String] LDAPHost [String] Reason [String]

**Recommended Action**
Ensure that the LDAP server is online. If SSL is used, please make sure the required certificate is available on local CM server. The application will automatically retry.

**CiscoDirSyncDBAccessFailure**

LDAPSync process failed to access local database.

**Facility/Sub-Facility**
CCM_JAVA_APPS-TOMCATAPPLICATIONS

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Java Applications

**Severity**
Error (3)

**Parameters**
AgreementId [String] Reason [String]

**Recommended Action**
Ensure that the local CallManager database is working properly. The failed sync process will restart at the next scheduled time.
CiscoLicenseManagerDown

License Manager Down and license provisioning will fail.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Reason [String]

Recommended Action
Restart License Manager service on specified node

CiscoLicenseRequestFailed

License Request Unsuccessful because it cannot fulfill the request.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Reason [String]

Recommended Action
See application logs for error

CiscoLicenseDataStoreError

License Database error because it cannot fulfill the request.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications
Error-Level Alarms

Severity
Error (3)

Parameters
Reason [String]

Recommended Action
See application logs for error.

CiscoLicenseInternalError

Licensing Internal Error.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Reason [String]

Recommended Action
See application logs for error.

CiscoLicenseFileError

License File Error due to an invalid or tampered license file.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Reason [String]

Recommended Action
See application logs, verify that the license file is proper.
Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Error-Level Alarms

**CLM_MsgIntChkError**

ClusterMgr message integrity check error. ClusterMgr has received a message which has failed a message integrity check. This can be an indication that another node in the cluster is configured with the wrong security password.

**Facility/Sub-Facility**
CCM_CLUSTERMANAGER/CLUSTERMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Cluster Manager

**Severity**
Error (3)

**Operating System**
Appliance

**Parameters**
Sender IP address(String)

**Recommended Action**
Verify message is coming from an expected IP address. Verify the security password on that node.

**CLM_UnrecognizedHost**

ClusterMgr unrecognized host. ClusterMgr has received a message from an IP address which is not configured as a node in this cluster.

**Facility/Sub-Facility**
CCM_CLUSTERMANAGER/CLUSTERMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Cluster Manager

**Severity**
Error (3)

**Operating System**
Appliance

**Parameters**
Node IP address(String)

**Recommended Action**
Verify that this IP address is currently configured as a server in this cluster.
Chapter 6 Cisco Unified Serviceability Alarms and CiscoLog Messages

Error-Level Alarms

ConfigItAllBuildFilesFailed

A complete rebuild of all device configuration files has failed. Probable causes of this alarm could be failure to access the Cisco Unified Communications Manager database, or misconfiguration of some devices.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kConfigItAllBuildFilesFailed.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Error.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_TFTP-TFTP

**Cisco Unified Serviceability Alarm Definition Catalog**

**System/TFTP**

**Severity**

Error

**Recommended Action**

In Cisco Unified Serviceability, enabled Detailed level traces in the Trace Configuration window for TFTP and Cisco Database Layer Monitor services. Also, use RTMT to look for errors that may have occurred around the time of the alarm.

ConfigItAllReadConfigurationFailed

Failed to retrieve enterprise parameter values from database when rebuilding all configuration files. This is usually caused by a failure to access the Cisco Unified Communications Manager database.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kConfigItAllReadConfigurationFailed.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Error.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_TFTP-TFTP

**Cisco Unified Serviceability Alarm Definition Catalog**

**System/TFTP**
Severity
Error

Recommended Action
In Cisco Unified Serviceability, enable Detailed level traces in the Trace Configuration window for TFTP and Cisco Database Layer Monitor services. Also, use RTMT to look for errors that may have occurred around the time of the alarm.

**ConfigThreadBuildFileFailed**

Failed to build all device configuration files at TFTP service startup. This is usually caused by database access failure.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kConfigThreadBuildFileFailed</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Error.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_TFTP-TFTP

**Cisco Unified Serviceability Alarm Definition Catalog**
System/TFTP

Severity
Error

Recommended Action
In Cisco Unified Serviceability, enable Detailed level traces in the Trace Configuration window for TFTP and Cisco Database Layer Monitor services. Also, use RTMT to look for errors that may have occurred around the time of the alarm.

**ConfigThreadCNCMGrpBuildFileFailed**

Failed to rebuild configuration files for changes in Cisco Unified Communications Manager Group settings. This is usually caused by a failure to access the Cisco Unified Communications Manager database.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kConfigThreadCNCMGrpBuildFileFailed.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Error.</td>
</tr>
</tbody>
</table>
Facility/Sub-Facility
CCM_TFTP-TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP

Severity
Error

Recommended Action
In Cisco Unified Serviceability, enable Detailed level traces in the Trace Configuration window for TFTP and Cisco Database Layer Monitor services. Also, use RTMT to look for errors that may have occurred around the time of the alarm.

ConfigThreadCNGrpBuildFileFailed

Failed to rebuild configuration files for changes at group level settings such as Device Pool or Common Device Config settings. This is usually caused by a failure to access the Cisco Unified Communications Manager database.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kConfigThreadCNGrpBuildFileFailed.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Error.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TFTP-TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP

Severity
Error

Recommended Action
In Cisco Unified Serviceability, enable Detailed level traces in the Trace Configuration window for TFTP and Cisco Database Layer Monitor services. Also, use RTMT to look for errors that may have occurred around the time of the alarm.

ConfigThreadReadConfigurationFailed

Failed to retrieve enterprise parameter values from database at TFTP service startup. This is usually caused by database access failure.
History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kConfigThreadReadConfigurationFailed.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Error.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TFTP-TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP

Severity
Error

Recommended Action
In Cisco Unified Serviceability, enable Detailed level traces in the Trace Configuration window for TFTP and Cisco Database Layer Monitor services. Also, use RTMT to look for errors that may have occurred around the time of the alarm.

**ConfigThreadUnknownExceptionCaught**

An exception is caught in the main processing routine. This alarm is sent in conjunction with other alarms for failure when building configuration files or when the TFTP service is attempting to retrieve the values in the system's enterprise parameters.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kConfigThreadUnknownExceptionCaught.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TFTP-TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP

Severity
Error (3)

Recommended Action
In Cisco Unified Serviceability, enable Detailed level traces in the Trace Configuration window for TFTP service. Also, use RTMT to look for errors that may have occurred around the time of the alarm.
**ConflictingDataIE**

A call has been rejected because the incoming PRI/BRI Setup message had an invalid IE.

A call has been rejected because an incoming PRI/BRI Setup message contained an invalid Coding Standard value in the Bearer Capability information element (IE). Unified CM only accepts PRI/BRI Setup messages with Coding Standard values of 0 or 1. When an invalid IE is received, Unified CM rejects the call setup and issues this alarm.

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CallManager

**Severity**

ERROR

**Routing List**

SDL
SDI
Sys Log
Event Log

**Parameter(s)**

Device Name(String)

**Recommended Action**

Notify the service provider responsible for sending the Setup message that an IE with Coding Standard values of 0 or 1 must be included in Setup messages.

**ConnectionFailure**

Cisco CallManager failed to open TLS connection for the indicated device. Possible reasons could be wrong "Device Security Mode" configured, wrong "X.509 Subject Name" configured or unsupported cipher algorithm.

**Facility/Sub-Facility**

CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CallManager

**Severity**

Error (3)

**Parameters**

## Enum Definitions for DeviceType

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CISCO_30SP+</td>
</tr>
<tr>
<td>2</td>
<td>CISCO_12SP+</td>
</tr>
<tr>
<td>3</td>
<td>CISCO_12SP</td>
</tr>
<tr>
<td>4</td>
<td>CISCO_12S</td>
</tr>
<tr>
<td>5</td>
<td>CISCO_30VIP</td>
</tr>
<tr>
<td>6</td>
<td>CISCO_7910</td>
</tr>
<tr>
<td>7</td>
<td>CISCO_7960</td>
</tr>
<tr>
<td>8</td>
<td>CISCO_7940</td>
</tr>
<tr>
<td>9</td>
<td>CISCO_7935</td>
</tr>
<tr>
<td>12</td>
<td>CISCO_ATA_186</td>
</tr>
<tr>
<td>20</td>
<td>SCCP_PHONE</td>
</tr>
<tr>
<td>61</td>
<td>H323_PHONE</td>
</tr>
<tr>
<td>72</td>
<td>CTI_PORT</td>
</tr>
<tr>
<td>115</td>
<td>CISCO_7941</td>
</tr>
<tr>
<td>119</td>
<td>CISCO_7971</td>
</tr>
<tr>
<td>131</td>
<td>SIP_TRUNK</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>302</td>
<td>CISCO_7989</td>
</tr>
<tr>
<td>307</td>
<td>CISCO_7911</td>
</tr>
<tr>
<td>308</td>
<td>CISCO_7941G_GE</td>
</tr>
<tr>
<td>309</td>
<td>CISCO_7961G_GE</td>
</tr>
<tr>
<td>335</td>
<td>MOTOROLA_CN622</td>
</tr>
<tr>
<td>336</td>
<td>BASIC_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>348</td>
<td>CISCO_7931</td>
</tr>
<tr>
<td>358</td>
<td>CISCO_UNIFIED_COMMUNICATOR</td>
</tr>
<tr>
<td>365</td>
<td>CISCO_7921</td>
</tr>
<tr>
<td>369</td>
<td>CISCO_7906</td>
</tr>
<tr>
<td>374</td>
<td>ADVANCED_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>375</td>
<td>CISCO_TELEPRESENCE</td>
</tr>
<tr>
<td>404</td>
<td>CISCO_7962</td>
</tr>
<tr>
<td>412</td>
<td>CISCO_3951</td>
</tr>
<tr>
<td>431</td>
<td>CISCO_7937</td>
</tr>
<tr>
<td>434</td>
<td>CISCO_7942</td>
</tr>
<tr>
<td>435</td>
<td>CISCO_7945</td>
</tr>
<tr>
<td>436</td>
<td>CISCO_7965</td>
</tr>
<tr>
<td>437</td>
<td>CISCO_7975</td>
</tr>
</tbody>
</table>
### Error-Level Alarms

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AuthenticationError</td>
</tr>
<tr>
<td>2</td>
<td>InvalidX509NameInCertificate</td>
</tr>
<tr>
<td>4</td>
<td>InvalidTLSCipher</td>
</tr>
</tbody>
</table>

#### Enum Reasons

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>446</td>
<td>CISCO_3911</td>
</tr>
<tr>
<td>468</td>
<td>CISCO_UNIFIED_MOBILE_COMMUNICATOR</td>
</tr>
<tr>
<td>478</td>
<td>CISCO_TELEPRESENCE_1000</td>
</tr>
<tr>
<td>479</td>
<td>CISCO_TELEPRESENCE_3000</td>
</tr>
<tr>
<td>480</td>
<td>CISCO_TELEPRESENCE_3200</td>
</tr>
<tr>
<td>481</td>
<td>CISCO_TELEPRESENCE_500</td>
</tr>
<tr>
<td>484</td>
<td>CISCO_7925</td>
</tr>
<tr>
<td>493</td>
<td>CISCO_9971</td>
</tr>
<tr>
<td>495</td>
<td>CISCO_6921</td>
</tr>
<tr>
<td>496</td>
<td>CISCO_6941</td>
</tr>
<tr>
<td>497</td>
<td>CISCO_6961</td>
</tr>
<tr>
<td>20000</td>
<td>CISCO_7905</td>
</tr>
<tr>
<td>30002</td>
<td>CISCO_7920</td>
</tr>
<tr>
<td>30006</td>
<td>CISCO_7970</td>
</tr>
<tr>
<td>30007</td>
<td>CISCO_7912</td>
</tr>
<tr>
<td>30008</td>
<td>CISCO_7902</td>
</tr>
<tr>
<td>30016</td>
<td>CISCO_IP_COMMUNICATOR</td>
</tr>
<tr>
<td>30018</td>
<td>CISCO_7961</td>
</tr>
<tr>
<td>30019</td>
<td>CISCO_7936</td>
</tr>
<tr>
<td>30035</td>
<td>IP_STE</td>
</tr>
</tbody>
</table>

#### ConnectionFailureToPDP

A connection request from Unified CM to the policy decision point (PDP) failed. The failure may have been a result of the following conditions:

- Network error causing limited or no connectivity between Unified CM and the PDP
- Authentication errors when Unified CM established an HTTPS connection to the PDP

Check the Security profile of the indicated device. Make sure "Device Security Mode" is either "Authenticated" or "Encrypted". Make sure "X.509 Subject Name" field has the right content. It should match the Subject Name in the certificate from the peer. Unified CM only supports AES_128_SHA cipher algorithm. Let the peer regenerate its certificate with the right algorithm.
• PDP was not in service.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Error(3)

Routing List
SDL
SDI
Sys Log
Event Log
Data Collector

Parameters
Policy Decision Point(String)
The cause of the connection failure(String)

Recommended Action
Verify the network connectivity between Unified CM and the PDP by pinging the policy server host from Cisco Unified OS Administration and take steps to establish connectivity if it has been lost. If the connection failure is due to an authentication problem, verify that the valid certificate of the PDP has been imported to Cisco Unified OS Administration and certificates from every node in the Unified CM cluster have been imported to every node in the PDP. Also, check whether the PDP service is active.

CNFFBuffWriteToFilefopenfailed
Failed to create Config File on disk or update existing Config File on disk. This may happen if disk is full or the file is in use.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kCNFFBuffWriteToFilefopenfailed.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Error.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TFTP-TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP
Severity
Error

Parameters
FileName [String]

Recommended Action
Using RTMT, check the disk utilization and correct any issue discovered. If you do not discover a disk space issue, try restarting the TFTP service from Cisco Unified Serviceability (Tools > Control Center - Feature Services). Stopping and restarting the TFTP service is useful because the Config File that the TFTP service is trying to save might be an existing file that is in use. If you still get this error, go to Cisco Unified Serviceability and enable Detailed level traces in the Trace Configuration window for the TFTP service and contact the Cisco Technical Assistance Center (TAC).

**CNFFBuffWriteToFilefwritefailed**

Failed to save Config File to disk. This may happen if disk is full or the file is in use.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kCNFFBuffWriteToFilefwritefailed.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Error.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TFTP-TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP

Severity
Error

Parameters
FileName [String]

Recommended Action
Using RTMT, check the disk utilization and correct any issue discovered. If you do not discover a disk space issue, try restarting the TFTP service from Cisco Unified Serviceability (Tools > Control Center - Feature Services). Stopping and restarting the TFTP service is useful because the Config File that the TFTP service is trying to save might be an existing file that is in use. If you still get this error, go to Cisco Unified Serviceability and enable Detailed level traces in the Trace Configuration window for the TFTP service and contact the Cisco Technical Assistance Center (TAC).
**CtiProviderOpenFailure**

CTI application is unable to open the provider. The IP address is shown in either IPv4 or IPv6 format depending on the IP addressing mode of the application.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiProviderOpenFailure.</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CtiManager

**Severity**

ERROR

**Routing List**

SDL
SDI
Sys Log
Event Log
Data Collector

**Parameter(s)**

Login User Id(String)
Reason code.(Enum)
IP Address(String)
IPV6 Address(String)

**Enum Definitions - Reason Code**

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown</td>
</tr>
<tr>
<td>0x8CCC0075</td>
<td>Login request to authenticate user has timed out. Possible causes include LDAP server misconfiguration such as LDAP server referrals misconfiguration or Unified CM node experiencing high CPU usage. Recommended action is to verify the CPU utilization is in the safe range for Unified CM (this can be monitored using RTMT via CPU Pegging Alert)</td>
</tr>
<tr>
<td>(2362179701)</td>
<td></td>
</tr>
<tr>
<td>0x8CCC0060</td>
<td>Directory login failed. Verify that credentials are not misconfigured, check the userID and password configured in the application matches with what is configured in Unified CM Admin under (User Management &gt; End User or Application User)</td>
</tr>
<tr>
<td>(2362179680)</td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>0x8CCC005E</td>
<td>Directory is unavailable. Verify that the LDAP server is reachable from Unified CM node, make sure that the network connectivity between Unified CM and the LDAP server by pinging the LDAP server host from Cisco Unified OS Administration and take steps to establish connectivity if it has been lost</td>
</tr>
<tr>
<td>(2362179678)</td>
<td></td>
</tr>
<tr>
<td>0x8CCC00D1</td>
<td>Application is connecting to a non secure port but has security privileges enabled for the user associated with the application. Check the user group configuration for the user in Unified CM Admin under (User Management &gt; End User/Application User), select the user and verify the associated permissions information</td>
</tr>
<tr>
<td>(2362179793)</td>
<td></td>
</tr>
<tr>
<td>0x8CCC005F</td>
<td>Standard CTI Use permission is not enabled. Users associated with applications are required to be included in &quot;Standard CTI Enabled&quot; user group. Verify the user group configuration for the user in Unified CM Admin under (User Management &gt; End User/Application User), select the user and review the associated permissions information</td>
</tr>
<tr>
<td>(2362179679)</td>
<td></td>
</tr>
<tr>
<td>0x8CCC00D0</td>
<td>User is not enabled for a secure connection but the application connecting to secure port. Consider the application configuration and security configuration for the user, for TAPI applications review the Control Panel &gt; Phone and Modem Options &gt; Advanced &gt; select a CiscoTSP &gt; Configure... &gt; Security and disable &quot;Secure Connection to CTIManager&quot;. For JTAPI applications from JTPrefs choose Security and disable &quot;Enable Secure Connection&quot;. Also check the user group configuration for the user in Unified CM Admin under (User Management &gt; End User/Application User), select the user and verify the associated permissions information</td>
</tr>
<tr>
<td>(2362179792)</td>
<td></td>
</tr>
</tbody>
</table>

**Recommended Action**

Review the reason code and the recommended action within the reason code.

**DBLGetVersionInfoError**

DBL GetVersionInfo function returned NULL.

**Facility/Sub-Facility**

CCM_TCD-TCD

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/TCD SRV

**Severity**

Error (3)

**Recommended Action**

None
**DeviceTypeMismatch**

Device type mismatch between the information contained in the device TFTP config file and what is configured in the database for that device.

The device type indicated in the device configuration file does not match the database configuration. This could indicate that a change was made in the database configuration that failed to get updated at the device itself.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Following information is updated:</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for DBDeviceType</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for DeviceType</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CallManager

**Severity**

Error

**Parameters**

Database device type [Enum]Device type. [Enum]Name of device. [String]

**Enum Definitions for DBDeviceType**

<table>
<thead>
<tr>
<th>Code</th>
<th>Device Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CISCO_30SP+</td>
</tr>
<tr>
<td>2</td>
<td>CISCO_12SP+</td>
</tr>
<tr>
<td>3</td>
<td>CISCO_12SP</td>
</tr>
<tr>
<td>4</td>
<td>CISCO_12S</td>
</tr>
<tr>
<td>5</td>
<td>CISCO_30VIP</td>
</tr>
<tr>
<td>6</td>
<td>CISCO_7910</td>
</tr>
<tr>
<td>7</td>
<td>CISCO_7960</td>
</tr>
<tr>
<td>8</td>
<td>CISCO_7940</td>
</tr>
<tr>
<td>9</td>
<td>CISCO_7935</td>
</tr>
<tr>
<td>12</td>
<td>CISCO_ATA_186</td>
</tr>
<tr>
<td>20</td>
<td>SCCP_PHONE</td>
</tr>
<tr>
<td>61</td>
<td>H323 PHONE</td>
</tr>
</tbody>
</table>
### Error-Level Alarms

<table>
<thead>
<tr>
<th>Code</th>
<th>Device Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>72</td>
<td>CTI_PORT</td>
</tr>
<tr>
<td>115</td>
<td>CISCO_7941</td>
</tr>
<tr>
<td>119</td>
<td>CISCO_7971</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>302</td>
<td>CISCO_7989</td>
</tr>
<tr>
<td>307</td>
<td>CISCO_7911</td>
</tr>
<tr>
<td>308</td>
<td>CISCO_7941G_GE</td>
</tr>
<tr>
<td>309</td>
<td>CISCO_7961G_GE</td>
</tr>
<tr>
<td>335</td>
<td>MOTOROLA_CN622</td>
</tr>
<tr>
<td>336</td>
<td>BASIC_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>348</td>
<td>CISCO_7931</td>
</tr>
<tr>
<td>358</td>
<td>CISCO_UNIFIED_COMMUNICATOR</td>
</tr>
<tr>
<td>365</td>
<td>CISCO_7921</td>
</tr>
<tr>
<td>369</td>
<td>CISCO_7906</td>
</tr>
<tr>
<td>374</td>
<td>ADVANCED_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>375</td>
<td>CISCO_TELEPRESENCE</td>
</tr>
<tr>
<td>404</td>
<td>CISCO_7962</td>
</tr>
<tr>
<td>412</td>
<td>CISCO_3951</td>
</tr>
<tr>
<td>431</td>
<td>CISCO_7937</td>
</tr>
<tr>
<td>434</td>
<td>CISCO_7942</td>
</tr>
<tr>
<td>435</td>
<td>CISCO_7945</td>
</tr>
<tr>
<td>436</td>
<td>CISCO_7965</td>
</tr>
<tr>
<td>437</td>
<td>CISCO_7975</td>
</tr>
<tr>
<td>446</td>
<td>CISCO_3911</td>
</tr>
<tr>
<td>468</td>
<td>CISCO_UNIFIED_MOBILE_COMMUNICATOR</td>
</tr>
<tr>
<td>478</td>
<td>CISCO_TELEPRESENCE_1000</td>
</tr>
<tr>
<td>479</td>
<td>CISCO_TELEPRESENCE_3000</td>
</tr>
<tr>
<td>480</td>
<td>CISCO_TELEPRESENCE_3200</td>
</tr>
<tr>
<td>481</td>
<td>CISCO_TELEPRESENCE_500</td>
</tr>
<tr>
<td>484</td>
<td>CISCO_7925</td>
</tr>
<tr>
<td>493</td>
<td>CISCO_9971</td>
</tr>
<tr>
<td>495</td>
<td>CISCO_6921</td>
</tr>
<tr>
<td>496</td>
<td>CISCO_6941</td>
</tr>
<tr>
<td>497</td>
<td>CISCO_6961</td>
</tr>
<tr>
<td>20000</td>
<td>CISCO_7905</td>
</tr>
<tr>
<td>30002</td>
<td>CISCO_7920</td>
</tr>
<tr>
<td>30006</td>
<td>CISCO_7970</td>
</tr>
<tr>
<td>30007</td>
<td>CISCO_7912</td>
</tr>
<tr>
<td>Code</td>
<td>Device Type</td>
</tr>
<tr>
<td>------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>30008</td>
<td>CISCO_7902</td>
</tr>
<tr>
<td>30016</td>
<td>CISCO_IP_COMMUNICATOR</td>
</tr>
<tr>
<td>30018</td>
<td>CISCO_7961</td>
</tr>
<tr>
<td>30019</td>
<td>CISCO_7936</td>
</tr>
<tr>
<td>30035</td>
<td>IP_STE</td>
</tr>
</tbody>
</table>

**Enum Definitions for DeviceType**

<table>
<thead>
<tr>
<th>Code</th>
<th>Device Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CISCO_30SP+</td>
</tr>
<tr>
<td>2</td>
<td>CISCO_12SP+</td>
</tr>
<tr>
<td>3</td>
<td>CISCO_12SP</td>
</tr>
<tr>
<td>4</td>
<td>CISCO_12S</td>
</tr>
<tr>
<td>5</td>
<td>CISCO_30VIP</td>
</tr>
<tr>
<td>6</td>
<td>CISCO_7910</td>
</tr>
<tr>
<td>7</td>
<td>CISCO_7960</td>
</tr>
<tr>
<td>8</td>
<td>CISCO_7940</td>
</tr>
<tr>
<td>9</td>
<td>CISCO_7935</td>
</tr>
<tr>
<td>12</td>
<td>CISCO_ATA_186</td>
</tr>
<tr>
<td>20</td>
<td>SCCP_PHONE</td>
</tr>
<tr>
<td>61</td>
<td>H323_PHONE</td>
</tr>
<tr>
<td>72</td>
<td>CTI_PORT</td>
</tr>
<tr>
<td>115</td>
<td>CISCO_7941</td>
</tr>
<tr>
<td>119</td>
<td>CISCO_7971</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>302</td>
<td>CISCO_7989</td>
</tr>
<tr>
<td>307</td>
<td>CISCO_7911</td>
</tr>
<tr>
<td>308</td>
<td>CISCO_7941G_GE</td>
</tr>
<tr>
<td>309</td>
<td>CISCO_7961G_GE</td>
</tr>
<tr>
<td>335</td>
<td>MOTOROLA_CN622</td>
</tr>
<tr>
<td>336</td>
<td>BASIC_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>348</td>
<td>CISCO_7931</td>
</tr>
<tr>
<td>358</td>
<td>CISCO_UNIFIED_COMMUNICATOR</td>
</tr>
<tr>
<td>365</td>
<td>CISCO_7921</td>
</tr>
<tr>
<td>369</td>
<td>CISCO_7906</td>
</tr>
<tr>
<td>374</td>
<td>ADVANCED_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>375</td>
<td>CISCO_TELEPRESENCE</td>
</tr>
<tr>
<td>404</td>
<td>CISCO_7962</td>
</tr>
<tr>
<td>412</td>
<td>CISCO_3951</td>
</tr>
</tbody>
</table>
Error-Level Alarms

Recommended Action
Check the Unified CM Database Status report in Cisco Unified Reporting to verify that database replication is working. You can also go to Real-Time Reporting Tool (RTMT) and check the Replication Status in the Database Summary page. If status shows 2, then replication is working. Restart the phone to download a new configuration file from TFTP. Also, refer to the reason code definitions for additional recommended actions.

DbInfoCorrupt
Database information returned is corrupt. Database configuration error was encountered.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager
Severity
ERROR

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Name of Device(String)

Recommended Action
Investigate configuration for the identified device.

DbInfoError

Error in the database information retrieved. Database configuration error was encountered.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
ERROR

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Name of Device(String)

Recommended Action
Investigate configuration for identified device.

DbInfoTimeout

Database Information request timed out. Timeout was encountered while trying to read database configuration.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager
Severity
ERROR

Routing List
SDL
SDI
Event Log
Sys Log

Parameter(s)
Name of Device(String)

Recommended Action
Investigate configuration for identified device.

DeviceCloseMaxEventsExceeded

The TCP socket for the SCCP device has been closed due to excessive events in a 5-second period. Under normal conditions, the device will reregister automatically.

The indicated SCCP device exceeded the maximum number of events allowed per-SCCP device. Events can be phone calls, KeepAlive messages, or excessive SCCP or non-SCCP messages. The maximum number of allowed events is controlled by the Cisco CallManager service parameter, Max Events Allowed. When an individual device exceeds the number configured in that service parameter, Unified CM closes the TCP connection to the device; automatic reregistration generally follows. This action is an attempt to stop malicious attacks on Unified CM or to ward off excessive CPU usage.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Error (3)

Parameters
Total Events Received [UInt] IP Address [String] TCP Handle [String] Max Events Allowed [UInt] Number Of Skinny Device Throttled [UInt]

Recommended Action
Check the CCM trace data for the indicated SCCP device to determine the reason for the high number of events. Confirm that the value configured in the Cisco CallManager service parameter, Max Events Allowed, is a suitable number for your deployment.
**DeviceInitTimeout**

Device initialization timeout occurred. This alarm does not occur under normal working conditions; it will only occur if a device fails to respond to an initialize request.

**Facility/Sub-Facility**

CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CallManager

**Severity**

Error (3)

**Parameters**

Device Name [String] Protocol [String] Side Number [UInt]

**Recommended Action**

Investigate the identified device.

**DirSyncSchedulerFailedToUpdateNextExecTime**

Scheduler failed to update next execution time.

**Facility/Sub-Facility**

CCM_JAVA_APPS-TOMCATAPPLICATIONS

**Cisco Unified Serviceability Alarm Definition Catalog**

System/Java Applications

**Severity**

Error (3)

**Parameters**

Message [String]

**Recommended Action**

Check the DirSync configuration and logs

**DirSyncScheduledTaskFailed**

Directory synchronization task failed.

**Facility/Sub-Facility**

CCM_JAVA_APPS-TOMCATAPPLICATIONS
Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
SchedulerID [String] ErrorMessage [String]

Recommended Action
Check the DirSync configuration and logs

DirSyncSchedulerFailedToGetDBSchedules

Failed to get directory synchronization schedules from DB.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Message [String]

Recommended Action
Check the DirSync configuration and logs.

DirSyncSchedulerInvalidEventReceived

Invalid event received by DirSync scheduler from database.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Action [String] Message [String]
Recommended Action
Check the DirSync configuration and logs

DirSyncInvalidScheduleFound
Invalid schedule read by DirSync scheduler from database.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
ScheduleID [String]

Recommended Action
Check the DirSync configuration and logs

DirSyncSchedulerFailedToRegisterDBEvents
DirSync scheduler failed to register DB notifications.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
ScheduleTable [String]

Recommended Action
Check the DirSync configuration and logs

DirSyncSchedulerEngineFailedToStart
DirSync scheduler engine failed to start.
Error-Level Alarms

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
ScheduleTable [String]

Recommended Action
Check the DirSync configuration and logs

DirSyncScheduleDeletionFailed

DirSync schedule deletion request failed.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
ScheduleID [String]

Recommended Action
Check the DirSync configuration and logs

DirSyncScheduleUpdateFailed

DirSync schedule update request failed.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)
DRFMasterAgentStartFailure

DRF Master Agent was unable to start because it was unable to open port 4040.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>New name changed from CiscoDRFMasterAgentStartFailure. Routing List elements added. Descriptive text and recommended action changed.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog

System/DRF

Severity

Error

Routing List

Event Log

Sys Log

Parameters

Reason [String]

Recommended Action

Check if port 4040 is not already in use.

DRFLocalAgentStartFailure

DRF Local Agent was not able to start because it was unable to connect to the Master Agent on port 4040.
Chapter 6   Cisco Unified Serviceability Alarms and CiscoLog Messages

Error-Level Alarms

Facility/Sub-Facility
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Error (3)

Routing List
Event Log
Sys Log

Parameters
Reason [String]

Recommended Action
Check if the CiscoDRFMaster and CiscoDRFLocal services are running.

**DRFRestoreFailure**

DRF Restore process encountered errors.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>New name changed from CiscoDRFLocalAgentStartFailure. Routing List elements added. Descriptive text and recommended action changed.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF
Severity
Error (3)

Routing List
Event log
Sys Log

Parameters
Reason [String]

Recommended Action
Check DRF logs for further details.

**DRFInternalProcessFailure**

DRF internal process has some problems.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>New name changed from CiscoDRFInternalProcessFailure. Routing list added and recommended action changed.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

**Cisco Unified Serviceability Alarm Definition Catalog**
System/DRF

Severity
Error (3)

Routing List
Event Log
Sys Log

Parameters
Reason [String]

Recommended Action
Check DRF logs for details.
DRFTruststoreMissing

DRF uses ipsec truststore certificate for securing communication between the MA and LA service. This certificate is missing on the node, DRF LA will not be able to connect to MA.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFTruststoreMissing. Routing List elements added.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Error message removed.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Error (3)

Routing List
Event Log
Sys Log

Parameters
Reason(String)

Recommended Action
Download ipsec.pem file from Publisher and upload it as ipsec-trust only on the missing node then restart Cisco DRF Local service.

DRFUnknownClient

The DRF Master Agent running on the Publisher has received a Client connection request from an unknown server outside the cluster. The request has been rejected.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFUnknownClient. Routing List elements added.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Error message removed.</td>
</tr>
</tbody>
</table>
Facility/Sub-Facility
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Error (3)

Routing List
event Log
Sys Log

Parameters
Reason(String)

Recommended Action
Remove the suspect server from the network. Refer to the Reason section for suspect servers: Hostname and IP Address.

**DRFSecurityViolation**

The DRF System has detected a malicious pattern which could result in a security violation. The DRF Network Message contains a malicious pattern which could result in a security violation like code injection or directory traversal. DRF Network Message has been blocked.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFSecurityViolation. Routing List elements added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Error (3)

Routing List
Event Log
Sys Log
Parameters
Reason(String)

Recommended Action
Stop the Cisco DRF Master and Cisco DRF Local Agent Services.

**DRFBackupDeviceError**

DRF Backup process is failed due to backup device error.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFBackupDeviceError. Routing List elements added.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DRF

**Severity**

Error (3)

**Routing List**

Event Log
Sys Log

**Parameters**

Reason(String)

**Recommended Action**

Check if the proper device has been specified in the DRF configurations.

**DRFTapeDeviceError**

DRF is unable to access tape device.
**Facility/Sub-Facility**
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

**Cisco Unified Serviceability Alarm Definition Catalog**
System/DRF

**Severity**
Error (3)

**Routing List**
Event Log
Sys Log

**Parameters**
Reason(String)

**Recommended Action**
Check if tape drive is working properly and it contains a valid tape.

**DRFRestoreInternalError**

DRF Restore operation has encountered an error. Restore cancelled internally.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFTapeDeviceError. Routing List elements added.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

**Cisco Unified Serviceability Alarm Definition Catalog**
System/DRF
Error-Level Alarms

Severity
Error (3)

Routing List
Event Log
Sys Log

Parameters
Reason(String)

Recommended Action
Check DRF logs for details.

**DRFMABackupComponentFailure**

DRF was unable to backup at least one component because of an error.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFMABackupComponentFailure. Routing List elements added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Error (3)

Routing List
Event Log
Sys Log

Parameters
Reason(String)

Recommended Action
Check the component backup logs and contact support if needed.
**DRFMA\_RestoreComponentFailure**

DRF was unable to restore at least one component due to an error.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFMA_RestoreComponentFailure. Routing List elements added.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM\_DRF\_LOCAL & CCM\_DRF\_MASTER/DRF

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DRF

**Severity**

Error (3)

**Routing List**

Event Log

Sys Log

**Parameters**

Reason(String)

**Recommended Action**

Check the component restore logs and contact support if needed.

---

**DRFMA\_BackupNodeDisconnect**

The DRF Master Agent was running a backup operation on a CCM cluster, when one of the nodes disconnected before the backup operation was completed.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFMA_BackupNodeDisconnect. Routing List elements added.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM\_DRF\_LOCAL & CCM\_DRF\_MASTER/DRF
Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Error (3)

Routing List
Event Log
Sys Log

Parameters
Reason(String)

Recommended Action
Check the computer that disconnected during backup. If the computer was accidentally shutdown, restart the backup.

DRFNoRegisteredComponent

No registered components available, backup failed.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFNoRegisteredComponent. Routing List elements added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Error (3)

Routing List
Event Log
Sys Log

Parameters
Reason(String)

Recommended Action
Ensure at least one component is registered before attempting a backup.
DRFNoRegisteredFeature

No feature selected for backup.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFNoRegisteredFeature. Routing List elements added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog

System/DRF

Severity

Error (3)

Routing List

Event Log

Sys Log

Parameters

Reason(String)

Recommended Action

Ensure at least one feature is configured before attempting a backup.

DRFMARestoreNodeDisconnect

The node being restored disconnected from the Master Agent prior to being fully restored.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFMARestoreNodeDisconnect. Routing List elements added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF
Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Error (3)

Routing List
Event Log
Sys Log

Parameters
Reason(String)

Recommended Action
Check the computer that disconnected during restore. If the computer was accidentally shutdown, restart the restore.

DRFSftpFailure

DRF (s)FTP operation has failed.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFSftpFailure.</td>
</tr>
<tr>
<td></td>
<td>Routing List elements added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Error (3)

Routing List
Event Log
Sys Log

Parameters
Reason(String)
### DRFRegistrationFailure

DRF Registration operation failed due to an internal error.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFRegistrationFailure. Routing List elements added.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DRF

**Severity**

Error (3)

**Routing List**

Event Log

Sys Log

**Parameters**

Reason(String)

**Recommended Action**

Check the DRF logs and contact support if needed.

### DRFBackupCancel InternalError

DRF Backup operation has encountered an error. Backup cancelled internally.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFBackupCancelInternalError. Routing List elements added.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DRF

**Severity**

Error (3)

**Routing List**

Event Log

Sys Log

**Parameters**

Reason(String)

**Recommended Action**

Check the DRF logs and contact support if needed.
Facility/Sub-Facility
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Error (3)

Routing List
Event Log
Sys Log

Parameters
Reason(String)

Recommended Action
Check DRF logs for details.

**DRFLogDirAccessFailure**

DRF could not access the log directory.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFLogDirAccessFailure. Routing List elements added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Error (3)

Routing List
Event Log
Sys Log

Parameters
Reason(String)
**Recommended Action**
Ensure that the DRF user has required permission/足够 space on DRF Log and Trace directory.

**DRFFailure**

DRF Backup or Restore process has failed because it encountered errors.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFFailure. Changed Routing List element Data Collector to Alert Manager and added Sys Log.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DRF

**Severity**

Error (3)

**Routing List**

Event Log
Alert Manager
Sys Log

**Parameters**

Reason(String)

**Recommended Action**
Check DRF logs for further details.

**DRFLocalDeviceError**

DRF unable to access local device.

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DRF

**Severity**

ERROR
Routing List
Event Log
Sys Log

Parameter(s)
Reason(String)

Recommended Action
Check if local location exists and is accessible.

**DuplicateLearnedPattern**

This alarm occurs when CCD requesting service received a duplicate Hosted DN.
The Call Control Discovery (CCD) requesting service received the same hosted DN from multiple call control entities such as Unified CM Express or another Unified CM cluster. The Cisco CallManager service parameter, Issue Alarm for Duplicate Learned Patterns, controls whether this alarm gets issued.

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/CallManager

Severity
ERROR

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Client Handle(String)
Service ID(UInt)
Sub Service ID(UInt)
InstanceID1(UInt)
InstancID2(UInt)
InstanceID3(UInt)
InstanceID4(UInt)

Recommended Action
In RTMT, check the Pattern Report (CallManager > Report > Learned Pattern) and look for the duplicate pattern identified in this alarm. Learned patterns must be unique. Determine which call control entity (such as Unified CM or Unified CM Express) needs to be changed so that there is no duplicate pattern. Refer to the call control entity's configuration guide (help text) to learn how to update a hosted DN.
pattern. In Unified CM, to change the Hosted DN Pattern go to Cisco Unified CM Administration to update the Hosted DN Pattern configuration (Call Routing > Call Control Discovery > Hosted DN Patterns).

**EMAppInitializationFailed**

EM Application not started. Error occurred while starting application.

**Cisco Unified Serviceability Alarm Definition Catalog**

System/EMAlarmCatalog

**Severity**

ERROR

**Routing List**

Sys Log
Event Log
Data Collector

**Parameter(s)**

Servlet Name(String)

**Recommended Action**

Action See application logs for error. Default location for the logs are at /var/log/active/tomcat/logs/em/log4j/

**EMCCFailedInLocalCluster**

EMCC login failure occurred due to one of the following conditions:

- Devices are incompatible with EMCC.
- Unable to retrieve remote cluster information.
- EMCC is restricted by the local cluster.
- Untrusted certificate received from the remote end while trying to establish a connection

**Reason Codes:**

- 31—User is not enabled for EMCC
- 211/38—EMCC or PSTN is not activated in InterClusterServiceProfile page
- 23—User does not exist in the end user table
- 35—No remote cluster entry is present for the home cluster

**Cisco Unified Serviceability Alarm Definition Catalog**

System/EMAlarmCatalog

**Severity**

ERROR(3)
Routing List
Sys Log
Event Log
Alert Manager

Parameters
Device Name(String)
Login Date/Time(String)
Login UserID(String)
Reason(String)

Recommended Action
Perform the following steps:

---

Step 1
Validate if the device model supports EMCC.

Step 2
Ensure that every remote cluster added for EMCC has valid hostname/IP address for EM and PSTN access in the Remote Cluster administration window (From Unified CM Administration window, go to System -> EMCC -> Remote Cluster).

Step 3
Ensure that the entries are enabled.

Step 4
Ensure that a bundle of all Tomcat certificates (PKCS12) has been imported into the local tomcat-trust keystore (From the OS Administration window, go to Security -> Certificate Management and check the certificates in tomcat-trust).

---

EMServiceConnectionError

EM Service not reachable. EM Service might be down in one or more nodes in the cluster.

Cisco Unified Serviceability Alarm Definition Catalog
System/EMAAlarmCatalog

Severity
ERROR

Routing List
Sys Log
Event Log

Parameter(s)
Servlet Name(String)

Recommended Action
Check if Cisco Extension Mobility service is running on all nodes of the cluster where the service is activated.
EndPointTransientConnection

End point transient connection attempt.
A connection was established and immediately dropped before completing registration. Incomplete registration may indicate that a device is rehoming in the middle of registration. The alarm could also indicate a device misconfiguration, database error, or an illegal/unknown device trying to attempt a connection. Network connectivity problems can affect device registration, or the restoration of a primary Unified CM may interrupt registration.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
ERROR

Routing List
SDL
SDI
Sys Log
Data Collector
SNMP Traps
Alternate Syslog

Parameter(s)
Device IP address(String)
Device name(String)
Device MAC address(String)
Protocol(String)
Device type(Enum)
Reason Code(Enum)
Connecting Port(UInt)
Registering SIP User(String)
IPv6Address(String)
IPAddressAttributes(Enum)
IPv6AddressAttributes(Enum)

Enum Definitions -Device type

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CISCO_30SP+</td>
</tr>
<tr>
<td>2</td>
<td>CISCO_12SP+</td>
</tr>
<tr>
<td>3</td>
<td>CISCO_12SP</td>
</tr>
<tr>
<td>4</td>
<td>CISCO_12S</td>
</tr>
</tbody>
</table>
## Error-Level Alarms

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>CISCO_30VIP</td>
</tr>
<tr>
<td>6</td>
<td>CISCO_7910</td>
</tr>
<tr>
<td>7</td>
<td>CISCO_7960</td>
</tr>
<tr>
<td>8</td>
<td>CISCO_7940</td>
</tr>
<tr>
<td>9</td>
<td>CISCO_7935</td>
</tr>
<tr>
<td>12</td>
<td>CISCO_ATA_186</td>
</tr>
<tr>
<td>20</td>
<td>SCCP_PHONE</td>
</tr>
<tr>
<td>61</td>
<td>H323_PHONE</td>
</tr>
<tr>
<td>72</td>
<td>CTI_PORT</td>
</tr>
<tr>
<td>115</td>
<td>CISCO_7941</td>
</tr>
<tr>
<td>119</td>
<td>CISCO_7971</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>302</td>
<td>CISCO_7989</td>
</tr>
<tr>
<td>307</td>
<td>CISCO_7911</td>
</tr>
<tr>
<td>308</td>
<td>CISCO_7941G_GE</td>
</tr>
<tr>
<td>309</td>
<td>CISCO_7961G_GE</td>
</tr>
<tr>
<td>335</td>
<td>MOTOROLA_CN622</td>
</tr>
<tr>
<td>336</td>
<td>BASIC_3RD_PARTY_SIPDEVICE</td>
</tr>
<tr>
<td>348</td>
<td>CISCO_7931</td>
</tr>
<tr>
<td>358</td>
<td>CISCO_UNIFIED_COMMUNICATOR</td>
</tr>
<tr>
<td>365</td>
<td>CISCO_7921</td>
</tr>
<tr>
<td>369</td>
<td>CISCO_7906</td>
</tr>
<tr>
<td>374</td>
<td>ADVANCED_3RD_PARTY_SIPDEVICE</td>
</tr>
<tr>
<td>375</td>
<td>CISCO_TELEPRESENCE</td>
</tr>
<tr>
<td>404</td>
<td>CISCO_7962</td>
</tr>
<tr>
<td>412</td>
<td>CISCO_3951</td>
</tr>
<tr>
<td>431</td>
<td>CISCO_7937</td>
</tr>
<tr>
<td>434</td>
<td>CISCO_7942</td>
</tr>
<tr>
<td>435</td>
<td>CISCO_7945</td>
</tr>
<tr>
<td>436</td>
<td>CISCO_7965</td>
</tr>
<tr>
<td>437</td>
<td>CISCO_7975</td>
</tr>
<tr>
<td>446</td>
<td>CISCO_3911</td>
</tr>
<tr>
<td>468</td>
<td>CISCO_UNIFIED_MOBILE_COMMUNICATOR</td>
</tr>
<tr>
<td>478</td>
<td>CISCO_TELEPRESENCE_1000</td>
</tr>
<tr>
<td>479</td>
<td>CISCO_TELEPRESENCE_3000</td>
</tr>
<tr>
<td>480</td>
<td>CISCO_TELEPRESENCE_3200</td>
</tr>
<tr>
<td>481</td>
<td>CISCO_TELEPRESENCE_500</td>
</tr>
<tr>
<td>484</td>
<td>CISCO_7925</td>
</tr>
</tbody>
</table>
### Error-Level Alarms

<table>
<thead>
<tr>
<th>Enum Definitions -Reason Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
</tbody>
</table>
### Error-Level Alarms

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>InitializationError—An internal error occurred within Cisco Unified CM while processing the device registration. It is recommended to restart the Cisco CallManager service. If this occurs repeatedly, collect SDL/SDI detailed traces with &quot;Enable SIP Keep Alive (REGISTER Refresh) Trace&quot; and &quot;Enable SCCP Keep Alive Trace&quot; under Cisco CallManager services turned on and contact TAC.</td>
</tr>
<tr>
<td>8</td>
<td>DeviceInitiatedReset—Indicates that the error was due to device initiated reset.</td>
</tr>
<tr>
<td>9</td>
<td>CallManagerReset—Indicates that the error was due to call manager reset.</td>
</tr>
<tr>
<td>10</td>
<td>AuthenticationError—The device failed either TLS or SIP digest security authentication. If the device is a SIP phone and is enabled for digest authentication (on the System &gt; Security Profile &gt; Phone Security Profile, check if &quot;Enable Digest Authentication&quot; checkbox is checked), verify the &quot;Digest Credentials&quot; in the End User config page are configured. Also, check the phone config page to see if the phone is associated with the specified end user in the Digest User drop box. If the device is a third-party SIP device, verify the digest credentials configured on the phone match the &quot;Digest Credentials&quot; configured in the End User page.</td>
</tr>
<tr>
<td>11</td>
<td>InvalidX509NameInCertificate—Configured &quot;X.509 Subject Name&quot; doesn't match what's in the certificate from the device. Check the Security Profile of the indicated device and verify the &quot;Device Security Mode&quot; is either &quot;Authenticated&quot; or &quot;Encrypted&quot;. Verify the &quot;X.509 Subject Name&quot; field has the right content. It should match the Subject Name in the certificate from the peer.</td>
</tr>
<tr>
<td>12</td>
<td>InvalidTLSCipher—Unsupported cipher algorithm used by the device; Cisco Unified CM only supports AES_128_SHA cipher algorithm. Recommended action is for the device to regenerate its certificate with the AES_128_SHA cipher algorithm.</td>
</tr>
<tr>
<td>13</td>
<td>DirectoryNumberMismatch—Indicates mismatch between the directory number that the SIP device is trying to register with and the directory number configured in the Cisco Unified CM for the SIP device.</td>
</tr>
<tr>
<td>14</td>
<td>MalformedRegisterMsg—(SIP only) A SIP REGISTER message could not be processed because of an illegal format. Possible causes include a missing Call-ID header, a missing AoR in the To header, and an expires value too small. Verify the REGISTER message does not suffer from any of these ills.</td>
</tr>
<tr>
<td>15</td>
<td>ProtocolMismatch—The protocol of the device (SIP or SCCP) does not match the configured protocol in Cisco Unified CM.</td>
</tr>
<tr>
<td><strong>Recommended actions:</strong></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>Verify the device is configured with the desired protocol.</td>
</tr>
<tr>
<td>2.</td>
<td>Verify the firmware load ID on the Device Defaults page is correct and actually exists on the TFTP server</td>
</tr>
<tr>
<td>3.</td>
<td>If there is a firmware load ID configured on the device page, verify it is correct and exists on the TFTP server (On Cisco Unified OS Administration page, Software Upgrades &gt; TFTP File Management, look for the file name as specified by load ID).</td>
</tr>
<tr>
<td>4.</td>
<td>Restart the TFTP and Cisco CallManager services. Use the Cisco Unified OS Administration TFTP File Management page to verify the configured firmware loads exist.</td>
</tr>
<tr>
<td>16</td>
<td>DeviceNotActive—The device has not been activated.</td>
</tr>
<tr>
<td>17</td>
<td>AuthenticatedDeviceAlreadyExists—A device with the same name is already registered. If this occurs repeatedly, collect SDL/SDI detailed traces with &quot;Enable SIP Keep Alive (REGISTER Refresh) Trace&quot; and &quot;Enable SCCP Keep AliveTrace&quot; underCisco CallManager services turned on and contact TAC. There may be an attempt by unauthorized devices to register.</td>
</tr>
</tbody>
</table>
18  ObsoleteProtocolVersion—(SCCP only) A SCCP device registered with an obsolete protocol version. Power cycle the phone. Verify that the TFTP service is activated. Verify that the TFTP server is reachable from the device. If there is a firmware load ID configured on the Phone Config page, verify that the firmware load ID exists on the TFTP server. (On Cisco Unified OS Administration page, Software Upgrades > TFTP File Management, look for the file name as specified by load ID).

23  DatabaseTimeout—Cisco Unified CM requested device configuration data from the database but did not receive a response within 10 minutes.

25  RegistrationSequenceError—(SCCP only) A device requested configuration information from the Cisco Unified CM at an unexpected time. The Cisco Unified CM had not yet obtained the requested information.

26  InvalidCapabilities—(SCCP only) Cisco Unified CM detected an error in the media capabilities reported by the device during registration. The device reported the capabilities in the StationCapabilitiesRes message.

27  CapabilityResponseTimeout—(SCCP only) Cisco Unified CM timed out while waiting for the device to respond to a request to report its media capabilities.

28  SecurityMismatch—Cisco Unified CM detected a mismatch in the security settings of the device and/or the Unified CM. The following mismatches are detected:
   1. The device established a secure connection, yet reported that it does not have the ability to do authenticated signaling.
   2. The device did not establish a secure connection, but the security mode configured for the device indicates that it should have done so.
   3. The device established a secure connection, but the security mode configured for the device indicates that it should not have done so.

29  AutoRegisterDBError—(SCCP only) Auto-registration of a device failed for one of the following reasons:
   1. Auto-registration is not allowed for the device type.
   2. An error occurred in the auto-registration stored procedure.

30  DBAccessError—(SCCP only) Auto-registration of a device failed because of an error that occurred while building the station registration profile.

31  AutoRegisterDBConfigTimeout—(SCCP only) Cisco Unified CM timed out during auto-registration of a device. The registration profile of the device did not get inserted into the database in time.

32  DeviceTypeMismatch—(SCCP only) The device type reported by the device does not match the device type configured on the Cisco Unified CM.

33  AddressingModeMismatch—(SCCP only) Cisco Unified CM detected an error related to the addressing mode configured for the device. One of the following errors was detected:
   1. The device is configured to use only IPv4 addressing, but did not specify an IPv4 address.
   2. The device is configured to use only IPv6 addressing, but did not specify an IPv6 address.

**Enum Definitions - IPAddressAttributes**

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown—The device has not indicated what this IPv4 address is used for</td>
</tr>
</tbody>
</table>
Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Error-Level Alarms

Enum Definitions -IPv6AddressAttributes

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown—The device has not indicated what this IPv6 address is used for</td>
</tr>
<tr>
<td>1</td>
<td>Administrative only—The device has indicated that this IPv6 address is used for administrative communication (web interface) only</td>
</tr>
<tr>
<td>2</td>
<td>Signal only—The device has indicated that this IPv6 address is used for control signaling only</td>
</tr>
<tr>
<td>3</td>
<td>Administrative and signal—The device has indicated that this IPv6 address is used for administrative communication (web interface) and control signaling</td>
</tr>
</tbody>
</table>

Recommended Action

Investigate any network connectivity problems in the system. It's possible that you have reached the maximum number of devices; the Cisco Unified Communications Manager service parameter, Maximum Number of Registered Devices, controls the number of devices allowed in the system. Taking licensing, system hardware and other related concerns into consideration, you could increase the value of the service parameter. Also, refer to the reason code definitions for recommended actions. No action is required if this event was issued as a result of a normal device rehome.

EndPointUnregistered

An endpoint that has previously registered with Cisco Unified Communications Manager has unregistered. In cases of normal unregistration with reason code 'CallManagerReset', 'CallManagerRestart', 'DeviceInitiatedReset', 'EMLoginLogout', or 'EMCCLoginLogout', the severity of this alarm is lowered to INFORMATIONAL. An endpoint can unregister for many reasons, both intentional, such as manually resetting the device after a configuration change, or unintentional, such as loss of network connectivity. Other causes for this alarm could include a phone being registered to a secondary node and then the primary node come back online, causing the phone to rehome to the primary Cisco Unified CM node or lack of a KeepAlive being returned from the Cisco Unified CM node to which this endpoint was registered. Unregistration also occurs if Cisco Unified CM receives a duplicate registration request for this same device.

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity
ERROR

Routing List
SDL
SDI
Sys Log
Data Collector
SNMP Traps
Alternate Syslog

Parameter(s)
Device name(String)
Device MAC address(String)
Device IP address(String)
Protocol(String)
Device type(Enum)
Device description(String)
Reason Code(Enum)
IPV6Address(String)
IPAddressAttributes(Enum)
IPV6AddressAttributes(Enum)

Enum Definitions -Device type

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CISCO_30SP+</td>
</tr>
<tr>
<td>2</td>
<td>CISCO_12SP+</td>
</tr>
<tr>
<td>3</td>
<td>CISCO_12SP</td>
</tr>
<tr>
<td>4</td>
<td>CISCO_12S</td>
</tr>
<tr>
<td>5</td>
<td>CISCO_30VIP</td>
</tr>
<tr>
<td>6</td>
<td>CISCO_7910</td>
</tr>
<tr>
<td>7</td>
<td>CISCO_7960</td>
</tr>
<tr>
<td>8</td>
<td>CISCO_7940</td>
</tr>
<tr>
<td>9</td>
<td>CISCO_7935</td>
</tr>
<tr>
<td>12</td>
<td>CISCO_ATA_186</td>
</tr>
<tr>
<td>20</td>
<td>SCCP_PHONE</td>
</tr>
<tr>
<td>61</td>
<td>H323_PHONE</td>
</tr>
<tr>
<td>72</td>
<td>CTI_PORT</td>
</tr>
<tr>
<td>115</td>
<td>CISCO_7941</td>
</tr>
<tr>
<td>119</td>
<td>CISCO_7971</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>302</td>
<td>CISCO_7989</td>
</tr>
<tr>
<td>307</td>
<td>CISCO_7911</td>
</tr>
<tr>
<td>308</td>
<td>CISCO_7941G_GE</td>
</tr>
<tr>
<td>309</td>
<td>CISCO_7961G_GE</td>
</tr>
<tr>
<td>335</td>
<td>MOTOROLA_CN622</td>
</tr>
<tr>
<td></td>
<td>Error Description</td>
</tr>
<tr>
<td>---</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>336</td>
<td>BASIC_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>348</td>
<td>CISCO_7931</td>
</tr>
<tr>
<td>358</td>
<td>CISCO_UNIFORMED_COMMUNICATOR</td>
</tr>
<tr>
<td>365</td>
<td>CISCO_7921</td>
</tr>
<tr>
<td>369</td>
<td>CISCO_7906</td>
</tr>
<tr>
<td>374</td>
<td>ADVANCED_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>375</td>
<td>CISCO_TELEPRESENCE</td>
</tr>
<tr>
<td>404</td>
<td>CISCO_7962</td>
</tr>
<tr>
<td>412</td>
<td>CISCO_3951</td>
</tr>
<tr>
<td>431</td>
<td>CISCO_7937</td>
</tr>
<tr>
<td>434</td>
<td>CISCO_7942</td>
</tr>
<tr>
<td>435</td>
<td>CISCO_7945</td>
</tr>
<tr>
<td>436</td>
<td>CISCO_7965</td>
</tr>
<tr>
<td>437</td>
<td>CISCO_7975</td>
</tr>
<tr>
<td>446</td>
<td>CISCO_3911</td>
</tr>
<tr>
<td>468</td>
<td>CISCO_UNIFORMED_MOBILE_COMMUNICATOR</td>
</tr>
<tr>
<td>478</td>
<td>CISCO_TELEPRESENCE_1000</td>
</tr>
<tr>
<td>479</td>
<td>CISCO_TELEPRESENCE_3000</td>
</tr>
<tr>
<td>480</td>
<td>CISCO_TELEPRESENCE_3200</td>
</tr>
<tr>
<td>481</td>
<td>CISCO_TELEPRESENCE_500</td>
</tr>
<tr>
<td>484</td>
<td>CISCO_7925</td>
</tr>
<tr>
<td>493</td>
<td>CISCO_9971</td>
</tr>
<tr>
<td>495</td>
<td>CISCO_6921</td>
</tr>
<tr>
<td>496</td>
<td>CISCO_6941</td>
</tr>
<tr>
<td>497</td>
<td>CISCO_6961</td>
</tr>
<tr>
<td>20000</td>
<td>CISCO_7905</td>
</tr>
<tr>
<td>30002</td>
<td>CISCO_7920</td>
</tr>
<tr>
<td>30006</td>
<td>CISCO_7970</td>
</tr>
<tr>
<td>30007</td>
<td>CISCO_7912</td>
</tr>
<tr>
<td>30008</td>
<td>CISCO_7902</td>
</tr>
<tr>
<td>30016</td>
<td>CISCO_IP_COMMUNICATOR</td>
</tr>
<tr>
<td>30018</td>
<td>CISCO_7961</td>
</tr>
<tr>
<td>30019</td>
<td>CISCO_7936</td>
</tr>
<tr>
<td>30035</td>
<td>IP_STE</td>
</tr>
</tbody>
</table>
### Enum Definitions - Reason Code

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Unknown—The device has unregistered for an unknown reason. If the device does not reregister within 5 minutes, verify it is powered-up and verify network connectivity between the device and Cisco Unified CM.</td>
</tr>
<tr>
<td>2</td>
<td>NoEntryInDatabase—Device not configured properly in the Cisco Unified CM database.</td>
</tr>
<tr>
<td>3</td>
<td>DatabaseConfigurationError—Device configuration error in the Cisco Unified CM database.</td>
</tr>
<tr>
<td>4</td>
<td>DeviceNameUnresolvable—The Cisco Unified CM is unable to resolve the device name to an IP Address internally.</td>
</tr>
<tr>
<td>5</td>
<td>MaxDevRegExceeded—Maximum number of device registrations have been reached.</td>
</tr>
<tr>
<td>6</td>
<td>ConnectivityError—Network communication between the device and Cisco Unified CM has been interrupted. Possible causes include device power outage, network power outage, network configuration error, network delay, packet drops and packet corruption. It is also possible to get this error if the Cisco Unified CM node is experiencing high CPU usage. Verify the device is powered up and operating, verify network connectivity between the device and Cisco Unified CM, and verify the CPU utilization is in the safe range (this can be monitored using RTMT via CPU Pegging Alert).</td>
</tr>
<tr>
<td>7</td>
<td>InitializationError—Indicates that an error occurred when the Cisco Unified CM tries to initialize the device.</td>
</tr>
<tr>
<td>8</td>
<td>DeviceInitiatedReset—The device has initiated a reset, possibly due to a power cycle or internal error. No action required; the device will reregister automatically.</td>
</tr>
<tr>
<td>9</td>
<td>CallManagerReset—A device reset was initiated from Cisco Unified CM Administration, either due to an explicit command from an administrator, or due to internal errors encountered. No action necessary, the device will reregister automatically.</td>
</tr>
<tr>
<td>10</td>
<td>DeviceUnregistered—The device has explicitly unregistered. Possible causes include a change in the IP address or port of the device. No action is necessary, the device will reregister automatically.</td>
</tr>
<tr>
<td>11</td>
<td>MalformedRegisterMsg—(SIP only) A SIP REGISTER message could not be processed because of an illegal format. Possible causes include a missing Call-ID header, a missing AoR in the To header, and an expires value too small. Verify the REGISTER message does not suffer from any of these ills.</td>
</tr>
<tr>
<td>12</td>
<td>SCCPDeviceThrottling—(SCCP only) The indicated SCCP device exceeded the maximum number of events allowed per-SCCP device. Events can be phone calls, KeepAlive messages, or excessive SCCP or non-SCCP messages. The maximum number of allowed events is controlled by the Cisco CallManager service parameter, Max Events Allowed. When an individual device exceeds the number configured in that service parameter, Unified CM closes the TCP connection to the device; automatic reregistration generally follows. This action is an attempt to stop malicious attacks on Unified CM or to ward of excessive CPU usage. No action necessary, the device will reregister automatically.</td>
</tr>
<tr>
<td>13</td>
<td>KeepAliveTimeout—A KeepAlive message was not received. Possible causes included device power outage, network power outage, network configuration error, network delay, packet drops and packet corruption. It is also possible to get this error if the Unified CM node is experiencing high CPU usage. Verify the device is powered up and operating, verify network connectivity between the device and Unified CM, and verify the CPU utilization is in the safe range (this can be monitored using RTMT via CPU Pegging Alert). No action necessary, the device will reregister automatically.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>14</td>
<td>ConfigurationMismatch—(SIP only) The configuration on the device does not match the configuration in Unified CM. This can be caused by database replication errors or other internal Unified CM communication errors. First go to the Cisco Unified Reporting web page, generate a Unified CM Database Status report, and verify &quot;all servers have a good replication status&quot;. If this device continues to unregister with this reason code, go to the Cisco Unified CMAadmin Device web page for the device and click Save. This allows a change notify to be generated to the Unified CM and TFTP services and rebuild a new config file. If the problem still persists, restart the TFTP service and Unified CM service.</td>
</tr>
<tr>
<td>15</td>
<td>CallManagerRestart—A device restart was initiated from Cisco Unified CM, either due to an explicit command from an administrator, or due to a configuration change such as adding, deleting or changing a DN associated with the device. No action necessary, the device will reregister automatically.</td>
</tr>
<tr>
<td>16</td>
<td>DuplicateRegistration—Cisco Unified CM detected that the device attempted to register to two nodes at the same time. Cisco Unified CM initiated a restart to the phone to force it to rehome to a single node. No action necessary, the device will reregister automatically.</td>
</tr>
<tr>
<td>17</td>
<td>CallManagerApplyConfig—An ApplyConfig command was invoked from Unified CM Administration resulting in an unregistration. No action necessary, the device will reregister automatically.</td>
</tr>
<tr>
<td>18</td>
<td>DeviceNoResponse—The device did not respond to a reset or restart notification, so it is being forcefully reset. If the device does not reregister within 5 minutes, confirm it is powered-up and confirm network connectivity between the device and Cisco Unified CM.</td>
</tr>
<tr>
<td>19</td>
<td>EMLoginLogout —The device has been unregistered due to an Extension Mobility login or logout.</td>
</tr>
<tr>
<td>20</td>
<td>EMCCLoginLogout—The device has been unregistered due to an Extension Mobility Cross Cluster login or logout.</td>
</tr>
<tr>
<td>21</td>
<td>PowerSavePlus—The device powered off as a result of the Power Save Plus feature that is enabled for this device. When the device powers off, it remains unregistered from Unified CM until the Phone On Time defined in the Product Specific Configuration for this device.</td>
</tr>
<tr>
<td>22</td>
<td>CallManagerForcedRestart—(SIP Only) The device did not respond to an Apply Config request and as a result, Unified CM sent a restart request to the device. The device may be offline due to a power outage or network problem. Confirm that the device is powered-up and that network connectivity exists between the device and Unified CM.</td>
</tr>
<tr>
<td>23</td>
<td>SourceIPAddrChanged—(SIP Only) The device has been unregistered because the IP address in the Contact header of the REGISTER message has changed. The device will be automatically reregistered. No action is necessary.</td>
</tr>
<tr>
<td>24</td>
<td>SourcePortChanged—(SIP Only) The device has been unregistered because the port number in the Contact header of the REGISTER message has changed. The device will be automatically re-registered. No action is necessary.</td>
</tr>
<tr>
<td>25</td>
<td>RegistrationSequenceError—(SCCP only) A device requested configuration information from the Unified CM at an unexpected time. The Unified CM no longer had the requested information in memory.</td>
</tr>
<tr>
<td>26</td>
<td>InvalidCapabilities—(SCCP only) Unified CM detected an error in the updated media capabilities reported by the device. The device reported the capabilities in one of the StationUpdateCapabilities message variants.</td>
</tr>
</tbody>
</table>
Error-Level Alarms

Enum Definitions - IPAddressAttributes

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown—The device has not indicated what this IPv4 address is used for</td>
</tr>
<tr>
<td>1</td>
<td>Administrative only—The device has indicated that this IPv4 address is used for administrative communication (web interface) only</td>
</tr>
<tr>
<td>2</td>
<td>Signal only—The device has indicated that this IPv4 address is used for control signaling only</td>
</tr>
<tr>
<td>3</td>
<td>Administrative and signal—The device has indicated that this IPv4 address is used for administrative communication (web interface) and control signaling</td>
</tr>
</tbody>
</table>

Enum Definitions - IPV6AddressAttributes

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown—The device has not indicated what this IPv6 address is used for</td>
</tr>
<tr>
<td>1</td>
<td>Administrative only—The device has indicated that this IPv6 address is used for administrative communication (web interface) only</td>
</tr>
<tr>
<td>2</td>
<td>Signal only—The device has indicated that this IPv6 address is used for control signaling only</td>
</tr>
<tr>
<td>3</td>
<td>Administrative and signal—The device has indicated that this IPv6 address is used for administrative communication (web interface) and control signaling</td>
</tr>
</tbody>
</table>

Recommended Action

Actions to take vary depending on the reason specified for the endpoint unregistration. If the reason is ConfigurationMismatch, go to the Device Configuration page in Cisco Unified CM Administration, make a change to the Description field for this device, click Save, then reset the device. In the case of a network connectivity or loss of KeepAlives problem, use network diagnostic tools and the Cisco Unified CM Reporting tool to fix any reported network or Unified CM system errors. In the case of an endpoint rehoming to the primary Unified CM node, watch for a successful registration of the device on the primary node. In the case of a duplicate registration request, it may be a non-malicious occurrence due to timing of an endpoint registering and unregistering; if duplicate registration requests continue or if the same endpoint has different IP addresses, confirm the IP address on the physical device itself by checking the settings on the device (settings button). If unregistration of this device was expected, no action is required. Also, refer to the reason code descriptions for recommended actions.
ErrorChangeNotifyClientTimeout

A change notification client was responding slowly and has been removed. A change notification recipient has not responded to change notification in several minutes and was thus removed. This may delay call processing features, such as call forwarding and so on.

Facility/Sub-Facility
CCM_DB_LAYER-DB

Cisco Unified Serviceability Alarm Definition Catalog
System/DB

Severity
Error (3)

Routing List
SDI
Event Log
Sys Log

Recommended Action
Rebooting the box will clear this situation. Alternatively, dbnotify trace could be analyzed to find the client that was removed and that service could be restarted in Cisco Unified Serviceability.

ErrorParsingDirectiveFromPDP

Cisco Unified Communications Manager (Unified CM) failed to parse the call routing directive or the diversion destination in the call routing response from the policy decision point (PDP).

A routing response was received but Cisco Unified Communications Manager (Unified CM) failed to parse the mandatory elements in the response. This means that a call routing directive or the call diversion destination could not be parsed correctly, or that the call routing directive was not recognized. The error may due to a syntax error or because the call routing directive is missing or the call diversion destination is missing in the call routing response.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
ERROR
Chapter 6      Cisco Unified Serviceability Alarms and CiscoLog Messages

Error-Level Alarms

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Policy Decision Point(String)
Called Party Number(String)
Calling Party Number(String)
Calling User Id(String)
Response XML Data(String)

Recommended Action
Check the external call control documentation, including any applicable API documentation, to determine whether the call routing directive that was included as part of the policy obligations in the call routing response are correctly entered according to the information defined in the external call control documentation.

ErrorReadingInstalledRPMS

Could not read installed RPMs to populate component version table. The function that reads the RPM version information and populates database failed.

Facility/Sub-Facility
CCM_DB_LAYER-DB

Cisco Unified Serviceability Alarm Definition Catalog
System/DB

Severity
Error (3)

Recommended Action
Report this error to the administrator.

FailureResponseFromPDP

The policy decision point (PDP) returned a 4xx (client) or 5xx (server) status code in the HTTP response. Cisco Unified Communications Manager (Unified CM) received a 4xx or 5xx response from the policy decision point (PDP). A 4xx response indicates errors in the call routing request from Unified CM, for example: a 400 response indicates the call routing request could not be understood by the PDP; a 404 indicates that the PDP did not find a matching request URI. A 5xx error indicates a PDP server error, for example: a 500 response indicates a PDP internal error; A 501 response indicates that the PDP does not support the functionality to generate a call routing response; a 503 indicates that the PDP is busy and
temporarily cannot generate a response; a 505 indicates that the HTTP version number included in the call routing request from Unified CM is not supported. Other such errors may be responsible; please refer to generally available guidelines on HTTP or check the RFC 2616 for detailed explanations about HTTP Status Code definitions.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
ERROR

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Policy Decision Point(String)
The status code and reason phrase for the failure(String)

Recommended Action
If a 4xx response caused the alarm, verify that the PDP has been accurately configured for the functionality and call routing that you expect it to perform. If a 500 response causes the alarm, check whether the PDP service is active and check the PDP server's logfiles for any errors. If a 503 causes the alarm, the PDP may be overloaded by requests. Take appropriate action to reduce the load on the PDP by following some or all of these recommendations: 1) consider adding more PDPs and provisioning Unified CM with additional call intercept profiles and call intercept trigger points in the various configuration pages under the Call Routing menu in Cisco Unified CM Administration; 2) provision a pair of policy servers per call-intercept profile to enable load balancing; or 3) verify that the PDP server in your deployment meets or exceed the hardware requirements specified in the documentation for Cisco Enterprise Policy Manager (CEPM) or the third-party PDP solution you have deployed. If a 505 response causes the alarm, check to be sure that the PDP supports HTTP version 1.1.

FailedToReadConfig

Service Manager failed to read configuration file.

Facility/Sub-Facility
CCM_SERVICEMANAGER GENERIC

Cisco Unified Serviceability Alarm Definition Catalog
System/Service Manager

Severity
Error (3)
FirewallMappingFailure

Firewall unreachable.

This alarm indicates that Unified CM was unable to contact the firewall in order to make a IME call. As a consequence, outbound calls are being sent over the PSTN, and inbound calls may be routed over the PSTN by your partner enterprises.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
ERROR

Recommended Action
Check to see that your firewall is up. Make sure the mapping service is enabled. Check that the IP address and port on the firewall for that mapping service match the configuration in Unified CM Administration. Check general IP connectivity between Unified CM and the firewall.

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
IP address(String)
Port number(UInt)

ICTCallThrottlingStart

Cisco CallManager stops handling calls for the indicated H.323 device due to heavy traffic or a route loop over the H.323 trunk.

Cisco Unified Communications Manager has detected a route loop over the H.323 trunk indicated in this alarm. As a result, Unified CM has temporarily stopped accepting calls for the indicated H.323 trunk. It's also possible that a high volume of calls are occurring over the intercluster trunk, which has triggered throttling.
Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Error (3)

Parameters

Enum Definitions for DeviceType
125—TRUNK

Recommended Action
In Real-Time Monitoring Tool, check the CallsActive and CallsInProgress counters for unusual activity on the indicated H.323 trunk. If the CallsActive count is significantly higher than usual, a traffic load issue may be occurring where the demand to send calls over the trunk is greater than the trunk's capacity. Monitor the situation and collect existing trace files. If the ICTCallThrottlingEnd alarm is not issued in a reasonable amount of time as deemed by your organization, contact TAC and supply the trace information you have collected. For a routing loop condition, the CallsInProgress counter will be significantly higher than usual. By examining trace files and CDR data for calls that occurred over the indicated trunk, you may be able to detect a translation pattern, route list or other routing mechanism that is part of the loop. Update the routing mechanism that resulted in the loop (generally the same number is configured on both near end and far end devices) and then reset the affected route list in an attempt to clear the route loop and if that fails, reset the affected trunk.

IDSEngineCritical

This alarm does not compromise data or prevent the use of the system but need to be monitored by the Administrator.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed severity level to Error from Critical.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_DB_LAYER-DB

Cisco Unified Serviceability Alarm Definition Catalog
System/DB

Severity
Error (3)
**Parameters**
Event Class ID [String] Event class message [String] Event Specific Message [String]

**Recommended Action**
This alarm needs monitoring by the db admin.

### IDSEngineFailure

Combined alarm for emergency and error situations. Something unexpected occurred that might compromise data or access to data or cause IDS to fail. This alarm indicates combined alarm for emergency and error situations. Something unexpected occurred that might compromise data or access to data or cause IDS to fail.

**Facility/Sub-Facility**
CCM_DB_LAYER-DB

**Cisco Unified Serviceability Alarm Definition Catalog**
System/DB

**Severity**
Error (3)

**Parameters**
Event Class ID [String] Event class message [String] Event Specific Message [String]

**Recommended Action**
Requires Database Admin. intervention

### IDSReplicationFailure

Combined alarm for emergency and error situations. IDS Replication has failed.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Route Listing element Data Collector changed to Alert Manager and existing parameters added.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
DB

**Cisco Unified Serviceability Alarm Definition Catalog**
System/DB
Error-Level Alarms

Severity
Error (3)

Routing List
SDI
Event Log
Sys Log
Alert Manager

Parameters
Event Class ID [String]
Event class message [String]
Event Specific Message [String]

Recommended Action
Requires Database Admin. intervention.

InsufficientFallbackIdentifiers

Cannot allocate fallback identifier.
This alarm is generated when Unified CM is processing a IME call, and is attempting to allocate a PSTN fallback DID and a DTMF digit sequence to associate with this call. However, there are too many IME calls currently in progress which are utilizing this same fallback DID, and as a result, there are no more DTMF digit sequences which could be allocated to this call. As such, this call will proceed, however mid-call fallback will not be possible for this call.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
ERROR

Routing List
SDL
SDI
Sys Log
Event Log
Alert Manager

Parameter(s)
Fallback profile name(String)
Fallback E.164 number(UInt)
Current number of DTMF digits(UInt)
E.164 called party number(String)
Recommended Action

Your first course of action should be to identify the fallback profile associated with this call. Its name will be present in the alarm. Check that profile from the admin interface, and examine the current setting for "Fallback Number of Correlation DTMF Digits". Increase that value by one, and check if that eliminates these alarms. In general, this parameter should be large enough such that the number of simultaneous IME calls made to enrolled numbers associated with that profile is always substantially less than 10 raised to the power of this number. "Substantially" should be at least a factor of ten. For example, if you always have less than 10,000 simultaneous IME calls for the patterns associated with this fallback profile, setting this value to 5 (10 to the power of 5 is 100,000) will give you plenty of headroom and you will not see this alarm.

However, increasing this value also results in a small increase in the amount of time it takes to perform the fallback. As such, it should not be set arbitrarily large; it should be set just large enough to keep clear of this alarm. Another alternative to increasing this parameter is to add another fallback profile with a different fallback DID, and associate that fallback profile with a smaller number of enrolled DID patterns. This will allow you to get by with a smaller number of digits.

InvalidIPNetPattern

An invalid IP address is configured in one or more SIP route patterns in Cisco Unified CM Administration.

Facility/Sub-Facility

CCM_CALLMANAGER/CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity

Error (3)

Parameters

Description(String)
IPAddress(String)
DeviceName(String)

Recommended Action

In Cisco Unified CM Administration, verify that the route pattern associated with the device that is identified in this alarm has an accurate and working IP address. You can learn more how to ensure that the IP address is valid by reviewing RFC 2373.

InvalidPortHandle

The handle for the opened serial port is invalid.

CMI cannot read/write to the serial port because the serial port returned an invalid handle value to CMI. The serial port may have returned an invalid handle because the system did not properly detect the USB cable.
Error-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
CMIAlarmCatalog/CMI

Severity
ERROR

Routing List
Event Log
SDI

Parameter(s)
Error Information(String)

Recommended Action
Make sure that the cable connecting the USB0 port and voice messaging system is firmly connected.

IPMAApplicationError

IPMA Facility/Sub-Facility error.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Servlet Name [String] Reason [String]

Recommended Action
See application logs for details

IPMAOverloaded

IPMA Facility/Sub-Facility overloaded.
Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Servlet Name [String] Reason [String]

Recommended Action
See application logs for details

IPMAFilteringDown

IPMA application filtering is down.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Servlet Name [String] Reason [String]

Recommended Action
Restart Cisco IP Manager Assistant Service.

IPv6InterfaceNotInstalled

IPv6 network interface is not installed. IPv6 option is enabled for TFTP service but the IPv6 network interface/address has not been configured on the system. Until the IPv6 network is functioning, devices that have been configured with IPv6-only will not be able to register. Devices that have been configured to use either IPv6 or IPv4 will register using IPv4. When the IPv6 network is online, IPv6-capable devices that have registered as IPv4 will remain IPv4 until they are reset, at which time they will use IPv6 if available.
## Error-Level Alarms

### Facility/Sub-Facility

**CCM_TFTP-TFTP**

### Cisco Unified Serviceability Alarm Definition Catalog

**System/TFTP**

### Severity

**Error (3)**

### Parameters

None

### Recommended Action

Install IPv6 network interface and then restart TFTP service.

### kANNDeviceRecordNotFound

ANN device record not found. A device record for the announcer device was not found in the database. The ANN device is normally automatically added when the server is added to the database.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Added to CallManager Catalog.</td>
</tr>
</tbody>
</table>

### Facility/Sub-Facility

**CCM_MEDIA_STREAMING_APP-IPVMS**

### Cisco Unified Serviceability Alarm Definition Catalog

**CallManager/IpVms**

### Severity

**Error**

### Recommended Action

To add the ANN device to database you will need to remove/delete the server and read the server. **WARNING:** This may result in having to manually reconfigure many different settings such as Media Resource Groups, CallManager Groups and many others.
kCFBDeviceRecordNotFound

CFB device record not found. A device record for the conference bridge device was not found in the database. The CFB device is normally automatically added when the server is added to the database.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). The severity changed from Informational to Error.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/IpVms

Severity

Error

Recommended Action

To add the CFB device to database you will need to remove/delete the server and read the server. WARNING: This may result in having to manually reconfigure many different settings such as Media Resource Groups, CallManager Groups and many others.

kCreateAudioSourcesFailed

Creating audio source class failed. Unable to create audio source subcomponent to provide audio for streaming. This may be due to lack of memory.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1)</td>
</tr>
</tbody>
</table>

Following parameters added:

- OS Error Code(Int)
- OS Error Description(String)
Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Error (3)

Parameters
OS Error Code(Int)
OS Error Description(String)

Recommended Action
Restart the Cisco IP Voice Media Streaming App service or restart the server.

kCreateControlFailed

Stream Control create failure. Create stream control subcomponent. The error may be due to lack of memory.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1) Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>– OS Error Code(Int)</td>
</tr>
<tr>
<td></td>
<td>– OS Error Description(String)</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Error (3)

Parameters
Codec Type [String]
OS Error Code [Int]
OS Error Description [String]

Recommended Action
Reset the MOH device. If continues to fail restart the Cisco IP Voice Media Streaming App service or restart the server.

kDbConnectionFailed

Database connection failed.

Facility/Sub-Facility
CCM_DB_LAYER-DB

Cisco Unified Serviceability Alarm Definition Catalog
System/DB

Severity
Error (3)

Parameters
Additional Information [String]

Recommended Action
Enable trace for the database layer monitor to get specific error information.

kIPVMSDeviceDriverNotFound

Cisco IP voice media streaming driver not found. The Cisco IP voice media streaming driver was not found or is not installed. The Cisco IP Voice Media Streaming App service cannot run until this error is resolved. All software media devices (ANN, CFB, MOH, MTP) for this server will not be available.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms
## Error-Level Alarms

### Severity

**Error (3)**

### Recommended Action

Check the system log for an error when the system attempted to load IpVms driver at the last server startup. A server restart is required to cause the driver to be loaded.

### kIpVmsMgrNoLocalHostName

Unable to retrieve the local host server name. The Cisco IP Voice Media Streaming App service will terminate. No software media devices (ANN, CFB, MOH, MTP) will be available while the service is stopped.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

### Severity

**Error (3)**

### Recommended Action

Check the configuration settings for the server name, DHCP, or DNS. Monitor the status of Cisco IP Voice Media Streaming App service. The service will not operate without a valid server name.

### kIpVmsMgrNoLocalNetworkIPAddr

Unable to retrieve the network IP address for host server. Unable to obtain the network IP (dotted) address. The Cisco IP Voice Media Streaming App service will terminate. The software media devices (ANN, CFB, MOH, MTP) will be unavailable while this service is stopped.
Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Error-Level Alarms

### Facility/Sub-Facility

CCM_MEDIA_STREAMING_APP-IPVMS

### Cisco Unified Serviceability Alarm Definition Catalog

CallManager/IpVms

### Severity

Error (3)

### Recommended Action

Monitor the status of the Cisco IP Voice Media Streaming App service. It should be automatically restarted. If the error occurs again, check the server IP configuration (DHCP, IP address).

#### kIPVMSMgrWrongDriverVersion

Wrong version of device driver. An incompatible device driver was found. The Cisco IP Voice Media Streaming App service will terminate. The software media devices (ANN, CFB, MOH, MTP) will be unavailable while the service is stopped.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
</tbody>
</table>

### Facility/Sub-Facility

CCM_MEDIA_STREAMING_APP-IPVMS

### Cisco Unified Serviceability Alarm Definition Catalog

CallManager/IpVms
**Severity**
Error (3)

**Recommended Action**
Restart the server to ensure the most recent driver is started. If the error continues, then reinstall Cisco Unified Communications Manager to get the proper driver version installed.

---

**kMOHTFTPGoRequestFailed**

Transfer of MOH source file to working path failed. An error was encountered when trying to copy or update a Music-on-Hold audio source file.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Following parameters added: Error Description [String] Source Path [String] Destination Path [String] OS Error Code [Int] OS Error Description [String]</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/IpVms

**Severity**
Error (3)

**Parameters**

**Recommended Action**
Use the Platform CLI to verify the source path and file exist. If the file does not exist then use Cisco Unified CM Admin to reupload the missing audio source to this specific server. Reinstall the Cisco Unified Communications Manager to have all required paths created.
**kPWavMgrThreadxFailed**

WAV playing manager thread creation failed. The process component used for playing WAV files failed to start, possibly due to low system resources.

**Facility/Sub-Facility**
CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/IpVms

**Severity**
Error (3)

**Parameters**
OS Error Description(String)

**Recommended Action**
Restart the Cisco IP Voice Media Streaming App service or restart server.

**kReadCfgUserLocaleEnterpriseSvcParm**

Error reading Enterprise User Locale configuration. A database exception was encountered when reading the default Enterprise User Locale setting. Default of US English will be used.

**Facility/Sub-Facility**
CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/IpVms

**Severity**
Error (3)

**Recommended Action**
Verify that the Enterprise parameter setting for User Locale is configured using the CCM Admin web page. Restart the Cisco IP Voice Media Streaming App service.
### kRequestedANNStreamsFailed

The requested resources for the configured number of annunciator calls (Call Count service parameter) was not available. If the value gets shown as “Allocated,” it is non-zero.

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

**Severity**

Error (3)

**Recommended Action**

Verify that the ANN Call Count service parameter is correct. A server restart may be needed to recover resources.

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Added descriptive text and Recommended Actions. Following parameters are removed: Requested streams [ULong] Allocated streams [ULong]</td>
</tr>
</tbody>
</table>

### LostConnectionToSAFForwarder

Connection to the SAF Forwarder has been lost.

A TCP connection failure caused the connection between the SAF Forwarder and Unified CM to be lost. When the TCP connection is restored, Unified CM attempts to connect to the SAF Forwarder automatically. If IP connectivity is unreachable for longer than the duration of the Cisco CallManager service parameter CCD Learned Pattern IP Reachable Duration, calls to learned patterns will be routed through the PSTN instead. Calls through the PSTN to learned patterns will be maintained for a certain period of time before the PSTN failover times out.

**Cisco Unified Serviceability Alarm Catalog**

CallManager/CallManager

**Severity**

Error

**Routing List**

SDL
SDI
Sys Log
Error-Level Alarms

Event Log
Data Collector

Parameters
IP Address(String)
SafClientHandle(UInt)

Recommended Action
Investigate possible causes of a TCP connection failure, such as power failure, loose cables, incorrect switch configuration, and so on, and correct any issues that you find. After the connection is restored, CCD will try to register/sync with the SAF Forwarder automatically.

MultipleSIPTrunksToSamePeerAndLocalPort

Multiple trunks have been configured to the same destination and local port, which resulted in a conflict. Only one trunk is allowed for one destination/local port combination. The latest trunk invalidated earlier.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Error

Parameters
Peer IP Address. [String] Local IP Port [UInt] Old Device name. [String] Old Device Instance. [String]
New Device name. [String] New Device Instance. [String]

Recommended Action
Check the SIP Trunk Configuration in Cisco Unified CallManager Administration and verify that only one SIP trunk has been configured to the same destination address and local port.

NodeNotTrusted

Untrusted Node was contacted. Application could not establish secure connection (SSL handshake failure) with another application. It could be due to certificate for tomcat service where the application is hosted is not trusted (not present in the keystore).

Cisco Unified Serviceability Alarm Definition Catalog
System/EMAlarmCatalog

Severity
ERROR
Routing List
Sys Log
Event Log
Alert Manager

Parameter(s)
Date/Time(String)
Hostname/Ip Address(String)

Recommended Action
1. Ensure that "tomcat-trust" keystore on each CCM node contains the tomcat certificates for every other node within a cluster (Logon to OS Administration Page -> Security -> Certificate Management -> Check the certificates in tomcat-trust). 2. If EMCC is enabled, then ensure that a bundle of all tomcat certificates (PKCS12) has been imported into the local tomcat-trust keystore (Logon to OS Administration Page -> Security -> Certificate Management -> Look for certificates in tomcat-trust).

NumDevRegExceeded

The allowed number of registered devices was exceeded.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Error (3)

Parameters
Maximum Devices [Int]

Recommended Action
If you did not expect to exceed the number of devices and you have auto-registration enabled, go to Device > Phones in Cisco Unified CM Administration and search for phones starting with "auto". If you see any unexpected devices which may not belong in the system (such as intruder devices) locate that device using the IP address and remove it from the system. Or, if your licenses and system resources allow, increase the value in the Cisco CallManager service parameter, Maximum Number of Registered Devices.

PublishFailedOverQuota

Each IME server has a fixed quota on the total number of DIDs it can write into the IME distributed cache. When this alarm is generated, it means that, even though you should be under quota, due to an extremely unlikely statistical anomaly, the IME distributed cache rejected your publication, believing you were over quota. You should only see this alarm if you are near, but below, your quota. This error
is likely to be persistent, so that the corresponding E.164 number from the alarm will not be published into the IME distributed cache. This means that you will not receive VoIP calls towards that number - they will remain over the PSTN.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>New Alarm for this release.</td>
</tr>
</tbody>
</table>

### Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

### Severity

ERROR_ALARM

### Recommended Action

The alarm will include the name of the IME server, and the current and target quota values. The first thing to check is to make sure that you have correctly provisioned the right set of DID prefixes on all of the Unified CM clusters sharing that same IME server on the same IME distributed cache. If that is correct, it means you have exceeded the capacity of your IME server, and you require another. Once you have another, you can now split your DID prefixes across two different IME client instances, each on a different IME server. That will alleviate the quota problem.

### Routing List

- SDL
- SDI
- Sys Log
- Event Log

### Parameter(s)

- The DID for which the Publish was attempted(String)
- Server name(String)
- Current quota(UInt)
- Maximum target quota(UInt)

### ReadConfigurationUnknownException

An exception is caught while retrieving enterprise parameters value from database at TFTP service startup. This is usually caused by a failure to access the Cisco Unified Communications Manager database.
Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Error-Level Alarms

Facility/Sub-Facility
CCM_TFTP-TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP

Severity
Error (3)

Recommended Action
In Cisco Unified Serviceability, enable Detailed level traces in the Trace Configuration window for TFTP and Cisco Database Layer Monitor services. Also, use RTMT to look for errors that may have occurred around the time of the alarm.

### ReadingFileFailure

CMI failed to read SMDI messages from the serial port.

CMI opened the serial port, however it failed to successfully read data from the serial port because the serial port returned an invalid handle value to CMI. The serial port may have returned an invalid handle because the system did not properly detect the USB cable.

#### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kReadingFileFailure.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CMIAlarmCatalog/CMI

Severity
ERROR

Routing List
Event Log
SDI
Parameter(s)
Error Information(String)

Recommended Action
Make sure that the cable connecting the USB0 port and voice messaging system is firmly connected.

**RsvpNoMoreResourcesAvailable**

RSVP Agent resource allocation failed.

The alarm occurs when allocation of an RSVP Agent fails for all the registered RSVP Agents (RSVP Agents are basically MTPs or transcoder devices which provide RSVP functionalities) belonging to the Media Resource Group List and Default List. Each RSVP Agent may fail for different reasons. Following are some of the reasons that could cause an RSVP Agent allocation to fail: available MTP/transcoders do not support RSVP functionality; a capability mismatch between the device endpoint and MTP/transcoder, codec mismatch between the endpoint and the MTP/transcoder; a lack of available bandwidth between the endpoint and the MTP/transcoder; or because the MTP/transcoder resources are already in use.

A capability mismatch may be due to the MTP/transcoder not supporting one or more of the required capabilities for the call such as Transfer Relay Point (which is needed for QoS or firewall traversal), RFC 2833 DTMF (which is necessary when one side of the call does not support RFC 2833 format for transmitting DTMF digits and the other side must receive the DTMF digits in RFC2833 format, resulting in conversion of the DTMF digits), RFC 2833 DTMF passthrough (in this case, the MTP or transcoder does not need to convert the DTMF digits from one format to another format but it needs to receive DTMF digits from one endpoint and transmit them to the other endpoint without performing any modifications), passthrough (where no codec conversion will occur, meaning the media device will receive media streams in any codec format and transmit them to the other side without performing any codec conversion), IPv4 to IPv6 conversion (when one side of the call supports only IPv4 and the other side of the call supports only IPv6 and so MTP needs to be inserted to perform the necessary conversion between IPv4 and IPv6 packets), or multimedia capability (if a call involving video and/or data in addition to audio requires insertion of an MTP or transcoder then the MTP/transcoder which supports multimedia will be inserted).

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Media Resource List Name(String) parameter is added.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CallManager

**Severity**

Error (3)
**Parameter(s)**

Media Resource List Name(String)

**Recommended Action**

RSVP Agents are basically Cisco IOS MTPs or transcoder devices which provide RSVP functionalities. Check the user manual of the configured MTPs and transcoders to see whether they support RSVP functionality. If none of them support RSVP functionality either they need to be upgraded (if upgraded version support RSVP functionality) or additional MTP or transcoders need to be installed which support RSVP functionality. If the RSVP Agent (MTP or transcorder) allocation is failing due to a capability mismatch, it's possible that the media device does not support the requested capability (such as IPv4 to IPv6 conversion, passthrough) or the capability might not be configured in the device. Please check the user guide and documentation of the media device to make sure that device supports all the necessary capabilities.

Also, caution should be taken care if all the MTP or transcoders are configured with all the supported capabilities. There are certain capabilities (such as RFC 2833 DTMF or RFC 2833 DTMF passthrough or passthrough) which could be supported by most of the MTPs or transcoders and there may be certain capabilities (such as IPv4 to IPv6 conversion and vice versa or RSVP Agent functionality or Transfer Relay Point or multimedia capability) which can be supported by only by a single MTP or transcoder depending on the devices that you have.

For example, you may have end devices belonging to different locations and may need to reserve the bandwidth only between two locations; calls between other locations may not need to reserve the bandwidth. Now, suppose all the MTPs or transcoders are configured with all the supported capabilities and only one MTP/transcoder supports RSVP functionality; if this MTP/transcoder is configured with all the supported capabilities (which all the other MTPs or transcoders in the same MRGL or default MRGL also support) it may happen that this MTP can get allocated for Transfer Relay Point or RFC2833 DTMF or RFC 2833 DTMF passthrough or passthrough instead. As a result, when a need arises to reserve the bandwidth (which other MTPs or transcoders in the same MRGL or default MRGL do not support), all the resources of this MTP/transcoder may be in use and the RSVP Agent allocation may fail.

To avoid this situation, set the priority of the media resources appropriately. This can be done only in the Media Resource Group List and not in the Default List of the media resources. In any Media Resource Group List all the Media Resource Groups have different priorities and during allocation the first Media Resource Group is checked for availability of the requested type of the media devices. The first Media Resource Group in the Media Resource Group List will have the highest priority, then the second one and so on. To check all the Media Resource Groups and their priority go the Media Resources and Media Resource Group List of Cisco Unified CM Administration page and click the appropriate Media Resource Group List and check the Selected Media Resource Groups; the priority decreases from top to bottom. Position the MTP or transcoder that you want to be selected for the basic functionalities in the higher priority Media Resource Groups whereas the ones with more rare functionality can be positioned in the Media Resource Groups with lower priority. RSVP Agent allocation may fail due to codec mismatch between the end point and the RSVP Agent or MTP/transcoder.

A solution may be to configure the MTP/transcoder with all the supported codecs (as specified in the user guide of the MTP/transcoder), but be aware that doing so might result in too much bandwidth being allocated for calls. You'll need to weigh different factors such as the total amount of available bandwidth, the average number of calls, approximate bandwidth use per call (not involving MTP/transcoder), and so on, and accordingly calculate the maximum bandwidth that can be allocated per call involving an MTP/transcoder and take that into consideration when configuring the supported codecs in the MTPs and transcoders. A good idea is to configure the media devices with all the supported codecs and set the region bandwidths to restrict too much bandwidth usage (refer to the Unified CM documentation for details on region and location settings).
Also, there may be codec mismatch between the endpoint and the MTP/transcoders after considering the region bandwidth between the MTP/transcoder and the endpoint. Increasing the region bandwidth may be a solution to the problem, but that decision should be made after careful consideration of the amount of bandwidth you're willing to allocate per call between the set of regions.

Another possible cause that an MTP/transcoder did not get allocated is because there was not enough available bandwidth for the call. This can happen if the MTP/transcoder and endpoint belong to different locations and the bandwidth that is set between the locations is already in use by other calls. Examine the bandwidth requirements in your deployment to determine whether bandwidth between the locations can be increased. However, note that increasing the bandwidth between these two locations means that you may need to reduce the bandwidth between other locations.

Refer to the System Guide, SRNDs, and related Unified CM documentation for more details. Be aware that reducing the bandwidth or removing the higher bandwidth codecs from configuration may result in poor voice quality during call. Consider increasing the total amount of network bandwidth. Finally, if RSVP Agent allocation fails due to MTP/transcoder not supporting RSVP functionality or capability mismatch or all the resources being in use, consider installing additional MTP or transcoder devices which support RSVP functionality.

**RTMT_ALERT**

A Real-Time Monitoring Tool (RTMT) process in the AMC service uses the alarm mechanism to facilitate delivery of RTMT alerts in the RTMT AlertCentral or through email.

**Cisco Unified Serviceability Alarm Definition Catalog**

**System/RTMT**

**Severity**

ERROR

**Routing List**

Event Log
Sys Log

**Parameter(s)**

Name(String)
Detail(String)

**Recommended Action**

Check AlertCentral in RTMT or any alerts that you have received through email to determine what issue has occurred and learn the recommended actions to resolve it. In AlertCentral, right-click the alert to open the alert information.

**RTMT-ERROR-ALERT**

This alert is generated by RTMT AlertMgr. See Alert Detail for explanation.

**Facility/Sub-Facility**

CCM_RTMT-RTMT
Error-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
System/RTMT

Severity
Error (3)

Parameters
Name [String] Detail [String]

Recommended Action
See Alert Detail for more information.

SAFForwarderError

SAF Forwarder error response sent to Unified CM.

Cisco Unified Serviceability Alarm Catalog
CallManager/CallManager

Severity
Error

Routing List
SDL
SDI
Sys Log
Event Log
Data Collector

Parameters
IP Address(String)
SafClientHandle(UInt)
Application User Name(String)
Reason Code and Description(Enum)
SAF Protocol Version Number(String)
Service ID(UInt)
Sub Service ID(UInt)

Recommended Action
Refer to the reason code and description (help text) for specific information and actions (where applicable) for this alarm.
## Enum Definitions - Reason Code

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>400</td>
<td>SAF_BAD_REQUEST - SAF Forwarder was unable to accept the request due to incorrect syntax (malformed), missing required attributes, and other similar reasons. Investigate the configuration between the SAF Forwarder and Unified CM to be certain that all settings are correct for your deployment. In particular, check the Client Label configured on the router to make certain that it matches the Client Label configured in Cisco Unified CM Administration on the SAF Forwarder Configuration window (SAF &gt; SAF Forwarder).</td>
</tr>
<tr>
<td>431</td>
<td>SAF_INTEGRITY_CHECK_FAILURE - A message failed to pass SAF Forwarder security validation. This can occur because of misconfiguration, a potential attack, or more commonly by incorrect provisioning of the password on the Forwarder and SAF client. Reprovision the password and keep a watch on further SAF INTEGRITY CHECK FAILURE alarms. If you receive a persistent number of SAF INTEGRITY CHECK FAILURE alarms, close the interface between SAF Forwarder and Unified CM and investigate the source of the IP packets.</td>
</tr>
<tr>
<td>435</td>
<td><strong>INFO LEVEL</strong> SAF_MISSING_NONCE - A nonce (a random parameter generated when the message is sent) is missing from the message. The system will resend with a new nonce automatically. No action is required.</td>
</tr>
<tr>
<td>436</td>
<td>SAF_UNKNOWN_USERNAME - Unified CM sent the SAF Forwarder an Application User name that is not configured on the router or that does not match the router's configuration. Check the Application User Name on the router and in the Application User Configuration window in Cisco Unified CM Administration to be sure they match.</td>
</tr>
<tr>
<td>438</td>
<td><strong>INFO LEVEL</strong> SAF_STALE_NONCE - A nonce (a random parameter generated when the message is sent) has aged out (gone stale). The system will resend with a new nonce automatically. No action is required.</td>
</tr>
<tr>
<td>471</td>
<td><strong>INFO LEVEL</strong> SAF_BAD_CLIENT_HANDLE - SAF_BAD_CLIENT_HANDLE - Unified CM sent the SAF Forwarder a Register message (for KeepAlive purposes) or unregister message with the mandatory CLIENT_HANDLE value, but the SAF Forwarder did not recognize the client handle. Unified CM will attempt to reregister with the SAF Forwarder without a client handle. This alarm is for informational purposes only; no action is required.</td>
</tr>
<tr>
<td>472</td>
<td><strong>INFO LEVEL</strong> SAF_VERSION_NUMBER_TOO_LOW - Unified CM published a service (such as Hosted DN) whose version number is now lower than when it was previously published to the SAF Forwarder. The service is out of sync with the SAF Forwarder. Unified CM will republish the service in an attempt to resynch with the SAF Forwarder. This alarm is for informational purposes only; no action is required.</td>
</tr>
<tr>
<td>473</td>
<td><strong>INFO LEVEL</strong> SAFUNKNOWN_SERVICE - Unified CM attempted to unpublish a service from the SAF network but the SAF Forwarder does not have a publish record for that service. This alarm is for informational purposes only; no action is required.</td>
</tr>
</tbody>
</table>
### Error-Level Alarms

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>474 <strong>INFO LEVEL</strong> SAF_UNREGISTERED - Unified CM attempted to publish or subscribe to the SAF Forwarder, but Unified CM is not registered with SAF Forwarder. Unified CM will automatically reregister with the SAF Forwarder before attempting to publish or subscribe. This alarm is for informational purposes only; no action is required.</td>
<td></td>
</tr>
<tr>
<td>475 <strong>INFO LEVEL</strong> SAF_BAD_FILTER - Unified CM attempted to subscribe to the SAF Forwarder with a filter that does not match any of the SAF Forwarder's current filters. Unified CM will resend the subscribe message with the appropriate filter value. This alarm is for informational purposes only; no action is required.</td>
<td></td>
</tr>
<tr>
<td>476 SAF_UNKNOWN_SUBSCRIPTION - Unified CM sent a subscribe or unsubscribe message to the SAF Forwarder but the message contained a Service ID that was not familiar to the SAF Forwarder. Without a recognized Service ID, Unified CM cannot subscribe to the SAF Forwarder. Recommended action is to contact the Cisco Technical Assistance Center (TAC).</td>
<td></td>
</tr>
<tr>
<td>477 <strong>INFO LEVEL</strong> SAF_ALREADY_REGISTERED - Unified CM attempted to register with the SAF Forwarder but SAF Forwarder indicates that Unified CM is already registered. Unified CM will close and reopen the TCP connection and send a new register request without a client handle to SAF Forwarder. This alarm is for informational purposes only; no action is required.</td>
<td></td>
</tr>
<tr>
<td>478 SAF_UNSUPPORTED_PROTOCOL_VERSION - Unified CM attempted to register with the SAF Forwarder using a SAF protocol version number that is greater than the protocol version number supported by the SAF Forwarder. Issue a show version command on the SAF Forwarder CLI to determine the SAF Forwarder protocol version; refer to the information in this alarm for the SAF protocol version number. If the versions do not match, check the Cisco Unified Communications Manager Software Compatibility Matrix (available on Cisco.com) to determine whether the SAF protocol version number that is in use on this Unified CM is compatible with the SAF Forwarder protocol version. If it is not, upgrade the lower-versioned component so that both Unified CM and the SAF Forwarder use the same, compatible version.</td>
<td></td>
</tr>
<tr>
<td>479 SAF_UNKNOWN_AS - Unified CM attempted to register to the SAF Forwarder but the registration message contained a Client Label that was not familiar to the Autonomous System (AS) on the SAF Forwarder router. Recommended action is to issue the appropriate CLI commands on the SAF Forwarder to associate the Client Label with the autonomous system on the router (refer to the Configuration Guide for the router) and configure the same Client Label in the Client Label field on the SAF Forwarder Configuration window in Cisco Unified CM Administration and click Save. When the Client Label is saved in Cisco Unified CM Administration, Unified CM automatically sends a new registration request to the SAF Forwarder with the updated Client Label information.</td>
<td></td>
</tr>
</tbody>
</table>
SAFResponderError

SAF Responder Error 500.
This is raised when SAF forwarder doesn't know the transaction ID within SAF response from this Cisco Unified CM.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
ERROR

Routing List
SDL
SDI
Sys Log
Event Log
Data Collector

Parameter(s)
Client Handle(String)
Service Id(UInt)
Sub Service ID(UInt)
Instance ID1(UInt)
Instance ID2(UInt)
Instance ID3(UInt)
Instance ID4(UInt)

Recommended Action
No action is required.
ScheduledCollectionError

An error occurred while executing scheduled collection.

Facility/Sub-Facility
CCM_TCT-LPMTCT

Cisco Unified Serviceability Alarm Definition Catalog
System/LpmTct

Severity
Error (3)

Parameters
JobID [String] Reason [String]

Recommended Action
Review configuration for scheduled collection job under Job Status window.

SerialPortGetStatusError

When CMI tries to get the status of serial port, the operating system returns an error.
CMI triggers this alarm when it cannot get the status of the serial port. An inability to receive the serial port status information can be caused by a loose or disconnected USB cable.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kSerialPortGetStatusError.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CMIAlarmCatalog/CMI

Severity
ERROR

Routing List
Event Log
SDI

Parameter(s)
Serial Port Getting Status Error(String)
Recommended Action
Make sure that the cable connecting the USB0 port and voice messaging system is firmly connected.

SerialPortSetStatusError

When CMI tries to set the status of serial port, the operating system returns an error. CMI triggers this alarm when it cannot set the status of the serial port. An inability to receive the serial port status information can be caused by a loose or disconnected USB cable.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kSerialPortSetStatusError.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CMIAAlarmCatalog/CMI

Severity
ERROR

Routing List
Event Log
SDI

Parameter(s)
Serial Port Setting Status Error(String)

Recommended Action
Make sure that the cable connecting the USB0 port and voice messaging system is firmly connected.

ServiceActivationFailed

Failed to activate a service.

Facility/Sub-Facility
CCM_SERVICEMANAGER-GENERIC

Cisco Unified Serviceability Alarm Definition Catalog
System/Service Manager

Severity
Error (3)
### Error-Level Alarms

#### Parameters
- Service Name (String)
- Reason (String)

#### Recommended Action
- None

### ServiceDeactivationFailed

Failed to deactivate a service.

#### Facility/Sub-Facility
- CCM_SERVICEMANAGER-GENERIC

#### Cisco Unified Serviceability Alarm Definition Catalog
- System/Service Manager

#### Severity
- Error (3)

#### Parameters
- Service Name (String)
- Reason (String)

#### Recommended Action
- None

### ServiceFailed

Service terminated.

#### Facility/Sub-Facility
- CCM_SERVICEMANAGER-GENERIC

#### Cisco Unified Serviceability Alarm Definition Catalog
- System/Service Manager

#### Severity
- Error (3)

#### Parameters
- Service Name (String)
- Process ID (Int)

#### Recommended Action
- None
ServiceStartFailed

Failed to start service.

Facility/Sub-Facility
CCM_SERVICEMANAGER-GENERIC

Cisco Unified Serviceability Alarm Definition Catalog
System/Service Manager

Severity
Error (3)

Parameters
Service Name(String)
Reason(String)

Recommended Action
None

ServiceStopFailed

Failed to stop service.

Facility/Sub-Facility
CCM_SERVICEMANAGER-GENERIC

Cisco Unified Serviceability Alarm Definition Catalog
System/Service Manager

Severity
Error (3)

Parameters
Service Name(String)
Reason(String)

Recommended Action
None

ServiceExceededMaxRestarts

Service exceeded maximum allowed restarts.

Facility/Sub-Facility
CCM_SERVICEMANAGER-GENERIC
SIPNormalizationResourceWarning

The normalization script has exceeded an internal resource threshold.

The normalization script for the indicated SIP device has exceeded an internal threshold for resource consumption. This alarm can occur for memory consumption, or when the script is close to exceeding the configured allowance of Lua instructions. When the amount of memory (as defined in the Memory Threshold field) or the number of Lua instructions utilized by this script (as defined by the Lua Instruction Threshold) exceeds an internal threshold, this alarm is triggered.

Examples
1. If the memory threshold is set to 100 KB and the internal threshold is 80%, this alarm will occur when this script has consumed 80 KB of memory. The internal threshold is not configurable and may fluctuate from Cisco Unified CM release to release.
2. If the Lua Instruction Threshold is set to 2000 and the internal threshold is 50%, this alarm will occur when the script has executed 1000 Lua instructions.

This alarm warns that the resources (either memory or Lua instructions) have crossed an internal mark, where investigation into the consumption of those resources may be advisable to ensure the health of the script.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.5(1)</td>
<td>• New alarm for this release.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Error
Routing List
SDI
Sys Log
Event Log

Parameters
Device Name(String)
Script Name(String)
Script Function(String)
Script Type(String)
Reason Code(Enum)
Reason Text(String)
In Use Memory(UInt)
Memory Threshold (UInt)
In Use Lua Instructions(UInt)
Lua Instruction Threshold(UInt)

Enum Definitions - Reason Code

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>InternalLuaInstructionsThreshold—The script exceeds the internal threshold for the number of Lua instructions.</td>
</tr>
<tr>
<td>2</td>
<td>InternalMemoryThreshold—The script exceeds the internal threshold for script memory usage.</td>
</tr>
</tbody>
</table>

Recommended Action
1. Examine the thresholds (Memory Threshold and Lua Instruction Threshold) configured in the SIP Normalization Script Configuration window.
2. Evaluate if the thresholds can be increased (take into consideration the CPU resources and memory when deciding to increase these values), or examine the script to determine if the message handlers can be written more efficiently to reduce the number of instructions in the script.
3. Examine the script for logic errors. If the script is functioning normally but contains extensive logic, consider increasing the value in the Lua Instruction Threshold field. Be aware that more computing resources will be consumed as a result. You can also examine SDI trace files for additional details about this resource condition. For scripts provided by Cisco, contact the Cisco Technical Assistance Center (TAC).
4. Investigate and correct the resource issue before the script closes. When the values that have been configured in the Memory Threshold field, or Lua Instruction Threshold field or both the fields on the SIP Normalization Script Configuration window are met, the script closes and the SIPNormalizationScriptClosed alarm also occurs. For additional information when troubleshooting, check the SIP Normalization counter, MemoryUsagePercentage to learn the current resource usage.

SIPNormalizationScriptError

Description
A script error occurred.

**Explanation**
Cisco Unified CM encountered an error during loading, initializing, or during execution of the SIP normalization script for the indicated SIP device. If the error was due to a resource issue, the SIPNormalizationResourceWarning alarm will also be issued. The Configured Action shown in this alarm may differ from the Resulting Action shown in this alarm because certain errors, such as those occurring during loading or initialization, cannot be configured. If the script closes three times within a 10 minute window due to errors, Cisco Unified CM will follow the configured action three times; on the fourth occurrence of the error, Unified CM disables the script and issues the SIPNormalizationAutoResetDisabled alarm.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.5(1)</td>
<td>• New alarm for this release.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/CallManager

**Severity**
Error

**Routing List**
SDL
SDI
Sys Log
Event Log

**Parameters**
Device Name(String)
Script Name(String)
Script Function(String)
Script Type(String)
Error Code(Enum)
Error Code Text(String)
Error Message(String)
Configured Action(String)
Resulting Action(String)
In Use Memory(UInt)
Memory Threshold(UInt)
In Use Lua Instructions(UInt)
Lua Instruction Threshold(UInt)

**Enum Definitions - Reason Code**

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>LoadError — The script failed to load either due to a syntax error in the script or a resource error; check the Recommended Actions for instructions.</td>
</tr>
<tr>
<td>2</td>
<td>InitializationError — The script encountered a failure while initializing either due to a syntax error in the script or a resource error; check the Recommended Actions for instructions.</td>
</tr>
<tr>
<td>3</td>
<td>ExecutionError — The script encountered a failure during execution; check the Recommended Actions for instructions.</td>
</tr>
<tr>
<td>4</td>
<td>InternalError — The system encountered an unexpected condition during execution; check the Recommended Actions for instructions.</td>
</tr>
</tbody>
</table>

**Recommended Action**

1. Examine SDI trace files for details regarding the error such as function calls and the call ID. This will help you to troubleshoot the error.

2. Examine the script for syntax or logic errors; for scripts provided by Cisco, contact the Cisco Technical Assistance Center (TAC). If the error was due to a resource issue, the SIPNormalizationResourceWarning alarm will also be issued. Check the SIPNormalizationResourceWarning alarm for additional information and recommended actions.

**SIPTrunkOOS**

All remote peers are out of service and unable to handle calls for this SIP trunk.

This alarm provides the list of unavailable remote peers, where each peer is separated by semicolon. It also provides the reason code received by the SIP trunk, in response to an Options request sent to remote peer. For each peer, the alarm provides the hostname or SRV (if configured on SIP trunk), resolved IP address, port number, and reason code in the following format:

ReasonCodeType=ReasonCode.

The ReasonCodeType depends on a SIP response from the remote peer as defined in SIP RFCs (remote), or depends on a reason code provided by Unified CM (local).

The examples of possible reason codes include:

- Remote = 503 ("503 Service Unavailable" a standard SIP RFC error code)
- Remote = 408 ("408 Request Timeout" a standard SIP RFC error code)
- Local = 1 (request timeout)
- Local = 2 (local SIP stack is unable to create a socket connection with remote peer)
- Local = 3 (DNS query failed)

For Local=3, IP address in the alarm is represented as zero, and when DNS SRV is configured on SIP trunk then the port is represented as zero.
## Error-Level Alarms

### Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

### Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

### Severity

Error

### Routing List

SDL
SDI
Sys Log
Event Log

### Parameters

SIP Trunk Name(String)
Unavailable remote peers with Reason Code(String)

### Recommended Action

- For Remote = 503, the possible reasons include:
  - Route/SIP trunk for originating side does not exist on remote peer. If remote peer is Unified CM, add a new SIP trunk in Unified CM Administration for the remote peer (Device > Trunk) and ensure the Destination Address and Destination Port fields are configured to point to the originating host (the originating host is the same node on which this alarm was generated).
  - Route/SIP trunk for originating side does exist on remote peer but the port is either used for a SIP phone or a different SIP trunk. If remote peer is Unified CM, in the Unified CM Administration for the remote peer (Device > Trunk), ensure the Destination Port on the originating side is configured to be the same as the incoming port on the terminating side SIP Trunk Security Profile.
  - Remote peer has limited resources to handle new calls. If remote peer is administered by a different system administrator, communicate the resource issue with the other administrator.

- For Remote = 408, the possible reason includes:
  - Remote peer has limited resources to handle new calls. If remote peer is administered by a different system administrator, communicate the resource issue with the other administrator.

- For Local = 1, the possible reason could be that no responses are received for OPTIONS request after all retries, when UDP transport is configured in the SIP trunk Security Profile assigned to the SIP trunk on the originating side.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.5(1)</td>
<td>• New alarm for this release.</td>
</tr>
</tbody>
</table>

Cisco Unified Communications Release 8.5(1) • New alarm for this release.
To fix this issue, perform the following steps:

- If remote peer is Unified CM, in the remote peer Serviceability application, choose **Tools > Control Center (Feature Services)** and ensure the Cisco CallManager service is activated and started.

- In the Unified CM Administration for the remote peer, choose **Device > Trunk**, and ensure the SIP trunk exists with the incoming port in associated SIP Trunk Security Profile configured to be same as originating side SIP Trunk destination port.

- Check the network connectivity by using the CLI command `utils network ping <remote peer>` at the originating side.

• For Local = 2, the possible reason could be that Unified CM is unable to create the socket connection with remote peer.

To fix this issue, perform the following steps:

- If remote peer is Unified CM, in the remote peer Serviceability application, choose **Tools > Control Center (Feature Services)** and ensure the Cisco CallManager service is activated and started.

- In the Unified CM Administration for the remote peer, choose **Device > Trunk** and ensure the SIP trunk exists with the incoming port in associated SIP Trunk Security Profile configured to be same as originating side SIP Trunk destination port.

- Check the network connectivity by using the CLI command `utils network ping <remote peer>` at the originating side.

• For Local = 3, the possible reason could be that DNS server is not reachable, or DNS is not properly configured to resolve the hostname or SRV which is configured on the local SIP trunk.

To fix this issue, perform the following steps:

a. In the OS Administration, choose** Show > Network** and verify whether the DNS details are correct. If it is not correct, configure the correct DNS server information by using the CLI command `set network dns primary`.

b. Check the network connectivity with DNS server by using the CLI command `utils network ping <remote peer>`, and ensure the DNS server is properly configured.

---

### SparePartitionLowWaterMarkExceeded

The percentage of used disk space in the spare partition has exceeded the configured low water mark.

**Note**

Spare Partition is not used for Intercompany Media Engine server. So this alert will not be triggered for Intercompany Media Engine.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
</tbody>
</table>
Error-Level Alarms

Facility/Sub-Facility
CCM_TCT-LPMTCT

Cisco Unified Serviceability Alarm Definition Catalog
System/LpmTct

Severity
Error (3)

Parameters
UsedDiskSpace [String] MessageString [Optional]. [String]

Recommended Action
Login into RTMT and check the configured threshold value for LogPartitionLowWaterMarkExceeded alert in Alert Central. If the configured value is set to a lower than the default threshold value unintentionally, change the value to default. Also, examine the trace and log file setting for each of the application in trace configuration page under Cisco Unified CM Serviceability.
If the number of configured traces or logs is set to greater than 1000, adjust the trace settings from trace configuration page to default. Also, clean up the trace files that are less than a week old. You can clean up the traces using cli "file delete" or using Remote Browse from RTMT Trace and Log Central function.

SystemResourceError

A system call failed.

Facility/Sub-Facility
CCM_SERVICEMANAGER-GENERIC

Cisco Unified Serviceability Alarm Definition Catalog
System/Service Manager

Severity
Error (3)

Parameters
System Call(String)
Service(String)
Reason(String)

Recommended Action
None

TestAlarmError

Testing error alarm.
Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
System/Test

Severity
Error (3)

Recommended Action
None

ThreadPoolProxyUnknownException

Unknown exception was caught while processing file request. This usually indicates a lack of memory when there is a system issue such as running out of resources.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kThreadPoolProxyUnknownException.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TFTP-TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP

Severity
Error (3)

Recommended Action
Use RTMT to monitor the system memory resources and consumption and correct any system issues that might be contributing to a reduced amount of system resources.

UnableToRegisterwithCallManagerService

CTI cannot communicate with Cisco CallManager service to register supplementary service features.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

Severity
ERROR
Error-Level Alarms

Routing List
SDL
SDI
Sys Log
Event Log

Recommended Action
Check the status of the Cisco CallManager service in Cisco Unified Serviceability > Tools > Control Center - Featured Services. At least one Cisco CallManager service should be running in the cluster for CTIManager to register feature managers. Restart the CTIManager service if the problem persists. If CallManager service is active, verify network connectivity between the Unified CM node that hosts CTIManager service and Unified CM node that hosts CallManager service.

UserLoginFailed
User login failed because of bad user ID or password.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Error (3)

Parameters
UserID [String]

Recommended Action
None

WritingFileFailure
CMI failed to write SMDI messages to the serial port.
CMI opened the serial port, however it failed to successfully write data to the serial port because the serial port returned an invalid handle value to CMI. The serial port may have returned an invalid handle because the system did not properly detect the USB cable.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kWritingFileFailure.</td>
</tr>
</tbody>
</table>
Cisco Unified Serviceability Alarm Definition Catalog
CMIAAlarmCatalog/CMI

Severity
ERROR

Routing List
Event Log
SDI

Parameter(s)
Error Information(String)

Recommended Action
Make sure that the cable connecting the USB0 port and voice messaging system is firmly connected.

**WDAppli**cation**Error**

WebDialer Facility/Sub-Facility error.

**Facility/Sub-Facility**
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Error (3)

Parameters
Servlet Name [String] Reason [String]

Recommended Action
See application logs for details

**WDOverloa**ded

WebDialer Facility/Sub-Facility overloaded.

**Facility/Sub-Facility**
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications
Warning-Level Alarms

The warning-level alarm is 4 and action is needed but priority of action is determined by the condition. A warning about some bad condition, which is not necessarily an error. Configuration error or an alarm that by itself does not indicate a warning but several instances of the same alarm do. Examples are:

- Configuration error
- One alarm of this level may not mean that an error has occurred but multiple of these would be considered an error

AnnunciatorNoMoreResourcesAvailable

No more Annunciator resources available.
Annunciator resource allocation failed for one or more of the following reasons: all Annunciator resources are already in use; there was a codec or capability mismatch (such as the endpoint using one type of IP addressing such as IPv6, while the Annunciator supports only IPv4) between the endpoint and the Annunciator resource; not enough bandwidth existed between the endpoint and the Annunciator.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed severity level from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity

Warning

Parameter(s)

Media Resource List Name(String)
Recommended Action

If all the resources of the Annunciator are already in use, check to be sure that all the Annunciators that belong to the Media Resource Groups of the indicated Media Resource Group List and Default List are configured and registered in all the applicable Unified CM nodes of the cluster. To check the registration status go to the Media Resources > Annunciator menu and click the Find button. It will display all the Annunciators with their status, device pool, and so on.

Check the status field to see whether it is registered with Unified CM. Note that the display on the status field is not a confirmation that the device is registered to Unified CM. It may happen in a Unified CM cluster that the Publisher can only write to the Unified CM database before the Publisher goes down. Because the Subscriber may not be able to write to the database, the devices may still display registered in Unified CM Administration after they are actually unregistered. However, if the Publisher is down that should generate another alarm with higher priority than this alarm.

The Annunciator allocation can fail due to codec mismatch or capability mismatch between the endpoint and the Annunciator. If there is a codec mismatch or capability mismatch (such as the endpoint using IPv6 addressing but Annunciator supporting only IPv4), an MTP or transcoder should be allocated. So, if the MTP or transcoder is not allocated then either MediaResourceListExhausted (with Media Resource Type as Media termination point or transcoder) or MtpNoMoreResourcesAvailable alarm will be generated for the same Media Resource Group List and you should first concentrate on that.

The Annunciator allocation may even fail after checking the region bandwidth between the regions to which the held party belongs and the region to which the Annunciator belongs. Increasing the region bandwidth may be a solution to the problem, but that decision should be made after careful consideration of the amount of bandwidth you’re willing to allocate per call between the set of regions. You'll need to weigh different factors such as the total amount of available bandwidth, the average number of calls, the average number of calls using the Annunciator, approximate bandwidth use per call, and so on, and accordingly calculate the region bandwidth.

Another possible cause is that the bandwidth needed for the call may not be available. This can happen if the Annunciator and endpoint belong to different locations and the bandwidth that is set between the locations is already in use by other calls. Examine the bandwidth requirements in your deployment to determine whether bandwidth between the locations can be increased.

However, note that increasing the bandwidth between these two locations means that you may need to reduce the bandwidth between other locations. Refer to the System Guide, SRNDs, and related Unified CM documentation for more details. Be aware that reducing the bandwidth or removing the higher bandwidth codecs from configuration may result in poor voice quality during call. Consider increasing the total amount of network bandwidth.

ApplicationConnectionDropped

Application has dropped the connection to CTIManager.
TCP or TLS connection between CTIManager and Application is disconnected.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

Severity
WARNING

Routing List
SDL
Warning-Level Alarms

SDI
Sys Log
Event Log

**Recommended Action**
Possible causes include Application server power outage, network power outage, network configuration error, network delay, packet drops or packet corruption. It is also possible to get this error if the Unified CM node or application server is experiencing high CPU usage. Verify the application is up and running, verify network connectivity between the application server and Unified CM, and verify the CPU utilization is in the safe range for application server and Unified CM (this can be monitored using RTMT via CPU Pegging Alert).

ApplicationConnectionError

CTIManager is unable to allow connections from Applications.

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/CtiManager

**Severity**
WARNING

**Routing List**
SDL
SDI
Sys Log
Event Log

**Parameter(s)**
CTI Connection type(String)

**Recommended Action**
CTIManager has encountered problems initializing TCP connections. Restart the CTIManager service to resolve this problem.

authAdminLock

User is locked out by administrator.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
</tbody>
</table>
Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Warning (4)

Parameters
lock(String)

Recommended Action
Administrator can unlock this user.

AuthenticationFailed

Login Authentication failed.

Facility/Sub-Facility
CCM_TOMCAT_APPS-LOGIN

Cisco Unified Serviceability Alarm Definition Catalog
System/Login

Severity
Warning

Parameters
Login IP Address/Hostname [String] Login Date/Time [String] Login UserID [String] Login Interface [String]

Recommended Action
If this event happens repeatedly, investigate the source of the failed login attempts.

authFail

Failed to authenticate this user.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Changed severity level from Notice to Warning.</td>
</tr>
<tr>
<td>8.5(1)</td>
<td>Updated parameters.</td>
</tr>
</tbody>
</table>
Warning-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Warning (4)

Parameters
UserID(String)
Message(String)

Recommended Action
Determine correct credentials and retry.

authHackLock

User attempted too many incorrect authentications. The maximum number of attempts gets set by the administrator.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Added more descriptive text and corrected the parameter.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Warning (4)

Parameters
UserID(String)

Recommended Action
Wait for administrator specified time to retry, or have administrator unlock the credential.
authInactiveLock

The user has been inactive for a specified time and the credential is locked.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Changed parameter text.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Warning (4)

Parameters
UserID(String)

Recommended Action
Reset credential.

authLdapInactive

Authentication failed because the user exists in the database and the system specifies LDAP authentication. A directory sync got performed in the immediate past (1 day).

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Revised the description and added text to Recommended Action.</td>
</tr>
<tr>
<td>8.5(1)</td>
<td>Parameter updated.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Warning (4)

Parameters
UserID(String)
**Recommended Action**
This user has yet to be removed from the database or the alarm will clear itself within 24 hours.

**BDIStopped**
BDI Application stopped. Application was unloaded from Tomcat.

**Facility/Sub-Facility**
CCM_JAVA_APPS-TOMCATAPPLICATIONS

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Java Applications

**Severity**
Warning (4)

**Recommended Action**
Check if Tomcat service is up.

**CallAttemptBlockedByPolicy**
A call was attempted but blocked or rejected by the policy decision point (PDP).
A call was rejected or blocked because it violated the enterprise policy as defined in a policy decision point (PDP) that was configured in Cisco Unified Communications Manager (Unified CM). The policy server returns a call reject decision stating that a policy violation was the reason for rejecting the call. Calls may be rejected because an unauthorized user attempted to dial a DN or pattern that is not allowed for him or her or because a call forward directive was invoked and the destination specified in the call forward operation violated the policy. Depending on email configuration in Real-Time Monitoring Tool (RTMT), the system may have generated an email alert when the call was rejected.

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/CallManager

**Severity**
WARNING

**Routing List**
SDL
SDI
Sys Log
Event Log
Alert Manager

**Parameter(s)**
Policy Decision Point(String)
Warning-Level Alarms

Reject Reason(String)
Called Party Number(String)
Calling Party Number(String)
Calling User Id(String)

Recommended Action
Evaluate the information provided in this alarm (caller's user ID, to and from DNs, and so on) to determine if the call attempt was an innocent mistake to dial a number that the user didn't realize was not routable for him or her, or to discover whether the user is intentionally trying to circumvent the policy restrictions. If the rejected call was caused by an innocent mistake, educate the affected user about the numbers that he or she is allowed to dial. Your organization may have a policy or guidelines to follow when investigating call rejects. In addition to or instead of the steps recommended here, please refer to your company's guidelines.

CCDLearnedPatternLimitReached

CCD has reached the maximum number of learned patterns allowed.
The CCD requesting service has limited the number of learned patterns to a number defined in the service parameter, CCD Maximum Numbers of Learned Patterns. This alarm indicates that the CCD requesting service has met the maximum number of learned patterns allowed.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
WARNING

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
CCD Maximum Numbers of Learned Patterns (UInt)
System Limit of CCD Learned Patterns (UInt)

Recommended Action
This alarm displays the value that is configured in the Cisco CallManager service parameter, CCD Maximum Numbers of Learned Patterns, as well as the maximum number of learned patterns that are allowed by the system (an internally-controlled maximum).
Consider whether the specified maximum number of learned patterns is correct for your deployment. If it is too low, compare it with the number shown in the SystemLimitCCDLearnedPatterns in this alarm. If the Max number is below the System Limit, you can go to the Service Parameters Configuration
Warning-Level Alarms

window and increase the CCD Maximum Numbers of Learned Patterns service parameter. If the Max and System Limit numbers match, the system is already configured to run at capacity of learned patterns; no action is required.

**CDRHWMExceeded**

The CDR files disk usage has exceeded the High Water Mark. CDRM deleted some successfully delivered CDR files that are still within the preservation duration, in order to bring the disk usage down to below HWM. E-mail alert will be sent to the admin.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed Data Collector Routing List element to Alert Manager.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CDRREP

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CDR Rep

**Severity**

Warning (4)

**Routing List**

Event Log
Sys Log
Alert Manager

**Parameters**

DiskUsageInMB [String]

**Recommended Action**

The preservation duration may be too long. Reduce it at serviceability->tools->CDRM Configuration. Or raise maximum allocated disk space and/or HWM for CDR files.

**CertValidLessThanMonth**

Alarm indicates that the certificate will expire in 30 days or less.
Cisco Unified Serviceability Alarm Definition Catalog
System/CertMonitorAlarmCatalog

Severity
Warning(4)

Routing List
Event Log
Sys Log

Parameters
Message(String)

Recommended Action
Regenerate the certificate that is about to expire by accessing the Cisco Unified Operating System and go to Certificate Management. If the certificate is issued by a CA, generate a CSR, submit the CSR to CA, obtain a fresh certificate from CA, and upload it to Cisco Unified CM.

ConferenceNoMoreResourcesAvailable

Conference resource allocation failed for one or more of the following reasons: the required number of conference resources were not available; for an IOS-based conference bridge, the number of participants to be added to the conference bridge exceeded the maximum number of participants allowed per conference; no lower precedence conference was available for preemption although MLPP preemption was enabled; a lower-precedence conference bridge was not preempted.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed severity level from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning

Parameter(s)
Media Resource List Name(String)
Recommended Action

For IOS-based conference bridges, make sure that the maximum number of participants configured in a conference bridge does not exceed the number of participants allowed per conference; please check the IOS-based conference bridge user manual for limitations on the number of participants. Also, be sure to educate end users about the maximum number of participants allowed. For IOS-based and non-IOS-based, consider installing additional conference resources.

CtiDeviceOpenFailure

Application is unable to open the device.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiDeviceOpenFailure.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CtiManager

Severity

WARNING

Routing List

SDL
SDI
Sys Log
Event Log
Data Collector

Parameter(s)

Device Name(String)
ReasonCode(Enum)

Enum Definitions - Reason Code

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x8CCC0013 (2362179603)</td>
<td>Device is already opened by another application; identify the application that is controlling this device. You can determine this information from RTMT (CallManager-&gt;CTI Manager and CallManager-&gt;CTI Search)</td>
</tr>
<tr>
<td>0x8CCC00DA (2362179802)</td>
<td>Unable to communicate with database; verify the CPU utilization is in the safe range for (this can be monitored using RTMT via CPU Pegging Alert)</td>
</tr>
</tbody>
</table>
## Value | Definition
--- | ---
0x8CCC009A (2362179738) | Device is unregistering; wait for the device to register. Due to user initiated reset or restart of the device from Unified CM. Device should automatically register wait for few moments for the device to register.
0x8CCC0018 (2362179608) | Device is not in the user control list; verify whether the device is configured for control by this application. For the application to control the device it should be included in the user control list. To check whether the device is in the user control list, if the application uses an End User, check the Device Association section under the End User Configuration in Cisco Unified CM Administration (User Management > End User). If the application uses an Application User, check under Device Information section for that Application User in Cisco Unified CM Administration (User Management > Application User).
0x8CCC00F3 (2362179827) | IPAddress mode (IPv4 or IPv6 or both) specified by the application does not match with IP Addressing mode that is configured in Unified CM Administration; check the IP addressing mode of the device in Cisco Unified CM Administration (Device > Device Settings > Common Device Configuration).

### Recommended Action
Check the reason code and take appropriate action to resolve the issue.

### CtiLineOpenFailure
Application is unable to open the line.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiLineOpenFailure.</td>
</tr>
</tbody>
</table>

### Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

### Severity
WARNING

### Routing List
SDL, SDI, Sys Log, Event Log
Data Collector

**Parameter(s)**
- Device Name(String)
- Directory Number(String)
- Partition(String)
- Reason(Enum)

**Enum Definitions - Reason Code**

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown</td>
</tr>
<tr>
<td>0x8CCC0018</td>
<td>Device is not in the user control list; verify whether the device is configured for control by this application. For the application to control the device it should be included in the user control list. To check whether the device is in the usercontrol list, if the application uses an End User, check the Device Association section under the End User Configuration in Cisco Unified CM Administration (User Management &gt; End User). If the application uses an Application User, check under Device Information section for that Application User in Cisco Unified CM Administration (User Management &gt; Application User)</td>
</tr>
<tr>
<td>0x8CCC0005</td>
<td>Line is not found in the device; possible cause could be that the line that previously existed on this device is not available. This could be due to a extension mobility login or logout</td>
</tr>
<tr>
<td>0x8CCC00D3</td>
<td>Administrator has restricted the Line to be controllable by application. If the intent of the Administrator is to allow control of this line, enable the check box labelled Allow control of Device from CTI, in Unified CM Administration under Call Routing &gt; Directory Number and choose the line that should be controlled by this application</td>
</tr>
</tbody>
</table>

**Recommended Action**
Review the reason code and take appropriate action to resolve the issue.

### CtiIncompatibleProtocolVersion

Incompatible protocol version.

The JTAPI/TAPI application version is not compatible with this version of CTIManager, so the received message has been rejected. The IP address is shown in either IPv4 or IPv6 format depending on the IP addressing mode of the Application.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiIncompatibleProtocolVersion.</td>
</tr>
</tbody>
</table>
Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

Severity
WARNING

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Unified CM Version(String)
IPAddress(String)
IPv6Address(String)

Recommended Action
Verify that the correct version of the application is being used. If you are not sure of the correct version, contact the application vendor and upgrade the JTAPI/TSP to the version provided by Cisco Unified Communications Manager. JTAPI/TSP plugins are available in Cisco Unified CM Administration (Application > Plugins).

CtiMaxConnectionReached

Maximum number of CTI connections has been reached, no new connection will be accepted unless an existing connection is closed.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiMaxConnectionReached.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

Severity
WARNING

Routing List
SDL
SDI
Sys Log
Warning-Level Alarms

Event Log

**Recommended Action**
Check the CTI Manager service parameter Maximum CTI Connections for the maximum number of connections. Carefully, consider increasing the service parameter value or disconnecting CTI applications that are unnecessary. Refer to Unified CM Solution Reference Network Design document in www.cisco.com based on the version you are using for maximum number of applications and devices supported by CTI.

**CtiProviderCloseHeartbeatTimeout**

CTI heartbeat timeout occurred causing CTIManager to close the application connection.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiProviderCloseHeartbeatTimeout.</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/CtiManager

**Severity**
WARNING

**Routing List**
SDL
SDI
Sys Log
Event Log

**Recommended Action**
Heartbeat timeout could occur due to high CPU usage or network connectivity problems. Check for and fix any network issues or high CPU usage on the application server. If the application server is running the Microsoft Windows OS use Task Manager or Perfmon to determine the CPU usage. For applications in Linux use the top command to review CPU usage.

**CtiQbeFailureResponse**

The requested operation from the application could not be performed because of a normal or abnormal condition.
Warning-Level Alarms

### Cisco Unified Serviceability Alarm Definition Catalog

**CallManager/CtiManager**

**Severity**

WARNING

**Routing List**

SDL
SDI
Sys Log
Event Log

**Parameter(s)**

Error message(String)

**Recommended Action**

Verify whether the affected application is experiencing a problem. Contact the support organization for the affected application if the problem persists and provide sequence number and error message for further investigation.

---

**DaTimeOut**

The digit analysis component in Cisco Unified Communications Manager has timed out. This can occur because Cisco Unified Communications Manager is busy and the resulting delay in processing request and response messages caused the digit analysis component to time out.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CallManager
Warning-Level Alarms

Severity
Warning

Recommended Action
In the Service Parameter Configuration window in Cisco Unified CM Administration, check the Cisco CallManager service parameter, Digit Analysis Timer, to confirm that the default value is in use. Use RTMT to monitor the system resources and correct any system issues that might be contributing to high CPU utilization on Cisco Unified CM.

DeviceImageDownloadFailure

Cisco IP Phone failed to download its image.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.5(1)</td>
<td>Enum Definitions for FailureReason.</td>
</tr>
<tr>
<td>7.1</td>
<td>Added DeviceImageDownloadFailure to the Phone Catalog.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog

Severity
Warning (4)

Parameters
DeviceName(String)
IPAddress(String)
Active(String)
Inactive(String)
FailedLoadID(String)
Method(Enum)
FailureReason(Enum)
Server(String)

Enum Definitions for Method

<table>
<thead>
<tr>
<th>Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TFTP</td>
</tr>
<tr>
<td>2</td>
<td>HTTP</td>
</tr>
<tr>
<td>3</td>
<td>PPID</td>
</tr>
</tbody>
</table>
Warning-Level Alarms

Enum Definitions for FailureReason

<table>
<thead>
<tr>
<th>Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A TFTP server error occurred - examine the TFTP log to determine whether other errors occurred at the same time the device was attempting to download its firmware and correct any TFTP errors that may have occurred. Also, investigate the load on the TFTP server to ensure that device download requests are being processed; check network connectivity to the TFTP server.</td>
</tr>
<tr>
<td>2</td>
<td>Specified firmware load ID is not found on the TFTP server. Check that file name is correct, or load (image) file exist on TFTP server.</td>
</tr>
<tr>
<td>3</td>
<td>An internal phone error occurred during the download attempt; reset the phone to correct the issue.</td>
</tr>
<tr>
<td>4</td>
<td>The Load server or TFTP server could not process the phone's firmware load request. It's possible that congestion is causing a delay in TFTP response. To allow the phone to attempt the download again, wait a few minutes then reset the phone. The phone will attempt to download its firmware load again. If resetting the phone does not solve the issue, restart the Load server or TFTP server (whichever server provides firmware loads).</td>
</tr>
<tr>
<td>5</td>
<td>An encryption error occurred on the phone while trying to load the new firmware load (image); reset the phone to correct the issue.</td>
</tr>
<tr>
<td>6</td>
<td>The downloaded firmware load (image) is not encrypted. Verify that correct load (image) name is provided to the phone and that the server that provides firmware loads has that encrypted load (image) file.</td>
</tr>
<tr>
<td>7</td>
<td>The downloaded firmware load (image) cannot be decrypted using the decryption key on the phone (resulting in an encryption key mismatch). If you have provided the image encryption key, try re-encrypting the image with the key that matches the key already on the phone, then attempt the download again. Otherwise, collect the phone logs from the time of this alarm (review the steps in the Administration Guide for the appropriate phone model to learn how to access the phone logs) and contact the Cisco Technical Assistance Center (TAC).</td>
</tr>
<tr>
<td>8</td>
<td>There is a problem with the encryption of the downloaded firmware load (image). Collect pertinent details such as the device's MAC address, device type, the firmware load ID, and phone logs from the time of this alarm (review the steps in the Administration Guide for the appropriate phone model to learn how to access the phone logs) and contact the Cisco Technical Assistance Center (TAC).</td>
</tr>
<tr>
<td>9</td>
<td>The phone did not receive a load server name or IP address and as a result, does not have the server information needed to download a firmware load. Check the Device Configuration page in Cisco Unified CM Administration to ensure that the IP address of the Load server or TFTP server is accurately configured. If the information is inaccurate or not present, supply the correct information and restart the phone. If the information is accurate, restart the phone. If this alarm recurs, contact the Cisco Technical Assistance Center (TAC).</td>
</tr>
<tr>
<td>10</td>
<td>The phone attempted an action that is not allowed by the Load server or TFTP server; reset the phone to attempt to clear the condition.</td>
</tr>
</tbody>
</table>
Warning-Level Alarms

Chapter 6   Cisco Unified Serviceability Alarms and CiscoLog Messages

<table>
<thead>
<tr>
<th>Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>The device has exceeded the internally-configured time allowed for a response from the Load server or TFTP server when requesting the firmware load file. It's possible that congestion is causing a delay in TFTP response. To allow the phone to attempt the download again, wait a few minutes then reset the phone. The phone will attempt to download the file again. If resetting the phone does not solve the issue, restart the Load server or TFTP server (whichever server provides the firmware load files).</td>
</tr>
<tr>
<td>14</td>
<td>The data that the phone received from the Load server or TFTP server was not intact; not enough information was received. Restart the phone to begin the download process again.</td>
</tr>
<tr>
<td>15</td>
<td>The data that the phone received from the Load server or TFTP server was not intact; too much information was received. Restart the phone to begin the download process again.</td>
</tr>
<tr>
<td>16</td>
<td>The phone cannot connect to the network; check for network connectivity to the image firmware load server or the TFTP server and correct any broken connection. Restart the phone to attempt connection again unless the restart occurs automatically.</td>
</tr>
<tr>
<td>17</td>
<td>The DNS server name that the phone is attempting to connect to could not be resolved. Examine the DNS server name(s) in the phone settings to verify that the information is accurate and if not, update the name on the phone. Restart the phone unless the restart occurs automatically.</td>
</tr>
<tr>
<td>18</td>
<td>No DNS server - Configure a DNS server IP address on the phone settings. Restart the phone unless the restart occurs automatically.</td>
</tr>
<tr>
<td>19</td>
<td>Connection to the Load server or TFTP server has timed out - The phone attempted to connect to the Load server or TFTP server but could not connect successfully. If you are using the TFTP server to serve firmware loads, check the TFTP server IP address as configured in the settings on the phone; make sure the IP address is accurate. If it is not, correct the IP address and press Apply; the phone should restart automatically. If you are using a Load server to serve firmware loads, check the IP address or hostname on the Phone Configuration page in Cisco Unified CM Administration for the phone identified in this alarm, to ensure that the information is accurate. If it is not, update the IP address or hostname and restart the phone. Also, verify that network connectivity exists between the phone and the Load server or TFTP server. Restart the phone to attempt connection again unless the restart occurs automatically.</td>
</tr>
<tr>
<td>20</td>
<td>Download was cancelled - A previous download request was superseded by a new download request. The original download was cancelled so that the new download could continue. No action is required.</td>
</tr>
</tbody>
</table>

Recommended Action

Verify that the IP address or hostname of the image download server (either the Load server or the TFTP server) is correct. If you're using a hostname, verify that the Domain Name Server (DNS) is accessible from the phone and can resolve the hostname. Verify that the TFTP service is activated and running on the Load server or TFTP server (the server you are using to serve firmware load files). Verify that the Load server or TFTP server is accessible from the phone. Also, refer to the reason code descriptions for recommended actions.

DevicePartiallyRegistered

Device partially registered. A device is partially registered with Cisco CallManager. Some, but not all, of the lines configured on the device have successfully registered.
### Warning-Level Alarms

#### Facility/Sub-Facility

**CCM_CALLMANAGER-CALLMANAGER**

#### Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

#### Severity

Warning (4)

#### Parameters


#### Enum Definitions for Performance Monitor Object type

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cisco  allManager</td>
</tr>
<tr>
<td>2</td>
<td>Cisco Phones</td>
</tr>
<tr>
<td>3</td>
<td>Cisco ines _L</td>
</tr>
<tr>
<td>4</td>
<td>Cisco H323</td>
</tr>
<tr>
<td>5</td>
<td>Cisco MGCP Gateway</td>
</tr>
<tr>
<td>6</td>
<td>Cisco MOH Device</td>
</tr>
<tr>
<td>7</td>
<td>Cisco Analog Access</td>
</tr>
<tr>
<td>8</td>
<td>Cisco MGCP FXS Device</td>
</tr>
<tr>
<td>9</td>
<td>Cisco MGCP FXO Device</td>
</tr>
<tr>
<td>10</td>
<td>Cisco MGCP T1CAS Device</td>
</tr>
<tr>
<td>11</td>
<td>Cisco MGCP PRI Device</td>
</tr>
</tbody>
</table>

#### Enum Definitions for DeviceType

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CISCO_30SP+</td>
</tr>
<tr>
<td>2</td>
<td>CISCO_12SP+</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-------</td>
</tr>
<tr>
<td>3</td>
<td>CISCO_12SP</td>
</tr>
<tr>
<td>4</td>
<td>CISCO_12S</td>
</tr>
<tr>
<td>5</td>
<td>CISCO_30VIP</td>
</tr>
<tr>
<td>6</td>
<td>CISCO_7910</td>
</tr>
<tr>
<td>7</td>
<td>CISCO_7960</td>
</tr>
<tr>
<td>8</td>
<td>CISCO_7940</td>
</tr>
<tr>
<td>9</td>
<td>CISCO_7935</td>
</tr>
<tr>
<td>10</td>
<td>CISCO_VGC_PHONE</td>
</tr>
<tr>
<td>11</td>
<td>CISCO_VGC_VIRTUAL_PHONE</td>
</tr>
<tr>
<td>12</td>
<td>CISCO_ATA_186</td>
</tr>
<tr>
<td>20</td>
<td>SCCP_PHONE</td>
</tr>
<tr>
<td>30</td>
<td>ANALOG_ACCESS</td>
</tr>
<tr>
<td>40</td>
<td>DIGITAL_ACCESS</td>
</tr>
<tr>
<td>42</td>
<td>DIGITAL_ACCESS+</td>
</tr>
<tr>
<td>43</td>
<td>DIGITAL_ACCESS_WS-X6608</td>
</tr>
<tr>
<td>47</td>
<td>ANALOG_ACCESS_WS-X6624</td>
</tr>
<tr>
<td>48</td>
<td>VGC_GATEWAY</td>
</tr>
<tr>
<td>50</td>
<td>CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>51</td>
<td>CONFERENCE_BRIDGE_HARDWARE</td>
</tr>
<tr>
<td>52</td>
<td>CONFERENCE_BRIDGE_HARDWARE_HDV2</td>
</tr>
<tr>
<td>53</td>
<td>CONFERENCE_BRIDGE_HARDWARE_WS-SVC-CMM</td>
</tr>
<tr>
<td>61</td>
<td>H323_PHONE</td>
</tr>
<tr>
<td>62</td>
<td>H323_GATEWAY</td>
</tr>
<tr>
<td>70</td>
<td>MUSIC_ON_HOLD</td>
</tr>
<tr>
<td>71</td>
<td>DEVICE_PILOT</td>
</tr>
<tr>
<td>72</td>
<td>CTI_PORT</td>
</tr>
<tr>
<td>73</td>
<td>CTI_ROUTE_POINT</td>
</tr>
<tr>
<td>80</td>
<td>VOICE_MAIL_PORT</td>
</tr>
<tr>
<td>83</td>
<td>SOFTWARE_MEDIA_TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>84</td>
<td>CISCO_MEDIA_SERVER</td>
</tr>
<tr>
<td>85</td>
<td>CISCO_VIDEO_CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>90</td>
<td>ROUTE_LIST</td>
</tr>
<tr>
<td>100</td>
<td>LOAD_SIMULATOR</td>
</tr>
<tr>
<td>110</td>
<td>MEDIA_TERMINATION_POINT</td>
</tr>
<tr>
<td>111</td>
<td>MEDIA_TERMINATION_POINT_HARDWARE</td>
</tr>
<tr>
<td>112</td>
<td>MEDIA_TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>113</td>
<td>MEDIA_TERMINATION_POINT_WS-SVC-CMM</td>
</tr>
<tr>
<td>115</td>
<td>CISCO_7941</td>
</tr>
</tbody>
</table>
### Warning-Level Alarms

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>119</td>
<td>CISCO_7971</td>
</tr>
<tr>
<td>120</td>
<td>MGCP_STATION</td>
</tr>
<tr>
<td>121</td>
<td>MGCP_TRUNK</td>
</tr>
<tr>
<td>122</td>
<td>GATEKEEPER</td>
</tr>
<tr>
<td>124</td>
<td>7914_14_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>125</td>
<td>TRUNK</td>
</tr>
<tr>
<td>126</td>
<td>TONE_ANNOUNCEMENT_PLAYER</td>
</tr>
<tr>
<td>131</td>
<td>SIP_TRUNK</td>
</tr>
<tr>
<td>132</td>
<td>SIP_GATEWAY</td>
</tr>
<tr>
<td>133</td>
<td>WSM_TRUNK</td>
</tr>
<tr>
<td>134</td>
<td>REMOTE_DESTINATION_PROFILE</td>
</tr>
<tr>
<td>227</td>
<td>7915_12_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>228</td>
<td>7915_24_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>229</td>
<td>7916_12_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>230</td>
<td>7916_24_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>232</td>
<td>CKEM_36_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>254</td>
<td>UNKNOWN_MGCP_GATEWAY</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>302</td>
<td>CISCO_7989</td>
</tr>
<tr>
<td>307</td>
<td>CISCO_7911</td>
</tr>
<tr>
<td>308</td>
<td>CISCO_7941G_GE</td>
</tr>
<tr>
<td>309</td>
<td>CISCO_7961G_GE</td>
</tr>
<tr>
<td>335</td>
<td>MOTOROLA_CN622</td>
</tr>
<tr>
<td>336</td>
<td>BASIC_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>348</td>
<td>CISCO_7931</td>
</tr>
<tr>
<td>358</td>
<td>CISCO_UNIFIED_COMMUNICATOR</td>
</tr>
<tr>
<td>365</td>
<td>CISCO_7921</td>
</tr>
<tr>
<td>369</td>
<td>CISCO_7906</td>
</tr>
<tr>
<td>374</td>
<td>ADVANCED_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>375</td>
<td>CISCO_TELEPRESENCE</td>
</tr>
<tr>
<td>404</td>
<td>CISCO_7962</td>
</tr>
<tr>
<td>412</td>
<td>CISCO_3951</td>
</tr>
<tr>
<td>431</td>
<td>CISCO_7937</td>
</tr>
<tr>
<td>434</td>
<td>CISCO_7942</td>
</tr>
<tr>
<td>435</td>
<td>CISCO_7945</td>
</tr>
<tr>
<td>436</td>
<td>CISCO_7965</td>
</tr>
<tr>
<td>437</td>
<td>CISCO_7975</td>
</tr>
<tr>
<td>446</td>
<td>CISCO_3911</td>
</tr>
</tbody>
</table>
Warning-Level Alarms

**Recommended Action**

In the Cisco Unified Reporting tool, run the Unified CM Multi-Line Devices report and check the number of lines that are supposed to be configured on the device identified in this alarm. If the device has registered an inconsistent number of lines compared the Multi-Lines report for this device, restart the device so that it can reregister all lines. If this alarm persists, verify that the appropriate number of lines has been configured on the device, and that the appropriate directory numbers have been configured. If the device is a third-party SIP phone, verify that the directory numbers configured on the phone match the directory numbers configured on the device in Unified CM Administration.

**DeviceTransientConnection**

A connection was established and immediately dropped before completing registration. Incomplete registration may indicate that a device is rehoming in the middle of registration. The alarm could also indicate a device misconfiguration, database error, or an illegal/unknown device trying to attempt a connection. Network connectivity problems can affect device registration, or the restoration of a primary Unified CM may interrupt registration.

<table>
<thead>
<tr>
<th>Device ID</th>
<th>Device Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>468</td>
<td>CISCO_UNIFIED_MOBILE_COMMUNICATOR</td>
</tr>
<tr>
<td>478</td>
<td>CISCO_TELEPRESENCE_1000</td>
</tr>
<tr>
<td>479</td>
<td>CISCO_TELEPRESENCE_3000</td>
</tr>
<tr>
<td>480</td>
<td>CISCO_TELEPRESENCE_3200</td>
</tr>
<tr>
<td>481</td>
<td>CISCO_TELEPRESENCE_500</td>
</tr>
<tr>
<td>484</td>
<td>CISCO_7925</td>
</tr>
<tr>
<td>493</td>
<td>CISCO_9971</td>
</tr>
<tr>
<td>495</td>
<td>CISCO_6921</td>
</tr>
<tr>
<td>496</td>
<td>CISCO_6941</td>
</tr>
<tr>
<td>497</td>
<td>CISCO_6961</td>
</tr>
<tr>
<td>20000</td>
<td>CISCO_7905</td>
</tr>
<tr>
<td>30002</td>
<td>CISCO_7920</td>
</tr>
<tr>
<td>30006</td>
<td>CISCO_7970</td>
</tr>
<tr>
<td>30007</td>
<td>CISCO_7912</td>
</tr>
<tr>
<td>30008</td>
<td>CISCO_7902</td>
</tr>
<tr>
<td>30016</td>
<td>CISCO_IP_COMMUNICATOR</td>
</tr>
<tr>
<td>30018</td>
<td>CISCO_7961</td>
</tr>
<tr>
<td>30019</td>
<td>CISCO_7936</td>
</tr>
<tr>
<td>30027</td>
<td>ANALOG_PHONE</td>
</tr>
<tr>
<td>30028</td>
<td>ISDN_BRI_PHONE</td>
</tr>
<tr>
<td>30032</td>
<td>SCCP_GATEWAY_VIRTUAL_PHONE</td>
</tr>
<tr>
<td>30035</td>
<td>IP_STE</td>
</tr>
</tbody>
</table>
History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following information is updated:</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for DeviceType</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for IPAddressAttributes</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for IPV6IPAddressAttributes</td>
</tr>
<tr>
<td>7.1</td>
<td>IPv6 parameters added: IPV6IPAddress[Optional][String],</td>
</tr>
<tr>
<td></td>
<td>IPAddressAttributes[Optional][Enum], and</td>
</tr>
<tr>
<td></td>
<td>IPV6IPAddressAttributes[Optional][Enum].</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning

Parameters
Device IP address [Optional].[String]
Device name [Optional].[String]
Device MAC address [Optional].[String]
Protocol.[String]
Device type. [Optional][Enum]
Reason Code [Optional][Enum]
Connecting Port [UInt]
Registering SIP User. [Optional].[String]
IPV6Address [Optional].[String]
IPAddressAttributes [Optional][Enum]
IPV6AddressAttributes [Optional][Enum]

Enum Definitions for DeviceType

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>CISCO_VGC_PHONE</td>
</tr>
<tr>
<td>11</td>
<td>CISCO_VGC_VIRTUAL_PHONE</td>
</tr>
<tr>
<td>30</td>
<td>ANALOG_ACCESS</td>
</tr>
</tbody>
</table>
### Warning-Level Alarms

<table>
<thead>
<tr>
<th>Warning Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>DIGITAL_ACCESS</td>
</tr>
<tr>
<td>42</td>
<td>DIGITAL_ACCESS+</td>
</tr>
<tr>
<td>43</td>
<td>DIGITAL_ACCESS_WS-X6608</td>
</tr>
<tr>
<td>47</td>
<td>ANALOG_ACCESS_WS-X6624</td>
</tr>
<tr>
<td>48</td>
<td>VGC_GATEWAY</td>
</tr>
<tr>
<td>50</td>
<td>CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>51</td>
<td>CONFERENCE_BRIDGE_HARDWARE</td>
</tr>
<tr>
<td>52</td>
<td>CONFERENCE_BRIDGE_HARDWARE_HDV2</td>
</tr>
<tr>
<td>53</td>
<td>CONFERENCE_BRIDGE_HARDWARE_WS-SVC-CMM</td>
</tr>
<tr>
<td>62</td>
<td>H323_GATEWAY</td>
</tr>
<tr>
<td>70</td>
<td>MUSIC_ON_HOLD</td>
</tr>
<tr>
<td>71</td>
<td>DEVICE_PILOT</td>
</tr>
<tr>
<td>73</td>
<td>CTI_ROUTE_POINT</td>
</tr>
<tr>
<td>80</td>
<td>VOICE_MAIL_PORT</td>
</tr>
<tr>
<td>83</td>
<td>SOFTWARE_MEDIA_TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>84</td>
<td>CISCO_MEDIA_SERVER</td>
</tr>
<tr>
<td>85</td>
<td>CISCO_VIDEO_CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>90</td>
<td>ROUTE_LIST</td>
</tr>
<tr>
<td>100</td>
<td>LOAD_SIMULATOR</td>
</tr>
<tr>
<td>110</td>
<td>MEDIA_TERMINATION_POINT</td>
</tr>
<tr>
<td>111</td>
<td>MEDIA_TERMINATION_POINT_HARDWARE</td>
</tr>
<tr>
<td>112</td>
<td>MEDIA_TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>113</td>
<td>MEDIA_TERMINATION_POINT_WS-SVC-CMM</td>
</tr>
<tr>
<td>120</td>
<td>MGCP_STATION</td>
</tr>
<tr>
<td>121</td>
<td>MGCP_TRUNK</td>
</tr>
<tr>
<td>122</td>
<td>GATEKEEPER</td>
</tr>
<tr>
<td>124</td>
<td>7914_14_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>125</td>
<td>TRUNK</td>
</tr>
<tr>
<td>126</td>
<td>TONE_announcement_PLAYER</td>
</tr>
<tr>
<td>131</td>
<td>SIP_TRUNK</td>
</tr>
<tr>
<td>132</td>
<td>SIP_GATEWAY</td>
</tr>
<tr>
<td>133</td>
<td>WSM_TRUNK</td>
</tr>
<tr>
<td>134</td>
<td>REMOTE_DESTINATION_PROFILE</td>
</tr>
<tr>
<td>227</td>
<td>7915_12_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>228</td>
<td>7915_24 BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>229</td>
<td>7916_12 BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>230</td>
<td>7916_24 BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>232</td>
<td>CKEM_36 BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
</tbody>
</table>
Warning-Level Alarms

Enum Definitions

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>254</td>
<td>UNKNOWN_MGCP_GATEWAY</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>30027</td>
<td>ANALOG_PHONE</td>
</tr>
<tr>
<td>30028</td>
<td>ISDN_BRI_PHONE</td>
</tr>
<tr>
<td>30032</td>
<td>SCCP_GATEWAY_VIRTUAL_PHONE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Unknown—(SCCP only) The device failed to register for an unknown reason. If this persists, collect SDL/SDI traces with &quot;Enable SCCP Keep Alive Trace&quot; enabled and contact TAC.</td>
</tr>
<tr>
<td>2</td>
<td>NoEntryInDatabase—(MGCP only) The device is not configured in the Unified CM Administration database and auto-registration is either not supported for the device type or is not enabled. To correct this problem, configure this device in Unified CM Administration.</td>
</tr>
<tr>
<td>3</td>
<td>DatabaseConfigurationError—The device is not configured in the Unified CM Administration database and auto-registration is either not supported for the device type or is not enabled. To correct this problem, configure this device in Unified CM Administration.</td>
</tr>
<tr>
<td>4</td>
<td>DeviceNameUnresolveable—For SIP third-party devices this means that Unified CM could not determine the name of the device from the Authorization header in the REGISTER message. The device did not provide an Authorization header after Unified CM challenged with a 401 Unauthorized message. Verify that the device is configured with digest credentials and is able to respond to 401 challenges with an Authorization header. If this is a Cisco IP phone, the configuration may be out-of-sync. First, go to the Cisco Unified Reporting web page, generate a Unified CM Database Status report, and verify &quot;all servers have a good replication status&quot;. If DB replications looks good, reset the phone. If that still doesn't fix the problem, restart the TFTP and the Cisco CallManager services. For all other devices, this reason code means that DNS lookup failed. Verify the DNS server configured via the OS Administration CLI is correct and that the DNS name used by the device is configured in the DNS server.</td>
</tr>
<tr>
<td>6</td>
<td>ConnectivityError—The network connection between the device and Cisco Unified CM dropped before the device was fully registered. Possible causes include device power outage, network power outage, network configuration error, network delay, packet drops and packet corruption. It is also possible to get this error if the Cisco Unified CM node is experiencing high CPU usage. Verify the device is powered up and operating, verify network connectivity between the device and Cisco Unified CM, and verify the CPU utilization is in the safe range (this can be monitored using RTMT via CPU Pegging Alert).</td>
</tr>
<tr>
<td>7</td>
<td>InitializationError—An internal error occurred within Cisco Unified CM while processing the device registration. It is recommended to restart the Cisco CallManager service. If this occurs repeatedly, collect SDL/SDI detailed traces with &quot;Enable SIP Keep Alive (REGISTER Refresh) Trace&quot; and &quot;Enable SCCP Keep Alive Trace&quot; under Cisco CallManager services turned on and contact TAC.</td>
</tr>
<tr>
<td>10</td>
<td>AuthenticationError—The device failed either TLS or SIP digest security authentication. If the device is a SIP phone and is enabled for digest authentication (on the System &gt; Security Profile &gt; Phone Security Profile, check if &quot;Enable Digest Authentication&quot; checkbox is checked), verify the &quot;Digest Credentials&quot; in the End User config page are configured. Also, check the phone config page to see if the phone is associated with the specified end user in the Digest User drop box. If the device is a third-party SIP device, verify the digest credentials configured on the phone match the &quot;Digest Credentials&quot; configured in the End User page.</td>
</tr>
</tbody>
</table>
Warning-Level Alarms

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>InvalidX509NameInCertificate—Configured &quot;X.509 Subject Name&quot; doesn't match what's in the certificate from the device. Check the Security profile of the indicated device and verify the &quot;Device Security Mode&quot; is either &quot;Authenticated&quot; or &quot;Encrypted&quot;. Verify the &quot;X.509 Subject Name&quot; field has the right content. It should match the Subject Name in the certificate from the peer.</td>
</tr>
<tr>
<td>12</td>
<td>InvalidTLSCipher—Unsupported cipher algorithm used by the device; Cisco Unified CM only supports AES_128_SHA cipher algorithm. Recommended action is for the device to regenerate its certificate with the AES_128_SHA cipher algorithm.</td>
</tr>
<tr>
<td>14</td>
<td>MalformedRegisterMsg—(SIP only) A SIP REGISTER message could not be processed because of an illegal format. Possible causes include a missing Call-ID header, a missing AoR in the To header, and an expires value too small. Verify the REGISTER message does not suffer from any of these ills.</td>
</tr>
<tr>
<td>15</td>
<td>ProtocolMismatch—The protocol of the device (SIP or SCCP) does not match the configured protocol in Cisco Unified CM. Recommended actions: 1) Verify the device is configured with the desired protocol; 2) Verify the firmware load ID on the Device Defaults page is correct and actually exists on the TFTP server; 3) If there is a firmware load ID configured on the device page, verify it is correct and exists on the TFTP server (On Cisco Unified OS Administration page, Software Upgrades &gt; TFTP File Management, look for the file name as specified by load ID); 4) Restart the TFTP and Cisco CallManager services. Use the Cisco Unified OS Administration TFTP File Management page to verify the configured firmware loads exist.</td>
</tr>
<tr>
<td>16</td>
<td>DeviceNotActive—The device has not been activated</td>
</tr>
<tr>
<td>17</td>
<td>AuthenticatedDeviceAlreadyExists—A device with the same name is already registered. If this occurs repeatedly, collect SDL/SDI detailed traces with &quot;Enable SIP Keep Alive (REGISTER Refresh) Trace&quot; and &quot;Enable SCCP Keep Alive Trace&quot; under Cisco CallManager services turned on and contact TAC. There may be an attempt by unauthorized devices to register.</td>
</tr>
<tr>
<td>18</td>
<td>ObsoleteProtocolVersion—(SCCP only) A SCCP device registered with an obsolete protocol version. Power cycle the phone. Verify that the TFTP service is activated. Verify that the TFTP server is reachable from the device. If there is a firmware load ID configured on the Phone Config page, verify that the firmware load ID exists on the TFTP server (On Cisco Unified OS Administration page, Software Upgrades &gt; TFTP File Management, look for the file name as specified by load ID).</td>
</tr>
</tbody>
</table>

Enum Definitions for IPAddrAttributes

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown—The device has not indicated what this IPv4 address is used for.</td>
</tr>
<tr>
<td>1</td>
<td>Administrative only—The device has indicated that this IPv4 address is used for administrative communication (web interface) only.</td>
</tr>
<tr>
<td>2</td>
<td>Signal only—The device has indicated that this IPv4 address is used for control signaling only.</td>
</tr>
<tr>
<td>3</td>
<td>Administrative and signal—The device has indicated that this IPv4 address is used for administrative communication (web interface) and control signaling.</td>
</tr>
</tbody>
</table>
Warning-Level Alarms

Enum Definitions for IPV6AddrAttributes

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown—The device has not indicated what this IPv6 address is used for.</td>
</tr>
<tr>
<td>1</td>
<td>Administrative only—The device has indicated that this IPv6 address is used for administrative communication (web interface) only.</td>
</tr>
<tr>
<td>2</td>
<td>Signal only—The device has indicated that this IPv6 address is used for control signaling only.</td>
</tr>
<tr>
<td>3</td>
<td>Administrative and signal—The device has indicated that this IPv6 address is used for administrative communication (web interface) and control signaling.</td>
</tr>
</tbody>
</table>

Recommended Action

In the Cisco Unified Reporting tool, check the Active Services section of the Unified CM Cluster Overview report to confirm that any failover/fallback scenarios have completed. Confirm that auto-registration is enabled if the phone attempting to connect is set to auto-register, or locate the phone that is attempting to auto-register if auto-registration has been intentionally disabled. Check the device indicated in this alarm and confirm that the device registration details in Cisco Unified CM Administration are accurate. Also, refer to the reason code definitions for recommended actions. No action is required if this event was issued as a result of a normal device rehome.

DeviceUnregistered

A device that has previously registered with Cisco CallManager has unregistered. In cases of normal unregistration with reasoncode 'CallManagerReset', 'CallManagerRestart', or 'DeviceInitiatedReset', the severity of this alarm is lowered to INFORMATIONAL. A device can unregister for many reasons, both intentional, such as manually resetting the device after a configuration change, or unintentional, such as loss of network connectivity. Other causes for this alarm could include a phone being registered to a secondary node and then the primary node come back online, causing the phone to rehome to the primary Unified CM node or lack of a KeepAlive being returned from the Unified CM node to which this device was registered. Unregistration also occurs if Unified CM receives a duplicate registration request for this same device.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following information is updated:</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for DeviceType</td>
</tr>
<tr>
<td></td>
<td>– Enum Definition</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for IPAddrAttributes</td>
</tr>
<tr>
<td></td>
<td>– Enum Definitions for IPV6AddrAttributes</td>
</tr>
<tr>
<td>7.1</td>
<td>Parameters added: IPV6Address,IPAddrAttributes, and IPV6AddrAttributes</td>
</tr>
</tbody>
</table>
Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning

Parameters
Device name. [String]
Device MAC address [Optional]. [String]
Device IP address [Optional]. [String]
Protocol. [String]
Device type. [Optional] [Enum]
Device description [Optional]. [String]
Reason Code [Optional]. [Enum]
IPV6Address [Optional]. [String]
IPAddressAttributes [Optional]. [Enum]
IPV6AddressAttributes [Optional]. [Enum]

Enum Definitions for DeviceType

<table>
<thead>
<tr>
<th>Code</th>
<th>Device Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>CISCO_VGC_PHONE</td>
</tr>
<tr>
<td>11</td>
<td>CISCO_VGC_VIRTUAL_PHONE</td>
</tr>
<tr>
<td>30</td>
<td>ANALOG_ACCESS</td>
</tr>
<tr>
<td>40</td>
<td>DIGITAL_ACCESS</td>
</tr>
<tr>
<td>42</td>
<td>DIGITAL_ACCESS+</td>
</tr>
<tr>
<td>43</td>
<td>DIGITAL_ACCESS_WS-X6608</td>
</tr>
<tr>
<td>47</td>
<td>ANALOG_ACCESS_WS-X6624</td>
</tr>
<tr>
<td>48</td>
<td>VGC_GATEWAY</td>
</tr>
<tr>
<td>50</td>
<td>CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>51</td>
<td>CONFERENCE_BRIDGE_HARDWARE</td>
</tr>
<tr>
<td>52</td>
<td>CONFERENCE_BRIDGE_HARDWARE_HDV2</td>
</tr>
<tr>
<td>53</td>
<td>CONFERENCE_BRIDGE_HARDWARE_WS-SVC-CMM</td>
</tr>
<tr>
<td>62</td>
<td>H323_GATEWAY</td>
</tr>
<tr>
<td>70</td>
<td>MUSIC_ON_HOLD</td>
</tr>
<tr>
<td>71</td>
<td>DEVICE_PILOT</td>
</tr>
<tr>
<td>73</td>
<td>CTI_ROUTE_POINT</td>
</tr>
<tr>
<td>80</td>
<td>VOICE_MAIL_PORT</td>
</tr>
</tbody>
</table>
Warning-Level Alarms

<table>
<thead>
<tr>
<th>Code</th>
<th>Enum Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>83</td>
<td>SOFTWARE_MEDIA_TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>84</td>
<td>CISCO_MEDIA_SERVER</td>
</tr>
<tr>
<td>85</td>
<td>CISCO_VIDEO_CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>90</td>
<td>ROUTE_LIST</td>
</tr>
<tr>
<td>100</td>
<td>LOAD_SIMULATOR</td>
</tr>
<tr>
<td>110</td>
<td>MEDIA_TERMINATION_POINT</td>
</tr>
<tr>
<td>111</td>
<td>MEDIA TERMINATION_POINT_HARDWARE</td>
</tr>
<tr>
<td>112</td>
<td>MEDIA TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>113</td>
<td>MEDIA TERMINATION_POINT_WS-SVC-CMM</td>
</tr>
<tr>
<td>120</td>
<td>MGCP_STATION</td>
</tr>
<tr>
<td>121</td>
<td>MGCP_TRUNK</td>
</tr>
<tr>
<td>122</td>
<td>GATEKEEPER</td>
</tr>
<tr>
<td>124</td>
<td>7914_14_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>125</td>
<td>TRUNK</td>
</tr>
<tr>
<td>126</td>
<td>TONE_ANNOUNCEMENT_PLAYER</td>
</tr>
<tr>
<td>131</td>
<td>SIP_TRUNK</td>
</tr>
<tr>
<td>132</td>
<td>SIP_GATEWAY</td>
</tr>
<tr>
<td>133</td>
<td>WSM_TRUNK</td>
</tr>
<tr>
<td>134</td>
<td>REMOTE_DESTINATION_PROFILE</td>
</tr>
<tr>
<td>227</td>
<td>7915_12_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>228</td>
<td>7915_24_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>229</td>
<td>7916_12_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>230</td>
<td>7916_24_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>232</td>
<td>CKEM_36_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>254</td>
<td>UNKNOWN_MGCP_GATEWAY</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>30027</td>
<td>ANALOG_PHONE</td>
</tr>
<tr>
<td>30028</td>
<td>ISDN_BRI_PHONE</td>
</tr>
<tr>
<td>30032</td>
<td>SCCP_GATEWAY_VIRTUAL_PHONE</td>
</tr>
</tbody>
</table>

**Enum Definition**

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Unknown - The device has unregistered for an unknown reason. If the device does not reregister within 5 minutes, verify it is powered-up and verify network connectivity between the device and Cisco Unified CM.</td>
</tr>
<tr>
<td></td>
<td>Description</td>
</tr>
<tr>
<td>---</td>
<td>-------------</td>
</tr>
<tr>
<td>6</td>
<td>ConnectivityError - Network communication between the device and Cisco Unified CM has been interrupted. Possible causes include device power outage, network power outage, network configuration error, network delay, packet drops and packet corruption. It is also possible to get this error if the Cisco Unified CM node is experiencing high CPU usage. Verify the device is powered up and operating, verify network connectivity between the device and Cisco Unified CM, and verify the CPU utilization is in the safe range (this can be monitored using RTMT via CPU Pegging Alert).</td>
</tr>
<tr>
<td>8</td>
<td>DeviceInitiatedReset - The device has initiated a reset, possibly due to a power cycle or internal error. No action required; the device will reregister automatically.</td>
</tr>
<tr>
<td>9</td>
<td>CallManagerReset - A device reset was initiated from Cisco Unified CM Administration, either due to an explicit command from an administrator, or due to internal errors encountered. No action necessary, the device will reregister automatically.</td>
</tr>
<tr>
<td>10</td>
<td>DeviceUnregistered - The device has explicitly unregistered. Possible causes include a change in the IP address or port of the device. No action is necessary, the device will reregister automatically.</td>
</tr>
<tr>
<td>11</td>
<td>MalformedRegisterMsg - (SIP only) A SIP REGISTER message could not be processed because of an illegal format. Possible causes include a missing Call-ID header, a missing AoR in the To header, and an expires value too small. Verify the REGISTER message does not suffer from any of these ills.</td>
</tr>
<tr>
<td>12</td>
<td>SCCPDeviceThrottling - (SCCP only) The indicated SCCP device exceeded the maximum number of events allowed per-SCCP device. Events can be phone calls, KeepAlive messages, or excessive SCCP or non-SCCP messages. The maximum number of allowed events is controlled by the Cisco CallManager service parameter, Max Events Allowed. When an individual device exceeds the number configured in that service parameter, Unified CM closes the TCP connection to the device; automatic reregistration generally follows. This action is an attempt to stop malicious attacks on Unified CM or to ward off excessive CPU usage.</td>
</tr>
<tr>
<td>13</td>
<td>KeepAliveTimeout - A keepalive message was not received. Possible causes include device power outage, network power outage, network configuration error, network delay, packet drops and packet corruption. It is also possible to get this error if the Cisco Unified CM node is experiencing high CPU usage. Verify the device is powered up and operating, verify network connectivity between the device and Cisco Unified CM, and verify the CPU utilization is in the safe range (this can be monitored using RTMT via CPU Pegging Alert).</td>
</tr>
<tr>
<td>14</td>
<td>ConfigurationMismatch (SIP only) The configuration on the device does not match the configuration in Cisco Unified CM. This can be caused by database replication errors or other internal Cisco Unified CM communication errors. First go to the Cisco Unified Reporting web page, generate a Unified CM Database Status report, and verify &quot;all servers have a good replication status&quot;. If this device continues to unregister with this reason code, go to the CCMAdmin Device web page for the device and click Save. This allows a change notify to be generated to the Unified CM and TFTP services and rebuild a new config file. If the problem still persists, restart the TFTP service and Cisco Unified CM service.</td>
</tr>
<tr>
<td>15</td>
<td>CallManagerRestart - A device restart was initiated from Cisco Unified CM, either due to an explicit command from an administrator, or due to a configuration change such as adding, deleting or changing a DN associated with the device. No action necessary, the device will reregister automatically.</td>
</tr>
</tbody>
</table>
Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Warning-Level Alarms

Recommendation Action

Actions to take vary depending on the reason specified for the device unregistration. If the reason is ConfigurationMismatch, go to the Device Configuration page in Cisco Unified CM Administration, make a change to the Description field for this device, click Save, then reset the device. In the case of a network connectivity or loss of KeepAlives problem, use network diagnostic tools and the Cisco Unified CM Reporting tool to fix any reported network or Unified CM system errors. In the case of a device rehoming to the primary Unified CM node, watch for a successful registration of the device on the primary node. In the case of a duplicate registration request, it may be a non-malicious occurrence due to timing of a device registering and unregistering; if duplicate registration requests continue or if the same device has different IP addresses, confirm the IP address on the physical device itself by checking the settings on the device (settings button). If unregistration of this device was expected, no action is required. Also, refer to the reason code descriptions for recommended actions.
DigitAnalysisTimeoutAwaitingResponse

Cisco Unified Communications Manager sent a routing request to the policy decision point but the request timed out without a response.

Cisco Unified Communications Manager (Unified CM) was unable to complete the routing request before timing out. This time out could occur due to low system resources, high CPU usage, or a high volume of call activities on this Unified CM node. Unified CM applies the Call Treatment on Failure that is configured for the External Call Control Profile associated with this call.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
WARNING

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Translation Pattern Triggering Point(String)
Policy Decision Point(String)

Recommended Action
• Check the External Call Control object in Real-Time Monitoring Tool (RTMT) to see whether the ExternalCallControlEnabledCallAttempted counter is spiking. If so, this indicates an unusually high number of calls at this time which could result in reduced system resources.
• Check the QueueSignalsPresent2-Normal for persistent long high signal queue. If the long signal queue exists, check whether the Code Yellow alarm has already issued and check the system CPU and memory usage for this Unified CM node.
• Follow the recommended actions for Code Yellow alarm if the Code Yellow alarm has fired. For high CPU usage, use RTMT to determine which areas may be contributing to the high CPU usage. If this alarm persists, collect system performance data (such as the percentage of Memory, Page and VM usage, partition read and write bytes per second, the percentage of CPU usages of all the processes, and the processor IOWait percentage) and contact Cisco Technical Assistance Center (TAC).

DirSyncNoSchedulesFound

No schedules found in DB for directory synchronization. No automatic LDAP directory synchronization possible.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS
Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Warning (4)

Parameters
ScheduleTableName [String]

Recommended Action
Check the DirSync configuration

DirSyncScheduledTaskTimeoutOccurred

Timeout occurred for directory synchronization task.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Warning (4)

Parameters
SchedulerID [String] TaskID [String]

Recommended Action
Check the DirSync configuration.

DRFComponentDeRegistered

DRF successfully de-registered the requested component.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFComponentDeRegistered. Routing List elements added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF
Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Warning (4)

Routing List
Event Log
Sys Log

Parameters
Reason(String)

Recommended Action
Ensure that the component that was de-registered is not needed for further backup/restore operation.

**DRFDeRegistrationFailure**

DRF de-registration request for a component failed.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFDeRegistrationFailure. Routing List elements added.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Warning (4)

Routing List
Event Log
Sys Log

Parameters
Reason(String)

Recommended Action
Check the DRF logs and contact support if needed.
**DRFDeRegisteredServer**

DRF automatically de-registered all the components for a server. This server might have got disconnected from CCM cluster.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFDeRegisteredServer. Routing List elements added.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DRF

**Severity**

Warning (4)

**Routing List**

Event Log
Sys Log

**Parameters**

Reason(String)

**Recommended Action**

None

**DRFNoBackupTaken**

A valid backup of the current system was not found after an Upgrade, Migration, or Fresh Install.

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DRF

**Severity**

WARNING

**Routing List**

Event Log
Sys Log

**Parameter(s)**

Reason(String)
Warning-Level Alarms

Recommended Action
It is recommended to perform a Backup using the Disaster Recovery System.

DRFSchedulerDisabled

DRF Scheduler is disabled because no configured features available for backup.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
</table>
| 8.0(1)                              | Name changed from CiscoDRFSchedulerDisabled.  
|                                     | Routing List elements added. |

Facility/Sub-Facility

CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

Cisco Unified Serviceability Alarm Definition Catalog

System/DRF

Severity

Warning (4)

Routing List

Event Log
Sys Log

Parameters

Reason(String)

Recommended Action
Ensure at least one feature is configured for the scheduled backup to run.

EMCCFailedInRemoteCluster

There was an EMCC login failure at a remote Unified CM. EMCC login could fail due to the following reasons:

- User does not exist in any of the configured remote cluster.
- User is not enabled for EMCC.
- No free EMCC base device.
- EMCC access was prevented by remote cluster.
- Untrusted certificate received from the remote end while trying to establish a connection.

Reason Codes:

- 38—EMCC or PSTN is not activated in InterClusterServiceProfile page
• 31—User is not enabled for EMCC
• 39—Default and Backup TFTP Service is not configured

**Cisco Unified Serviceability Alarm Definition Catalog**

**System/EMAlarmCatalog**

**Severity**

Warning(4)

**Routing List**

Sys Log
Event Log
Alert Manager

**Parameters**

Device Name(String)
Login Date/Time(String)
Login UserID(String)
Reason(String)

**Recommended Action**

Perform the following steps:

---

**Step 1** Ensure that the user is a valid EMCC user and that user home cluster is added as a EMCC remote cluster (From Unified CM Administration window, go to System > EMCC > Remote Cluster > Add New).

**Step 2** Contact remote site administrator to enable user for EMCC (From Unified CM Administration window, go to User Management > End User > Select User > Enable Extension Mobility Cross Cluster checkbox).

**Step 3** Contact remote site administrator for adding or freeing EMCC Base Devices (From Unified CM Administration window, go to Bulk Administration > EMCC > Insert/Update EMCC).

**Step 4** Contact remote site administrator to validate the remote cluster setting for this cluster.

**Step 5** Ensure that a bundle of all Tomcat certificates (PKCS12) got imported into the local tomcat-trust keystore (From the OS Administration window, go to Security > Certificate Management).

---

**ErrorParsingResponseFromPDP**

Cisco Unified Communications Manager failed to parse one or multiple optional elements or attributes in the call routing response from the policy decision point.

A routing response was received from the policy decision point (PDP) but Cisco Unified Communications Manager (Unified CM) failed to parse the optional elements in the response. Optional elements may include modified calling numbers or called numbers, call reject or call diversion reasons, and so on. The cause may be a syntax error or missing attributes in the call routing response.

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CallManager
Warning-Level Alarms

Severity
WARNING

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Policy Decision Point(String)
Called Party Number(String)
Calling Party Number(String)
Calling User Id(String)
Request XML Data(String)

Recommended Action
Check if call routing response from the policy decision point complies with the guidelines specified for external call control in the Cisco Unified Communications Manager documentation. Check if any optional elements included as the policy obligations in the call routing response are correctly entered according to the external call control documentation, including any applicable API documentation.

FailedToFulfillDirectiveFromPDP
Cisco Unified Communications Manager cannot fulfill the call routing directive returned by the PDP. The failure can occur because of the following conditions:

- Call was cleared by a CTI application before Cisco Unified Communications Manager was able to route it to the location defined by the PDP.
- Call that was allowed by a policy server was redirected by the CTI application to a destination.
- Annunciator ID was misconfigured in the PDP.
- Unified CM attempted to invoke a media resource such as Annunciator but no resources were available.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning(4)

Routing List
SDL
SDI
Sys Log
Event Log
Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

**Warning-Level Alarms**

**Parameters**
- Policy Decision Point(String)
- Reason, Unified CM failed to fulfill the directive(String)
- Called Party Number(String)
- Calling Party Number(String)
- Calling User Id(String)

**Recommended Action**
- In many cases, the cause for a failure occurs because of the intervention by a CTI application which scoops up the call before Unified CM is able to fulfill the routing directive in the PDP. Examine the CTI application to ensure that the call is in alerting or connected state before the CTI begins to interact with it.
- If the failure is caused by a problem with the annunciator ID, ensure the ID has been accurately configured in the PDP and that it exists in Unified CM Administration.
- If the failure was caused by a lack of media resources, try increasing the Annunciator Call Count service parameter in the Cisco IP Voice Media Streaming App service.

**H323Stopped**

Cisco CallManager is not ready to handle calls for the indicated H323 device.

Cisco Unified Communications Manager (Unified CM) is not ready to handle calls for the indicated H.323 device. This could be due to Unified CM being unable to resolve the gateway name to IP address. For trunks, this alarm should only occur when a system administrator has made a configuration change such as resetting the H.323 trunk. For H.323 clients, this alarm occurrence is normal on lower-priority Unified CM nodes when a high-priority Unified CM node starts.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Following information updated:</td>
</tr>
<tr>
<td></td>
<td>• Parameters</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for DeviceType</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
- CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**
- CallManager/CallManager

**Severity**
- Warning (4)
### Warning-Level Alarms

**Parameters**


**Enum Definitions for DeviceType**

<table>
<thead>
<tr>
<th>Code</th>
<th>Device Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>61</td>
<td>H323_PHONE</td>
</tr>
<tr>
<td>62</td>
<td>H323_GATEWAY</td>
</tr>
<tr>
<td>122</td>
<td>GATEKEEPER</td>
</tr>
<tr>
<td>125</td>
<td>TRUNK</td>
</tr>
</tbody>
</table>

**Recommended Action**

If the service was stopped intentionally, no action is required. Check the domain name system (DNS) configuration for any errors in the gateway name or IP address and correct.

### InvalidSubscription

A message has been received from an IME server that contains a subscription identifier that is not handled by this node.

Each node that communicates with a IME server saves a subscription identifier associated with each IME client instance. A IME server has sent a message with a subscription identifier that does not match any of the previously sent subscription identifiers.

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CallManager

**Severity**

WARNING

**Recommended Action**

This may be a race condition if the IME client instance has been recently added or deleted. If this error continues, there may be a synchronization issue between this node and the IME server sending this message.

**Routing List**

SDL
SDI
Sys Log
Event Log

**Parameter(s)**

Subscription Identifier(UInt)
IME Server(String)
InvalidQBEMessage

QBE PDU from application is invalid.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

Severity
WARNING

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
CTI Connection type(String)

Recommended Action
This alarm indicates that TSP/JTAPI has reported a QBE PDU that cannot be recognized by CTIManager. Contact the support organization for the affected application, install the JTAPI or TSP plugin and restart the application. JTAPI/TSP plugins are available from the Find and List Plugins window in Cisco Unified CM Administration (Application > Plugins).

IPMAManagerLogout

IPMA Manager Logged out.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Warning (4)

Parameters
Servlet Name [String] Reason [String]

Recommended Action
To relogin the user, click update in the CCMAadmin IPMA Service configuration page for this user.
Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Warning-Level Alarms

IPMAStopped

IPMA Application stopped and unloaded from Tomcat.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Warning (4)

Parameters
Servlet Name [String] Reason [String]

Recommended Action
Check if Tomcat service is up.

kANNAudioFileMissing

Announcement file not found. The annunciator was unable to access an announcement audio file. This may be caused by not uploading a custom announcement to each server in the cluster or a locale has not been installed on the server.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
WARNING

Routing List
SDI
Event Log
Sys Log

Parameter(s)
Missing filename(String)

Recommended Action
Upload the custom announcement to the server or install the missing locale package.
**kANNAudioUndefinedAnnID**

Requested announcement not found. This may be caused by using an incorrect announcement identifier for a custom announcement. Use the Cisco Unified CM Admin to view a list of custom announcement identifiers and verify the correct one is being used.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Parameter list removed.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

**Severity**

Warning

**Recommended Action**

Add the announcement.

---

**kANNAudioUndefinedLocale**

Unknown ANN locale. The requested Locale for an announcement is not installed. For network locale you use the platform CLI interface to run (run sql select * from typecountry where enum = #), #=locale. This will tell you what country locale is being requested.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Parameter list is updated.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms
Severity
Warning

Parameters
Locale Type [String]

Recommended Action
Install the locale package or check device settings for an incorrect locale value.

kANNDeviceStartingDefaults

The ANN device configuration was not found. A service parameter for Cisco IP Voice Media Streaming App service related to the ANN device configuration was not found. The system will start with the given default setting.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
</table>
| 8.0(1)                              | • Severity changed from Informational to Warning.  
|                                     | • Parameter list added. |

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameter(s)
Parameter Name [String]
Value Used [String]

Recommended Action
Review the service parameter settings and configure the ANN device settings properly using the Cisco Unified CM Administration.

kCFBDeviceStartingDefaults

CFB device configuration not found. A service parameter for Cisco IP Voice Media Streaming App service related to the CFB device configuration was not found. The system will use the given default setting.
### Facility/Sub-Facility

**CCM_MEDIA_STREAMING_APP-IPVMS**

### Cisco Unified Serviceability Alarm Definition Catalog

**CallManager/IpVms**

#### Severity

**Warning**

#### Parameter(s)

- **Parameter Name(String)**
- **Value Used(String)**

#### Recommended Action

Review the service parameter settings and configure the CFB device settings properly using the Cisco Unified CM Administration.

### kChangeNotifyServiceCreationFailed

Database change notification subsystem not starting. The background process to activate database changes has failed to start. Database changes affecting the Cisco IP Voice Media Streaming App service will not automatically take effect.

#### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
<tr>
<td></td>
<td>• Severity changed from Informational to Warning.</td>
</tr>
<tr>
<td></td>
<td>• New parameters added:</td>
</tr>
<tr>
<td></td>
<td>– Parameter Name(String)</td>
</tr>
<tr>
<td></td>
<td>– Value Used(String)</td>
</tr>
</tbody>
</table>
Warning-Level Alarms

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameter(s)
OS Error Code(Int)
OS Error Description(String)

Recommended Action
Restart the Cisco IP Voice Media Streaming App service to get the DB notification reenabled.

kChangeNotifyServiceGetEventFailed

Invalid notification event returned by database change notification. The change notification subsystem returned an invalid notification event. The Cisco IP Voice Media Streaming App service will terminate. The SW media devices (ANN, CFB, MOH, MTP) will be temporarily out of service and calls in progress may be dropped.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
</tbody>
</table>

7.0(1) Obsoleted.
8.0(1) This alarm is available in 8.0(1).
• Severity changed from Error to Warning.
• Following parameters added:
  – OS Error Code(Int)
  – OS Error Description(String)
### Warning-Level Alarms

#### Facility/Sub-Facility

**CCM_MEDIA_STREAMING_APP-IPVMS**

#### Cisco Unified Serviceability Alarm Definition Catalog

**CallManager/IpVms**

#### Severity

**Warning**

#### Parameter(s)

- OS Error Code(Int)
- OS Error Description(String)

#### Recommended Action

Check the current status of the Cisco IP Voice Media Streaming App service and monitor for repeated occurrences.

### kChangeNotifyServiceRestartFailed

Database change notification restart failure. The change notification subsystem failed to restart.

#### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1)</td>
</tr>
<tr>
<td></td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>• OS Error Code(Int)</td>
</tr>
<tr>
<td></td>
<td>• OS Error Description(String)</td>
</tr>
</tbody>
</table>
Warning-Level Alarms

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameter(s)
OS Error Code(Int)
OS Error Description(String)

Recommended Action
This service has change notification disabled, it may be reenabled at a later time or restart Cisco IP Voice Media Streaming App service to reenable immediately.

kDeviceDriverError

IP voice media streaming device driver error. The IP voice media streaming device driver returned an error. This may indicate a significant media error or resource shortage.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
<tr>
<td></td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameters
Error [String]

Recommended Action
Restarting the Cisco IP Voice Media Streaming App service or possibly restarting the server may resolve the error condition.
kDeviceMgrCreateFailed

Device connection manager failed to start. The device controller was unable to start a connection to control device registration with CallManager. This is possibly due to lack of memory.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/IpVms

Severity

Warning

Parameters

Device Name [String] Server Name [String]

Recommended Action

Restart the Cisco IP Voice Media Streaming App service or restart the Cisco Unified CM server.

kDeviceMgrOpenReceiveFailedOutOfStreams

Open receive failure. The open receive channel failed. This may indicate a mismatch of media resources between Cisco Unified Call Manager and the Cisco IP Voice Media Streaming App service.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Severity changed from Error to warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_MEDIA_STREAMING_APP-IPVMS
Warning-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameters
Trace Name [String]

Recommended Action
Check the performance monitor counters for resource availability on Cisco Unified CM and on Cisco IP Voice Media Streaming App. Also, you might run the Platform CLI command "Show Media Streams" to identify possible media connection resource leaks. Possibly reset the media device or restart Cisco IP Voice Media Streaming App or restart the Cisco Unified CM server.

kDeviceMgrRegisterKeepAliveResponseError

Cisco Unified Communications Manager not responding. The specified Cisco Unified Communications Manager is not responding to the keepalive messages. The connection with Cisco Unified CM is being terminated and the media device will reregister with another Cisco Unified Call Manager if a secondary is configured. Otherwise, the media device will be unavailable until the device is able to reregister with Cisco Unified CM.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameters
Trace Name [String]

Recommended Action
Cisco Unified Communications Manager may have gone down or is unable to respond. Check status of Cisco Unified CM. The media device should automatically reregister.
**kDeviceMgrRegisterWithCallManagerError**

Connection error with Cisco Unified Communications Manager. The media device was registered with the specified Cisco Unified Communications Manager and received a socket error or disconnect. This may occur normally when Cisco Unified Communications Manager is stopped.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

**Severity**

Warning

**Parameters**

Trace Name [String]

**Recommended Action**

No action is required; The media device will reregister.

**kDeviceMgrSocketDrvNotifyEvtCreateFailed**

This alarm get generated when creating a signaling event for communication with the media streaming kernel driver. It can be caused by memory or system resource shortages/

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Added Routing List elements. Changed severity level to Warning from Error.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms
Severity
Warning (4)

Routing List
SDI
Event Log
Sys Log

Parameters
Device Name [String]
Trace Name [String]
OS Error Description [String]

Recommended Action
Restart the Cisco IP Voice Media Streaming App service or restart the Cisco Unified Communications Manager server.

**kDeviceMgrSocketNotifyEventCreateFailed**

Creation socket event failure. An error was reported when creating a notification event for a socket interface. This may be due to a resource shortage. The media device will remain unavailable.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameters
Device Name [String] Trace Name [String] OS Error Description [String]
Recommended Action
Restart the Cisco IP Voice Media Streaming App service and monitor for reoccurrence or restart the Cisco Unified CM server.

kDeviceMgrStartTransmissionOutOfStreams

Start transmission failure. An error was encountered while starting an RTP transmission audio stream. This may indicate a mismatch of resources between Cisco Unified Communications Manager and Cisco IP Voice Media Streaming App service.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameters
Trace Name [String]

Recommended Action
Check the performance counters for the media resources on Cisco Unified CM and Cisco IP Voice Media Streaming App to determine if there is a resource leak. You should also use the platform CLI command "Show Media Streams" to check for orphaned media RTP connections.

kDeviceMgrThreadxFailed

Creation of thread failure. An error was reported when starting a process for the specified media device. This may be due to a system resource shortage.
Warning-Level Alarms

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameters

Recommended Action
Restart the Cisco IP Voice Media Streaming App service or restart the Cisco Unified CM server to recover from this error.

kFixedInputCodecStreamFailed

Fixed input codec stream initialization failure. Initialization of sound card codec source transcoding process failed. The fixed audio source will not play possibly due to memory or resource shortage.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
</table>
| 8.0(1)                              | This alarm is available in 8.0(1).
  | • Severity changed from Error to Warning. |
  | • Following parameters added: |
  |   – OS Error Code[Int] |
  |   – OS Error Description [String] |

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameters

Recommended Action
Restart the Cisco IP Voice Media Streaming App service or restart the Cisco Unified CM server to recover from this error.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters removed:</td>
</tr>
<tr>
<td></td>
<td>– Audio Source ID [ULong]</td>
</tr>
<tr>
<td></td>
<td>– System error code [ULong]</td>
</tr>
</tbody>
</table>
Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS
Cisco Unified Serviceability Alarm Definition Catalog
CallManager/Ipm
Severity
Warning
Parameters
Codec Type [String]
Recommended Action
Reset MOH device, or restart Cisco IO Voice Media Streaming App service, or restart server.

**kFixedInputCreateControlFailed**

Fixed stream control create failure. The audio stream control subsystem for the Fixed MOH audio source failed to start. Audio from the MOH Fixed audio source will not be provided for streaming out. This may be due to resource shortage such as memory or availability of the Fixed MOH audio source device.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
<tr>
<td></td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Audio Source ID [ULong] parameter is removed.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS
Cisco Unified Serviceability Alarm Definition Catalog
CallManager/Ipm
Severity
Warning
Parameters
Codec Type [String]
Recommended Action
Reset MOH device, if failure continues restart the server. Monitor for errors in trace files and system log.
**kFixedInputCreateSoundCardFailed**

Fixed stream sound card interface create failure. An error was encountered when starting the interface to access the sound card for providing MOH fixed audio. The audio source will not play possibly due to shortage of memory.

**History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Action</th>
</tr>
</thead>
</table>
| 8.0(1)  | • Severity changed from Error to Warning.  
          • Audio Source ID [ULong] parameter is removed. |

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

**Severity**

Warning

**Parameters**

Codec Type [String]

**Recommended Action**

Reset MOH device, or restart the Cisco IP Voice Media Streaming App service, or restart the server. Check the system log and possibly the traces for Cisco IP Voice Media Streaming App service.

**kFixedInputInitSoundCardFailed**

Fixed stream sound card interface initialization failure. Initialization of sound card failed. Fixed audio source will not play possibly due to missing or unconfigured USB sound device.

**History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Action</th>
</tr>
</thead>
</table>
| 8.0(1)  | • Severity changed from Error to Warning.  
          • Following parameters are removed:  
          – Audio Source ID [ULong]  
          – System error code [ULong] |
### Facility/Sub-Facility

CCM_MEDIA_STREAMING_APP-IPVMS

### Cisco Unified Serviceability Alarm Definition Catalog

CallManager/IpVms

### Severity

Warning

### Parameters

Error text [String] Device name [String]

### Recommended Action

Check that the USB sound is installed. Reset MOH device, or restart Cisco IP Voice Media Streaming App service, or restart the server. The system log and traces from Cisco IP Voice Media Streaming App may contain additional information.

#### kFixedInputTranscoderFailed

Fixed input audio stream transcoder failure. An error was encountered while transcoding audio from the sound card. The audio source will not play possibly due an error accessing the sound card.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters are removed:</td>
</tr>
<tr>
<td></td>
<td>– Audio Source ID [ULong]</td>
</tr>
<tr>
<td></td>
<td>– System error code [ULong]</td>
</tr>
</tbody>
</table>

### Facility/Sub-Facility

CCM_MEDIA_STREAMING_APP-IPVMS

### Cisco Unified Serviceability Alarm Definition Catalog

CallManager/IpVms

### Severity

Warning

### Parameters

Error text [String]
Warning-Level Alarms

Recommended Action
Check that the USB sound device is properly installed. Unplug the USB sound device and replug back into the USB connector. Reset MOH device, restart Cisco IP Voice Media Streaming App service, or restart the server.

kGetFileNameFailed
Get audio source file name failure. The Music-on-Hold audio source is not assigned to an audio file.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
<tr>
<td></td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Audio Source ID [ULong] parameter is removed.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameters
Codec Type [String]

Recommended Action
Assign the audio source to an audio file or change the value of the MOH audio source to a value that has been configured.

kIPVMSMgrEventCreationFailed
Creation of required signaling event failed. An error was encountered when creating a signaling event component. This may be due to a resource shortage. The Cisco IP Voice Media Streaming App service will terminate.
**Facility/Sub-Facility**
CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/IpVms

**Severity**
Warning

**Parameters**
OS Error Description(String)

**Recommended Action**
Check the trace files for more information. The service should automatically be restarted. If this error continues to reoccur the server may need to be restarted.

### kIPVMSMgrThreadxFailed

Creation of the IPVMSMgr thread failed. An error was encountered while starting a process thread. The Cisco IP Voice Media Streaming App service will terminate. The software media devices (ANN, CFB, MOH, MTP) will be unavailable while the service is stopped.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/IpVms
### Severity
Warning

### Parameters
OS Error Description(String)

### Recommended Action
Monitor the status of the Cisco IP Voice Media Streaming App service. It should automatically be restarted. If the error reoccurs, restart the server.

#### kIpVmsMgrThreadWaitFailed

Error while waiting for asynchronous notifications of events. An error was reported while the primary control process for Cisco IP Voice Media Streaming App was waiting on asynchronous events to be signaled. The service will terminate and should automatically be restarted. This will cause a temporary loss of availability for the software media devices (ANN, CFB, MOH, MTP).

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

### Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

### Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

### Severity
Warning

### Recommended Action
Monitor the service and status of the software media devices. The service should automatically restart. If the problem continues, review the trace files for additional information. A server restart may be required if this repeats.

#### kMOHMgrCreateFailed

Error starting MOH Audio source subcomponent. A error was encountered by the Music-on-Hold device while starting the sub-component that provides audio from files or sound card. This may be due to shortage of resources (memory).
Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameter(s)
OS Error Description(String)

Recommended Action
Restart the Cisco IP Voice Media Streaming App service or restart the server.

kMOHMgrExitEventCreationFailed

Creation of MOH manager exit event failure. An error was encountered when allocating a signaling event. This may be caused by a resource shortage.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms
Warning-Level Alarms

Severity
Warning

Parameters
OS Error Description(String)

Recommended Action
Restart the Cisco IP Voice Media Streaming App service or restart the server.

kMOHMgrThreadxFailed

Starting of MOH audio manager failed. An error was encountered when starting the Music-on-Hold audio manager subcomponent. Music-on-Hold audio services will not be available.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
<tr>
<td></td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• OS Error Description(String) parameter is added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning

Parameters
OS Error Description(String)

Recommended Action
Restart the Cisco IP Voice Media Streaming App service.
**kMTPDeviceRecordNotFound**

MTP device record not found. A device record for the software media termination point device was not found in the database. This is normally automatically added to the database when a server is added to the database. The software MTP device will be disabled.

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

**Severity**

Warning

**Recommended Action**

If MTP functionality is required, you will need to delete the server and readd the server back to the database using CCMAdmin. WARNING: This may require many additional configuration settings to be reapplied such as CallManager Groups, Media Resource groups and more.

**kRequestedCFBStreamsFailed**

CFB requested streams failure. The resources for the number of requested full-duplex streams was not available.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

**Severity**

WARNING
**Warning-Level Alarms**

**Recommended Action**
Verify the Cisco IP Voice Media Streaming App service parameter for number of CFB calls. Restart the server to reset the stream resources.

**kRequestedMOHStreamsFailed**

MOH requested streams failure. The resources for the number of requested streams was not available.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

**Severity**

WARNING

**Recommended Action**
Verify the number of calls configuration setting for Music-on-Hold device. Restart the server to reset the resources.

**kRequestedMTPStreamsFailed**

MTP requested streams failure. The resources for the number of requested full-duplex Media Termination Point streams was not available.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

**Severity**

WARNING

**Recommended Action**
Verify the Cisco IP Voice Media Streaming App service parameter setting for number of MTP calls is correct. Restart the server to reset the available resources.
**LogCollectionJobLimitExceeded**

The number of Log Collection Jobs have exceeded the allowed limit. The number of concurrent trace collection from the server has exceeded the allowed limit of trace collection. The allowed limit is defined in the documentation for Trace and Log Central, however this limit can not be changed by sysadmin.

<table>
<thead>
<tr>
<th>History</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Unified Communications Release</td>
</tr>
<tr>
<td>8.0(1)</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_TCT-LPM_TCT

**Cisco Unified Serviceability Alarm Definition Catalog**

System/LpmTct

**Severity**

Warning

**Parameters**

JobType [String]

**Recommended Action**

Cancel one or more of the currently running queries and try again to configure the trace collection.

**LDAPServerUnreachable**

Authentication server could not be reached.

<table>
<thead>
<tr>
<th>History</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Unified Communications Release</td>
</tr>
<tr>
<td>8.5(1)</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**

System/IMS

**Severity**

Warning

**Parameters**

Message(String)
Warning-Level Alarms

Recommended Action
Check reachability to Authentication Server.

LogPartitionLowWaterMarkExceeded

The percentage of used disk space in the log partition has exceeded the configured low water mark.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TCT-LPMTCT

Cisco Unified Serviceability Alarm Definition Catalog
System/LpmTct

Severity
Warning

Parameters
UsedDiskSpace [String] MessageString [Optional]. [String]

Recommended Action
Login into RTMT and check the configured threshold value for LogPartitionLowWaterMarkExceeded alert in Alert Central. If the configured value is set to a lower than the default threshold value unintentionally, change the value to default. Also, examine the trace and log file setting for each of the application in trace configuration page under CCM Serviceability. If the number of configured traces/logs is set to greater than 1000, adjust the trace settings from trace configuration page to default. Also, clean up the trace files that are less than a week old. You can clean up the traces using cli "file delete" or using Remote Browse from RTMT Trace and Log Central function.

MaliciousCall

Malicious Call Identification feature is invoked in Cisco CallManager.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Warning.</td>
</tr>
</tbody>
</table>
Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning

Parameters
Called Party Number [String] Called Device Name [String] Called Display Name [String] Calling Party Number [String] Calling Device Name [String] Calling Display Name [String]

Recommended Action
No action is required.

MaxDevicesPerNodeExceeded

An application has opened more devices than the limit set in the CTIManager service parameter, Maximum Devices Per Node.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiMaxDevicesPerNodeExceeded.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

Severity
WARNING

Routing List
SDL
SDI
Sys Log
Event Log

Recommended Action
One or more applications are controlling more devices than the CTI support allows on the specified Unified CM node. Review the application configuration and remove devices that are not required to be controlled. The stability of the system will be impacted if the total number of devices controlled by applications is not properly restricted to the device limit specified by the CTIManager service parameter, Maximum Devices Per Node.
MaxDevicesPerProviderExceeded

An application has opened more devices than the limit set in the CTIManager service parameter, Maximum Devices Per Provider.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiMaxDevicesPerProviderExceeded.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CtiManager

Severity

WARNING

Routing List

SDL
SDI
Sys Log
Event Log

Recommended Action

The application is controlling more devices than the CTI support allows. Review the application configuration and remove devices that are not required to be controlled. The stability of the system will be impacted if the application does not restrict support to the device limit specified by CTI in the CTIManager service parameter, Maximum Devices Per Provider.

MediaResourceListExhausted

The requested device type is not found in the media resource list or default list or the configured devices are not registered.

The requested device is not configured in the Media Resource Group List or Default List, or it's possible that one or more of the devices that are configured in the Media Resource Group List or Default List are not registered to Cisco Unified Communications Manager.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Enum Definitions for MediaResourceType is updated.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER
Warning-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning (4)

Parameters
Media Resource Type [Enum] Media Resource List Name [String]

Enum Definitions for MediaResourceType

<table>
<thead>
<tr>
<th>Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MediaTerminationPoint</td>
</tr>
<tr>
<td>2</td>
<td>Transcoder</td>
</tr>
<tr>
<td>3</td>
<td>ConferenceBridge</td>
</tr>
<tr>
<td>9</td>
<td>RSVP Agent</td>
</tr>
</tbody>
</table>

Recommended Action
First, go to Cisco Unified CM Administration to check the configuration of the devices that are part of the Media Resource Groups in the Media Resource Group List that was specified in the alarm (Media Resource Group List Configuration window and Media Resource Group Configuration window in Unified CM Administration).

Check whether the requested type of device is configured in any of the Media Resource Groups in that particular Media Resource Group List in Cisco Unified CM Administration; for RSVP Agent, check whether any media termination point or transcoder is configured in any of the Media Resource Groups in that particular Media Resource Group List. Next, go to the Media Resources menu in Cisco Unified CM Administration to see all the devices of the requested type and then check all the Media Resource Groups (irrespective of whether they belong to the Media Resource Group List for which the alarm is generated) to see whether the devices belong to at least one Media Resource Group.

If there exists some media resources of the requested type which do not belong to any Media Resource Groups, then these devices will belong to the default list. If the requested type of devices are not configured in any of the Media Resource Groups of the Media Resource Group List for which the alarm is generated or the Default List, add the requested type of device to a Media Resource Group in the specified Media Resource Group List or add it to the Default List.

To add a media resource to the Default List remove the Media Device from all the Media Resource Groups. In general, when a new media device is initially added to Unified CM it will automatically be added to the Default List. This Default List can be used by any device or trunk. But when the media device is added to any particular Media Resource Group it will not be available to the Default List. It can only be used by devices and trunks that are configured with the Media Resource Group List which have that particular Media Resource Group.

Note that a particular Media Resource Group can be added to multiple Media Resource Group Lists. If the requested device is properly configured in Cisco Unified CM Administration, check whether the device is registered to Unified CM. To do that go to the Media Resources menu of the requested type of device (such as Annunciator or Conference Bridge or Media Termination Point or Music On Hold Server or Transcoder) and click the Find button. It will display all the devices of that type with their status, device pool, etc. Check the status field to see whether it is registered with the Cisco Unified CallManager. Note that the display on the status field is not a confirmation that the device is registered to Unified CM. It may happen in a Unified CM cluster that the Publisher can only write to the Unified
CM database and suppose the Publisher goes down. Because the Subscriber may not be able to write to the database the devices may still display as registered in Unified CM Administration after they are unregistered. However, if the Publisher is down that should generate another alarm with higher priority than this alarm. If the device is not registered, click on the name of that particular device and check the type of the device.

Device types including Cisco Conference Bridge Software, Cisco Media Termination Point Software, or that specify a server name that is the same name as a Unified CM node of the cluster indicate that the requested device is a software device and is part of the Cisco IP Voice Media Streaming application. Check to be sure that the IP Voice Media Streaming App service is enabled on that Unified CM node (Cisco Unified Serviceability > Tools > Service Activation) and if it is not enabled, activate the Cisco IP Voice Media Streaming App service. Devices should try to register. You can also check the status of the service to be sure it is showing as Started (Tools > Control Center > Feature Services). If the device type is a type other than Cisco Conference Bridge Software, Cisco Media Termination Point Software, or a server name that is the same name as a Unified CM node, that indicates that the device is an external media resource to Unified CM.

Check the configuration (such as Conference Bridge type, MAC address, and conference bridge name in the case of a conference bridge; Media Termination Point name in the case of a Media Termination Point; Transcoder type, MAC address, and Transcoder name in the case of a Transcoder) of the device in Cisco Unified CM Administration and compare it with the configuration of the actual device. To check the configuration of the actual device you may need to refer to the user manual of the media device.

The user manual should provide all the details such as connecting to the media device to check the configuration, commands needed to view and update the configuration, and so on. If configuration in Unified CM and on the actual devices are different, make the necessary changes so that the configurations match. If the configuration matches and the device is still not registered, restart the external media device or the service associated with the external media device. If the external media device continues to fail to register with Unified CM, check the network connectivity between Unified CM and the media device.

### MemAllocFailed

CMI tried to allocate memory and failed.

Cisco Unified Communications Manager tried to read the Cisco Messaging Interface service parameters but not enough memory was allocated for the task and so the information could not be read.

### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Added to CallManager Catalog.</td>
</tr>
</tbody>
</table>

### Cisco Unified Serviceability Alarm Definition Catalog

CMIAAlarmCatalog/CMI

### Severity

WARNING

### Routing List

Event Log
SDI

**Parameter(s)**

Memory Allocation Failure(String)

**Recommended Action**

Use the Real-Time Monitoring Tool to check the performance counters related to system memory, to learn whether any memory leaks or spikes in CPU are occurring. Correct any anomalous memory issues you find. If you do not find any issues with memory, collect the system/application event logs and the performance (perfmon) logs and report this alarm to the Cisco Technical Assistance Center (TAC).

**MohNoMoreResourcesAvailable**

No more MOH resources available.

This alarm occurs when allocation of Music On Hold fails for all the registered MOH servers belonging to the Media Resource Group List and Default List. Each MOH server may fail for different reasons. Following are some of the reasons that could cause an MOH server allocation to fail: All the resources of MOH server are already in use; No matching codecs or capability mismatch between the held party and MOH server; Not enough bandwidth between the held party and MOH source; No audio stream available for the MOH server.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CallManager

**Severity**

Warning

**Recommended Action**

If all the resources of the MOH servers are already in use, check to be sure that all the MOH servers that belong to the Media Resource Groups of the indicated Media Resource Group List and Default List are configured and registered in all the applicable Unified CM nodes. To check the registration status go to the Media Resources > Music On Hold Server menu and click the Find button. It will display all the MOH servers with their status, device pool, and so on.

Check the status field to discover whether it is registered with Unified CM. Note that the display on the status field is not a confirmation that the device is registered to Unified CM. It may happen in a Unified CM cluster that the Publisher can only write to the Unified CM database and the Publisher goes down.
Warning-Level Alarms

Because the Subscriber may not be able to write to the database, the devices may still display as registered in Unified CM Administration after they are actually unregistered. However, if the Publisher is down that should generate another alarm with higher priority than this alarm.

The MOH allocation can also fail due to codec mismatch or capability mismatch between the endpoint and the MOH server. If there is a codec mismatch or capability mismatch (such as the endpoint using IPv6 addressing but MOH server supporting only IPv4), an MTP or transcoder should be allocated. If the MTP or transcoder is not allocated then either MediaResourceListExhausted (with Media Resource Type as Media termination point or transcoder) or MtpNoMoreResourcesAvailable alarm will be generated for the same Media Resource Group List and you should first concentrate on that alarm.

The MOH allocation may even fail after checking the region bandwidth between the regions to which the held party belongs and the region to which the MOH server belongs. Increasing the region bandwidth may be a solution to the problem, but that decision should be made after careful consideration of the amount of bandwidth you're willing to allocate per call between the set of regions.

You'll need to weigh different factors such as the total amount of available bandwidth, the average number of calls, the average number of calls using the MOH servers, approximate bandwidth use per call, and so on, and accordingly calculate the region bandwidth. Another possible cause is that the bandwidth needed for the call may not be available. This can occur if the MOH server and endpoint belong to different locations and the bandwidth that is set between the locations is already in use by other calls.

Examine the bandwidth requirements in your deployment to determine whether bandwidth between the locations can be increased. However, please note that increasing the bandwidth between these two locations means that you may need to reduce the bandwidth between other locations.

Refer to the System Guide, SRNDs, and related Unified CM documentation for more details. Be aware that reducing the bandwidth or removing the higher bandwidth codecs from configuration may result in poor voice quality during call. Consider increasing the total amount of network bandwidth. Another reason for the MOH allocation failure may be due to meeting the maximum number of unicast or multicast streams supported by the MOH server.

If all available streams are already in use none can be allocated. Finally, check the Music On Hold Audio Source Configuration window in Cisco Unified CM Administration to confirm that at least one audio source is configured. If an audio source is not configured, upload an audio file and then configure the audio source in Cisco Unified CM Administration (refer to the Music On Hold configuration documentation for specific details).

MtpNoMoreResourcesAvailable

Media termination point or transcoder allocation failed.

The alarm occurs when allocation of a media termination point (MTP) or transcoder fails for all the registered MTPs or transcoders belonging to the Media Resource Group List and Default List. Each MTP or transcoder may fail for different reasons. Following are some of the reasons that could cause an MTP or transcoder allocation to fail: a capability mismatch between the device endpoint and MTP/transcoder, codec mismatch between the endpoint and the MTP/transcoder; a lack of available bandwidth between the endpoint and the MTP/transcoder; or because the MTP/transcoders resources are already in use.

A capability mismatch may be due to the MTP/transcoder not supporting one or more of the required capabilities for the call such as Transfer Relay Point (which is needed for QoS or firewall traversal), RFC 2833 DTMF (which is necessary when one side of the call does not support RFC 2833 format for transmitting DTMF digits and the other side must receive the DTMF digits in RFC2833 format, resulting in conversion of the DTMF digits), RFC 2833 DTMF passthrough (in this case, the MTP or transcoder does not need to convert the DTMF digits from one format to another format but it needs to receive
Warning-Level Alarms

DTMF digits from one endpoint and transmit them to the other endpoint without performing any modifications), passthrough (where no codec conversion will occur, meaning the media device will receive media streams in any codec format and transmit them to the other side without performing any codec conversion), IPv4 to IPv6 conversion (when one side of the call supports only IPv4 and the other side of the call supports only IPv6 and so an MTP needs to be inserted to perform the necessary conversion between IPv4 and IPv6 packets), or multimedia capability (if a call involving video and/or data in addition to audio requires insertion of an MTP or transcoder then the MTP/transcoder which supports multimedia will be inserted).

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning

Parameter(s)
Media Resource List Name(String)

Recommended Action
If the MTP or transcoder allocation is failing due to a capability mismatch, it’s possible that the media device does not support the capability (such as IPv4 to IPv6 conversion, passthrough) or the capability might not be configured in the device. Please check the user guide and documentation of the media device to make sure that device supports all the necessary capabilities. Also, caution should be taken care if all the MTP or transcoders are configured with all the supported capabilities.

There are certain capabilities (such as RFC 2833 DTMF or RFC 2833 DTMF passthrough or passthrough) which could be supported by most of the MTPs or transcoders and there may be certain capabilities (such as IPv4 to IPv6 conversion and vice versa or Transfer Relay Point or multimedia capability) which can be supported by only a single MTP or transcoder depending on the devices that you have. For example, you may have IP phones that support only IPv4 protocol and there may also be IP phones that support only IPv6 protocol.

To make a call between IPv4-only and IPv6-only phones, you need to have an MTP configured to perform the conversion of IPv4 to IPv6 and vice versa. However, suppose all the MTPs or transcoders are configured with all the supported capabilities and only one MTP supports IPv4 to IPv6 conversion; if this MTP is configured with all the supported capabilities (which all the other MTPs or transcoders in the same MRGL or default MRGL also support) it may happen that this MTP can get allocated for Transfer Relay Point or RFC 2833 DTMF or RFC 2833 DTMF passthrough or passthrough instead. As a result, when the need arises for IPv4 to IPv6 conversion (which other MTPs or transcoders in the same

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Media Resource List Name parameter added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning

Parameter(s)
Media Resource List Name(String)

Recommended Action
If the MTP or transcoder allocation is failing due to a capability mismatch, it’s possible that the media device does not support the capability (such as IPv4 to IPv6 conversion, passthrough) or the capability might not be configured in the device. Please check the user guide and documentation of the media device to make sure that device supports all the necessary capabilities. Also, caution should be taken care if all the MTP or transcoders are configured with all the supported capabilities.

There are certain capabilities (such as RFC 2833 DTMF or RFC 2833 DTMF passthrough or passthrough) which could be supported by most of the MTPs or transcoders and there may be certain capabilities (such as IPv4 to IPv6 conversion and vice versa or Transfer Relay Point or multimedia capability) which can be supported by only a single MTP or transcoder depending on the devices that you have. For example, you may have IP phones that support only IPv4 protocol and there may also be IP phones that support only IPv6 protocol.

To make a call between IPv4-only and IPv6-only phones, you need to have an MTP configured to perform the conversion of IPv4 to IPv6 and vice versa. However, suppose all the MTPs or transcoders are configured with all the supported capabilities and only one MTP supports IPv4 to IPv6 conversion; if this MTP is configured with all the supported capabilities (which all the other MTPs or transcoders in the same MRGL or default MRGL also support) it may happen that this MTP can get allocated for Transfer Relay Point or RFC 2833 DTMF or RFC 2833 DTMF passthrough or passthrough instead. As a result, when the need arises for IPv4 to IPv6 conversion (which other MTPs or transcoders in the same

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Media Resource List Name parameter added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning

Parameter(s)
Media Resource List Name(String)

Recommended Action
If the MTP or transcoder allocation is failing due to a capability mismatch, it's possible that the media device does not support the capability (such as IPv4 to IPv6 conversion, passthrough) or the capability might not be configured in the device. Please check the user guide and documentation of the media device to make sure that device supports all the necessary capabilities. Also, caution should be taken care if all the MTP or transcoders are configured with all the supported capabilities.

There are certain capabilities (such as RFC 2833 DTMF or RFC 2833 DTMF passthrough or passthrough) which could be supported by most of the MTPs or transcoders and there may be certain capabilities (such as IPv4 to IPv6 conversion and vice versa or Transfer Relay Point or multimedia capability) which can be supported by only a single MTP or transcoder depending on the devices that you have. For example, you may have IP phones that support only IPv4 protocol and there may also be IP phones that support only IPv6 protocol.

To make a call between IPv4-only and IPv6-only phones, you need to have an MTP configured to perform the conversion of IPv4 to IPv6 and vice versa. However, suppose all the MTPs or transcoders are configured with all the supported capabilities and only one MTP supports IPv4 to IPv6 conversion; if this MTP is configured with all the supported capabilities (which all the other MTPs or transcoders in the same MRGL or default MRGL also support) it may happen that this MTP can get allocated for Transfer Relay Point or RFC 2833 DTMF or RFC 2833 DTMF passthrough or passthrough instead. As a result, when the need arises for IPv4 to IPv6 conversion (which other MTPs or transcoders in the same
Warning-Level Alarms

MRGL or default MRGL do not support), all the resources of MTP may be in use and the IPv4 to IPv6 conversion may fail. To avoid this kind of problem, setting the priority of the media resources may be a good idea.

This can be done only in the Media Resource Group List and not in the Default List of the media resources. In any Media Resource Group List all the Media Resource Groups have different priorities; during allocation the first Media Resource Group is always checked for availability of the requested type of the media devices. The first Media Resource Group in the Media Resource Group List will have the highest priority, then the second one, and so on.

To check all the Media Resource Groups and their priority go the Media Resources and Media Resource Group List of Cisco Unified CM Administration page and click the appropriate Media Resource Group List and check the Selected Media Resource Groups; the priority decreases from top to bottom. So, the MTP or transcoder that you want to be selected for the most basic functionalities should be positioned in the higher priority Media Resource Groups whereas the ones with more rare functionality should be positioned in the Media Resource Groups with lower priority. MTP/transcoder allocation may fail due to codec mismatch between the endpoint and the MTP/transcoder.

A solution may be to configure the MTP/transcoder with all the supported codecs (as specified in the user guide of the MTP/transcoder), but be aware that doing so might result in too much bandwidth being allocated for calls. You'll need to weigh different factors such as the total amount of available bandwidth, the average number of calls, approximate bandwidth use per call (not involving MTP/transcoder), and so on, and accordingly calculate the maximum bandwidth that can be allocated per call involving an MTP/transcoder and take that into consideration when configuring the supported codecs in the MTPs and transcoders. A good idea is to configure the media devices with all the supported codecs and set the region bandwidths to restrict too much bandwidth usage (refer to the Unified CM documentation for details on region and location settings).

Also, there may be a codec mismatch between the endpoint and the MTP/transcoders after considering the region bandwidth between the MTP/transcoder and the endpoint. Increasing the region bandwidth may be a solution to the problem, but again, that decision should be made after careful consideration of the amount of bandwidth you're willing to allocate per call between the set of regions. Another possible cause that an MTP/transcoder did not get allocated is because there was not enough available bandwidth for the call.

This can happen if the MTP/transcoder and endpoint belong to different locations and the bandwidth that is set between the locations is already in use by other calls. Examine the bandwidth requirements in your deployment to determine whether bandwidth between the locations can be increased.

However, please note that increasing the bandwidth between these two locations means that you may need to reduce the bandwidth between other locations. Refer to the System Guide, SRNDs, and related Unified CM documentation for more details. Be aware that reducing the bandwidth or removing the higher bandwidth codecs from configuration may result in poor voice quality during call. Consider increasing the total amount of network bandwidth available. Finally, if MTP or transcoder allocation fails due to capability mismatch or all the resources being in use, consider installing additional MTP or transcoder devices.

MTPDeviceRecoveryCreateFailed

MTP device recovery create failure. An error was encountered trying to restart the Media Termination Point device. This may be due to a shortage of application memory.
### Warning-Level Alarms

**Cisco Unified Serviceability Alarm Definition Catalog**

**CallManager/IpVms**

**Severity**

Warning

**Routing List**

SDI

Event Log

Sys Log

**Parameters**

OS Error Description(String)

**Recommended Action**

Restart the IP Voice Media Streaming App service or restart the server.

---

**NotEnoughChans**

Call attempt was rejected because requested gateway channel(s) could not be allocated. Some of the more common reasons for the lack of channel to place outgoing calls include: High call traffic volume that has the B-channels in the device fully utilized; B-channels have gone out of service for the following reasons: Taking the channel out of service intentionally to perform maintenance on either the near- or far-end; MGCP gateway returns an error code 501 or 510 for a MGCP command sent from Cisco Unified Communications Manager; MGCP gateway doesn't respond to an MGCP command sent by Unified CM three times; a speed and duplex mismatch exists on the Ethernet port between Unified CM and the MGCP gateway.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed severity level from Error to Warning and added existing Routing List elements and Parameters.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_CALLMANAGER-CALLMANAGER
Warning-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning

Parameters
Device Name(String)

Recommended Action
Add more gateway resources; Check the Unified CM advanced service parameter, Change B-channel Maintenance Status to determine if the B-channel has been taken out of service intentionally; Check the Q.931 trace for PRI SERVICE message to determine whether a PSTN provider has taken the B-channel out of service; Reset the MGCP gateway; Check the speed and duplex settings on the Ethernet port.

NoCallManagerFound

No Cisco Unified Communications Manager (Cisco Unified CM, formerly known as Cisco Unified CallManager) node has been configured. A Cisco Unified Communications Manager Group exists but it has no Cisco Unified CM node configured as its group member.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kNoCallManagerFound.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TFTP-TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP

Severity
Warning

Parameters
Error [String]

Recommended Action
In Cisco Unified CM Administration (System > Cisco Unified CM Group), configure at least one Cisco Unified CM node for the Cisco Unified CM Group referenced in this alarm. The Cisco Unified CM Group is part of the device pool to which the specified phone belongs.
PublishFailed

Publish Failed.

Unified CM attempted to store a number into the IME distributed cache, but the attempt failed. This is typically due to a transient problem in the IME distributed cache. The problem will self-repair under normal conditions. However, you should be aware that, as a consequence of this failure, the E.164 DID listed as part of the alarm will not be present in the IME distributed cache for a brief interval. Consequently, this may delay the amount of time until which you will receive VoIP calls made to that number - they may continue over the PSTN for somecallers. It is useful to be aware of this, in case you are trying to understand why a call is not being made over VoIP.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
WARNING

Recommended Action
If you notice single small numbers of these alarms in isolation, no action is required on your part. However, a large number of them indicates a problem in the IME distributed cache, most likely due to problems with Internet connectivity. Check your Internet connectivity.

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
DID(String)

QRTRequest

User submitted problem report using Quality Report Tool. User has experienced a problem with Phone and has submitted problem report.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed Data Collector Routing List element to Alert Manager.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CBB-CALLBACK
Warning-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CEF

Severity
Warning (4)

Routing List
SDI
Sys Log
Event Log
Alert Manager
SNMP Traps

Parameters
Category(String)
Reason Code(String)
Report Timestamp(String)
Device name(String)
Device IP address(String)
Directory number(String)

Recommended Action
Investigate the cause for problem report.

Rejected Routes

Rejected route due to Untrusted status.
This alarm is generated when Unified CM learned a route from the IME server. However, due to the configured Trusted or Untrusted list, the route was rejected.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
WARNING_ALARM

Recommended Action
This condition is not an error. However, it indicates to you that one of your users called a number which was reachable over IME, however, due to your configured Trusted or Untrusted list, an IME call will not be made. You might wish to consider adding the domain or prefix to your Trusted list or removing it from the Untrusted list.

Routing List
SDL
SDI
RouteListExhausted

An available route could not be found in the indicated route list. This alarm is generated when all members' status is unavailable or busy or when the member is down (out of service), not registered, or busy.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning (4)

Parameters
Route List Name [String]

Recommended Action
Consider adding additional routes in the indicated route list. For shared line when some phones are not ringing, check the busy trigger and maximum call settings of shared line phones; check whether there are some outstanding calls on that DN.

When one shared line phone answers an incoming call, the other shared line phone cannot see that remote-in-use call; check the privacy setting of the phone that answers the call.

Try to make a call directly to the member, bypassing the route list, to verify that there is not a device or connectivity issue. If you cannot identify the cause through these steps, gather the CCM (SDI) trace and contact the Cisco Technical Assistance Center; TAC may be able to locate a cause code which may provide additional explanation for this alarm.

ServiceStartupFailed

Service startup failure.

Cisco Unified Serviceability Alarm Definition Catalog
System/Generic

Severity
Warning (4)
Warning-Level Alarms

Parameters
None

Recommended Action
Restart the service.

ServingFileWarning

There was an error during processing of file request. This could happen if the requested file is not found by the server, or other error indicated by the “Reason” clause when processing the file request.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Name changed from kServingFileWarning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TFTP-TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP

Severity
Warning (4)

Parameters

Recommended Action
You can safely ignore this alarm if the reason shown in this alarm is “File not found” and if that file is the MAC address-based file name for a phone that you are auto-registering; in that case, the phone is not yet registered with the database and so it is normal for the phone's file not be found. In the case that auto-registration is disabled, this alarm shows that the phone or device is not added to Cisco Unified Communications Manager (Cisco Unified CM). Either add the phone to Cisco Unified CM or remove the phone from the network. If you still get this error after removing the phone(s), go to Cisco Unified Serviceability and enable Detailed level traces in the Trace Configuration window for the TFTP service and contact the Cisco Technical Assistance Center (TAC).

SparePartitionHighWaterMarkExceeded

The percentage of used disk space in the spare partition has exceeded the configured high water mark. Some of the trace files will be purged until the percentage of used disk space in the spare partition gets below the configured low water mark.
Warning-Level Alarms

Note
Spare Partition is not used for Intercompany Media Engine server. So this alert will not be triggered for Intercompany Media Engine.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_TCT-LPMTCT

Cisco Unified Serviceability Alarm Definition Catalog
System/LpmTct

Severity
Warning

Parameters
UsedDiskSpace [String] MessageString [Optional]. [String]

Recommended Action
Login into RTMT and check the configured threshold value for SparePartitionHighWaterMarkExceeded alert in Alert Central. If the configured value is set to a lower than the default threshold value unintentionally, change the value to default.

If you continue to receive this alert for half an hour after receiving the 1st alert, check for the disk usage for Spare partition under "Disk Usage" tab in RTMT. If the disk usage shown under that tab is higher than configured value in SparePartitionLowWaterMarkExceeded alert configuration, contact Cisco TAC to troubleshoot the cause of high disk usage in Common partition.

SSOuserNotInDB

User not found in database.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.5(1)</td>
<td>New alarm for this release.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER
Warning-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning

Parameters
Message(String)

Recommended Action
Perform sync manually or wait till next scheduled next sync.

SIPStopped

Cisco CallManager is not ready to handle calls for the indicated SIP device. Possible reasons could be internal database error, the SIP device is not activated on this node, the SIP device failed to register or the SIP device was deleted from admin page.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Enum Definitions for InTransportType and OutTransportType are updated. Recommended Action changed.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>IPV6Address parameter added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning (4)

Parameters

Enum Definitions for DeviceType
131—SIP_TRUNK
Enum Definitions for InTransportType

<table>
<thead>
<tr>
<th>Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TCP</td>
</tr>
<tr>
<td>2</td>
<td>UDP</td>
</tr>
<tr>
<td>3</td>
<td>TLS</td>
</tr>
<tr>
<td>4</td>
<td>TCP/UDP</td>
</tr>
</tbody>
</table>

Enum Definitions for OutTransportType

<table>
<thead>
<tr>
<th>Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TCP</td>
</tr>
<tr>
<td>2</td>
<td>UDP</td>
</tr>
<tr>
<td>3</td>
<td>TLS</td>
</tr>
</tbody>
</table>

Recommended Action
This alarm doesn't necessarily mean an error. It could occur as a result of normal administrative changes. If the alarm is unexpected, check whether the StationPortInitError alarm also fired. Check the Device Pool assigned to the SIP device identified in this alarm to ensure that the Cisco Unified Communications Manager Group of the Device Pool includes the Unified CM node that issued the alarm.

SIPLineRegistrationError

A SIP line attempted to register with CallManager and failed due to the error indicated in the Reason Code parameter. The alarm could indicate a device misconfiguration, database error, or an illegal/unknown device trying to attempt a connection.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
</table>
| 8.0(1)                              | • Severity changed from Error to Warning.  
• Enum Definitions for DeviceType are updated.  
• Enum Reasons table is updated. |

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning
### Parameters


### Enum Definitions for DeviceType

<table>
<thead>
<tr>
<th>Code</th>
<th>Device Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CISCO_30SP+</td>
</tr>
<tr>
<td>2</td>
<td>CISCO_12SP+</td>
</tr>
<tr>
<td>3</td>
<td>CISCO_12SP</td>
</tr>
<tr>
<td>4</td>
<td>CISCO_12S</td>
</tr>
<tr>
<td>5</td>
<td>CISCO_30VIP</td>
</tr>
<tr>
<td>6</td>
<td>CISCO_7910</td>
</tr>
<tr>
<td>7</td>
<td>CISCO_7960</td>
</tr>
<tr>
<td>8</td>
<td>CISCO_7940</td>
</tr>
<tr>
<td>9</td>
<td>CISCO_7935</td>
</tr>
<tr>
<td>12</td>
<td>CISCO_ATA_186</td>
</tr>
<tr>
<td>20</td>
<td>SCCP_PHONE</td>
</tr>
<tr>
<td>61</td>
<td>H323_PHONE</td>
</tr>
<tr>
<td>72</td>
<td>CTI_PORT</td>
</tr>
<tr>
<td>115</td>
<td>CISCO_7941</td>
</tr>
<tr>
<td>119</td>
<td>CISCO_7971</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>302</td>
<td>CISCO_7989</td>
</tr>
<tr>
<td>307</td>
<td>CISCO_7911</td>
</tr>
<tr>
<td>308</td>
<td>CISCO_7941G_GE</td>
</tr>
<tr>
<td>309</td>
<td>CISCO_7961G_GE</td>
</tr>
<tr>
<td>335</td>
<td>MOTOROLA_CN622</td>
</tr>
<tr>
<td>336</td>
<td>BASIC_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>348</td>
<td>CISCO_7931</td>
</tr>
<tr>
<td>358</td>
<td>CISCO_UNIFIED_COMMUNICATOR</td>
</tr>
<tr>
<td>365</td>
<td>CISCO_7921</td>
</tr>
<tr>
<td>369</td>
<td>CISCO_7906</td>
</tr>
<tr>
<td>374</td>
<td>ADVANCED_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>375</td>
<td>CISCO_TELEPRESENCE</td>
</tr>
<tr>
<td>404</td>
<td>CISCO_7962</td>
</tr>
<tr>
<td>412</td>
<td>CISCO_3951</td>
</tr>
<tr>
<td>431</td>
<td>CISCO_7937</td>
</tr>
<tr>
<td>434</td>
<td>CISCO_7942</td>
</tr>
</tbody>
</table>
### Enum Reason

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>MisconfiguredDirectoryNumber - There is a configuration mismatch between the directory numbers configured on the phone and the directory numbers configured in the Cisco Unified CM database. If this is a third-party phone, confirm that the phone configuration is correct and matches the Cisco Unified CM configuration. If this is a Cisco IP phone, confirm database replication has a &quot;good status&quot; in the Unified CM Database Status report. This can be found on the Cisco Unified Reporting web page. If the database replication status is good, reset the device. If the problem still persists, restart the TFTP service and the Cisco Unified CM service from the Control Center - Feature Services web page.</td>
</tr>
<tr>
<td>3</td>
<td>MalformedRegisterMessage - Cisco Unified CM cannot process a REGISTER message because of a problem with the format of the message. If the device is a third-party phone, confirm that the endpoint is sending a properly formatted REGISTER message.</td>
</tr>
<tr>
<td>#</td>
<td>Alarm Name</td>
</tr>
<tr>
<td>----</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>4</td>
<td>AuthenticationError</td>
</tr>
<tr>
<td>6</td>
<td>MaxLinesExceeded</td>
</tr>
<tr>
<td>7</td>
<td>TransportProtocolMismatch</td>
</tr>
<tr>
<td>8</td>
<td>BulkRegistrationError</td>
</tr>
</tbody>
</table>

**Recommended Action**

Verify that the directory number(s) on the device itself match the directory number(s) that are configured for that device in Cisco Unified CM Administration. Also, confirm that database replication is working. Refer to the reason code definitions for additional recommended actions.
SIPTrunkPartiallyISV

Some of the remote peers are not available to handle calls for this SIP Trunk.

The alarm provides a list of available remote peers and a list of unavailable remote peers, where each peer is separated by semicolon. For each available peer, the alarm provides resolved IP address and port number, and hostname or SRV (if configured on SIP trunk). For each unavailable peer, the alarm provides the hostname or SRV (if configured on SIP trunk), resolved IP address, port number, and reason code in the following format: ReasonCodeType=ReasonCode.

The ReasonCodeType depends on a SIP response from remote peer as defined in SIP RFCs (Remote), or depends on a reason code provided by Unified CM (Local).

The examples of possible reason codes include:
- Remote = 503 ("503 Service Unavailable" a standard SIP RFC error code)
- Remote = 408 ("408 Request Timeout" a standard SIP RFC error code)
- Local = 1 ("Request Timeout")
- Local = 2 (local SIP stack is unable to create a socket connection with remote peer)
- Local = 3 (DNS query failed)

For Local=3, IP address in the alarm is represented as zero, and when DNS SRV is configured on SIP trunk then port is represented as zero.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.5(1)</td>
<td>• New alarm for this release.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity

Warning

Routing List

SDL
SDI
Sys Log
Event Log

Parameters

SIP Trunk Name(String)
Unavailable remote peers with Reason Code(String)
Available remote peers for this SIP trunk (String)

Relevant Services

The available peer list is for notification purposes only; no action is required. For unavailable peers, the following corrective action should be taken.

- For Remote = 503, the possible reasons are:
  - Route/SIP trunk for originating side does not exist on remote peer. If remote peer is Unified CM, add a new SIP trunk in Unified CM Administration for the remote peer (Device > Trunk) and ensure the Destination Address and Destination Port fields are configured to point to the originating host (the originating host is the same node on which this alarm was generated). Also ensure the new SIP trunk has the incoming port in associated SIP Trunk Security Profile configured to be the same as originating side SIP Trunk destination port.
  
  - Route/SIP trunk for originating side does exist on remote peer but port is either used for SIP phone or other SIP trunk. If remote peer is Unified CM, in the Unified CM Administration for the remote peer (Device > Trunk), ensure the incoming port in associated SIP Trunk Security Profile is configured to be same as originating side SIP Trunk destination port.
  
  - Remote peer has limited resources to handle new calls. If remote peer is administered by a different system administrator, communicate the resource issue with the other administrator.

- For Remote = 408, the possible reason includes:
  - Remote peer has limited resources to handle new calls. If remote peer is administered by a different system administrator, communicate the resource issue with the other administrator.

- For Local = 1, the possible reason could be that no responses are received for OPTIONS request after all retries, when UDP transport is configured in SIP Trunk Security Profile assigned to the SIP trunk on originating side.

To fix this issue, perform the following steps:
  
  - If remote peer is Unified CM, in the remote peer Serviceability application, choose Tools > Control Center (Feature Services) and ensure the Cisco CallManager service is activated and started.
  
  - In the Unified CM Administration for the remote peer, choose Device > Trunk, and ensure the SIP trunk exists with the incoming port in associated SIP Trunk Security Profile configured to be same as originating side SIP Trunk destination port.
  
  - Check the network connectivity by using the CLI command utils network ping <remote peer> at the originating side.

- For Local = 2, the possible reason could be that Unified CM is unable to create the socket connection with remote peer.

To fix this issue, perform the following steps:
  
  - If remote peer is Unified CM, in the remote peer Serviceability application, choose Tools > Control Center (Feature Services) and ensure the Cisco CallManager service is activated and started.
  
  - In the Unified CM Administration for the remote peer, choose Device > Trunk, and ensure the SIP trunk exists with the incoming port in associated SIP Trunk Security Profile configured to be same as originating side SIP Trunk destination port.
  
  - Check the network connectivity by using the CLI command utils network ping <remote peer> at the originating side.

- For Local = 3, the possible reason could be that DNS server is not reachable, or DNS is not properly configured to resolve the hostname or SRV which is configured on the local SIP trunk.
Warning-Level Alarms

To fix this issue, perform the following steps:

– In the OS Administration, choose **Show > Network**, and verify that the DNS Details are correct. If it is not correct, then configure the correct DNS server information by using the CLI command `set network dns primary`.

– Check the network connectivity with DNS server by using the CLI command `utils network ping <remote peer>`, and ensure the DNS server is properly configured.

SoftwareLicenseNotValid

There is no valid software license; the Cisco IP Voice Media Streaming App service requires a valid software license to operate.

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/IpVms

Severity

Warning

Routing List

SDI

Event Log

Sys Log

Recommended Action

Install a valid software license and restart Cisco IP Voice Media Streaming App service.

StationEventAlert

A station device sent an alert to Cisco Unified Communications Manager, which acts as a conduit from the device to generate this alarm.

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Warning.</td>
</tr>
</tbody>
</table>
Warning-Level Alarms

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning

Parameters

Recommended Action
Refer to the specific device type and information passed via this alarm to determine the appropriate action.

TestAlarmWarning

Testing warning alarm.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
System/Test

Severity
Warning (4)

Recommended Action
None

TotalProcessesAndThreadsExceededThresholdStart

The current total number of processes and threads has exceeded the maximum number of tasks configured for Cisco RIS Data Collector service parameter. This situation could indicate some process is leaking or some process has thread leaking.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
System/System Access

Severity
Warning (4)
Warning-Level Alarms

Parameters
NumberOfProcesses [String] NumberOfThreads [String] Reason [String]
ProcessWithMostInstances [String] ProcessWithMostThreads [String]

Recommended Action
Check the Cisco RIS Data Collector service parameter, Maximum Number of Processes and Threads, to see if the parameter has been set to a low value. If it has been set to a low value, consider increasing the value or using the default value. Another possible action is that when a new Cisco product is integrated into Cisco Unified Communications Manager (Cisco Unified CM), new processes or threads are added to the system. Even in the normal process load situation, it's possible that the total number of processes and threads has exceeded the configured or default value of the Cisco RIS Data Collector service parameter, Maximum Number of Processes and Threads. Set that parameter to the maximum allowed value.

You can also review the details of this alarm to check the ProcessWithMostThreads description and the ProcessWithMostInstances description to discover which processes have the most threads and the most instances. Determine whether these values are reasonable for this process; if not, contact the owner of the process for troubleshooting the reasons why the thread count or the number of process instances is so high. It is also possible that Cisco RIS Data Collector sent a false alarm, which would indicate a defect in the Cisco RIS Data Collector service.

To determine if this is the cause of the alarm - after you have checked all the other errors described here - use RTMT to check the System object for performance counters Total Threads and Total Processes to confirm that the values in those counters do not exceed the value configured in the Cisco RIS Data Collector service parameter, Maximum Number of Processes and Threads. If the counters do not show a value that is higher than what is configured in the service parameter, restart Cisco RIS Data Collector service. If the alarm persists after restarting the service, go to Cisco Unified Serviceability and collect trace logs (Trace > Configuration) for Cisco Syslog, Cisco RIS Data Collector, Cisco AMC Service, and Cisco RIS Perfmon Logs and contact Cisco Technical Assistance Center (TAC) for detailed assistance.

ThreadKillingError

An error occurred when CMI tried to stop the CMI service.

As a normal part of the process of stopping the CMI service, open threads are closed (killed). This alarm indicates that a timeout has occurred which means that the shutdown process is taking longer than expected, causing the operating system to return an error.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kThreadKillingError.</td>
</tr>
<tr>
<td></td>
<td>Enum Definitions for MediaResourceType is updated.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CMIAlarmCatalog/CMI

Severity
WARNING
**Warning-Level Alarms**

**Routing List**
- Event Log
- SDI

**Parameter(s)**
- Error Information(String)

**Recommended Action**
Try restarting the CMI service. If the problem persists, collect the system/application event logs and the performance (perfmon) logs and report to Cisco Technical Assistance Center (TAC).

**UnableToSetorResetMWI**

An error occurred when setting the message waiting indication (MWI) lamp

**Cisco Unified Serviceability Alarm Definition Catalog**
- CallManager/CtiManager

**Severity**
- WARNING

**Routing List**
- SDL
- SDI
- Sys Log
- Event Log

**Parameter(s)**
- Directory Number(String)

**Recommended Action**
The line issuing the request to set the MWI lamp on the target line might not have the proper partitions/calling search space settings to allow it to reach the target line. Check the partitions and calling search space of the line that is requesting to set MWI on the target line. The target line should be able to receive a call from the line that is attempting to set MWI.

**UserInputFailure**

EMCC login failure due to invalid user input due to invalid user credentials or the credentials have expired. Reason Code: 2—Authentication Error.

**Cisco Unified Serviceability Alarm Definition Catalog**
- System/EMAAlarmCatalog

**Severity**
- Warning(4)
Routing List
Sys Log
Event Log
Alert Manager

Parameters
Device Name(String)
Login Date/Time(String)
Login UserID(String)
Reason(String)

Recommended Action
Try again with valid credentials or try resetting the credentials.

UserUserPrecedenceAlarm
User-to-user IE was not successfully tunneled to destination; please refer to reason code for additional details.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>• Severity changed from Error to Warning.</td>
</tr>
<tr>
<td></td>
<td>• Enum definitions updated.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Warning

Parameters
Device Name. [String] Reason Code [Enum]
Warning-Level Alarms

Enum Definitions

<table>
<thead>
<tr>
<th>Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>HopCountExceeded—The hop count field in passing User-to-User IE exceeded the maximum value of 10. The reason could be the presence of routing loops across the Unified CM trunk interfaces (PRI, intercluster trunk, and so on). The recommended action is to check that no routing loops exist across the Unified CM trunk interfaces (PRI, intercluster trunk, and so on) and gateway (H.323) devices related to the indicated failed call. By examining trace files and CDR data in all Unified CM nodes and route patterns in gateways (H.323) that are involved in routing of the indicated failed call, you may be able to detect a translation pattern, route list or other routing mechanism that is part of the loop. Update the routing mechanism that resulted in the loop, and then if the looping route pattern was on a Unified CM, reset the affected route list/pattern in an attempt to clear the route loop; if that fails, reset the affected trunk/gateway or if the looping route pattern was on an H.323 gateway, restart the gateway.</td>
</tr>
<tr>
<td>3</td>
<td>UserUserIEDropped—The passing UserUserIE is dropped. If the indicated device is an H.323 intercluster trunk then the possible reason could be that the Passing Precedence Level Through UIIE checkbox in the Trunk Configuration window in Unified CM is not enabled; the recommended action is to verify that the Passing Precedence Level Through UIIE checkbox has been enabled. If the indicated device is an MGCP gateway with Device Protocol set to Digital Access PRI, the possible reason could be that in the incoming UIIE message, either the IEID is not set to USER_USER_IE (126) or the User specific protocol ID value is not set to PRI_4ESS_UIIE_DEFAULT_PROT_DISC (0x00); the recommended action is to verify that the far-end side of the configured PRI trunk interface supports PRI 4ESS UIIE-based MLPP and sends the UIIE message with IEID value set to USER_USER_IE (126) and the User specific protocol ID value is set to PRI_4ESS_UIIE_DEFAULT_PROT_DISC (0x00).</td>
</tr>
</tbody>
</table>

Recommended Action

For HopCountExceeded alarm, the recommended action is to check that no routing loops exist across the Unified CM trunk interfaces (PRI, intercluster trunk, and so on) and gateway (H.323) devices related to the indicated failed call. By examining trace files and CDR data in all Unified CM nodes and route patterns in gateways (H.323) that are involved in routing of the indicated failed call, you may be able to detect a translation pattern, route list or other routing mechanism that is part of the loop.

Update the routing mechanism that resulted in the loop, and then if the looping route pattern was on a Unified CM, reset the affected route list/pattern in an attempt to clear the route loop; if that fails, reset the affected trunk/gateway or if the looping route pattern was on a H.323 gateway, restart the gateway.

For call failure reason UserUserIEDropped, if the indicated device is an H.323 intercluster trunk then the recommended action is to verify that the Passing Precedence Level Through UIIE checkbox has been enabled on the Trunk Configuration window. If the indicated device is an MGCP gateway with Device Protocol set to Digital Access PRI and Passing Precedence Level Through UIIE is enabled on the gateway, then verify that the far-end side of the configured PRI trunk interface supports PRI 4ESS UIIE-based MLPP and sends the UIIE message with IEID value set to USER_USER_IE (126) and the User specific protocol ID value set to PRI_4ESS_UIIE_DEFAULT_PROT_DISC (0x00).
BeginThrottlingCallListBLFSubscriptions

Cisco Unified Communications Manager has initiated throttling of CallList BLF Subscriptions as a preventive measure to avoid overloading the system. This alarm is raised when the total number of active BLF subscriptions exceeds the configured limit set by the Presence Subscription Throttling Threshold service parameter.

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity
Warning (4)

Parameters
Active External Presence Subscriptions [UInt] CallList BLF Subscriptions Throttling Threshold [UInt] CallList BLF Subscriptions Resume Threshold [UInt] Total Begin Throttling CallList BLF Subscriptions [UInt]

Recommended Action
Determine if CPU and memory resources are available to meet the higher demand for CallList BLF Subscriptions. If so, increase the CallListBLFSubscriptionsThrottlingThreshold and correspondingly the CallListBLFSubscriptionsResumeThreshold. If not, increase system resources to meet the demand.

kANNAudioCreateDirFailed

Unable to create a subdirectory to contain announcement files. This may be caused by insufficient disk storage. Announcements may not play correctly as a result of this error.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Added more Recommended Action text. Updated parameters and changed severity level from Error to Warning.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/IpVms

Severity
Warning (4)

Parameters
OS Error Text(String)
Path Name(String)
Recommended Action
Check for available free space on the common data storage area. If full, take action to remove old trace files to free space. Restart the Cisco IP Voice Media Streaming App service.

**MOHDeviceRecoveryCreateFailed**

An error got triggered restarting the Music On Hold (MOH) device. It may have been caused by a shortage of memory resources.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed severity level from Error to Warning and added existing Routing List elements.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

**Severity**

Warning (4)

**Routing List**

SDI
Event Log
Sys Log

**Parameters**

ErrorText(String)
Error(ULong0)

**Recommended Action**

Check the status of the MOH device. If it is not registered and available, restart the Cisco IP Voice Media Streaming App service or restart the server.
**kDeviceMgrExitEventCreationFailed**

Creation of device manager exit event failure. An error was reported when allocating an exit-control event for a SW media device. The device will not be registered with CallManager or active.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Added Routing List elements. Changed severity level from Error to Warning.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

**Severity**

Warning (4)

**Routing List**

SDI
Event Log
Sys Log

**Parameters**

Device Name [String]
Trace Name [String]
OS Error Text [String]

**Recommended Action**

This error may be due to a memory resource shortage. Restart the Cisco IP Voice Media Streaming App service or restart the Cisco Unified CM server.

**kMOHDeviceRecordNotFound**

MOH device was not found for the server. This device gets added automatically when a server gets added to the configuration.
Warning-Level Alarms

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning (4)

Recommended Action
If MOH functionality is required, you will have to remove and readd the device to database.

Caution
Adding and removing the device may impact other configuration settings, for example, Cisco Unified Communications Manager groups and media resource groups.

kMOHBadMulticastIP

An invalid multicast IP address (out of range) was found.

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS
Warning-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Warning (4)

Routing List
SDI
Event Log
Sys Log

Parameters
Codec Type [String]
Multicast IP Address [String]

Recommended Action
Correct the setting on the Music-on-Hold device configuration for multicast address.

SSODisabled

Single Sign On (SSO) disabled on Cisco Unified CM.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.5(1)</td>
<td>New alarm added for this release.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Warning

Parameters
Message(String)

Recommended Action
Run CLI command to enable SSO.

SSONullTicket

A null ticket was passed.
Warning-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Warning

Parameters
Message(String)

Recommended Action
Get non null ticket and retry.

SSOServerUnreachable

SSO server could not be reached.

History

Cisco Unified Communications Release  Action
Communications  8.5(1)  New alarm added for this release.
Release

Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Warning

Parameters
Message(String)

Recommended Action
Check reachability to SSO server.

WDStopped

WebDialer application stopped and was unloaded from Tomcat.
Facility/Sub-Facility
CCM_JAVA_APPS_TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Warning

Parameters
Servlet Name [String] Reason [String]

Recommended Action
Check if Tomcat service is up.

Notice-Level Alarms

The notice-level alarm is 5 and no action is needed unless the information is unexpected. Notifications about interesting system-level conditions which are not error conditions. Informational in nature but having a more important need-to-know status. Examples are:

- System-wide notifications
- Process is shutting down gracefully on request
- Clearing of previously raised conditions
- A device or subsystem un-registering or shutting down for expected and normal reason (for individual phone related expected and normal unregistering or shutting down, informational level should be used)
- Password change notification and upgrade notification

authExpired

Authentication failure due to expired soft lock. User credentials have expired.
Notice-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Notice (5)

Routing List
Event Log

Parameters
Authentication failure due to expired soft lock.(String)

Recommended Action
Administrator may reset the credential.

authMustChange

Authentication failed because it is marked that it must be changed by the user. “User must change” is set on this credential. The user must change the credential.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Added more description and Routing List element. Corrected the parameter.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Notice (5)

Routing List
Event Log
Parameters
UserID [String]

Recommended Action
User or Administrator may reset credential.

BChannelISV

B-channel is in service.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed severity level from Informational to Notice.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Notice

Parameters
Channel Id. [UInt] Unique channel ID [String] Device name. [String]

Recommended Action
None

CallManagerOnline

Cisco CallManager service has completed initialization is online.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Notice (5)
CertValidityOver30Days

Alarm indicates that the certificate expiry is approaching but the expiry date is more than 30 days.

Parameters
CCM Version [String]

Recommended Action
None

CertValidityOver30Days

Alarm indicates that the certificate expiry is approaching but the expiry date is more than 30 days.

Parameters
Message(String)

Recommended Action
Regenerate the certificate that is about to expire by accessing the Cisco Unified Operating System and go to Certificate Management. If the certificate is issued by a CA, generate a CSR, submit the CSR to CA, obtain a fresh certificate from CA, and upload it to Cisco Unified CM.

CodeYellowExit

CodeYellowExit. Unified CM has ceased throttling calls and has exited the Code Yellow state.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Error to Notice.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager
Severity
Notice

Parameters

Recommended Action
None

credReadFailure

Error occurred attempting to read a credential in the database. This could be a network or database issue.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Changed severity level to Notice from Informational. Corrected parameter and added Routing List element.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Notice (5)

Routing List
Event

Parameters
Credential read failure for(String)

Recommended Action
Ensure credential (user name) exists. Could be a database problem.

DbInsertValidatedDIDFailure

The Insertion of a IME provided e164DID has failed. A failure occurred attempting to insert a Cisco Unified Active Link learned DID

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager
Notice-Level Alarms

Severity
NOTICE

Routing List
SDL
SDI
Sys Log
Event Log
SNMP Traps
Data Collector

Parameter(s)
e164 DID(String)
Granting Domain(String)

Recommended Action
Verify the DID and the granting domain. Check other associated alarms. Verify the database integrity.

DChannelISV

Indicated D-channel has gone in service.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Notice.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Notice

Parameters
Channel Id. [UInt] Unique channel Id [String] Device Name. [String] Device IP address [String]

Recommended Action
None
EMAppStopped

EM Application started. Application is shutting down gracefully because of an unloaded from Tomcat.

Cisco Unified Serviceability Alarm Definition Catalog
System/EMAlarmCatalog

Severity
NOTICE

Routing List
Sys Log
Event Log

Parameter(s)
Servlet Name(String)

Recommended Action
No action required.

EndPointRegistered

This alarm occurs when a device is successfully registered with Cisco Unified Communications Manager.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
NOTICE

Routing List
SDL
SDI
Sys Log
Data Collector
SNMP Traps
Alternate Syslog

Parameter(s)
Device name(String)
Device MAC address(String)
Device IP address(String)
Protocol(String)
Device description(String)
User ID(String)
Load ID(String)
Associated directory numbers(String)
Performance monitor object type(Enum)
Device type(Enum)
Configured Gatekeeper Name(String)
Technology Prefix Name(String)
Zone Information(String)
Alternate Gatekeeper List(String)
Active Gatekeeper(String)
Call Signal Address(String)
RAS Address(String)
IPV6Address(String)
IPAddressAttributes(Enum)
IPV6AddressAttributes(Enum)
ActiveLoadId(String)
InactiveLoadId(String)

Enum Definitions - Performance monitor object type

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Cisco Phone</td>
</tr>
</tbody>
</table>

Enum Definitions - Device type

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CISCO_30SP+</td>
</tr>
<tr>
<td>2</td>
<td>CISCO_12SP+</td>
</tr>
<tr>
<td>3</td>
<td>CISCO_12SP</td>
</tr>
<tr>
<td>4</td>
<td>CISCO_12S</td>
</tr>
<tr>
<td>5</td>
<td>CISCO_30VIP</td>
</tr>
<tr>
<td>6</td>
<td>CISCO_7910</td>
</tr>
<tr>
<td>7</td>
<td>CISCO_7960</td>
</tr>
<tr>
<td>8</td>
<td>CISCO_7940</td>
</tr>
<tr>
<td>9</td>
<td>CISCO_7935</td>
</tr>
<tr>
<td>12</td>
<td>CISCO_ATA_186</td>
</tr>
<tr>
<td>20</td>
<td>SCCP_PHONE</td>
</tr>
<tr>
<td>61</td>
<td>H323_PHONE</td>
</tr>
<tr>
<td>72</td>
<td>CTI_PORT</td>
</tr>
<tr>
<td>Value</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>115</td>
<td>CISCO_7941</td>
</tr>
<tr>
<td>119</td>
<td>CISCO_7971</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>302</td>
<td>CISCO_7989</td>
</tr>
<tr>
<td>307</td>
<td>CISCO_7911</td>
</tr>
<tr>
<td>308</td>
<td>CISCO_7941G_GE</td>
</tr>
<tr>
<td>309</td>
<td>CISCO_7961G_GE</td>
</tr>
<tr>
<td>335</td>
<td>MOTOROLA_CN622</td>
</tr>
<tr>
<td>336</td>
<td>BASIC_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>348</td>
<td>CISCO_7931</td>
</tr>
<tr>
<td>358</td>
<td>CISCO_UNIFIED_COMMUNICATOR</td>
</tr>
<tr>
<td>365</td>
<td>CISCO_7921</td>
</tr>
<tr>
<td>369</td>
<td>CISCO_7906</td>
</tr>
<tr>
<td>374</td>
<td>ADVANCED_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>375</td>
<td>CISCO_TELEPRESENCE</td>
</tr>
<tr>
<td>404</td>
<td>CISCO_7962</td>
</tr>
<tr>
<td>412</td>
<td>CISCO_3951</td>
</tr>
<tr>
<td>431</td>
<td>CISCO_7937</td>
</tr>
<tr>
<td>434</td>
<td>CISCO_7942</td>
</tr>
<tr>
<td>435</td>
<td>CISCO_7945</td>
</tr>
<tr>
<td>436</td>
<td>CISCO_7965</td>
</tr>
<tr>
<td>437</td>
<td>CISCO_7975</td>
</tr>
<tr>
<td>446</td>
<td>CISCO_3911</td>
</tr>
<tr>
<td>468</td>
<td>CISCO_UNIFIED_MOBILE_COMMUNICATOR</td>
</tr>
<tr>
<td>478</td>
<td>CISCO_TELEPRESENCE_1000</td>
</tr>
<tr>
<td>479</td>
<td>CISCO_TELEPRESENCE_3000</td>
</tr>
<tr>
<td>480</td>
<td>CISCO_TELEPRESENCE_3200</td>
</tr>
<tr>
<td>481</td>
<td>CISCO_TELEPRESENCE_500</td>
</tr>
<tr>
<td>484</td>
<td>CISCO_7925</td>
</tr>
<tr>
<td>493</td>
<td>CISCO_9971</td>
</tr>
<tr>
<td>495</td>
<td>CISCO_6921</td>
</tr>
<tr>
<td>496</td>
<td>CISCO_6941</td>
</tr>
<tr>
<td>497</td>
<td>CISCO_6961</td>
</tr>
<tr>
<td>20000</td>
<td>CISCO_7905</td>
</tr>
<tr>
<td>30002</td>
<td>CISCO_7920</td>
</tr>
<tr>
<td>30006</td>
<td>CISCO_7970</td>
</tr>
<tr>
<td>30007</td>
<td>CISCO_7912</td>
</tr>
</tbody>
</table>
Notice-Level Alarms

Enum Definitions - IPAddressAttributes

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>30008</td>
<td>CISCO_7902</td>
</tr>
<tr>
<td>30016</td>
<td>IP_COMMUNICATOR</td>
</tr>
<tr>
<td>30018</td>
<td>CISCO_7961</td>
</tr>
<tr>
<td>30019</td>
<td>7936</td>
</tr>
<tr>
<td>30035</td>
<td>IP_STE</td>
</tr>
</tbody>
</table>

Enum Definitions - IPV6AddressAttributes

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown - The device has not indicated what this IPv6 address is used for</td>
</tr>
<tr>
<td>1</td>
<td>Administrative only - The device has indicated that this IPv6 address is used for administrative communication (web interface) only</td>
</tr>
<tr>
<td>2</td>
<td>Signal only - The device has indicated that this IPv6 address is used for control signaling only</td>
</tr>
<tr>
<td>3</td>
<td>Administrative and signal - The device has indicated that this IPv6 address is used for administrative communication (web interface) and control signaling</td>
</tr>
</tbody>
</table>

Recommended Action

No action is required.

H323Started

Cisco CallManager is ready to handle calls for the indicated H323 device. Cisco Unified Communications Manager is ready to communicate with the indicated H.323 device. Note that this alarm describes the readiness of Unified CM to communicate with the indicated device, but does not provide information about the state of the H.323 device (whether it is ready to communicate as well).
 ICTCallThrottlingEnd
Cisco CallManager starts handling calls for the indicated H323 device. Cisco CallManager has ceased throttling calls on the indicated H.323 device.
### Notice-Level Alarms

#### Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

#### Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

#### Severity
Notice

#### Parameters
- Device Name. [String]
- IP Address [String]
- Device type. [Optional] [Enum]
- Device description [Optional]. [String]

#### Enum Definitions for DeviceType
- 125—TRUNK

#### Recommended Action
None.

**kDeviceMgrMoreThan50SocketEvents**

More than 50 events returned from TCP link. The specified Cisco Unified Communications Manager TCP link has returned a large number of TCP events. This indicates an unexpected flood of events.

#### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Severity changed from Informational to Notice.</td>
</tr>
</tbody>
</table>

#### Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

#### Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms
Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Notice-Level Alarms

Severity
Notice

Parameters
Trace Name [String]

Recommended Action
No action is required. Monitor for reoccurrence. This could be an indication of a security issue.

MGCPGatewayGainedComm

The MGCP gateway has established communication with Cisco Unified Communications Manager.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Notice.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Notice

Parameters
Device Name [String]

Recommended Action
Informational purposes only; no action is required.

MaxCallDurationTimeout

An active call was cleared because the amount of time specified in the Maximum Call Duration Timer service parameter had elapsed. If the allowed call duration is too short, you can increase the value. If you do not want a limit on the duration of an active call, you can disable the limit. If the duration is correct but you did not expect a call to ever exceed that duration, check the trace information around the time that this alarm occurred to try to determine if a gateway port had failed to release a call.
Notice-Level Alarms

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>- Severity changed from Informational to Notice.</td>
</tr>
<tr>
<td></td>
<td>- Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>- Originating Device name(String)</td>
</tr>
<tr>
<td></td>
<td>- Destination Device name(String)</td>
</tr>
<tr>
<td></td>
<td>- Call start time(UInt)</td>
</tr>
<tr>
<td></td>
<td>- Call stop time(UInt)</td>
</tr>
<tr>
<td></td>
<td>- Calling Party Number(String)</td>
</tr>
<tr>
<td></td>
<td>- Called Party Number(String)</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Notice

Parameters
Maximum Call Duration (minutes) [UInt]
Originating Device name(String)
Destination Device name(String)
Call start time(UInt)
Call stop time(UInt)
Calling Party Number(String)
Called Party Number(String)

Recommended Action
If the duration of the call is too short, increase the value in the Cisco CallManager service parameter or disable the maximum duration by setting the Maximum Call Duration Timer parameter to zero. If you suspect a hung gateway port, check the trace files around the time that this alarm occurred to search for the gateway that was involved in the call, then check the status of that gateway to determine if all ports are functioning normally.

SDLinkISV

SDL link to remote application is restored. This alarm indicates that the local Cisco CallManager has gained communication with the remote Cisco CallManager.
The remote Cisco CallManager should also indicate SDLLinkISV with a different LinkID.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Notice.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity

Notice

Parameters


Enum Definitions for LocalApplicationId and RemoteApplicationID

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>CallManager</td>
</tr>
<tr>
<td>200</td>
<td>CTI Manager</td>
</tr>
</tbody>
</table>

Recommended Action

None
SIPNormalizationScriptOpened

Cisco Unified CM opened the script for the SIP device.
The normalization script for the indicated SIP device has been successfully loaded, initialized, and activated on Cisco Unified CM.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.5(1)</td>
<td>• New alarm for this release.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Notice

Routing List
SDL
SDI
Sys Log
Event Log

Parameters
Device Name(String)
Script Name(String)
In Use Memory(UInt)

Recommended Action
Notification purposes only; no action is required.

SIPNormalizationScriptClosed

Cisco Unified CM has closed the script for the SIP device. The script is closed at one of the following conditions:
• The indicated device (SIP trunk) was reset manually or automatically.
• The trunk was deleted manually.
• Due to script error or resource error or internal error.

When the script is closed, Cisco Unified CM will not invoke normalization script message handlers for the indicated SIP device.
Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Notice-Level Alarms

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Notice

Routing List
SDL
SDI
Sys Log
Event Log

Parameters
Device Name(String)
Script Name(String)
Reason Code(Enum)
Reason Text(String)
Additional Information(String)

Enum Definitions - Reason Code

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DeviceResetManually—The associated device is reset manually using Cisco Unified CM Administration.</td>
</tr>
<tr>
<td>2</td>
<td>DeviceResetAutomatically—The associated device is reset automatically; the reset was triggered by an execution error in the script.</td>
</tr>
<tr>
<td>3</td>
<td>DeviceDeleted—The associated device is manually deleted in Cisco Unified CM Administration.</td>
</tr>
<tr>
<td>4</td>
<td>ScriptDisassociated—A configuration change occurred in Cisco Unified CM Administration and the script is no longer associated with the device.</td>
</tr>
</tbody>
</table>
Chapter 6      Cisco Unified Serviceability Alarms and CiscoLog Messages

Notice-Level Alarms

Recommended Action

This alarm serves as a notification of the script closure, if the alarm has occurred due to a SIP trunk maintenance window or any other expected reason for the script to close. If this alarm is unexpected, check for an occurrence of the SIPNormalizationScriptError alarm and refer to the specific action based on the reason code identified in the SIPNormalizationScriptError alarm.

SIPNormalizationAutoResetDisabled

An error occurred repeatedly and Cisco Unified CM disabled the script.

The script failed due to execution errors that occurred three times within a 10 minute period. As a result, the normalization script for the indicated SIP device has been disabled. Cisco Unified CM do not attempts to automatically reset either the script or the device for the purpose of recovering the script.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.5(1)</td>
<td>• New alarm for this release.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity

Notice

Routing List

SDL
SDI
Sys Log
Event Log

Parameters

Device Name(String)
Script Name(String)
Script Type(String)
Reason Code(Enum)
Reason Text(String)
Additional Information(String)

**Enum Definitions - Reason Code**

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ScriptResetDisabled—The system has automatically reset the script three times within a 10 minute period due to script execution errors; on the fourth occurrence of this error, Cisco Unified CM disabled the script.</td>
</tr>
<tr>
<td>2</td>
<td>TrunkResetDisabled—The system has automatically reset the trunk three times within a 10 minute period due to script execution errors; on the fourth occurrence of this error, Cisco Unified CM disabled the script.</td>
</tr>
</tbody>
</table>

**Recommended Action**
Notification purposes; examine the information and perform the recommended actions in the SIPNormalizationScriptError alarm, which should have been issued before this alarm.

### SIPStarted

Cisco CallManager is ready to handle calls for the indicated SIP device. This alarm does not indicate the current state of the SIP device, only that Cisco CallManager is prepared to handle calls to or from the SIP device.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
</table>
| 8.0(1)                              | • Severity changed from Informational to Notice.  
• Enum Definitions for InTransportType and OutTransportType are updated. |
| 7.1                                 | IPV6Address parameter added. |

**Facility/Sub-Facility**
CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/CallManager

**Severity**
Notice
Parameters
Device Name. [String]
IP Address [Optional]. [String]
Device type. [Optional] [Enum]
Device description [Optional]. [String]
Incoming Port Number. [UInt]
Outgoing Port Number. [UInt]
Incoming Transport Type [Enum]
Outgoing Transport Type [Enum]
IPV6Address [Optional]. [String]

Enum Definitions for DeviceType
- 131—SIP_TRUNK

Enum Definitions for InTransportType

<table>
<thead>
<tr>
<th>Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TCP</td>
</tr>
<tr>
<td>2</td>
<td>UDP</td>
</tr>
<tr>
<td>3</td>
<td>TLS</td>
</tr>
<tr>
<td>4</td>
<td>TCP/UDP</td>
</tr>
</tbody>
</table>

Enum Definitions for OutTransportType

<table>
<thead>
<tr>
<th>Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TCP</td>
</tr>
<tr>
<td>2</td>
<td>UDP</td>
</tr>
<tr>
<td>3</td>
<td>TLS</td>
</tr>
</tbody>
</table>

Recommended Action
None

SIPTrunkISV

All remote peers are available to handle calls for this SIP trunk.

This alarm indicates that all remote peers are available to handle the calls for this SIP trunk. For each peer, the alarm provides the resolved IP address and port number, and hostname or SRV (if configured on SIP trunk).
Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Notice

Routing List
SDL
SDI
Sys Log
Event Log

Parameters
SIP Trunk Name(String)
Available remote peers for this SIP trunk(String)

Recommended Action
Notification purpose only; no action is required.

SMDICmdError

CMI receives an invalid incoming SMDI message.

There are two kinds of incoming messages that Cisco Unified Communications Manager can accept from the voice messaging system; they are OP:MWI(SP)nnnnnnn!(D) and RMV:MWI(SP)nnnnnnn!(D) (where:nnnnnnnnnn = station number (can be 7 or 10 digits), (D) = End Of Transmission, (SP) = space). The first message activates the message waiting indicator (MWI). The second deactivates the message waiting indicator. CMI triggers this alarm if the received MWI message does not have one of the acceptable formats as described.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.5(1)</td>
<td>• New alarm for this release.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kSMDICmdError.</td>
</tr>
</tbody>
</table>
Notice-Level Alarms

**Cisco Unified Serviceability Alarm Definition Catalog**
CMIAlarmCatalog/CMI

**Severity**
NOTICE

**Routing List**
Event Log
SDI

**Parameter(s)**
Invalid SMDI command(String)

**Recommended Action**
Contact the vendor of the third-party voice messaging system and discover why it is sending SMDI message with an invalid format.

### SMDIMessageError

SMDI message contains invalid DN.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kSMDIMessageError.</td>
</tr>
</tbody>
</table>

Some voice messaging systems send SMDI messages to Cisco Unified Communications Manager (Unified CM) with an invalid DN specifically for the purpose of verifying that Unified CM is functioning properly. In such cases, if the Validate DNs service parameter is set to True, CMI triggers this alarm because the DN cannot be found in the Unified CM database.

**Cisco Unified Serviceability Alarm Definition Catalog**
CMIAlarmCatalog/CMI

**Severity**
NOTICE

**Routing List**
Event Log
SDI

**Parameter(s)**
Invalid SMDI command(String)
**Recommended Action**
Verify that the Cisco Messaging Interface service parameter Validate DNs is set to false.

**TestAlarmNotice**

Testing notice alarm.

**Facility/Sub-Facility**
CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Test

**Severity**
Notice (5)

**Recommended Action**
None

**TotalProcessesAndThreadsExceededThresholdEnd**

The current total number of processes and threads is less than the maximum number of tasks configured in the Cisco RIS Data Collector service parameter, Maximum Number of Processes and Threads. This can occur because a product which was integrated into Cisco Unified Communications Manager has been disabled or deactivated, which reduces the total number of processes and threads running on the system. Another cause for the number of processes or threads to decrease is that one or more processes has been stopped, which reduces the total number of processes and threads running on the system.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Informational to Notice.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_CALLMANAGER-CALLMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**
System/System Access

**Severity**
Notice

**Parameters**
NumberOfProcesses [String] NumberOfThreads [String] Reason [String]
Informational-Level Alarms

The informational-level of alarm is 6 and no action is needed. Informational messages provide historical data such as internal flows of the application or per-request information. Informational messages are used for troubleshooting by users who are familiar with the basic flows of the application. An example would be a normal (expected) event occurred that the customer may want to be notified about.

AdministrativeEvent

Failed to write into the primary file path. Audit Event is generated by this application.

Cisco Unified Serviceability Alarm Catalog
AuditLog

Severity
INFORMATIONAL

Recommended Action
Ensure that the primary file path is valid and the corresponding drive has sufficient disk space. Also, make sure that the path has security permissions similar to default log file path.

AdminPassword

Administrative password got changed. If the change was unsuccessful or successful, a message gets displayed.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Added descriptive text.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Informational (6)

Parameters
(String)
**AuditEventGenerated**

Audit Event is generated by this application because failed to write into the primary file path.

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Generic

**Severity**
Informational (6)

**Parameters**
- UserID (String)
- ClientAddress (String)
- EventType (String)
- ResourceAccessed(String)
- EventStatus (String)
- AuditDetails (String)
- ComponentID (String)

**Recommended Action**
Ensure that the primary file path is valid and the corresponding drive has sufficient disk space. Also, make sure that the path has security permissions similar to default log file path.

**AgentOnline**

Agent online

**Facility/Sub-Facility**
CCM_TCD-TCD

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/TCD SRV

**Severity**
Informational (6)

**Recommended Action**
None
AgentOffline

Agent offline

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Informational (6)

Recommended Action
None

AuthenticationSucceeded

Login Authentication succeeded.

Facility/Sub-Facility
CCM_TOMCAT_APPS-LOGIN

Cisco Unified Serviceability Alarm Definition Catalog
System/Login

Severity
Informational (6)

Parameters
Login IP Address/Hostname [String] Login Date/Time [String] Login UserID [String] Login Interface [String]

Recommended Action
If this event is expected, no action is required; otherwise, notify the administrator.

authSuccess

Successfully authenticated this user.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
<tr>
<td>8.5(1)</td>
<td>Parameter updated.</td>
</tr>
</tbody>
</table>
Cisco Unified Serviceability Alarm Definition Catalog
System/IMS

Severity
Informational (6)

Parameters
UserID(String)

Recommended Action
None

**BDIStarted**

Application started successfully.

**Facility/Sub-Facility**
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Informational (6)

Recommended Action
None

**BuildStat**

Device configuration files are being built. This alarm provides information about the BUILD ALL operation to build all types of configuration files.

**Facility/Sub-Facility**
CCM_TFTP-TFTP

Cisco Unified Serviceability Alarm Definition Catalog
System/TFTP

Severity
Informational (6)

Parameters
**Informational-Level Alarms**

**CiscoDirSyncStarted**

Cisco DirSync Application started. Application started successfully.

**Facility/Sub-Facility**

CCM_JAVA_APPS-TOMCATAPPLICATIONS

**Cisco Unified Serviceability Alarm Definition Catalog**

System/Java Applications

**Severity**

Informational (6)

**Recommended Action**

None

**CiscoDirSyncProcessStarted**

LDAPSync process started to sync user data on configured agreement ID.

**Facility/Sub-Facility**

CCM_JAVA_APPS-TOMCATAPPLICATIONS

**Cisco Unified Serviceability Alarm Definition Catalog**

System/Java Applications

**Severity**

Informational (6)

**Parameters**

AgreementId [String]

**Recommended Action**

None

**CiscoDirSyncProcessCompleted**

LDAPSync process completed on particular sync agreement.

**Facility/Sub-Facility**

CCM_JAVA_APPS-TOMCATAPPLICATIONS
Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Informational (6)

Parameters
AgreementId [String]

Recommended Action
None

CiscoDirSyncProcessStoppedManually

LDAPSync process stopped manually on particular sync agreement.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Informational (6)

Parameters
AgreementId [String]

Recommended Action
None

CiscoDirSyncProcessStoppedAuto

LDAPSync process stopped automatically on particular sync agreement. It will restart automatically.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Informational (6)

Parameters
AgreementId [String]
**Recommended Action**
None

### CLM_ConnectivityTest

CLM Connectivity Test Failed. Cluster Manager detected a network error.

**Facility/Sub-Facility**
CCM_CLUSTERMANAGER/CLUSTERMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Cluster Manager

**Severity**
Informational (6)

**Operating System**
Appliance

**Parameters**
Node's IP(String)
Error (String)

**Recommended Action**
Verify connectivity between cluster nodes.

### CLM_IPSecCertUpdated

IPSec self-signed cert updated. The IPSec self-signed cert from a peer node in the cluster has been imported due to a change.

**Facility/Sub-Facility**
CCM_CLUSTERMANAGER/CLUSTERMANAGER

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Cluster Manager

**Severity**
Informational (6)

**Operating System**
Appliance

**Parameters**
Node's or IP(String)
Recommended Action
None

CLM_IPAddressChange

IP address change in cluster. The IP address of a peer node in the cluster has changed.

Facility/Sub-Facility
CCM_CLUSTERMANAGER/CLUSTERMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
System/Cluster Manager

Severity
Informational (6)

Operating System
Appliance

Parameters
Node's (String)
Node's Old IP(String)
Node's New IP(String)

Recommended Action
None

CLM_PeerState

Current ClusterMgr session state. The ClusterMgr session state with another node in the cluster has changed to the current state.

Facility/Sub-Facility
CCM_CLUSTERMANAGER/CLUSTERMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
System/Cluster Manager

Severity
Informational (6)

Operating System
Appliance

Parameters
Node's or IP(String)
Node's State(String)

**Recommended Action**
None

---

**credFullUpdateSuccess**

Credential was successfully updated.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**

**System/IMS**

**Severity**
Informational (6)

**Parameters**
(String)

**Recommended Action**
None

---

**credFullUpdateFailure**

An error was encountered during update of credential fields.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**

**System/IMS**

**Severity**
Informational (6)

**Parameters**
(String)
credReadSuccess

Successfully read a credential.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
</tbody>
</table>

credUpdateFailure

The credential update failed most likely because the credential did not pass the security requirements (too short or credential used before, for example).

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>Added more descriptive text.</td>
</tr>
</tbody>
</table>
Informational-Level Alarms

Recommended Action
Determine issue (check length requirements, etc.) for this credential and retry.

credUpdateSuccess

Credential was successfully updated.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0(1)</td>
<td>Error message added.</td>
</tr>
</tbody>
</table>

DirSyncScheduledTaskOver

Directory synchronization operation started.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Informational (6)

Parameters
SchedulerID [String] TaskID [String]

Recommended Action
None
DirSyncSchedulerEngineStopped

DirSync scheduler engine stopped.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Informational (6)

Parameters
DirSyncSchedulerVersion [String]

Recommended Action
None

DirSyncNewScheduleInserted

New schedule inserted in the DirSync Scheduler.

Facility/Sub-Facility
CCM_JAVA_APPS/JAVAAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Informational (6)

Parameters
EngineScheduleID [String]

Recommended Action
None

DRFLA2MAFailure

DRF Local Agent to Master Agent connection has some problems.
Informational-Level Alarms

Facility/Sub-Facility
CCM_JAVA_APPS/JAVAAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Informational (6)

Parameters
Reason [String]

Recommended Action
Check if the Master Agent is up and the port is authorized.

**DRFMA2LAFailure**

Master Agent was unable to send a backup/restore request to the local agent.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Manager Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>New name changed from CiscoDRFMA2LAFailure. Descriptive text and Recommended action changed.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_JAVA_APPS/JAVAAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/DRF

Severity
Informational (6)

Parameters
Reason [String]
**CiscoDRFComponentRegistered**

DRF Successfully Registered the requested component.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Manager Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from CiscoDRFComponentRegistered.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_DRF_LOCAL & CCM_DRF_MASTER/DRF

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DRF

**Severity**

Informational (6)

**Parameters**

Reason(String)

**Recommended Action**

Ensure that the registered component is needed for backup/restore operation.

---

**CiscoDhcpdRestarted**

DHCP Daemon restarted successfully.

**Facility/Sub-Facility**

CCM_JAVA_APPS-TOMCATAPPLICATIONS

**Cisco Unified Serviceability Alarm Definition Catalog**

System/Java Applications

**Severity**

Informational (6)

**Parameters**

Reason [String]

**Recommended Action**

None
CiscoHardwareLicenseInvalid

Installation on invalid or obsolete hardware. Cannot upload license files.

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
INFORMATIONAL

Routing List
Sys Log
Event Log
SNMP Traps

Parameter(s)
Reason(String)

Recommended Action
Obtain correct hardware and reinstall.

CiscoLicenseFileInvalid

License File is invalid.

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
INFORMATIONAL

Routing List
Sys Log
Event Log
SNMP Traps

Parameter(s)
Reason(String)

Recommended Action
Rehost the License files.

CMInitializationStateTime

Indicates the amount of time required to complete initialization for the specified state.
**Facility/Sub-Facility**
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

**Severity**
Informational (6)

**Parameters**
Initialization State [String] Initialization Time [String] Initialization Time in Milliseconds [UInt]

**Recommended Action**
None

**CMIServiceStatus**

CMI service is running and working properly.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCMIServiceStatus.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CMIAAlarmCatalog/CMI

**Severity**
INFORMATIONAL

**Routing List**
Event Log
SDI

**Parameter(s)**
Service Priority(String)

**Recommended Action**
Informational purpose only; no action is required.

**CMTotalInitializationStateTime**

Indicates the amount of time required to complete the specified total system initialization state.
Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational (6)

Parameters
Total Initialization Time [String] Total Initialization Time in Milliseconds [UInt]

Recommended Action
None

ConnectionToPDPInService

A connection was successfully established between Cisco Unified Communications Manager (Unified CM) and the policy decision point (PDP).

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational(6)

Routing List
SDL
SDI
Sys Log
Event Log

Parameters
Policy Decision Point(String)

Recommended Action
None

CriticalEvent

Failed to write into the primary file path. Audit Event is generated by this application.

Cisco Unified Serviceability Alarm Catalog
AuditLog
Severity
INFORMATIONAL

Recommended Action
Ensure that the primary file path is valid and the corresponding drive has sufficient disk space. Also, make sure that the path has security permissions similar to default log file path.

CtiDeviceClosed

Application closed a device.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiDeviceClosed.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

Severity
INFORMATIONAL

Routing List
SDL
SDI
Sys Log
Event Log
Data Collector

Parameter(s)
Device Name(String)
RTP Address(String)
Reason code.(Enum)

Enum Definitions - Reason Code

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown</td>
</tr>
<tr>
<td>1</td>
<td>CallManager service is not available to process request; verify that the Call-Manager service is active. Check the Cisco Unified Serviceability Control Center section in Cisco Unified CM Administration (Tools &gt; Control Center - Feature Services)</td>
</tr>
<tr>
<td>2</td>
<td>Device has unregistered with Cisco Unified Communications Manager</td>
</tr>
</tbody>
</table>
**Recommended Action**
This alarm is for informational purposes only; no action is required.

### CtiDeviceInService

Device is back in service.

#### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiDeviceInService.</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CtiManager

**Severity**
INFORMATIONAL
Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Device Name(String)

Recommended Action
This alarm is for informational purposes only; no action is required.

CtiDeviceOpened

Application opened a device.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiDeviceOpened.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

Severity
INFORMATIONAL

Routing List
SDL
SDI
Sys Log
Event Log
Data Collector

Parameter(s)
Device Name(String)
RTP Address(String)

Recommended Action
This alarm is for informational purposes only; no action is required.
CtiLineOpened

Application opened the line.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiLineOpened.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

Severity
INFORMATIONAL

Routing List
SDL
SDI
Sys Log
Event Log
Data Collector

Parameter(s)
Directory Number(String)
Partition(String)
Device Name(String)

Recommended Action
This alarm is for informational purposes only; no action is required.

CtiLineOutOfService

Line is out of service.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiLineOutOfService.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager
Severity
INFORMATIONAL

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Directory Number(String)
Device Name(String)

Recommended Action
This alarm is for informational purposes only; no action is required.

CtiProviderClosed

CTI application closed the provider. The IP address is shown in either IPv4 or IPv6 format depending on the IP addressing mode of the application.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiProviderClosed.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

Severity
INFORMATIONAL

Routing List
SDL
SDI
Sys Log
Event Log
Data Collector

Parameter(s)
Login User Id(String)
IPAddress(String)
IPV6Address(String)
## Enum Definitions - Reason Code

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown</td>
</tr>
<tr>
<td>1</td>
<td>Heartbeat from application missed. Possible causes include network connectivity issues or Unified CM node experiencing high CPU usage. Make sure that the network connectivity between Unified CM and the application by pinging the application server host from Cisco Unified OS Administration and take steps to establish connectivity if it has been lost. Also check for and fix any network issues or high CPU usage on the application server.</td>
</tr>
<tr>
<td>2</td>
<td>Unexpected shutdown; possibly cause is application disconnected the TCP connection. Also check for and fix any network issues or high CPU usage on the application server.</td>
</tr>
<tr>
<td>3</td>
<td>Application requested provider close</td>
</tr>
<tr>
<td>4</td>
<td>Provider open failure; application could not be initialized</td>
</tr>
<tr>
<td>5</td>
<td>User deleted. User associated with the application is deleted from the Unified CM Administration</td>
</tr>
<tr>
<td>6</td>
<td>SuperProvider permission associated with the application is removed. Verify the user group configuration for the user in Unified CM Admin under (User Management &gt; End User/Application User), select the user and review the associated permissions information</td>
</tr>
<tr>
<td>7</td>
<td>Duplicate certificate used by application. Verify the CAPF profile configuration for the user in Unified CM Admin under (User Management &gt; End User CAPF Profile/Application User CAPF Profile), select the CAPF profile of the user and review the associated information</td>
</tr>
<tr>
<td>8</td>
<td>CAPF information unavailable. Verify the CAPF profile configuration for the user in Unified CM Admin under (User Management &gt; End User CAPF Profile/Application User CAPF Profile), select the CAPF profile of the user and review the associated information</td>
</tr>
<tr>
<td>9</td>
<td>Certificate compromised. Verify the CAPF profile configuration for the user in Unified CM Admin under (User Management &gt; End User CAPF Profile/Application User CAPF Profile), select the CAPF profile of the user and review the associated information</td>
</tr>
</tbody>
</table>
Recommended Action
This alarm is for informational purposes only; no action is required.

**CtiProviderOpened**

CTI Application opened the provider successfully. The IP address is shown in either IPv4 or IPv6 format depending on the IP addressing mode of the Application.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiProviderOpened.</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CtiManager

**Severity**

INFORMATIONAL

**Routing List**

SDL
SDI
Sys Log
Event Log
Data Collector

**Parameter(s)**

Login User Id(String)
### Informational-Level Alarms

Version Number(String)  
IPAddress(String)  
IPV6Address(String)  

**Recommended Action**  
This alarm is for informational purposes only; no action is required.

**CtiDeviceOutofService**

Device is out of service.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiDeviceOutofService.</td>
</tr>
<tr>
<td></td>
<td>Severity changed from Notice to Informational.</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CtiManager  

**Severity**  
INFORMATIONAL  

**Routing List**

SDL  
SDI  
Sys Log  
Event Log  

**Parameter(s)**  
Device Name(String)  

**Recommended Action**  
This alarm is for informational purposes only; no action is required.

**CtiLineClosed**

Application closed the line.
Informational-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CtiManager

Severity
INFORMATIONAL

Routing List
SDL
SDI
Sys Log
Event Log
Data Collector

Parameter(s)
Directory Number(String)
Partition(String)
Device Name(String)
Reason code.(Enum)

Enum Definitions - Reason Code

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown</td>
</tr>
<tr>
<td>1</td>
<td>CallManager failure</td>
</tr>
<tr>
<td>2</td>
<td>Device has unregistered with Cisco Unified Communications Manager; wait for the device to register</td>
</tr>
<tr>
<td>3</td>
<td>CTI failed to rehome the line; verify that the device is registered</td>
</tr>
<tr>
<td>4</td>
<td>Undefined line, possible cause could be that line is no more active on that device due to extension mobility login or logout</td>
</tr>
<tr>
<td>5</td>
<td>Device removed</td>
</tr>
<tr>
<td>6</td>
<td>Provider controlling the device is closed</td>
</tr>
<tr>
<td>7</td>
<td>Protocol used by the device is not supported</td>
</tr>
</tbody>
</table>

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiLineClosed.</td>
</tr>
<tr>
<td></td>
<td>Severity changed from Notice to Informational.</td>
</tr>
</tbody>
</table>

Cisco Unified Communications Release Action
8.0(1) Name changed from kCtiLineClosed.
Severity changed from Notice to Informational.
## Informational-Level Alarms

### CtiLineInService

*Line is back in service*

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Name changed from kCtiLineInService.</td>
</tr>
<tr>
<td></td>
<td>Severity changed from Notice to Informational.</td>
</tr>
</tbody>
</table>

**Cisco Unified Serviceability Alarm Definition Catalog**

*CallManager/CtiManager*

**Severity**

INFORMATIONAL

**Routing List**

SDL
SDI
Sys Log
Event Log

**Recommended Action**

This alarm is for informational purposes only; no action is required.
DatabaseDefaultsRead

Database default information was read successfully.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Notice to Informational.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational

Parameters
None

Recommended Action
None

DefaultDurationInCacheModified

Default value of a Certificate duration in cache is modified in the Service Parameter page. This usually means that the Default Certificate duration in cache value is modified in the Service Parameter page.

Cisco Unified Serviceability Alarm Catalog
System/TVS

Severity
INFORMATIONAL

Routing List
SDI
Event Log
Data Collector
Sys Log

Recommended Action
None
DeviceApplyConfigInitiated

Device Apply Config initiated.

This alarm occurs when a system administrator presses the Apply Config button in Cisco Unified Communications Manager (Unified CM). The Apply Config button initiates a conditional restart on devices that support conditional restart. This button triggers the system to determine if any relevant configuration has changed for the device. If the configuration changes can be applied dynamically, they are made without service interruption. If a change requires that the device reregister with Unified CM, reregistration occurs automatically. If a change requires a restart, the device will be automatically restarted. If the load ID for a device changes, the device will initiate a background download of the new firmware. The new firmware can then be applied immediately or at a later time. For phones and devices that do not support conditional restart, clicking Apply Config causes these devices to restart.

Severity
Informational

Routing List
SDL
SDI
Sys Log

Parameter(s)
Device name(String)
Product type(String)
Device type(Enum)

Enum Definitions for Device type
- 493—CISCO_9971

Recommended Action
None

DeviceApplyConfigResult

Cisco IP Phone has applied its configuration.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>Added DeviceApplyConfigResult to the Phone Catalog in the CallManager alarm definitions.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/Phone
Severity
Informational (6)

Parameters
DeviceName(String)
IPAddress(String)
UnifiedCM_Result(String)
Phone_Result(String)
Reason(String)

Recommended Action
No action is required.

DeviceDnInformation

List of directory numbers associated with the device.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational (6)

Parameters
Device Name [String] Device type. [Optional] [Enum]Station Desc [String] Station Dn [String]

Enum Definitions for DeviceType

<table>
<thead>
<tr>
<th>Code</th>
<th>Device Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CISCO_30SP+</td>
</tr>
<tr>
<td>2</td>
<td>CISCO_12SP+</td>
</tr>
<tr>
<td>3</td>
<td>CISCO_12SP</td>
</tr>
<tr>
<td>4</td>
<td>CISCO_12S</td>
</tr>
<tr>
<td>5</td>
<td>CISCO_30VIP</td>
</tr>
<tr>
<td>6</td>
<td>CISCO_7910</td>
</tr>
<tr>
<td>7</td>
<td>CISCO_7960</td>
</tr>
<tr>
<td>8</td>
<td>CISCO_7940</td>
</tr>
<tr>
<td>9</td>
<td>CISCO_7935</td>
</tr>
<tr>
<td>10</td>
<td>CISCO_VGC_PHONE</td>
</tr>
<tr>
<td>11</td>
<td>CISCO_VGC_VIRTUAL_PHONE</td>
</tr>
</tbody>
</table>
## Informational-Level Alarms

<table>
<thead>
<tr>
<th>Code</th>
<th>Device Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>CISCOATA_186</td>
</tr>
<tr>
<td>20</td>
<td>SCCP_PHONE</td>
</tr>
<tr>
<td>21</td>
<td>STATION_PHONE_APPLICATION</td>
</tr>
<tr>
<td>30</td>
<td>ANALOG_ACCESS</td>
</tr>
<tr>
<td>40</td>
<td>DIGITAL_ACCESS</td>
</tr>
<tr>
<td>41</td>
<td>DIGITAL_ACCESS_T1</td>
</tr>
<tr>
<td>42</td>
<td>DIGITAL_ACCESS+</td>
</tr>
<tr>
<td>43</td>
<td>DIGITAL_ACCESS_WS-X6608</td>
</tr>
<tr>
<td>47</td>
<td>ANALOG_ACCESS_WS-X6624</td>
</tr>
<tr>
<td>48</td>
<td>VGC_GATEWAY</td>
</tr>
<tr>
<td>50</td>
<td>CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>51</td>
<td>CONFERENCE_BRIDGE_HARDWARE</td>
</tr>
<tr>
<td>52</td>
<td>CONFERENCE_BRIDGE_HARDWARE_HDV2</td>
</tr>
<tr>
<td>53</td>
<td>CONFERENCE_BRIDGE_HARDWARE_WS-SVC-CMM</td>
</tr>
<tr>
<td>61</td>
<td>H323_PHONE</td>
</tr>
<tr>
<td>62</td>
<td>H323_GATEWAY</td>
</tr>
<tr>
<td>70</td>
<td>MUSIC_ON_HOLD</td>
</tr>
<tr>
<td>71</td>
<td>DEVICE_PILOT</td>
</tr>
<tr>
<td>72</td>
<td>CTI_PORT</td>
</tr>
<tr>
<td>73</td>
<td>CTI_ROUTE_POINT</td>
</tr>
<tr>
<td>80</td>
<td>VOICE_MAIL_PORT</td>
</tr>
<tr>
<td>83</td>
<td>SOFTWARE_MEDIA TERMINATION POINT_HDV2</td>
</tr>
<tr>
<td>84</td>
<td>CISCO_MEDIA_SERVER</td>
</tr>
<tr>
<td>85</td>
<td>CISCO_VIDEO_CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>90</td>
<td>ROUTE_LIST</td>
</tr>
<tr>
<td>100</td>
<td>LOAD_SIMULATOR</td>
</tr>
<tr>
<td>110</td>
<td>MEDIA_TERMINATION_POINT</td>
</tr>
<tr>
<td>111</td>
<td>MEDIA_TERMINATION_POINT_HARDWARE</td>
</tr>
<tr>
<td>112</td>
<td>MEDIA_TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>113</td>
<td>MEDIA_TERMINATION_POINT_WS-SVC-CMM</td>
</tr>
<tr>
<td>115</td>
<td>CISCO_7941</td>
</tr>
<tr>
<td>119</td>
<td>CISCO_7971</td>
</tr>
<tr>
<td>120</td>
<td>MGCP_STATION</td>
</tr>
<tr>
<td>121</td>
<td>MGCP_TRUNK</td>
</tr>
<tr>
<td>122</td>
<td>GATEKEEPER</td>
</tr>
<tr>
<td>124</td>
<td>7914_14_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>125</td>
<td>TRUNK</td>
</tr>
<tr>
<td>Code</td>
<td>Device Type</td>
</tr>
<tr>
<td>------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>126</td>
<td>TONE_ANNOUNCEMENT_PLAYER</td>
</tr>
<tr>
<td>131</td>
<td>SIP_TRUNK</td>
</tr>
<tr>
<td>132</td>
<td>SIP_GATEWAY</td>
</tr>
<tr>
<td>133</td>
<td>WSM_TRUNK</td>
</tr>
<tr>
<td>134</td>
<td>REMOTEDESTINATIONPROFILE</td>
</tr>
<tr>
<td>254</td>
<td>UNKNOWNMGCP_GATEWAY</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>302</td>
<td>CISCO_7989</td>
</tr>
<tr>
<td>307</td>
<td>CISCO_7911</td>
</tr>
<tr>
<td>308</td>
<td>CISCO_7941G_GE</td>
</tr>
<tr>
<td>309</td>
<td>CISCO_7961G_GE</td>
</tr>
<tr>
<td>335</td>
<td>MOTOROLA_CN622</td>
</tr>
<tr>
<td>336</td>
<td>BASIC_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>358</td>
<td>CISCO_UNIFIED_COMMUNICATOR</td>
</tr>
<tr>
<td>365</td>
<td>CISCO_7921</td>
</tr>
<tr>
<td>369</td>
<td>CISCO_7906</td>
</tr>
<tr>
<td>374</td>
<td>ADVANCED_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>375</td>
<td>CISCO_TELEPRESENCE</td>
</tr>
<tr>
<td>404</td>
<td>CISCO_7962</td>
</tr>
<tr>
<td>412</td>
<td>CISCO_3951</td>
</tr>
<tr>
<td>431</td>
<td>CISCO_7937</td>
</tr>
<tr>
<td>434</td>
<td>CISCO_7942</td>
</tr>
<tr>
<td>435</td>
<td>CISCO_7945</td>
</tr>
<tr>
<td>436</td>
<td>CISCO_7965</td>
</tr>
<tr>
<td>437</td>
<td>CISCO_7975</td>
</tr>
<tr>
<td>20000</td>
<td>CISCO_7905</td>
</tr>
<tr>
<td>30002</td>
<td>CISCO_7920</td>
</tr>
<tr>
<td>30006</td>
<td>CISCO_7970</td>
</tr>
<tr>
<td>30007</td>
<td>CISCO_7912</td>
</tr>
<tr>
<td>30008</td>
<td>CISCO_7902</td>
</tr>
<tr>
<td>30016</td>
<td>CISCO_IP_COMMUNICATOR</td>
</tr>
<tr>
<td>30018</td>
<td>CISCO_7961</td>
</tr>
<tr>
<td>30019</td>
<td>CISCO_7936</td>
</tr>
<tr>
<td>30027</td>
<td>ANALOG_PHONE</td>
</tr>
<tr>
<td>30028</td>
<td>ISDN_BRI_PHONE</td>
</tr>
<tr>
<td>30032</td>
<td>SCCP_GATEWAYVIRTUAL_PHONE</td>
</tr>
<tr>
<td>30035</td>
<td>IP_STE</td>
</tr>
</tbody>
</table>
Recommended Action

None

DeviceImageDownloadStart

Cisco IP Phone has started downloading its firmware load (image).

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>Added DeviceImageDownloadStart to the Phone Catalog in the CallManager alarm definitions.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/Phone

Severity

Informational (6)

Routing List

SDL
SDI
Sys Log
Alternate Syslog
Data Collector

Parameters

DeviceName(String)
IPAddress(String)
Active(String)
RequestedLoadId(String)

Recommended Action

No action is required.

DeviceImageDownloadSuccess

Cisco IP Phone has successfully downloaded its image.
Informational-Level Alarms

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/Phone

Severity
Informational (6)

Routing List
SDL
SDI
Sys Log
Alternate Syslog
Data Collector

Parameters
DeviceName(String)
IPAddress(String)
Method(Enum)
Active(String)
Inactive(String)
Server from which the firmware was downloaded(String)

Enum Definitions - Method

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TFTP</td>
</tr>
<tr>
<td>2</td>
<td>HTTP</td>
</tr>
<tr>
<td>3</td>
<td>PPID</td>
</tr>
</tbody>
</table>

Recommended Action
No action is required.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.1(5)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Included Routing List.</td>
</tr>
<tr>
<td></td>
<td>• Updated Parameters.</td>
</tr>
<tr>
<td></td>
<td>• Included Enum Definitions - Method</td>
</tr>
<tr>
<td>7.1</td>
<td>Added DeviceImageDownloadSuccess to the Phone Catalog in the CallManager alarm definitions.</td>
</tr>
</tbody>
</table>
DeviceRegistered

A device successfully registered with Cisco Unified Communications Manager.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
</table>
| 8.5(1)                              | Following information is updated:  
|                                     | • Enum Definitions for Performance Monitor ObjType |
| 8.0(1)                              | Following information is updated:  
|                                     | • Enum Definitions for Performance Monitor ObjType  
|                                     | • Enum Definitions for Device type |
| 7.1                                 | Parameters added for IPv6: IPV6Address[Optional].[String], IPAddressAttributes[Optional].[Enum], IPV6AddressAttributes [Optional].[Enum], and ActiveLoadId [Optional].[String]. |

Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity

Informational (6)

Routing List

SDL
SDI
Sys Log
Event Log
Data Collector
SNMP Traps

Parameters

Device name.[String]
Device MAC address [Optional].[String]
Device IP address [Optional].[String]
Protocol.[String]
Device description [Optional].[String]
User ID [Optional].[String]
Load ID. [Optional][String]
Associated directory numbers.[Optional].[String]
Performance monitor object type [Enum]
Device type. [Optional][Enum]
Configured GateKeeper Name [Optional].[String]
Technology Prefix Name [Optional].[String]
Zone Information [Optional].[String]
Alternate Gatekeeper List [Optional].[String]
Active Gatekeeper [Optional].[String]
Call Signal Address [Optional].[String]
RAS Address [Optional].[String]
IPV6Address[Optional].[String]
IPAddressAttributes[Optional].[Enum]
IPV6AddressAttributes [Optional].[Enum]
ActiveLoadId [Optional].[String]
InactiveLoadId [Optional],[String]

Enum Definitions for Performance Monitor ObjType

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cisco CallManager</td>
</tr>
<tr>
<td>3</td>
<td>Cisco Lines</td>
</tr>
<tr>
<td>4</td>
<td>Cisco H.323</td>
</tr>
<tr>
<td>5</td>
<td>Cisco MGCP Gateway</td>
</tr>
<tr>
<td>6</td>
<td>Cisco MOH Device</td>
</tr>
<tr>
<td>7</td>
<td>Cisco Analog Access</td>
</tr>
<tr>
<td>8</td>
<td>Cisco MGCP FXS Device</td>
</tr>
<tr>
<td>9</td>
<td>Cisco MGCP FXO Device</td>
</tr>
<tr>
<td>10</td>
<td>Cisco MGCP T1CAS Device</td>
</tr>
<tr>
<td>11</td>
<td>Cisco MGCP PRI Device</td>
</tr>
<tr>
<td>12</td>
<td>Cisco MGCP BRI Device</td>
</tr>
<tr>
<td>13</td>
<td>Cisco MTP Device</td>
</tr>
<tr>
<td>14</td>
<td>Cisco Transcode Device</td>
</tr>
<tr>
<td>15</td>
<td>Cisco SW Conference Bridge Device</td>
</tr>
<tr>
<td>16</td>
<td>Cisco HW Conference Bridge Device</td>
</tr>
<tr>
<td>17</td>
<td>Cisco Locations</td>
</tr>
<tr>
<td>18</td>
<td>Cisco Gatekeeper</td>
</tr>
<tr>
<td>19</td>
<td>Cisco CallManager System Performance</td>
</tr>
<tr>
<td>20</td>
<td>Cisco Video Conference Bridge Device</td>
</tr>
<tr>
<td>21</td>
<td>Cisco Hunt Lists</td>
</tr>
<tr>
<td>22</td>
<td>Cisco SIP</td>
</tr>
</tbody>
</table>
### Enum Definitions for DeviceType

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>CISCO_VGC_PHONE</td>
</tr>
<tr>
<td>11</td>
<td>CISCO_VGC_VIRTUAL_PHONE</td>
</tr>
<tr>
<td>12</td>
<td>CISCO_VGC_VIRTUAL_PHONE_WS-X6608</td>
</tr>
<tr>
<td>30</td>
<td>ANALOG_ACCESS</td>
</tr>
<tr>
<td>40</td>
<td>DIGITAL_ACCESS</td>
</tr>
<tr>
<td>42</td>
<td>DIGITAL_ACCESS+</td>
</tr>
<tr>
<td>43</td>
<td>DIGITAL_ACCESS_WS-X6608</td>
</tr>
<tr>
<td>47</td>
<td>ANALOG_ACCESS_WS-X6624</td>
</tr>
<tr>
<td>48</td>
<td>VGC_GATEWAY</td>
</tr>
<tr>
<td>50</td>
<td>CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>51</td>
<td>CONFERENCE_BRIDGE_HARDWARE</td>
</tr>
<tr>
<td>52</td>
<td>CONFERENCE_BRIDGE_HARDWARE_HDV2</td>
</tr>
<tr>
<td>53</td>
<td>CONFERENCE_BRIDGE_HARDWARE_WS-SVC-CMM</td>
</tr>
<tr>
<td>62</td>
<td>H323_GATEWAY</td>
</tr>
<tr>
<td>70</td>
<td>MUSIC_ON_HOLD</td>
</tr>
<tr>
<td>71</td>
<td>DEVICE_PILOT</td>
</tr>
<tr>
<td>73</td>
<td>CTI_ROUTE_POINT</td>
</tr>
<tr>
<td>80</td>
<td>VOICE_MAIL_PORT</td>
</tr>
<tr>
<td>83</td>
<td>SOFTWARE_MEDIA_TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>84</td>
<td>CISCO_MEDIA_SERVER</td>
</tr>
<tr>
<td>85</td>
<td>CISCO_VIDEO_CONFERENCE_BRIDGE</td>
</tr>
</tbody>
</table>

### Code Reason

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>Cisco Annunciator Device</td>
</tr>
<tr>
<td>24</td>
<td>Cisco QSIG Features</td>
</tr>
<tr>
<td>25</td>
<td>Cisco SIP Stack</td>
</tr>
<tr>
<td>26</td>
<td>Cisco Presence Features</td>
</tr>
<tr>
<td>27</td>
<td>Cisco WSMConnector</td>
</tr>
<tr>
<td>28</td>
<td>Cisco Dual-Mode Mobility</td>
</tr>
<tr>
<td>29</td>
<td>Cisco SIP Station</td>
</tr>
<tr>
<td>30</td>
<td>Cisco Mobility Manager</td>
</tr>
<tr>
<td>31</td>
<td>Cisco Signaling</td>
</tr>
<tr>
<td>32</td>
<td>Cisco Call Restriction</td>
</tr>
<tr>
<td>33</td>
<td>External Call Control</td>
</tr>
<tr>
<td>34</td>
<td>Cisco SAF Client</td>
</tr>
<tr>
<td>35</td>
<td>IME Client</td>
</tr>
<tr>
<td>36</td>
<td>IME Client Instance</td>
</tr>
</tbody>
</table>
### Enum Definitions for IPAddrAttributes

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown—The device has not indicated what this IPv4 address is used for.</td>
</tr>
<tr>
<td>1</td>
<td>Administrative only—The device has indicated that this IPv4 address is used for administrative communication (web interface) only.</td>
</tr>
<tr>
<td>2</td>
<td>Signal only—The device has indicated that this IPv4 address is used for control signaling only.</td>
</tr>
<tr>
<td>3</td>
<td>Administrative and signal—The device has indicated that this IPv4 address is used for administrative communication (web interface) and control signaling.</td>
</tr>
</tbody>
</table>
Enum Definitions for IPV6AddrAttributes

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Unknown—The device has not indicated what this IPv6 address is used for.</td>
</tr>
<tr>
<td>1</td>
<td>Administrative only—The device has indicated that this IPv6 address is used for administrative communication (web interface) only.</td>
</tr>
<tr>
<td>2</td>
<td>Signal only—The device has indicated that this IPv6 address is used for control signaling only.</td>
</tr>
<tr>
<td>3</td>
<td>Administrative and signal—The device has indicated that this IPv6 address is used for administrative communication (web interface) and control signaling.</td>
</tr>
</tbody>
</table>

Recommended Action
None

DeviceResetInitiated

Device reset initiated on the specified device.

This alarm occurs when a device is reset via the Reset button in Cisco Unified CM Administration. Reset may cause the device to shut down and come back in service. A device can be reset only when it is registered with Cisco Unified Communications Manager.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
</table>
| 8.0(1)                              | • Enum Definitions for DeviceType are updated.  
                                       | • Parameters added: Product type [String] |

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational (6)

Parameters
Device name [Optional] [String] Device type. [Optional] [Enum] Product type [String]

Enum Definitions for DeviceType

<table>
<thead>
<tr>
<th>Code</th>
<th>Device Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>CISCO_VGC_PHONE</td>
</tr>
<tr>
<td>11</td>
<td>CISCO_VGC_VIRTUAL_PHONE</td>
</tr>
<tr>
<td>30</td>
<td>ANALOG_ACCESS</td>
</tr>
<tr>
<td>40</td>
<td>DIGITAL_ACCESS</td>
</tr>
<tr>
<td>42</td>
<td>DIGITAL_ACCESS+</td>
</tr>
<tr>
<td>43</td>
<td>DIGITAL_ACCESS_WS-X6608</td>
</tr>
<tr>
<td>47</td>
<td>ANALOG_ACCESS_WS-X6624</td>
</tr>
<tr>
<td>48</td>
<td>VGC_GATEWAY</td>
</tr>
<tr>
<td>50</td>
<td>CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>51</td>
<td>CONFERENCE_BRIDGE_HARDWARE</td>
</tr>
<tr>
<td>52</td>
<td>CONFERENCE_BRIDGE_HARDWARE_HDV2</td>
</tr>
<tr>
<td>53</td>
<td>CONFERENCE_BRIDGE_HARDWARE_WS-SVC-CMM</td>
</tr>
<tr>
<td>62</td>
<td>H323_GATEWAY</td>
</tr>
<tr>
<td>70</td>
<td>MUSIC_ON_HOLD</td>
</tr>
<tr>
<td>71</td>
<td>DEVICE_PILOT</td>
</tr>
<tr>
<td>73</td>
<td>CTI_ROUTE_POINT</td>
</tr>
<tr>
<td>80</td>
<td>VOICE_MAIL_PORT</td>
</tr>
<tr>
<td>83</td>
<td>SOFTWARE_MEDIA_TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>84</td>
<td>CISCO_MEDIA_SERVER</td>
</tr>
<tr>
<td>85</td>
<td>CISCO_VIDEO_CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>90</td>
<td>ROUTE_LIST</td>
</tr>
<tr>
<td>100</td>
<td>LOAD_SIMULATOR</td>
</tr>
<tr>
<td>110</td>
<td>MEDIA_TERMINATION_POINT</td>
</tr>
<tr>
<td>111</td>
<td>MEDIA_TERMINATION_POINT_HARDWARE</td>
</tr>
<tr>
<td>112</td>
<td>MEDIA_TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>113</td>
<td>MEDIA_TERMINATION_POINT_WS-SVC-CMM</td>
</tr>
<tr>
<td>120</td>
<td>MGCP_STATION</td>
</tr>
<tr>
<td>121</td>
<td>MGCP_TRUNK</td>
</tr>
<tr>
<td>122</td>
<td>GATEKEEPER</td>
</tr>
<tr>
<td>124</td>
<td>7914_14_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>125</td>
<td>TRUNK</td>
</tr>
<tr>
<td>126</td>
<td>TONE_ANNOUNCEMENT_PLAYER</td>
</tr>
<tr>
<td>131</td>
<td>SIP_TRUNK</td>
</tr>
<tr>
<td>132</td>
<td>SIP_GATEWAY</td>
</tr>
<tr>
<td>133</td>
<td>WSM_TRUNK</td>
</tr>
<tr>
<td>134</td>
<td>REMOTE_DESTINATION_PROFILE</td>
</tr>
<tr>
<td>227</td>
<td>7915_12_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>228</td>
<td>7915_24_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>229</td>
<td>7916_12_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>230</td>
<td>7916_24_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
</tbody>
</table>
DeviceRestartInitiated

Device restart initiated or Apply Config initiated on the specified device.

This alarm occurs when a device is restarted via the Restart button in Cisco Unified CM Administration window or when a system administrator presses the Apply Config button for a device that does not support conditional restart. Restart causes the device to unregister, receive updated configuration, and reregister with Cisco Unified Communications Manager (Unified CM) without shutting down. A device can be restarted only when it is registered with Unified CM.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>• Enum Definitions for DeviceType are updated.</td>
</tr>
<tr>
<td></td>
<td>• Parameters added: Product type [String]</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational (6)

Parameters
Device name [Optional]. [String] Device type. [Optional] [Enum] Product type [String]

Enum Definitions for DeviceType

<table>
<thead>
<tr>
<th>Code</th>
<th>Device Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>CISCO_VGC_PHONE</td>
</tr>
<tr>
<td>11</td>
<td>CISCO_VGC_VIRTUAL_PHONE</td>
</tr>
<tr>
<td>30</td>
<td>ANALOG_ACCESS</td>
</tr>
<tr>
<td></td>
<td>Alarm Name</td>
</tr>
<tr>
<td>---</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>40</td>
<td>DIGITAL_ACCESS</td>
</tr>
<tr>
<td>42</td>
<td>DIGITAL_ACCESS+</td>
</tr>
<tr>
<td>43</td>
<td>DIGITAL_ACCESS_WS-X6608</td>
</tr>
<tr>
<td>47</td>
<td>ANALOG_ACCESS_WS-X6624</td>
</tr>
<tr>
<td>48</td>
<td>VGC_GATEWAY</td>
</tr>
<tr>
<td>50</td>
<td>CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>51</td>
<td>CONFERENCE_BRIDGE_HARDWARE</td>
</tr>
<tr>
<td>52</td>
<td>CONFERENCE_BRIDGE_HARDWARE_HDV2</td>
</tr>
<tr>
<td>53</td>
<td>CONFERENCE_BRIDGE_HARDWARE_WS-SVC-CMM</td>
</tr>
<tr>
<td>62</td>
<td>H323_GATEWAY</td>
</tr>
<tr>
<td>70</td>
<td>MUSIC_ON_HOLD</td>
</tr>
<tr>
<td>71</td>
<td>DEVICE_PILOT</td>
</tr>
<tr>
<td>73</td>
<td>CTI_ROUTE_POINT</td>
</tr>
<tr>
<td>80</td>
<td>VOICE_MAIL_PORT</td>
</tr>
<tr>
<td>83</td>
<td>SOFTWARE_MEDIA_TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>84</td>
<td>CISCO_MEDIA_SERVER</td>
</tr>
<tr>
<td>85</td>
<td>CISCO_VIDEO_CONFERENCE_BRIDGE</td>
</tr>
<tr>
<td>90</td>
<td>ROUTE_LIST</td>
</tr>
<tr>
<td>100</td>
<td>LOAD_SIMULATOR</td>
</tr>
<tr>
<td>110</td>
<td>MEDIA_TERMINATION_POINT</td>
</tr>
<tr>
<td>111</td>
<td>MEDIA_TERMINATION_POINT_HARDWARE</td>
</tr>
<tr>
<td>112</td>
<td>MEDIA_TERMINATION_POINT_HDV2</td>
</tr>
<tr>
<td>113</td>
<td>MEDIA_TERMINATION_POINT_WS-SVC-CMM</td>
</tr>
<tr>
<td>120</td>
<td>MGCP_STATION</td>
</tr>
<tr>
<td>121</td>
<td>MGCP_TRUNK</td>
</tr>
<tr>
<td>122</td>
<td>GATEKEEPER</td>
</tr>
<tr>
<td>124</td>
<td>7914_14_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>125</td>
<td>TRUNK</td>
</tr>
<tr>
<td>126</td>
<td>TONE_ANNOUNCEMENT_PLAYER</td>
</tr>
<tr>
<td>131</td>
<td>SIP_TRUNK</td>
</tr>
<tr>
<td>132</td>
<td>SIP_GATEWAY</td>
</tr>
<tr>
<td>133</td>
<td>WSM_TRUNK</td>
</tr>
<tr>
<td>134</td>
<td>REMOTE_DESTINATION_PROFILE</td>
</tr>
<tr>
<td>227</td>
<td>7915_12_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>228</td>
<td>7915_24_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>229</td>
<td>7916_12_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>230</td>
<td>7916_24_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
<tr>
<td>232</td>
<td>CKEM_36_BUTTON_LINE_EXPANSION_MODULE</td>
</tr>
</tbody>
</table>
Informational-Level Alarms

<table>
<thead>
<tr>
<th>ScheduleID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>254</td>
<td>UNKNOWMGCP_GATEWAY</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>30027</td>
<td>ANALOG_PHONE</td>
</tr>
<tr>
<td>30028</td>
<td>ISDN_BRI_PHONE</td>
</tr>
<tr>
<td>30032</td>
<td>SCCP_GATEWAY_VIRTUAL_PHONE</td>
</tr>
</tbody>
</table>

**Recommended Action**

None

**DirSyncScheduleInsertFailed**

DirSync schedule insertion failed.

**Facility/Sub-Facility**

CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog

System/Java Applications

**Severity**

Informational (6)

**Parameters**

ScheduleID [String]

**Recommended Action**

Check the DirSync configuration and logs

**DirSyncSchedulerEngineStarted**

DirSync scheduler engine started.

**Facility/Sub-Facility**

CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog

System/Java Applications

**Severity**

Informational (6)

**Parameters**

DirSyncSchedulerVersion [String]
**Recommended Action**
None

**DRFBackupCompleted**

DRF backup completed successfully.

**Cisco Unified Serviceability Alarm Definition Catalog**
System/DRF

**Severity**
INFORMATIONAL

**Routing List**
Event Log
Sys Log

**Parameter(s)**
Reason(String)

**Recommended Action**
Ensure that the backup operation is completed successfully.

**DRFRestoreCompleted**

DRF restore completed successfully.

**Cisco Unified Serviceability Alarm Definition Catalog**
System/DRF

**Severity**
INFORMATIONAL

**Routing List**
Event Log
Sys Log

**Parameter(s)**
Reason(String)

**Recommended Action**
Ensure that the restore operation is completed successfully.
Chapter 6      Cisco Unified Serviceability Alarms and CiscoLog Messages

### DRFSchedulerUpdated

DRF Scheduled backup configurations is updated automatically due to feature de-registration.

<table>
<thead>
<tr>
<th>Facility/Sub-Facility</th>
<th>CCM_DRF_LOCAL &amp; CCM_DRF_MASTER/DRF</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cisco Unified Serviceability Alarm Definition Catalog</strong></td>
<td>System/DRF</td>
</tr>
<tr>
<td><strong>Severity</strong></td>
<td>INFORMATIONAL</td>
</tr>
<tr>
<td><strong>Parameters</strong></td>
<td>Reason(String)</td>
</tr>
<tr>
<td><strong>Recommended Action</strong></td>
<td>Ensure that the new configurations is appropriate one for the backup/restore operation.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>History</strong></th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Unified Communications Manager Release</td>
<td>8.0(1) Name changed from CiscoDRFSchedulerUpdated.</td>
</tr>
</tbody>
</table>

### EMAppStarted

EM Application started successfully.

**Cisco Unified Serviceability Alarm Definition Catalog**

System/EMAlarmCatalog

**Severity**

INFORMATIONAL

**Routing List**

Sys Log
Event Log

**Parameter(s)**

Servlet Name(String)

**Recommended Action**

No action required.
EMCCUserLoggedIn

EMCC login was successful.

Cisco Unified Serviceability Alarm Definition Catalog
System/EMAlarmCatalog

Severity
Informational(6)

Routing List
Sys Log
Event Log

Parameters
Device Name(String)
Login Date/Time(String)
Login UserID(String)

Recommended Action
None

EMCCUserLoggedOut

EMCC logout was successful.

Cisco Unified Serviceability Alarm Definition Catalog
System/EMAlarmCatalog

Severity
Informational(6)

Routing List
Sys Log
Event Log

Parameters
Device Name(String)
Login Date/Time(String)
UserID(String)

Recommended Action
None
**EndPointResetInitiated**

This alarm occurs when a device is reset via the Reset button in Cisco Unified CM Administration. Reset causes the device to shut down and come back in service. A device can be reset only when it is registered with Cisco Unified Communications Manager.

**Cisco Unified Serviceability Alarm Definition Catalog**

*CallManager/CallManager*

**Severity**

INFORMATIONAL

**Routing List**

SDL
SDI
Sys Log
Alternate Syslog

**Parameter(s)**

Device name(String)
Product type(String)
Device type(Enum)

**Enum Definitions -Device type**

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CISCO_30SP+</td>
</tr>
<tr>
<td>2</td>
<td>CISCO_12SP+</td>
</tr>
<tr>
<td>3</td>
<td>CISCO_12SP</td>
</tr>
<tr>
<td>4</td>
<td>CISCO_12S</td>
</tr>
<tr>
<td>5</td>
<td>CISCO_30VIP</td>
</tr>
<tr>
<td>6</td>
<td>CISCO_7910</td>
</tr>
<tr>
<td>7</td>
<td>CISCO_7960</td>
</tr>
<tr>
<td>8</td>
<td>CISCO_7940</td>
</tr>
<tr>
<td>9</td>
<td>CISCO_7935</td>
</tr>
<tr>
<td>12</td>
<td>CISCO_ATA_186</td>
</tr>
<tr>
<td>20</td>
<td>SCCP_PHONE</td>
</tr>
<tr>
<td>61</td>
<td>H323_PHONE</td>
</tr>
<tr>
<td>72 CTI_</td>
<td>PORT</td>
</tr>
<tr>
<td>115 CISCO_</td>
<td>7941</td>
</tr>
<tr>
<td>119 CISCO_</td>
<td>7971</td>
</tr>
<tr>
<td>255 UNK</td>
<td>NOWN</td>
</tr>
<tr>
<td>302 CISCO_</td>
<td>7989</td>
</tr>
<tr>
<td>Value</td>
<td>Definition</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>307 CISCO_</td>
<td>7911</td>
</tr>
<tr>
<td>308 CISCO_</td>
<td>7941G_GE</td>
</tr>
<tr>
<td>309 CISCO_</td>
<td>7961G_GE</td>
</tr>
<tr>
<td>335 MO</td>
<td>TOROLA_CN622</td>
</tr>
<tr>
<td>336</td>
<td>BASIC_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>348 CISCO_</td>
<td>7931</td>
</tr>
<tr>
<td>358 CISCO_</td>
<td>UNIFIED_COMMUNICATOR</td>
</tr>
<tr>
<td>365 CISCO_</td>
<td>7921</td>
</tr>
<tr>
<td>369 CISCO_</td>
<td>7906</td>
</tr>
<tr>
<td>374 ADV</td>
<td>ANCED_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>375 CISCO_TEL</td>
<td>EPRESENCE</td>
</tr>
<tr>
<td>404 CISCO_</td>
<td>7962</td>
</tr>
<tr>
<td>412 CISCO_</td>
<td>3951</td>
</tr>
<tr>
<td>431 CISCO_</td>
<td>7937</td>
</tr>
<tr>
<td>434 CISCO_</td>
<td>7942</td>
</tr>
<tr>
<td>435 CISCO_</td>
<td>7945</td>
</tr>
<tr>
<td>436 CISCO_</td>
<td>7965</td>
</tr>
<tr>
<td>437 CISCO_</td>
<td>7975</td>
</tr>
<tr>
<td>446 CISCO_</td>
<td>3911</td>
</tr>
<tr>
<td>468</td>
<td>CISCO_UNIFIED_MOBILE_COMMUNICATOR</td>
</tr>
<tr>
<td>478 CISCO_</td>
<td>TELEPRESENCE_1000</td>
</tr>
<tr>
<td>479 CISCO_</td>
<td>TELEPRESENCE_3000</td>
</tr>
<tr>
<td>480 CISCO_</td>
<td>TELEPRESENCE_3200</td>
</tr>
<tr>
<td>481 CISCO_</td>
<td>TELEPRESENCE_500</td>
</tr>
<tr>
<td>484 CISCO_</td>
<td>7925</td>
</tr>
<tr>
<td>493 CISCO_</td>
<td>9971</td>
</tr>
<tr>
<td>495 CISCO_</td>
<td>6921</td>
</tr>
<tr>
<td>496 CISCO_</td>
<td>6941</td>
</tr>
<tr>
<td>497 CISCO_</td>
<td>6961</td>
</tr>
<tr>
<td>20000 CISCO_</td>
<td>7905</td>
</tr>
<tr>
<td>30002 CISCO_</td>
<td>7920</td>
</tr>
<tr>
<td>30006 CISCO_</td>
<td>7970</td>
</tr>
<tr>
<td>30007 CISCO_</td>
<td>7912</td>
</tr>
<tr>
<td>30008 CISCO_</td>
<td>7902</td>
</tr>
<tr>
<td>30016 CISCO_</td>
<td>IP_COMMUNICATOR</td>
</tr>
<tr>
<td>30018 CISCO_</td>
<td>7961</td>
</tr>
</tbody>
</table>
Informational-Level Alarms

Recommended Action
Informational purposes only; no action is required.

EndPointRestartInitiated

Device restart initiated or Apply Config initiated on the specified device.

This alarm occurs when a device is restarted via the Restart button in Cisco Unified CM Administration window or when a system administrator presses the Apply Config button for a device that does not support conditional restart. Restart causes the device to unregister, receive an updated configuration file, and reregister with Cisco Unified Communications Manager (Unified CM) without shutting down. A device can be restarted only when it is registered with Unified CM.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
INFORMATIONAL

Routing List
SDL
SDI
Sys Log
Alternate Syslog

Parameter(s)
Device name(String)
Product type(String)
Device type(Enum)

Enum Definitions -Device type

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>30019</td>
<td>CISCO_7936</td>
</tr>
<tr>
<td>30035</td>
<td>P_STE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>30019</td>
<td>CISCO_30SP+</td>
</tr>
<tr>
<td>30035</td>
<td>CISCO_12SP+</td>
</tr>
<tr>
<td>30035</td>
<td>CISCO_12SP</td>
</tr>
<tr>
<td>30035</td>
<td>CISCO_12S</td>
</tr>
<tr>
<td>30035</td>
<td>CISCO_30VIP</td>
</tr>
<tr>
<td>30035</td>
<td>CISCO_7910</td>
</tr>
<tr>
<td>30035</td>
<td>CISCO_7960</td>
</tr>
</tbody>
</table>
### Informational-Level Alarms

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>CISCO_7940</td>
</tr>
<tr>
<td>9</td>
<td>CISCO_7935</td>
</tr>
<tr>
<td>12</td>
<td>CISCO_ATA_186</td>
</tr>
<tr>
<td>20</td>
<td>SCCP_PHONE</td>
</tr>
<tr>
<td>61</td>
<td>H323_PHONE</td>
</tr>
<tr>
<td>72</td>
<td>CTI_PORT</td>
</tr>
<tr>
<td>115</td>
<td>CISCO_7941</td>
</tr>
<tr>
<td>119</td>
<td>CISCO_7971</td>
</tr>
<tr>
<td>255</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>302</td>
<td>CISCO_7989</td>
</tr>
<tr>
<td>307</td>
<td>CISCO_7911</td>
</tr>
<tr>
<td>308</td>
<td>CISCO_7941G_GE</td>
</tr>
<tr>
<td>309</td>
<td>CISCO_7961G_GE</td>
</tr>
<tr>
<td>335</td>
<td>MOTOROLA_CN622</td>
</tr>
<tr>
<td>336</td>
<td>BASIC_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>348</td>
<td>CISCO_7931</td>
</tr>
<tr>
<td>358</td>
<td>CISCO_UNIFIED_COMMUNICATOR</td>
</tr>
<tr>
<td>365</td>
<td>CISCO_7921</td>
</tr>
<tr>
<td>369</td>
<td>CISCO_7906</td>
</tr>
<tr>
<td>374</td>
<td>ADVANCED_3RD_PARTY_SIP_DEVICE</td>
</tr>
<tr>
<td>375</td>
<td>CISCO_TELEPRESENCE</td>
</tr>
<tr>
<td>404</td>
<td>CISCO_7962</td>
</tr>
<tr>
<td>412</td>
<td>CISCO_3951</td>
</tr>
<tr>
<td>431</td>
<td>CISCO_7937</td>
</tr>
<tr>
<td>434</td>
<td>CISCO_7942</td>
</tr>
<tr>
<td>435</td>
<td>CISCO_7945</td>
</tr>
<tr>
<td>436</td>
<td>CISCO_7965</td>
</tr>
<tr>
<td>437</td>
<td>CISCO_7975</td>
</tr>
<tr>
<td>446</td>
<td>CISCO_3911</td>
</tr>
<tr>
<td>468</td>
<td>CISCO_UNIFIED_MOBILE_COMMUNICATOR</td>
</tr>
<tr>
<td>478</td>
<td>CISCO_TELEPRESENCE_1000</td>
</tr>
<tr>
<td>479</td>
<td>CISCO_TELEPRESENCE_3000</td>
</tr>
<tr>
<td>480</td>
<td>CISCO_TELEPRESENCE_3200</td>
</tr>
<tr>
<td>481</td>
<td>CISCO_TELEPRESENCE_500</td>
</tr>
<tr>
<td>484</td>
<td>CISCO_7925</td>
</tr>
<tr>
<td>493</td>
<td>CISCO_9971</td>
</tr>
<tr>
<td>495</td>
<td>CISCO_6921</td>
</tr>
<tr>
<td>496</td>
<td>CISCO_6941</td>
</tr>
</tbody>
</table>
EndThrottlingCallListBLFSubscriptions

CallManager has resumed accepting CallList BLF Subscriptions subsequent to prior throttling.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Severity changed from Warning to Informational.</td>
</tr>
</tbody>
</table>

Cisco Unified Serviceability Alarm Definition Catalog

CallManager/CallManager

Severity

Informational

Parameters

EndThrottlingCallListBLFSubscriptionsActive External Presence Subscriptions [UInt] CallList BLF Subscriptions Throttling Threshold [UInt] CallList BLF Subscriptions Resume Threshold [UInt] Time Duration Of Throttling CallList BLF Subscriptions [UInt] Number of CallList BLF Subscriptions Rejected Due To Throttling [UInt] Total End Throttling CallList BLF Subscriptions [UInt]

Recommended Action

Determine if CPU and memory resources are available to meet the higher demand for CallList BLF Subscriptions. If so, increase the CallListBLFSubscriptionsThrottlingThreshold and correspondingly the CallListBLFSubscriptionsResumeThreshold. If not, increase system resources to meet the demand.
### IDSEngineDebug

Indicates debug events from IDS database engine. This alarm provides low-level debugging information from IDS database engine. System administrator can disregard this alarm.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Changed severity level to Informational from Debug.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_DB_LAYER-DB

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DB

**Severity**

Informational

**Parameters**

- Event Class ID [String]
- Event class message [String]
- Event Specific Message [String]

**Recommended Action**

None

### IDSEngineInformation

No error has occurred but some routine event completed in IDS database engine. This alarm is informational. No error has occurred but some routine event completed in IDS database engine.

**Facility/Sub-Facility**

CCM_DB_LAYER-DB

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DB

**Severity**

Informational (6)

**Parameters**

- Event Class ID [String]
- Event class message [String]
- Event Specific Message [String]

**Recommended Action**

None
IDSReplicationInformation

Information about IDS replication.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Added Recommended Action comments.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

DB

**Cisco Unified Serviceability Alarm Definition Catalog**

System/DB

**Severity**

Informational (6)

**Parameters**

- Event Class ID [String]
- Event class message [String]
- Event Specific Message [String]

**Recommended Action**

Information only. No action is required.

IPMAInformation

IPMA Information.

**Facility/Sub-Facility**

CCM_JAVA_APPS-TOMCATAPPLICATIONS

**Cisco Unified Serviceability Alarm Definition Catalog**

System/Java Applications

**Severity**

Informational (6)

**Parameters**

- Servlet Name [String]
- Reason [String]

**Recommended Action**

None
IPMAStarted

IPMA Application started successfully.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Informational (6)

Parameters
Servlet Name [String] Reason [String]

Recommended Action
None

ITLFileRegenerated

New ITL File has been generated. This usually means that a new certificate related to ITLFile has been modified.

Cisco Unified Serviceability Alarm Catalog
System/TVS

Severity
INFORMATIONAL

Routing List
SDI
Event Log
Data Collector
Sys Log

Recommended Action
None.

kANNICMPErrorNotification

ANN stream ICMP port unreachable error. An announcement RTP stream had an ICMP (Internet Control Message Protocol) port unreachable error. The stream has been terminated. This ICMP error is a result of the destination end-point not having the receiving UDP/RTP port open to receive packets.
Informational-Level Alarms

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Informational (6)

Parameters
Destination IP Address [String]

Recommended Action
No action is required. This may occur at times when connections are being stopped or redirected.

kCFBICMPErrorNotification

CFB stream ICMP error. A SW CFB RTP stream had an ICMP (Internet Control Message Protocol) port unreachable error. The stream has been terminated. This ICMP error is a result of the destination end-point does not have the receiving UDP/RTP port open to receive packets.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Following parameters removed: Call ID [ULong] Party ID [ULong] IP Port [ULong]</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Informational (6)
Parameters
Destination IP Address [String]

Recommended Action
No action is required. This may occur at times when connections are being stopped or redirected.

kReadCfgIpTosMediaResourceToCmNotFound

IP TOS MediaResource to Cm value not found. The IP Type-of-Service Media Resource To Call Manager service parameter value was not found in the database. Defaulting its value to 0x60 for CS3(precedence 3) DSCP (011000).

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Informational (6)

Recommended Action
Set the Ip Type-of-Service Media Resource To Call Manager service parameter for the Cisco IP Voice Media Streaming App service.

kDeviceMgrLockoutWithCallManager

Cisco Unified Communications Manager in lockout. The specified Cisco Unified Communications Manager has failed to respond to control messages. The TCP control connection to Cisco Unified CM is being suspended. This will cause a switch to another Cisco Unified CM if one is available otherwise the device will be unavailable. There may be a shortage of CPU resource or some other error condition on the Cisco Unified CM server.
### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Severity changed from Error to Informational.</td>
</tr>
</tbody>
</table>

#### Facility/Sub-Facility

CCM_MEDIA_STREAMING_APP-IPVMS

#### Cisco Unified Serviceability Alarm Definition Catalog

CallManager/IpVms

#### Severity

Informational

#### Parameters

Trace Name [String]

#### Recommended Action

Check the status of the Cisco Unified Communications Manager service. You may have to restart the Cisco Unified CM service or the Cisco Unified CM server.

### kDeviceMgrRegisterWithCallManager

Register with Cisco Unified Communications Manager. The software media device registered with the specified Cisco Unified Communications Manager.

#### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
</tbody>
</table>

#### Facility/Sub-Facility

CCM_MEDIA_STREAMING_APP-IPVMS

#### Cisco Unified Serviceability Alarm Definition Catalog

CallManager/IpVms
**Severity**
Informational (6)

**Parameters**
Trace Name [String]

**Recommended Action**
None

### kDeviceMgrThreadWaitFailed

Wait call failure in device manager thread. An error was reported during a system request to wait on an event, the media device will be restarted.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
<tr>
<td></td>
<td>• Severity changed from Error to Informational.</td>
</tr>
<tr>
<td></td>
<td>• Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>– OS Error Code [Int]</td>
</tr>
<tr>
<td></td>
<td>– OS Error Description [String]</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**
CallManager/IpVms

**Severity**
Informational

**Parameters**
Trace Name [String]
OS Error Code [Int]
OS Error Description [String]

**Recommended Action**
None
**kDeviceMgrUnregisterWithCallManager**

Unregister with Cisco Unified Communications Manager. A media device has unregistered with the specified Cisco Unified Communications Manager.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/IpVms

**Severity**

Informational (6)

**Parameters**

Trace Name [String]

**Recommended Action**

No action is required. The media device will automatically reregister.

**kIPVMSStarting**

The Cisco IP Voice Media Streaming App service is starting.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
<tr>
<td></td>
<td>ProcessID [ULong] parameter is removed.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**

CCM_MEDIA_STREAMING_APP-IPVMS
Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Informational (6)

Parameters

Recommended Action
No action is required.

kIPVMSStopping

The Cisco IP voice media streaming application is shutting down.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
<tr>
<td></td>
<td>ProcessID [ULONG] parameter is removed.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Informational (6)

Parameters

Recommended Action
No action is required.

kMOHICMPErrorNotification

MOH stream ICMP error. A Music-on-Hold transmission stream had an ICMP (Internet Control Message Protocol) port unreachable error. The stream has been terminated. This may occur occasionally depending on call termination sequences.
### Informational-Level Alarms

#### Facility/Sub-Facility

**CCM_MEDIA_STREAMING_APP-IPVMS**

#### Cisco Unified Serviceability Alarm Definition Catalog

**CallManager/IpVms**

**Severity**

Informational (6)

**Parameters**

Destination IP Address [String]

**Recommended Action**

No action is required.

---

**kMOHMgrThreadWaitFailed**

Wait call failure in MOH manager thread. An error was encountered in Music-on-Hold audio manager subcomponent while waiting for asynchronous event signaling. The MOH device will be restarted.

#### History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). Following parameters are removed: Call ID [ULong] Party ID [ULong] IP Port [ULong]</td>
</tr>
</tbody>
</table>

#### Facility/Sub-Facility

**CCM_MEDIA_STREAMING_APP-IPVMS**

- Severity changed from Error to Informational.
- OS Error Description(String) parameter is added.
Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Informational

Parameter(s)
OS Error Description(String)

Recommended Action
No action is required.

kMOHMgrIsAudioSourceInUseThisIsNULL

Synchronization error detected in MOH audio manager. A synchronization error was detected. Condition has been resolved automatically.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Informational (6)

Recommended Action
No action is required.

kMOHRewindStreamControlNull

Attempted to rewind an inactive MOH audio source. An attempt was made to rewind or restart the Music-on-Hold audio source that is inactive. This has been ignored.
### Informational-Level Alarms

#### Facility/Sub-Facility

<table>
<thead>
<tr>
<th>Facility/Sub-Facility</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCM_MEDIA_STREAMING_APP-IPVMS</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>CallManager/IpVms</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>Cisco Unified Communications Alarm Definition Catalog</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
<tr>
<td></td>
<td>• Severity changed from Error to Informational.</td>
</tr>
<tr>
<td></td>
<td>• Audio Source ID [ULong] parameter is removed.</td>
</tr>
</tbody>
</table>

#### Codec Type [String]

**Recommended Action**

None

---

#### kMOHRewindStreamMediaPositionObjectNull

Error rewinding MOH audio source that is not playing. An attempt was made to rewind or restart a Music-on-Hold wav file that was not being played. This has been ignored.

#### Facility/Sub-Facility

<table>
<thead>
<tr>
<th>Facility/Sub-Facility</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCM_MEDIA_STREAMING_APP-IPVMS</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>CallManager/IpVms</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>Cisco Unified Communications Alarm Definition Catalog</td>
<td>This alarm is available in 8.0(1).</td>
</tr>
<tr>
<td></td>
<td>• Severity changed from Error to Informational.</td>
</tr>
<tr>
<td></td>
<td>• Audio Source ID [ULong] parameter is removed.</td>
</tr>
</tbody>
</table>
Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Informational

Parameters
Codec Type [String]

Recommended Action
None

kMTPDeviceStartingDefaults

One or more Cisco IP Voice Media Streaming App service parameter settings for the MTP device were not found in the database. The default values are included here.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.x and 4.x</td>
<td>Added for Windows.</td>
</tr>
<tr>
<td>7.0(1)</td>
<td>Obsoleted.</td>
</tr>
<tr>
<td>8.0(1)</td>
<td>This alarm is available in 8.0(1). MTP Run Flag(String) parameter is added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Informational (6)

Parameter(s)
MTP Run Flag(String)

Recommended Action
Configure the service parameter settings for the MTP device.

kReadCfgMOHEnabledCodecsNotFound

MOH enabled codecs not found. The Music-on-Hold service parameter for codec selection could not be read from database. Defaulting to G.711 mu-law codec.
Facility/Sub-Facility
CCM_MEDIA_STREAMING_APP-IPVMS

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
Informational (6)

Recommended Action
Set the Music-on-Hold service parameter for Cisco IP Voice Media Streaming App service.

LoadShareDeActivateTimeout

There was timeout during wait for DeActivateLoadShare acknowledgement.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Informational (6)

Recommended Action
None

LogFileSearchStringFound

The search string has been found in the log file. Trace and Log Central has found the search string that the user has configured.

Facility/Sub-Facility
CCM_TCT-LPMTCT

Cisco Unified Serviceability Alarm Definition Catalog
System/LpmTct

Severity
Informational (6)

Parameters
SearchString [String]
Recommended Action
If sysadmin is interested in collecting the traces around the time of generation of alert, use Trace and Log Central to collect the traces for that service.

MaxHoldDurationTimeout

A held call was cleared because the amount of time specified in the Maximum Hold Duration Timer service parameter had elapsed. If the allowed call-on-hold duration is too short, you can increase the value. If you do not want a limit on the duration of a held call, you can disable the limit.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>Following parameters added:</td>
</tr>
<tr>
<td></td>
<td>• Originating Device Name(String)</td>
</tr>
<tr>
<td></td>
<td>• Destination Device Name(String)</td>
</tr>
<tr>
<td></td>
<td>• Hold start time(UInt)</td>
</tr>
<tr>
<td></td>
<td>• Hold stop time(UInt)</td>
</tr>
<tr>
<td></td>
<td>• Calling Party Number(String)</td>
</tr>
<tr>
<td></td>
<td>• Called Party Number(String)</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility

CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational (6)

Parameters
Maximum Hold Duration (minutes) [Int]
Originating Device Name(String)
Destination Device Name(String)
Hold start time(UInt)
Hold stop time(UInt)
Calling Party Number(String)
Called Party Number(String)
Recommended Action
If the duration of the hold time is too short, increase the value in the Cisco CallManager service parameter or disable the maximum duration by setting the Maximum Hold Duration Timer parameter to zero.

PermissionDenied
An operation could not be completed because the process did not have authority to perform it.

Cisco Unified Serviceability Alarm Definition Catalog
System/Generic
Severity
Informational (6)
Parameters
None
Recommended Action
None

PktCapServiceStarted
Packet capture service started. Packet capture feature has been enabled on the Cisco Unified Communications Manager server. A Cisco CallManager service parameter, Packet Capture Enable, must be set to True for packet capture to occur.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager
Severity
Informational (6)
Recommended Action
None

PktCapServiceStopped
Packet capture service stopped. The packet capture feature has been disabled on the Cisco Unified Communications Manager server.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER
Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational (6)

Recommended Action
None

PktCapOnDeviceStarted
Packet capture started on the device. Indicated packet capture has been enabled on the device.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational (6)

Parameters
Device Name [String] Packet Capture Mode [String] Packet Capture Duration [String]

Recommended Action
None

PktCapOnDeviceStopped
Packet capture stopped on the device. Indicated packet capture has been disabled on the device.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational (6)

Parameters
Device Name [String] Packet Capture Mode [String] Packet Capture Duration [String]

Recommended Action
None
PublicationRunCompleted

Completion of publication of published DID patterns.
This alarm is generated when Unified CM completes a publication of the DID patterns into the IME
network.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
INFORMATIONAL

Recommended Action
This alarm is provided for historic and informational purposes. It can be used to give you feedback that
the system is working and is correctly publishing numbers into the IME network. It can also be used for
troubleshooting. If some of the publishes fail for some reason, the alarm will contain a list of those
numbers which were not published. If your users are receiving calls, and they are not over IP but you
think they ought to be, you can check the history of these alarms to see if the number failed to be
published into the network.

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Start time(String)
End time(String)
DID count(UInt)
Failed DID count(UInt)
Failed DIDs(String)

RedirectCallRequestFailed

CTIManager is unable to redirect a call

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CtiManager

Severity
INFORMATIONAL

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Directory Number(String)
Partition(String)

Recommended Action
This alarm is for informational purposes only; no action is required.

**RollBackToPre8.0Disabled**

Roll Back to Pre 8.0 has been disabled in the Enterprise Parameter page. This usually means that the RollBack to Pre 8.0 feature is modified in the Enterprise Parameter page.

Cisco Unified Serviceability Alarm Catalog
System/TVS
Severity
INFORMATIONAL

Routing List
SDI
Event Log
Data Collector
Sys Log

Recommended Action
None.

**RollBackToPre8.0Enabled**

Roll Back to Pre 8.0 has been enabled in the Enterprise Parameter page.

Cisco Unified Serviceability Alarm Catalog
System/TVS
Severity
INFORMATIONAL

Routing List
SDI
Event Log
Data Collector
Sys Log
**Recommended Action**

None.

**RouteRemoved**

Route removed automatically.

This alarm is generated when UC Manager removes a route from its routing tables because the route is stale and has expired, or because the far end has indicated the number is no longer reachable at that domain.

**Cisco Unified Serviceability Alarm Definition Catalog**

CallManager/CallManager

**Severity**

INFORMATIONAL

**Routing List**

SDL
SDI
Sys Log
Event Log

**Parameter(s)**

E.164 number(String)
Domain name(String)
Route learned time(String)
Reason Code(Enum)

**Enum Definitions - Reason Code**

<table>
<thead>
<tr>
<th>Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Expired</td>
</tr>
<tr>
<td>2</td>
<td>Unreachable</td>
</tr>
</tbody>
</table>

**Recommended Action**

This alarm is provided for historic and informational purposes. It helps you understand why certain numbers are in your routing tables, and why others are not. This historical information is useful to help determine why a call to a particular number is not going over IP, when you expect it to.

**SAFPublishRevoke**

A CLI command revoked the publish action for the specified service or subservice ID.

A system administrator issued a CLI command on the SAF Forwarder router to revoke the publish action for the service or subservice ID specified in this alarm.
Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
INFORMATIONAL

Routing List
SDL
SDI
Sys Log
Event Log

Parameter(s)
Client Handle(String)
Service ID(UInt)
Sub Service ID(UInt)
InstanceId1(UInt)
InstanceId2(UInt)
InstanceId3(UInt)
InstanceId4(UInt)

Recommended Action
Informational purposes only; no action is required.

SAFUnknownService

Unified CM does not recognize the service ID in a publish revoke or withdraw message.
Unified CM received a Publish Revoke message or Withdraw message from the SAF Forwarder but the service ID in the message is not recognized by Unified CM. Unified CM may not recognize the service ID if the service ID was mistyped in the publish revoke CLI command, or if the service was previously withdrawn.

Cisco Unified Serviceability Alarm Catalog
CallManager/CallManager

Severity
Informational(6)

Routing List
SDL
SDI
Sys Log
Event Log
Parameters
Client Handle(String)
Service ID(UInt)
Sub Service ID(UInt)
InstanceID1(UInt)
InstanceID2(UInt)
InstanceID3(UInt)
InstanceID4(UInt)

Recommended Action
None

SecurityEvent
Failed to write into the primary file path. Audit Event is generated by this application.

Cisco Unified Serviceability Alarm Catalog
AuditLog

Severity
INFORMATIONAL

Recommended Action
Ensure that the primary file path is valid and the corresponding drive has sufficient disk space. Also, make sure that the path has security permissions similar to default log file path.

ServiceActivated
This service is now activated.

Facility/Sub-Facility
CCM_SERVICEMANAGER-GENERIC

Cisco Unified Serviceability Alarm Definition Catalog
System/Service Manager

Severity
Informational (6)

Parameters
Service Name(String)

Recommended Action
None
ServiceDeactivated

The service is now deactivated.

**Facility/Sub-Facility**
CCM_SERVICEMANAGER GENERIC

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Service Manager

**Severity**
Informational (6)

**Parameters**
Service Name(String)

**Recommended Action**
None

ServiceStarted

Service has started.

**History**

<table>
<thead>
<tr>
<th>Cisco Unified Communications Manager Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>Added IPv6Address[Optional][String] parameter.</td>
</tr>
</tbody>
</table>

**Facility/Sub-Facility**
CCM_CBB GENERIC

**Cisco Unified Serviceability Alarm Definition Catalog**
System/Generic

**Severity**
Informational (6)

**Parameters**
IP Address of hosting node(String)
IPV6Address[Optional][String]
Host name of hosting node(String)
Service Name(String)
Version Information(String)
Informational-Level Alarms

Recommended Action
None

ServiceStopped

Service stopped.

Cisco Unified Serviceability Alarm Definition Catalog
System/Generic

Severity
Informational (6)

Parameters
IP Address of hosting node.(String)
Host of hosting node.(String)
Service (String)

Recommended Action
None

SoftwareLicenseValid

A valid software license has been detected by the IP Voice Media Streaming App service.

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/IpVms

Severity
INFORMATIONAL

Routing List
SDI
Event Log

Recommended Action
No action required. This informational message indicates alarm SoftwareLicenseNotValid is cleared.

StationAlarm

A station device sent an alarm to Cisco Unified Communications Manager, which acts as a conduit from the device to generate this alarm.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER
Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational (6)

Parameters

Recommended Action
Refer to the specific device type and information passed via this alarm to determine the appropriate action.

StationConnectionError

Station device is closing its connection with Cisco Unified Communications Manager because of the reason that is stated in this alarm.

History

<table>
<thead>
<tr>
<th>Cisco Unified Communications Release</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0(1)</td>
<td>• Reason Code [Enum] parameter added.</td>
</tr>
<tr>
<td></td>
<td>• Enum Definitions for Reason Code table added.</td>
</tr>
</tbody>
</table>

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/CallManager

Severity
Informational

Parameters
Device Name [String]
Reason Code [Enum]

Enum Definitions -Reason Code

<table>
<thead>
<tr>
<th>Code</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>deviceInitiatedReset—The device has initiated a reset, possibly due to a power cycle or internal error. No action required; the device will reregister automatically.</td>
</tr>
</tbody>
</table>
TestAlarmAppliance

Testing alarm for Appliance OS based server only.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
System/Test

Severity
Informational (6)

Recommended Action
None

TestAlarmInformational

Testing informational alarm.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER
Cisco Unified Serviceability Alarm Definition Catalog
System/Test

Severity
Informational (6)

Recommended Action
None

**TVSCertificateRegenerated**

TVS Server certificate has been regenerated. This usually means that the TVS certificate has been regenerated. TVS server will automatically be restarted.

Cisco Unified Serviceability Alarm Catalog
System/TVS
Severity
INFORMATIONAL

Routing List
SDI
Event Log
Data Collector
Sys Log

Recommended Action
None.

**UserAlreadyLoggedIn**

User is already logged in.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Informational (6)

Parameters
UserID [String]

Recommended Action
None
UserLoggedOut

User logged out.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Informational (6)

Parameters
UserID [String]

Recommended Action
None

UserLoginSuccess

User successfully logged in.

Facility/Sub-Facility
CCM_TCD-TCD

Cisco Unified Serviceability Alarm Definition Catalog
CallManager/TCD SRV

Severity
Informational (6)

Parameters
UserID [String]

Recommended Action
None

WDInformation

WebDialer informational alarm.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications
Severity
Informational (6)

Parameters
Servlet Name [String] Reason [String]

Recommended Action
None

WDStarted
WebDialer Application started successfully.

Facility/Sub-Facility
CCM_JAVA_APPS-TOMCATAPPLICATIONS

Cisco Unified Serviceability Alarm Definition Catalog
System/Java Applications

Severity
Informational (6)

Parameters
Servlet Name [String] Reason [String]

Recommended Action
None

Debug-Level Alarms
The debug-level alarm is 7 and no action needed. Debug messages are used for troubleshooting.

TestAlarmDebug
Testing debug alarm.

Facility/Sub-Facility
CCM_CALLMANAGER-CALLMANAGER

Cisco Unified Serviceability Alarm Definition Catalog
System/Test

Severity
Debug (7)
Obsolete Alarms in Cisco Unified Communications Manager Release 8.0(1)

Recommended Action
None

Obsolete Alarms in Cisco Unified Communications Manager Release 8.0(1)

This section explains the alarms obsoleted in Cisco Unified Serviceability.
## Obsolete Alarms in CallManager Catalog

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConferenceCreated</td>
<td>INFORMATIONAL</td>
<td>An application controlled conference is created.</td>
</tr>
<tr>
<td>ConferenceDeleted</td>
<td>INFORMATIONAL</td>
<td>An application controlled conference is deleted.</td>
</tr>
<tr>
<td>CtiCallAcceptTimeout</td>
<td>WARNING</td>
<td>Call Accept Timeout</td>
</tr>
<tr>
<td>CtiStaleCallHandle</td>
<td>INFORMATIONAL</td>
<td>CTI stale call handle</td>
</tr>
<tr>
<td>DatabaseAuditInfo_074</td>
<td>INFORMATIONAL</td>
<td>Database audit information</td>
</tr>
<tr>
<td>DatabaseDeviceNoDirNum</td>
<td>NOTICE</td>
<td>No directory number for database device.</td>
</tr>
<tr>
<td>DatabaseInternalDataError_06e</td>
<td>ALERT</td>
<td>Database internal data error</td>
</tr>
<tr>
<td>DatabaseInternalDataError_06f</td>
<td>NOTICE</td>
<td>Database internal data error</td>
</tr>
<tr>
<td>DatabaseInternalDataError_070</td>
<td>INFORMATIONAL</td>
<td>Database internal data error</td>
</tr>
<tr>
<td>DatabaseInternalDataError_071</td>
<td>INFORMATIONAL</td>
<td>Database internal data error</td>
</tr>
<tr>
<td>DatabaseInternalDataError_072</td>
<td>INFORMATIONAL</td>
<td>Database internal data error</td>
</tr>
<tr>
<td>DatabaseInternalDataError_073</td>
<td>INFORMATIONAL</td>
<td>Database internal data error</td>
</tr>
<tr>
<td>DatabaseInternalDataError_075</td>
<td>INFORMATIONAL</td>
<td>Database internal data error</td>
</tr>
<tr>
<td>DnTimeout</td>
<td>ERROR</td>
<td>DN Timeout.</td>
</tr>
<tr>
<td>GatewayAlarm</td>
<td>INFORMATIONAL</td>
<td>Gateway alarm.</td>
</tr>
<tr>
<td>H323AddressResolutionError</td>
<td>WARNING</td>
<td>H323 address not resolved.</td>
</tr>
<tr>
<td>H323CallFailureAlarm</td>
<td>WARNING</td>
<td>H323 Call failure</td>
</tr>
<tr>
<td>MWIParamMisMatch</td>
<td>WARNING</td>
<td>MWI parameter mismatch</td>
</tr>
<tr>
<td>NoConnection</td>
<td>INFORMATIONAL</td>
<td>No TCP connection.</td>
</tr>
<tr>
<td>OutOfDnForAutoRegistration</td>
<td>WARNING</td>
<td>Out of directory numbers for auto-registration.</td>
</tr>
<tr>
<td>PktCapDownloadFailed</td>
<td>ERROR</td>
<td>Did not get captured packet or key file.</td>
</tr>
<tr>
<td>PktCapDownloadOK</td>
<td>INFORMATIONAL</td>
<td>Downloaded captured packet or key file.</td>
</tr>
<tr>
<td>PktCapLoginFailed</td>
<td>ERROR</td>
<td>Login failed for getting captured packet or key file.</td>
</tr>
<tr>
<td>PktCapLoginOK</td>
<td>INFORMATIONAL</td>
<td>Login OK for getting captured packet or key file.</td>
</tr>
<tr>
<td>Redirection</td>
<td>WARNING</td>
<td>Redirection Manager cannot register with the Call Control.</td>
</tr>
<tr>
<td>SIP IPPortConflict</td>
<td>WARNING</td>
<td>The local port for this device is already in use.</td>
</tr>
<tr>
<td>ThrottlingSampleActivity</td>
<td>ERROR</td>
<td>ThrottlingSampleActivity</td>
</tr>
<tr>
<td>TotalCodeYellowEntry</td>
<td>INFORMATIONAL</td>
<td>TotalCodeYellowEntry</td>
</tr>
</tbody>
</table>
### Obsolete Alarms in CertMonitor Alarm Catalog

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CertExpired</td>
<td>EMERGENCY</td>
<td>Certificate has Expired and needs to be changed at the earliest.</td>
</tr>
<tr>
<td>CertExpiryApproaching</td>
<td>INFORMATIONAL</td>
<td>Information Alarm that indicates a Certificate Validity Period is approaching and the expiry date is within the notification window configured.</td>
</tr>
<tr>
<td>CertExpiryDebug</td>
<td>DEBUG</td>
<td>Alarm to Debug Certificate Management.</td>
</tr>
</tbody>
</table>

### Obsolete Alarms in CMI Alarm Catalog

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCMConnectionError ER</td>
<td>ROR</td>
<td>CMI cannot establish connection with the Cisco Unified Communications Manager.</td>
</tr>
<tr>
<td>CMIDebugAlarm DEBUG</td>
<td>DEBUG</td>
<td>This alarm is generated only for the purpose of debugging.</td>
</tr>
<tr>
<td>CMIServiceStarted NOTICE</td>
<td>NOTICE</td>
<td>Service is now running.</td>
</tr>
<tr>
<td>CMIServiceStopped NOTICE</td>
<td>NOTICE</td>
<td>Service is now stopping.</td>
</tr>
<tr>
<td>COMException ALE RT</td>
<td>RT</td>
<td>CMI catches an COM exception.</td>
</tr>
<tr>
<td>ConfigParaNotFound NOT</td>
<td>OTICE</td>
<td>CMI service configuration parameter is not found in Database.</td>
</tr>
<tr>
<td>DisconnectionToCCM ER</td>
<td>ROR</td>
<td>CMI loses the connection with Unified Communications Manager.</td>
</tr>
<tr>
<td>WSAStartupFailed CRITICAL</td>
<td></td>
<td>Windows Socket startup failed.</td>
</tr>
</tbody>
</table>

### Obsolete Alarms in CTI Manager Alarm Catalog

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>kCtiDeviceOpenFailAccessDenied</td>
<td>WARNING</td>
<td>DeviceOpenRequest failure.</td>
</tr>
<tr>
<td>kCtiDirectoryLoginFailure</td>
<td>WARNING</td>
<td>CTI directory login failure.</td>
</tr>
<tr>
<td>kCtiEnvProcDevListRegTimeout</td>
<td>ERROR</td>
<td>Directory change notification request time out.</td>
</tr>
<tr>
<td>kCtiExistingCallNotifyArrayOverflow</td>
<td>WARNING</td>
<td>Possible internal array overflow condition while generating CTI ExistingCall event.</td>
</tr>
<tr>
<td>kCtiIllegalEnumHandle</td>
<td>WARNING</td>
<td>Enumeration handle is not valid.</td>
</tr>
<tr>
<td>kCtiIllegalFilterSize</td>
<td>ERROR</td>
<td>ProviderOpenRequest; illegal filter size.</td>
</tr>
<tr>
<td>kCtiIllegalQbeHeader</td>
<td>ERROR</td>
<td>Illegal QBE header.</td>
</tr>
<tr>
<td>kCtiInvalidQbeSizeAndOffsets</td>
<td>ERROR</td>
<td>InvalidQBESizeAndOffsets; QBE message decoding encountered illegal size or offset.</td>
</tr>
<tr>
<td>Alarm Name</td>
<td>Severity</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>kCtiLineCallInfoResArrayOverflow</td>
<td>WARNING</td>
<td>Possible internal array overflow condition while generating response to application request for call information.</td>
</tr>
<tr>
<td>kCtiLineOpenFailAccessDenied</td>
<td>WARNING</td>
<td>Line open failed.</td>
</tr>
<tr>
<td>kCtiMYTCPSendError</td>
<td>ERROR</td>
<td>MYTCP_Send: send error.</td>
</tr>
<tr>
<td>kCtiMytcpErrSocketBroken</td>
<td>WARNING</td>
<td>Socket connection has been broken.</td>
</tr>
<tr>
<td>kCtiNewCallNotifyArrayOverflow</td>
<td>WARNING</td>
<td>Possible internal array overflow condition while generating CTI NewCall event.</td>
</tr>
<tr>
<td>kCtiNullTcpHandle</td>
<td>WARNING</td>
<td>TranslateCtiQbeInputMessage: NULL TCP HANDLE!!! (QBE packet is dropped)</td>
</tr>
<tr>
<td>kCtiProviderOpenInvalidUserName-Size</td>
<td>ERROR</td>
<td>Invalid userName size in ProviderOpen request.</td>
</tr>
<tr>
<td>kCtiQbeLengthMisMatch</td>
<td>ERROR</td>
<td>OutputQbeMessage: length mismatch.</td>
</tr>
<tr>
<td>kCtiQbeMessageTooLong</td>
<td>WARNING</td>
<td>Incoming QBE message exceeds input buffer size</td>
</tr>
<tr>
<td>kCtiSdlErrorvException</td>
<td>CRITICAL</td>
<td>Failed to create an internal process that is required to service CTI applications.</td>
</tr>
<tr>
<td>kCtiSsRegisterManagerErr</td>
<td>ERROR</td>
<td>Unable to register CtiLine with SSAPI.</td>
</tr>
<tr>
<td>kCtiTcpInitError</td>
<td>ERROR</td>
<td>CTIManager service is unable to initialize TCP connection</td>
</tr>
<tr>
<td>kCtiUnknownConnectionHandle</td>
<td>WARNING</td>
<td>Connection handle is not valid</td>
</tr>
</tbody>
</table>
### Obsolete Alarms in DB Alarm Catalog

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ErrorChangeNotifyReconcile ALE</td>
<td>RT</td>
<td>A change notification shared memory reconciliation has occurred.</td>
</tr>
</tbody>
</table>

### Obsolete Alarms in IpVms Alarm Catalog

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>kANNAudioComException</td>
<td>ERROR</td>
<td>ANN TFTP COM exception</td>
</tr>
<tr>
<td>kANNAudioOpenFailed</td>
<td>ERROR</td>
<td>Open announcement file failed</td>
</tr>
<tr>
<td>kANNAudioTftpFileMissing</td>
<td>ERROR</td>
<td>ANN TFTP file missing</td>
</tr>
<tr>
<td>kANNAudioTftpMgrCreate</td>
<td>ERROR</td>
<td>Unable to create TFTP client</td>
</tr>
<tr>
<td>kANNAudioTftpMgrStartFailed</td>
<td>ERROR</td>
<td>TFTP start file transfer failed</td>
</tr>
<tr>
<td>kANNAudioThreadException</td>
<td>ERROR</td>
<td>ANN TFTP transfer exception failure</td>
</tr>
<tr>
<td>kANNAudioThreadWaitFailed</td>
<td>ERROR</td>
<td>ANN TFTP event wait error</td>
</tr>
<tr>
<td>kANNAudioThreadxFailed</td>
<td>ERROR</td>
<td>ANN TFTP transfer thread creation failed</td>
</tr>
<tr>
<td>kANNAudioXmlLoadFailed</td>
<td>ERROR</td>
<td>ANN XML parsing error</td>
</tr>
<tr>
<td>kANNAudioXmlSyntaxERROR</td>
<td>ERROR</td>
<td>ANN XML invalid element</td>
</tr>
<tr>
<td>kAddIpVmsRenderFailed</td>
<td>ERROR</td>
<td>Add IP VMS render filter-to-filter graph failure.</td>
</tr>
<tr>
<td>kCfgListComException</td>
<td>ERROR</td>
<td>Configuration COM Exception</td>
</tr>
<tr>
<td>kCfgListDb1Exception</td>
<td>ERROR</td>
<td>Configuration DBL Exception</td>
</tr>
<tr>
<td>kCfgListUnknownException</td>
<td>ERROR</td>
<td>Unknown Configuration Exception</td>
</tr>
<tr>
<td>kCreateGraphManagerFailed</td>
<td>ERROR</td>
<td>Get graph manager failure.</td>
</tr>
<tr>
<td>kDeviceMgrThreadException</td>
<td>ERROR</td>
<td>Exception in device manager thread.</td>
</tr>
<tr>
<td>kDownloadMOHFileFailed</td>
<td>ERROR</td>
<td>Download request failure.</td>
</tr>
<tr>
<td>kFixedInputAddAudioCaptureDeviceFailed</td>
<td>ERROR</td>
<td>Add fixed audio source to filter graph failure.</td>
</tr>
<tr>
<td>kFixedInputAddG711AlawIpVmsRenderFailed</td>
<td>ERROR</td>
<td>Add fixed G711 a-law IP VMS render filter-to-filter graph failure.</td>
</tr>
<tr>
<td>kFixedInputAddG711UlawIpVmsRenderFailed</td>
<td>ERROR</td>
<td>Add fixed G711 ulaw IP VMS render filter to filter graph failed</td>
</tr>
<tr>
<td>kFixedInputAddG729IpVmsRenderFailed</td>
<td>ERROR</td>
<td>Add fixed G729 IP VMS render filter-to-filter graph failure.</td>
</tr>
<tr>
<td>kFixedInputAddMOHEncoderFailed</td>
<td>ERROR</td>
<td>Add fixed MOH encode filter-to-filter graph failure.</td>
</tr>
<tr>
<td>kFixedInputAddWideBandIpVmsRenderFailed</td>
<td>ERROR</td>
<td>Add fixed wideband IP VMS render filter-to-filter graph failure.</td>
</tr>
<tr>
<td>kFixedInputAudioCapMOHEncoderConnFailed</td>
<td>ERROR</td>
<td>Connect fixed audio capture device to MOH encoder failure.</td>
</tr>
</tbody>
</table>
### Alarm Name

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>kFixedInputAudioCaptureCreateFailed</td>
<td>ERROR</td>
<td>Get fixed system device enumerator failure.</td>
</tr>
<tr>
<td>kFixedInputClassEnumeratorCreateFailed</td>
<td>ERROR</td>
<td>Create fixed class enumerator failure.</td>
</tr>
<tr>
<td>kFixedInputCreateGraphManagerFailed</td>
<td>ERROR</td>
<td>Get fixed graph manager failure.</td>
</tr>
<tr>
<td>kFixedInputFindAudioCaptureDeviceFailed</td>
<td>ERROR</td>
<td>Unable to find fixed audio source device.</td>
</tr>
<tr>
<td>kFixedInputGetEventNotificationFailed</td>
<td>ERROR</td>
<td>Get fixed notification event failure.</td>
</tr>
<tr>
<td>kFixedInputGetNameFailed</td>
<td>ERROR</td>
<td>Get fixed audio source device name failure.</td>
</tr>
<tr>
<td>kFixedInputGetG711AlawIpVmsRenderInfFailed</td>
<td>ERROR</td>
<td>Get fixed G711 a-law IP VMS render filter private interface failure.</td>
</tr>
<tr>
<td>kFixedInputGetG711AlawIpVmsRenderErrorFailed</td>
<td>ERROR</td>
<td>Get fixed G711 a-law IP VMS render filter failure.</td>
</tr>
<tr>
<td>kFixedInputGetG711UlawIpVmsRenderInfFailed</td>
<td>ERROR</td>
<td>Get fixed G711 mu-law IP VMS render filter private interface failure.</td>
</tr>
<tr>
<td>kFixedInputGetG711UlawIpVmsRenderErrorFailed</td>
<td>ERROR</td>
<td>Get fixed G711 mu-law IP VMS render filter failure.</td>
</tr>
<tr>
<td>kFixedInputGetG729IpVmsRenderInfFailed</td>
<td>ERROR</td>
<td>Get fixed G729 IP VMS render filter private interface failure.</td>
</tr>
<tr>
<td>kFixedInputGetG729IpVmsRenderErrorFailed</td>
<td>ERROR</td>
<td>Get fixed G729 IP VMS render filter failure.</td>
</tr>
<tr>
<td>kFixedInputGetMOHEncoderFailed</td>
<td>ERROR</td>
<td>Get fixed MOH encode filter failure.</td>
</tr>
<tr>
<td>kFixedInputGetMediaControlFailed</td>
<td>ERROR</td>
<td>Get fixed media control failure.</td>
</tr>
<tr>
<td>kFixedInputGetMediaPositionFailed</td>
<td>ERROR</td>
<td>Get fixed media position failure.</td>
</tr>
<tr>
<td>kFixedInputGetWideBandIpVmsRenderInfFailed</td>
<td>ERROR</td>
<td>Get fixed wideband IP VMS render filter private interface failure.</td>
</tr>
<tr>
<td>kFixedInputGetWideBandIpVmsRenderErrorFailed</td>
<td>ERROR</td>
<td>Get fixed wideband IP VMS render filter failure.</td>
</tr>
<tr>
<td>kFixedInputMOHEncG711AlawRendererConnFail</td>
<td>ERROR</td>
<td>Connect fixed MOH encoder to G711 a-law IP VMS render filter failure.</td>
</tr>
<tr>
<td>kFixedInputMOHEncG711UlawRendererConnFail</td>
<td>ERROR</td>
<td>Connect fixed MOH encoder to G711 u-law IP VMS render filter failure.</td>
</tr>
<tr>
<td>kFixedInputMOHEncG729RendererConnFailed</td>
<td>ERROR</td>
<td>Connect fixed MOH encoder to G729 IP VMS render filter failure.</td>
</tr>
<tr>
<td>kFixedInputMOHEncWidebandRendererConnFail</td>
<td>ERROR</td>
<td>Connect fixed MOH encoder to wideband IP VMS render filter failure.</td>
</tr>
<tr>
<td>kFixedInputSetNotifyWindowFailed</td>
<td>ERROR</td>
<td>Set fixed notify window failure.</td>
</tr>
<tr>
<td>kGetEventNotificationFailed</td>
<td>ERROR</td>
<td>Get notification event failure.</td>
</tr>
<tr>
<td>kGetIpVmsRenderFailed</td>
<td>ERROR</td>
<td>Get IP VMS render filter failure.</td>
</tr>
<tr>
<td>Alarm Name</td>
<td>Severity</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>----------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>kGetIpVmsRenderInterfaceFailed</td>
<td>ERROR</td>
<td>Get IP VMS render filter private interface failure.</td>
</tr>
<tr>
<td>kGetMediaControlFailed</td>
<td>ERROR</td>
<td>Get media control failure.</td>
</tr>
<tr>
<td>kGetMediaPositionFailed</td>
<td>ERROR</td>
<td>Get media position failure.</td>
</tr>
<tr>
<td>kMOHFilterNotifyError</td>
<td>ERROR</td>
<td>Error on DirectShow returned or user abort.</td>
</tr>
<tr>
<td>kMOHMgrThreadCreateWindowExFailed</td>
<td>ERROR</td>
<td>Creation of MOH manager message window failure.</td>
</tr>
<tr>
<td>kMOHPlayStreamControlNull</td>
<td>ERROR</td>
<td>Stream Control pointer is NULL.</td>
</tr>
<tr>
<td>kMOHPlayStreamMediaControlObjectNull</td>
<td>ERROR</td>
<td>Media Position COM interface is NULL.</td>
</tr>
<tr>
<td>kMOHThreadException</td>
<td>ERROR</td>
<td>Exception in MOH manager thread.</td>
</tr>
<tr>
<td>kMTPICMPErrorNotification</td>
<td>INFORMATIONAL</td>
<td>MTP stream ICMP error.</td>
</tr>
<tr>
<td>kPWavMgrExitEventCreateFailed</td>
<td>ERROR</td>
<td>Creation of needed event failed.</td>
</tr>
<tr>
<td>kPWavMgrThreadException</td>
<td>ERROR</td>
<td>WAV file manager thread exception.</td>
</tr>
<tr>
<td>kReadCfgANNComException</td>
<td>ERROR</td>
<td>COM error.</td>
</tr>
<tr>
<td>kReadCfgANNDblException</td>
<td>ERROR</td>
<td>Database exception.</td>
</tr>
<tr>
<td>kReadCfgANNListComException</td>
<td>ERROR</td>
<td>COM error.</td>
</tr>
<tr>
<td>kReadCfgANNListDbIException</td>
<td>ERROR</td>
<td>Database exception.</td>
</tr>
<tr>
<td>kReadCfgANNListUnknownException</td>
<td>ERROR</td>
<td>Unknown exception.</td>
</tr>
<tr>
<td>kReadCfgANNUnknownException</td>
<td>ERROR</td>
<td>Unknown exception.</td>
</tr>
<tr>
<td>kReadCfgCFBComException</td>
<td>ERROR</td>
<td>COM error.</td>
</tr>
<tr>
<td>kReadCfgCFBDblException</td>
<td>ERROR</td>
<td>Database exception.</td>
</tr>
<tr>
<td>kReadCfgCFBListComException</td>
<td>ERROR</td>
<td>COM error.</td>
</tr>
<tr>
<td>kReadCfgCFBListDbIException</td>
<td>ERROR</td>
<td>Database exception.</td>
</tr>
<tr>
<td>kReadCfgCFBListUnknownException</td>
<td>ERROR</td>
<td>Unknown exception.</td>
</tr>
<tr>
<td>kReadCfgCFBUUnknownException</td>
<td>ERROR</td>
<td>Unknown exception.</td>
</tr>
<tr>
<td>kReadCfgDbIGetChgNotifyFailed</td>
<td>INFORMATIONAL</td>
<td>Get change notification port failure.</td>
</tr>
<tr>
<td>kReadCfgDbIGetNodeNameFailed</td>
<td>ERROR</td>
<td>Database layer select my process node failed.</td>
</tr>
<tr>
<td>kReadCfgEnterpriseComException</td>
<td>ERROR</td>
<td>COM error.</td>
</tr>
<tr>
<td>kReadCfgEnterpriseDbIException</td>
<td>ERROR</td>
<td>Database exception.</td>
</tr>
<tr>
<td>kReadCfgEnterpriseException</td>
<td>ERROR</td>
<td>Enterpriseswide configuration exception</td>
</tr>
<tr>
<td>kReadCfgEnterpriseUnknownException</td>
<td>ERROR</td>
<td>Unknown exception.</td>
</tr>
<tr>
<td>kReadCfgMOHAudioSourceComException</td>
<td>ERROR</td>
<td>COM error.</td>
</tr>
<tr>
<td>kReadCfgMOHAudioSourceDbIException</td>
<td>ERROR</td>
<td>Database exception.</td>
</tr>
</tbody>
</table>
### Obsolete Alarms in Cisco Unified Communications Manager Release 8.0(1)

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>kReadCfgMOHAudioSourceUnknownException</td>
<td>ERROR</td>
<td>Unknown exception.</td>
</tr>
<tr>
<td>kReadCfgMOHComException</td>
<td>ERROR</td>
<td>COM error.</td>
</tr>
<tr>
<td>kReadCfgMOHDbException</td>
<td>ERROR</td>
<td>Database exception.</td>
</tr>
<tr>
<td>kReadCfgMOHListComException</td>
<td>ERROR</td>
<td>COM error.</td>
</tr>
<tr>
<td>kReadCfgMOHListDbException</td>
<td>ERROR</td>
<td>Database exception.</td>
</tr>
<tr>
<td>kReadCfgMOHListUnknownException</td>
<td>ERROR</td>
<td>Unknown exception.</td>
</tr>
<tr>
<td>kReadCfgMOHServerComException</td>
<td>ERROR</td>
<td>COM error.</td>
</tr>
<tr>
<td>kReadCfgMOHServerDbException</td>
<td>ERROR</td>
<td>Database exception.</td>
</tr>
<tr>
<td>kReadCfgMOHServerUnknownException</td>
<td>ERROR</td>
<td>Unknown exception.</td>
</tr>
<tr>
<td>kReadCfgMOHTFTIPAddressNotFound</td>
<td>ERROR</td>
<td>MOH TFTP IP address not found.</td>
</tr>
<tr>
<td>kReadCfgMOHUnknownException</td>
<td>ERROR</td>
<td>Unknown exception.</td>
</tr>
<tr>
<td>kReadCfgMTPComException</td>
<td>ERROR</td>
<td>COM error.</td>
</tr>
<tr>
<td>kReadCfgMTPDbException</td>
<td>ERROR</td>
<td>Database exception.</td>
</tr>
<tr>
<td>kReadCfgMTPListComException</td>
<td>ERROR</td>
<td>COM error.</td>
</tr>
<tr>
<td>kReadCfgMTPListDbException</td>
<td>ERROR</td>
<td>Database exception.</td>
</tr>
<tr>
<td>kReadCfgMTPListUnknownException</td>
<td>ERROR</td>
<td>Unknown exception.</td>
</tr>
<tr>
<td>kReadCfgMTPUnknownException</td>
<td>ERROR</td>
<td>Unknown exception.</td>
</tr>
<tr>
<td>kRenderFileFailed</td>
<td>ERROR</td>
<td>Render file-to-filter graph failure.</td>
</tr>
<tr>
<td>kSetNotifyWindowFailed</td>
<td>ERROR</td>
<td>Set notify window failure.</td>
</tr>
</tbody>
</table>

### Obsolete Alarms in Test Alarm Catalog

<table>
<thead>
<tr>
<th>Alarm Name</th>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TestAlarmWindows</td>
<td>INFORMATIONAL</td>
<td>Testing INFORMATIONAL_ALARM.</td>
</tr>
</tbody>
</table>
Chapter 6  Cisco Unified Serviceability Alarms and CiscoLog Messages

Obsolete Alarms in Cisco Unified Communications Manager Release 8.0(1)
Cisco Management Information Base

This chapter describes the Management Information Base (MIB) text files that are supported by Cisco Unified Communications Manager (Cisco Unified CM) and are used with Simple Network Management Protocol (SNMP). The chapter contains the following sections:

- CISCO-CCM-MIB, page 7-1
- CISCO-CCM-CAPABILITY, page 7-121
- CISCO-CDP-MIB, page 7-127
- CISCO-SYSLOG-MIB, page 7-144
- CISCO-SYSLOG-EXT-MIB, page 7-152

CISCO-CCM-MIB

This is a reformatted version of CISCO-CCM-MIB. Download and compile all of the MIBs in this section from http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

This MIB manages the Cisco Unified Communications Manager (Cisco Unified CM) application running with a Cisco Communication Network (CCN) system. Cisco Unified CM is an IP-PBX that controls the call processing of a VoIP network.

A CCN system comprises multiple regions, with each region consisting of several Cisco Unified CM groups with multiple Cisco Unified CM servers. The MIB can be used by the Cisco Unified CM application, Cisco Unified CM Administration, to present provision and statistics information.

The following terminology applies to this MIB:

- SCCP—Skinny Client Control Protocol
- SIP—Session Initiation Protocol
- TLS—Transport Layer Security
- MGCP—Media Gateway Control Protocol

Before you can compile CISCO-CCM-MIB, you need to download and compile the MIBs listed below in the order listed.

1. SNMPv2-SMI
2. SNMPv2-TC
3. SNMPv2-CONF
4. CISCO-SMI
5. INET-ADDRESS-MIB
6. SNMP-FRAMEWORK-MIB
7. RFC1155-SMI
8. RFC1212
9. SNMPv2-TC-v1
10. CISCO-CCM-MIB

Additional downloads are:
- OID File: CISCO-CCM-MIB.OID
- Capability File: CISCO-CCM-CAPABILITY

The following are contained in this section:
- Revisions, page 7-3
- Definitions, page 7-14
- Textual Conventions, page 7-14
- Objects, page 7-20
- Tables, page 7-21
- Alarms, page 7-67
- Notification and Alarms, page 7-70
- Cisco Unified CM Managed Services and SNMP Traps, page 7-106
- Cisco Unified CM Alarms to Enable, page 7-106
- Traps to Monitor, page 7-107
- Dynamic Table Objects, page 7-109
- Static Table Objects, page 7-110
- Troubleshooting, page 7-111
Revisions

Table 7-1 lists the revisions to this MIB beginning with the latest revision first.

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>July 2010</td>
<td>Updated the TEXTUAL-CONVENTIONS</td>
<td>CcmDevUnregCauseCode, CcmDevRegFailCauseCode</td>
</tr>
<tr>
<td>Dec 2009</td>
<td>Deprecated</td>
<td>CcmDevFailCauseCode; Added CcmDevRegFailCauseCode and CcmDevUnregCauseCode</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneStatusReason; Added ccmPhoneUnregReason and ccmPhoneRegFailReason in ccmPhoneTable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneFailCauseCode; Added ccmPhoneFailedRegFailReason in ccmPhoneFailedTable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneStatusUpdateReason; Added ccmPhoneStatusUnregReason and ccmPhoneStatusRegFailReason in ccmPhoneStatusUpdateTable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmGatewayStatusReason; Added ccmGatewayUnregReason and ccmGatewayRegFailReason in ccmGatewayTable.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmMediaDeviceStatusReason; Added ccmMediaDeviceUnregReason and ccmMediaDeviceRegFailReason in ccmMediaDeviceTable.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmCTIDeviceStatusReason; Added ccmCTIDeviceUnregReason and ccmCTIDeviceRegFailReason in ccmCTIDeviceTable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmH323DevStatusReason; Added ccmH323DevUnregReason and ccmH323DevRegFailReason in ccmH323DeviceTable.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmVMailDevStatusReason; Added ccmVMailDevUnregReason and ccmVMailDevRegFailReason in ccmVoiceMailDeviceTable.</td>
</tr>
</tbody>
</table>
Table 7-1  
*History of MIB Revisions (continued)*

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Deprecated</td>
<td>ccmGatewayFailCauseCode; Added ccmGatewayRegFailCauseCode in ccmNotificationsInfo.</td>
</tr>
<tr>
<td></td>
<td>Deprecated the following Notification Type</td>
<td>ccmGatewayFailed and added ccmGatewayFailedReason.</td>
</tr>
<tr>
<td>08-21-2008</td>
<td>Deprecated following MODULE-COMPLIANCE</td>
<td>ciscoCcmMIBComplianceRev6; Added ciscoCcmMIBComplianceRev7.</td>
</tr>
<tr>
<td></td>
<td>Obsoleted following OBJECT_GROUPS</td>
<td>ccmInfoGroupRev3, ccmH323DeviceInfoGroupRev1</td>
</tr>
<tr>
<td></td>
<td>Added following objects in ccmCTIDeviceTable</td>
<td>ccmCTIDeviceNetAddressIPv4, ccmCTIDeviceNetAddressIPv6. These objects replaced the ccmCTIDeviceNetAddressType and ccmCTIDeviceNetAddress.</td>
</tr>
<tr>
<td></td>
<td>Deprecated following objects in ccmCTIDeviceTable</td>
<td>ccmCTIDeviceNetAddressType, ccmCTIDeviceNetAddress.</td>
</tr>
<tr>
<td></td>
<td>Added following OBJECT-GROUP</td>
<td>ccmCTIDeviceInfoGroupRev3. This group replaced the ccmCTIDeviceInfoGroupRev2</td>
</tr>
<tr>
<td></td>
<td>Deprecated following OBJECT-GROUP</td>
<td>ccmCTIDeviceInfoGroupRev2.</td>
</tr>
</tbody>
</table>
### Table 7-1  History of MIB Revisions (continued)

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>02-12-2008</td>
<td>Added following MODULE-COMPLIANCE</td>
<td>ciscoCcmMIBComplianceRev6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This compliance replaced the ciscoCcmMIBComplianceRev5.</td>
</tr>
<tr>
<td></td>
<td>Deprecated</td>
<td>ciscoCcmMIBComplianceRev5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODULE-COMPLIANCE</td>
</tr>
<tr>
<td></td>
<td>Added following objects in ccmTable</td>
<td>ccmInetAddress2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmInetAddress2Type</td>
</tr>
<tr>
<td></td>
<td>Added following objects in ccmPhoneTable</td>
<td>ccmPhoneInetAddressIPv4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneInetAddressIPv6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneIPv4Attribute</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneIPv6Attribute</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneActiveLoadID</td>
</tr>
<tr>
<td></td>
<td>Added following objects in ccmPhoneFailedTable</td>
<td>ccmPhoneFailedInetAddressIPv4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneFailedInetAddressIPv6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneFailedIPv4Attribute</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneFailedIPv6Attribute</td>
</tr>
<tr>
<td></td>
<td>Added following objects in ccmSIPDeviceTable</td>
<td>ccmSIPDevInetAddressIPv4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmSIPDevInetAddressIPv6</td>
</tr>
<tr>
<td></td>
<td>Added following objects in ccmMediaDeviceTable</td>
<td>ccmMediaDeviceInetAddressIPv4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmMediaDeviceInetAddressIPv6</td>
</tr>
<tr>
<td></td>
<td>Deprecated following objects in ccmPhoneTable</td>
<td>ccmPhoneInetAddressType</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneInetAddress</td>
</tr>
<tr>
<td></td>
<td>Deprecated following objects in ccmPhoneFailedTable</td>
<td>ccmPhoneFailedInetAddressType</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneFailedInetAddress</td>
</tr>
<tr>
<td></td>
<td>Deprecated following objects in ccmSIPDeviceTable</td>
<td>ccmSIPDevInetAddressType</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmSIPDevInetAddress</td>
</tr>
<tr>
<td></td>
<td>Deprecated following objects in ccmMediaDeviceTable</td>
<td>ccmMediaDeviceInetAddressType</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmMediaDeviceInetAddress</td>
</tr>
<tr>
<td></td>
<td>Added following scalar objects</td>
<td>ccmH323TableEntries</td>
</tr>
<tr>
<td></td>
<td></td>
<td>csmSIPTableEntries</td>
</tr>
<tr>
<td></td>
<td>Obsoleted</td>
<td>ciscoCcmMIBComplianceRev3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODULE-COMPLIANCE</td>
</tr>
<tr>
<td></td>
<td>Deprecated ciscoCcm</td>
<td>MIBComplianceRev4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODULE-COMPLIANCE</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>ciscoCcmMIBComplianceRev5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MODULE-COMPLIANCE</td>
</tr>
</tbody>
</table>
### Table 7-1 History of MIB Revisions (continued)

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Obsoleted following NOTIFICATION-GROUPS</td>
<td>ccmNotificationsGroup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmNotificationsGroupRev1</td>
</tr>
<tr>
<td></td>
<td>Obsoleted following OBJECT-GROUPS</td>
<td>ccmInfoGroupRev2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneInfoGroupRev3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmSIPDeviceInfoGroup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmNotificationsInfoGroupRev1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmNotificationsInfoGroupRev2</td>
</tr>
<tr>
<td></td>
<td>Deprecated following OBJECT-GROUPS</td>
<td>ccmInfoGroupRev3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneInfoGroupRev4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmSIPDeviceInfoGroupRev1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmMediaDeviceInfoGroupRev2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmH323DeviceInfoGroupRev1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmNotificationsInfoGroupRev3</td>
</tr>
<tr>
<td></td>
<td>Added following OBJECT-GROUPS</td>
<td>ccmInfoGroupRev4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneInfoGroupRev5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmMediaDeviceInfoGroupRev3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmNotificationsInfoGroupRev4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmH323DeviceInfoGroupRev2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmSIPDeviceInfoGroupRev2</td>
</tr>
</tbody>
</table>
09-14-2005
Updated CcmDevFailCauseCode definition to include more cause codes.
authenticationError
invalidX509NameInCertificate
invalidTLS_CIPHER, directoryNumberMismatch
malformedRegisterMsg
Updated the description of these objects.

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
</table>
| 09-14-2005 | Updated CcmDevFailCauseCode definition to include more cause codes. | authenticationError
invalidX509NameInCertificate
invalidTLS_CIPHER, directoryNumberMismatch
malformedRegisterMsg
Updated the description of these objects. | ccmPhoneFailedInetAddress
ccmGatewayInetAddress
ccmMediaDeviceInetAddress
ccmGatekeeperInetAddress
ccmCTIDeviceInetAddress
ccmH323DevInetAddress
ccmH323DevCnfgGKInetAddress
ccmH323DevAltGK1InetAddress
ccmH323DevAltGK2InetAddress
ccmH323DevAltGK3InetAddress
ccmH323DevAltGK4InetAddress
ccmH323DevAltGK5InetAddress
ccmH323DevActGKInetAddress
ccmH323DevRmtCM1InetAddress
ccmH323DevRmtCM2InetAddress
ccmH323DevRmtCM3InetAddress
ccmVMailDevInetAddress

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>09-05-2005</td>
<td>Added partiallyregistered to CcmDeviceStatus TC</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>Added phonePartiallyregistered to ccmPhoneStatusUpdateType TC</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>Added these TCs</td>
<td>CcmPhoneProtocolType</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CcmDeviceLineStatus</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CcmSIPTransportProtocolType</td>
</tr>
<tr>
<td></td>
<td>Added these objects to ccmPhoneTable ccmPhoneProtocol</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneName</td>
</tr>
<tr>
<td></td>
<td>Added ccmPhoneExtnStatus to ccmPhoneExtnTable</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>Added following objects to ccmSIPDeviceTable:</td>
<td>ccmSIPInTransportProtocolType</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmSIPOutTransportProtocolType</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmSIPInPortNumber, ccmSIPOutPortNumber</td>
</tr>
<tr>
<td></td>
<td>Added ccmTLSConnectionFailure notification</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>Updated the description of following objects under ccmSIPDeviceTable</td>
<td>ccmTLSConnectionFailReasonCode</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmSIPDevName</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmSIPDevDescription</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmSIPDevInetAddress</td>
</tr>
<tr>
<td></td>
<td>Updated the description of ccmCallManagerAlarmEnable</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>Added the following object groups</td>
<td>ccmPhoneInfoGroupRev4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmNotificationsInfoGroupRev3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmSIPDeviceInfoGroupRev1</td>
</tr>
<tr>
<td></td>
<td>Added the following notification groups:</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmNotificationsGroupRev2</td>
</tr>
<tr>
<td></td>
<td>Added MIB compliance</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>ciscoCcmMIBComplianceRev4</td>
<td>—</td>
</tr>
</tbody>
</table>
Table 7-1  History of MIB Revisions (continued)

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
</table>
| 08-02-2004 | Obsoleted    | ccmDeviceProductId  
ccmTimeZoneOffset  
ccmPhoneType  
ccmPhoneLastError  
ccmPhoneTimeLastError  
ccmPhoneExtensionTable  
ccmPhoneExtensionTable  
ccmPhoneExtensionEntry  
ccmPhoneExtensionEntry  
ccmPhoneExtensionIndex  
ccmPhoneExtensionIndex  
ccmPhoneExtension  
ccmPhoneExtensionMultiLines  
ccmPhoneExtensionInetAddressType  
ccmPhoneExtensionInetAddress  
ccmPhoneFailedName  
ccmGatewayType  
ccmGatewayProductId  
ccmActivePhones  
ccmInActivePhones  
ccmActiveGateways  
ccmInActiveGateways  
ccmMediaDeviceType  
ccmCTIDeviceType  
ccmCTIDeviceAppInfo |
Table 7-1  History of MIB Revisions (continued)

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
</table>
### Table 7-1  
**History of MIB Revisions (continued)**

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>08-25-2003</td>
<td>Added</td>
<td>The definition of ccmMaliciousCall and ccmQualityReport notifications and its objects</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>H323 trunk types and SIP trunk type in ccmDeviceProductId</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>More media device types in ccmMediaDevice table</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>The definition of ccmSystemVersion and ccmInstallationId objects to ccmGlobalInfo group</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>ccmSIPDeviceInfo definition</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>More phone types</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>The definition of ccmProductTypeTable to list the product types supported at run time</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>ccmPhoneProductTypeIndex</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmGatewayProductTypeIndex</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmMediaDeviceProductTypeIndex</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmCTIDeviceProductTypeIndex</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmH323DevProductTypeIndex</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmVMailDevProductTypeIndex</td>
</tr>
<tr>
<td></td>
<td>Deprecated</td>
<td>ccmPhoneType, ccmGatewayType, ccmGatewayProductId, ccmMediaDeviceType, ccmCTIDeviceType, ccmH323DevProductTypeId, ccmVMailDevProductTypeIndex objects</td>
</tr>
<tr>
<td>05-08-2003</td>
<td>Added</td>
<td>More phone types in the ccmPhoneType definition</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>More gateway types in the ccmGatewayType and CcmDeviceProductId definition</td>
</tr>
<tr>
<td>01-11-2002</td>
<td>Updated</td>
<td>CcmDevFailCauseCode definition to include more cause codes deviceInitiatedReset, callManagerReset and noError</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>ccmH323DeviceInfo and ccmVoiceMailDeviceInfo objects</td>
</tr>
<tr>
<td></td>
<td>Updated</td>
<td>ccmRegionAvailableBandwidth definition to include two more bandwidth types: bwGSM and bwWideband</td>
</tr>
<tr>
<td></td>
<td>Deprecated</td>
<td>ccmTimeZoneOffset object</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>ccmTimeZoneOffsetHours and ccmTimeZoneOffsetMinutes to ccmTimeZoneTable</td>
</tr>
<tr>
<td>Date</td>
<td>Action</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmCTIDeviceStatusReason</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmCTIDeviceStatusReason</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmCTIDeviceTimeLastStatusUpdt</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmCTIDeviceTimeLastRegistered to ccmCTIDeviceTable</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>Rejected status to ccmCTIDeviceStatus</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>More objects to the ccmGlobalInfo</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmPhoneStatusUpdate</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmPhoneStatusUpdateReason</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmPhoneStatusUpdate</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmPhoneStatusUpdateReason</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>object to ccmPhoneStatusUpdate</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmPhoneStatusUpdate table</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmGatewayProductId</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmGatewayStatusReason</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmGatewayStatusReason</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmGatewayTimeLastStatusUpdt</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmGatewayTimeLastRegistered</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmGatewayDChannelStatus</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmGatewayDChannelNumber objects to ccmGatewayTable</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>New types to ccmGatewayType</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>Rejected status to ccmGatewayStatus</td>
</tr>
<tr>
<td>Obsoleted</td>
<td></td>
<td>The ccmGatewayTrunkInfo (this was never supported)</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmMediaDeviceStatusReason</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmMediaDeviceStatusReason, ccmMediaDeviceTimeLastStatusUpdt</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>ccmMediaDeviceTimeLastRegistered to ccmMediaDeviceTable</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>More types to ccmMediaDeviceType</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>Rejected status to ccmMediaDeviceStatus</td>
</tr>
<tr>
<td>Deprecated</td>
<td></td>
<td>The ccmGatekeeperTable definition</td>
</tr>
<tr>
<td>Added</td>
<td></td>
<td>Rejected status to ccmGatekeeperstatus</td>
</tr>
<tr>
<td>Updated</td>
<td></td>
<td>ccmMIBCompliance statements</td>
</tr>
<tr>
<td>Added ccmPhoneS</td>
<td></td>
<td>statusReason</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneStatusReason</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneTimeLastStatusUpdt to ccmPhoneTable</td>
</tr>
</tbody>
</table>
### Table 7-1 History of MIB Revisions (continued)

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Added</td>
<td>Rejected status to ccmPhoneStatus</td>
</tr>
<tr>
<td></td>
<td>Deprecated</td>
<td>ccmPhoneFailedTableName and added ccmPhoneMacAddress to ccmPhoneFailedTable</td>
</tr>
<tr>
<td></td>
<td>Deprecated</td>
<td>ccmPhoneLastError and ccmPhoneTimeLastError in ccmPhoneTable</td>
</tr>
<tr>
<td></td>
<td>Deprecated</td>
<td>ccmCTIDeviceAppInfo in ccmCTIDeviceTable</td>
</tr>
<tr>
<td></td>
<td>Defined</td>
<td>CcmDeviceProductId and CcmDeviceStatus textual conventions</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>ccmPhoneExtTable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhStatUpdtTblLastAddedIndex</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhFailedTblLastAddedIndex</td>
</tr>
<tr>
<td></td>
<td>Deprecated</td>
<td>ccmPhoneExtensionTable</td>
</tr>
<tr>
<td></td>
<td>Changed the default values</td>
<td>ccmCallManagerAlarmEnable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmGatewayAlarmEnable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneFailedStorePeriod</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneStatusUpdate</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneStatusUpdateStorePeriod objects</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneFailedStorePeriod</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneStatusUpdate</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneStatusUpdateStorePeriod objects</td>
</tr>
<tr>
<td>12-01-2000</td>
<td>Added</td>
<td>ccmMediaDeviceInfo</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmGatekeeperInfo</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmCTIDeviceInfo</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmAlarmConfigInfo</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmNotificationsInfo objects</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>ccmClusterId to the ccmEntry</td>
</tr>
<tr>
<td></td>
<td>Deprecated</td>
<td>ccmGatewayTrunkInfo (this was never implemented and it should have been in the gateway MIB)</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>ccmPhoneFailedTable and ccmPhoneStatusUpdateTable</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>ccmMIBNotifications</td>
</tr>
<tr>
<td></td>
<td>Added</td>
<td>New ccmGatewayType and ccmPhoneType</td>
</tr>
<tr>
<td>03-10-2000</td>
<td>Added</td>
<td>This revision clause.</td>
</tr>
<tr>
<td></td>
<td>The initial version of this MIB module</td>
<td>::= { ciscoMgmt 156 }</td>
</tr>
</tbody>
</table>
Definitions

The following definitions are imported for CISCO-CCM-MIB:

- MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE, IpAddress, Counter32, Integer32, Unsigned32
- From SNMPv2-SMI—DateAndTime, TruthValue, MacAddress, TEXTUAL-CONVENTION
- From SNMPv2-TC—SnmpAdminString
- From SNMP-FRAMEWORK-MIB—MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
- From SNMPv2-CONF—ciscoMgmt
- From CISCO-SMI—InetAddressType, InetAddress, InetPortNumber
- From INET-ADDRESS-MIB

Textual Conventions

CcmIndex ::= TEXTUAL-CONVENTION
   DISPLAY-HINT d
   STATUS current
   DESCRIPTION
   This syntax is used as the Index into a table. A positive value is used to identify a unique entry in the table.
   SYNTAX   Unsigned32(1..4294967295)

CcmIndexOrZero ::= TEXTUAL-CONVENTION
   DISPLAY-HINT d
   STATUS current
   DESCRIPTION
   This textual convention is an extension of the CcmIndex convention. The latter defines a greater than zero to identify an entry of the CCM MIB table in the managed system. This extension permits the additional value of zero. The value zero is object-specific and must be defined as part of the description of any object that uses this syntax.
   SYNTAX   Unsigned32 (0..4294967295)

CcmDevRegFailCauseCode ::= TEXTUAL-CONVENTION
   STATUS current
   DESCRIPTION
   This syntax is used as means of identifying the reasons for a device registration failure. The scope of this enumeration can expand to comply with RFC 2578.
   noError: No Error
   unknown: Unknown error cause
   noEntryInDatabase: Device not configured properly in the Cisco Unified CM database
   databaseConfigurationError: Device configuration error in the Cisco Unified CM database
deviceNameUnresolveable: The Cisco Unified CM is unable to resolve the device name to an IP Address internally

maxDevRegExceeded: Maximum number of device registrations have been reached

connectivityError: Cisco Unified CM is unable to establish communication with the device during registration

initializationError: Indicates an error occurred when the Cisco Unified CM tries to initialize the device

deviceInitiatedReset: Indicates that the error was due to device initiated reset

callManagerReset: Indicates that the error was due to Cisco Unified CM reset

authenticationError: Indicates mismatch between configured authentication mode and the authentication mode that the device is using to connect to the Cisco Unified CM

invalidX509NameInCertificate: Indicates mismatch between the peer X.509 certificate subject name and what is configured for the device

invalidTLSCipher: Indicates Cipher mismatch during TLS handshake process

directoryNumberMismatch: Indicates mismatch between the directory number that the SIP device is trying to register with and the directory number configured in the Cisco Unified CM for the SIP device

malformedRegisterMsg: Indicates that SIP device attempted to register with Cisco Unified CM, but the REGISTER message contained formatting errors

protocolMismatch: The protocol of the device (SIP or SCCP) does not match the configured protocol in Cisco Unified CM

deviceNotActive: The device has not been activated

authenticatedDeviceAlreadyExists: A device with the same name is already registered with Cisco Unified CM

obsoleteProtocolVersion: The SCCP device registered with an obsolete protocol version

databaseTimeout: Cisco Unified CM requested device configuration data from the database but did not receive a response within 10 minutes

registrationSequenceError: (SCCP only) A device requested configuration information from the Cisco Unified CM at an unexpected time. The Cisco Unified CM had not yet obtained the requested information. The device will automatically attempt to register again. If this alarm occurs again, manually reset the device. If this alarm continues to occur after the manual reset, there may be an internal firmware error

invalidCapabilities: (SCCP only) The Cisco Unified CM detected an error in the media capabilities reported in the StationCapabilitiesRes message by the device during registration. The device will automatically attempt to register again. If this alarm occurs again, manually reset the device. If this alarm continues to occur after the manual reset, there may be a protocol error

capabilityResponseTimeout: (SCCP only) The Cisco Unified CM timed out while waiting for the device to respond to a request to report its media capabilities. Possible causes include device power outage, network power outage, network configuration error, network delay, packet drops, and packet corruption. It is also possible to get this error if the Cisco Unified CM node is experiencing high CPU usage. Verify that the device is powered up and operating. Verify that network connectivity exists between the device and Cisco Unified CM, and verify that the CPU utilization is in the safe range

securityMismatch: The Cisco Unified CM detected a mismatch in the security settings of the device and/or the Cisco Unified CM. The mismatches that can be detected are:
The device established a secure connection, yet reported that it does not have the ability to do authenticated signaling.

The device did not establish a secure connection, but the security mode configured for the device indicates that it should have done so.

The device established a secure connection, but the security mode configured for the device indicates that it should not have done so.

**autoRegisterDBError**—Auto-registration of a device failed for one of the following reasons:

- Auto-registration is not allowed for the device type.
- An error occurred while adding the auto-registering device to the database (stored procedure).

**dbAccessError**—Device registration failed because of an error that occurred while building the station registration profile. This usually indicates a synchronization problem with the database.

**autoRegisterDBConfigTimeout**—(SCCP only) The Cisco Unified CM timed out during auto-registration of a device. The registration profile of the device did not get inserted into the database in time. The device will automatically attempt to register again.

**deviceTypeMismatch**—The device type reported by the device does not match the device type configured on the Cisco Unified CM.

**addressingModeMismatch**—(SCCP only) The Cisco Unified CM detected an error related to the addressing mode configured for the device. One of the following errors were detected:

- The device is configured to use only IPv4 addressing, but did not specify an IPv4 address.
- The device is configured to use only IPv6 addressing, but did not specify an IPv6 address.

**SYNTAX INTEGER {**

- `noError(0),`
- `unknown(1),`
- `noEntryInDatabase(2),`
- `databaseConfigurationError(3),`
- `deviceNameUnresolveable(4),`
- `maxDevRegExceeded(5),`
- `connectivityError(6),`
- `initializationError(7),`
- `deviceInitiatedReset(8),`
- `callManagerReset(9),`
- `authenticationError(10),`
- `invalidX509NameInCertificate(11),`
- `invalidTLSCipher(12),`
- `directoryNumberMismatch(13),`
- `malformedRegisterMsg(14),`
- `protocolMismatch(15),`
- `deviceNotActive(16),`
- `authenticatedDeviceAlreadyExists(17),`
- `obsoleteProtocolVersion(18),`
databaseTimeout(23),
registrationSequenceError(25),
invalidCapabilities(26),
capabilityResponseTimeout(27),
securityMismatch(28),
autoRegisterDBError(29),
dbAccessError(30),
autoRegisterDBConfigTimeout(31),
deviceTypeMismatch(32),
addressingModeMismatch(33)
}

CcmDevUnregCauseCode ::= TEXTUAL-CONVENTION

STATUS current
DESCRIPTION
This syntax is used as means of identifying the reasons for a device getting unregistered. The scope of this enumeration can expand to comply with RFC 2578.
noError: No Error
unknown: Unknown error cause
noEntryInDatabase: Device not configured properly in the Cisco Unified CM database
databaseConfigurationError: Device configuration error in the Cisco Unified CM database
deviceNameUnresolveable: The Cisco Unified CM is unable to resolve the device name to an IP Address internally
maxDevRegExceeded: Maximum number of device registrations have been reached
connectivityError: Cisco Unified CM is unable to establish communication with the device during registration
initializationError: Indicates that an error occurred when the Cisco Unified CM tries to initialize the device
deviceInitiatedReset: Indicates that the error was due to device initiated reset
callManagerReset: Indicates that the error was due to Cisco Unified CM reset.
deviceUnregistered: DeviceUnregistered.
malformedRegisterMsg: Indicates that SIP device attempted to register with Cisco Unified CM, but the REGISTER message contained formatting errors.
scmpDeviceThrottling: The indicated SCCP device exceeded the maximum number of events allowed per-SCCP device.
keepAliveTimeout: A KeepAlive message was not received. Possible causes include device power outage, network power outage, network configuration error, network delay, packet drops, packet corruption and Cisco Unified CM node experiencing high CPU usage.
configurationMismatch: The configuration on the SIP device does not match the configuration in Cisco Unified CM.
callManagerRestart: A device restart was initiated from Cisco Unified CM Administration, either due to an explicit command from an administrator or due to a configuration change such as adding, deleting or changing a directory number associated with the device.

duplicateRegistration: Cisco Unified CM detected that the device attempted to register to two nodes at the same time. Cisco Unified CM initiated a restart to the phone to force it to re-home to a single node.

callManagerApplyConfig: Cisco Unified CM configuration is changed.

deviceNoResponse: Device is not responding Service Control Notify from Cisco Unified CM.

emLoginLogout: The device has been unregistered due to an Extension Mobility login or logout.

eemccLoginLogout: The device has been unregistered due to an Extension Mobility Cross Cluster login or logout.

powerSavePlus: The device powered off as a result of the Power Save Plus feature that is enabled for this device. When the device powers off, it remains unregistered from Cisco Unified CM until the Phone On Time defined in the Product Specific Configuration for this device.

callManagerForcedRestart: (SIP Only) The device did not respond to an Apply Config request and as a result, Cisco Unified CM had sent a restart request to the device. The device may be offline due to a power outage or network problem. Confirm that the device is powered-up and that network connectivity exists between the device and Cisco Unified CM.

sourceIPAddrChanged: (SIP Only) The device has been unregistered because the IP address in the Contact header of the REGISTER message has changed. The device will be automatically reregistered. No action is necessary.

sourcePortChanged: (SIP Only) The device has been unregistered because the port number in the Contact header of the REGISTER message has changed. The device will be automatically reregistered. No action is necessary.

registrationSequenceError: (SCCP only) A device requested configuration information from the Cisco Unified CM at an unexpected time. The Cisco Unified CM no longer had the requested information in memory.

invalidCapabilities: (SCCP only) The Cisco Unified CM detected an error in the updated media capabilities reported by the device. The device reported the capabilities in one of the StationUpdateCapabilities message variants.

fallbackInitiated: The device has initiated a fallback and will automatically reregister to a higher-priority Cisco Unified CM. No action is necessary.

deviceSwitch: A second instance of an endpoint with the same device name has registered and assumed control. No action is necessary.

SYNTAX INTEGER { 
  noError(0),
  unknown(1),
  noEntryInDatabase(2),
  databaseConfigurationError(3),
  deviceNameUnresolvable(4),
  maxDevRegExceeded(5),
  connectivityError(6),
  initializationError(7),
}
deviceInitiatedReset(8),
callManagerReset(9),
deviceUnregistered(10),
malformedRegisterMsg(11),
sccpDeviceThrottling(12),
keepAliveTimeout(13),
configurationMismatch(14),
callManagerRestart(15),
duplicateRegistration(16),
callManagerApplyConfig(17),
deviceNoResponse(18),
emLoginLogout(19),
emccLoginLogout(20),
ergywisePowerSavePlus(21),
callManagerForcedRestart(22),
sourceIPAddrChanged(23),
sourcePortChanged(24),
registrationSequenceError(25),
invalidCapabilities(26),
fallbackInitiated(28),
deviceSwitch(29)
}

CcmDeviceStatus ::= TEXTUAL-CONVENTION

STATUS current
DESCRIPTION
This syntax is used to identify the registration status of a device with the local Cisco Unified CM. The status is as follows:
- unknown—The registration status of the device is unknown
- registered—The device has successfully registered with the local Cisco Unified CM.
- unregistered—The device is no longer registered with the local Cisco Unified CM.
- rejected—Registration request from the device was rejected by the local Cisco Unified CM.
- partiallyregistered—At least one but not all of the lines are successfully registered to the Cisco Unified CM. Applicable only to SIP Phones.

SYNTAX INTEGER { unknown (1), registered (2), unregistered (3), rejected (4), partiallyregistered (5)}

CcmPhoneProtocolType ::= TEXTUAL-CONVENTION

STATUS current
DESCRIPTION
This syntax is used to identify the protocol between phone and Cisco Unified CM. The protocols are as follows:

- **unknown**—The phone protocol is unknown
- **sccp**—The phone protocol is SCCP
- **sip**—The phone protocol is SIP

SYNTAX INTEGER { unknown(1), sccp (2), sip(3) }

**CcmDeviceLineStatus ::= TEXTUAL-CONVENTION**

**STATUS** current

**DESCRIPTION**

This syntax is used to identify the registration status of a line of the device with the local Cisco Unified CM. The status is as follows:

- **unknown**—The registration status of the device line is unknown
- **registered**—The device line has successfully registered with the local Cisco Unified CM.
- **unregistered**—The device line is no longer registered with the local Cisco Unified CM.
- **rejected**—Registration request from the device line was rejected by the local Cisco Unified CM.

SYNTAX INTEGER { unknown   (1), registered(2), unregistered (3), rejected (4)}

**CcmSIPTransportProtocolType ::= TEXTUAL-CONVENTION**

**STATUS** current

**DESCRIPTION**

This textual convention defines the possible transport protocol types that are used for setting up SIP calls unknown. The possible transport types are:

- **unknown**—The SIP Trunk transport type is unknown
- **tcp**—The SIP Trunk transport type is tcp
- **udp**—The SIP Trunk transport type is udp
- **tcpAndUdp**—The SIP Trunk transport type is tcp and udp
- **tls**—Applicable only for InTransportProtocolType is tls. The SIP Trunk transport type is tls.

SYNTAX INTEGER { unknown(1), tcp(2), udp(3), tcpAndUdp (4), tls(5) }

---

**Objects**

ciscoCcmMIBObjects OBJECT IDENTIFIER ::= { ciscoCcmMIB 1 }
ccmGeneralInfo OBJECT IDENTIFIER ::= { ciscoCcmMIBObjects 1 }
ccmPhoneInfo OBJECT IDENTIFIER ::= { ciscoCcmMIBObjects 2 }
ccmGatewayInfo OBJECT IDENTIFIER ::= { ciscoCcmMIBObjects 3 }
ccmGatewayTrunkInfo OBJECT IDENTIFIER ::= { ciscoCcmMIBObjects 4 }
ccmGlobalInfo OBJECT IDENTIFIER ::= { ciscoCcmMIBObjects 5 }
ccmMediaDeviceInfo OBJECT IDENTIFIER ::= { ciscoCcmMIBObjects 6 }
ccmGatekeeperInfo OBJECT IDENTIFIER ::= { ciscoCcmMIBObjects 7 }
ccmCTIDeviceInfo OBJECT IDENTIFIER ::= { ciscoCcmMIBObjects 8 }
Tables

Cisco Unified CM Group Table

ccmGroupTable OBJECT-TYPE
SYNTAX SEQUENCE OF CcmGroupEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing the CallManager groups in a Cisco Unified CM cluster.
::= { ccmGeneralInfo 1 }

ccmGroupEntry OBJECT-TYPE
SYNTAX CcmGroupEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the CallManager Group table, containing the information about a CallManager group in a Cisco Unified CM cluster. An entry is created to represent a CallManager Group. New entries to the CallManager Group table in the database are created when the User inserts a new CallManager Group via the CallManager Web Admin pages. This entry is subsequently picked up by the Cisco Unified CM SNMP Agent.
INDEX { ccmGroupIndex }
::= { ccmGroupTable 1 }

CcmGroupEntry
 ::= SEQUENCE
 { ccmGroupIndex CcmIndex,
   ccmGroupName SnmpAdminString,
   ccmGroupTftpDefault TruthValue
  }

ccmGroupIndex OBJECT-TYPE
SYNTAX CcmIndex
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM that uniquely identifies a Cisco Unified
CM Group.
::= { ccmGroupEntry 1 }

ccmGroupName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The name of the Cisco Unified CM Group.
::= { ccmGroupEntry 2 }

ccmGroupTftpDefault OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Whether this is the default TFTP server group or not.
::= { ccmGroupEntry 3 }

Cisco Unified CM Table

ccmTable OBJECT-TYPE
SYNTAX SEQUENCE of CcmEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing information of all the Cisco Unified CMs in a Cisco Unified CM cluster that
the local Cisco Unified CM knows about. When the local Cisco Unified CM is restarted, this table
will be refreshed.
::= { ccmGeneralInfo 2 }

ccmEntry OBJECT-TYPE
SYNTAX CcmEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the CallManager table, containing the information about a
CallManager.
INDEX { ccmIndex }
::= { ccmTable 1 }

CcmEntry ::= SEQUENCE
{
  ccmIndex CcmIndex,
  ccmName SnmpAdminString,
  ccmDescription SnmpAdminString,
  ccmVersion SnmpAdminString,
  ccmStatus Integer,
  ccmInetAddressType InetAddressType,
  ccmInetAddress InetAddress,
  ccmClusterId SnmpAdminString,
  ccmInetAddress2Type InetAddressType,
  ccmInetAddress2 InetAddress
}

ccmIndex OBJECT-TYPE
SYNTAX CcmIndex
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that uniquely identifies a CallManager in a Cisco Unified CM cluster.
::= { ccmEntry 1 }

ccmName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The host name of the CallManager.
::= { ccmEntry 2 }

ccmDescription OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..255))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The description for the CallManager.
::= { ccmEntry 3 }

ccmVersion OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..24))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The version number of the CallManager software.
::= { ccmEntry 4 }

ccmStatus OBJECT-TYPE
SYNTAX INTEGER
{
   unknown(1),
   up(2),
   down(3)
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The current status of the CallManager. A CallManager is up if the SNMP Agent received a system
up event from the local Cisco Unified CM:
unknown: Current status of the CallManager is Unknown
up: CallManager is running and is able to communicate with other CallManagers
down: CallManager is down or the Agent is unable to communicate with the local
CallManager.
::= { ccmEntry 5 }

ccmInetAddressType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the IP address type of the Cisco Unified CM defined in ccmInetAddress.
::= { ccmEntry 6 }

ccmInetAddress OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies IP address of the Cisco Unified CM. The type of address for this is identified
by ccmInetAddressType.
::= { ccmEntry 7 }
**ccmClusterId** OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The unique ID of the Cluster to which this Cisco Unified CM belongs. At any point in time, the Cluster ID helps in associating a Cisco Unified CM to any given Cluster.
::= { ccmEntry 8 }

**ccmInetAddress2Type** OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies IP address type of the Cisco Unified Communications Manager defined in ccmInetAddress2.
::= { ccmEntry 9 }

**ccmInetAddress2** OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the additional IP(v4/v6) address details of Cisco Unified Communications Manager. The type of address for this object is identified by ccmInetAddress2Type.
::= { ccmEntry 10 }

**Cisco Unified CM Group Mapping Table**

**ccmGroupMappingTable** OBJECT-TYPE
SYNTAX SEQUENCE OF CcmGroupMappingEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing the list of all CallManager to group mappings in a Cisco Unified CM cluster. When the local Cisco Unified CM is down, this table will be empty.
::= { ccmGeneralInfo 3 }

**ccmGroupMappingEntry** OBJECT-TYPE
SYNTAX CcmGroupMappingEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the CallManager group Mapping table, containing the information about a mapping between a CallManager and a CallManager group.
INDEX { ccmGroupIndex, ccmIndex }
::= { ccmGroupMappingTable 1 }
CcmGroupMappingEntry ::= SEQUENCE {
 ccmCMGroupMappingCMPriority Unsigned32
}

ccmCMGroupMappingCMPriority OBJECT-TYPE
SYNTAX  Unsigned32
MAX-ACCESS read-only
STATUS  current
DESCRIPTION
The priority of the CallManager in the group. Sets the order of the CallManager in the list.
::= { ccmGroupMappingEntry 1 }

Cisco Unified CM Region Table

ccmRegionTable OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmRegionEntry
MAX-ACCESS not-accessible
STATUS  current
DESCRIPTION
The table containing the list of all geographically separated regions in a CCN system.
::= { ccmGeneralInfo 4 }

ccmRegionEntry OBJECT-TYPE
SYNTAX  CcmRegionEntry
MAX-ACCESS not-accessible
STATUS  current
DESCRIPTION
An entry (conceptual row) in the Region Table, containing the information about a region.
INDEX { ccmRegionIndex }
::= { ccmRegionTable 1 }
CcmRegionEntry ::= SEQUENCE {
 ccmRegionIndex CcmIndex,
 ccmRegionName   SnmpAdminString
}

ccmRegionIndex OBJECT-TYPE
SYNTAX  CcmIndex
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that uniquely identifies a Region Name in the table.
::= { ccmRegionEntry 1 }

ccmRegionName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The name of the CallManager region.
::= { ccmRegionEntry 2 }

Cisco Unified CM Region Pair Table

ccmRegionPairTable OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmRegionPairEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
The table containing the list of all geographical region pairs defined for a Cisco Unified CM cluster. The pair consists of the Source region and Destination region.
::= { ccmGenerallInfo 5 }

ccmRegionPairEntry OBJECT-TYPE
SYNTAX  CcmRegionPairEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
An entry (conceptual row) in the Region Pair Table, containing the information about bandwidth restrictions when communicating between the two specified regions.
INDEX { ccmRegionSrcIndex, ccmRegionDestIndex }
::= { ccmRegionPairTable 1 }
CcmRegionPairEntry ::= SEQUENCE {
ccmRegionSrcIndex CcmIndex,
ccmRegionDestIndex CcmIndex,
ccmRegionAvailableBandWidth INTEGER
}

ccmRegionSrcIndex OBJECT-TYPE
SYNTAX CcmIndex
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The index of the Source Region in the Region table.
::= { ccmRegionPairEntry 1 }

ccmRegionDestIndex OBJECT-TYPE
SYNTAX CcmIndex
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The index of the Destination Region in the Region table.
::= { ccmRegionPairEntry 2 }

ccmRegionAvailableBandWidth OBJECT-TYPE
SYNTAX INTEGER {
    unknown(1),
    other(2),
    bwG723(3),
    bwG729(4),
    bwG711(5),
    bwGSM(6),
    bwWideband(7)
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The maximum available bandwidth between the two given regions.
unknown: Unknown Bandwidth
other: Unidentified Bandwidth
bwG723: For low bandwidth using G.723 codec
bwG729: For low bandwidth using G.729 codec
bwG711: For high bandwidth using G.711 codec
bwGSM: For GSM bandwidth 13K
bwWideband: For Wideband 256K.
::= { ccmRegionPairEntry 3 }
Cisco Unified CM Time Zone Table

ccmTimeZoneTable OBJECT-TYPE
SYNTAX SEQUENCE OF CcmTimeZoneEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing the list of all time zone groups in a call manager cluster.
::= { ccmGeneralInfo 6 }

ccmTimeZoneEntry OBJECT-TYPE
SYNTAX CcmTimeZoneEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the time zone Table, containing the information about a particular time zone group.
INDEX  { ccmTimeZoneIndex }
::= { ccmTimeZoneTable 1 }
CcmTimeZoneEntry ::= SEQUENCE {
ccmTimeZoneIndex CcmIndex,
ccmTimeZoneName SnmpAdminString,
ccmTimeZoneOffset   Integer32,
ccmTimeZoneOffsetHours  Integer32,
ccmTimeZoneOffsetMinutes Integer32
}

ccmTimeZoneIndex OBJECT-TYPE
SYNTAX  CcmIndex
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that uniquely identifies a Time Zone group entry in the table.
::= { ccmTimeZoneEntry 1 }

ccmTimeZoneName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The name of the time zone group.
::= { ccmTimeZoneEntry 2 }

**ccmTimeZoneOffsetHours OBJECT-TYPE**

SYNTAX Integer32 (-12..12)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The offset hours of the time zone group's time zone from GMT.
::= { ccmTimeZoneEntry 4 }

**ccmTimeZoneOffsetMinutes OBJECT-TYPE**

SYNTAX Integer32 (-59..59)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The offset minutes of the time zone group's time zone from GMT.
::= { ccmTimeZoneEntry 5 }

### Device Pool Table

**ccmDevicePoolTable OBJECT-TYPE**

SYNTAX SEQUENCE OF CcmDevicePoolEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing the list of all device pools in a call manager cluster. A Device Pool contains Region, Date/Time Group and CallManager Group criteria that will be common among many devices.
::= { ccmGenerallInfo 7 }

**ccmDevicePoolEntry OBJECT-TYPE**

SYNTAX CcmDevicePoolEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the device pool Table, containing the information about a particular device pool.
INDEX { ccmDevicePoolIndex }
::= { ccmDevicePoolTable 1 }

**CcmDevicePoolEntry**

::= SEQUENCE {

ccmDevicePoolIndex CcmIndex, ccmDevicePoolName SnmpAdminString, ccmDevicePoolRegionIndex CcmIndexOrZero, ccmDevicePoolTimeZoneIndex CcmIndexOrZero, ccmDevicePoolGroupIndex CcmIndexOrZero
}

ccmDevicePoolIndex OBJECT-TYPE
SYNTAX CcmIndex
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that uniquely identifies a Device Pool entry in the table. Each entry contains Region, Date/Time Group and CallManager Group criteria that will be common among many devices, for that entry.
::= { ccmDevicePoolEntry 1 }

ccmDevicePoolName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The name of the device pool.
::= { ccmDevicePoolEntry 2 }

ccmDevicePoolRegionIndex OBJECT-TYPE
SYNTAX CcmIndexOrZero
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A positive value of this index is used to identify the Region to which this Device Pool entry belongs. A value of zero indicates that the index to the Region table is Unknown.
::= { ccmDevicePoolEntry 3 }

ccmDevicePoolTimeZoneIndex OBJECT-TYPE
SYNTAX CcmIndexOrZero
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A positive value of this index is used to identify the TimeZone to which this Device Pool entry belongs. A value of zero indicates that the index to the TimeZone table is Unknown.
::= { ccmDevicePoolEntry 4 }

ccmDevicePoolGroupIndex OBJECT-TYPE
SYNTAX CcmIndexOrZero
MAX-ACCESS read-only
STATUS current

DESCRIPTION
A positive value of this index is used to identify the CallManager Group to which this Device Pool entry belongs. A value of zero indicates that the index to the CallManager Group table is Unknown.

::= { ccmDevicePoolEntry 5 }

Cisco Unified CM Product Type Table

ccmProductTypeTable OBJECT-TYPE
SYNTAX CcmProductTypeEntry
MAX-ACCESS not-accessible
STATUS current

DESCRIPTION
The table containing the list of product types supported in a call manager cluster. The product types will include the list of phone types, gateway types, media device types, H323 device types, CTI device types, Voice Messaging device types and SIP device types.

::= { ccmGeneralInfo 8 }

ccmProductTypeEntry OBJECT-TYPE
SYNTAX CcmProductTypeEntry
MAX-ACCESS not-accessible
STATUS current

DESCRIPTION
An entry (conceptual row) in the ccmProductTypeTable, containing the information about a product type supported in a call manager cluster. An entry is created to represent a product type.

INDEX { ccmProductTypeIndex }

::= { ccmProductTypeTable 1 }

CcmProductTypeEntry ::= SEQUENCE {
ccmProductTypeIndex CcmIndex,
ccmProductType Unsigned32,
ccmProductName SnmpAdminString,
ccmProductCategory INTEGER
}

ccmProductTypeIndex OBJECT-TYPE
SYNTAX CcmIndex
MAX-ACCESS not-accessible
STATUS current

DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that uniquely identifies an entry in the ccmProductTypeTable.

::= { ccmProductTypeEntry 1 }


```plaintext
ccmProductType OBJECT-TYPE
SYNTAX  Unsigned32
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The type of the product as defined in the Cisco Unified CM database.
::= { ccmProductTypeEntry 2 }

ccmProductName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..100))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The name of the product as defined in the Cisco Unified CM database.
::= { ccmProductTypeEntry 3 }

ccmProductCategory OBJECT-TYPE
SYNTAX  INTEGER {
    unknown(-1),
    notApplicable(0),
    phone(1),
    gateway(2),
    h323Device(3),
    ctiDevice(4),
    voiceMailDevice(5),
    mediaResourceDevice(6),
    huntListDevice(7),
    sipDevice(8)
}
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The category of the product.
    unknown: Unknown product category
    notApplicable: Not Applicable
    phone: Phone
    gateway: Gateway
    h323Device: H323 Device
    ctiDevice: CTI Device
    voiceMailDevice: Voice Messaging Device
```
mediaResourceDevice: Media Resource Device
huntListDevice: Hunt List Device
sipDevice: SIP Device.
::= { ccmProductTypeEntry 4 }

Phone Table

ccmPhoneTable OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmPhoneEntry
MAX-ACCESS not-accessible
STATUS  current
DESCRIPTION
The table containing the list of all IP Phone devices that have tried to register with the local Cisco Unified CM at least once. When the local Cisco Unified CM is restarted, this table will be refreshed.
::= { ccmPhoneInfo 1 }

ccmPhoneEntry OBJECT-TYPE
SYNTAX  CcmPhoneEntry
MAX-ACCESS not-accessible
STATUS  current
DESCRIPTION
An entry (conceptual row) in the phone Table, containing information about a particular phone device.
INDEX  { ccmPhoneIndex }
::= { ccmPhoneTable 1 }
CcmPhoneEntry ::= SEQUENCE {
ccmPhoneIndex   CcmIndex,
ccmPhonePhysicalAddress MacAddress,
ccmPhoneType INTEGER,
ccmPhoneDescription SnmpAdminString,
ccmPhoneUserName SnmpAdminString,
ccmPhoneIpAddress   IpAddress,
ccmPhoneStatus  CcmDeviceStatus,
ccmPhoneTimeLastRegistered  DateAndTime,
ccmPhoneE911Location SnmpAdminString,
ccmPhoneLoadID  SnmpAdminString,
ccmPhoneLastError   Integer32,
ccmPhoneTimeLastError   DateAndTime,
ccmPhoneDevicePoolIndex CcmIndexOrZero,
ccmPhoneInetAddressType InetAddressType,
ccmPhoneInetAddress InetAddress,
ccmPhoneStatusReason CcmDevFailCauseCode,
ccmPhoneTimeLastStatusUpdt DateAndTime,
ccmPhoneProductTypeIndexCcmIndexOrZero,
ccmPhoneProtocolCcmPhoneProtocolType,
ccmPhoneName SnmpAdminString
ccmPhoneInetAddressIPv4 InetAddressIPv4,
ccmPhoneInetAddressIPv6 InetAddressIPv6,
ccmPhoneIPv4Attribute INTEGER,
ccmPhoneIPv6Attribute INTEGER,
ccmPhoneActiveLoadID SnmpAdminString,
ccmPhoneUnregReason CcmDevUnregCauseCode,
ccmPhoneRegFailReason CcmDevRegFailCauseCode
}

ccmPhoneIndex OBJECT-TYPE
SYNTAX  CcmIndex
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that uniquely identifies a Phone within
the Cisco Unified CM.
::= { ccmPhoneEntry 1 }

ccmPhonePhysicalAddress OBJECT-TYPE
SYNTAX  MacAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The physical address(MAC address) of the IP phone.
::= { ccmPhoneEntry 2 }

ccmPhoneDescription OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..255))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The description of the phone.
::= { ccmPhoneEntry 4 }

ccmPhoneUserName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..255))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The name of the user of the phone. When the phone is not in use, the name would refer to the last
known user of the phone.
 ::= { ccmPhoneEntry 5 }

ccmPhoneStatus OBJECT-TYPE
SYNTAX  CcmDeviceStatus
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The status of the phone. The status of the Phone changes from Unknown to registered when it
registers itself with the local Cisco Unified CM.
 ::= { ccmPhoneEntry 7 }

ccmPhoneTimeLastRegistered OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The time when the phone last registered with the Cisco Unified CM.
 ::= { ccmPhoneEntry 8 }

ccmPhoneE911Location OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..255))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The E911 location of the phone.
 ::= { ccmPhoneEntry 9 }

ccmPhoneLoadID OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object identifies the configured load ID for the phone device.
 ::= { ccmPhoneEntry 10 }

ccmPhoneDevicePoolIndex OBJECT-TYPE
SYNTAX  CcmIndexOrZero
MAX-ACCESS  read-only
STATUS   current
DESCRIPTION
A positive value of this index is used to identify the Device Pool to which this Phone entry belongs. A value of 0 indicates that the index to the Device Pool table is Unknown.
::= { ccmPhoneEntry 13 }

ccmPhoneInetAddressType OBJECT-TYPE
SYNTAX   InetAddressType
MAX-ACCESS read-only
STATUS   deprecated
DESCRIPTION
This object identifies the IP address type of the phone.
::= { ccmPhoneEntry 14 }

ccmPhoneInetAddress OBJECT-TYPE
SYNTAX   InetAddress
MAX-ACCESS read-only
STATUS   current
DESCRIPTION
This object identifies the last known IP address of the phone. The type of address for this is identified by ccmPhoneInetAddressType.
::= { ccmPhoneEntry 15 }

ccmPhoneTimeLastStatusUpdt OBJECT-TYPE
SYNTAX   DateAndTime
MAX-ACCESS read-only
STATUS   current
DESCRIPTION
The time the status of the phone changed.
::= { ccmPhoneEntry 17 }

ccmPhoneProductTypeIndex OBJECT-TYPE
SYNTAX   CcmIndexOrZero
MAX-ACCESS read-only
STATUS   current
DESCRIPTION
A positive value of this index is used to identify the related product type entry in the ccmProductTypeTable. A value of 0 indicates that the index to the ccmProductTypeTable is Unknown.
::= { ccmPhoneEntry 18 }

ccmPhoneProtocol OBJECT-TYPE
SYNTAX   CcmPhoneProtocolType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The protocol used between the phone and Cisco Unified CM.
 ::= { ccmPhoneEntry 19 }
ccmPhoneName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The name of the phone. The name of the phone can be <prefix> + MAC Address, where <prefix> is SEP for Cisco SCCP and SIP Phones. In the case of other phones such as communicator (soft phone) it can be free-form name, a string that uniquely identifies the phone.
 ::= { ccmPhoneEntry 20 }
ccmPhoneInetAddressIPv4 OBJECT-TYPE
SYNTAX InetAddressIPv4
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the last known primary IPv4 address of the Phone Device. This object contains value zero if IPV4 address is not available.
 ::= { ccmPhoneEntry 21 }
ccmPhoneInetAddressIPv6 OBJECT-TYPE
SYNTAX InetAddressIPv6
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the last known primary IPv6 address of the Phone device. This object contains value zero if IPV6 address is not available.
 ::= { ccmPhoneEntry 22 }
ccmPhoneIPv4Attribute OBJECT-TYPE
SYNTAX INTEGER {
     unknown(0),
     adminOnly(1),
     controlOnly(2),
     adminAndControl(3)
 }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the scope of ccmPhoneInetAddressIPv4.
unknown(0): It is not known if ccmPhoneInetAddressIPv4 is used for Administration purpose or Controlling purpose.
adminOnly(1): ccmPhoneInetAddressIPv4 is used for the serviceability or administrative purpose.
controlOnly(2): ccmPhoneInetAddressIPv4 is used for signaling or registration purpose.
adminAndControl(3): ccmPhoneInetAddressIPv4 is used for controlling as well as administrative purpose.
 ::= { ccmPhoneEntry 23 }

**ccmPhoneIPv6Attribute** OBJECT-TYPE
SYNTAX INTEGER {
unknown(0),
adminOnly(1),
controlOnly(2),
adminAndControl(3)
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the scope of ccmPhoneInetAddressIPv6.
unknown(0): It is not known if ccmPhoneInetAddressIPv6 is used for Administration purpose or Controlling purpose.
adminOnly(1): ccmPhoneInetAddressIPv6 is used for the serviceability or administrative purpose.
controlOnly(2): ccmPhoneInetAddressIPv6 is used for signaling or registration purpose.
adminAndControl(3): ccmPhoneInetAddressIPv6 is used for controlling as well as administrative purpose.
 ::= { ccmPhoneEntry 24 }

**ccmPhoneActiveLoadID** OBJECT-TYPE
SYNTAX SnmpAdminString
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the ID of actual load that is successfully loaded and running on the phone device. If the phone is successfully upgraded to the new load then ccmPhoneLoadID and ccmPhoneActiveLoadID will have same value. If the upgrade fails then the ccmPhoneLoadID has the configured load ID and ccmPhoneActiveLoadID has the actual load ID that is running on the phone.
 ::= { ccmPhoneEntry 25 }

**ccmPhoneUnregReason** OBJECT-TYPE
SYNTAX CcmDevUnregCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with unregistered phone.
::= { ccmPhoneEntry 26 }

ccmPhoneRegFailReason OBJECT-TYPE
SYNTAX CcmDevRegFailCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with registration failed phone.
::= { ccmPhoneEntry 27 }

Phone Failed Table

ccmPhoneFailedTable OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmPhoneFailedEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
The table containing the list of all phones that attempted to register with the local call manager and failed. The entries that have not been updated and kept at least for the duration specified in the ccmPhoneFailedStorePeriod will be deleted. Reasons for these failures could be due to configuration error, maximum number of phones has been reached, lost contact, etc.
::= { ccmPhoneInfo 3 }

ccmPhoneFailedEntry OBJECT-TYPE
SYNTAX  CcmPhoneFailedEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
An entry (conceptual row) in the PhoneFailed Table, one for each phone failure in the Cisco Unified CM.
INDEX  { ccmPhoneFailedIndex }
::= { ccmPhoneFailedTable 1 }
CcmPhoneFailedEntry ::= SEQUENCE {
  ccmPhoneFailedIndex CcmIndex,
  ccmPhoneFailedTime DateAndTime,
  ccmPhoneFailedName SnmpAdminString,
  ccmPhoneFailedInetAddressType  InetAddressType,
  ccmPhoneFailedInetAddress  InetAddress,
ccmPhoneFailCauseCode CcmDevFailCauseCode,
ccmPhoneFailedMacAddress MacAddress
ccmPhoneFailedInetAddressIPv4 InetAddressIPv4,
ccmPhoneFailedInetAddressIPv6 InetAddressIPv6,
ccmPhoneFailedIPv4Attribute INTEGER,
ccmPhoneFailedIPv6Attribute INTEGER,
ccmPhoneFailedRegFailReason CcmDevRegFailCauseCode
}

ccmPhoneFailedIndex OBJECT-TYPE
SYNTAX CcmIndex
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that is incremented with each new
entry in the ccmPhoneFailedTable. This integer value will wrap if needed.
 ::= { ccmPhoneFailedEntry 1 }

ccmPhoneFailedTime OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The time when the phone failed to register with the Cisco Unified CM.
 ::= { ccmPhoneFailedEntry 2 }

ccmPhoneFailedMacAddress OBJECT-TYPE
SYNTAX MacAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The MAC address of the failed phone.
 ::= { ccmPhoneFailedEntry 7 }

ccmPhoneFailedInetAddressIPv4 OBJECT-TYPE
SYNTAX InetAddressIPv4
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the last known primary IPv4 address of the phone experiencing a
communication failure. This object contains value zero if IPV4 address is not available.
 ::= { ccmPhoneFailedEntry 8 }
ccmPhoneFailedInetAddressIPv6 OBJECT-TYPE
SYNTAX InetAddressIPv6
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the last known primary IPv6 address of the phone experiencing a communication failure. This object contains value zero if IPV6 address is not available.
::= { ccmPhoneFailedEntry 9 }

ccmPhoneFailedIPv4Attribute OBJECT-TYPE
SYNTAX INTEGER
{
  unknown(0),
  adminOnly(1),
  controlOnly(2),
  adminAndControl(3)
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the scope of ccmPhoneFailedInetAddressIPv4.
unknown(0): It is not known if ccmPhoneFailedInetAddressIPv4 is used for Administration purpose or Controlling purpose.
adminOnly(1): ccmPhoneFailedInetAddressIPv4 is used for the serviceability or administrative purpose.
controlOnly(2): ccmPhoneFailedInetAddressIPv4 is used for signaling or registration purpose.
adminAndControl(3): ccmPhoneFailedInetAddressIPv4 is used for controlling as well as administrative purpose.
::= { ccmPhoneFailedEntry 10 }

ccmPhoneFailedIPv6Attribute OBJECT-TYPE
SYNTAX INTEGER
{
  unknown(0),
  adminOnly(1),
  controlOnly(2),
  adminAndControl(3)
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the scope of ccmPhoneFailedInetAddressIPv6.
unknown(0): It is not known if ccmPhoneFailedInetAddressIPv6 is used for Administration purpose or Controlling purpose.
adminOnly(1): ccmPhoneFailedInetAddressIPv6 is used for the serviceability or administrative purpose.
controlOnly(2): ccmPhoneFailedInetAddressIPv6 is used for signaling or registration purpose.
adminAndControl(3): ccmPhoneFailedInetAddressIPv6 is used for controlling as well as administrative purpose.

::= { ccmPhoneFailedEntry 11 }

ccmPhoneFailedRegFailReason OBJECT-TYPE
SYNTAX CcmDevRegFailCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with registration failed phone.

::= { ccmPhoneFailedEntry 12 }

Phone Status Update Table

ccmPhoneStatusUpdateTable OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmPhoneStatusUpdateEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing the list of all phone status updates with respect to the local call manager. This table will only have registered, unregistered, and partially-registered status updates. The rejected phones are stored in the ccmPhoneFailedTable. Each entry of this table is stored at least for the duration specified in the ccmPhoneStatusUpdateStorePeriod object, after that it will be deleted.

::= { ccmPhoneInfo 4 }

ccmPhoneStatusUpdateEntry OBJECT-TYPE
SYNTAX  CcmPhoneStatusUpdateEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the PhoneStatusUpdate Table, one for each phone status update in the Cisco Unified CM.
INDEX  { ccmPhoneStatusUpdateIndex }

::= { ccmPhoneStatusUpdateTable 1 }

CcmPhoneStatusUpdateEntry ::= SEQUENCE {
ccmPhoneStatusUpdateIndex CcmIndex,
ccmPhoneStatusPhoneIndex CcmIndexOrZero,
ccmPhoneStatusUpdateTime DateAndTime,
ccmPhoneStatusUpdateType INTEGER,
ccmPhoneStatusUpdateReason CcmDevFailCauseCode
ccmPhoneStatusUnregReason CcmDevUnregCauseCode,
ccmPhoneStatusRegFailReason CcmDevRegFailCauseCode
}

ccmPhoneStatusUpdateIndex OBJECT-TYPE
SYNTAX  CcmIndex
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that is incremented with each new
entry in the ccmPhoneStatusUpdateTable. This integer value will wrap if needed.
::= { ccmPhoneStatusUpdateEntry 1 }

ccmPhoneStatusPhoneIndex OBJECT-TYPE
SYNTAX  CcmIndexOrZero
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
A positive value of this index is used to identify an entry in the ccmPhoneTable. A value of zero
indicates that the index to the ccmPhoneTable is Unknown.
::= { ccmPhoneStatusUpdateEntry 2 }

ccmPhoneStatusUpdateTime OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The time of the phone's registration status change.
::= { ccmPhoneStatusUpdateEntry 3 }

ccmPhoneStatusUpdateType OBJECT-TYPE
SYNTAX INTEGER {
unknown(1),
phoneRegistered(2),
phoneUnregistered(3),
phonePartiallyregistered(4)
}
MAX-ACCESS  read-only
STATUS current
DESCRIPTION
States the type of phone status change.
unknown: Unknown status
phoneRegistered: Phone has registered with the Cisco Unified CM
phoneUnregistered: Phone is no longer registered with the Cisco Unified CM
phonePartiallyregistered: Phone is partially registered with the Cisco Unified CM
::= { ccmPhoneStatusUpdateEntry 4 }
ccmPhoneStatusUnregReason OBJECT-TYPE
SYNTAX CcmDevUnregCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with unregistered phone.
::= { ccmPhoneStatusUpdateEntry 6 }
ccmPhoneStatusRegFailReason OBJECT-TYPE
SYNTAX CcmDevRegFailCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with registration failed phone.
::= { ccmPhoneStatusUpdateEntry 7 }

Enhanced Phone Extension Table with Combination Index
ccmPhoneExtnTable OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmPhoneExtnEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing the list of all phone extensions associated with the registered and unregistered phones in the ccmPhoneTable. This table has combination index ccmPhoneIndex, ccmPhoneExtnIndex so the ccmPhoneTable and the ccmPhoneExtnTable entries can be related.
::= { ccmPhoneInfo 5 }
ccmPhoneExtnEntry OBJECT-TYPE
SYNTAX  CcmPhoneExtnEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the phone extension Table, containing the information about a particular phone extension.

INDEX  { ccmPhoneIndex, ccmPhoneExtnIndex }
 ::=  { ccmPhoneExtnTable 1 }

CcmPhoneExtnEntry ::= SEQUENCE {
  ccmPhoneExtnIndex CcmIndex,
  ccmPhoneExtn SnmpAdminString,
  ccmPhoneExtnMultiLines   Unsigned32,
  ccmPhoneExtnInetAddressType InetAddressType,
  ccmPhoneExtnInetAddress InetAddress,
  ccmPhoneExtnStatus   CcmDeviceLineStatus
}

cmPhoneExtnIndex OBJECT-TYPE
  SYNTAX  CcmIndex
  MAX-ACCESS  not-accessible
  STATUS  current
  DESCRIPTION
  An arbitrary integer, selected by the local Cisco Unified CM, that uniquely identifies a Phone Extension within the Cisco Unified CM.
  ::=  { ccmPhoneExtnEntry 1 }

cmPhoneExtn OBJECT-TYPE
  SYNTAX  SnmpAdminString (SIZE(0..24))
  MAX-ACCESS  read-only
  STATUS  current
  DESCRIPTION
  The extension number of the extension.
  ::=  { ccmPhoneExtnEntry 2 }

cmPhoneExtnMultiLines OBJECT-TYPE
  SYNTAX  Unsigned32
  MAX-ACCESS  read-only
  STATUS  current
  DESCRIPTION
  The number of multiline appearances for each phone extension.
  ::=  { ccmPhoneExtnEntry 3 }

cmPhoneExtnInetAddressType OBJECT-TYPE
  SYNTAX  InetAddressType
  MAX-ACCESS  read-only
  STATUS  current
DESCRIPTION
This object identifies the IP address type of the phone extension.
::= { ccmPhoneExtnEntry 4 }

ccmPhoneExtnInetAddress OBJECT-TYPE
SYNTAX  InetAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the IP address of the phone extension. The type of address for this is identified by ccmPhoneExtnInetAddressType.
::= { ccmPhoneExtnEntry 5 }

ccmPhoneExtnStatus OBJECT-TYPE
SYNTAX  CcmDeviceLineStatus
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Represents the status of this phone line.
::= { ccmPhoneExtnEntry 6 }

Gateway Table

ccmGatewayTable OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmGatewayEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing the list of all gateway devices that have tried to register with the local Cisco Unified CM at least once. When the local Cisco Unified CM is restarted, this table will be refreshed.
::= { ccmGatewayInfo 1 }

ccmGatewayEntry OBJECT-TYPE
SYNTAX  CcmGatewayEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the gateway Table, one for each gateway device in the Cisco Unified CM.
INDEX  { ccmGatewayIndex }
::= { ccmGatewayTable 1 }
CcmGatewayEntry ::= SEQUENCE {


ccmGatewayIndex CcmIndex,
ccmGatewayName SnmpAdminString,
ccmGatewayType Integer,
ccmGateway Description SnmpAdminString,
ccmGatewayStatus CcmDeviceStatus,
ccmGatewayDevicePoolIndex CcmIndexOrZero,
ccmGatewayInetAddressType InetAddressType,
ccmGatewayInetAddress InetAddress,
ccmGatewayProductId CcmDeviceProductId,
ccmGatewayStatusReason CcmDevFailCauseCode,
ccmGatewayTimeLastStatusUpdt DateAndTime,
ccmGatewayTimeLastRegistered DateAndTime,
ccmGatewayDChannelStatus INTEGER,
ccmGatewayDChannelNumber Integer32,
ccmGatewayProductTypeIndex CcmIndexOrZero
ccmGatewayUnregReason CcmDevUnregCauseCode,
ccmGatewayRegFailReason CcmDevRegFailCauseCode
}

ccmGatewayIndex OBJECT-TYPE
SYNTAX CcmIndex
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that uniquely identifies a Gateway within the scope of the local call manager.
 ::= { ccmGatewayEntry 1 }

ccmGatewayName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This is the Gateway name assigned to the Gateway in the Cisco Unified CM. This name is assigned when a new device of type Gateway is added to the Cisco Unified CM.
 ::= { ccmGatewayEntry 2 }

ccmGatewayType OBJECT-TYPE
SYNTAX INTEGER
unknown(1),
other(2),
ciscoAnalogAccess(3),
ciscoDigitalAccessPRI(4),
ciscoDigitalAccessT1(5),
ciscoDigitalAccessPRIPlus(6),
ciscoDigitalAccessWSX6608E1(7),
ciscoDigitalAccessWSX6608T1(8),
ciscoAnalogAccessWSX6624(9),
ciscoMGCPStation(10),
ciscoDigitalAccessE1Plus(11),
ciscoDigitalAccessT1Plus(12),
ciscoDigitalAccessWSX6608PRI(13),
ciscoAnalogAccessWSX6612(14),
ciscoMGCPTrunk(15),
ciscoVG200(16),
cisco26XX(17),
cisco362X(18),
cisco364X(19),
cisco366X(20),
ciscoCat4224VoiceGatewaySwitch(21),
ciscoCat4000AccessGatewayModule(22),
ciscoAD2400(23),
ciscoVGCEndPoint(24),
ciscoVG224VG248Gateway(25),
ciscoVGCBox(26),
ciscoATA186(27),
ciscoICS77XXMRP2XX(28),
ciscoICS77XXASI81(29),
ciscoICS77XXASI160(30),
ciscoSlotVGCPort(31),
ciscoCat6000AVVIDServModule(32),
ciscoWSX6600(33),
ciscoWSSVCCMMMS(34),
cisco3745(35),
cisco3725(36),
ciscoICS77XXMRP3XX(37),
ciscoICS77XXMRP38FXS(38),
ciscoICS77XXMRP316FXS(39),
ciscoICS77XXMRP38FXOM1(40),
Cisco269X(41),
cisco1760(42),
cisco1751(43),

**ccmGatewayDescription OBJECT-TYPE**

SYNTAX SnmpAdminString (SIZE(0..255))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The description attached to the gateway device.
::= { ccmGatewayEntry 4 }

**ccmGatewayStatus OBJECT-TYPE**

SYNTAX CcmDeviceStatus
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The status of the gateway. The Gateway status changes from Unknown to Registered when the Gateway registers itself with the local Cisco Unified CM.
::= { ccmGatewayEntry 5 }

**ccmGatewayDevicePoolIndex OBJECT-TYPE**

SYNTAX CcmIndexOrZero
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A positive value of this index is used to identify the Device Pool to which this Gateway entry belongs. A value of zero indicates that the index to the Device Pool table is Unknown.
::= { ccmGatewayEntry 6 }

**ccmGatewayInetAddressType OBJECT-TYPE**

SYNTAX InetAddressType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the IP address type of the Gateway device. The value of this object is 'unknown(0)' if the IP address of a Gateway device is not available.
::= { ccmGatewayEntry 7 }

**ccmGatewayInetAddress OBJECT-TYPE**

SYNTAX InetAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies last known IP Address of the gateway. If the IP address is not available then
this object contains an empty string. The type of address for this is identified by
ccmGatewayInetAddressType.
::= { ccmGatewayEntry 8 }

ccmGatewayTimeLastStatusUpdt OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The time the status of the gateway changed.
::= { ccmGatewayEntry 11 }

ccmGatewayTimeLastRegistered OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The time the gateway last registered with the call manager.
::= { ccmGatewayEntry 12 }

ccmGatewayDChannelStatus OBJECT-TYPE
SYNTAX INTEGER
active(1),
inActive(2),
unknown(3),
notApplicable(4)
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The D-Channel status of the gateway.
active(1): The D-Channel is up
inActive(1): The D-Channel is down
unknown(3): The D-Channel status is unknown
notApplicable(4): The D-channel status is not applicable for this gateway.
::= { ccmGatewayEntry 13 }

ccmGatewayDChannelNumber OBJECT-TYPE
SYNTAX Integer32 (-1..24)
MAX-ACCESS read-only
<table>
<thead>
<tr>
<th>STATUS</th>
<th>current</th>
</tr>
</thead>
<tbody>
<tr>
<td>DESCRIPTION</td>
<td></td>
</tr>
<tr>
<td>The D-Channel number of the gateway. A value of -1 in this field indicates that the DChannel number is not applicable for this gateway.</td>
<td></td>
</tr>
<tr>
<td>::= { ccmGatewayEntry 14 }</td>
<td></td>
</tr>
</tbody>
</table>

**ccmGatewayProductTypeIndex OBJECT-TYPE**

<table>
<thead>
<tr>
<th>SYNTAX</th>
<th>CcmIndexOrZero</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX-ACCESS</td>
<td>read-only</td>
</tr>
<tr>
<td>STATUS</td>
<td>current</td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td></td>
</tr>
<tr>
<td>A positive value of this index is used to identify the related product type entry in the ccmProductTypeTable. A value of 0 indicates that the index to the ccmProductTypeTable is Unknown.</td>
<td></td>
</tr>
<tr>
<td>::= { ccmGatewayEntry 15 }</td>
<td></td>
</tr>
</tbody>
</table>

**ccmGatewayUnregReason OBJECT-TYPE**

<table>
<thead>
<tr>
<th>SYNTAX</th>
<th>CcmDevUnregCauseCode</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX-ACCESS</td>
<td>read-only</td>
</tr>
<tr>
<td>STATUS</td>
<td>current</td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td></td>
</tr>
<tr>
<td>The reason code associated with unregistered gateway.</td>
<td></td>
</tr>
<tr>
<td>::= { ccmGatewayEntry 16 }</td>
<td></td>
</tr>
</tbody>
</table>

**ccmGatewayRegFailReason OBJECT-TYPE**

<table>
<thead>
<tr>
<th>SYNTAX</th>
<th>CcmDevRegFailCauseCode</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX-ACCESS</td>
<td>read-only</td>
</tr>
<tr>
<td>STATUS</td>
<td>current</td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td></td>
</tr>
<tr>
<td>The reason code associated with registration failed gateway.</td>
<td></td>
</tr>
<tr>
<td>::= { ccmGatewayEntry 17 }</td>
<td></td>
</tr>
</tbody>
</table>

**Gateway Trunk Table**

**CcmGatewayTrunkEntry**

 ::= SEQUENCE {
  ccmGatewayTrunkIndex CcmIndex,
  ccmGatewayTrunkType INTEGER,
  ccmGatewayTrunkName SnmpAdminString,
  ccmTrunkGatewayIndex CcmIndexOrZero,
  ccmGatewayTrunkStatus INTEGER
}
All Scalar Objects

ccmRegisteredPhones OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of phones that are registered and actively in communication with the local call manager.
 ::= { ccmGlobalInfo 5 }

ccmUnregisteredPhones OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of phones that are unregistered or have lost contact with the local call manager.
 ::= { ccmGlobalInfo 6 }

ccmRejectedPhones OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of phones whose registration requests were rejected by the local call manager.
 ::= { ccmGlobalInfo 7 }

ccmRegisteredGateways OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of gateways that are registered and actively in communication with the local call manager.
 ::= { ccmGlobalInfo 8 }

ccmUnregisteredGateways OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of gateways that are unregistered or have lost contact with the local call manager.
 ::= { ccmGlobalInfo 9 }
ccmRejectedGateways OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of gateways whose registration requests were rejected by the local call manager.
 ::= { ccmGlobalInfo 10 }

ccmRegisteredMediaDevices OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of media devices that are registered and actively in communication with the local call manager.
 ::= { ccmGlobalInfo 11 }

ccmUnregisteredMediaDevices OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of media devices that are unregistered or have lost contact with the local call manager.
 ::= { ccmGlobalInfo 12 }

ccmRejectedMediaDevices OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of media devices whose registration requests were rejected by the local call manager.
 ::= { ccmGlobalInfo 13 }

ccmRegisteredCTIDevices OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of CTI devices that are registered and actively in communication with the local call manager.
 ::= { ccmGlobalInfo 14 }

ccmUnregisteredCTIDevices OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of CTI devices that are unregistered or have lost contact with the local call manager.
::= { ccmGlobalInfo 15 }

ccmRejectedCTIDevices OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of CTI devices whose registration requests were rejected by the local call manager.
::= { ccmGlobalInfo 16 }

ccmRegisteredVoiceMailDevices OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of voice messaging devices that are registered and actively in communication with the local call manager.
::= { ccmGlobalInfo 17 }

ccmUnregisteredVoiceMailDevices OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of voice messaging devices that are unregistered or have lost contact with the local call manager.
::= { ccmGlobalInfo 18 }

ccmRejectedVoiceMailDevices OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of voice messaging devices whose registration requests were rejected by the local call manager.
::= { ccmGlobalInfo 19 }

ccmCallManagerStartTime OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The last time the local call manager service started. This is available only when the local call
manager is up and running.
::= { ccmGlobalInfo 20 }

ccmPhoneTableStateId OBJECT-TYPE
SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The current state of ccmPhoneTable. The initial value of this object is 0 and it will be incremented
every time when there is a change (addition/deletion/modification) to the ccmPhoneTable. This
value and ccmCallManagerStartTime should be used together to find if the table has changed or not.
When the call manager is restarted, this will be reset to 0.
::= { ccmGlobalInfo 21 }

ccmPhoneExtensionTableStateId OBJECT-TYPE
SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The current state of ccmPhoneExtensionTable. The initial value of this object is 0 and it will be incremented
every time when there is a change (addition/deletion/modification) to the ccmPhoneExtensionTable. This
value and ccmCallManagerStartTime should be used together to find if the table has changed or not. When the call manager is restarted, this will be reset to 0.
::= { ccmGlobalInfo 22 }

ccmPhoneStatusUpdateTableStateId OBJECT-TYPE
SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The current state of ccmPhoneStatusUpdateTable. The initial value of this object is 0 and it will be incremented
every time when there is a change (addition/deletion/modification) to the ccmPhoneStatusUpdateTable. This value and sysUpTime should be used together to find if the table has changed or not. When the SNMP service is restarted this value will be reset to 0.
::= { ccmGlobalInfo 23 }

ccmGatewayTableStateId OBJECT-TYPE
SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-only
Chapter 7       Cisco Management Information Base

CISCO-CCM-MIB

STATUS  current
DESCRIPTION
The current state of ccmGatewayTable. The initial value of this object is 0 and it will be incremented every time when there is a change (addition/deletion/modification) to the ccmGatewayTable. This value and ccmCallManagerStartTime should be used together to find if the table has changed or not. When the call manager is restarted, this will be reset to 0.

::= { ccmGlobalInfo 24 }

ccmCTIDeviceTableStateId OBJECT-TYPE
SYNTAX  Integer32 (0..2147483647)
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The current state of ccmCTIDeviceTable. The initial value of this object is 0 and it will be incremented every time when there is a change (addition/deletion/modification) to the ccmCTIDeviceTable. This value and ccmCallManagerStartTime should be used together to find if the table has changed or not. When the call manager is restarted, this will be reset to 0.

::= { ccmGlobalInfo 25 }

ccmCTIDeviceDirNumTableStateId OBJECT-TYPE
SYNTAX  Integer32 (0..2147483647)
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The current state of ccmCTIDeviceDirNumTable. The initial value of this object is 0 and it will be incremented every time when there is a change (addition/deletion/modification) to the ccmCTIDeviceDirNumTable. This value and ccmCallManagerStartTime should be used together to find if the table has changed or not. When the call manager is restarted, this will be reset to 0.

::= { ccmGlobalInfo 26 }

ccmPhStatUpdtTblLastAddedIndex OBJECT-TYPE
SYNTAX  CcmIndexOrZero
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The ccmPhoneStatusUpdateIndex value of the last entry that was added to the ccmPhoneStatusUpdateTable. This value together with sysUpTime can be used by the manager applications to identify the new entries in the ccmPhoneStatusUpdateTable since their last poll. This value need not be the same as the highest index in the ccmPhoneStatusUpdateTable as the index could have wrapped around. The initial value of this object is 0, which indicates that no entries have been added to this table. When the SNMP service is restarted this value will be reset to 0.

::= { ccmGlobalInfo 27 }

ccmPhFailedTblLastAddedIndex OBJECT-TYPE
SYNTAX  CcmIndexOrZero
MAX-ACCESS  read-only
DESCRIPTION
The ccmPhoneFailedIndex value of the last entry that was added to the ccmPhoneFailedTable. This value together with sysUpTime can be used by the manager applications to identify the new entries in the ccmPhoneFailedTable since their last poll. This value need not be the same as the highest index in the ccmPhoneFailedTable as the index could have wrapped around. The initial value of this object is 0, which indicates that no entries have been added to this table. When the SNMP service is restarted this value will be reset to 0.
The `ccmPhoneFailedIndex` value of the last entry that was added to the `ccmPhoneFailedTable`. This value together with `sysUpTime` can be used by the manager applications to identify the new entries in the `ccmPhoneFailedTable` since their last poll. This value need not be the same as the highest index in the `ccmPhoneFailedTable` as the index could have wrapped around. The initial value of this object is 0, which indicates that no entries have been added to this table. When the SNMP service is restarted this value will be reset to 0.

```plaintext
c::= { ccmGlobalInfo 28 }
```

**ccmSystemVersion** OBJECT-TYPE

- **SYNTAX**: `SnmpAdminString (SIZE(0..128))`
- **MAX-ACCESS**: read-only
- **STATUS**: current
- **DESCRIPTION**: The installed version of the local Cisco Unified CM system.

```plaintext
c::= { ccmGlobalInfo 29 }
```

**ccmInstallationId** OBJECT-TYPE

- **SYNTAX**: `SnmpAdminString (SIZE(0..128))`
- **MAX-ACCESS**: read-only
- **STATUS**: current
- **DESCRIPTION**: The installation component identifier of the local Cisco Unified CM component (`ccm.exe`).

```plaintext
c::= { ccmGlobalInfo 30 }
```

**ccmPartiallyRegisteredPhones** OBJECT-TYPE

- **SYNTAX**: `Counter32`
- **MAX-ACCESS**: read-only
- **STATUS**: current
- **DESCRIPTION**: The number of phones that are partially registered with the local Cisco Unified CM.

```plaintext
c::= { ccmGlobalInfo 31 }
```

**ccmH323TableEntries** OBJECT-TYPE

- **SYNTAX**: `Integer32 (0..2147483647)`
- **MAX-ACCESS**: read-only
- **STATUS**: current
- **DESCRIPTION**: The current number of entries in `ccmH323DeviceTable`. The initial value of this object is 0 and it will be incremented every time when there is an addition to the `ccmH323DeviceTable`. When the Cisco Unified CM is restarted, this will be reset to 0.

```plaintext
c::= { ccmGlobalInfo 32 }
```

**ccmSIPTableEntries** OBJECT-TYPE
SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The current number of entries in ccmSIPDeviceTable. The initial value of this object is 0 and it will be incremented every time when there is an addition to the ccmSIPDeviceTable. When the Cisco Unified CM is restarted, this will be reset to zero.
 ::= { ccmGlobalInfo 33 }

Media Device Table

ccmMediaDeviceTable OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmMediaDeviceEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing a list of all Media Devices that have tried to register with the local Cisco Unified CM at least once. When the local Cisco Unified CM is restarted, this table will be refreshed.
 ::= { ccmMediaDeviceInfo 1 }

ccmMediaDeviceEntry OBJECT-TYPE
SYNTAX  CcmMediaDeviceEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the MediaDevice Table, containing the information about a particular Media Resource device.
INDEX  { ccmMediaDeviceIndex }
 ::= { ccmMediaDeviceTable 1 }
CcmMediaDeviceEntry ::= SEQUENCE {
  ccmMediaDeviceIndex CcmIndex,
  ccmMediaDeviceName SnmpAdminString,
  ccmMediaDeviceType INTEGER,
  ccmMediaDeviceDescription SnmpAdminString,
  ccmMediaDeviceDeviceStatus CcmDeviceStatus,
  ccmMediaDeviceDevicePoolIndex CcmIndexOrZero,
  ccmMediaDeviceInetAddressType InetAddressType,
  ccmMediaDeviceInetAddress InetAddress,
  ccmMediaDeviceStatusReason CcmDevFailCauseCode,
ccmMediaDeviceTimeLastStatusUpdt DateAndTime,
ccmMediaDeviceTimeLastRegistered DateAndTime,
ccmMediaDeviceProductTypeIndex CcmIndexOrZero
ccmMediaDeviceInetAddressIPv4 InetAddressIPv4,
ccmMediaDeviceInetAddressIPv6 InetAddressIPv6,
ccmMediaDeviceUnregReason CcmDevUnregCauseCode,
ccmMediaDeviceRegFailReason CcmDevRegFailCauseCode
}

ccmMediaDeviceIndex OBJECT-TYPE
SYNTAX CcmIndex
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that identifies a Media Device entry in the table.
 ::= { ccmMediaDeviceEntry 1 }

ccmMediaDeviceName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This is the device name assigned to the Media Device. This name is assigned when a new device of this type is added to the Cisco Unified CM.
 ::= { ccmMediaDeviceEntry 2 }

ccmMediaDeviceDescription OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This description is given when the device is configured in the Cisco Unified CM.
 ::= { ccmMediaDeviceEntry 4 }

ccmMediaDeviceStatus OBJECT-TYPE
SYNTAX CcmDeviceStatus
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The status of the Media Device. The status changes from unknown to registered when it registers itself with the local Cisco Unified CM.
::= { ccmMediaDeviceEntry 5 }

ccmMediaDeviceDevicePoolIndex OBJECT-TYPE
SYNTAX  CcmIndexOrZero
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
A positive value of this index is used to identify the Device Pool to which this MediaDevice entry
belongs. A value of zero indicates that the index to the Device Pool table is Unknown.
::= { ccmMediaDeviceEntry 6 }

ccmMediaDeviceTimeLastStatusUpdt OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The time the status of the media device changed.
::= { ccmMediaDeviceEntry 10 }

ccmMediaDeviceTimeLastRegistered OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The time the media device last registered with the call manager.
::= { ccmMediaDeviceEntry 11 }

ccmMediaDeviceProductTypeIndex OBJECT-TYPE
SYNTAX  CcmIndexOrZero
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
A positive value of this index is used to identify the related product type entry in the
ccmProductTypeTable. A value of zero indicates that the index to the ccmProductTypeTable is
Unknown.
::= { ccmMediaDeviceEntry 12 }

ccmMediaDeviceInetAddressIPv4 OBJECT-TYPE
SYNTAX InetAddressIPv4
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the last known primary IPv4 address of the Media Device. This object contains
value zero if IPV4 address is not available.
::= { ccmMediaDeviceEntry 13 }

**ccmMediaDeviceInetAddressIPv6** OBJECT-TYPE
SYNTAX InetAddressIPv6
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the last known primary IPv6 address of the Media Device. This object contains value zero if IPV6 address is not available.
::= { ccmMediaDeviceEntry 14 }

**ccmMediaDeviceUnregReason** OBJECT-TYPE
SYNTAX CcmDevUnregCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with unregistered Media Device.
::= { ccmMediaDeviceEntry 15 }

**ccmMediaDeviceRegFailReason** OBJECT-TYPE
SYNTAX CcmDevRegFailCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with registration failed Media Device.
::= { ccmMediaDeviceEntry 16 }

**CTI Device Table**

**ccmCtiDeviceTable** OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmCtiDeviceEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION—The table containing a list of all CTI (Computer Telephony Integration) Devices that have tried to register with the local Cisco Unified CM at least once. When the local Cisco Unified CM is restarted, this table will be refreshed.
::= { ccmCtiDeviceInfo 1 }

**ccmCtiDeviceEntry** OBJECT-TYPE
SYNTAX  CcmCtiDeviceEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION—An entry (conceptual row) in the CTIDevice Table, containing the information about a particular CTI Device.

INDEX  { ccmCTIDeviceIndex }
 ::= { ccmCTIDeviceTable 1 }

CcmCTIDeviceEntry ::= SEQUENCE {
    ccmCTIDeviceIndex CcmIndex,
    ccmCTIDeviceName SnmpAdminString,
    ccmCTIDeviceType INTEGER,
    ccmCTIDeviceDescription SnmpAdminString,
    ccmCTIDeviceStatus CcmDeviceStatus,
    ccmCTIDevicePoolIndex CcmIndexOrZero,
    ccmCTIDeviceInetAddressType [DEPRECATED] InetAddressType,
    ccmCTIDeviceInetAddress [DEPRECATED] InetAddress,
    ccmCTIDeviceAppInfo SnmpAdminString,
    ccmCTIDeviceStatusReason CcmDevFailCauseCode,
    ccmCTIDeviceTimeLastStatusUpdt DateAndTime,
    ccmCTIDeviceTimeLastRegistered DateAndTime,
    ccmCTIDeviceProductTypeIndex CcmIndexOrZero,
    ccmCTIDeviceInetAddressIPv4 InetAddressIPv4
    ccmCTIDeviceInetAddressIPv6 InetAddressIPv6
    ccmCTIDeviceUnregReason CcmDevUnregCauseCode,
    ccmCTIDeviceRegFailReason CcmDevRegFailCauseCode
}

ccmCTIDeviceIndex OBJECT-TYPE
    SYNTAX  CcmIndex
    MAX-ACCESS not-accessible
    STATUS  current
    DESCRIPTION
    An arbitrary integer, selected by the local Cisco Unified CM, that identifies a CTI Device entry in the table.
    ::= { ccmCTIDeviceEntry 1 }

ccmCTIDeviceName OBJECT-TYPE
    SYNTAX  SnmpAdminString (SIZE(0..64))
    MAX-ACCESS read-only
    STATUS  current
    DESCRIPTION
    The name of the CTI Device. This name is assigned to the CTI Device when it is added to the Cisco Unified CM.
::= { ccmCTIDeviceEntry 2 }

\textbf{ccmCTIDeviceDescription \text{OBJECT-TYPE}}
\begin{itemize}
  \item SYNTAX SnmpAdminString (SIZE(0..128))
  \item MAX-ACCESS read-only
  \item STATUS current
  \item DESCRIPTION
  A description of the CTI Device. This description is given when the CTI Device is configured in the Cisco Unified CM.
\end{itemize}
::= { ccmCTIDeviceEntry 4 }

\textbf{ccmCTIDeviceStatus \text{OBJECT-TYPE}}
\begin{itemize}
  \item SYNTAX CcmDeviceStatus
  \item MAX-ACCESS read-only
  \item STATUS current
  \item DESCRIPTION
  The status of the CTI Device. The CTI Device status changes from unknown to registered when it registers itself with the local Cisco Unified CM.
\end{itemize}
::= { ccmCTIDeviceEntry 5 }

\textbf{ccmCTIDevicePoolIndex \text{OBJECT-TYPE}}
\begin{itemize}
  \item SYNTAX CcmIndexOrZero
  \item MAX-ACCESS read-only
  \item STATUS current
  \item DESCRIPTION
  A positive value of this index is used to identify the Device Pool to which this CTI Device entry belongs. A value of zero indicates that the index to the Device Pool table is Unknown.
\end{itemize}
::= { ccmCTIDeviceEntry 6 }

\textbf{ccmCTIDeviceTimeLastStatusUpdt \text{OBJECT-TYPE}}
\begin{itemize}
  \item SYNTAX DateAndTime
  \item MAX-ACCESS read-only
  \item STATUS current
  \item DESCRIPTION
  The time the status of the CTI device changed.
\end{itemize}
::= { ccmCTIDeviceEntry 11 }

\textbf{ccmCTIDeviceTimeLastRegistered \text{OBJECT-TYPE}}
\begin{itemize}
  \item SYNTAX DateAndTime
  \item MAX-ACCESS read-only
  \item STATUS current
  \item DESCRIPTION
  The time the CTI Device last registered with the call manager.
::= { ccmCTIDeviceEntry 12 }

ccmCTIDeviceProductTypeIndex OBJECT-TYPE
SYNTAX  CcmIndexOrZero
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A positive value of this index is used to identify the related product type entry in the ccmProductTypeTable. A value of 0 indicates that the index to the ccmProductTypeTable is Unknown.
::= { ccmCTIDeviceEntry 13 }

ccmCTIDeviceInetAddressIPv4 OBJECT-TYPE
SYNTAX InetAddressIPv4
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies IPv4 Address of the host where this CTI Device is running. If the IPv4 address is not available then this object contains an empty string.
::= { ccmCTIDeviceEntry 14 }

ccmCTIDeviceInetAddressIPv6 OBJECT-TYPE
SYNTAX InetAddressIPv6
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies IPv6 Address of the host where this CTI Device is running. If the IPv6 address is not available then this object contains an empty string.
::= { ccmCTIDeviceEntry 15 }

ccmCTIDeviceUnregReason OBJECT-TYPE
SYNTAX CcmDevUnregCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with unregistered CTI Device.
::= { ccmCTIDeviceEntry 16 }

ccmCTIDeviceRegFailReason OBJECT-TYPE
SYNTAX CcmDevRegFailCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with registration failed CTI Device.
::= { ccmCTIDeviceEntry 17 }

CTI Device Directory Number Table

cctmCTIDeviceDirNumTable OBJECT-TYPE
   SYNTAX  SEQUENCE OF CcmCTIDeviceDirNumEntry
   MAX-ACCESS  not-accessible
   STATUS  current
   DESCRIPTION
   The table containing a list of directory numbers that are assigned to all of the registered and
   unregistered CTI Devices in the ccmCTIDeviceTable.
   ::= { ccmCTIDeviceInfo 2 }

cctmCTIDeviceDirNumEntry OBJECT-TYPE
   SYNTAX  CcmCTIDeviceDirNumEntry
   MAX-ACCESS  not-accessible
   STATUS  current
   DESCRIPTION
   An entry (conceptual row) in the CTIDeviceDirNum Table, containing the information about a
   particular CTI Device extension.
   INDEX  { ccmCTIDeviceIndex, ccmCTIDeviceDirNumIndex }
   ::= { ccmCTIDeviceDirNumTable 1 }

CcmCTIDeviceDirNumEntry ::= SEQUENCE {
   ccmCTIDeviceDirNumIndex CcmIndex,
   ccmCTIDeviceDirNum SnmpAdminString
}

cctmCTIDeviceDirNumIndex OBJECT-TYPE
   SYNTAX  CcmIndex
   MAX-ACCESS  not-accessible
   STATUS  current
   DESCRIPTION
   An arbitrary integer, selected by the local system, that identifies a Directory Number of a CTI
   Device.
   ::= { ccmCTIDeviceDirNumEntry 1 }

cctmCTIDeviceDirNum OBJECT-TYPE
   SYNTAX  SnmpAdminString (SIZE(0..24))
   MAX-ACCESS  read-only
   STATUS  current
   DESCRIPTION
   A Directory Number of the CTI Device.
Alarms

Cisco Unified CM Alarm Enable

ccmCallManagerAlarmEnable OBJECT-TYPE
SYNTAX  TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Allows the generation of alarms in response to Cisco Unified CM general failures.
true(1): Enabling this object will allow the Cisco Unified CM agent to generate the following alarms:
   ccmCallManagerFailure,
   ccmMediaResourceListExhausted,
   ccmRouteListExhausted and
   ccmTLSConnectionFailure. This is the default value.
false(2): Disabling this object will stop the generation of the following alarms by the Cisco Unified CM agent:
   ccmCallManagerFailure
   ccmMediaResourceListExhausted,
   ccmRouteListExhausted and
   ccmTLSConnectionFailure.
DEFVAL { true }
 ::= { ccmAlarmConfigInfo 1 }

Phone Failed Config Objects

ccmPhoneFailedAlarmInterval OBJECT-TYPE
SYNTAX  Integer32 (0 | 30..3600)
UNITS seconds
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The minimum interval between sending of the ccmPhoneFailed notification in seconds. The ccmPhoneFailed notification is only sent when there is at least one entry in the ccmPhoneFailedTable and the notification has not been sent for the last ccmPhoneFailedAlarmInterval defined in this object. A value of zero indicates that the alarm notification is disabled.

DEFVAL { 0 }
::= { ccmAlarmConfigInfo 2 }

ccmPhoneFailedStorePeriod OBJECT-TYPE
SYNTAX Integer32 (1800..3600)
UNITS seconds
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The time duration for storing each entry in the ccmPhoneFailedTable. The entries that have not been updated and kept at least this period will be deleted. This value should ideally be set to a higher value than the ccmPhoneFailedAlarmInterval object.

DEFVAL { 1800 }
::= { ccmAlarmConfigInfo 3 }

Phone Status Update Config Objects

ccmPhoneStatusUpdateAlarmInterv OBJECT-TYPE
SYNTAX Integer32 (0 | 30..3600)
UNITS seconds
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The minimum interval between sending of the ccmPhoneStatusUpdate notification in seconds. The ccmPhoneStatusUpdate notification is only sent when there is at least one entry in the ccmPhoneStatusUpdateTable and the notification has not been sent for the last ccmPhoneStatusUpdateAlarmInterval defined in this object. A value of zero indicates that the alarm notification is disabled.

DEFVAL { 0 }
::= { ccmAlarmConfigInfo 4 }

ccmPhoneStatusUpdateStorePeriod OBJECT-TYPE
SYNTAX Integer32 (1800..3600)
UNITS seconds
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The time duration for storing each entry in the ccmPhoneStatusUpdateTable. The entries that have been kept at least this period will be deleted. This value should ideally be set to a higher value than the ccmPhoneStatusUpdateAlarmInterv object.

DEFVAL { 1800 }
::= { ccmAlarmConfigInfo 5 }

**Gateway Alarm Enable**

ccmGatewayAlarmEnable OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Allows the generation of alarms in response to Gateway general failures that the Cisco Unified CM is aware of.

true(1): Enabling this object will allow the Cisco Unified CM agent to generate the following alarms:

- ccmGatewayFailedReason
- ccmGatewayLayer2Change (This is the default value.)

false(2): Disabling this object will stop the generation of the following alarms by the Cisco Unified agent:

- ccmGatewayFailed
- ccmGatewayLayer2Change.

DEFVAL { true }
::= { ccmAlarmConfigInfo 6 }

**Malicious Call Alarm Enable**

ccmMaliciousCallAlarmEnable OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Allows the generation of alarms for malicious calls that the local call manager is aware of.

true(1): Enabling this object will allow the Cisco Unified CM agent to generate the ccmMaliciousCall alarm. This is the default value.

false(2): Disabling this object will stop the generation of the ccmMaliciousCall alarm.

DEFVAL { true }
::= { ccmAlarmConfigInfo 7 }
Notification and Alarms

ccmAlarmSeverity OBJECT-TYPE
SYNTAX INTEGER {
  emergency(1),
  alert(2),
  critical(3),
  error(4),
  warning(5),
  notice(6),
  informational(7)
}
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The Alarm Severity code.
  emergency: System unusable
  alert: Immediate response needed
  critical: Critical condition
  error: Error condition
  warning: Warning condition
  notice: Normal but significant condition
  informational: Informational situation.
 ::= { ccmNotificationsInfo 1 }

ccmFailCauseCode OBJECT-TYPE
SYNTAX INTEGER {
  unknown(1),
  heartBeatStopped(2),
  routerThreadDied(3),
  timerThreadDied(4),
  criticalThreadDied(5),
  deviceMgrInitFailed(6),
  digitAnalysisInitFailed(7),
  callControlInitFailed(8),
  linkMgrInitFailed(9),
  dbMgrInitFailed(10),
  msgTranslatorInitFailed(11),
  suppServicesInitFailed(12)
The Cause code of the failure. This cause is derived from a monitoring thread in the Cisco Unified CM or from a heartbeat monitoring process.

unknown: Unknown
heartBeatStopped: The Cisco Unified CM stops generating a heartbeat
routerThreadDied: The Cisco Unified CM detects the death of the router thread
timerThreadDied: The Cisco Unified CM detects the death of the timer thread
criticalThreadDied: The Cisco Unified CM detects the death of one of its critical threads
deviceMgrInitFailed: The Cisco Unified CM fails to start its device manager subsystem
digitAnalysisInitFailed: The Cisco Unified CM fails to start its digit analysis subsystem
callControlInitFailed: The Cisco Unified CM fails to start its call control subsystem
linkMgrInitFailed: The Cisco Unified CM fails to start its link manager subsystem
dbMgrInitFailed: The Cisco Unified CM fails to start its database manager subsystem
msgTranslatorInitFailed: The Cisco Unified CM fails to start its message translation manager subsystem
suppServicesInitFailed: The Cisco Unified CM fails to start its supplementary services subsystem.

::= { ccmNotificationsInfo 2 }

ccmPhoneFailures OBJECT-TYPE
SYNTAX  Unsigned32
MAX-ACCESS accessible-for-notify
STATUS  current
DESCRIPTION
The count of the phone initialization or communication failures that are stored in the ccmPhoneFailedTable object.
::= { ccmNotificationsInfo 3 }

ccmPhoneUpdates OBJECT-TYPE
SYNTAX  Unsigned32
MAX-ACCESS accessible-for-notify
STATUS  current
DESCRIPTION
The count of the phone status changes that are stored in the ccmPhoneStatusUpdateTable object.
::= { ccmNotificationsInfo 4 }

ccmMediaResourceType OBJECT-TYPE
SYNTAX  INTEGER {
unknown(1),
mediaTerminationPoint(2),
transcoder(3),
conferenceBridge(4),
musicOnHold(5)
}
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The type of media resource.
unknown: Unknown resource type
mediaTerminationPoint: Media Termination Point
transcoder: Transcoder
conferenceBridge: Conference Bridge
musicOnHold: Music On Hold.
 ::= { ccmNotificationsInfo 6 }

ccmMediaResourceListName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The name of a Media Resource List. This name is assigned when a new Media Resource List is added to the Cisco Unified CM.
 ::= { ccmNotificationsInfo 7 }

ccmRouteListName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The name of a Route List. This name is assigned when a new Route List is added to the Cisco Unified CM.
 ::= { ccmNotificationsInfo 8 }

ccmGatewayPhysIfIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
This object is the identifier of an interface in a gateway that has registered with the local Cisco Unified CM. On a DS1/E1 interface, this should be the same as the ifIndex value in the gateway.
::= { ccmNotificationsInfo 9 }

ccmGatewayPhysIfL2Status OBJECT-TYPE
SYNTAX INTEGER {
unknown(1),
up(2),
down(3)
}
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The layer 2 status of a physical interface in a gateway that has registered with the local Cisco Unified CM.
unknown: Unknown status
up: Interface is up
down: Interface is down.
::= { ccmNotificationsInfo 10 }

ccmMaliCallCalledPartyName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The display name of the called party who received the malicious call.
::= { ccmNotificationsInfo 11 }

ccmMaliCallCalledPartyNumber OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The phone number of the device where the malicious call is received.
::= { ccmNotificationsInfo 12 }

ccmMaliCallCalledDeviceName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The name of the device where the malicious call is received.
::= { ccmNotificationsInfo 13 }

ccmMaliCallCallingPartyName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS  current
DESCRIPTION
The display name of the caller whose call is registered as malicious with the local call manager.
::= { ccmNotificationsInfo 14 }

ccmMaliCallCallingPartyNumber OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS  current
DESCRIPTION
The phone number of the caller whose call is registered as malicious with the local call manager.
::= { ccmNotificationsInfo 15 }

ccmMaliCallCallingDeviceName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS  current
DESCRIPTION
The edge device name through which the malicious call originated or passed through.
::= { ccmNotificationsInfo 16 }

ccmMaliCallTime OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS accessible-for-notify
STATUS  current
DESCRIPTION
The time when the malicious call is detected by the local call manager.
::= { ccmNotificationsInfo 17 }

ccmQualityRprtSourceDevName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS  current
DESCRIPTION
The name of the source device from where the problem was reported.
::= { ccmNotificationsInfo 18 }

ccmQualityRprtClusterId OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS  current
DESCRIPTION
The cluster identifier of the source device.
::= { ccmNotificationsInfo 19 }

ccmQualityRprtCategory OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The category of the problem reported.
::= { ccmNotificationsInfo 20 }

ccmQualityRprtReasonCode OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The description of the problem reported.
::= { ccmNotificationsInfo 21 }

ccmQualityRprtTime OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The time when the problem was reported.
::= { ccmNotificationsInfo 22 }

ccmTLSDevName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The device for which TLS connection failure was reported.
::= { ccmNotificationsInfo 23 }

ccmTLSDevInetAddressType OBJECT-TYPE
SYNTAX  InetAddressType
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
This object identifies the type of address for the device for which TLS connection failure was reported.
::= { ccmNotificationsInfo 24 }

ccmTLSDevInetAddress OBJECT-TYPE
SYNTAX  InetAddress
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
This object identifies IP Address of the device, for which TLS connection failure was reported. The
type of address for this is identified by ccmTLSDevInetAddressType.
::= { ccmNotificationsInfo 25 }

ccmTLSConnFailTime OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The time when TLS connection failure was detected by the local Cisco Unified CM.
::= { ccmNotificationsInfo 26 }

ccmTLSConnectionFailReasonCode OBJECT-TYPE
SYNTAX  INTEGER {
  unknown (1),
  authenticationerror(2),
  invalidx509nameincertificate(3),
  invalidtlscipher(4)
}
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
The reason for connection failure.
::= { ccmNotificationsInfo 27 }

ccmGatewayRegFailCauseCode OBJECT-TYPE
SYNTAX  CcmDevRegFailCauseCode
MAX-ACCESS accessible-for-notify
STATUS current
DESCRIPTION
States the reason for a gateway device registration failure.
::= { ccmNotificationsInfo 28 }

H323 Device Table

ccmH323DeviceTable OBJECT-TYPE
SYNTAX SEQUENCE OF CcmH323DeviceEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing a list of all H323 devices in the Cisco Unified CM cluster that the local Cisco Unified CM is aware of. When the local Cisco Unified CM is restarted, this table will be refreshed.
::= { ccmH323DeviceInfo 1 }
ccmH323DeviceEntry OBJECT-TYPE
SYNTAX CcmH323DeviceEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the H323Device Table, containing the information about a particular H323 Device.
INDEX { ccmH323DevIndex }
::= { ccmH323DeviceTable 1 }
CcmH323DeviceEntry ::= SEQUENCE {
  ccmH323DevIndex CcmIndex,
  ccmH323DevName SnmpAdminString,
  ccmH323DevProductId CcmDeviceProductId,
  ccmH323DevDESCRIPTION SnmpAdminString,
  ccmH323DevInetAddressType InetAddressType,
  ccmH323DevInetAddress InetAddress,
  ccmH323DevCnfgGKInetAddressType InetAddressType,
  ccmH323DevCnfgGKInetAddress InetAddress,
  ccmH323DevAltGK1InetAddressType InetAddressType,
  ccmH323DevAltGK1InetAddress InetAddress,
  ccmH323DevAltGK2InetAddressType InetAddressType,
  ccmH323DevAltGK2InetAddress InetAddress,
  ccmH323DevAltGK3InetAddressType InetAddressType,
  ccmH323DevAltGK3InetAddress InetAddress,
  ccmH323DevAltGK4InetAddressType InetAddressType,
  ccmH323DevAltGK4InetAddress InetAddress,
  ccmH323DevAltGK5InetAddressType InetAddressType,
  ccmH323DevAltGK5InetAddress InetAddress,
  ccmH323DevActGKInetAddressType InetAddressType,
  ccmH323DevActGKInetAddress InetAddress,
  ccmH323DevStatus INTEGER,
ccmH323DevStatusReason  CcmDevFailCauseCode,
ccmH323DevTimeLastStatusUpdt DateAndTime,
ccmH323DevTimeLastRegistered DateAndTime,
ccmH323DevRmtCM1InetAddressType InetAddressType,
ccmH323DevRmtCM1InetAddress InetAddress,
ccmH323DevRmtCM2InetAddressType InetAddressType,
ccmH323DevRmtCM2InetAddress InetAddress,
ccmH323DevRmtCM3InetAddressType InetAddressType,
ccmH323DevRmtCM3InetAddress InetAddress,
ccmH323DevProductTypeIndex  CcmIndexOrZero
ccmH323DevUnregReason CcmDevUnregCauseCode,
ccmH323DevRegFailReason CcmDevRegFailCauseCode

ccmH323DevIndex OBJECT-TYPE
SYNTAX  CcmIndex
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that identifies a H323 Device entry in
the table.
::= { ccmH323DeviceEntry 1 }

ccmH323DevName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The device name assigned to the H323 Device. This name is assigned when a new H323 device is
added to the Cisco Unified CM.
::= { ccmH323DeviceEntry 2 }

ccmH323DevDESCRIPTION OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..255))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
A description of the H323 device. This description is given when the H323 device is configured in
the Cisco Unified CM.
::= { ccmH323DeviceEntry 4 }
ccmH323DevInetAddressType OBJECT-TYPE
  SYNTAX  InetAddressType
  MAX-ACCESS  read-only
  STATUS  current
  DESCRIPTION
  This object identifies the IP address type of the H323 device. The value of this object is 'unknown(0)' if the IP address of a H323 device is not available.
  ::= { ccmH323DeviceEntry 5 }

ccmH323DevInetAddress OBJECT-TYPE
  SYNTAX  InetAddress
  MAX-ACCESS  read-only
  STATUS  current
  DESCRIPTION
  This object identifies last known IP Address of the H323 device. If the IP address is not available then this object contains an empty string. The type of address for this is identified by ccmH323DevInetAddressType.
  ::= { ccmH323DeviceEntry 6 }

ccmH323DevCnfgGKInetAddressType OBJECT-TYPE
  SYNTAX  InetAddressType
  MAX-ACCESS  read-only
  STATUS  current
  DESCRIPTION
  This object identifies the IP address type of the gatekeeper device. The value of this object is 'unknown(0)' if the IP address of a H323 gatekeeper is not available.
  ::= { ccmH323DeviceEntry 7 }

ccmH323DevCnfgGKInetAddress OBJECT-TYPE
  SYNTAX  InetAddress
  MAX-ACCESS  read-only
  STATUS  current
  DESCRIPTION
  This object represents configured gatekeeper DNS name or IP address for this H323 device. This is applicable only for H323 devices with gatekeepers configured. When there is no H323 gatekeeper configured, this object contains an empty string. The type of address for this is identified by ccmH323DevCnfgGKInetAddressType.
  ::= { ccmH323DeviceEntry 8 }

ccmH323DevAltGK1InetAddressType OBJECT-TYPE
  SYNTAX  InetAddressType
  MAX-ACCESS  read-only
  STATUS  current
  DESCRIPTION
This object identifies the IP address type of the first alternate gatekeeper. The value of this object is 'unknown(0)' if the IP address of a H323 gatekeeper is not available.

::= \{ ccmH323DeviceEntry 9 \}

ccmH323DevAltGK1InetAddress OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the first alternate gatekeeper DNS name or IP address for this H323 device. This is applicable only for H323 devices with gatekeepers configured. When there is no first alternate H323 gatekeeper, this object contains an empty string. The type of address for this is identified by ccmH323DevAltGK1InetAddressType.

::= \{ ccmH323DeviceEntry 10 \}

ccmH323DevAltGK2InetAddressType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the IP address type of the second alternate gatekeeper. The value of this object is 'unknown(0)' if the IP address of a H323 gatekeeper is not available.

::= \{ ccmH323DeviceEntry 11 \}

ccmH323DevAltGK2InetAddress OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the second alternate gatekeeper DNS name or IP address for this H323 device. This is applicable only for H323 devices with gatekeepers configured. When there is no second alternate H323 gatekeeper, this object contains an empty string. The type of address for this is identified by ccmH323DevAltGK2InetAddressType.

::= \{ ccmH323DeviceEntry 12 \}

ccmH323DevAltGK3InetAddressType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the third alternate gatekeeper DNS name or IP address for this H323 device. This is applicable only for H323 devices with gatekeepers configured. When there is no third alternate H323 gatekeeper, this object contains an empty string. The type of address for this is identified by ccmH323DevAltGK3InetAddressType.

::= \{ ccmH323DeviceEntry 13 \}

ccmH323DevAltGK3InetAddress OBJECT-TYPE
SYNTAX  InetAddress
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object identifies the third alternate gatekeeper DNS name or IP address for this H323 device. This is applicable only for H323 devices with gatekeepers configured. When there is no third alternate H323 gatekeeper, this object contains an empty string. The type of address for this is identified by ccmH323DeviceEntry 14.

::= { ccmH323DeviceEntry 14 }

ccmH323DevAltGK3InetAddressType OBJECT-TYPE
SYNTAX  InetAddressType
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object identifies the IP address type of the fourth alternate gatekeeper. The value of this object is 'unknown(0)' if the IP address of a H323 gatekeeper is not available.

::= { ccmH323DeviceEntry 15 }

ccmH323DevAltGK4InetAddress OBJECT-TYPE
SYNTAX  InetAddress
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object identifies the fourth alternate gatekeeper DNS name or IP address for this H323 device. This is applicable only for H323 devices with gatekeepers configured. When there is no fourth H323 alternate gatekeeper, this object contains an empty string. The type of address for this is identified by ccmH323DevAltGK4InetAddressType.

::= { ccmH323DeviceEntry 16 }

ccmH323DevAltGK5InetAddressType OBJECT-TYPE
SYNTAX  InetAddressType
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object identifies the IP address type of the fifth alternate gatekeeper. The value of this object is 'unknown(0)' if the IP address of a H323 gatekeeper is not available.

::= { ccmH323DeviceEntry 17 }

ccmH323DevAltGK5InetAddress OBJECT-TYPE
SYNTAX  InetAddress
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object identifies the fifth alternate gatekeeper DNS name or IP address for this H323 device. This is applicable only for H323 devices with gatekeepers configured. When there is no fifth H323 alternate gatekeeper, this object contains an empty string. The type of address for this is identified by `ccmH323DevAltGK5InetAddressType`.

```
::= { ccmH323DeviceEntry 18 }
```

### `ccmH323DevActGKInetAddressType` OBJECT-TYPE

- **SYNTAX** `InetAddressType`
- **MAX-ACCESS** read-only
- **STATUS** current
- **DESCRIPTION**
  
  This object identifies the IP address type of the active gatekeeper. The value of this object is 'unknown(0)' if the IP address of a gatekeeper is not available.

```
::= { ccmH323DeviceEntry 19 }
```

### `ccmH323DevActGKInetAddress` OBJECT-TYPE

- **SYNTAX** `InetAddress`
- **MAX-ACCESS** read-only
- **STATUS** current
- **DESCRIPTION**
  
  This object identifies the active alternate gatekeeper DNS name or IP address for this H323 device. This is applicable only for H323 devices with gatekeepers configured. When there is no active alternate H323 gatekeeper, this object contains an empty string. The type of address for this is identified by `ccmH323DevActGKInetAddressType`.

```
::= { ccmH323DeviceEntry 20 }
```

### `ccmH323DevStatus` OBJECT-TYPE

- **SYNTAX** `INTEGER` {
  - `notApplicable(0)`,
  - `unknown(1)`,
  - `registered(2)`,
  - `unregistered(3)`,
  - `rejected(4)`
}
- **MAX-ACCESS** read-only
- **STATUS** current
- **DESCRIPTION**
  
  The H323 device registration status with the gatekeeper. The status changes from unknown to registered when the H323 device successfully registers itself with the gatekeeper.

  - notApplicable: The registration status is not applicable for this H323 device
  - unknown: The registration status of the H323 device with the gatekeeper is unknown
  - registered: The H323 device has registered with the gatekeeper successfully
  - unregistered: The H323 device is no longer registered with the gatekeeper
rejected: Registration request from the H323 device was rejected by the gatekeeper.

::= { ccmH323DeviceEntry 21 }

ccmH323DevTimeLastStatusUpdt OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The time the registration status with the gatekeeper changed. This is applicable only for H323 devices with gatekeepers configured.

::= { ccmH323DeviceEntry 23 }

ccmH323DevTimeLastRegistered OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The time when the H323 device last registered with the gatekeeper. This is applicable only for H323 devices with gatekeepers configured.

::= { ccmH323DeviceEntry 24 }

ccmH323DevRmtCM1InetAddressType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the IP address type of the first remote call manager. The value of this object is 'unknown(0)' if the first remote call manager is not configured.

::= { ccmH323DeviceEntry 25 }

ccmH323DevRmtCM1InetAddress OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the first remote call manager DNS name or IP address configured for this H323 device. When there is no first remote call manager configured, this object contains an empty string. The type of address for this is identified by ccmH323DevRmtCM1InetAddressType.

::= { ccmH323DeviceEntry 26 }

ccmH323DevRmtCM2InetAddressType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS read-only
STATUS current

ccmH323DevRmtCM2InetAddress OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS read-only
STATUS current
This object identifies the IP address type of the second remote call manager. The value of this object is 'unknown(0)' if the second remote call manager is not configured.

::= { ccmH323DeviceEntry 27 }

ccmH323DevRmtCM2InetAddress OBJECT-TYPE
SYNTAX  InetAddress
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object identifies the second remote call manager DNS name or IP address configured for this H323 device. When there is no second remote call manager configured, this object contains an empty string. The type of address for this is identified by ccmH323DevRmtCM2InetAddressType.

::= { ccmH323DeviceEntry 28 }

ccmH323DevRmtCM2InetAddressType OBJECT-TYPE
SYNTAX  InetAddressType
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object identifies the IP address type of the second remote call manager. The value of this object is 'unknown(0)' if the second remote call manager is not configured.

::= { ccmH323DeviceEntry 29 }

ccmH323DevRmtCM3InetAddress OBJECT-TYPE
SYNTAX  InetAddress
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object identifies the third remote call manager DNS name or IP address configured for this H323 device. When there is no third remote call manager configured, this object contains an empty string. The type of address for this is identified by ccmH323DevRmtCM3InetAddressType.

::= { ccmH323DeviceEntry 30 }

ccmH323DevProductTypeIndex OBJECT-TYPE
SYNTAX  CcmIndexOrZero
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
A positive value of this index is used to identify the related product type entry in the ccmProductTypeTable. A value of zero indicates that the index to the ccmProductTypeTable is Unknown.

::= { ccmH323DeviceEntry 31 }

ccmH323DevUnregReason OBJECT-TYPE
SYNTAX CcmDevUnregCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with unregistered H323 Device. This is applicable only for H323 devices with gatekeepers configured.
::= { ccmH323DeviceEntry 32 }

ccmH323DevRegFailReason OBJECT-TYPE
SYNTAX CcmDevRegFailCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with registration failed H323 Device. This is applicable only for H323 devices with gatekeepers configured.
::= { ccmH323DeviceEntry 33 }

Voice Mail Device Table

ccmVoiceMailDeviceTable OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmVoiceMailDeviceEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
The table containing a list of all voice messaging devices that have tried to register with the local Cisco Unified CM at least once. When the local Cisco Unified CM is restarted, this table will be refreshed.
::= { ccmVoiceMailDeviceInfo 1 }

ccmVoiceMailDeviceEntry OBJECT-TYPE
SYNTAX  CcmVoiceMailDeviceEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
An entry (conceptual row) in the VoiceMailDevice Table, containing the information about a particular Voice Messaging Device.
INDEX  { ccmVMailDevIndex }
::= { ccmVoicMailDeviceTable 1 }
CcmVoiceMailDeviceEntry ::= SEQUENCE {
ccmVMailDevIndex CcmIndex,
ccmVMailDevName  SnmpAdminString,
ccmVMailDevProductId CcmDeviceProductId,
ccmVMailDevDescription, SnmpAdminString,
ccmVMailDevStatus CcmDeviceStatus,
ccmVMailDevInetAddressType InetAddressType,
ccmVMailDevInetAddress InetAddress,
ccmVMailDevStatusReason CcmDevFailCauseCode,
ccmVMailDevTimeLastStatusUpdt DateAndTime,
ccmVMailDevTimeLastRegistered DateAndTime,
ccmVMailDevProductTypeIndex CcmIndexOrZero
ccmVMailDevUnregReason CcmDevUnregCauseCode,
ccmVMailDevRegFailReason CcmDevRegFailCauseCode

}

ccmVMailDevIndex OBJECT-TYPE
SYNTAX  CcmIndex
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that identifies a voice messaging
device entry in the table.
::= { ccmVoiceMailDeviceEntry 1 }

ccmVMailDevName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The name of the Voice Messaging Device. This name is assigned to the Voice Messaging
Device when it is added to the Cisco Unified CM.
::= { ccmVoiceMailDeviceEntry 2 }

ccmVMailDevDESCRIPTION OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..255))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The description of the Voice Messaging Device. This description is given when the Voice Messaging
Device is configured in the Cisco Unified CM.
::= { ccmVoiceMailDeviceEntry 4 }

ccmVMailDevStatus OBJECT-TYPE
SYNTAX  CcmDeviceStatus
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The status of the Voice Messaging Device. The Voice Messaging Device status changes from
unknown to registered when it registers itself with the local Cisco Unified CM.
::= { ccmVoiceMailDeviceEntry 5 }

ccmVMailDevInetAddressType OBJECT-TYPE
SYNTAX  InetAddressType
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object identifies the IP address type of the Voice Messaging device. The value of this object is
'unknown(0)' if the IP address of the Voice Messaging device is not available.
::= { ccmVoiceMailDeviceEntry 6 }

ccmVMailDevInetAddress OBJECT-TYPE
SYNTAX  InetAddress
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object identifies the IP Address of the Voice Messaging Device. If the IP Address is not
available then this object contains an empty string. The type of address for this is identified by
ccmVMailDevInetAddressType.
::= { ccmVoiceMailDeviceEntry 7 }

ccmVMailDevTimeLastStatusUpdt OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The time the status of the voice messaging device changed.
::= { ccmVoiceMailDeviceEntry 9 }

ccmVMailDevTimeLastRegistered OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The time the Voice Messaging Device has last registered with the call manager.
::= { ccmVoiceMailDeviceEntry 10 }

ccmVMailDevProductTypeIndex OBJECT-TYPE
SYNTAX  CcmIndexOrZero
MAX-ACCESS  read-only
STATUS current

DESCRIPTION
A positive value of this index is used to identify the related product type entry in the ccmProductTypeTable. A value of 0 indicates that the index to the ccmProductTypeTable is Unknown.

 ::= { ccmVoiceMailDeviceEntry 11 }

ccmVMailDevUnregReason OBJECT-TYPE
SYNTAX CcmDevUnregCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with unregistered Voice Messaging Device.

 ::= { ccmVoiceMailDeviceEntry 12 }

ccmVMailDevRegFailReason OBJECT-TYPE
SYNTAX CcmDevRegFailCauseCode
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The reason code associated with registration failed Voice Messaging Device.

 ::= { ccmVoiceMailDeviceEntry 13 }

Voice Mail Directory Number Table

ccmVoiceMailDeviceDirNumTable OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmVoiceMailDeviceDirNumEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing a list of directory numbers that are assigned to all of the registered and unregistered Voice Messaging Devices in the ccmVoiceMailDeviceTable.

 ::= { ccmVoiceMailDeviceInfo 2 }

ccmVoiceMailDeviceDirNumEntry OBJECT-TYPE
SYNTAX CcmVoiceMailDeviceDirNumEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the VoiceMailDirNum Table, has the associated directory number for a Voice Messaging Device.
INDEX  { ccmVMailDevIndex, ccmVMailDevDirNumIndex }
::= { ccmVoiceMailDeviceDirNumTable 1 }

CcmVoiceMailDeviceDirNumEntry ::= SEQUENCE {
    ccmVMailDevDirNumIndex CcmIndex,
    ccmVMailDevDirNum SnmpAdminString
}

ccmVMailDevDirNumIndex OBJECT-TYPE
SYNTAX  CcmIndex
MAX-ACCESS not-accessible
STATUS  current
DESCRIPTION
An arbitrary integer, selected by the local system, that identifies a Directory Number of a Voice Messaging Device.

::= { ccmVoiceMailDeviceDirNumEntry 1 }

ccmVMailDevDirNum OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..24))
MAX-ACCESS read-only
STATUS  current
DESCRIPTION
The Directory Number of the Voice Messaging Device.

::= { ccmVoiceMailDeviceDirNumEntry 2 }

Quality Report Alarm Configuration Information

ccmQualityReportAlarmEnable OBJECT-TYPE
SYNTAX  TruthValue
MAX-ACCESS read-write
STATUS  current
DESCRIPTION
Allows the generation of the quality report alarm.

true(1): Enabling this object will allow the Cisco Unified CM agent to generate the ccmQualityReport alarm. This is the default value.

false(2): Disabling this object will stop the generation of the ccmQualityReport alarm by the Cisco Unified CM agent.

DEFVAL { true }

::= { ccmQualityReportAlarmConfigInfo 1 }

Sip Device Table

ccmSIPDeviceTable OBJECT-TYPE
SYNTAX  SEQUENCE OF CcmSIPDeviceEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table containing a list of all SIP trunks in the Cisco Unified CM cluster that the local Cisco
Unified CM is aware of. When the local Cisco Unified CM is restarted, this table will be refreshed.
If the local Cisco Unified CM is down, then this table will be empty.
::= { ccmSIPDeviceInfo 1 }

ccmSIPDeviceEntry OBJECT-TYPE
SYNTAX  CcmSIPDeviceEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the SIP Device Table, containing the information about a particular
SIP Trunk Device.
INDEX { ccmSIPDevIndex }
::= { ccmSIPDeviceTable 1 }
CcmSIPDeviceEntry ::= SEQUENCE {
  ccmSIPDevIndex  CcmIndex,
  ccmSIPDevName  SnmpAdminString,
  ccmSIPDevProductTypeIndex  CcmIndexOrZero,
  ccmSIPDevDescription SnmpAdminString,
  ccmSIPDevInetAddressType   InetAddressType,
  ccmSIPDevInetAddress   InetAddress,
  ccmSIPInTransportProtocolType  CcmSIPTransportProtocolType,
  ccmSIPInPortNumber  InetPortNumber,
  ccmSIPOutTransportProtocolType CcmSIPTransportProtocolType,
  ccmSIPOutPortNumber  InetPortNumber
  ccmSIPDevInetAddressIPv4 InetAddressIPv4,
  ccmSIPDevInetAddressIPv6 InetAddressIPv6
}

ccmSIPDevIndex OBJECT-TYPE
SYNTAX  CcmIndex
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An arbitrary integer, selected by the local Cisco Unified CM, that identifies a SIP Trunk Device
entry in the table.
::= { ccmSIPDeviceEntry 1 }

ccmSIPDevName OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..128))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The device name assigned to the SIP Trunk Device. This name is assigned when a new SIP Trunk device is added to the Cisco Unified CM.
::= { ccmSIPDeviceEntry 2 }

ccmSIPDevProductTypeIndex OBJECT-TYPE
SYNTAX  CcmIndexOrZero
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
A positive value of this index is used to identify the related product type entry in the ccmProductTypeTable. A value of zero indicates that the index to the ccmProductTypeTable is Unknown.
::= { ccmSIPDeviceEntry 3 }

ccmSIPDevDescription
OBJECT-TYPE
SYNTAX  SnmpAdminString (SIZE(0..255))
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
A description of the SIP Trunk device. This Description is given when the SIP Trunk device is configured in the Cisco Unified CM.
::= { ccmSIPDeviceEntry 4 }

ccmSIPInTransportProtocolType OBJECT-TYPE
SYNTAX  CcmSIPTransportProtocolType
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
Specifies the transport protocol type used by Cisco Unified CM for setting up incoming SIP call.
::= { ccmSIPDeviceEntry 7 }

ccmSIPInPortNumber OBJECT-TYPE
SYNTAX  InetPortNumber
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
Specifies the port number used by Cisco Unified CM for setting up incoming SIP call.
::= { ccmSIPDeviceEntry 8 }
ccmSIPOutTransportProtocolType OBJECT-TYPE
SYNTAX  CcmSIPTransportProtocolType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Specifies the transport protocol type used by Cisco Unified CM for setting up outgoing SIP call.
::= { ccmSIPDeviceEntry 9 }

ccmSIPOutPortNumber OBJECT-TYPE
SYNTAX InetPortNumber
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Specifies the port number used by Cisco Unified CM for setting up outgoing SIP call.
::= { ccmSIPDeviceEntry 10 }

ccmSIPDevInetAddressIPv4 OBJECT-TYPE
SYNTAX InetAddressIPv4
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the last known primary IPv4 address of the SIP Trunk Device. This object contains value zero if IPV4 address is not available.
::= { ccmSIPDeviceEntry 11 }

ccmSIPDevInetAddressIPv6 OBJECT-TYPE
SYNTAX InetAddressIPv6
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object identifies the last known primary IPv6 address of the SIP Trunk Device. This object contains value zero if IPV6 address is not available.
::= { ccmSIPDeviceEntry 12 }

Notifications Types

ccmMIBNotificationPrefix OBJECT IDENTIFIER
::= { ciscoCcmMIB 2 }

ccmMIBNotifications OBJECT IDENTIFIER
::= { ccmMIBNotificationPrefix 0 }

ccmCallManagerFailed NOTIFICATION-TYPE
OBJECTS {
ccmAlarmSeverity,
ccmFailCauseCode
}
STATUS current
DESCRIPTION
This Notification signifies that the Cisco Unified CM process detects a failure in one of its critical subsystems. It can also be detected from a heartbeat/event monitoring process.
::= { ccmMIBNotifications 1 }

ccmPhoneFailed NOTIFICATION-TYPE
OBJECTS {
ccmAlarmSeverity,
ccmPhoneFailures
}
STATUS current
DESCRIPTION
This Notification will be generated in the intervals specified in ccmPhoneFailedAlarmInterval if there is at least one entry in the ccmPhoneFailedTable.
::= { ccmMIBNotifications 2 }

ccmPhoneStatusUpdate NOTIFICATION-TYPE
OBJECTS {
ccmAlarmSeverity,
ccmPhoneUpdates
}
STATUS current
DESCRIPTION
This Notification will be generated in the intervals specified in ccmPhoneStatusUpdateInterval if there is at least one entry in the ccmPhoneStatusUpdateTable.
::= { ccmMIBNotifications 3 }

ccmMediaResourceListExhausted NOTIFICATION-TYPE
OBJECTS {
ccmAlarmSeverity,
ccmMediaResourceType,
ccmMediaResourceListName
}
STATUS current
DESCRIPTION
This Notification indicates that the Cisco Unified CM has run out a certain specified type of resource.
::= { ccmMIBNotifications 5 }
ccmRouteListExhausted NOTIFICATION-TYPE
  OBJECTS {
    ccmAlarmSeverity,
    ccmRouteListName
  }
  STATUS  current
  DESCRIPTION
  This Notification indicates that the Cisco Unified CM could not find an available route in the
  indicated route list.
  ::= { ccmMIBNotifications 6 }

ccmGatewayLayer2Change NOTIFICATION-TYPE
  OBJECTS {
    ccmAlarmSeverity,
    ccmGatewayName,
    ccmGatewayInetAddressType,
    ccmGatewayInetAddress,
    ccmGatewayPhysIfIndex,
    ccmGatewayPhysIfL2Status
  }
  STATUS  current
  DESCRIPTION
  This Notification is sent when the D-Channel/Layer 2 of an interface in a skinny gateway that has
  registered with the Cisco Unified CM changes state.
  ::= { ccmMIBNotifications 7 }

ccmMaliciousCall NOTIFICATION-TYPE
  OBJECTS {
    ccmAlarmSeverity,
    ccmMaliCallCalledPartyName,
    ccmMaliCallCalledPartyNumber,
    ccmMaliCallCalledDeviceName,
    ccmMaliCallCallingPartyName,
    ccmMaliCallCallingPartyNumber,
    ccmMaliCallCallingDeviceName,
    ccmMaliCallTime
  }
  STATUS  current
  DESCRIPTION
  This Notification is sent when a user registers a call as malicious with the local call manager.
:= \{ ccmMIBNotifications 8 \}

ccmQualityReport NOTIFICATION-TYPE
OBJECTS {
  ccmAlarmSeverity,
  ccmQualityRprtSourceDevName,
  ccmQualityRprtClusterId,
  ccmQualityRprtCategory,
  ccmQualityRprtReasonCode,
  ccmQualityRprtTime
}
STATUS current
DESCRIPTION
This Notification is sent when a user reports a quality problem using the Quality Report Tool.

:= \{ ccmMIBNotifications 9 \}

ccmTLSConnectionFailure NOTIFICATION-TYPE
OBJECTS {
  ccmAlarmSeverity,
  ccmTLSDevName,
  ccmTLSDevInetAddressType,
  ccmTLSDevInetAddress,
  ccmTLSConnectionFailReasonCode,
  ccmTLSConnFailTime
}
STATUS current
DESCRIPTION
This Notification is sent when Cisco Unified CM fails to open TLS connection for the indicated device.

:= \{ ccmMIBNotifications 10 \}

ccmGatewayFailedReason NOTIFICATION-TYPE
OBJECTS {
  ccmAlarmSeverity,
  ccmGatewayName,
  ccmGatewayInetAddressType,
  ccmGatewayInetAddress,
  ccmGatewayRegFailCauseCode
}
STATUS current
DESCRIPTION
This Notification indicates that at least one gateway has attempted to register or communicate with the Cisco Unified CM and failed.

::= { ccmMIBNotifications 11 }

MIB Conformance Statements

ciscoCcmMIBConformance OBJECT IDENTIFIER
::= { ciscoCcmMIB 3 }

ciscoCcmMIBCompliances OBJECT IDENTIFIER
::= { ciscoCcmMIBConformance 1 }

ciscoCcmMIBGroups OBJECT IDENTIFIER
::= { ciscoCcmMIBConformance 2 }

Compliance Statements

ciscoCcmMIBComplianceRev7 MODULE-COMPLIANCE
STATUS current
DESCRIPTION
The compliance statement for entities that implement the Cisco Unified CM MIB.
MANDATORY-GROUPS {
  ccmInfoGroupRev4,
  ccmPhoneInfoGroupRev6,
  ccmGatewayInfoGroupRev4,
  ccmMediaDeviceInfoGroupRev4,
  ccmCTIDeviceInfoGroupRev4,
  ccmNotificationsInfoGroupRev5,
  ccmNotificationsGroupRev3,
  ccmH323DeviceInfoGroupRev3,
  ccmVoiceMailDeviceInfoGroupRev2,
  ccmSIPDeviceInfoGroupRev2
}
::= { ciscoCcmMIBCompliances 8 }

Units of Conformance

ccmMediaDeviceInfoGroupRev2 OBJECT-GROUP
OBJECTS {
  ccmMediaDeviceName,
  ccmMediaDeviceDescription,
  ccmMediaDeviceStatus,
  ccmMediaDeviceDevicePoolIndex,
ccmMediaDeviceInetAddressType,
ccmMediaDeviceInetAddress,
ccmMediaDeviceStatusReason,
ccmMediaDeviceTimeLastStatusUpdt,
ccmMediaDeviceTimeLastRegistered,
ccmMediaDeviceProductTypeIndex,
ccmRegisteredMediaDevices,
ccmUnregisteredMediaDevices,
ccmRejectedMediaDevices
}

STATUS  current
DESCRIPTION
A collection of objects that provide info about all Media Devices within the scope of the local Cisco Unified CM. It comprises of the MediaDevice table.
::= { ciscoCcmMIBGroups 26 }

ccmCTIDeviceInfoGroup Rev2 OBJECT-GROUP
OBJECTS {
ccmCTIDeviceName,
ccmCTIDeviceDescription,
ccmCTIDeviceStatus,
ccmCTIDevicePoolIndex,
ccmCTIDeviceInetAddressType,
ccmCTIDeviceInetAddress,
ccmCTIDeviceStatusReason,
ccmCTIDeviceTimeLastStatusUpdt,
ccmCTIDeviceTimeLastRegistered,
ccmCTIDeviceProductTypeIndex,
ccmCTIDeviceDirNum,
ccmRegisteredCTIDevices,
ccmUnregisteredCTIDevices,
ccmRejectedCTIDevices,
ccmCTIDeviceTableStateId,
ccmCTIDeviceDirNumTableStateId
}

STATUS  current
DESCRIPTION
A collection of objects that provide info about all CTI Devices within the scope of the local Cisco Unified CM. It comprises of the ccmCTIDevice and ccmCTIDeviceDirNum tables.
::= { ciscoCcmMIBGroups 27 }

ccmInfoGroupRev4 OBJECT-GROUP

OBJECTS {
  ccmGroupName,
  ccmGroupTftpDefault,
  ccmName,
  ccmDescription,
  ccmVersion,
  ccmStatus,
  ccmInetAddressType,
  ccmInetAddress,
  ccmClusterId,
  ccmCMGroupMappingCMPriority,
  ccmRegionName,
  ccmRegionAvailableBandWidth,
  ccmTimeZoneName,
  ccmTimeZoneOffsetHours,
  ccmTimeZoneOffsetMinutes,
  ccmDevicePoolName,
  ccmDevicePoolRegionIndex,
  ccmDevicePoolTimeZoneIndex,
  ccmDevicePoolGroupId,
  ccmProductType,
  ccmProductName,
  ccmProductCategory,
  ccmCallManagerStartTime,
  ccmSystemVersion,
  ccmInstallationId,
  ccmInetAddress2Type,
  ccmInetAddress2
}

STATUS current

DESCRIPTION

A collection of objects that provide information about all Cisco Unified Communications Managers and its related information within a Cisco Unified CM cluster. It comprises of GroupTable, ccmTable, GroupMappingTable, Region, TimeZone, Device Pool and ProductType tables.

::= { ciscoCcmMIBGroups 34 }

ccmSIPDeviceInfoGroupRev2 OBJECT-GROUP
OBJECTS {
  ccmSIPDevName,
  ccmSIPDevProductTypeIndex,
  ccmSIPDevDescription,
  ccmSIPInTransportProtocolType,
  ccmSIPInPortNumber,
  ccmSIPOutTransportProtocolType,
  ccmSIPOutPortNumber,
  ccmSIPDevInetAddressIPv4,
  ccmSIPDevInetAddressIPv6,
  ccmSIPTableEntries
}
STATUS current
DESCRIPTION
A collection of objects that provide information about all SIP Trunk devices within the scope of the local Cisco Unified Communications Manager. It comprises of the SIP Device table.
::= { ciscoCcmMIBGroups 37 }

ccmPhoneInfoGroupRev6 OBJECT-GROUP

OBJECTS {
  ccmPhonePhysicalAddress,
  ccmPhoneDescription,
  ccmPhoneUserName,
  ccmPhoneStatus,
  ccmPhoneTimeLastRegistered,
  ccmPhoneE911Location,
  ccmPhoneLoadID,
  ccmPhoneDevicePoolIndex,
  ccmPhoneTimeLastStatusUpdt,
  ccmPhoneProductTypeIndex,
  ccmPhoneProtocol,
  ccmPhoneName,
  ccmPhoneExtn,
  ccmPhoneExtnMultiLines,
  ccmPhoneExtnInetAddressType,
  ccmPhoneExtnInetAddress,
  ccmPhoneExtnStatus,
  ccmRegisteredPhones,
  ccmUnregisteredPhones,
}
ccmRejectedPhones,
ccmPartiallyRegisteredPhones,
ccmPhoneTableStateId,
ccmPhoneExtensionTableStateId,
ccmPhoneInetAddressIPv4,
ccmPhoneInetAddressIPv6,
ccmPhoneIPv4Attribute,
ccmPhoneIPv6Attribute,
ccmPhoneActiveLoadID,
ccmPhoneUnregReason,
ccmPhoneRegFailReason
}
STATUS current
DESCRIPTION
A collection of objects that provide information about all phones within the scope of the local Cisco Unified Communications Manager. It comprises of the Phone and Phone Extension tables.
 ::= { ciscoCcmMIBGroups 41 }

cmNotificationsInfoGroupRev5 OBJECT-GROUP
 OBJECTS {
cmAlarmSeverity,
cmCallManagerAlarmEnable,
cmFailCauseCode,
cmPhoneFailures,
cmPhoneFailedTime,
cmPhoneFailedMacAddress,
cmPhoneFailedAlarmInterval,
cmPhoneFailedStorePeriod,
cmPhFailedTbltLastAddedIndex,
cmPhoneUpdates,
cmPhoneStatusPhoneIndex,
cmPhoneStatusUpdateTime,
cmPhoneStatusUpdateType,
cmPhoneStatusUpdateAlarmInterv,
cmPhoneStatusUpdateStorePeriod,
cmPhoneStatusUpdateTableStateId,
cmStatUpdtTbltLastAddedIndex,
cmGatewayAlarmEnable,
cmMediaResourceType,
ccmMediaResourceListName,
ccmRouteListName,
ccmGatewayPhysIfIndex,
ccmGatewayPhysIfL2Status,
ccmMaliciousCallAlarmEnable,
ccmMaliCallCalledPartyName,
ccmMaliCallCalledPartyNumber,
ccmMaliCallCalledDeviceName,
ccmMaliCallCallingPartyName,
ccmMaliCallCallingPartyNumber,
ccmMaliCallCallingDeviceName,
ccmMaliCallTime,
ccmQualityReportAlarmEnable,
ccmQualityRprtSourceDevName,
ccmQualityRprtClusterId,
ccmQualityRprtCategory,
ccmQualityRprtReasonCode,
ccmQualityRprtTime,
ccmTLSDevName,
ccmTLSDevInetAddressType,
ccmTLSDevInetAddress,
ccmTLSConnFailTime,
ccmTLSConnectionFailReasonCode,
ccmPhoneFailedInetAddressIPv4,
ccmPhoneFailedInetAddressIPv6,
ccmPhoneFailedIPv4Attribute,
ccmPhoneFailedIPv6Attribute,
ccmPhoneFailedRegFailReason,
ccmPhoneStatusUnregReason,
ccmPhoneStatusRegFailReason,
ccmGatewayRegFailCauseCode
}

STATUS current
DESCRIPTION
A collection of objects that provide information about all the Notifications generated by the Cisco Unified CM Agent.

::= { ciscoCcmMIBGroups 42 }

ccmGatewayInfoGroupRev4 OBJECT-GROUP
CISCO-CCM-MIB

OBJECTS {
  ccmGatewayName,
  ccmGatewayDescription,
  ccmGatewayStatus,
  ccmGatewayDevicePoolIndex,
  ccmGatewayInetAddressType,
  ccmGatewayInetAddress,
  ccmGatewayTimeLastStatusUpdt,
  ccmGatewayTimeLastRegistered,
  ccmGatewayDChannelStatus,
  ccmGatewayDChannelNumber,
  ccmGatewayProductTypeIndex,
  ccmRegisteredGateways,
  ccmUnregisteredGateways,
  ccmRejectedGateways,
  ccmGatewayTableStateId,
  ccmGatewayUnregReason,
  ccmGatewayRegFailReason
}

STATUS current

DESCRIPTION
A collection of objects that provide information about all Gateways within the scope of the local Cisco Unified CM. It comprises of the Gateway table.

::= { ciscoCcmMIBGroups 43 }

ccmMediaDeviceInfoGroupRev4 OBJECT-GROUP

OBJECTS {
  ccmMediaDeviceName,
  ccmMediaDeviceDescription,
  ccmMediaDeviceStatus,
  ccmMediaDeviceDevicePoolIndex,
  ccmMediaDeviceInetAddressIPv4,
  ccmMediaDeviceInetAddressIPv6,
ccmMediaDeviceUnregReason,
ccmMediaDeviceRegFailReason
}
STATUS current
DESCRIPTION
A collection of objects that provide information about all Media Devices within the scope of the
local Cisco Unified Communications Manager. It comprises of the MediaDevice table.
::= { ciscoCcmMIBGroups 44 }

ccmCTIDeviceInfoGroupRev4 OBJECT-GROUP
OBJECTS {
ccmCTIDeviceName,
ccmCTIDeviceDescription,
ccmCTIDeviceStatus,
ccmCTIDevicePoolIndex,
ccmCTIDeviceTimeLastStatusUpdt,
ccmCTIDeviceTimeLastRegistered,
ccmCTIDeviceProductTypeIndex,
ccmCTIDeviceDirNum,
ccmRegisteredCTIDevices,
ccmUnregisteredCTIDevices,
ccmRejectedCTIDevices,
ccmCTIDeviceTableStateId,
ccmCTIDeviceDirNumTableStateId,
ccmCTIDeviceInetAddressIPv4,
ccmCTIDeviceInetAddressIPv6,
ccmCTIDeviceUnregReason,
ccmCTIDeviceRegFailReason
}
STATUS current
DESCRIPTION
A collection of objects that provide information about all CTI Devices within the scope of the local
Cisco Unified CM. It comprises of the ccmCTIDevice and ccmCTIDeviceDirNum tables.
::= { ciscoCcmMIBGroups 45 }

ccmH323DeviceInfoGroupRev3 OBJECT-GROUP
OBJECTS {
ccmH323DevName,
ccmH323DevDescription,
ccmH323DevInetAddressType,
```
ccmH323DevInetAddress,
ccmH323DevCnfGK1netAddressType,
ccmH323DevCnfGK1netAddress,
ccmH323DevAltGK1netAddressType,
ccmH323DevAltGK1netAddress,
ccmH323DevAltGK2netAddressType,
ccmH323DevAltGK2netAddress,
ccmH323DevAltGK3netAddressType,
ccmH323DevAltGK3netAddress,
ccmH323DevAltGK4netAddressType,
ccmH323DevAltGK4netAddress,
ccmH323DevAltGK5netAddressType,
ccmH323DevAltGK5netAddress,
ccmH323DevActGK1netAddressType,
ccmH323DevActGK1netAddress,
ccmH323DevStatus,
ccmH323DevTimeLastStatusUpdt,
ccmH323DevTimeLastRegistered,
ccmH323DevRmtCM1netAddressType,
ccmH323DevRmtCM1netAddress,
ccmH323DevRmtCM2netAddressType,
ccmH323DevRmtCM2netAddress,
ccmH323DevRmtCM3netAddressType,
ccmH323DevRmtCM3netAddress,
ccmH323DevProductTypeIndex,
ccmH323TableEntries,
ccmH323DevUnregReason,
ccmH323DevRegFailReason
}
STATUS current
DESCRIPTION
A collection of objects that provide information about all H323 devices within the scope of the local
Cisco Unified Communications Manager. It comprises of the H323Device table.
 ::= { ciscoCcmMIBGroups 46 }

ccmVoiceMailDeviceInfoGroupRev2 OBJECT-GROUP
  OBJECTS {
    ccmVMailDevName,
    ccmVMailDevDescription,
```
ccmVMailDevStatus,
ccmVMailDevInetAddressType,
ccmVMailDevInetAddress,
ccmVMailDevTimeLastStatusUpdt,
ccmVMailDevTimeLastRegistered,
ccmVMailDevProductTypeIndex,
ccmVMailDevDirNum,
ccmRegisteredVoiceMailDevices,
ccmUnregisteredVoiceMailDevices,
ccmRejectedVoiceMailDevices,
ccmVMailDevUnregReason,
ccmVMailDevRegFailReason
}

STATUS current
DESCRIPTION
A collection of objects that provide information about all Voice Messaging Devices within the scope of the local Cisco Unified CM. It comprises of the ccmVoiceMailDevice and ccmVoiceMailDirNum tables.

::= { ciscoCcmMIBGroups 47 }

ccmNotificationsGroupRev3 NOTIFICATION-GROUP

NOTIFICATIONS {
ccmCallManagerFailed,
ccmPhoneFailed,
ccmPhoneStatusUpdate,
ccmGatewayFailedReason,
ccmMediaResourceListExhausted,
ccmRouteListExhausted,
ccmGatewayLayer2Change,
ccmMaliciousCall,
ccmQualityReport,
ccmTLSConnectionFailure
}

STATUS current
DESCRIPTION
A collection of notifications that are generated by the Cisco Unified CM Agent.

::= { ciscoCcmMIBGroups 48 }
Cisco Unified CM Managed Services and SNMP Traps

The services that are provided in Cisco Unified Serviceability and the SNMP trap components to which they track are described in Table 7-2.

Table 7-2 Cisco Unified CM Managed Services, Alarms/Notifications, and Trap Components

<table>
<thead>
<tr>
<th>Cisco Unified CM Managed Service in CISCO-CCM-MIB</th>
<th>Alarm/Notifications</th>
<th>Trap Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Unified CM Failure</td>
<td>ccmCallManagerFailed</td>
<td>ccmAlarmSeverity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmFailCauseCode</td>
</tr>
<tr>
<td>Gateway Failure</td>
<td>ccmGatewayFailed</td>
<td>ccmAlarmSeverity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmGatewayName</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmGatewayInetAddressType</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmGatewayInetAddress</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmGatewayFailCauseCode</td>
</tr>
<tr>
<td>Note ccmGatewayFailed is deprecated and replaced</td>
<td></td>
<td></td>
</tr>
<tr>
<td>by ccmGatewayFailedReason.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cisco Unified CM Phones</td>
<td>ccmPhoneFailed</td>
<td>ccmAlarmSeverity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmPhoneFailures</td>
</tr>
<tr>
<td>Cisco Unified CM Media Resources</td>
<td>ccmMediaResourceList</td>
<td>ccmAlarmSeverity</td>
</tr>
<tr>
<td></td>
<td>Exhausted</td>
<td>ccmMediaResourceType</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ccmMediaResourceListName</td>
</tr>
<tr>
<td>Cisco Unified CM Route List</td>
<td>ccmRouteListExhausted</td>
<td></td>
</tr>
<tr>
<td>Gateway Layer 2 Change</td>
<td>ccmGatewayLayer2Change</td>
<td></td>
</tr>
<tr>
<td>Malicious Call Status</td>
<td>ccmMaliciousCall</td>
<td></td>
</tr>
<tr>
<td>Quality Report</td>
<td>ccmQualityReport</td>
<td></td>
</tr>
<tr>
<td>TLS Connection Failure</td>
<td>ccmTLSConnectionFailure</td>
<td></td>
</tr>
</tbody>
</table>

Cisco Unified CM Alarms to Enable

Enabling the ccmCallManagerAlarmEnable object in the CISCO-CCM-MIB allows the Cisco Unified CM agent to generate traps and send the following alarms:

- ccmCallManagerFailed
- ccmGatewayFailed
- ccmPhoneFailed
- ccmMediaResourceListExhausted
- ccmRouteListExhausted
- ccmGatewayLayer2Change
- ccmMaliciousCall
- ccmQualityReport
- ccmTLSConnectionFailure
Traps to Monitor

The following are Cisco Unified CM traps to monitor:

- **ccmCallManagerFailed.** This trap means that Cisco Unified CM has detected a failure in one of its critical subsystems. It can also be detected from a heartbeat/event monitoring process. The OID is 1.3.6.1.4.1.9.9.156.2.0.1. The trap components are ccmAlarmSeverity and ccmFailCauseCode.
  - **ccmAlarmSeverity** OID is 1.3.6.1.4.1.9.9.156.1.10.1. The values are:
    1—Emergency
    2—Alert
    3—Critical
    4—Error
    5—Warning
    6—Notice
    7—Informational
  - **ccmFailCauseCode** is derived from a monitoring thread in the Cisco Unified CM or from a heartbeat monitoring process. OID is 1.3.6.1.4.1.9.9.156.1.10.2. The values are:
    1—Unknown
    2—Heart Beat Stopped
    3—Router Thread Died
    4—Timer Thread Died
    5—Critical Thread Died
    6—Device MgrInit Failed
    7—Digit Analysis Init Failed
    8—Call Control Init Failed
    9—Link Mgr Init Failed
    10—DB Mgr Init Failed
    11—Msg Translator Init Failed
    12—Supp Services Init Failed

- **Cisco Phone Failures—CISCO-CCM-MIB::ccmPhoneFailed.** This notification is generated in the intervals specified in ccmPhoneFailedAlarmInterval if there is at least one entry in the ccmPhoneFailedTable. The OID is 1.3.6.1.4.1.9.9.156.2.0.2. The trap components are:
  - **ccmAlarmSeverity** OID is 1.3.6.1.4.1.9.9.156.1.10.1. The values are:
    1—Emergency
    2—Alert
    3—Critical

- **Cisco Unified CM Gateway Failure—CISCO-CCM-MIB::ccmGatewayFailed.** This notification indicates that at least one gateway has attempted to register or communicate with the Cisco Unified CM and failed. The OID is 1.3.6.1.4.1.9.9.156.2.0.4. The trap components are:
  - **ccmAlarmSeverity** OID is 1.3.6.1.4.1.9.9.156.1.10.1. The values are:
    1—Emergency
    2—Alert
    3—Critical
4—Error
5—Warning
6—Notice
7—Informational

- ccmGatewayFailCauseCode OID is 1.3.6.1.4.1.9.9.156.1.10.5. The type is CcmDevFailCauseCode and contains the following values:
  0—No Error
  1—Unknown
  2—No Entry In Database
  3—Database Configuration Error
  4—Device Name Unresolveable
  5—Max Dev Reg Reached
  6—Connectivity Error
  7—Initialization Error
  8—Device Initiated Reset
  9—Cisco Unified CM Reset
  10—Authentication Error
  11—Invalid X509 Name In Certificate
  12—Invalid TLS Cipher
  13—Directory Number Mismatch
  14—Malformed Register Msg

Note: CcmDevFailCauseCode is deprecated and replaced by CcmDevRegFailCauseCode and CcmDevUnregCauseCode.

- Cisco Unified CM Media Resource Exhausted—CISCO-CCM-MIB::ccmMediaResourceListExhausted. This notification indicates that Cisco Unified CM has run out a certain specified type of resource. The OID is 1.3.6.1.4.1.9.9.156.2.0.5. The critical trap components are:
  - ccmAlarmSeverity OID is 1.3.6.1.4.1.9.9.156.1.10.1. The values are:
    1—Emergency
    2—Alert
    3—Critical
    4—Error
    5—Warning
    6—Notice
    7—Informational
  - ccmMediaResourceType OID is 1.3.6.1.4.1.9.9.156.1.10.6. The values are:
    1—Unknown
2—Media Termination Point  
3—Transcoder  
4—Conference Bridge  
5—Music On Hold  
- 1.3.6.1.4.1.9.9.156.2.0.6 ccmRouteListExhausted  
- 1.3.6.1.4.1.9.9.156.2.0.7 ccmGatewayLayer2Change  
- 1.3.6.1.4.1.9.9.156.2.0.8 ccmMaliciousCall  
- 1.3.6.1.4.1.9.9.156.2.0.9 ccmQualityReport  
- 1.3.6.1.4.1.9.9.156.2.0.10 ccmTLSConnectionFailure

### Dynamic Table Objects

Table 7-3 lists the objects that are populated only if the Cisco Unified Communications Manager service is up and running or the local Cisco Unified Communications Manager service in the case of a Cisco Unified Communications Manager cluster configuration.

**Table 7-3 CISCO-CCM-MIB Dynamic Tables**

<table>
<thead>
<tr>
<th>Object</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>ccmTable</td>
<td>This table stores the version and installation ID for the local Cisco Unified CM. The table also stores information about all the Cisco Unified CMs in a cluster that the local Cisco Unified CM knows about but shows “unknown” for the version detail. If the local Cisco Unified CM is down, the table remains empty, except for the version and installation ID values.</td>
</tr>
<tr>
<td>ccmPhoneFailed, ccmPhoneStatusUpdate, ccmPhoneExtn, ccmPhone, ccmPhoneExtension</td>
<td>For the Cisco Unified IP Phone, the number of registered phones in ccmPhoneTable should match Cisco Unified Communications Manager/RegisteredHardware Phones perfmon counter. The ccmPhoneTable includes one entry for each registered, unregistered, or rejected Cisco Unified IP Phone. The ccmPhoneExtnTable uses a combined index, ccmPhoneIndex and ccmPhoneExtnIndex, for relating the entries in the ccmPhoneTable and ccmPhoneExtnTable.</td>
</tr>
<tr>
<td>ccmCTIDevice, ccmCTIDeviceDirNum</td>
<td>The ccmCTIDeviceTable stores each CTI device as one device. Based on the registration status of the CTI Route Point or CTI Port, the ccmRegisteredCTIDevices, ccmUnregisteredCTIDevices, and ccmRejectedCTIDevices counters in the Cisco Unified Communications Manager MIB get updated.</td>
</tr>
<tr>
<td>ccmSIPDevice</td>
<td>The CCMSIPDeviceTable stores each SIP trunk as one device.</td>
</tr>
</tbody>
</table>
Table 7-3  CISCO-CCM-MIB Dynamic Tables (continued)

<table>
<thead>
<tr>
<th>Object</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>ccmH323Device</td>
<td>The ccmH323DeviceTable contains the list of H323 devices for which Cisco Unified Communications Manager contains information (or the local Cisco Unified Communications Manager in the case of a cluster configuration). For H.323 phones or H.323 gateways, the ccmH.323DeviceTable contains one entry for each H.323 device. (The H.323 phone and gateway do not register with Cisco Unified Communications Manager. Cisco Unified Communications Manager generates the H.323Started alarm when it is ready to handle calls for the indicated H.323 phone and gateway.) The system provides the gatekeeper information as part of the H323 trunk information.</td>
</tr>
<tr>
<td>ccmVoiceMailDevice, ccmVoiceMailDirNum</td>
<td>For Cisco uOne, ActiveVoice, the ccmVoiceMailDeviceTable includes one entry for each voice-messaging device. Based on the registration status, the ccmRegisteredVoiceMailDevices, ccmUnregisteredVoiceMailDevices, and ccmRejectedVoiceMailDevices counters in the Cisco Unified Communications Manager MIB get updated.</td>
</tr>
<tr>
<td>ccmGateway</td>
<td>The ccmRegisteredGateways, ccmUnregistered gateways, and ccmRejectedGateways keep track of the number of registered gateway devices or ports, number of unregistered gateway devices or ports, and number of rejected gateway devices or ports, respectively. Cisco Unified Communications Manager generates alarms at the device or port level. The ccmGatewayTable, based on Cisco Unified CM alarms, contains device- or port-level information. Each registered, unregistered, or rejected device or port has one entry in ccmGatewayTable. The VG200 with two FXS ports and one T1 port has three entries in ccmGatewayTable. The ccmActiveGateway and ccmInActiveGateway counters track number of active (registered) and lost contact with (unregistered or rejected) gateway devices or ports. Based on the registration status, ccmRegisteredGateways, ccmUnregisteredGateways, and ccmRejectedGateways counters get updated.</td>
</tr>
<tr>
<td>ccmMediaDeviceInfo</td>
<td>The table contains a list of all media devices that have tried to register with the local Cisco Unified CM at least once.</td>
</tr>
<tr>
<td>ccmGroup</td>
<td>This tables contains the Cisco Unified CM groups in a Cisco Unified Communications Manager cluster.</td>
</tr>
<tr>
<td>ccmGroupMapping</td>
<td>This table maps all Cisco Unified CMs in a cluster to a Cisco Unified CM group. The table remains empty when the local Cisco Unified CM node is down.</td>
</tr>
</tbody>
</table>

**Static Table Objects**

Table 7-4 lists the objects that get populated when the Cisco Unified Communications Manager SNMP Service is running.
Troubleshooting

The following areas are discussed in this section:

- **General Tips, page 7-111**
- **For Linux and Cisco Unified CM Releases 5.x, 6.x, 7.x, page 7-114**
- **Windows and Cisco Unified CM version 4.x, page 7-115**
- **Limitations, page 7-115**
- **Frequently Asked Questions, page 7-116**

General Tips

The following are general troubleshooting tips:

- Check the community string or snmp user is properly configured on the system using the SNMP configuration web pages
- Check if Cisco Unified CM SNMP Service is activated and running by checking the ccmService window and clicking **Tools > Service Activation/ ControlCenter - Feature Services**.
- Check if SNMP Master Agent is running by checking the ccmService window and clicking **Tools > Service Activation/ ControlCenter - Network Services**
- Check if Cisco Unified CM is running.
- If Cisco Unified CM is not running, only the following MIB tables respond:
  - ccmGroupTable
  - ccmRegionTable
  - ccmRegionPairTable

### Table 7-4  CISCO-CCM-MIB Static Tables

<table>
<thead>
<tr>
<th>Object</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>ccmProductType</td>
<td>The table contains the list of product types that are supported with Cisco Unified Communications Manager (or cluster, in the case of a Cisco Unified Communications Manager cluster configuration), including phone types, gateway types, media device types, H323 device types, CTI device types, voice-messaging device types, and SIP device types.</td>
</tr>
<tr>
<td>ccmRegion, ccmRegionPair</td>
<td>ccmRegionTable contains the list of all geographically separated regions in a Cisco Communications Network (CCN) system. The ccmRegionPairTable contains the list of geographical region pairs for a Cisco Unified Communications Manager cluster. Geographical region pairs are defined by Source region and Destination region.</td>
</tr>
<tr>
<td>ccmTimeZone</td>
<td>The table contains the list of all time zone groups in a Cisco Unified Communications Manager cluster.</td>
</tr>
<tr>
<td>ccmDevicePool</td>
<td>The tables contains the list of all device pools in a Cisco Unified Communications Manager cluster. Device pools are defined by Region, Date/Time Group, and Cisco Unified CM Group.</td>
</tr>
</tbody>
</table>
- ccmDevicePoolTable
- ccmProductTypeTable
- ccmQualityReportAlarmConfigInfo
- ccmGlobalInfo

- For the rest of the tables to respond Cisco Unified CM needs to be running.
- Set the debug trace level to detailed for Cisco CallManager SNMP Service. Go to the Serviceability web window and click **Trace** > **Configuration** > `<select serverCisco>` **Performance and Monitoring Services** > **CallManager SNMP Service**.
- Execute the CLI command: `utils snmp walk 2c <community> <ipaddress> 1.3.6.1.4.1.9.9.156` or execute the walk from any other management application on this OID.
- Get the Cisco Unified Communication Manager release details, Cisco SNMP CallManager Service trace, and SNMP Master agent traces after the testing above for troubleshooting reference.

Review this section for Cisco CallManager SNMP Service Troubleshooting tips:

- Be sure to set the trace setting to detailed for Cisco CallManager SNMP Service (see the “SNMP Trace Configuration” chapter of the **Cisco Unified Serviceability Administration Guide**).
- Execute the command: `snmp walk -c <community> -v2c <ipaddress> 1.3.6.1.4.1.9.9.156.1.1.2`
- Get the Cisco Unified Communications Manager version details
- Collect the following logs and information:
  - SNMP Master Agent (path: platform/snmp/snmpdm/*) and Cisco CallManager SNMP Service (path: cm/trace/ccmmib/sdi/*) by using TLC in RTMT or this CLI command: `file get activelog`
  - SNMP package version by using this CLI command: `show packages active snmp`
  - MMF Spy output for phone by using this CLI command: `show risdb query phone`
- Send the trace logs and MMF Spy data for further analysis

**Table 7-5** provides procedures for verifying that CISCO-CCM-MIB SNMP traps get sent.
### Table 7-5  How to Check CISCO-CCM-MIB SNMP Traps

<table>
<thead>
<tr>
<th>Trap</th>
<th>Verification Procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td>ccmPhoneStatusUpdate</td>
<td>1. Set MaxSeverity=Info in CiscoSyslog-&gt;dogBasic MIB table.</td>
</tr>
<tr>
<td></td>
<td>2. Set PhoneStatusUpdateAlarmInterv=30 or higher in ccmAlarmConfigInfo MIB table.</td>
</tr>
<tr>
<td></td>
<td>3. Disconnect a Cisco Unified CM server that your phones point to.</td>
</tr>
<tr>
<td></td>
<td>4. Phones will unregister.</td>
</tr>
<tr>
<td></td>
<td>5. Connect the Cisco Unified CM server again.</td>
</tr>
<tr>
<td></td>
<td>6. Phones will re-register.</td>
</tr>
<tr>
<td></td>
<td>7. Check that the ccmPhoneStatusUpdate trap is generated.</td>
</tr>
<tr>
<td>ccmPhoneFailed</td>
<td>1. Set MaxSeverity=Info in CiscoSyslog-&gt;clogBasic MIB table.</td>
</tr>
<tr>
<td></td>
<td>2. Set PhoneFailedAlarmInterv=30 or higher in ccmAlarmConfigInfo MIB table.</td>
</tr>
<tr>
<td></td>
<td>3. Make a phone fail. Delete a phone Cisco Unified Communications Manager Administration and register the phone again.</td>
</tr>
<tr>
<td></td>
<td>4. Check that the ccmPhoneFailed trap is generated.</td>
</tr>
<tr>
<td>MediaResourceListExhausted</td>
<td>1. Create a Media Resource Group (MRG) that contains one of the standard Conference Bridge resources (CFB-2).</td>
</tr>
<tr>
<td></td>
<td>2. Create a Media Resource Group List (MRGL) that contains the MRG just created.</td>
</tr>
<tr>
<td></td>
<td>3. In the Phone Configuration window (for actual phones), set MRGL as the phone Media Resource Group List.</td>
</tr>
<tr>
<td></td>
<td>4. Stop the IPVMS, which makes the Conference Bridge resource(CFB-2) stop working.</td>
</tr>
<tr>
<td></td>
<td>5. If you make conference calls with phones that use the media list, you will see &quot;No Conference Bridge available&quot; in the phone screen.</td>
</tr>
<tr>
<td></td>
<td>6. Check that a MediaListExhausted Alarm/Alert/Trap is generated.</td>
</tr>
</tbody>
</table>
Table 7-5  How to Check CISCO-CCM-MIB SNMP Traps (continued)

<table>
<thead>
<tr>
<th>Trap</th>
<th>Verification Procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td>RouteListExhausted</td>
<td>1. Create a Route Group (RG) that contains one gateway.</td>
</tr>
<tr>
<td></td>
<td>2. Create a Route Group List (RGL) that contains the RG that was just created.</td>
</tr>
<tr>
<td></td>
<td>3. Create a Route Pattern (9.XXXX) that routes a 9XXXX call through the RGL.</td>
</tr>
<tr>
<td></td>
<td>4. Unregister the gateway.</td>
</tr>
<tr>
<td></td>
<td>5. Dial 9XXXX on one of the phones.</td>
</tr>
<tr>
<td></td>
<td>6. Check that a RouteListExhausted Alarm/Alert/Trap is generated.</td>
</tr>
<tr>
<td>MaliciousCallFailed</td>
<td>1. Similar to QRT, create a softkey template. In the template, add all available</td>
</tr>
<tr>
<td></td>
<td>“MaliciousCall” softkey to the phone different status.</td>
</tr>
<tr>
<td></td>
<td>2. Assign the new softkey template to actual phones; reset the phones.</td>
</tr>
<tr>
<td></td>
<td>3. Make some calls and select the “MaliciousCall” softkey in the phone screen</td>
</tr>
<tr>
<td></td>
<td>during or after the call.</td>
</tr>
<tr>
<td></td>
<td>4. Check that a “MaliciousCallFailed” Alarm/Alert/Trap is generated.</td>
</tr>
</tbody>
</table>

For Linux and Cisco Unified CM Releases 5.x, 6.x, 7.x

Collect the following logs and information for analysis:

- SNMP Master Agent (Path: /platform/snmp/snmpdm/*)
- Cisco CallManager SNMP Service (Path: /cm/trace/ccmmib/sdi/*)
- The files can be collected using TLC (Real Time Monitoring Tool (RTMT)) or CLI by using the following command: `file get activelog <path mentioned above>`.
- All the files in /usr/local/Snmpri/conf folder. (This is possible only if ROOT/REMOTE login is available)
- The 'ls -l' listing of the above folder. (This is possible only if ROOT/REMOTE login is available)
- Collect Perfmon logs. Execute the following CLI command: `file get activelog /cm/log/ris/csv/`.
- Details of the set of actions performed that resulted in the issue.
- Ccmservice logs. Execute the following CLI command: `file get activelog /tomcat/logs/ccmservice/log4j/`.
- Collect the SNMP package version. Use the `show packages active snmp` CLI command.
- Get the MMF Spy output for Phone. Use the `show risdb query phone` CLI command.
Windows and Cisco Unified CM version 4.x

Collect the following logs for analysis:

- Set the Alarm level from the ccmservice Alarm Configuration window for Cisco Unified CM to Detailed.
- Set the RIS Trace configuration from the ccmservice window to Detailed.
- Do a snmpwalk on the ccm MIB from the network management application or execute command from any linux box by using the `snmpwalk -c <community> -v2c <ipaddress> 1.3.6.1.4.1.9.9.156`.
- Capture the output of the snmpwalk.
- Collect the logs under C:\Program Files\Cisco\Trace\RIS\CCMSNMP_*.log.
- Collect the logs under C:\Program Files\Cisco\Trace\DBL\DBL_SNMP*.txt.
- Event logs (both application and system).
- mmfSpy output for 'misc', 'CMnode' tables.
- MMFSpy tool to dump registration status (C:\Program Files\Cisco\Bin\MMFSpy.exe, gives different options). Usage: "mmfSpy -j > OutputFileName".

CISCO-CCM-MIB only supports a limited amount of configuration information about a device. For more complete configuration information, the AXL interface accessing the data in DB serves the purpose.

The list of MMFs that are created by the Cisco Unified CM Agent are as follows:

- cmnode
- cmsgroup
- cmsgroupmember
- region
- regionmatrix
- timezone
- devicepool
- phonefailed
- phonestatusupd
- cmproduct
- cmmodel

Limitations

If multiple OIDs are specified in the SNMP request and if the variables are pointing to empty tables in CISCO-CCM-MIB, then the request will take longer. In case the getbulk/getnext/getmany request has multiple OIDs in its request PDU with the subsequent tables being empty in the CISCO-CCM-MIB, the responses may be NO_SUCH_NAME for SNMP v1 version or GENERIC_ERROR for SNMP v2c or v3 version.

- Reason—This timeout occurs due to the code added to enhance the performance of the CCMAgent and throttle when it gets a large number of queries thus protecting the priority of Cisco Unified CM call processing engine.
Chapter 7  Cisco Management Information Base

- Workaround:
  - Use the available scalar variables (1.3.6.1.4.1.9.9.156.1.5) to determine the table size before accessing the table. Or do the get operation on the desired table first and then query the non-empty tables.
  - Reduce the number of variables queried in a single request. For example, for empty tables, if Management application has timeout set at 3 sec, then recommendations is to specify no more than 1 OID. For non-empty tables it takes 1 second to retrieve 1 row of data.
  - Increase the response timeout.
  - Reduce the number of retries.
  - Avoid using getbulk SNMP API. Getbulk API gets number of records specified by MaxRepetitions. This means even if the next object goes outside the table or MIB, it gets those objects. So if the CISCO-CCM-MIB has empty tables then it goes to next MIB and so will more time to respond. Use getbulk API when it is known that the table is not empty, and also know the number of records. Under this condition limit the max repetition counts to 5 to get response within 5 sec.
  - Structured SNMP queries to adapt to current limits.
  - Avoid doing a number of getbulks on the PhoneTable in case there are a number of phones registered to the Cisco Unified CM, walking it periodically may not be optimal. In such a scenario whenever there is an update, ccmPhoneStatusUpdateTable will be updated, use this information to decide whether to walk the PhoneTable.

Frequently Asked Questions

Not getting any SNMP traps from the Cisco Unified Communication Manager node for the CISCO-CCM-MIB.

For receiving SNMP traps in CISCO-CCM-MIB, you need to ensure that the value of the following MIB OIDs are set to appropriate values: `ccmPhoneFailedAlarmInterval` (1.3.6.1.4.1.9.9.156.1.9.2) and `ccmPhoneStatusUpdateAlarmInterval` (1.3.6.1.4.1.9.9.156.1.9.4) are set between 30 and 3600. The default is set to 0.

Execute the following commands from any Linux machine:

- `snmpset -c <Community String> -v 2c <transmitter ip address> 1.3.6.1.4.1.9.9.156.1.9.2.0 i <value>`
- `snmpset -c <Community String> -v 2c <transmitter ip address> 1.3.6.1.4.1.9.9.156.1.9.4.0 i <value>`

These are related to registration/deregistration/failure of phones.

You need to ensure that notification destinations are configured. This can be done from the Serviceability Web window. There is a menu for SNMP > Notification destination.

Before you configure notification destination, verify that the required SNMP services are activated and running (SNMP Master Agent and Cisco CallManager SNMP Services). Also, make sure that you configured the privileges for the community string/user correctly which should contain Notify permissions as well.

If still Traps are not generated check if corresponding alarms are generated. Since these traps are generated based on the alarm events, ensure that SNMP agents are getting these alarm events. Enable 'Local Syslog', setup the Cisco Unified CM Alarm configuration to 'Informational' level for 'Local
Syslog' destination from the Alarm configuration available on Cisco Unified CM Serviceability web page->Alarm->Configuration. Then repro the traps and see if corresponding alarms are logged in CiscoSyslog file.

**Receiving syslog messages as traps**—To receive syslog messages above a particular severity as traps, set the following 2 MIB objects in the clogBasic table:

- **clogNotificationsEnabled (1.3.6.1.4.1.9.9.41.1.1.2)**—Set this to true(1) to enable syslog trap notification. Default value is false (2). For example, `snmpset -c <Community String> -v 2c <transmitter ip address> 1.3.6.1.4.1.9.9.41.1.1.2.0 i <value>.

- **clogMaxSeverity (1.3.6.1.4.1.9.9.41.1.1.3)**—Set the severity level above which traps are desired. Default value is warning (5). All syslog messages with alarm severity lesser than or equal to configured severity level will be sent as traps if notification is enabled. For example, `snmpset -c <Community String> -v 2c <transmitter ip address> 1.3.6.1.4.1.9.9.41.1.1.3.0 i <value>

**What are the different traps defined for Cisco Unified Communication Manager?**

The CISCO-CCM-MIB contains the traps related information. Following are the list of defined traps defined:

- **ccmCallManagerFailed**—Indication that the CallManager process detects a failure in one of its critical subsystems. It can also be detected from a heartbeat/event monitoring process.

- **ccmPhoneFailed**—Notification that the intervals specified in ccmPhoneFailedAlarmInterval indicate at least one entry in the ccmPhoneFailedTable.

- **ccmPhoneStatusUpdate**—Notification that is generated in the intervals specified in ccmPhoneStatusUpdateInterval if there is at least one entry in the ccmPhoneStatusUpdateTable.

- **ccmGatewayFailed**—Indication that at least one gateway has attempted to register or communicate with the CallManager and failed.

- **ccmMediaResourceListExhausted**—Indication that the CallManager has run out a certain specified type of resource

- **ccmRouteListExhausted**—Indication that the CallManager could not find an available route in the indicated route list.

- **ccmGatewayLayer2Change**—Sent when the D-Channel/Layer 2 of an interface in a skinny gateway that has registered with the CallManager changes state.

- **ccmMaliciousCall**—Sent when a user registers a call as malicious with the local Cisco Unified CM

- **ccmQualityReport**—Sent when a user reports a quality problem using the Quality Report Tool

- **ccmTLSConnectionFailure**—Sent when CallManager fails to open TLS connection for the indicated device

The mapping of the traps to alarms is as follows:

- **ccmCallManagerFailed**—CallManagerFailure

- **ccmPhoneFailed**—DeviceTransientConnection

- **ccmPhoneStatusUpdate**

- **ccmGatewayFailed**—DeviceTransientConnection

- **ccmMaliciousCall**—MaliciousCall

- **ccmMediaResourceListExhausted**—MediaResourceListExhausted
How can different SNMP traps from Cisco Unified Communication Manager be checked?

Following is the procedure for triggering few traps:

- **ccmPhoneStatusUpdate trap**
  - Set ccmPhoneStatusUpdateAlarmInterval (1.3.6.1.4.1.9.9.156.1.9.4) to 30 or higher in ccmAlarmConfigInfo MIB table.
  - Disconnect a ccm server that your phones are pointing to.
  - Phones will unregister.
  - Connect the ccm server again.
  - Phones will re-register.
  - Will get the ccmPhoneStatusUpdate trap.

- **ccmPhoneFailed trap**
  - Set ccmPhoneFailedAlarmInterval (1.3.6.1.4.1.9.9.156.1.9.2) to 30 or higher in ccmAlarmConfigInfo MIB table.
  - Make a phone fail. Delete a phone from CM and register the phone again.
  - For phone failed traps two different scenarios can be tried:
    - Set the phone to point to tftp/ccm server A. plugin the phone to ccm server B on different switch. The phone status is unknown. Will see following: 2007-10-31:2007-10-31 14:53:40 Local7.Debug 172.19.240.221 community=public, enterprise=1.3.6.1.4.1.9.9.156.2.0.2, enterprise_mib_name=ccmPhoneFailed, uptime=7988879, agent_ip=128.107.143.68, version=Ver2, ccmAlarmSeverity=error, ccmPhoneFailures=1.
    - Register a 7960 phone as 7940 phone in the Cisco Unified CM and thus cause the db issue that makes the phone fail trap.

- **MediaResourceListExhausted trap**
  - Create a Media Resource Group (MRG), have it contains one of the standard ConferenceBridge resource (CFB-2).
  - Create a Media Resource Group List (MRGL), have it contains the MRG just created.
  - In the Phone Configuration page for real phones, set MRGL as the phone Media Resource Group List.
  - Stop the IPVMS which make the ConferenceBridge resource (CFB-2) stop working.
  - Make conference calls with phones that using the media list, you will see "No Conference Bridge available" in the phone screen.
  - Then check if a "MediaListExhausted" Alarm/Alert/Trap is generated.

- **RouteListExhausted trap**
  - Create a Route Group (RG), have it contains one Gateway.
  - Create a Route Group List (RGL), have it contains the RG just created.
  - Create a Route Pattern (9.XXXX) that reroute a 9XXXX call through the RGL.
  - Unregister the gateway.
– Dial 9XXXX in one of the phone.
– Then check if a "RouteListExhausted" Alarm/Alert/Trap is generated.

• MaliciousCallFailed trap
– Similar as QRT, create a softkey template. In the template, add all available "MaliciousCall" softkey to the phone's different status.
– Assign the new softkey template to real phones, reset the phones.
– Making calls, select the "MaliciousCall" in the phone screen during or after the call.
– Then check if a "MaliciousCallFailed" Alarm/Alert/Trap is generated

• GatewayFailed trap (Method 1)
– Remove the configuration of the gateway from the database through Web Admin (or) Change the MAC address of the gateway to some invalid value and update.
– Reboot the gateway
– Another way is to restart the Cisco Unified CM service to which the gateway is connected.

• GatewayFailed trap (Method 2)
– Set GatewayAlarmEnable=true in ccmAlarmConfigInfo mib table
– In ccm serviceability->Snmp configuration page, make sure you have SNMP community string and trap destination set correctly.
– Create a gateway failure event and the trap will be seen on the trap receiver.
– To cause a gateway fail, Restart Cisco Unified CM service which will cause gateway failover to the redundant ccm manager server. On that server, the gateway should not be configured in the database.

• ccmGatewayLayer2Change trap
– ccmGatewayLayer2Change trap is triggered during DChannelOOS(D Channel Out of service) or DChannelISV (D Channel Inservice) from Cisco Unified CM. Please check if any such events can be triggered to test it out

• ccmCallManagerFailed trap
– The CallManager Failed Alarm is generated when an internal error is encountered. These include an internal thread dying due to lack of CPU, timer issues and a couple others. This trap would be something that is hard to reproduce unless the CallManager team give a friendly that intentionally causes one of these occurrences.

If the Cisco Unified CM Agent consumes high CPU continuously, what needs to be done?
Collect the logs as mentioned above (under Troubleshooting) for analysis and refer to defect CSCsm74316 to check if it is being hit. Verify if the fix for the defect has gone into the Cisco Unified CM version used by the customer.

If the CTI Routepoint is deleted from Cisco Unified CM Admin UI, an entry exists for that in ccmCTIDeviceTable mib. Why?
There is service parameter called “RIS Unused Cisco CallManager Device Store Period” which defines how long Unregistered devices (when a registered device is removed from db, it unregisters) will remain in RISDB and hence in the MIB. The ccmadmin page and the SNMP MIB WALK may or may not be in sync, since the ccmadmin page shows the info from the database however SNMP uses the RISDB.
When ccmPhoneType is queried from ccmPhoneTable in Cisco-CCM-MIB, no information is returned. Why?
The ccmPhoneType has been made obsolete. The same information can be retrieved from ccmPhoneProductTypeIndex against CcmProductTypeEntry. In the table, the indexes correspond to the index and name as listed in that table.

Some of other obsolete and alternate OIDs to be referred:
- ccmGatewayType is obsolete and need to refer ccmGateWayProductTypeIndex.
- ccmMediaDeviceType is obsolete and need to refer to ccmMediaDeviceProductTypeIndex
- ccmCTIDeviceType is obsolete and need to refer to ccmCTIDeviceProductTypeIndex

A query on ccmPhoneProductTypeIndex returns zero. Why?
Verify that the Cisco Unified CM release that you are using has this capability.

While performing a WALK on ccmPhoneTable, ccmPhoneUserName is not returning any value. How are usernames associated to the IP Phones?
Create an end user and then go to the phone that has been registered and associate the Owner User ID. Once this is done, the user will be shown by the OID in the SNMP Walk.

How do I get the firmware versions of each phone by using SNMP?
ccmPhoneLoadID object in the ccmPhoneTable will give the firmware version of each phone. But this value may differ if new image download failed. In case of 7.x versions SNMP will expose both configured firmware ID (ccmPhoneLoadID) and the actual running firmware (ccmPhoneActiveLoad).

CCM MIB returns ccmVersion as 5.0.1, which is the incorrect.
Verify the Cisco Unified CM release that you are using has this capability. If it does not, upgrade.

CCM MIB returns incorrect ccmPhoneLoadID
ccmPhoneLoadID values are picked up from RISDB which is populated based on the alarm received during Phone registration. Perform the following steps and collect the logs for further analysis:

Step 1 Go to Serviceability web page -> Alarm -> Configuration -> Service Group (CM Services) -> Service (Cisco CallManager).
Step 2 Check Local Syslog, SDI Trace, SDL Trace. Ensure the Alarm Event Level for these selected destinations is set to Informational.
Step 3 Set the Cisco CallManager trace level to Detailed.
Step 4 Reset the phones showing incorrect LoadID.
Step 5 Collect the Syslog and Cisco CallManager traces.
Step 6 Collect the phone details.

How Cisco Unified CM status (START/STOP) monitored?
For service monitoring we have following options:
- SYSAPPL MIB
- HOST-RESOURCE-MIB
- CISCO-CCM-MIB (ccmStatus)
• SOAP interface
• Real-TimeMonitoringTool (RTMT) alerts

There is a ccmCallManagerFailed trap for Cisco Unified CM service failure. But this does not cover normal service stop and unknown crashes.

The device pool information seems incorrect for any device polled for. The OID used is ccmPhoneDevicePoolIndex.

As stated in the CISCO-CCM-CAPABILITY MIB, ccmPhoneDevicePoolIndex is not supported, hence it returns 0. The CallManager device registration alarm currently does not contain the devicepool information.

CISCO-CCM-CAPABILITY

Note
This is a reformatted version of CISCO-CCM-CAPABILITY. Download and compile all of the MIBs in this section from http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

Note
This MIB is not meant to perform SNMP queries like MIB walk as there is no agent supporting this MIB. It is only used as documentation supplement to the CISCO-CCM-MIB.

Before you can compile CISCO-CCM-CAPABILITY, you need to compile the MIBs listed below in the order listed.

1. SNMPv2-SMI
2. SNMPv2-TC
3. SNMPv2-CONF
4. SNMPv2-MIB
5. IANAifType-MIB
6. IF-MIB
7. CISCO-SMI
8. SNMP-FRAMEWORK-MIB
9. RMON-MIB
10. CISCO-TC
11. CISCO-VTP-MIB
12. RFC1155-SMI
13. RFC-1212
14. SNMPv2-TC-v1
15. CISCO-CDP-MIB
16. CISCO-CCM-CAPABILITY
CISCO-CCM-CAPABILITY

Additional downloads are:
- OID File: CISCO-CCM-CAPABILITY.OID

The following are contained in this section:
- Revisions, page 7-122
- Definitions, page 7-122
- Agent Capabilities, page 7-122

Revisions

Table 7-1 lists the revisions to this MIB beginning with the latest revision first.

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10-03-2003</td>
<td>Added</td>
<td>Agent capability for CISCO-CCM-MIB</td>
</tr>
<tr>
<td>10-03-2003</td>
<td>Added</td>
<td>Agent capabilities for Cisco Call Manager 4.0 release</td>
</tr>
<tr>
<td>03-21-2002</td>
<td>Added</td>
<td>DESCRIPTION Added the agent capabilities for Cisco Call Manager 3.3 release.</td>
</tr>
<tr>
<td>07-02-2001</td>
<td>Added</td>
<td>DESCRIPTION Added the agent capabilities for Cisco Call Manager 3.0 release.</td>
</tr>
<tr>
<td>06-19-2001</td>
<td>Initial</td>
<td>::= { ciscoAgentCapability 211 }</td>
</tr>
</tbody>
</table>

Definitions

The following definitions are imported for CISCO-CCM-CAPABILITY:
- MODULE-IDENTITY
- From SNMPv2-SMI—AGENT-CAPABILITIES
- From SNMPv2-CONF—ciscoAgentCapability
- From CISCO-SMI—ciscoCCMCapability MODULE-IDENTITY

Agent Capabilities

ciscoCCMCapabilityV3R00 AGENT-CAPABILITIES
PRODUCT RELEASE Cisco Call Manager 3.0
STATUS Current
DESCRIPTION Cisco Call Manager Agent Capabilities
SUPPORTS Cisco-ccm-mib
INCLUDES { ccmInfoGroup, ccmPhoneInfoGroup, ccmGatewayInfoGroup }
VARIATION ccmPhoneE911Location
ACCESS not-implemented
DESCRIPTION ccmPhoneE911Location is not supported
VARIATION ccmPhoneLastError
ACCESS not-implemented
DESCRIPTION ccmPhoneLastError is not supported
VARIATION ccmPhoneTimeLastError
ACCESS not-implemented
DESCRIPTION ccmPhoneTimeLastError is not supported
VARIATION ccmPhoneDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmPhoneDevicePoolIndex is not supported
VARIATION ccmGatewayDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmGatewayDevicePoolIndex is not supported
VARIATION ccmGatewayTrunkIndex
ACCESS not-implemented
DESCRIPTION ccmGatewayTrunkIndex is not supported
VARIATION ccmGatewayTrunkType
ACCESS not-implemented
DESCRIPTION ccmGatewayTrunkType is not supported
VARIATION ccmGatewayTrunkName
ACCESS not-implemented
DESCRIPTION ccmGatewayTrunkName is not supported
VARIATION ccmTrunkGatewayIndex
ACCESS not-implemented
DESCRIPTION ccmTrunkGatewayIndex is not supported
VARIATION ccmGatewayTrunkStatus
ACCESS not-implemented
DESCRIPTION ccmGatewayTrunkStatus is not supported
::= { ciscoCCMCapability 1 }
ciscoCCMCapabilityV3R01 AGENT-CAPABILITIES
PRODUCT-RELEASE Cisco Call Manager 3.1
STATUS current
DESCRIPTION Cisco Call Manager Agent capabilities
SUPPORTS CISCO-CCM-MIB
INCLUDES { ccmInfoGroupRev1, ccmPhoneInfoGroupRev1, ccmGatewayInfoGroupRev1, ccmMediaDeviceInfoGroup, ccmGatekeeperInfoGroup, ccmCTIDeviceInfoGroup, ccmNotificationsInfoGroup, ccmNotificationsGroup }
VARIATION ccmPhoneE911Location
ACCESS not-implemented
DESCRIPTION ccmPhoneE911Location is not supported
VARIATION ccmPhoneLastError
ACCESS not-implemented
DESCRIPTION ccmPhoneLastError is not supported
VARIATION ccmPhoneTimeLastError
ACCESS not-implemented
DESCRIPTION ccmPhoneTimeLastError is not supported
VARIATION ccmPhoneDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmPhoneDevicePoolIndex is not supported
VARIATION ccmGatewayDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmGatewayDevicePoolIndex is not supported
VARIATION ccmMediaDeviceDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmMediaDeviceDevicePoolIndex is not supported
VARIATION ccmGatekeeperDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmGatekeeperDevicePoolIndex is not supported
VARIATION ccmCTIDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmCTIDevicePoolIndex is not supported
VARIATION ccmCTIDeviceAppInfo
ACCESS not-implemented
DESCRIPTION ccmCTIDeviceAppInfo is not supported
VARIATION ccmPhonePhysicalAddress
SYNTAX MacAddress
DESCRIPTION Represents the MAC address of the phone
::= { ciscoCCMCapability 2 }

ciscoCCMCapabilityV3R03 AGENT-CAPABILITIES
PRODUCT-RELEASE Cisco Call Manager 3.3
STATUS obsolete and superseded by ciscoCCMCapabilityV3R03Rev1
DESCRIPTION Cisco Call Manager Agent capabilities
SUPPORTS CISCO-CCM-MIB
VARIATION  ccmPhoneE911Location
ACCESS not-implemented
DESCRIPTION ccmPhoneE911Location is not supported

VARIATION  ccmPhoneDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmPhoneDevicePoolIndex is not supported

VARIATION  ccmGatewayDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmGatewayDevicePoolIndex is not supported

VARIATION  ccmMediaDeviceDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmMediaDeviceDevicePoolIndex is not supported

VARIATION  ccmCTIDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmCTIDevicePoolIndex is not supported

VARIATION  ccmPhoneFailedTable
DESCRIPTION The table containing the list of all phones which attempted to register with the local call manager and failed. The entries which have not been updated and kept at least for the duration specified in the ccmPhoneFailedStorePeriod will be deleted. Reasons for these failures could be due to configuration error, maximum number of phones has been reached, lost contact, etc.

VARIATION  ccmPhoneStatusUpdateTableStateId
DESCRIPTION The current state of ccmPhoneStatusUpdateTable. The initial value of this object is 0 and it will be incremented everytime when there is a change (addition/deletion/modification) to the ccmPhoneStatusUpdateTable. This value and sysUpTime should be used together to find if the table has changed or not. When the SNMP service is restarted this value will be reset to 0.

VARIATION  ccmPhStatUpdtTblLastAddedIndex
SYNTAX CcmIndexOrZero
DESCRIPTION The ccmPhoneStatusUpdateIndex value of the last entry that was added to the ccmPhoneStatusUpdateTable. This value together with sysUpTime can be used by the manager applications to identify the new entries in the ccmPhoneStatusUpdateTable since their last poll. This value need not be the same as the highest index in the ccmPhoneStatusUpdateTable as the index could have wrapped around. The initial value of this object is 0 which indicates that there has been no entries added to this table. When the SNMP service is restarted this value will be reset to 0.

VARIATION  ccmPhFailedTblLastAddedIndex
SYNTAX CcmIndexOrZero
DESCRIPTION The ccmPhoneFailedIndex value of the last entry that was added to the ccmPhoneFailedTable. This value together with sysUpTime can be used by the manager applications to identify the new entries in the ccmPhoneFailedTable since their last poll. This value need not be the same as the highest index in the ccmPhoneFailedTable as the index could have wrapped around. The initial value of this object is 0 which indicates that there has been no entries added to this table. When the SNMP service is restarted this value will be reset to 0.

VARIATION  ccmPhoneFailedStorePeriod
DESCRIPTION The time duration for storing each entry in the ccmPhoneFailedTable. The entries which have not been updated and kept at least this period will be deleted. This value should ideally be set to a higher value than the ccmPhoneFailedAlarmInterval object. The default value is 1800 seconds.

::= { ciscoCCMCapability 3 }

ciscoCCMCapabilityV3R03Rev1 AGENT-CAPABILITIES

PRODUCT-RELEASE Cisco Call Manager 3.3

STATUS current

DESCRIPTION Cisco Call Manager Agent capabilities

SUPPORTS CISCO-CCM-MIB


VARIATION ccmPhoneE911Location

ACCESS not-implemented

DESCRIPTION ccmPhoneE911Location is not supported

VARIATION ccmPhoneDevicePoolIndex

ACCESS not-implemented

DESCRIPTION ccmPhoneDevicePoolIndex is not supported

VARIATION ccmGatewayDevicePoolIndex

ACCESS not-implemented

DESCRIPTION ccmGatewayDevicePoolIndex is not supported

VARIATION ccmMediaDeviceDevicePoolIndex

ACCESS not-implemented

DESCRIPTION ccmMediaDeviceDevicePoolIndex is not supported

VARIATION ccmCTIDevicePoolIndex

ACCESS not-implemented

DESCRIPTION ccmCTIDevicePoolIndex is not supported

::= { ciscoCCMCapability 4 }

ciscoCCMCapabilityV4R00 AGENT-CAPABILITIES

PRODUCT-RELEASE Cisco Call Manager 4.0

STATUS current

DESCRIPTION Cisco Call Manager Agent capabilities

SUPPORTS CISCO-CCM-MIB


VARIATION ccmPhoneE911Location
ACCESS not-implemented
DESCRIPTION ccmPhoneE911Location is not supported
VARIATION  ccmPhoneDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmPhoneDevicePoolIndex is not supported
VARIATION  ccmGatewayDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmGatewayDevicePoolIndex is not supported
VARIATION  ccmMediaDeviceDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmMediaDeviceDevicePoolIndex is not supported
VARIATION  ccmCTIDevicePoolIndex
ACCESS not-implemented
DESCRIPTION ccmCTIDevicePoolIndex is not supported

::= { ciscoCCMCapability 5 }

---

**CISCO-CDP-MIB**


This MIB is for the management of the Cisco Discovery Protocol (CDP) in Cisco devices. Before you can compile CISCO-CDP-MIB, you need to compile the MIBs listed below in the order listed.

1. SNMPv2-SMI
2. SNMPv2-TC
3. SNMPv2-CONF
4. SNMPv2-MIB
5. IANAifType-MIB
6. IF-MIB
7. CISCO-SMI
8. SNMP-FRAMEWORK-MIB
9. RMON-MIB
10. CISCO-TM
11. CISCO-VTP-MIB
12. RFC1155-SMI
13. RFC-1212
14. SNMPv2-TC-v1
15. CISCO-CDP-MIB

Additional downloads are:
- OID File: CISCO-CDP-MIB.oid
- Capability File: CISCO-CDP-CAPABILITY

The following are contained in this section:
- Revisions, page 7-128
- Definitions, page 7-129
- CDP Interface Group, page 7-129
- CDP Address Cache Group, page 7-132
- CDP Global Group, page 7-139
- Conformance Information, page 7-140
- Compliance Statements, page 7-141
- Units Of Conformance, page 7-141
- Troubleshooting, page 7-143

Revisions

Table 7-6 lists the revision to this MIB beginning with the lastest revision.

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>11-23-2001</td>
<td>Added</td>
<td>cdpInterfaceExtTable which contains the following objects:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpInterfaceExtendedTrust, cdpInterfaceCosForUntrustedPort</td>
</tr>
<tr>
<td>04-23-2001</td>
<td>Added</td>
<td>cdpGlobalDeviceIdFormatCpb,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpGlobalDeviceIdFormatCpb, cdpGlobalDeviceIdFormat</td>
</tr>
<tr>
<td>11-22-2000</td>
<td>Added</td>
<td>cdpCacheApplianceID, cdpCacheVlanID,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpCachePowerConsumption, cdpCacheMTU,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpCachePrimaryMgmtAddrType,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpCachePrimaryMgmtAddr,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpCacheSecondaryMgmtAddrType,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpCacheSecondaryMgmtAddr,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpCacheLastChange,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpCachePhysLocation,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpCacheSysName,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpCacheSysObjectID,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdpGlobalLastChange</td>
</tr>
<tr>
<td>12-10-1998</td>
<td>Added</td>
<td>cdpGlobalDeviceId</td>
</tr>
</tbody>
</table>
### Definitions

The following definitions are imported for CISCO-CDP-MIB:

- MODULE-IDENTITY, OBJECT-TYPE, Integer32
- From SNMPv2-SMI—MODULE-COMPLIANCE, OBJECT-GROUP
- From SNMPv2-CONF—TruthValue, DisplayString, TimeStamp
- From SNMPv2-TC—ciscoMgmt
- From CISCO-SMI—CiscoNetworkProtocol, CiscoNetworkAddress, Unsigned32
- From CISCO-TC—VlanIndex
- From CISCO-VTP-MIB—ifIndex
- From IF-MIB—ciscoCdpMIB MODULE-IDENTITY

### CDP Interface Group

**cdpInterfaceTable OBJECT-TYPE**

SYNTAX SEQUENCE OF CdpInterfaceEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

The (conceptual) table containing the status of CDP on the device interfaces.

```asn1
::= { cdpInterface 1 }
```

**cdpInterfaceEntry OBJECT-TYPE**

SYNTAX CdpInterfaceEntry

MAX-ACCESS not-accessible

### Table 7-6 History of Revisions (continued)

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>09-16-1998</td>
<td>Added</td>
<td>These objects to cdpCacheTable: cdpCacheVTPMgmtDomain, cdpCacheNativeVLAN, cdpCacheDuplex</td>
</tr>
<tr>
<td>07-08-1996</td>
<td>Obsoleted and defined cdpGlobal</td>
<td>cdpInterfaceMessageInterval</td>
</tr>
<tr>
<td>08-15-1995</td>
<td>—</td>
<td>Specified a correct (non-negative) range for several index objects</td>
</tr>
<tr>
<td>07-27-1995</td>
<td>—</td>
<td>Corrected range of cdpInterfaceMessageInterval</td>
</tr>
<tr>
<td>01-25-1995</td>
<td>Moved from ciscoExperiment to ciscoMgmt OID subtree ::= { ciscoMgmt 23 }</td>
<td>ciscoCdpMIBObjects OBJECT IDENTIFIER ::= { ciscoCdpMIB 1 } cdpInterface OBJECT IDENTIFIER ::= { ciscoCdpMIBObjects 1 } cdpCache OBJECT IDENTIFIER ::= { ciscoCdpMIBObjects 2 } cdpGlobal OBJECT IDENTIFIER ::= { ciscoCdpMIBObjects 3 }</td>
</tr>
</tbody>
</table>
STATUS current
DESCRIPTION
An entry (conceptual row) in the cdpInterfaceTable, containing the status of CDP on an interface.
INDEX { cdpInterfaceIfIndex }
::= { cdpInterfaceTable 1 }

CdpInterfaceEntry ::= SEQUENCE {
  cdpInterfaceIfIndex   Integer32,
  cdpInterfaceEnableTruthValue,
  cdpInterfaceMessageInterval   INTEGER,
  cdpInterfaceGroup Integer32,
  cdpInterfacePort  Integer32
}

cdpInterfaceIfIndex OBJECT-TYPE
SYNTAX Integer32 (0..2147483647)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The ifIndex value of the local interface. For 802.3 Repeaters on which the repeater ports do not have
ifIndex values assigned, this value is a unique value for the port, and greater than any ifIndex value
supported by the repeater; in this case, the specific port is indicated by corresponding values of
cdpInterfaceGroup and cdpInterfacePort, where these values correspond to the group number and
port number values of RFC 1516.
::= { cdpInterfaceEntry 1 }

cdpInterfaceEnable OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
An indication of whether the Cisco Discovery Protocol is currently running on this interface. This
variable has no effect when CDP is disabled (cdpGlobalRun = FALSE).
::= { cdpInterfaceEntry 2 }

cdpInterfaceMessageInterval OBJECT-TYPE
SYNTAX INTEGER (5..254)
UNITS seconds
MAX-ACCESS read-write
STATUS obsolete and replaced by cdpGlobalMessageInterval. This object should be applied to the
whole system instead of per interface.
DESCRIPTION
The interval at which CDP messages are to be generated on this interface. The default value is 60
seconds.
cdpInterfaceGroup OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object is only relevant to interfaces which are repeater ports on 802.3 repeaters. In this situation, it indicates the RFC1516 group number of the repeater port which corresponds to this interface.

cdpInterfacePort OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object is only relevant to interfaces which are repeater ports on 802.3 repeaters. In this situation, it indicates the RFC1516 port number of the repeater port which corresponds to this interface.

cdpInterfaceExtTable OBJECT-TYPE
SYNTAX SEQUENCE OF CdpInterfaceExtEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
This table contains the additional CDP configuration on the device interfaces.

cdpInterfaceExtEntry OBJECT-TYPE
SYNTAX CdpInterfaceExtEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry in the cdpInterfaceExtTable contains the values configured for Extented Trust TLV and COS (Class of Service) for Untrusted Ports TLV on an interface which supports the sending of these TLVs.
INDEX  { ifIndex }

CdpInterfaceExtEntry ::= SEQUENCE {
cdpInterfaceExtendedTrustINTEGER, cdpInterfaceCosForUntrustedPort Unsigned32}
cdpInterfaceExtendedTrust OBJECT-TYPE
SYNTAX INTEGER {trusted(1), noTrust(2) }
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Indicates the value to be sent by Extended Trust TLV. If trusted(1) is configured, the value of
Extended Trust TLV is one byte in length with its least significant bit equal to 1 to indicate extended
trust. All other bits are 0. If noTrust(2) is configured, the value of Extended Trust TLV is one byte
in length with its least significant bit equal to 0 to indicate no extended trust. All other bits are 0.
::= { cdpInterfaceExtEntry 1 }

cdpInterfaceCosForUntrustedPort OBJECT-TYPE
SYNTAX Unsigned32 (0..7)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Indicates the value to be sent by COS for Untrusted Ports TLV.
::= { cdpInterfaceExtEntry 2 }

CDP Address Cache Group

cdpCacheTable OBJECT-TYPE
SYNTAX SEQUENCE OF CdpCacheEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The (conceptual) table containing the cached information obtained via receiving CDP messages.
::= { cdpCache 1 }

cdpCacheEntry OBJECT-TYPE
SYNTAX CdpCacheEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry (conceptual row) in the cdpCacheTable, containing the information received via CDP on
one interface from one device. Entries appear when a CDP advertisement is received from a
neighbor device. Entries disappear when CDP is disabled on the interface, or globally.
INDEX  { cdpCacheIfIndex, cdpCacheDeviceIndex }
::= { cdpCacheTable 1 }
CdpCacheEntry ::= SEQUENCE {
cdpCacheIfIndex INTEGER32,

cdpCacheDeviceIndex INTEGER32,

cdpCacheAddressType CiscoNetworkProtocol,

cdpCacheAddress CiscoNetworkAddress,

cdpCacheVersion DisplayString,

cdpCacheDeviceId DisplayString,

cdpCacheDevicePort DisplayString,

cdpCachePlatform DisplayString,

cdpCacheCapabilites OCTET STRING,

cdpCacheVTMPmgDomain DisplayString,

cdpCacheNativeVLAN VLANIndex,

cdpCacheDuplex INTEGER,

cdpCacheApplianceID Unsigned32,

cdpCacheVlanID Unsigned32,

cdpCachePowerConsumption Unsigned32,

cdpCacheMTU Unsigned32,

cdpCacheSysName DisplayString,

cdpCacheSysObjectID OBJECT IDENTIFIER,

cdpCachePrimaryMgmtAddrType CiscoNetworkProtocol,

cdpCachePrimaryMgmtAddr CiscoNetworkAddress,

cdpCacheSecondaryMgmtAddrType CiscoNetworkProtocol,

cdpCacheSecondaryMgmtAddr CiscoNetworkAddress,

cdpCachePhysLocation DisplayString,

cdpCacheLastChange TimeStamp

}

**cdpCacheIfIndex** OBJECT-TYPE

SYNTAX Integer32 (0..2147483647)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

Normally, the ifIndex value of the local interface. For 802.3 repeaters for which the repeater ports

do not have ifIndex values assigned, this value is a unique value for the port, and greater than any

ifIndex value supported by the repeater; the specific port number in this case, is given by the

corresponding value of cdpInterfacePort.

::= { cdpCacheEntry 1 }

**cdpCacheDeviceIndex** OBJECT-TYPE

SYNTAX Integer32 (0..2147483647)

MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A unique value for each device from which CDP messages are being received.
 ::= { cdpCacheEntry 2 }

cdpCacheAddressType OBJECT-TYPE
SYNTAX CiscoNetworkProtocol
MAX-ACCESS read-only
STATUS current
DESCRIPTION
An indication of the type of address contained in the corresponding instance of cdpCacheAddress.
 ::= { cdpCacheEntry 3 }

cdpCacheAddress OBJECT-TYPE
SYNTAX CiscoNetworkAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The (first) network-layer address of the device's SNMP-agent as reported in the Address TLV of the most recently received CDP message. For example, if the corresponding instance of cacheAddressType had the value 'ip(1)', then this object would be an IP-address.
 ::= { cdpCacheEntry 4 }

cdpCacheVersion OBJECT-TYPE
SYNTAX DisplayString
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The Version string as reported in the most recent CDP message. The zero-length string indicates no Version field (TLV) was reported in the most recent CDP message.
 ::= { cdpCacheEntry 5 }

cdpCacheDeviceId OBJECT-TYPE
SYNTAX DisplayString
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The Device-ID string as reported in the most recent CDP message. The zero-length string indicates no Device-ID field (TLV) was reported in the most recent CDP message.
 ::= { cdpCacheEntry 6 }

cdpCacheDevicePort OBJECT-TYPE
SYNTAX DisplayString
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The Port-ID string as reported in the most recent CDP message. This will typically be the value of the ifName object (e.g. Ethernet0). The zero-length string indicates no Port-ID field (TLV) was reported in the most recent CDP message.
 ::= { cdpCacheEntry 7 }

cdpCachePlatform OBJECT-TYPE
SYNTAX DisplayString
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The Device Hardware Platform as reported in the most recent CDP message. The zero-length string indicates that no Platform field (TLV) was reported in the most recent CDP message.
 ::= { cdpCacheEntry 8 }

cdpCacheCapabilities OBJECT-TYPE
SYNTAX OCTET STRING (SIZE (0..4))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The Device Functional Capabilities as reported in the most recent CDP message. For latest set of specific values, see the latest version of the CDP specification. The zero-length string indicates no Capabilities field (TLV) was reported in the most recent CDP message.
REFERENCE Cisco Discovery Protocol Specification, 10/19/94.
 ::= { cdpCacheEntry 9 }

cdpCacheVTPMgmtDomain OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..32))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The VTP Management Domain for the remote device interface, as reported in the most recently received CDP message. This object is not instantiated if no VTP Management Domain field (TLV) was reported in the most recently received CDP message.
REFERENCE managementDomainName in CISCO-VTP-MIB
 ::= { cdpCacheEntry 10 }

cdpCacheNativeVLAN OBJECT-TYPE
SYNTAX VlanIndex
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The remote device interface native VLAN, as reported in the most recent CDP message. The value 0 indicates no native VLAN field (TLV) was reported in the most recent CDP message.
::= { cdpCacheEntry 11 }

cdpCacheDuplex OBJECT-TYPE
SYNTAX INTEGER { unknown(1), halfduplex(2), fullduplex(3) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The remote device interface duplex mode, as reported in the most recent CDP message. The value unknown(1) indicates no duplex mode field (TLV) was reported in the most recent CDP message.
::= { cdpCacheEntry 12 }

cdpCacheApplianceID OBJECT-TYPE
SYNTAX Unsigned32 (0..255)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The remote device Appliance ID, as reported in the most recent CDP message. This object is not instantiated if no Appliance VLAN-ID field (TLV) was reported in the most recently received CDP message.
::= { cdpCacheEntry 13 }

cdpCacheVlanID OBJECT-TYPE
SYNTAX Unsigned32 (0..4095)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The remote device VoIP VLAN ID, as reported in the most recent CDP message. This object is not instantiated if no Appliance VLAN-ID field (TLV) was reported in the most recently received CDP message.
::= { cdpCacheEntry 14 }

cdpCachePowerConsumption OBJECT-TYPE
SYNTAX Unsigned32
UNITS milliwatts
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The amount of power consumed by remote device, as reported in the most recent CDP message. This object is not instantiated if no Power Consumption field (TLV) was reported in the most recently received CDP message.
::= { cdpCacheEntry 15 }

cdpCacheMTU OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Indicates the size of the largest datagram that can be sent/received by remote device, as reported in
the most recent CDP message. This object is not instantiated if no MTU field (TLV) was reported
in the most recently received CDP message.
::= { cdpCacheEntry 16 }

cdpCacheSysName OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..255))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Indicates the value of the remote device sysName MIB object. By convention, it is the device fully
qualified domain name. This object is not instantiated if no sysName field (TLV) was reported in
the most recently received CDP message.
::= { cdpCacheEntry 17 }

cdpCacheSysObjectID OBJECT-TYPE
SYNTAX OBJECT IDENTIFIER
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Indicates the value of the remote device sysObjectID MIB object. This object is not instantiated if
no sysObjectID field (TLV) was reported in the most recently received CDP message.
::= { cdpCacheEntry 18 }

cdpCachePrimaryMgmtAddrType OBJECT-TYPE
SYNTAX CiscoNetworkProtocol
MAX-ACCESS read-only
STATUS current
DESCRIPTION
An indication of the type of address contained in the corresponding instance of
cdpCachePrimaryMgmtAddress.
::= { cdpCacheEntry 19 }

cdpCachePrimaryMgmtAddr OBJECT-TYPE
SYNTAX CiscoNetworkAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object indicates the (first) network layer address at which the device will accept SNMP
messages as reported in the most recently received CDP message. If the corresponding instance of
cdpCachePrimaryMgmtAddrType has the value ip(1), then this object would be an IP-address. If the
remote device is not currently manageable via any network protocol, this object has the special
value of the IPv4 address 0.0.0.0. If the most recently received CDP message did not contain any primary address at which the device prefers to receive SNMP messages, then this object is not instantiated.

::= { cdpCacheEntry 20 }

cdpCacheSecondaryMgmtAddrType OBJECT-TYPE
SYNTAX CiscoNetworkProtocol
MAX-ACCESS read-only
STATUS current
DESCRIPTION
An indication of the type of address contained in the corresponding instance of cdpCacheSecondaryMgmtAddress.

::= { cdpCacheEntry 21 }

cdpCacheSecondaryMgmtAddr OBJECT-TYPE
SYNTAX CiscoNetworkAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object indicates the alternate network layer address (other than the one indicated by cdpCachePrimaryMgmtAddr) at which the device will accept SNMP messages as reported in the most recently received CDP message. If the corresponding instance of cdpCacheSecondaryMgmtAddrType has the value ip(1), then this object would be an IP-address. If the most recently received CDP message did not contain such an alternate network layer address, then this object is not instantiated.

::= { cdpCacheEntry 22 }

cdpCachePhysLocation OBJECT-TYPE
SYNTAX DisplayString
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Indicates the physical location, as reported by the most recent CDP message, of a connector which is on, or physically connected to, the remote device's interface over which the CDP packet is sent. This object is not instantiated if no Physical Location field (TLV) was reported by the most recently received CDP message.

::= { cdpCacheEntry 23 }

cdpCacheLastChange OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Indicates the time when this cache entry was last changed. This object is initialised to the current time when the entry gets created and updated to the current time whenever the value of any (other) object instance in the corresponding row is modified.

::= { cdpCacheEntry 24 }

## CDP Global Group

**cdpGlobalRun OBJECT-TYPE**

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

DESCRIPTION

An indication of whether the Cisco Discovery Protocol is currently running. Entries in cdpCacheTable are deleted when CDP is disabled.

DEFVAL { true }

::= { cdpGlobal 1 }

**cdpGlobalMessageInterval OBJECT-TYPE**

SYNTAX INTEGER (5..254)

UNITS seconds

MAX-ACCESS read-write

STATUS current

DESCRIPTION

The interval at which CDP messages are to be generated. The default value is 60 seconds.

DEFVAL { 60 }

::= { cdpGlobal 2 }

**cdpGlobalHoldTime OBJECT-TYPE**

SYNTAX INTEGER (10..255)

UNITS seconds

MAX-ACCESS read-write

STATUS current

DESCRIPTION

The time for the receiving device holds CDP message. The default value is 180 seconds.

DEFVAL { 180 }

::= { cdpGlobal 3 }

**cdpGlobalDeviceId OBJECT-TYPE**

SYNTAX DisplayString

MAX-ACCESS read-only

STATUS current

DESCRIPTION
The device ID advertised by this device. The format of this device id is characterized by the value of cdpGlobalDeviceIdFormat object.

::= { cdpGlobal 4 }

cdpGlobalLastChange OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Indicates the time when the cache table was last changed. It is the most recent time at which any row was last created, modified or deleted.

::= { cdpGlobal 5 }

cdpGlobalDeviceIdFormatCpb OBJECT-TYPE
SYNTAX BITS { serialNumber(0), macAddress(1), other (2) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Indicates the Device-Id format capability of the device. The serialNumber(0) indicates that the device supports using serial number as the format for its DeviceId. The macAddress(1) indicates that the device supports using layer 2 MAC address as the format for its DeviceId. The other(2) indicates that the device supports using its platform specific format as the format for its DeviceId.

::= { cdpGlobal 6 }

cdpGlobalDeviceIdFormat OBJECT-TYPE
SYNTAX INTEGER { serialNumber(1), macAddress(2), other(3) }
MAX-ACCESS read-write
STATUS current
DESCRIPTION
An indication of the format of Device-Id contained in the corresponding instance of cdpGlobalDeviceId. User can only specify the formats that the device is capable of as denoted in cdpGlobalDeviceIdFormatCpb object. The serialNumber(1) indicates that the value of cdpGlobalDeviceId object is in the form of an ASCII string contain the device serial number. The macAddress(2) indicates that the value of cdpGlobalDeviceId object is in the form of Layer 2 MAC address. The other(3) indicates that the value of cdpGlobalDeviceId object is in the form of a platform specific ASCII string contain info that identifies the device. For example: ASCII string contains serialNumber appended/prepended with system name.

::= { cdpGlobal 7 }

Conformance Information

ciscoCdpMIBConformance OBJECT IDENTIFIER ::= { ciscoCdpMIB 2 }
ciscoCdpMIBCompliances OBJECT IDENTIFIER ::= { ciscoCdpMIBConformance 1 }
ciscoCdpMIBGroups OBJECT IDENTIFIER ::= { ciscoCdpMIBConformance 2 }
Compliance Statements

ciscoCdpMIBCompliance MODULE-COMPLIANCE
STATUS obsoleted and superseded by ciscoCdpMIBComplianceV11R01
DESCRIPTION
The compliance statement for the CDP MIB.
MODULE This module
MANDATORY-GROUPS { ciscoCdpMIBGroup }
::= { ciscoCdpMIBCompliances 1 }

ciscoCdpMIBComplianceV11R01 MODULE-COMPLIANCE
STATUS obsoleted and superseded by ciscoCdpMIBComplianceV11R02
DESCRIPTION
The compliance statement for the CDP MIB.
MANDATORY-GROUPS { ciscoCdpMIBGroupV11R01 }
::= { ciscoCdpMIBCompliances 2 }

ciscoCdpMIBComplianceV11R02 MODULE-COMPLIANCE
STATUS obsoleted and superseded by ciscoCdpMIBComplianceV12R02
DESCRIPTION
The compliance statement for the CDP MIB.
MANDATORY-GROUPS { ciscoCdpMIBGroupV11R02 }
::= { ciscoCdpMIBCompliances 3 }

ciscoCdpMIBComplianceV12R02 MODULE-COMPLIANCE
STATUS current
DESCRIPTION
The compliance statement for the CDP MIB.
MANDATORY-GROUPS { ciscoCdpMIBGroupV12R02 }
::= { ciscoCdpMIBCompliances 4 }

Units Of Conformance

ciscoCdpMIBGroup OBJECT-GROUP
OBJECTS { cdpInterfaceEnable, cdpInterfaceMessageInterval, cdpCacheAddressType>cdpCacheAddressType, cdpCacheAddress, cdpCacheVersion, cdpCacheDeviceId, cdpCacheDevicePort, cdpCacheCapabilities, cdpCachePlatform }
STATUS obsoleted and superseded by ciscoCdpMIBGroupV11R01
DESCRIPTION
A collection of objects for use with the Cisco Discovery Protocol.
::= { ciscoCdpMIBGroups 1 }
ciscoCdpMIBGroupV11R01  OBJECT-GROUP
OBJECTS { cdpInterfaceEnable, cdpInterfaceMessageInterval, cdpInterfaceGroup, cdpInterfacePort, cdpCacheAddressType, cdpCacheAddressType, cdpCacheAddress, cdpCacheVersion, cdpCacheDeviceId, cdpCacheDevicePort, cdpCacheCapabilities, cdpCachePlatform }

STATUS  obsoleted and superseded by ciscoCdpMIBGroupV11R02
DESCRIPTION
A collection of objects for use with the Cisco Discovery Protocol.
::= { ciscoCdpMIBGroups 2 }

ciscoCdpMIBGroupV11R02  OBJECT-GROUP
OBJECTS { cdpInterfaceEnable, cdpInterfaceGroup, cdpInterfacePort, cdpCacheAddressType, cdpCacheAddressType, cdpCacheAddress, cdpCacheVersion, cdpCacheDeviceId, cdpCacheDevicePort, cdpCacheCapabilities, cdpCachePlatform, cdpGlobalRun, cdpGlobalMessageInterval, cdpGlobalHoldTime }

STATUS  obsoleted and superseded by ciscoCdpMIBGroupV12R02
DESCRIPTION
A collection of objects for use with the Cisco Discovery Protocol.
::= { ciscoCdpMIBGroups 3 }

ciscoCdpMIBGroupV12R02  OBJECT-GROUP
OBJECTS { cdpInterfaceEnable, cdpInterfaceGroup, cdpInterfacePort, cdpCacheAddressType, cdpCacheAddressType, cdpCacheAddress, cdpCacheVersion, cdpCacheDeviceId, cdpCacheDevicePort, cdpCacheCapabilities, cdpCachePlatform, cdpCacheVTPMgmtDomain, cdpCacheNativeVLAN, cdpCacheDuplex, cdpGlobalRun, cdpGlobalMessageInterval, cdpGlobalHoldTime, cdpGlobalDeviceId }

STATUS  current
DESCRIPTION
A collection of objects for use with the Cisco Discovery Protocol.
::= { ciscoCdpMIBGroups 5 }

ciscoCdpV2MIBGroup  OBJECT-GROUP
OBJECTS { cdpCacheApplianceID, cdpCacheVlanID, cdpCachePowerConsumption, cdpCacheMTU, cdpCacheSysName, cdpCacheSysObjectID, cdpCacheLastChange, cdpCachePhysLocation, cdpCachePrimaryMgmtAddrType, cdpCachePrimaryMgmtAddr, cdpCacheSecondaryMgmtAddrType, cdpCacheSecondaryMgmtAddr, cdpGlobalLastChange, cdpGlobalDeviceIdFormatCpb, cdpGlobalDeviceIdFormat }

STATUS  current
DESCRIPTION
A collection of objects for use with the Cisco Discovery Protocol version 2.
::= { ciscoCdpMIBGroups 6 }

ciscoCdpV2IfExtGroup  OBJECT-GROUP
OBJECTS { cdpInterfaceExtendedTrust, cdpInterfaceCosForUntrustedPort }
Chapter 7  Cisco Management Information Base

CISCO-CDP-MIB

STATUS  current
DESCRIPTION
A collection of objects for use with the Cisco Discovery Protocol version 2 to configure the value for Extended Trust TLV and COS for Untrusted Port TLV.
::= { ciscoCdpMIBGroups 7 }

Troubleshooting

For Linux and Cisco Unified CM Release 5.x, 6.x, 7.x., collect the following logs and information for analysis:

• Use the set trace enable Detailed cdpmb CL set the detailed trace for cdpAgt ()
• Restart the Cisco CDP Agent service from the serviceability Web Page (Tools-> Controlcenter-Network Services) and wait for some time.
• Collect the following trace files:
  – Enable the Cisco CDP Agent traces by using the file get activelog cm/trace/cdpmb/sdi command and Cisco CDP daemon traces using the file get activelog cm/trace/cdp/sdi command.
  – Enable the Cisco CDP Agent and daemon traces by using the Real-Time Monitoring Tool (RTMT) > Trace & Log Central > Collect Files > Cisco CallManager SNMP Service > Cisco CDP Agent and Cisco CDP.
• Once the logs are collected, reset the trace setting by using the set trace disable cdpmb command.

For Windows and Cisco Unified CM Release 4.x, perform the following to collect logs for analysis.

• Set TraceEnabled to true under the registry HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\SnmpCDPAgent and restart SNMP service.
• After restarting SNMP service, another option TraceLevel displays. Set this to value 3.
• Restart SNMP service again.
• Do the walk on CDP MIB.
• Collect the log file from location C:\Program Files\Cisco\bin\SnmpCDPImpl.log.
• Collect the output of c:\utils\tlist.exe snmp.exe and output of dir c:\program files\cisco\bin.

Frequently Asked Questions

The CDP interface table and globalinfo tables are blank.
Verify that you Cisco Unified CM release that you are using has this capability. If not, upgrade.

How is the MessageInterval value set in the Interface table as well as Global table in CDP MIB?
Check to see if the HoldTime value is greater than MessageInterval value. If it is less, then the MessageInterval value can not be set from both Interface table as well as Global table.

This MIB provides a means to gather syslog messages generated by the Cisco IOS. Various textual messages are generated by the Cisco IOS. Cisco IOS can be configured such that these messages are sent to a syslog server. With this MIB these same messages can also be received via the SNMP. These messages are hereupon referred to as syslog messages in this document.

Messages generated as a result of entering CLI debug commands are not made available via the SNMP at this time.

All Cisco IOS syslog messages have timestamps (optional), facility names (where the message came from), severity, message name, and message text. The following example is often seen:

```
%SYS-5-CONFIG_I: configured from console where facility=SYS, severity=5, message name=CONFIG_I.
```

Before you can compile CISCO-SYSLOG-MIB, you need to compile the MIBs listed below in the order listed.

1. SNMPv2-SMI
2. SNMPv2-TC
3. SNMPv2-CONF
4. CISCO-SMI
5. INET-ADDRESS-MIB
6. SNMP-FRAMEWORK-MIB
7. RFC1155-SMI
8. RFC-1212
9. RFC-1215
10. SNMPv2-TC-v1
11. CISCO-SYSLOG-MIB

Additional downloads are:

- OID File: CISCO-SYSLOG-MIB.oid
- Capability File: CISCO-SYSLOG-CAPABILITY

The following are contained in this section:

- Revisions, page 7-145
- Definitions, page 7-145
- Object Identifiers, page 7-145
- Textual Conventions, page 7-145
Revisions

Table 7-7 lists the revisions to the MIB beginning with the latest revision.

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>08-07-1995</td>
<td>Initial Version</td>
<td>The MIB module describes how to store the system messages generated by the Cisco IOS software. ::= { ciscoMgmt 41 }</td>
</tr>
</tbody>
</table>

Definitions

The following definitions are imported for CISCO-SYSLOG-MIB:

- MODULE-IDENTITY, NOTIFICATION-TYPE, OBJECT-TYPE, Integer32, Counter32
- From SNMPv2-SMI—TEXTUAL-CONVENTION, DisplayString, TimeStamp, TruthValue
- From SNMPv2-TC—MODULE-COMPLIANCE, OBJECT-GROUP
- From SNMPv2-CONF—ciscoMgmt
- From CISCO-SMI—ciscoSyslogMIB MODULE-IDENTITY

`ciscoSyslogMIBObjects` OBJECT IDENTIFIER ::= { ciscoSyslogMIB 1 }

Object Identifiers

clogBasicOBJECT IDENTIFIER ::= { ciscoSyslogMIBObjects 1 }
clogHistoryOBJECT IDENTIFIER ::= { ciscoSyslogMIBObjects 2 }

Textual Conventions

`SyslogSeverity ::= TEXTUAL-CONVENTION`

- STATUS current
- DESCRIPTION
  The severity of a syslog message. The enumeration values are equal to the values that syslog uses + 1. For example, with syslog, emergency=0.
  SYNTAX INTEGER { emergency(1), alert(2), critical(3), error(4), warning(5), notice(6), info(7), debug(8) }
Basic Syslog Objects

**clogNotificationsSent OBJECT-TYPE**

SYNTAX Counter32
UNITS notifications
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of clogMessageGenerated notifications that have been sent. This number may include notifications that were prevented from being transmitted due to reasons such as resource limitations and/or non-connectivity. If one is receiving notifications, one can periodically poll this object to determine if any notifications were missed. If so, a poll of the clogHistoryTable might be appropriate.

::= { clogBasic 1 }

**clogNotificationsEnabled OBJECT-TYPE**

SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Indicates whether clogMessageGenerated notifications will or will not be sent when a syslog message is generated by the device. Disabling notifications does not prevent syslog messages from being added to the clogHistoryTable.

DEFVAL { false }

::= { clogBasic 2 }

**clogMaxSeverity OBJECT-TYPE**

SYNTAX SyslogSeverity
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Indicates which syslog severity levels will be processed. Any syslog message with a severity value greater than this value will be ignored by the agent.

Note: Severity numeric values increase as their severity decreases, e.g. error(4) is more severe than debug(8).

DEFVAL { warning }

::= { clogBasic 3 }

**clogMsgIgnores OBJECT-TYPE**

SYNTAX Counter32
UNITS messages
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of syslog messages which were ignored. A message will be ignored if it has a severity value greater than clogMaxSeverity.
::= { clogBasic 4 }

clogMsgDrops OBJECT-TYPE
SYNTAX Counter32
UNITS messages
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of syslog messages which could not be processed due to lack of system resources. Most likely this will occur at the same time that syslog messages are generated to indicate this lack of resources. Increases in this object's value may serve as an indication that system resource levels should be examined via other mib objects. A message that is dropped will not appear in the history table and no notification will be sent for this message.
::= { clogBasic 5 }

Syslog Message History Table

clogHistTableMaxLength OBJECT-TYPE
SYNTAX Integer32 (0..500)
UNITS entries
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The upper limit on the number of entries that the clogHistoryTable may contain. A value of zero prevents any history from being retained. When this table is full, the oldest entry will be deleted and a new one will be created.
DEFVAL { 1 }
::= { clogHistory 1 }

clogHistMsgsFlushed OBJECT-TYPE
SYNTAX Counter32
UNITS messages
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of entries that have been removed from the clogHistoryTable in order to make room for new entries. This object can be utilized to determine whether your polling frequency on the history table is fast enough and/or the size of your history table is large enough such that you are not missing messages.
::= { clogHistory 2 }

clogHistoryTable OBJECT-TYPE
SYNTAX SEQUENCE OF ClogHistoryEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A table of syslog messages generated by this device. All 'interesting' syslog messages (i.e. severity <= clogMaxSeverity) are entered into this table.
::= { clogHistory 3 }

clogHistoryEntry OBJECT-TYPE
SYNTAX ClogHistoryEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A syslog message that was previously generated by this device. Each entry is indexed by a message index.
INDEX { clogHistIndex }
::= { clogHistoryTable 1 }

ClogHistoryEntry ::= SEQUENCE { clogHistIndex Integer32, clogHistFacility DisplayString, clogHistSeverity SyslogSeverity, clogHistMsgName DisplayString, clogHistMsgText DisplayString, clogHistTimestamp TimeStamp }

clogHistIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A monotonically increasing integer for the sole purpose of indexing messages. When it reaches the maximum value the agent flushes the table and wraps the value back to 1.
::= { clogHistoryEntry 1 }

clogHistFacility OBJECT-TYPE
SYNTAX DisplayString (SIZE (1..20))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
Name of the facility that generated this message. For example: 'SYS'.
::= { clogHistoryEntry 2 }

clogHistSeverity OBJECT-TYPE
SYNTAX SyslogSeverity
MAX-ACCESS read-only
STATUS current
DESCRIPTION The severity of the message.
::= { clogHistoryEntry 3 }

clogHistMsgName OBJECT-TYPE
SYNTAX DisplayString (SIZE (1..30))
MAX-ACCESS read-only
STATUS current
DESCRIPTION A textual identification for the message type. A facility name in conjunction with a message name uniquely identifies a message type.
::= { clogHistoryEntry 4 }

clogHistMsgText OBJECT-TYPE
SYNTAX DisplayString (SIZE (1..255))
MAX-ACCESS read-only
STATUS current
DESCRIPTION The text of the message. If the text of the message exceeds 255 bytes, the message will be truncated to 254 bytes and a '*' character will be appended indicating that the message has been truncated.
::= { clogHistoryEntry 5 }

clogHistTimestamp OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION The value of sysUpTime when this message was generated.
::= { clogHistoryEntry 6 }

Notifications
ciscoSyslogMIBNotificationPrefix OBJECT IDENTIFIER ::= { ciscoSyslogMIB 2 }
ciscoSyslogMIBNotifications OBJECT IDENTIFIER ::= { ciscoSyslogMIBNotificationPrefix 0 }
clogMessageGenerated NOTIFICATION-TYPE
OBJECTS {clogHistFacility, clogHistSeverity, clogHistMsgName, clogHistMsgText, clogHistTimestamp }
STATUS current
DESCRIPTION When a syslog message is generated by the device a clogMessageGenerated notification is sent. The sending of these notifications can be enabled/disabled via the clogNotificationsEnabled object.
Conformance Information

```ciscoSyslogMIBConformance OBJECT IDENTIFIER ::= { ciscoSyslogMIB 3 }
ciscoSyslogMIBCompliances OBJECT IDENTIFIER ::= { ciscoSyslogMIBConformance 1 }
ciscoSyslogMIBGroups OBJECT IDENTIFIER ::= { ciscoSyslogMIBConformance 2 }
```

Compliance Statements

```ciscoSyslogMIBCompliance MODULE-COMPLIANCE
 STATUS current
 DESCRIPTION
 The compliance statement for entities which implement the Cisco syslog MIB.
 MANDATORY-GROUPS { ciscoSyslogMIBGroup }
 ::= { ciscoSyslogMIBCompliances 1 }
```

Units of Conformance

```ciscoSyslogMIBGroup OBJECT-GROUP
 OBJECTS { clogNotificationsSent, clogNotificationsEnabled, clogMaxSeverity, clogMsgIgnores, clogMsgDrops, clogHistTableMaxLength, clogHistMsgsFlushed, clogHistFacility, clogHistSeverity, clogHistMsgName, clogHistMsgText, clogHistTimestamp }
 STATUS current
 DESCRIPTION
 A collection of objects providing the syslog MIB capability.
 ::= { ciscoSyslogMIBGroups 1 }
```

Troubleshooting

Syslog has standard buffer size while generating a SNMP trap message; the data is trimmed to the specified field size (255). This avoids any errors caused by data that is too large for the field. For example, if you have specified the message text field to be 255 bytes, but a message arrives that is 300 bytes, the data will be truncated to 255 bytes before being logged.

Trap Configuration

To configure the traps, set clogsNotificationEnabled (1.3.6.1.4.1.9.9.41.1.1.2) to TRUE(1) by using SNMP set operation in any SNMP management application. Set the severity using clogMaxSeverity (1.3.6.1.4.1.9.9.41.1.1.3) by using any SNMP management application. This object indicates the syslog severity level that needs to be processed. Any syslog message with a severity value greater than this value will be ignored by the agent. Severity numeric values increase as their severity decreases.
Collect the following logs and information:

- Set the detailed trace for CiscoSyslogAgent with the `set trace enable Detailed syslogmib` command.
- Restart the Cisco Syslog Agent service from the serviceability Web window Tools > Control Center - Network Services and wait for some time.
- Collect the Cisco Syslog Agent trace files by:
  - Using the `file get activelog cm/trace/syslogmib/sdi/` command.
  - Using RTMT Trace & Log Central > Collect Files > Cisco CallManager SNMP Service > Cisco Syslog Agent.
- Once the logs are collected, reset the trace settings by using the `set trace disable syslogmib` command.

**Frequently Asked Questions**

How is a remote syslog server configured? You can configure a remote syslog server from Cisco Unified Communications Manager Administration > System > Enterprise Parameters > plus the following:

- **Remote Syslog Server Name**—You can enter the name or IP address of the remote Syslog server that you want to use to accept Syslog messages. If the server name is not specified, Cisco Unified Serviceability does not send the Syslog messages. Do not specify a Cisco Unified Communications Manager server as the destination because the Cisco Unified Communications Manager server does not accept Syslog messages from another server.
  - Maximum length: 255
  - Allowed values: Provide a valid remote syslog server name that comprises (A-Z,a-z,0-9,.,-)

- **Syslog Severity For Remote Syslog messages**—You can select the desired Syslog messages severity for remote syslog server. The system sends all the syslog messages with selected or higher severity levels to the remote syslog. If the remote server name is not specified, Cisco Unified Serviceability does not send the Syslog messages.

How is a remote syslog server configured to redirect alarms specific to a particular service? You can configure a remote syslog server from Cisco Unified Serviceability window > Alarm > Configuration:

- Select the Service Group and Service from drop down list for the particular server.
- Enable Alarm for Remote Syslogs and set the desired Alarm Event Level. Enter the remote syslog server name or IP address for redirection.
- The system sends all the syslog messages for the particular service with selected or higher severity levels to the remote syslog.

How are messages captured in the configured remote server? Kiwi Syslog Daemon is a freeware tool which can be installed in the remote server to capture the syslog messages.

What happens if the same remote server is configured from Enterprise Parameters and Alarm Configuration page?

- Enterprise parameters configuration of remote syslog redirects all the syslog messages which have severity equal to or higher than configured severity. There is no classification done for different types of syslog messages. It is just a plain redirection of all the syslog messages generated.
- Alarm configuration sends the specific service syslog messages to the configured remote server based on the severity.
Enterprise Parameters configuration is used by the Cisco Syslog Agent to send the messages. Corresponding application Alarm configuration will use the alarm interface to send to remote syslog server configured.

If the "Local Syslogs" Alarm is enabled in Alarm page, there will be duplication of the service specific messages, incase the same remote server is configured in both pages (provided the severity conditions are matched). For example: Enterprise window has severity level as "Error", Alarm page has severity "Debug" and "Local syslogs" alarm is enabled. If a syslog message of a particular service configured via alarm page, has a severity higher than 'Debug' and 'Error', then it will be duplicated.

Does the SysLog subagent generate traps for the alarms in Syslog automatically? Is there any configuration? Syslog subagent can be configured to generate traps for the syslog alarms. Some limitations are:

- Traps are sent out based on selected severity. If the given alarm is of low severity then the management application needs to set the severity threshold lower to capture this low severity alarm/trap. In other words mgmt apps need to deal with flooding of other low severity traps.
- SNMP Trap message size limited to 255 and not enabled by default. i.e. by default clogsNotificationEnabled (1.3.6.1.4.1.9.9.41.1.1.2) is set to FALSE (2).

**CISCO-SYSLOG-EXT-MIB**

Note


Before you can compile CISCO-SYSLOG-EXT-MIB, you need to download and compile the MIBs listed below in the order listed.

1. SNMPv2-SMI
2. SNMPv2-TC
3. SNMPv2-CONF
4. CISCO-SMI
5. INET-ADDRESS-MIB
6. SNMP-FRAMEWORK-MIB
7. CISCO-SYSLOG-MIB
8. RFC1155-SMI
9. RFC-1212
10. SNMPv2-TC-v1
11. CISCO-SYSLOG-EXT-MIB

Additional downloads are:
- OID File: CISCO-SYSLOG-EXT-MIB.oid
- Capability File: CISCO-SYSLOG-EXT-CAPABILITY

The following are contained in this section:
Revisions

Table 7-8 lists the revisions to the MIB beginning with the latest revision.

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>12/15/2003</td>
<td>Added</td>
<td>New enumerations. MIB module for configuring and monitoring System Log related management parameters as defined by RFC 3164.</td>
</tr>
<tr>
<td>11/13/2002</td>
<td>Added</td>
<td>cseSyslogServerFacility to cseSyslogServerTable. Added two TCs SyslogFacility and SyslogExFacility.</td>
</tr>
<tr>
<td>10/04/2002</td>
<td>Initial Version</td>
<td>:= { ciscoMgmt 301 }</td>
</tr>
</tbody>
</table>

Definitions

The following definitions are imported for CISCO-SYSLOG-EXT-MIB

- From MODULE-IDENTITY, OBJECT-TYPE, Unsigned32
- From SNMPv2-SMI—MODULE-COMPLIANCE, OBJECT-GROUP
- From SNMPv2-CONF—TruthValue, RowStatus, TEXTUAL-CONVENTION
- From SNMPv2-TC—snmpAdminString
- From SNMP-FRAMEWORK-MIB—inetAddressType, InetAddress
- From INET-ADDRESS-MIB—ciscoMgmt
- From CISCO-SMI—syslogSeverity
- From CISCO-SYSLOG-MIB

ciscoSyslogExtMIBObjects OBJECT IDENTIFIER ::= { ciscoSyslogExtMIB 1 }
cseSyslogConfigurationGroup OBJECT IDENTIFIER ::= { ciscoSyslogExtMIBObjects 1 }

Textual Conventions

SyslogFacility ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
The Syslog standard facilities.

REFERENCE
  – RFC 3014—The BSD Syslog protocol, Section 4.

SYNTAX
  INTEGER { kernel (0), -- Kernel user (8), -- User Level mail (16), -- Mail System
decision(24), -- System Daemon auth (32), -- Security/Authorization syslog (40), -- Internal Syslog
lpr (48), -- Line Printer subsystem news (56), -- Network New subsystem uucp (64), -- UUCP
subsystem cron (72), -- Clock Daemon authPriv (80), -- Security/Auth(private) ftp (88), -- FTP
Daemon local0 (128), -- Reserved local use loca1 (136), -- Reserved local use loca2 (144),
-- Reserved local use loca3 (152), -- Reserved local use loca4 (160), -- Reserved local use
local5 (168), -- Reserved local use loca6 (176), -- Reserved local use loca7 (184)-- Reserved
local use

SyslogExFacility ::= TEXTUAL-CONVENTION

STATUS  current

DESCRIPTION
The Syslog facilities including both standard and proprietary facilities.

REFERENCE
  – RFC 3014—The BSD Syslog protocol, Section 4.

SYNTAX
  INTEGER { kernel (0), -- Kernel user (8), -- User Level mail (16), -- Mail System
decision(24), -- System Daemon auth (32), -- Security/Authorization syslog (40), -- Internal Syslog
lpr (48), -- Line Printer subsystem news (56), -- Network New subsystem uucp (64), -- UUCP
subsystem cron (72), -- Clock Daemon authPriv (80), -- Security/Auth(private) ftp (88), -- FTP
Daemon local0 (128), -- Reserved local use loca1 (136), -- Reserved local use loca2 (144),
-- Reserved local use loca3 (152), -- Reserved local use loca4 (160), -- Reserved local use
local5 (168), -- Reserved local use loca6 (176), -- Reserved local use loca7 (184), -- Reserved
local use vsanMgr (200), -- VSAN Manager fspf (208), -- FSPF domainMgr (216), --
Domain Manager mtsDaemon (224), -- MTS Daemon linecardMgr (232), -- Line Card Mgr
sysMgr (240), -- System Manager sysMgrLib (248), -- System Mgr Library zoneServer (256), --
Zone Server virtualFaultMgr (264), -- VirtualInterfaceMgr ipConfMgr (272), -- IP Config Manager
ipC (280), -- IP Over FC xBarMgr (288), -- Xbar Manager fcDns (296), -- Fibre Channel DNS
fabricConfMgr (304), -- Fabric Config Server aclMgr (312), -- AccessConfigList Mgr tPortMgr
(320), -- TL Port Manager portMgr (328), -- Port Manager pfortServer (336), -- FPort Server
portChMgr (344), -- Port Channel Mgr mpls (352), -- MPLS tftpLib (360), -- TFTP Library
wwnMgr (368), -- WWN Mgr fcc (376), -- FCC Process qosMgr (384), -- QOS Mgr vhba (392), --
VHBA procMgr (400), -- Proc Mgr vddMgr (408), -- VELOD Mgr span (416), -- SPAVrrpMgr
(424), -- VRRP Mgr fcf (432), -- FCFWD ntp (440), -- NTP plmfmMgr (448), -- Platform
Mg xbarClient (456), -- XBAR Client vrrpEngine (464), -- VRRP Engine callhome (472), --
Callhome ipsMgr (480), -- IPS Mgr fc2 (488), -- FC2 debugLib (496), -- Debug Library vpm (504),
-- VPM mcast (512), -- Multicast rdl (520), -- RDL scn (536), -- RSCN bootvar (552), -- BootVar
pss (576), -- Persistent Storage -- System snmp (584), -- SNMP security (592), -- Security vhbad
(608), -- VHBA dns (648), -- DNS rib (656), -- RIB vsph (672), -- VSH Daemon fypd (688), --
Fabric Virtual Port -- Daemon mplsTunnel (816), -- MPLS Tunnel cdp (848), -- CDP Daemon
ohms (920), -- OHMs Daemon portSec (960), -- Port Security Manager ethPortMgr (976), --
Ethernet Port Manager ipaclMgr (1016), -- IP ACL Manager ficonMgr (1064), -- FICON
Manager ficonContDev (1096), -- Ficon Control Device rir (1128), -- RLIR Module fdmi (1136),--
Fabric Device -- Management Interface licMgr (1152), -- License Manager fcsMgr (1160), --
FCSP Manager confCheck (1192), -- Configuration Check ivr (1232), -- Inter-VSAN Routing aad
Syslog Configuration Group

This group provides the System log (Syslog) configuration options.

cseSyslogConsoleEnable OBJECT-TYPE

SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Indicate whether the Syslog messages should be sent to the console.
DEFVAL { false }
::= { cseSyslogConfigurationGroup 1 }
cseSyslogConsoleMsgSeverity OBJECT-TYPE

SYNTAX SyslogSeverity
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Minimum severity of the message that are sent to the Console.
DEFVAL { debug }
::= { cseSyslogConfigurationGroup 2 }
cseSyslogLogFileName OBJECT-TYPE

SYNTAX SnmpAdminString (SIZE (0..255))
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Name of file to which the Syslog messages are logged. Set operation with a zero length will fail.
DEFVAL { "messages" }
::= { cseSyslogConfigurationGroup 3 }
cseSyslogLogFileMsgSeverity OBJECT-TYPE

SYNTAX SyslogSeverity
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Minimum severity of the message that are sent to the log file (cseSyslogLogFileName).
DEFVAL { debug }

(1240),-- AAA Daemon tacacsd (1248), -- TACACS Daemon radiusd (1256), -- Radius Daemon fc2d (1320),-- FC2 Daemon lcOhmsd (1336), -- LC Ohms Daemon ficonStat (1352), -- FICON Statistics, featureMgr (1360), -- Feature Manager lttd (1376) -- LTT Daemon)
cseSyslogFileLoggingDisable OBJECT-TYPE
SYNTAX Integer { true (1), noOp (2) }
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Indicates whether the Syslog messages should be sent to the file indicated by cseSyslogLogFileName. Once this object is set to 'true', the Syslog messages are no longer sent to the file. The value of 'cseSyslogLogFileName' is set to zero length string. To restart the file logging, the cseSyslogLogFileName should be set to a valid file name.
No action is taken if this object is set to 'noOp'. The value of the object when read is always 'noOp'.

::= { cseSyslogConfigurationGroup 4 }

::= { cseSyslogConfigurationGroup 5 }

cseSyslogServerTableMaxEntries OBJECT-TYPE
SYNTAX Unsigned32 (0..65535)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The maximum number of entries that the agent supports in the cseSyslogServerTable.

::= { cseSyslogConfigurationGroup 6 }

cseSyslogServerTable

cseSyslogServerTable OBJECT-TYPE
SYNTAX Sequence of CseSyslogServerEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
This table contains all the Syslog servers which are configured.

::= { cseSyslogConfigurationGroup 7 }

cseSyslogServerEntry

CseSyslogServerEntry OBJECT-TYPE
SYNTAX CseSyslogServerEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry containing information about a Syslog server.
INDEX { cseSyslogServerIndex}

::= { cseSyslogServerTable 1 }
CseSyslogServerEntry ::=
SEQUENCE { cseSyslogServerIndex Unsigned32, cseSyslogServerAddressType InetAddressType, cseSyslogServerAddress InetAddress, cseSyslogServerMsgSeverity SyslogSeverity, cseSyslogServerStatus RowStatus, cseSyslogServerFacility SyslogFacility }

**cseSyslogServerIndex OBJECT-TYPE**
SYNTAX Unsigned32 (1..65535)
MAX-ACCESS not-accessible
DESCRIPTION
An arbitrary integer value, greater than zero, and less than and equal to cseSyslogServerTableMaxEntries, which identifies a Syslog server row in this table.
::= { cseSyslogServerEntry 1 }

**cseSyslogServerAddressType OBJECT-TYPE**
SYNTAX InetAddressType
MAX-ACCESS read-create
STATUS current
DESCRIPTION
The type of the address of the Syslog server which is given by the corresponding value of cseSyslogServerAddress.
::= { cseSyslogServerEntry 2 }

**cseSyslogServerAddress OBJECT-TYPE**
SYNTAX InetAddress
MAX-ACCESS read-create
STATUS current
DESCRIPTION
The address of the Syslog server.
::= { cseSyslogServerEntry 3 }

**cseSyslogServerMsgSeverity OBJECT-TYPE**
SYNTAX SyslogSeverity
MAX-ACCESS read-create
STATUS current
DESCRIPTION
Minimum severity of the message that are sent to this Syslog server.
DEFVAL {debug}
::= { cseSyslogServerEntry 4 }

**cseSyslogServerStatus OBJECT-TYPE**
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION
The status of this row. A row can not become 'active' until the values for cseSyslogServerAddressType and cseSyslogServerAddress in that row have both been set. A row cannot be created until corresponding instances of following objects are instantiated.

- cseSyslogServerAddressType
- cseSyslogServerAddress

The following objects may not be modified while the value of this object is active (1):

- cseSyslogServerAddressType
- cseSyslogServerAddress.

::= { cseSyslogServerEntry 5 }

cseSyslogServerFacility OBJECT-TYPE
SYNTAX SyslogFacility
MAX-ACCESS read-create
STATUS current
DESCRIPTION
The facility to be used when sending Syslog messages to this server.
DEFVAL {local7}
::= { cseSyslogServerEntry 6 }

cseSyslogMessageControlTable

cseSyslogMessageControlTable OBJECT-TYPE
SYNTAX Sequence of CseSyslogMessageControlEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
This table contains the information about what system log messages should be sent to Syslog host, console, log file, and/or logged into the internal buffer.
::= { cseSyslogConfigurationGroup 8 }

cseSyslogMessageControlEntry OBJECT-TYPE
SYNTAX cseSyslogMessageControlEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A system log message control table entry. Each entry specifies a severity for a particular 'facility' which generates Syslog messages. Any generated message which is at least as severe as the specified severity will be logged.
INDEX { cseSyslogMessageFacility }
::= { cseSyslogMessageControlTable 1 }
CseSyslogMessageControlEntry ::=
SEQUENCE { cseSyslogMessageFacility SyslogExFacility, cseSyslogMessageSeverity SyslogSeverity }

cseSyslogMessageFacility OBJECT-TYPE
SYNTAX SyslogExFacility
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
System log message facility.
::= { cseSyslogMessageControlEntry 1 }

cseSyslogMessageSeverity OBJECT-TYPE
SYNTAX SyslogSeverity
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Minimum severity of the message that are generated by this Syslog message facility.
::= { cseSyslogMessageControlEntry 2 }

cseSyslogTerminalEnable OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Indicate whether the Syslog messages should be sent to the terminals.
DEFVAL { false }
::= { cseSyslogConfigurationGroup 9 }

cseSyslogTerminalMsgSeverity OBJECT-TYPE
SYNTAX SyslogSeverity
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Minimum severity of the message that are sent to the terminals.
DEFVAL { debug }
::= { cseSyslogConfigurationGroup 10 }

cseSyslogLinecardEnable OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Indicate whether the Syslog messages should be generated at the line cards.
cseSyslogLinecardMsgSeverity OBJECT-TYPE
SYNTAX SyslogSeverity
MAX-ACCESS read-write
STATUS current
DESCRIPTION
Minimum severity of the message that are sent from linecards.
DEFVAL { debug }
 ::= { cseSyslogConfigurationGroup 12 }

Conformance

ciscoSyslogExtMIBConformance OBJECT IDENTIFIER ::= { ciscoSyslogExtMIB 2 }
ciscoSyslogExtMIBCompliances OBJECT IDENTIFIER ::= { ciscoSyslogExtMIBConformance 1 }
ciscoSyslogExtMIBGroups OBJECT IDENTIFIER ::= { ciscoSyslogExtMIBConformance 2 }
ciscoSyslogExtMIBCompliance MODULE-COMPLIANCE
STATUS current
DESCRIPTION
The compliance statement for entities which implement the CISCO-SYSLOG-EXT-MIB.
MODULE MANDATORY-GROUPS { ciscoSyslogExtGroup }
OBJECT cseSyslogServerAddressType
SYNTAX Integer { ipv4 (1), dns (16) }
DESCRIPTION
Only dns and ipv4 addresses are need to be supported.
OBJECT cseSyslogServerStatus
SYNTAX Integer { active (1), createAndGo (4), destroy (6) }
DESCRIPTION
Only three values 'createAndGo', 'destroy' and 'active' need to be supported.
OBJECT cseSyslogLinecardEnable
MIN-ACCESS read-only
DESCRIPTION
Write access is not required.
OBJECT cseSyslogLinecardMsgSeverity
MIN-ACCESS read-only
DESCRIPTION
Write access is not required.
OBJECT cseSyslogMessageFacility
SYNTAX     SyslogFacility
DESCRIPTION
Only the standard facilities need to be supported.
 ::= { ciscoSyslogExtMIBCompliances 1 }

Units of Conformance

ciscoSyslogExtGroup  OBJECT-GROUP
  OBJECTS  { cseSyslogConsoleEnable, cseSyslogLogFileName, cseSyslogFileLoggingDisable,
  cseSyslogConsoleMsgSeverity, cseSyslogLogFileMsgSeverity, cseSyslogServerTableMaxEntries,
  cseSyslogServerAddress, cseSyslogServerAddressType, cseSyslogServerMsgSeverity,
  cseSyslogServerStatus, cseSyslogServerFacility, cseSyslogMessageSeverity,
  cseSyslogTerminalEnable, cseSyslogTerminalMsgSeverity, cseSyslogLinecardEnable,
  cseSyslogLinecardMsgSeverity }
  STATUS     current
  DESCRIPTION
  A collection of objects for Syslog management.
  ::= { ciscoSyslogExtMIBGroups 1 }
Industry-Standard Management Information Base

This chapter describes the industry-standard Management Information Base (MIB) text files that are supported by Cisco Unified Communications Manager (Cisco Unified CM) and used with Simple Network Management Protocol (SNMP). It contains the following sections:

- **SYSAPPL-MIB**, page 8-1
- **RFC1213-MIB (MIB-II)**, page 8-28
- **HOST-RESOURCES-MIB**, page 8-73
- **IF-MIB**, page 8-105

**SYSAPPL-MIB**


The MIB module defines management objects that model applications as collections of executables and files installed and executing on a host system. The MIB presents a system-level view of applications; i.e., objects in this MIB are limited to those attributes that can typically be obtained from the system itself without adding special instrumentation to the applications.

Before you can compile SYSAPPL-MIB, you need to compile the MIBs listed below in the order listed.

1. RFC1155-SMI
2. RFC-1212
3. SNMPv2-SMI-v1
4. SNMPv2-TC-v1
5. SYSAPPL-MIB

Additional downloads are:

- OID File: SYSAPPL-MIB.oid
The following are contained in this section:

- Revisions, page 8-2
- Definitions, page 8-2
- System Application MIB, page 8-2
- Textual Conventions, page 8-3
- Installed Application Groups, page 8-3
- Additional Scalar Objects that Control Table Sizes, page 8-21
- Conformance Macros, page 8-25
- Troubleshooting, page 8-26

Revisions

Table 8-1 lists the revisions to the MIS beginning with the latest revision.

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10-20-1997</td>
<td>IETF Applications MIB Working Group</td>
<td>::= { mib-2 54 }</td>
</tr>
</tbody>
</table>

Definitions

The following definitions are imported for SYSAPP-MIB:

- MODULE-IDENTITY, OBJECT-TYPE, mib-2, Unsigned32 (gotten from CISCO-TC for the time being until it becomes available in SNMPv2-SMI), Unsigned32, TimeTicks, Counter32, Gauge32 TimeTicks, Counter32, Gauge32
- From SNMPv2-SMI—Unsigned32
- From CISCO-TC—DateAndTime, TEXTUAL-CONVENTION
- From SNMPv2-TC—MODULE-COMPLIANCE, OBJECT-GROUP
- From SNMPv2-CONF;

System Application MIB

sysApplMIB MODULE-IDENTITY
sysApplOBJ OBJECT IDENTIFIER ::= { sysApplMIB 1 }
sysApplInstalled OBJECT IDENTIFIER ::= { sysApplOBJ 1 }
sysApplRun OBJECT IDENTIFIER ::= { sysApplOBJ 2 }
sysApplMap OBJECT IDENTIFIER ::= { sysApplOBJ 3 }
sysApplNotifications OBJECT IDENTIFIER ::= { sysApplMIB 2 }
sysApplConformance OBJECT IDENTIFIER ::= { sysApplMIB 3 }
Textual Conventions

**RunState :::= TEXTUAL-CONVENTION**

**STATUS** current

**DESCRIPTION**

This TC describes the current execution state of a running application or process. The possible values are: running(1), runnable(2), waiting for a resource (CPU, etc.) waiting(3), waiting for an event exiting(4), other(5) other invalid state.

**SYNTAX** INTEGER { running (1); runnable (2); waiting for resource and waiting (3); waiting for event and exiting (4); other (5) }

**LongUtf8String :::= TEXTUAL-CONVENTION**

**DISPLAY-HINT** 1024a

**STATUS** current

**DESCRIPTION**

To facilitate internationalization, this TC represents information taken from the ISO/IEC IS 10646-1 character set, encoded as an octet string using the UTF-8 character encoding scheme described in RFC 2044 [10]. For strings in 7-bit US-ASCII, there is no impact since the UTF-8 representation is identical to the US-ASCII encoding.

**SYNTAX** OCTET STRING (SIZE (0..1024))

**Utf8String :::= TEXTUAL-CONVENTION**

**DISPLAY-HINT** 255a

**STATUS** current

**DESCRIPTION**

To facilitate internationalization, this TC represents information taken from the ISO/IEC IS 10646-1 character set, encoded as an octet string using the UTF-8 character encoding scheme described in RFC 2044 [10]. For strings in 7-bit US-ASCII, there is no impact since the UTF-8 representation is identical to the US-ASCII encoding.

**SYNTAX** OCTET STRING (SIZE (0..255))

Installed Application Groups

This group provides information about application packages that have been installed on the host computer. The group contains two tables as follows:

- *sysApplInstallPkgTable*: Describes the application packages
- *sysApplInstallElmtTable*: Describes the constituent elements (files and executables) which compose an application package

In order to appear in the group, an application and its component files must be discoverable by the system itself, possibly through some type of software installation mechanism or registry.
sysAppInstallPkgTable

The system installed application packages table provides information on the software packages installed on a system. These packages may consist of many different files including executable and non-executable files.

sysAppInstallPkgTable OBJECT-TYPE
SYNTAX SysAppInstallPkgEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table listing the software application packages installed on a host computer. In order to appear in this table, it may be necessary for the application to be installed using some type of software installation mechanism or global registry so that its existence can be detected by the agent implementation.

::= { sysApplInstalled 1 }

sysAppInstallPkgEntry OBJECT-TYPE
SYNTAX SysAppInstallPkgEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The logical row describing an installed application package.

INDEX { sysApplInstallPkgIndex }

::= { sysAppInstallPkgTable 1 }

SysAppInstallPkgEntry ::= SEQUENCE { sysApplInstallPkgIndex Unsigned32, sysAppInstallPkgManufacturer Utf8String, sysAppInstallPkgProductName Utf8String, sysAppInstallPkgVersion Utf8String, sysAppInstallPkgSerialNumber Utf8String, sysAppInstallPkgDate DateAndTime, sysAppInstallPkgLocation LongUtf8String }

sysAppInstallPkgIndex OBJECT-TYPE
SYNTAX Unsigned32 (1..ffffffffh)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An integer used only for indexing purposes. Generally monotonically increasing from 1 as new applications are installed. The value for each installed application must remain constant at least from one re-initialization of the network management entity which implements this MIB module to the next re-initialization. The specific value is meaningful only within a given SNMP entity. A sysAppInstallPkgIndex value must not be re-used until the next agent entity restart in the event the installed application entry is deleted.

::= { sysAppInstallPkgEntry 1 }

sysAppInstallPkgManufacturer OBJECT-TYPE
SYNTAX Utf8String
MAX-ACCESS read-only
Chapter 8  Industry-Standard Management Information Base

SYSAPPL-MIB

STATUS current
DESCRIPTION
The Manufacturer of the software application package.
::= { sysApplInstallPkgEntry 2 }

sysApplInstallPkgProductName OBJECT-TYPE
SYNTAX Utf8String
MAX-ACCESS  read-only
STATUS current
DESCRIPTION
The name assigned to the software application package by the Manufacturer.
::= { sysApplInstallPkgEntry 3 }

sysApplInstallPkgVersion OBJECT-TYPE
SYNTAX Utf8String
MAX-ACCESS  read-only
STATUS current
DESCRIPTION
The version number assigned to the application package by the manufacturer of the software.
::= { sysApplInstallPkgEntry 4 }

sysApplInstallPkgSerialNumber OBJECT-TYPE
SYNTAX Utf8String
MAX-ACCESS  read-only
STATUS current
DESCRIPTION
The serial number of the software assigned by the manufacturer.
::= { sysApplInstallPkgEntry 5 }

sysApplInstallPkgDate OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS  read-only
STATUS current
DESCRIPTION
The date and time this software application was installed on the host.
::= { sysApplInstallPkgEntry 6 }

sysApplInstallPkgLocation OBJECT-TYPE
SYNTAX LongUtf8String
MAX-ACCESS  read-only
STATUS current
DESCRIPTION
The complete path name where the application package is installed. For example, the value would be /opt/MyapplDir if the application package was installed in the /opt/MyapplDir directory.

\[::= \{ \text{sysApplInstallPkgEntry 7} \}\]

**sysApplInstallElmtTable**

This table details the individual application package elements (files and executables) installed on the host computer which comprise the applications defined in the `sysApplInstallPkg` Table. Each entry in this table has an index to the `sysApplInstallPkg` table to identify the application package of which it is a part. As a result, there may be many entries in this table for each instance in the `sysApplInstallPkg` Table.

Table entries are indexed by `sysApplInstallPkgIndex`, `sysApplInstallElmtIndex` to facilitate retrieval of all elements associated with a particular installed application package.

**sysApplInstallElmtTable OBJECT-TYPE**

SYNTAX SEQUENCE OF SysApplInstallElmtEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

This table details the individual application package elements (files and executables) installed on the host computer which comprise the applications defined in the `sysApplInstallPkg` Table. Each entry in this table has an index to the `sysApplInstallPkg` table to identify the application package of which it is a part. As a result, there may be many entries in this table for each instance in the `sysApplInstallPkg` Table.

Table entries are indexed by `sysApplInstallPkgIndex`, `sysApplInstallElmtIndex` to facilitate retrieval of all elements associated with a particular installed application package.

\[::= \{ \text{sysApplInstalled 2} \}\]

**sysApplInstallElmtEntry OBJECT-TYPE**

SYNTAX SysApplInstallElmtEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

The logical row describing an element of an installed application. The element may be an executable or non-executable file.

INDEX \{sysApplInstallPkgIndex, sysApplInstallElmtIndex\}

\[::= \{ \text{sysApplInstallElmtTable 1} \}\]

SysApplInstallElmtEntry ::= SEQUENCE { sysApplInstallElmtIndex INTEGER, sysApplInstallElmtNameUtf8String, sysApplInstallElmtTypeINTEGER, sysApplInstallElmtDateDateAndTime, sysApplInstallElmtPathLongUtf8String, sysApplInstallElmtSizeHighUnsigned32, sysApplInstallElmtSizeLowUnsigned32, sysApplInstallElmtRoleBITS, sysApplInstallElmtRoleOCTET STRING, sysApplInstallElmtModifyDateDateAndTime, sysApplInstallElmtCurSizeHighUnsigned32, sysApplInstallElmtCurSizeLowUnsigned32 }

**sysApplInstallElmtIndex OBJECT-TYPE**

SYNTAX Unsigned32 (1...fffffffh)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An arbitrary integer used for indexing. The value of this index is unique among all rows in this table that exist or have existed since the last agent restart.
::= { sysApplInstallElmtEntry 1 }

sysApplInstallElmtName OBJECT-TYPE
SYNTAX Utf8String
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The name of this element which is contained in the application.
::= { sysApplInstallElmtEntry 2 }

sysApplInstallElmtType OBJECT-TYPE
SYNTAX INTEGER { unknown(1), nonexecutable(2), operatingSystem(3), executable deviceDriver(4), executable application(5), executable }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The type of element that is part of the installed application.
::= { sysApplInstallElmtEntry 3 }

sysApplInstallElmtDate OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The date and time that this component was installed on the system.
::= { sysApplInstallElmtEntry 4 }

sysApplInstallElmtPath OBJECT-TYPE
SYNTAX LongUtf8String
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The full directory path where this element is installed. For example, the value would be /opt/EMPuma/bin for an element installed in the directory /opt/EMPuma/bin. Most application packages include information about the elements contained in the package. In addition, elements are typically installed in sub-directories under the package installation directory. In cases where the element path names are not included in the package information itself, the path can usually be
determined by a simple search of the sub-directories. If the element is not installed in that location and there is no other information available to the agent implementation, then the path is unknown and null is returned.

::= { sysApplInstallElmtEntry 5 }

**sysApplInstallElmtSizeHigh** OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

The installed file size in 2^32 byte blocks. This is the size of the file on disk immediately after installation. For example, for a file with a total size of 4,294,967,296 bytes, this variable would have a value of 1; for a file with a total size of 4,294,967,295 bytes this variable would be 0.

::= { sysApplInstallElmtEntry 6 }

**sysApplInstallElmtSizeLow** OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

The installed file size modulo 2^32 bytes. This is the size of the file on disk immediately after installation. For example, for a file with a total size of 4,294,967,296 bytes this variable would have a value of 0; for a file with a total size of 4,294,967,295 bytes this variable would be 4,294,967,295.

::= { sysApplInstallElmtEntry 7 }

**sysApplInstallElmtRole** OBJECT-TYPE

SYNTAX OCTET STRING (SIZE(1))

SYNTAX BITS { executable (0), exclusive (1), primary (2), required (3), dependent (4), unknown(5) }

MAX-ACCESS read-write

STATUS current

DESCRIPTION

An operator assigned value used in the determination of application status. This value is used by the agent to determine both the mapping of started processes to the initiation of an application, as well as to allow for a determination of application health. The default value, unknown(5), is used when an operator has not yet assigned one of the other values. If unknown(5) is set, bits 1 - 4 have no meaning. The possible values are:

- executable (0)—An application may have one or more executable elements. The rest of the bits have no meaning if the element is not executable.
- exclusive(1)—Only one copy of an exclusive element may be running per invocation of the running application.
- primary(2)—The primary executable. An application can have one, and only one element that is designated as the primary executable. The execution of this element constitutes an invocation of the application. This is used by the agent implementation to determine the initiation of an application. The primary executable must remain running long enough for the agent implementation to detect its presence.
– required(3)—An application may have zero or more required elements. All required elements must be running in order for the application to be judged to be running and healthy.
– dependent(4)—An application may have zero or more dependent elements. Dependent elements may not be running unless required elements are.
– unknown(5)—Default value for the case when an operator has not yet assigned one of the other values. When set, bits 1, 2, 3, and 4 have no meaning.

sysApplInstallElmtRole is used by the agent implementation in determining the initiation of an application, the current state of a running application (see sysApplRunCurrentState), when an application invocation is no longer running, and the exit status of a terminated application invocation (see sysApplPastRunExitState).

```plaintext
--DEFVAL { 5 }
::= { sysApplInstallElmtEntry 8 }
```

**sysApplInstallElmtModifyDate OBJECT-TYPE**

SYNTAX DateAndTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

The date and time that this element was last modified. Modification of the sysApplInstallElmtRole columnar object does NOT constitute a modification of the element itself and should not affect the value of this object.

```plaintext
::= { sysApplInstallElmtEntry 9 }
```

**sysApplInstallElmtCurSizeHigh OBJECT-TYPE**

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

The current file size in \(2^{32}\) byte blocks. For example, for a file with a total size of 4,294,967,296 bytes, this variable would have a value of 1; for a file with a total size of 4,294,967,295 bytes this variable would be 0.

```plaintext
::= { sysApplInstallElmtEntry 10 }
```

**sysApplInstallElmtCurSizeLow OBJECT-TYPE**

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

The current file size modulo \(2^{32}\) bytes. For example, for a file with a total size of 4,294,967,296 bytes this variable would have a value of 0; for a file with a total size of 4,294,967,295 bytes this variable would be 4,294,967,295.

```plaintext
::= { sysApplInstallElmtEntry 11 }
```
sysApplRun Group

This group models activity information for applications that have been invoked and are either currently running, or have previously run on the host system. Likewise, the individual elements of an invoked application are also modeled to show currently running processes, and processes that have run in the past.

sysApplRunTable

The sysApplRunTable contains the application instances which are currently running on the host. Since a single application might be invoked multiple times, an entry is added to this table for each invocation of an application. The table is indexed by sysApplInstallPkgIndex, sysApplRunIndex to enable managers to easily locate all invocations of a particular application package.

sysApplRunTable OBJECT-TYPE
SYNTAX SEQUENCE OF SysApplRunEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The table describes the applications which are executing on the host. Each time an application is invoked, an entry is created in this table. When an application ends, the entry is removed from this table and a corresponding entry is created in the SysApplPastRunTable.

A new entry is created in this table whenever the agent implementation detects a new running process that is an installed application element whose sysApplInstallElmtRole designates it as being the application's primary executable (sysApplInstallElmtRole = primary(2)).

The table is indexed by sysApplInstallPkgIndex, sysApplRunIndex to enable managers to easily locate all invocations of a particular application package.

 ::= { sysApplRun 1 }

sysApplRunEntry OBJECT-TYPE
SYNTAX SysApplRunEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The logical row describing an application which is currently running on this host.

INDEX { sysApplInstallPkgIndex, sysApplRunIndex }
 ::= { sysApplRunTable 1 }
SysApplRunEntry ::= SEQUENCE { sysApplRunIndex Unsigned32, sysApplRunStarted DateAndTime, sysApplRunCurrentState RunState }

sysApplRunIndex OBJECT-TYPE
SYNTAX Unsigned32 (1..fffffff'h)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
Part of the index for this table. An arbitrary integer used only for indexing purposes. Generally monotonically increasing from 1 as new applications are started on the host, it uniquely identifies application invocations.

The numbering for this index increases by 1 for each INVOCATION of an application, regardless of which installed application package this entry represents a running instance of. An example of the indexing for a couple of entries is shown below.

sysApplRunStarted.17.14
sysApplRunStarted.17.63
sysApplRunStarted.18.13

: 

In this example, the agent has observed 12 application invocations when the application represented by entry 18 in the sysApplInstallPkgTable is invoked. The next invocation detected by the agent is an invocation of installed application package 17. Some time later, installed application 17 is invoked a second time.

---

**Note**

This index is not intended to reflect a real-time (wall clock time) ordering of application invocations; it is merely intended to uniquely identify running instances of applications. Although the sysApplInstallPkgIndex is included in the INDEX clause for this table, it serves only to ease searching of this table by installed application and does not contribute to uniquely identifying table entries.

::= { sysApplRunEntry 1 }

**sysApplRunStarted** OBJECT-TYPE

SYNTAX DateAndTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

The date and time that the application was started.

::= { sysApplRunEntry 2 }

**sysApplRunCurrentState** OBJECT-TYPE

SYNTAX RunState

MAX-ACCESS read-only

STATUS current

DESCRIPTION

The current state of the running application instance. The possible values are running(1), runnable(2) but waiting for a resource such as CPU, waiting(3) for an event, exiting(4), or other(5). This value is based on an evaluation of the running elements of this application instance (see sysApplElmRunState) and their Roles as defined by sysApplInstallElmtRole. An agent implementation may detect that an application instance is in the process of exiting if one or more of its REQUIRED elements are no longer running. Most agent implementations will wait until a second internal poll has been completed to give the system time to start REQUIRED elements before marking the application instance as exiting.

::= { sysApplRunEntry 3 }
sysApplPastRunTable

The sysApplPastRunTable provides a history of applications previously run on the host computer. Entries are removed from the sysApplRunTable and corresponding entries are added to this table when an application becomes inactive. Entries remain in this table until they are aged out when either the table size reaches a maximum as determined by the sysApplPastRunMaxRows, or when an entry has aged to exceed a time limit as set by sysApplPastRunTblTimeLimit.

When aging out entries, the oldest entry, as determined by the value of sysApplPastRunTimeEnded, will be removed first.

sysApplPastRunTable OBJECT-TYPE
SYNTAX SEQUENCE OF SysApplPastRunEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A history of the applications that have previously run on the host computer. An entry's information is moved to this table from the sysApplRunTable when the invoked application represented by the entry ceases to be running. An agent implementation can determine that an application invocation is no longer running by evaluating the running elements of the application instance and their Roles as defined by sysApplInstallElmtRole. Obviously, if there are no running elements for the application instance, then the application invocation is no longer running.

If any one of the REQUIRED elements is not running, the application instance may be in the process of exiting. Most agent implementations will wait until a second internal poll has been completed to give the system time to either restart partial failures or to give all elements time to exit. If, after the second poll, there are REQUIRED elements that are not running, then the application instance may be considered by the agent implementation to no longer be running.

Entries remain in the sysApplPastRunTable until they are aged out when either the table size reaches a maximum as determined by the sysApplPastRunMaxRows, or when an entry has aged to exceed a time limit as set by sysApplPastRunTblTimeLimit.

Entries in this table are indexed by sysApplInstallPkgIndex, sysApplPastRunIndex to facilitate retrieval of all past run invocations of a particular installed application.

::= { sysApplRun 2 }

sysApplPastRunEntry OBJECT-TYPE
SYNTAX SysApplPastRunEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The logical row describing an invocation of an application which was previously run and has terminated. The entry is basically copied from the sysApplRunTable when the application instance terminates. Hence, the entry's value for sysApplPastRunIndex is the same as its value was for sysApplRunIndex.
INDEX { sysApplInstallPkgIndex, sysApplPastRunIndex }
::= { sysApplPastRunTable 1 }
SysApplPastRunEntry ::= SEQUENCE { sysApplPastRunIndex Unsigned32,
    sysApplPastRunStarted DateAndTime, sysApplPastRunExitState INTEGER,
    sysApplPastRunTimeEnded DateAndTime
}

sysApplPastRunIndex OBJECT-TYPE
SYNTAX  Unsigned32 (1...ffffffffff)
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
Part of the index for this table. An integer matching the value of the removed sysApplRunIndex
corresponding to this row.
::= { sysApplPastRunEntry 1 }

sysApplPastRunStarted OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The date and time that the application was started.
::= { sysApplPastRunEntry 2 }

sysApplPastRunExitState OBJECT-TYPE
SYNTAX  INTEGER { complete (1), failed (2), other (3) }
    – complete (1)—normal exit at sysApplRunTimeEnded
    – failed (2)—abnormal exit
    – other (3)
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The state of the application instance when it terminated. This value is based on an evaluation of the
running elements of an application and their Roles as defined by sysApplInstallElmtRole. An
application instance is said to have exited in a COMPLETE state and its entry is removed from the
sysApplRunTable and added to the sysApplPastRunTable when the agent detects that ALL elements
of an application invocation are no longer running. Most agent implementations will wait until a
second internal poll has been completed to give the system time to either restart partial failures or
to give all elements time to exit. A failed state occurs if, after the second poll, any elements continue
to run but one or more of the REQUIRED elements are no longer running.
All other combinations MUST be defined as OTHER.
::= { sysApplPastRunEntry 3 }

sysApplPastRunTimeEnded OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The DateAndTime the application instance was determined to be no longer running.
::= { sysApplPastRunEntry 4 }

sysApplElmtRunTable

The sysApplElmtRunTable contains an entry for each process that is currently running on the host. An entry is created in this table for each process at the time it is started, and will remain in the table until the process terminates. The table is indexed by sysApplElmtRunInstallPkg, sysApplElmtRunInvocID, and sysApplElmtRunIndex to make it easy to locate all running elements of a particular invoked application which has been installed on the system.

sysApplElmtRunTable OBJECT-TYPE
SYNTAX  SEQUENCE OF SysApplElmtRunEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
The table describes the processes which are currently executing on the host system. Each entry represents a running process and is associated with the invoked application of which that process is a part, if possible. This table contains an entry for every process currently running on the system, regardless of whether its 'parent' application can be determined. So, for example, processes like 'ps' and 'grep' will have entries though they are not associated with an installed application package.

Because a running application may involve more than one executable, it is possible to have multiple entries in this table for each application. Entries are removed from this table when the process terminates. The table is indexed by sysApplElmtRunInstallPkg, sysApplElmtRunInvocID, and sysApplElmtRunIndex to facilitate the retrieval of all running elements of a particular invoked application which has been installed on the system.
::= { sysApplRun 3 }

sysApplElmtRunEntry OBJECT-TYPE
SYNTAX  SysApplElmtRunEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
The logical row describing a process currently running on this host. When possible, the entry is associated with the invoked application of which it is a part.
INDEX{ sysApplElmtRunInstallPkg, sysApplElmtRunInvocID, sysApplElmtRunIndex }
::= { sysApplElmtRunTable  1 }

SysApplElmtRunEntry ::= SEQUENCE { sysApplElmtRunInstallPkg Unsigned32,
sysApplElmtRunInvocIDUnsigned32, sysApplElmtRunIndex  Unsigned32,
sysApplElmtRunInstallIID Unsigned32, sysApplElmtRunTimeStartedDateAndTime,
sysApplElmtRunState  RunState, sysApplElmtRunNameLongUtf8String,
sysApplElmtRunParameters Utf8String, sysApplElmtRunCPU TimeTicks,
sysApplElmtRunMemory Gauge32, sysApplElmtRunNumFiles  Gauge32,
sysApplElmtRunUserUtf8String }
sysApplElmtRunInstallPkg OBJECT-TYPE
SYNTAX  Unsigned32 (0...ffffffffh)
MAX-ACCESS not-accessible
STATUS  current
DESCRIPTION
Part of the index for this table, this value identifies the installed software package for the application of which this process is a part. Provided that the process's 'parent' application can be determined, the value of this object is the same value as the sysApplInstallPkgIndex for the entry in the sysApplInstallPkgTable that corresponds to the installed application of which this process is a part.
If, however, the 'parent' application cannot be determined, (for example the process is not part of a particular installed application), the value for this object is then '0', signifying that this process cannot be related back to an application, and in turn, an installed software package.
::= { sysApplElmtRunEntry 1 }

sysApplElmtRunInvocID OBJECT-TYPE
SYNTAX  Unsigned32 (0...ffffffffh)
MAX-ACCESS not-accessible
STATUS  current
DESCRIPTION
Part of the index for this table, this value identifies the invocation of an application of which this process is a part. Provided that the 'parent' application can be determined, the value of this object is the same value as the sysApplRunIndex for the corresponding application invocation in the sysApplRunTable.
If, however, the 'parent' application cannot be determined, the value for this object is then '0', signifying that this process cannot be related back to an invocation of an application in the sysApplRunTable.
::= { sysApplElmtRunEntry 2 }

sysApplElmtRunIndex OBJECT-TYPE
SYNTAX  Unsigned32 (0...ffffffffh)
MAX-ACCESS not-accessible
STATUS  current
DESCRIPTION
Part of the index for this table. A unique value for each process running on the host. Wherever possible, this should be the system's native, unique identification number.
::= { sysApplElmtRunEntry 3 }

sysApplElmtRunInstallID OBJECT-TYPE
SYNTAX  Unsigned32 (0...ffffffffh)
MAX-ACCESS read-only
STATUS  current
DESCRIPTION
The index into the sysAppInstallElmtTable. The value of this object is the same value as the sysAppInstallElmtIndex for the application element of which this entry represents a running instance.

If this process cannot be associated with an installed executable, the value should be '0'.

::= { sysAppElmtRunEntry 4 }

sysAppElmtRunTimeStarted OBJECT-TYPE
SYNTAX  DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The time the process was started.
::= { sysAppElmtRunEntry 5 }

sysAppElmtRunState OBJECT-TYPE
SYNTAX  RunState
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The current state of the running process. The possible values are running(1), runnable(2) but waiting for a resource such as CPU, waiting(3) for an event, exiting(4), or other(5).
::= { sysAppElmtRunEntry 6 }

sysAppElmtRunName OBJECT-TYPE
SYNTAX  LongUtf8String
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The full path and filename of the process. For example, /opt/MYYpkg/bin/myyproc would be returned for process myyproc whose execution path is /opt/MYYpkg/bin/myyproc.
::= { sysAppElmtRunEntry 7 }

sysAppElmtRunParameters OBJECT-TYPE
SYNTAX  Utf8String
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The starting parameters for the process.
::= { sysAppElmtRunEntry 8 }

sysAppElmtRunCPU OBJECT-TYPE
SYNTAX  TimeTicks
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of centi-seconds of the total system CPU resources consumed by this process. Note that on a multi-processor system, this value may have been incremented by more than one centi-second in one centi-second of real (wall clock) time.
 ::= { sysApplElmtRunEntry 9 }

sysApplElmtRunMemory OBJECT-TYPE
SYNTAX Gauge32
UNITS Kbytes
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total amount of real system memory measured in Kbytes currently allocated to this process.
 ::= { sysApplElmtRunEntry 10 }

sysApplElmtRunNumFiles OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of regular files currently open by the process. Transport connections (sockets) should NOT be included in the calculation of this value, nor should operating system specific special file types.
 ::= { sysApplElmtRunEntry 11 }

sysApplElmtRunUser OBJECT-TYPE
SYNTAX Utf8String
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The process owner's login name (e.g. root).
 ::= { sysApplElmtRunEntry 12 }

sysApplElmtPastRunTable
The sysApplElmtPastRunTable maintains a history of processes which have previously executed on the host as part of an application. Upon termination of a process, the entry representing the process is removed from the sysApplElmtRunTable and a corresponding entry is created in this table provided that the process was part of an identifiable application. If the process could not be associated with an invoked application, no corresponding entry is created.

Hence, whereas the sysApplElmtRunTable contains an entry for every process currently executing on the system, the sysApplElmtPastRunTable only contains entries for processes that previously executed as part of an invoked application.

Entries remain in this table until they are aged out when either the number of entries in the table reaches a
maximum as determined by `sysApplElmtPastRunMaxRows`, or when an entry has aged to exceed a time limit as set by `sysApplElmtPastRunTblTimeLimit`. When aging out entries, the oldest entry, as determined by the value of `sysApplElmtPastRunTimeEnded`, will be removed first.

The table is indexed by `sysApplInstallPkgIndex` (from the `sysApplInstallPkgTable`), `sysApplElmtPastRunInvocID`, and `sysApplElmtPastRunIndex` to make it easy to locate all previously executed processes of a particular invoked application that has been installed on the system.

### `sysApplElmtPastRunTable` OBJECT-TYPE

SYNTAX SEQUENCE OF `SysApplElmtPastRunEntry`

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

The table describes the processes which have previously executed on the host system as part of an application. Each entry represents a process which has previously executed and is associated with the invoked application of which it was a part. Because an invoked application may involve more than one executable, it is possible to have multiple entries in this table for each application invocation. Entries are added to this table when the corresponding process in the `sysApplElmtRunTable` terminates.

Entries remain in this table until they are aged out when either the number of entries in the table reaches a maximum as determined by `sysApplElmtPastRunMaxRows`, or when an entry has aged to exceed a time limit as set by `sysApplElmtPastRunTblTimeLimit`. When aging out entries, the oldest entry, as determined by the value of `sysApplElmtPastRunTimeEnded`, will be removed first.

The table is indexed by `sysApplInstallPkgIndex` (from the `sysApplInstallPkgTable`), `sysApplElmtPastRunInvocID`, and `sysApplElmtPastRunIndex` to make it easy to locate all previously executed processes of a particular invoked application that has been installed on the system.

::= { `sysApplRun` 4 }

### `sysApplElmtPastRunEntry` OBJECT-TYPE

SYNTAX `SysApplElmtPastRunEntry`

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

The logical row describing a process which was previously executed on this host as part of an installed application. The entry is basically copied from the `sysApplElmtRunTable` when the process terminates. Hence, the entry's value for `sysApplElmtPastRunIndex` is the same as its value was for `sysApplElmtRunIndex`. Note carefully: only those processes which could be associated with an identified application are included in this table.

INDEX `{ `sysApplInstallPkgIndex`, `sysApplElmtPastRunInvocID`, `sysApplElmtPastRunIndex` }

::= { `sysApplElmtPastRunTable` 1 }

`SysApplElmtPastRunEntry` ::= SEQUENCE { `sysApplElmtPastRunInvocIDUnsigned32`,
  `sysApplElmtPastRunIndex  Unsigned32`,
  `sysApplElmtPastRunInstallID  Unsigned32`,
  `sysApplElmtPastRunTimeStartedDateAndTime`,
  `sysApplElmtPastRunTimeEnded  DateAndTime`,
  `sysApplElmtPastRunNameLongUtf8String`,
  `sysApplElmtPastRunParameters Utf8String`,
  `sysApplElmtPastRunCPU TimeTicks`,
  `sysApplElmtPastRunMemory Unsigned32`,
  `sysApplElmtPastRunNumFiles  Unsigned32`,
  `sysApplElmtPastRunUserUtf8String` }

### `sysApplElmtPastRunInvocID` OBJECT-TYPE
SYNTAX  Unsigned32 (1...fffffff’h)
MAX-ACCESS  not-accessible
STATUS   current
DESCRIPTION
Part of the index for this table, this value identifies the invocation of an application of which the
process represented by this entry was a part. The value of this object is the same value as the
sysApplRunIndex for the corresponding application invocation in the sysApplRunTable. If the
invoked application as a whole has terminated, it will be the same as the sysApplPastRunIndex.
::= { sysApplElmtPastRunEntry 1 }

sysApplElmtPastRunIndex OBJECT-TYPE
   SYNTAX  Unsigned32 (0..fffffff’h)
   MAX-ACCESS  not-accessible
   STATUS   current
   DESCRIPTION
Part of the index for this table. An integer assigned by the agent equal to the corresponding
sysApplElmtRunIndex which was removed from the sysApplElmtRunTable and moved to this table
when the element terminated. Note that entries in this table are indexed by
sysApplElmtPastRunInvocID, sysApplElmtPastRunIndex.
The possibility exists, though unlikely, of a collision occurring by a new entry which was run by the
same invoked application (InvocID), and was assigned the same process identification number
(ElmtRunIndex) as an element which was previously run by the same invoked application.
Should this situation occur, the new entry replaces the old entry.
See the Implementation Issues section, sysApplElmtPastRunTable Entry Collisions for the
conditions that would have to occur in order for a collision to occur.
::= { sysApplElmtPastRunEntry 2 }

sysApplElmtPastRunInstallID OBJECT-TYPE
   SYNTAX  Unsigned32 (1..fffffff’h)
   MAX-ACCESS  read-only
   STATUS   current
   DESCRIPTION
The index into the installed element table. The value of this object is the same value as the
sysApplInstallElmtIndex for the application element of which this entry represents a previously
executed process.
::= { sysApplElmtPastRunEntry 3 }

sysApplElmtPastRunTimeStarted OBJECT-TYPE
   SYNTAX  DateAndTime
   MAX-ACCESS  read-only
   STATUS   current
   DESCRIPTION
The time the process was started.
::= { sysApplElmtPastRunEntry 4 }
SYSAPPL-MIB

sysApplElmtPastRunTimeEnded OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The time the process ended.
::= { sysApplElmtPastRunEntry 5 }

sysApplElmtPastRunName OBJECT-TYPE
SYNTAX LongUtf8String
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The full path and filename of the process. For example, '/opt/MYYpkg/bin/myyproc' would be returned for process 'myyproc' whose execution path was '/opt/MYYpkg/bin/myyproc'.
::= { sysApplElmtPastRunEntry 6 }

sysApplElmtPastRunParameters OBJECT-TYPE
SYNTAX Utf8String
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The starting parameters for the process.
::= { sysApplElmtPastRunEntry 7 }

sysApplElmtPastRunCPU OBJECT-TYPE
SYNTAX TimeTicks
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The last known number of centi-seconds of the total system's CPU resources consumed by this process. Note that on a multi-processor system, this value may increment by more than one centi-second in one centi-second of real (wall clock) time.
::= { sysApplElmtPastRunEntry 8 }

sysApplElmtPastRunMemory OBJECT-TYPE
SYNTAX Unsigned32 (0..ffffffff'h)
UNITS Kbytes
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The last known total amount of real system memory measured in Kbytes allocated to this process before it terminated.
::= { sysApplElmtPastRunEntry 9 }

sysApplElmtPastRunNumFiles OBJECT-TYPE
SYNTAX  Unsigned32 (0..'ffffffff'h)
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The last known number of files open by the process before it terminated. Transport connections (sockets) should NOT be included in the calculation of this value.
::= { sysApplElmtPastRunEntry 10 }

sysApplElmtPastRunUser OBJECT-TYPE
SYNTAX  Utf8String
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The process owner's login name (e.g. root).
::= { sysApplElmtPastRunEntry 11 }

Additional Scalar Objects that Control Table Sizes

sysApplPastRunMaxRows OBJECT-TYPE
SYNTAX  Unsigned32 (0..'ffffffff'h)
MAX-ACCESS  read-write
STATUS  current
DESCRIPTION
The maximum number of entries allowed in the sysApplPastRunTable. Once the number of rows in the sysApplPastRunTable reaches this value, the management subsystem will remove the oldest entry in the table to make room for the new entry to be added. Entries will be removed on the basis of oldest sysApplPastRunTimeEnded value first.
This object may be used to control the amount of system resources that can used for sysApplPastRunTable entries. A conforming implementation should attempt to support the default value, however, a lesser value may be necessary due to implementation-dependent issues and resource availability.
DEFVAL  { 500 }
::= { sysApplRun 5 }

sysApplPastRunTableRemItems OBJECT-TYPE
SYNTAX  Counter32
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
A counter of the number of entries removed from the sysApplPastRunTable because of table size limitations as set in sysApplPastRunMaxRows. This counter is the number of entries the management subsystem has had to remove in order to make room for new entries (so as not to exceed the limit set by sysApplPastRunMaxRows) since the last initialization of the management subsystem.

::= { sysApplRun 6 }

sysApplPastRunTblTimeLimit OBJECT-TYPE
SYNTAX  Unsigned32 (0.'ffffffff'h)
UNITS seconds
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The maximum time in seconds which an entry in the sysApplPastRunTable may exist before it is removed. Any entry that is older than this value will be removed (aged out) from the table. Note that an entry may be aged out prior to reaching this time limit if it is the oldest entry in the table and must be removed to make space for a new entry so as to not exceed sysApplPastRunMaxRows.

DEFVAL { 7200 }
::= { sysApplRun 7 }

sysApplElemPastRunMaxRows OBJECT-TYPE
SYNTAX  Unsigned32 (0.'ffffffff'h)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The maximum number of entries allowed in the sysApplElmtPastRunTable. Once the number of rows in the sysApplElmtPastRunTable reaches this value, the management subsystem will remove the oldest entry to make room for the new entry to be added. Entries will be removed on the basis of oldest sysApplElmtPastRunTimeEnded value first. This object may be used to control the amount of system resources that can used for sysApplElemPastRunTable entries. A conforming implementation should attempt to support the default value, however, a lesser value may be necessary due to implementation-dependent issues and resource availability.

DEFVAL { 500 }
::= { sysApplRun 8 }

sysApplElemPastRunTableRemItems OBJECT-TYPE
SYNTAX  Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A counter of the number of entries removed from the sysApplElemPastRunTable because of table size limitations as set in sysApplElemPastRunMaxRows. This counter is the number of entries the management subsystem has had to remove in order to make room for new entries (so as not to exceed the limit set by sysApplElemPastRunMaxRows) since the last initialization of the management subsystem.

::= { sysApplRun 9 }
sysApplElemPastRunTblTimeLimit OBJECT-TYPE
   SYNTAX  Unsigned32 (0..'ffffffff'h)
   UNITS seconds
   MAX-ACCESS  read-write
   STATUS  current
   DESCRIPTION
   The maximum time in seconds which an entry in the sysApplElemPastRunTable may exist before it
   is removed. Any entry that is older than this value will be removed (aged out) from the table. Note
   that an entry may be aged out prior to reaching this time limit if it is the oldest entry in the table and
   must be removed to make space for a new entry so as to not exceed sysApplElemPastRunMaxRows.
   DEFVAL  { 7200 }
   ::= { sysApplRun 10 }

sysApplAgentPollInterval  OBJECT-TYPE
   SYNTAX  Unsigned32 (0..'ffffffff'h)
   UNITS seconds
   MAX-ACCESS  read-write
   STATUS  current
   DESCRIPTION
   The minimum interval in seconds that the management subsystem implementing this MIB will poll
   the status of the managed resources. Because of the non-trivial effort involved in polling the
   managed resources, and because the method for obtaining the status of the managed resources is
   implementation-dependent, a conformant implementation may chose a lower bound greater than 0.
   A value of 0 indicates that there is no delay in the passing of information from the managed
   resources to the agent.
   DEFVAL  { 60 }
   ::= { sysApplRun 11 }

sysApplMap Group

This group contains a table, the sysApplMapTable, whose sole purpose is to provide a 'backwards'
mapping so that, given a known sysApplElmtRunIndex (process identification number), the
 corresponding invoked application (sysApplRunIndex), installed element (sysApplInstallElmtIndex),
 and installed application package (sysApplInstallPkgIndex) can be quickly determined. The table will
 contain one entry for each process currently running on the system.

A backwards mapping is extremely useful since the tables in this MIB module are typically indexed with
 the installed application package (sysApplInstallPkgIndex) as the primary key, and on down as required
 by the specific table, with the process ID number (sysApplElmtRunIndex) being the least significant
 key.

It is expected that management applications will use this mapping table by doing a 'getNext' operation
 with the known process ID number (sysApplElmtRunIndex) as the partial instance identifier. Assuming
 that there is an entry for the process, the result should return a single columnar value, the
 sysApplMapInstallPkgIndex, with the sysApplElmtRunIndex, sysApplRunIndex, and
 sysApplInstallElmtIndex contained in the instance identifier for the returned MIB object value.
Note
If the process can not be associated back to an invoked application installed on the system, then the value returned for the columnar value sysApplMapInstallPkgIndex will be '0' and the instance portion of the object-identifier will be the process ID number (sysApplElmtRunIndex) followed by 0.0.

sysApplMapTable OBJECT-TYPE
  SYNTAX  SEQUENCE OF SysApplMapEntry
  MAX-ACCESS not-accessible
  STATUS  current
  DESCRIPTION
  The sole purpose of this table is to provide a 'backwards' mapping so that, given a known sysApplElmtRunIndex (process identification number), the corresponding invoked application (sysApplRunIndex), installed element (sysApplElmtElmtIndex), and installed application package (sysApplElmtPkgIndex) can be quickly determined.
 ::= { sysApplMapTable 1 }

sysApplMapEntry OBJECT-TYPE
  SYNTAX  SysApplMapEntry
  MAX-ACCESS not-accessible
  STATUS  current
  DESCRIPTION
  A logical row representing a process currently running on the system. This entry provides the index mapping from process identifier, back to the invoked application, installed element, and finally, the installed application package. The entry includes only one accessible columnar object, the sysApplMapInstallElmtIndex, but the invoked application and installed element can be determined from the instance identifier since they form part of the index clause.
  INDEX  { sysApplElmtRunIndex, sysApplElmtRunInvocID, sysApplMapInstallElmtIndex }
 ::= { sysApplMapEntry 1 }

sysApplMapInstallElmtIndex OBJECT-TYPE
  SYNTAX  Unsigned32 (0..'ffffffff'h)
  MAX-ACCESS not-accessible
  STATUS  current
  DESCRIPTION
  The index into the sysApplInstallElmtTable. The value of this object is the same value as the sysApplInstallElmtIndex for the application element of which this entry represents a running instance. If this process cannot be associated to an installed executable, the value should be '0'.
 ::= { sysApplMapEntry 1 }

sysApplMapInstallPkgIndex OBJECT-TYPE
  SYNTAX  Unsigned32 (0..'ffffffff'h)
  MAX-ACCESS read-only
The value of this object identifies the installed software package for the application of which this process is a part. Provided that the process's 'parent' application can be determined, the value of this object is the same value as the sysAppInstallPkgIndex for the entry in the sysAppInstallPkgTable that corresponds to the installed application of which this process is a part.

If, however, the 'parent' application cannot be determined, (for example the process is not part of a particular installed application), the value for this object is then '0', signifying that this process cannot be related back to an application, and in turn, an installed software package.

::= { sysApplMapEntry 2 }
Chapter 8  Industry-Standard Management Information Base

SYSAPPL-MIB

sysApplPastRunTableRemItems, sysApplPastRunTblTimeLimit, sysApplElemPastRunMaxRows, sysApplElemPastRunTableRemItems, sysApplElemPastRunTblTimeLimit, sysApplAgentPollInterval }

STATUS  current

DESCRIPTION

The system application run group contains information about applications and associated elements which have run or are currently running on the host system.

::= { sysApplMIBGroups 2 }

sysApplMapGroup OBJECT-GROUP

OBJECTS { sysApplMapInstallPkgIndex }

STATUS  current

DESCRIPTION

The Map Group contains a single table, sysApplMapTable, that provides a backwards mapping for determining the invoked application, installed element, and installed application package given a known process identification number.

::= { sysApplMIBGroups 3 }

Troubleshooting

The following subsections have troubleshooting tips:

• Linux and Cisco Unified CM Releases 5.x, 6.x, 7.x, page 8-26
• Windows and Cisco Unified CM Release 4.x, page 8-26
• Using Servlets in Cisco Unified CM 7.x, page 8-27
• Frequently Asked Questions, page 8-28

Linux and Cisco Unified CM Releases 5.x, 6.x, 7.x

Collect the following logs and information for analysis. Execute the command file get activelog <paths below>:

• SNMP Master Agent Path : /platform/snmp/snmpdm/*
• System Application Agent Path: /platform/snmp/sappagt/*

Windows and Cisco Unified CM Release 4.x

Collect the following logs and information for analysis:

• Set the sysapp trace level to Detailed as follows, Enable TraceEnabled to "true" and TraceLevel to 3 from Registry HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\SnmpSysAppAgent.
• Once you have edited it, restart the SNMP Service from the Services tab. You will see a trace file C:\Program Files\Cisco\bin\SnmpSysAppImpl.log created.
• Run a snmpwalk on the sysApplInstallPkgTable.
• Run a snmpwalk on the SysApplRunTable.
Using Servlets in Cisco Unified CM 7.x

The SysAppl MIB provides a way to get inventory of what is installed and running at a given time. SysAppl agent cannot give the list of services activated or deactivated. It can only provide the running/not running states of the application/services. Web App services/Servlets cannot be monitored using the SysAppl MIB. Following are servlets for a 7.x system:

- Cisco CallManager Admin
- Cisco CallManager Cisco IP Phone Services
- Cisco CallManager Personal Directory
- Cisco CallManager Serviceability
- Cisco CallManager Serviceability RTMT
- Cisco Dialed Number Analyzer
- Cisco Extension Mobility
- Cisco Extension Mobility Application
- Cisco RTMT Reporter Servlet
- Cisco Tomcat Stats Servlet
- Cisco Trace Collection Servlet
- Cisco AXL Web Service
- Cisco Unified Mobile Voice Access Service
- Cisco Extension Mobility
- Cisco IP Manager Assistant
- Cisco WebDialer Web Service
- Cisco CAR Web Service
- Cisco Dialed Number Analyzer

For monitoring important service status for system health purposes, the following approaches are recommended:

- Use the Serviceability API called GetServiceStatus. This API can provide complete status information including activation status for both web application type and non web app services. (See AXL Serviceability API Guide for more details.)
- Use the utils service list command to check the status of different services.
- Use the Syslog message and monitor the servM generated messages. For example:

```
Mar 18 16:40:52 ciscart26 local7 6 : 92: Mar 18 11:10:52.630 UTC :
%CCM_SERVICEMANAGER-SERVICEMANAGER-6-ServiceActivated: Service Activated. Service Name:Cisco CallManager SNMP Service App ID:Cisco Service Manager Cluster ID: Node ID:ciscart26
```
Frequently Asked Questions

When the CCMVersion MIB and sysAppRunCurrentState returns incorrect values in Cisco Unified CM Release 4.x, refer to CSCsk74156 to check if it is being hit. Verify if the fix for the defect has gone into the Cisco Unified CM version used by customer.

When the SNMP walk on sysApp MIB is not responding, refer to CSCsh72473 to check if it is being hit. Verify if the fix for the defect has gone into the Cisco Unified CM version used by customer.

RFC1213-MIB (MIB-II)

Note

This is a reformatted version of MIB-II. Download and compile all of the MIBs in this section from http://tools.cisco.com/Support/SNMP/do/BrowseMIB.do?local=en&step=2.

Before you can compile RFC1213-MIB, you need to compile the MIBs listed below in the order listed.

1. SNMPv2-SMI
2. SNMPv2-TC
3. IANAifType-MIB
4. RFC1155-SMI
5. RFC-1212
6. RFC1213-MIB

The following are contained in this section:

- Revisions, page 8-29
- Definitions, page 8-29
- Object Identifiers, page 8-29
- Textual Conventions, page 8-29
- Groups in MIB-II, page 8-29
- Historical, page 8-30
- System Group, page 8-30
- Interfaces Group, page 8-32
- Address Translation Group, page 8-37
- IP Group, page 8-39
- ICMP Group, page 8-50
- TCP Group, page 8-55
- UDP Group, page 8-60
- EGP Group, page 8-62
- SNMP Group, page 8-67
Revisions

The following changes have been applied:

- The enumerations unknown(4) and dormant(5) have been added to ifOperStatus to reflect a change to the ifTable introduced in RFC 1573.
- The SYNTAX of ifType has been changed to IANAifType, to reflect the change to the ifTable introduced in RFC1573.

Definitions

The following definitions are imported for MIB-II:

- mgmt, NetworkAddress, IpAddress, Counter, Gauge, TimeTicks
- From RFC1155-SMI—OBJECT-TYPE
- From RFC-1212—TEXTUAL-CONVENTION
- From SNMPv2-TC—IANAifType
- From IANAifType-MIB;

Object Identifiers

This MIB module uses the extended OBJECT-TYPE macro as defined in [14]. MIB-II (same prefix as MIB-I) mib-2 OBJECT IDENTIFIER ::= { mgmt 1 }.

Textual Conventions

**DisplayString ::= OCTET STRING**

This data type is used to model textual information taken from the NVT ASCII character set. By convention, objects with this syntax are declared as having SIZE (0..255).

**PhysAddress ::= OCTET STRING**

This data type is used to model media addresses. For many types of media, this will be in a binary representation. For example, an ethernet address would be represented as a string of 6 octets.

Groups in MIB-II

- system OBJECT IDENTIFIER ::= { mib-2 1 }
- interfaces OBJECT IDENTIFIER ::= { mib-2 2 }
- at OBJECT IDENTIFIER ::= { mib-2 3 }
- ip OBJECT IDENTIFIER ::= { mib-2 4 }
- icmp OBJECT IDENTIFIER ::= { mib-2 5 }
- tcp OBJECT IDENTIFIER ::= { mib-2 6 }
- udp OBJECT IDENTIFIER ::= { mib-2 7 }
egp

OBJECT IDENTIFIER ::= { mib-2 8 }

Historical

cmot

OBJECT IDENTIFIER ::= { mib-2 9 }

transmission

OBJECT IDENTIFIER ::= { mib-2 10 }

snmp

OBJECT IDENTIFIER ::= { mib-2 11 }

System Group

Implementation of the system group is mandatory for all systems. If an agent is not configured to have a value for any of these variables, a string of length 0 is returned.

sysDescr OBJECT-TYPE
SYNTAX  DisplayString (SIZE (0..255))
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
A textual description of the entity. This value should include the full name and version identification of the system's hardware type, software operating-system, and networking software. It is mandatory that this only contain printable ASCII characters.

::= { system 1 }

sysObjectID OBJECT-TYPE
SYNTAX  Object Identifier
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The vendor authoritative identification of the network management subsystem contained in the entity. This value is allocated within the SMI enterprises subtree (1.3.6.1.4.1) and provides an easy and unambiguous means for determining “what kind of box” is being managed. For example, if vendor “Flintstones, Inc.” was assigned the subtree 1.3.6.1.4.1.4242, it could assign the identifier 1.3.6.1.4.1.4242.1.1 to its “Fred Router”.

::= { system 2 }

sysUpTime OBJECT-TYPE
SYNTAX  TimeTicks
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The time (in hundredths of a second) since the network management portion of the system was last re-initialized.

::= { system 3 }
sysContact OBJECT-TYPE
SYNTAX  DisplayString (SIZE (0..255))
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
The textual identification of the contact person for this managed node, together with information on
how to contact this person.
 ::= { system 4 }

sysName OBJECT-TYPE
SYNTAX  DisplayString (SIZE (0..255))
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
An administratively-assigned name for this managed node. By convention, this is the node's
fully-qualified domain name.
 ::= { system 5 }

sysLocation OBJECT-TYPE
SYNTAX  DisplayString (SIZE (0..255))
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
The physical location of this node (e.g., telephone closet, 3rd floor).
 ::= { system 6 }

sysServices OBJECT-TYPE
SYNTAX  Integer (0..127)
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
A value which indicates the set of services that this entity primarily offers. The value is a sum. This
sum initially takes the value zero, Then, for each layer, L, in the range 1 through 7, that this node
performs transactions for, 2 raised to (L - 1) is added to the sum. For example, a node which
performs primarily routing functions would have a value of 4 (2^(3-1)). In contrast, a node which
is a host offering application services would have a value of 72 (2^(4-1) + 2^(7-1)). Note that in the
context of the Internet suite of protocols, values should be calculated accordingly (layer first, then
functionality):
  1  physical (e.g., repeaters)
  2  datalink/subnetwork (e.g., bridges)
  3  internet (e.g., IP gateways)
  4  end-to-end (e.g., IP hosts)
  7  applications (e.g., mail relays)
For systems including OSI protocols, layers 5 and 6 may also be counted.
::= { system 7 }

Interfaces Group

Implementation of the Interfaces group is mandatory for all systems.

ifNumber OBJECT-TYPE
   SYNTAX  Integer
   ACCESS  read-only
   STATUS  mandatory
   DESCRIPTION
   The number of network interfaces (regardless of their current state) present on this system.
   ::= { interfaces 1 }

Interfaces Table

The interfaces table contains information on the entity interfaces. Each interface is thought of as being attached to a subnetwork. Note that this term should not be confused with subnet which refers to an addressing partitioning scheme used in the Internet suite of protocols.

ifTable OBJECT-TYPE
   SYNTAX  Sequence of ifEntry
   ACCESS  not-accessible
   STATUS  mandatory
   DESCRIPTION
   A list of interface entries. The number of entries is given by the value of ifNumber.
   ::= { interfaces 2 }

ifEntry OBJECT-TYPE
   SYNTAX  IfEntry
   ACCESS  not-accessible
   STATUS  mandatory
   DESCRIPTION
   An interface entry containing objects at the subnetwork layer and below for a particular interface.
   INDEX   { ifIndex }
   ::= { ifTable 1 }
   IfEntry ::= SEQUENCE { ifIndex INTEGER, ifDescr DisplayString, ifType IANAifType, ifMtu INTEGER, ifSpeed Gauge, ifPhysAddress PhysAddress, ifAdminStatus INTEGER, ifOperStatus INTEGER, ifLastChange TimeTicks, ifInOctets Counter, ifInUcastPkts Counter, ifInNUcastPkts Counter, ifInDiscards Counter, ifInErrors Counter, ifInUnknownProtos Counter, ifOutOctets Counter, ifOutUcastPkts Counter, ifOutNUcastPkts Counter, ifOutDiscards Counter, ifOutErrors Counter, ifOutQLen Gauge, ifSpecific OBJECT IDENTIFIER }
ifIndex OBJECT-TYPE
SYNTAX Integer
ACCESS read-only
STATUS mandatory
DESCRIPTION
A unique value for each interface. Its value ranges between 1 and the value of ifNumber. The value for each interface must remain constant at least from one re-initialization of the entity network management system to the next re-initialization.
 ::= { ifEntry 1 }

ifDescr OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..255))
ACCESS read-only
STATUS mandatory
DESCRIPTION
A textual string containing information about the interface. This string should include the name of the manufacturer, the product name and the version of the hardware interface.
 ::= { ifEntry 2 }

ifType OBJECT-TYPE
SYNTAX IANAifType
ACCESS read-only
STATUS mandatory
DESCRIPTION
The type of interface. Additional values for ifType are assigned by the Internet Assigned Numbers Authority (IANA), through updating the syntax of the IANAifType textual convention.
 ::= { ifEntry 3 }

ifMtu OBJECT-TYPE
SYNTAX Integer
ACCESS read-only
STATUS mandatory
DESCRIPTION
The size of the largest datagram which can be sent/received on the interface, specified in octets. For interfaces that are used for transmitting network datagrams, this is the size of the largest network datagram that can be sent on the interface.
 ::= { ifEntry 4 }

ifSpeed OBJECT-TYPE
SYNTAX Gauge
ACCESS read-only
STATUS mandatory
DESCRIPTION
An estimate of the interface current bandwidth in bits per second. For interfaces which do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth.

::= { ifEntry 5 }

ifPhysAddress OBJECT-TYPE
SYNTAX  PhysAddress
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The interface address at the protocol layer immediately below the network layer in the protocol stack. For interfaces which do not have such an address (e.g., a serial line), this object should contain an octet string of zero length.

::= { ifEntry 6 }

ifAdminStatus OBJECT-TYPE
SYNTAX  Integer { up(1), ready to pass packets down(2), testing(3) in some test mode }
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
The desired state of the interface. The testing(3) state indicates that no operational packets can be passed.

::= { ifEntry 7 }

ifOperStatus OBJECT-TYPE
SYNTAX  INTEGER { up(1),-- ready to pass packets down(2), testing(3),-- in some test mode unknown(4), dormant(5) }
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The current operational state of the interface. The testing(3) state indicates that no operational packets can be passed.

::= { ifEntry 8 }

ifLastChange OBJECT-TYPE
SYNTAX  TimeTicks
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The value of sysUpTime at the time the interface entered its current operational state. If the current state was entered prior to the last re-initialization of the local network management subsystem, then this object contains a zero value.

::= { ifEntry 9 }

ifInOctets OBJECT-TYPE
SYNTAX  Counter  
ACCESS  read-only  
STATUS  mandatory  
DESCRIPTION The total number of octets received on the interface, including framing characters.  
 ::= { ifEntry 10 }

ifInUcastPkts OBJECT-TYPE  
SYNTAX  Counter  
ACCESS  read-only  
STATUS  mandatory  
DESCRIPTION The number of subnetwork-unicast packets delivered to a higher-layer protocol.  
 ::= { ifEntry 11 }

ifInNUcastPkts OBJECT-TYPE  
SYNTAX  Counter  
ACCESS  read-only  
STATUS  mandatory  
DESCRIPTION The number of non-unicast (i.e., subnetwork-broadcast or subnetwork-multicast) packets delivered to a higher-layer protocol.  
 ::= { ifEntry 12 }

ifInDiscards OBJECT-TYPE  
SYNTAX  Counter  
ACCESS  read-only  
STATUS  mandatory  
DESCRIPTION The number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol. One possible reason for discarding such a packet could be to free up buffer space.  
 ::= { ifEntry 13 }

ifInErrors OBJECT-TYPE  
SYNTAX  Counter  
ACCESS  read-only  
STATUS  mandatory  
DESCRIPTION The number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol.  
 ::= { ifEntry 14 }

ifInUnknownProtos OBJECT-TYPE  
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of packets received via the interface which were discarded because of an unknown or unsupported protocol.
 ::= { ifEntry 15 }

ifOutOctets  OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of octets transmitted out of the interface, including framing characters.
 ::= { ifEntry 16 }

ifOutUcastPkts  OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of packets that higher-level protocols requested be transmitted to a subnetwork-unicast address, including those that were discarded or not sent.
 ::= { ifEntry 17 }

ifOutNUcastPkts  OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of packets that higher-level protocols requested be transmitted to a non-unicast (i.e., a subnetwork-broadcast or subnetwork-multicast) address, including those that were discarded or not sent.
 ::= { ifEntry 18 }

ifOutDiscards  OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.
 ::= { ifEntry 19 }
ifOutErrors OBJECT-TYPE
   SYNTAX  Counter
   ACCESS  read-only
   STATUS  mandatory
   DESCRIPTION
   The number of outbound packets that could not be transmitted because of errors.
   ::= { ifEntry 20 }

ifOutQLen OBJECT-TYPE
   SYNTAX  Gauge
   ACCESS  read-only
   STATUS  mandatory
   DESCRIPTION
   The length of the output packet queue (in packets).
   ::= { ifEntry 21 }

ifSpecific OBJECT-TYPE
   SYNTAX  OBJECT IDENTIFIER
   ACCESS  read-only
   STATUS  mandatory
   DESCRIPTION
   A reference to MIB definitions specific to the particular media being used to realize the interface. For example, if the interface is realized by a ethernet, then the value of this object refers to a document defining objects specific to ethernet. If this information is not present, its value should be set to the OBJECT IDENTIFIER { 0  0 }, which is a syntactically valid object identifier, and any conformant implementation of ASN.1 and BER must be able to generate and recognize this value.
   ::= { ifEntry 22 }

Address Translation Group

Implementation of the Address Translation group is mandatory for all systems. Note however that this group is deprecated by MIB-II. That is, it is being included solely for compatibility with MIB-I nodes, and will most likely be excluded from MIB-III nodes. From MIB-II and onwards, each network protocol group contains its own address translation tables. The Address Translation group contains one table which is the union across all interfaces of the translation tables for converting a NetworkAddress (e.g., an IP address) into a subnetwork-specific address. For lack of a better term, this document refers to such a subnetwork-specific address as a physical address.

Examples of such translation tables are: for broadcast media where ARP is in use, the translation table is equivalent to the ARP cache; or, on an X.25 network where non-algorithmic translation to X.121 addresses is required, the translation table contains the NetworkAddress to X.121 address equivalences.

atTable OBJECT-TYPE
   SYNTAX  Sequence of atEntry
   ACCESS  not-accessible
The Address Translation tables contain the NetworkAddress to physical address equivalences. Some interfaces do not use translation tables for determining address equivalences (e.g., DDN-X.25 has an algorithmic method); if all interfaces are of this type, then the Address Translation table is empty, i.e., has zero entries.

```plaintext
::= { at 1 }
```

**atEntry OBJECT-TYPE**

SYNTAX AtEntry

ACCESS not-accessible

STATUS deprecated

DESCRIPTION

Each entry contains one NetworkAddress to physical address equivalence.

INDEX  { atIfIndex, atNetAddress }

 ::= { atTable 1 }

AtEntry ::= SEQUENCE { atIfIndex INTEGER, atPhysAddress PhysAddress, atNetAddress NetworkAddress }

**atIfIndex OBJECT-TYPE**

SYNTAX Integer

ACCESS read-write

STATUS deprecated

DESCRIPTION

The interface on which this entry equivalence is effective. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex.

 ::= { atEntry 1 }

**atPhysAddress OBJECT-TYPE**

SYNTAX PhysAddress

ACCESS read-write

STATUS deprecated

DESCRIPTION

The media-dependent physical address. Setting this object to a null string (one of zero length) has the effect of invaliding the corresponding entry in the atTable object. That is, it effectively disassociates the interface identified with said entry from the mapping identified with said entry. It is an implementation-specific matter as to whether the agent removes an invalidated entry from the table. Accordingly, management stations must be prepared to receive tabular information from agents that corresponds to entries not currently in use.

Proper interpretation of such entries requires examination of the relevant atPhysAddress object.

 ::= { atEntry 2 }

**atNetAddress OBJECT-TYPE**

SYNTAX NetworkAddress
ACCESS read-write
STATUS deprecated
DESCRIPTION
The NetworkAddress (e.g., the IP address) corresponding to the media-dependent physical address.
::= { atEntry 3 }

**IP Group**

Implementation of the IP group is mandatory for all systems.

**ipForwarding OBJECT-TYPE**

SYNTAX INTEGER { forwarding(1), -- acting as a gateway not-forwarding(2) -- NOT acting as a gateway }
ACCESS read-write
STATUS mandatory
DESCRIPTION
The indication of whether this entity is acting as an IP gateway in respect to the forwarding of datagrams received by, but not addressed to, this entity. IP gateways forward datagrams. IP hosts do not (except those source-routed via the host). Note that for some managed nodes, this object may take on only a subset of the values possible. Accordingly, it is appropriate for an agent to return a badValue response if a management station attempts to change this object to an inappropriate value.
::= { ip 1 }

**ipDefaultTTL OBJECT-TYPE**

SYNTAX Integer
ACCESS read-write
STATUS mandatory
DESCRIPTION
The default value inserted into the Time-To-Live field of the IP header of datagrams originated at this entity, whenever a TTL value is not supplied by the transport layer protocol.
::= { ip 2 }

**ipInReceives OBJECT-TYPE**

SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of input datagrams received from interfaces, including those received in error.
::= { ip 3 }

**ipInHdrErrors OBJECT-TYPE**

SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of input datagrams discarded due to errors in their IP headers, including bad checksums, version number mismatch, other format errors, time-to-live exceeded, errors discovered in processing their IP options, etc.
::= { ip 4 }

ipInAddrErrors OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of input datagrams discarded because the IP address in their IP header's destination field was not a valid address to be received at this entity. This count includes invalid addresses (e.g., 0.0.0.0) and addresses of unsupported Classes (e.g., Class E). For entities which are not IP Gateways and therefore do not forward datagrams, this counter includes datagrams discarded because the destination address was not a local address.
::= { ip 5 }

ipForwDatagrams OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of input datagrams for which this entity was not their final IP destination, as a result of which an attempt was made to find a route to forward them to that final destination. In entities which do not act as IP Gateways, this counter will include only those packets which were Source-Routed via this entity, and the Source-Route option processing was successful.
::= { ip 6 }

ipInUnknownProtos OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of locally-addressed datagrams received successfully but discarded because of an unknown or unsupported protocol.
::= { ip 7 }

ipInDiscards OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of input IP datagrams for which no problems were encountered to prevent their continued processing, but which were discarded (e.g., for lack of buffer space). Note that this counter does not include any datagrams discarded while awaiting re-assembly.

::= { ip 8 }

**ipInDelivers** OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of input datagrams successfully delivered to IP user-protocols (including ICMP).

::= { ip 9 }

**ipOutRequests** OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of IP datagrams which local IP user-protocols (including ICMP) supplied to IP in requests for transmission. Note that this counter does not include any datagrams counted in ipForwDatagrams.

::= { ip 10 }

**ipOutDiscards** OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of output IP datagrams for which no problem was encountered to prevent their transmission to their destination, but which were discarded (e.g., for lack of buffer space). Note that this counter would include datagrams counted in ipForwDatagrams if any such packets met this (discretionary) discard criterion.

::= { ip 11 }

**ipOutNoRoutes** OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of IP datagrams discarded because no route could be found to transmit them to their destination. Note that this counter includes any packets counted in ipForwDatagrams which meet this no-route criterion. Note that this includes any datagrams which a host cannot route because all of its default gateways are down.

::= { ip 12 }
ipReasmTimeout OBJECT-TYPE
SYNTAX  Integer
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The maximum number of seconds which received fragments are held while they are awaiting reassembly at this entity.
::= { ip 13 }

ipReasmReqds OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of IP fragments received which needed to be reassembled at this entity.
::= { ip 14 }

ipReasmOKs OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of IP datagrams successfully re-assembled.
::= { ip 15 }

ipReasmFails OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of failures detected by the IP re-assembly algorithm (for whatever reason: timed out, errors, etc). Note that this is not necessarily a count of discarded IP fragments since some algorithms (notably the algorithm in RFC 815) can lose track of the number of fragments by combining them as they are received.
::= { ip 16 }

ipFragOKs OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of IP datagrams that have been successfully fragmented at this entity.
::= { ip 17 }
ipFragFails OBJECT-TYPE
   SYNTAX  Counter
   ACCESS  read-only
   STATUS  mandatory
   DESCRIPTION
   The number of IP datagrams that have been discarded because they needed to be fragmented at this
   entity but could not be, e.g., because their Don't Fragment flag was set.
   ::= { ip 18 }

ipFragCreates OBJECT-TYPE
   SYNTAX  Counter
   ACCESS  read-only
   STATUS  mandatory
   DESCRIPTION
   The number of IP datagram fragments that have been generated as a result of fragmentation at this
   entity.
   ::= { ip 19 }

IP Address Table

The IP address table contains this entity IP addressing information.

ipAddrTable OBJECT-TYPE
   SYNTAX  Sequence of ipAddrEntry
   ACCESS  not-accessible
   STATUS  mandatory
   DESCRIPTION
   The table of addressing information relevant to this entity IP addresses.
   ::= { ip 20 }

ipAddrEntry OBJECT-TYPE
   SYNTAX  IpAddrEntry
   ACCESS  not-accessible
   STATUS  mandatory
   DESCRIPTION
   The addressing information for one of this entity IP addresses.
   INDEX  { ipAdEntAddr }
   ::= { ipAddrTable 1 }
   IpAddrEntry ::= 
   SEQUENCE { ipAdEntAddr IpAddress, ipAdEntIfIndex INTEGER, ipAdEntNetMask IpAddress,
               ipAdEntBcastAddr INTEGER, ipAdEntReasmMaxSize INTEGER (0..65535) }

ipAdEntAddr OBJECT-TYPE
SYNTAX IpAddress
ACCESS read-only
STATUS mandatory
DESCRIPTION
The IP address to which this entry addressing information pertains.
::= { ipAddrEntry 1 }

ipAdEntIfIndex OBJECT-TYPE
SYNTAX Integer
ACCESS read-only
STATUS mandatory
DESCRIPTION
The index value which uniquely identifies the interface to which this entry is applicable. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex.
::= { ipAddrEntry 2 }

ipAdEntNetMask OBJECT-TYPE
SYNTAX IpAddress
ACCESS read-only
STATUS mandatory
DESCRIPTION
The subnet mask associated with the IP address of this entry. The value of the mask is an IP address with all the network bits set to 1 and all the hosts bits set to 0.
::= { ipAddrEntry 3 }

ipAdEntBcastAddr OBJECT-TYPE
SYNTAX Integer
ACCESS read-only
STATUS mandatory
DESCRIPTION
The value of the least-significant bit in the IP broadcast address used for sending datagrams on the (logical) interface associated with the IP address of this entry. For example, when the Internet standard all-ones broadcast address is used, the value will be 1. This value applies to both the subnet and network broadcasts addresses used by the entity on this (logical) interface.
::= { ipAddrEntry 4 }

ipAdEntReasmMaxSize OBJECT-TYPE
SYNTAX Integer (0..65535)
ACCESS read-only
STATUS mandatory
DESCRIPTION
The size of the largest IP datagram which this entity can re-assemble from incoming IP fragmented datagrams received on this interface.
::= { ipAddrEntry 5 }

IP Routing Table

-- The IP routing table contains an entry for each route
-- presently known to this entity.

ipRouteTable OBJECT-TYPE
SYNTAX  Sequence of ipRouteEntry
ACCESS  not-accessible
STATUS  mandatory
DESCRIPTION
This entity IP Routing table.
::= { ip 21 }

ipRouteEntry OBJECT-TYPE
SYNTAX  IpRouteEntry
ACCESS  not-accessible
STATUS  mandatory
DESCRIPTION
A route to a particular destination.
INDEX   { ipRouteDest }
::= { ipRouteTable 1 }
IpRouteEntry ::= 
SEQUENCE { ipRouteDest IpAddress, ipRouteIfIndex INTEGER, ipRouteMetric1 INTEGER,
ipRouteMetric2 INTEGER, ipRouteMetric3 INTEGER, ipRouteMetric4 INTEGER,
ipRouteNextHop IpAddress, ipRouteType INTEGER, ipRouteProto INTEGER, ipRouteAge
INTEGER, ipRouteMask IpAddress, ipRouteMetric5 INTEGER, ipRouteInfo OBJECT
IDENTIFIER }

ipRouteDest OBJECT-TYPE
SYNTAX  IpAddress
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
The destination IP address of this route. An entry with a value of 0.0.0.0 is considered a default
route. Multiple routes to a single destination can appear in the table, but access to such multiple
entries is dependent on the table-access mechanisms defined by the network management protocol
in use.
::= { ipRouteEntry 1 }

ipRouteIfIndex OBJECT-TYPE
SYNTAX  Integer
ACCESS  read-write
ipRouteMetric1 OBJECT-TYPE
SYNTAX  Integer
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
The primary routing metric for this route. The semantics of this metric are determined by the routing-protocol specified in the route ipRouteProto value. If this metric is not used, its value should be set to -1.
 ::= { ipRouteEntry 3 }

ipRouteMetric2 OBJECT-TYPE
SYNTAX  Integer
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
An alternate routing metric for this route. The semantics of this metric are determined by the routing-protocol specified in the route ipRouteProto value. If this metric is not used, its value should be set to -1.
 ::= { ipRouteEntry 4 }

ipRouteMetric3 OBJECT-TYPE
SYNTAX  Integer
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
An alternate routing metric for this route. The semantics of this metric are determined by the routing-protocol specified in the route ipRouteProto value. If this metric is not used, its value should be set to -1.
 ::= { ipRouteEntry 5 }

ipRouteMetric4 OBJECT-TYPE
SYNTAX  Integer
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
An alternate routing metric for this route. The semantics of this metric are determined by the routing-protocol specified in the route ipRouteProto value. If this metric is not used, its value should be set to -1.

::= { ipRouteEntry 6 }

ipRouteNextHop OBJECT-TYPE
SYNTAX  IpAddress
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
The IP address of the next hop of this route. (In the case of a route bound to an interface which is realized via a broadcast media, the value of this field is the agent's IP address on that interface.)

::= { ipRouteEntry 7 }

ipRouteType OBJECT-TYPE
SYNTAX  Integer { other(1),        -- none of the following invalid(2),      -- an invalidated route -- route to directly direct(3),       -- connected (sub-)network -- route to a non-local indirect(4)      -- host/network/sub-network }
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
The type of route. Note that the values direct(3) and indirect(4) refer to the notion of direct and indirect routing in the IP architecture. Setting this object to the value invalid(2) has the effect of invalidating the corresponding entry in the ipRouteTable object. That is, it effectively disassociates the destination identified with said entry from the route identified with said entry. It is an implementation-specific matter as to whether the agent removes an invalidated entry from the table. Accordingly, management stations must be prepared to receive tabular information from agents that corresponds to entries not currently in use. Proper interpretation of such entries requires examination of the relevant ipRouteType object.

::= { ipRouteEntry 8 }

ipRouteProto OBJECT-TYPE
SYNTAX  INTEGER { other(1),       -- none of the following -- non-protocol information, -- e.g., manually configured local(2),       -- entries -- set via a network netmgmt(3), -- management protocol -- obtained via ICMP, icmp(4),        -- e.g., Redirect -- the remaining values are -- all gateway routing -- protocols egp(5), ggp(6), hello(7), rip(8), is-is(9), es-is(10), ciscoIgrp(11), bbnSpfIgp(12), ospf(13), bgp(14) }
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The routing mechanism via which this route was learned. Inclusion of values for gateway routing protocols is not intended to imply that hosts should support those protocols.

::= { ipRouteEntry 9 }

ipRouteAge OBJECT-TYPE
SYNTAX  Integer
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
The number of seconds since this route was last updated or otherwise determined to be correct. Note
that no semantics of too old can be implied except through knowledge of the routing protocol by
which the route was learned.
::= { ipRouteEntry 10 }

ipRouteMask OBJECT-TYPE
SYNTAX  IpAddress
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
Indicate the mask to be logical-ANDed with the destination address before being compared to the
value in the ipRouteDest field. For those systems that do not support arbitrary subnet masks, an
agent constructs the value of the ipRouteMask by determining whether the value of the
correspondent ipRouteDest field belong to a class-A, B, or C network, and then using one of: mask
network 255.0.0.0 class-A, 255.255.0.0 class-B, 255.255.255.0  class-C. If the value of the
ipRouteDest is 0.0.0.0 (a default route), then the mask value is also 0.0.0.0. It should be noted that
all IP routing subsystems implicitly use this mechanism.
::= { ipRouteEntry 11 }

ipRouteMetric5 OBJECT-TYPE
SYNTAX  Integer
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
An alternate routing metric for this route. The semantics of this metric are determined by the
routing-protocol specified in the route ipRouteProto value. If this metric is not used, its value
should be set to -1.
::= { ipRouteEntry 12 }

ipRouteInfo OBJECT-TYPE
SYNTAX  OBJECT IDENTIFIER
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
A reference to MIB definitions specific to the particular routing protocol which is responsible for
this route, as determined by the value specified in the route ipRouteProto value. If this information
is not present, its value should be set to the OBJECT IDENTIFIER { 0 0 }, which is a syntactically
valid object identifier, and any conformant implementation of ASN.1 and BER must be able to
generate and recognize this value.
::= { ipRouteEntry 13 }
IP Address Translation Table

The IP address translation table contain the IP Address to physical address equivalences. Some interfaces do not use translation tables for determining address equivalences (e.g., DDN-X.25 has an algorithmic method); if all interfaces are of this type, then the Address Translation table is empty, i.e., has zero entries.

**ipNetToMediaTable OBJECT-TYPE**
- **SYNTAX** Sequence of ipNetToMediaEntry
- **ACCESS** not-accessible
- **STATUS** mandatory
- **DESCRIPTION**
The IP Address Translation table used for mapping from IP addresses to physical addresses.

```plaintext
::= { ip 22 }
```

**ipNetToMediaEntry OBJECT-TYPE**
- **SYNTAX** IpNetToMediaEntry
- **ACCESS** not-accessible
- **STATUS** mandatory
- **DESCRIPTION**
Each entry contains one IPAddress to physical address equivalence.

```plaintext
INDEX   { ipNetToMediaIfIndex, ipNetToMediaNetAddress }
::= { ipNetToMediaTable 1 }
```

```plaintext
IpNetToMediaEntry ::= SEQUENCE { ipNetToMediaIfIndex INTEGER, ipNetToMediaPhysAddress PhysAddress, ipNetToMediaNetAddress IpAddress, ipNetToMediaType INTEGER }
```

**ipNetToMediaIfIndex OBJECT-TYPE**
- **SYNTAX** INTEGER
- **ACCESS** read-write
- **STATUS** mandatory
- **DESCRIPTION**
The interface on which this entry's equivalence is effective. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex.

```plaintext
::= { ipNetToMediaEntry 1 }
```

**ipNetToMediaPhysAddress OBJECT-TYPE**
- **SYNTAX** PhysAddress
- **ACCESS** read-write
- **STATUS** mandatory
- **DESCRIPTION**
The media-dependent physical address.

```plaintext
::= { ipNetToMediaEntry 2 }
```

**ipNetToMediaNetAddress OBJECT-TYPE**
SYNTAX IpAddress
ACCESS read-write
STATUS mandatory
DESCRIPTION
The IpAddress corresponding to the media-dependent physical address.
::= { ipNetToMediaEntry 3 }

ipNetToMediaType OBJECT-TYPE
SYNTAX Integer { other(1),    -- none of the following invalid(2),    -- an invalidated mapping
dynamic(3), static(4) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
The type of mapping. Setting this object to the value invalid(2) has the effect of invalidating the corresponding entry in the ipNetToMediaTable. That is, it effectively disassociates the interface identified with said entry from the mapping identified with said entry. It is an implementation-specific matter as to whether the agent removes an invalidated entry from the table. Accordingly, management stations must be prepared to receive tabular information from agents that corresponds to entries not currently in use. Proper interpretation of such entries requires examination of the relevant ipNetToMediaType object.
::= { ipNetToMediaEntry 4 }

Additional IP Objects

ipRoutingDiscards OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of routing entries which were chosen to be discarded even though they are valid. One possible reason for discarding such an entry could be to free-up buffer space for other routing entries.
::= { ip 23 }

ICMP Group

Implementation of the ICMP group is mandatory for all systems.

icmpInMsgs OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of ICMP messages which the entity received. Note that this counter includes all those counted by icmpInErrors.

::= { icmp 1 }

**icmpInErrors OBJECT-TYPE**

SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of ICMP messages which the entity received but determined as having ICMP-specific errors (bad ICMP checksums, bad length, etc.).

::= { icmp 2 }

**icmpInDestUnreaches OBJECT-TYPE**

SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of ICMP Destination Unreachable messages received.

::= { icmp 3 }

**icmpInTimeExcds OBJECT-TYPE**

SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of ICMP Time Exceeded messages received.

::= { icmp 4 }

**icmpInParmProbs OBJECT-TYPE**

SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of ICMP Parameter Problem messages received.

::= { icmp 5 }

**icmpInSrcQuenches OBJECT-TYPE**

SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of ICMP Source Quench messages received.
::= { icmp 6 }

icmpInRedirects OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Redirect messages received.
::= { icmp 7 }

icmpInEchos OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Echo (request) messages received.
::= { icmp 8 }

icmpInEchoReps OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Echo Reply messages received.
::= { icmp 9 }

icmpInTimestamps OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Timestamp (request) messages received.
::= { icmp 10 }

icmpInTimestampReps OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Timestamp Reply messages received.
::= { icmp 11 }

icmpInAddrMasks OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Address Mask Request messages received.
 ::= { icmp 12 }

**icmpInAddrMaskReps OBJECT-TYPE**
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Address Mask Reply messages received.
 ::= { icmp 13 }

**icmpOutMsgs OBJECT-TYPE**
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of ICMP messages which this entity attempted to send. Note that this counter includes all those counted by icmpOutErrors.
 ::= { icmp 14 }

**icmpOutErrors OBJECT-TYPE**
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP messages which this entity did not send due to problems discovered within ICMP such as a lack of buffers. This value should not include errors discovered outside the ICMP layer such as the inability of IP to route the resultant datagram. In some implementations there may be no types of error which contribute to this counter value.
 ::= { icmp 15 }

**icmpOutDestUnreachs OBJECT-TYPE**
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Destination Unreachable messages sent.
 ::= { icmp 16 }

**icmpOutTimeExcds OBJECT-TYPE**
SYNTAX  Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of ICMP Time Exceeded messages sent.
 ::= { icmp 17 }

`icmpOutParmProbs` OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of ICMP Parameter Problem messages sent.
 ::= { icmp 18 }

`icmpOutSrcQuenches` OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of ICMP Source Quench messages sent.
 ::= { icmp 19 }

`icmpOutRedirects` OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of ICMP Redirect messages sent. For a host, this object will always be zero, since hosts do not send redirects.
 ::= { icmp 20 }

`icmpOutEchos` OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of ICMP Echo (request) messages sent.
 ::= { icmp 21 }

`icmpOutEchoReps` OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Echo Reply messages sent.
::= { icmp 22 }

**icmpOutTimestamps OBJECT-TYPE**
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Timestamp (request) messages sent.
::= { icmp 23 }

**icmpOutTimestampReps OBJECT-TYPE**
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Timestamp Reply messages sent.
::= { icmp 24 }

**icmpOutAddrMasks OBJECT-TYPE**
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Address Mask Request messages sent.
::= { icmp 25 }

**icmpOutAddrMaskReps OBJECT-TYPE**
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of ICMP Address Mask Reply messages sent.
::= { icmp 26 }

**TCP Group**

Implementation of the TCP group is mandatory for all systems that implement the TCP. Note that instances of object types that represent information about a particular TCP connection are transient; they persist only as long as the connection in question.
RFC1213-MIB (MIB-II)

Chapter 8      Industry-Standard Management Information Base

**tcpRtoAlgorithm** OBJECT-TYPE
SYNTAX  Integer { other(1), -- none of the following constant(2), -- a constant rto rsre(3), -- MIL-STD-1778, Appendix B vanj(4) -- Van Jacobson's algorithm [10] }
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The algorithm used to determine the timeout value used for retransmitting unacknowledged octets.
::= { tcp 1 }

**tcpRtoMin** OBJECT-TYPE
SYNTAX  INTEGER
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The minimum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds. More refined semantics for objects of this type depend upon the algorithm used to determine the retransmission timeout. In particular, when the timeout algorithm is rsre(3), an object of this type has the semantics of the LBOUND quantity described in RFC 793.
::= { tcp 2 }

**tcpRtoMax** OBJECT-TYPE
SYNTAX  Integer
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The maximum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds. More refined semantics for objects of this type depend upon the algorithm used to determine the retransmission timeout. In particular, when the timeout algorithm is rsre(3), an object of this type has the semantics of the UBOUND quantity described in RFC 793.
::= { tcp 3 }

**tcpMaxConn** OBJECT-TYPE
SYNTAX  Integer
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The limit on the total number of TCP connections the entity can support. In entities where the maximum number of connections is dynamic, this object should contain the value -1.
::= { tcp 4 }

**tcpActiveOpens** OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION

Cisco Unified Communications Manager Managed Services Guide

OL-22523-01
Chapter 8  Industry-Standard Management Information Base

DESCRIPTION
The number of times TCP connections have made a direct transition to the SYN-SENT state from the CLOSED state.
::= { tcp 5 }

tcpPassiveOpens OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of times TCP connections have made a direct transition to the SYN-RCVD state from the LISTEN state.
::= { tcp 6 }

tcpAttemptFails OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of times TCP connections have made a direct transition to the CLOSED state from either the SYN-SENT state or the SYN-RCVD state, plus the number of times TCP connections have made a direct transition to the LISTEN state from the SYN-RCVD state.
::= { tcp 7 }

tcpEstabResets OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of times TCP connections have made a direct transition to the CLOSED state from either the ESTABLISHED state or the CLOSE-WAIT state.
::= { tcp 8 }

tcpCurrEstab OBJECT-TYPE
SYNTAX  Gauge
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of TCP connections for which the current state is either ESTABLISHED or CLOSE-WAIT.
::= { tcp 9 }

tcpInSegs OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of segments received, including those received in error. This count includes segments received on currently established connections.
::= { tcp 10 }

tcpOutSegs OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of segments sent, including those on current connections but excluding those containing only retransmitted octets.
::= { tcp 11 }

tcpRetransSegs OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of segments retransmitted that is, the number of TCP segments transmitted containing one or more previously transmitted octets.
::= { tcp 12 }

TCP Connection Table

The TCP connection table contains information about this entity existing TCP connections.

tcpConnTable OBJECT-TYPE
SYNTAX  Sequence of tcpConnEntry
ACCESS  not-accessible
STATUS  mandatory
DESCRIPTION
A table containing TCP connection-specific information.
::= { tcp 13 }

tcpConnEntry OBJECT-TYPE
SYNTAX  TcpConnEntry
ACCESS  not-accessible
STATUS  mandatory
DESCRIPTION
Information about a particular current TCP connection. An object of this type is transient, in that it ceases to exist when (or soon after) the connection makes the transition to the CLOSED state.

INDEX   { tcpConnLocalAddress, tcpConnLocalPort, tcpConnRemAddress, tcpConnRemPort }
::= { tcpConnTable 1 }

TcpConnEntry ::= SEQUENCE { tcpConnState INTEGER, tcpConnLocalAddress IpAddress, tcpConnLocalPort INTEGER (0..65535), tcpConnRemAddress IpAddress, tcpConnRemPort INTEGER (0..65535) }

tcpConnState OBJECT-TYPE
SYNTAX  INTEGER { closed(1), listen(2), synSent(3), synReceived(4), established(5), finWait1(6), finWait2(7), closeWait(8), lastAck(9), closing(10), timeWait(11), deleteTCB(12) }
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
The state of this TCP connection. The only value which may be set by a management station is deleteTCB(12). Accordingly, it is appropriate for an agent to return a badValue response if a management station attempts to set this object to any other value. If a management station sets this object to the value deleteTCB(12), then this has the effect of deleting the TCB (as defined in RFC 793) of the corresponding connection on the managed node, resulting in immediate termination of the connection.

As an implementation-specific option, a RST segment may be sent from the managed node to the other TCP endpoint (note however that RST segments are not sent reliably).
::= { tcpConnEntry 1 }

tcpConnLocalAddress OBJECT-TYPE
SYNTAX  IpAddress
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The local IP address for this TCP connection. In the case of a connection in the listen state which is willing to accept connections for any IP interface associated with the node, the value 0.0.0.0 is used.
::= { tcpConnEntry 2 }

tcpConnLocalPort OBJECT-TYPE
SYNTAX  INTEGER (0..65535)
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The local port number for this TCP connection.
::= { tcpConnEntry 3 }

tcpConnRemAddress OBJECT-TYPE
SYNTAX  IpAddress
ACCESS  read-only
STATUS mandatory
DESCRIPTION
The remote IP address for this TCP connection.
 ::= { tcpConnEntry 4 }

tcpConnRemPort OBJECT-TYPE
SYNTAX  INTEGER (0..65535)
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The remote port number for this TCP connection.
 ::= { tcpConnEntry 5 }

Additional TCP Objects

tcpInErrs OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of segments received in error (e.g., bad TCP checksums).
 ::= { tcp 14 }

tcpOutRsts OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of TCP segments sent containing the RST flag.
 ::= { tcp 15 }

UDP Group

Implementation of the UDP group is mandatory for all systems which implement the UDP.

udpInDatagrams OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of UDP datagrams delivered to UDP users.
 ::= { udp 1 }
udpNoPorts OBJECT-TYPE
  SYNTAX  Counter
  ACCESS  read-only
  STATUS  mandatory
  DESCRIPTION
  The total number of received UDP datagrams for which there was no application at the destination port.
  ::= { udp 2 }

udpInErrors OBJECT-TYPE
  SYNTAX  Counter
  ACCESS  read-only
  STATUS  mandatory
  DESCRIPTION
  The number of received UDP datagrams that could not be delivered for reasons other than the lack of an application at the destination port.
  ::= { udp 3 }

udpOutDatagrams OBJECT-TYPE
  SYNTAX  Counter
  ACCESS  read-only
  STATUS  mandatory
  DESCRIPTION
  The total number of UDP datagrams sent from this entity.
  ::= { udp 4 }

UDP Listener Table

The UDP listener table contains information about this entity UDP end-points on which a local application is currently accepting datagrams.

udpTable OBJECT-TYPE
  SYNTAX  SEQUENCE OF UdpEntry
  ACCESS  not-accessible
  STATUS  mandatory
  DESCRIPTION
  A table containing UDP listener information.
  ::= { udp 5 }

udpEntry OBJECT-TYPE
  SYNTAX  UdpEntry
  ACCESS  not-accessible
  STATUS  mandatory
DESCRIPTION
Information about a particular current UDP listener.
INDEX   { udpLocalAddress, udpLocalPort }
 ::= { udpTable 1 }
UdpEntry ::= 
 SEQUENCE { udpLocalAddress IpAddress, udpLocalPort INTEGER (0..65535) }

udpLocalAddress OBJECT-TYPE
 SYNTAX  IpAddress
 ACCESS  read-only
 STATUS  mandatory
 DESCRIPTION
 The local IP address for this UDP listener. In the case of a UDP listener which is willing to accept
datagrams for any IP interface associated with the node, the value 0.0.0.0 is used.
 ::= { udpEntry 1 }

udpLocalPort OBJECT-TYPE
 SYNTAX  INTEGER (0..65535)
 ACCESS  read-only
 STATUS  mandatory
 DESCRIPTION
 The local port number for this UDP listener.
 ::= { udpEntry 2 }

EGP Group

Implementation of the EGP group is mandatory for all systems which implement the EGP.

egpInMsgs OBJECT-TYPE
 SYNTAX  Counter
 ACCESS  read-only
 STATUS  mandatory
 DESCRIPTION
 The number of EGP messages received without error.
 ::= { egp 1 }

egpInErrors OBJECT-TYPE
 SYNTAX  Counter
 ACCESS  read-only
 STATUS  mandatory
 DESCRIPTION
 The number of EGP messages received that proved to be in error.
::= { egp 2 }

egpOutMsgs OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of locally generated EGP messages.
::= { egp 3 }

egpOutErrors OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The number of locally generated EGP messages not sent due to resource limitations within an EGP entity.
::= { egp 4 }

EGP Neighbor Table
The EGP neighbor table contains information about this entity EGP neighbors.

egpNeighTable OBJECT-TYPE
SYNTAX  SEQUENCE OF EgpNeighEntry
ACCESS  not-accessible
STATUS  mandatory
DESCRIPTION
The EGP neighbor table.
::= { egp 5 }

egpNeighEntry OBJECT-TYPE
SYNTAX  EgpNeighEntry
ACCESS  not-accessible
STATUS  mandatory
DESCRIPTION
Information about this entity's relationship with a particular EGP neighbor.
INDEX   { egpNeighAddr }
::= { egpNeighTable 1 }
EgpNeighEntry ::=
SEQUENCE { egpNeighState INTEGER, egpNeighAddr IpAddress, egpNeighAs INTEGER, egpNeighInMsgs Counter, egpNeighInErrs Counter, egpNeighOutMsgs Counter, egpNeighOutErrs Counter, egpNeighInErrMsgs Counter, egpNeighOutErrMsgs Counter, egpNeighStateUps Counter, egpNeighState Downs Counter, egpNeighIntervalHello INTEGER, egpNeighIntervalPoll INTEGER, egpNeighMode INTEGER, egpNeighEventTrigger INTEGER }

egpNeighState OBJECT-TYPE
SYNTAX Integer { idle(1), acquisition(2), down(3), up(4), cease(5) }
ACCESS read-only
STATUS mandatory
DESCRIPTION
The EGP state of the local system with respect to the entry EGP neighbor. Each EGP state is represented by a value that is one greater than the numerical value associated with said state in RFC 904.
 ::= { egpNeighEntry 1 }

egpNeighAddr OBJECT-TYPE
SYNTAX IpAddress
ACCESS read-only
STATUS mandatory
DESCRIPTION
The IP address of this entry's EGP neighbor.
 ::= { egpNeighEntry 2 }

egpNeighAs OBJECT-TYPE
SYNTAX Integer
ACCESS read-only
STATUS mandatory
DESCRIPTION
The autonomous system of this EGP peer. Zero should be specified if the autonomous system number of the neighbor is not yet known.
 ::= { egpNeighEntry 3 }

egpNeighInMsgs OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of EGP messages received without error from this EGP peer.
 ::= { egpNeighEntry 4 }

egpNeighInErrs OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of EGP messages received from this EGP peer that proved to be in error (e.g., bad EGP checksum).

::= { egpNeighEntry 5 }

egpNeighOutMsgs OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of locally generated EGP messages to this EGP peer.

::= { egpNeighEntry 6 }

egpNeighOutErrs OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of locally generated EGP messages not sent to this EGP peer due to resource limitations within an EGP entity.

::= { egpNeighEntry 7 }

egpNeighInErrMsgs OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of EGP-defined error messages received from this EGP peer.

::= { egpNeighEntry 8 }

egpNeighOutErrMsgs OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of EGP-defined error messages sent to this EGP peer.

::= { egpNeighEntry 9 }

egpNeighStateUps OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION


The number of EGP state transitions to the UP state with this EGP peer.
::= { egpNeighEntry 10 }

egpNeighStateDowns OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The number of EGP state transitions from the UP state to any other state with this EGP peer.
::= { egpNeighEntry 11 }

egpNeighIntervalHello OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
The interval between EGP Hello command retransmissions (in hundredths of a second). This represents the t1 timer as defined in RFC 904.
::= { egpNeighEntry 12 }

egpNeighIntervalPoll OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
The interval between EGP poll command retransmissions (in hundredths of a second). This represents the t3 timer as defined in RFC 904.
::= { egpNeighEntry 13 }

egpNeighMode OBJECT-TYPE
SYNTAX INTEGER { active(1), passive(2) }
ACCESS read-only
STATUS mandatory
DESCRIPTION
The polling mode of this EGP entity, either passive or active.
::= { egpNeighEntry 14 }

egpNeighEventTrigger OBJECT-TYPE
SYNTAX INTEGER { start(1), stop(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
A control variable used to trigger operator-initiated Start and Stop events. When read, this variable always returns the most recent value that egpNeighEventTrigger was set to. If it has not been set since the last initialization of the network management subsystem on the node, it returns a value of stop. When set, this variable causes a Start or Stop event on the specified neighbor, as specified on pages 8-10 of RFC 904. Briefly, a Start event causes an Idle peer to begin neighbor acquisition and a non-Idle peer to reinitiate neighbor acquisition. A stop event causes a non-Idle peer to return to the Idle state until a Start event occurs, either via egpNeighEventTrigger or otherwise.

```markdown
::= { egpNeighEntry 15 }
```

**Additional EGP Objects**

`egpAs OBJECT-TYPE`  
SYNTAX INTEGER  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
The autonomous system number of this EGP entity.  
 ::= { egp 6 }

**Transmission Group**

Based on the transmission media underlying each interface on a system, the corresponding portion of the Transmission group is mandatory for that system. When Internet-standard definitions for managing transmission media are defined, the transmission group is used to provide a prefix for the names of those objects. Typically, such definitions reside in the experimental portion of the MIB until they are proven, then as a part of the Internet standardization process, the definitions are accordingly elevated and a new object identifier, under the transmission group is defined. By convention, the name assigned is: type OBJECT IDENTIFIER ::= { transmission number } where type is the symbolic value used for the media in the ifType column of the ifTable object, and number is the actual integer value corresponding to the symbol.

**SNMP Group**

Implementation of the SNMP group is mandatory for all systems which support an SNMP protocol entity. Some of the objects defined below will be zero-valued in those SNMP implementations that are optimized to support only those functions specific to either a management agent or a management station. In particular, it should be observed that the objects below refer to an SNMP entity, and there may be several SNMP entities residing on a managed node (e.g., if the node is hosting acting as a management station).

`snmpInPkts OBJECT-TYPE`  
SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION
The total number of Messages delivered to the SNMP entity from the transport service.

```plaintext
::= { snmp 1 }
```

### snmpOutPkts OBJECT-TYPE

- **SYNTAX**: Counter
- **ACCESS**: read-only
- **STATUS**: mandatory
- **DESCRIPTION**: The total number of SNMP Messages which were passed from the SNMP protocol entity to the transport service.

```plaintext
::= { snmp 2 }
```

### snmpInBadVersions OBJECT-TYPE

- **SYNTAX**: Counter
- **ACCESS**: read-only
- **STATUS**: mandatory
- **DESCRIPTION**: The total number of SNMP Messages which were delivered to the SNMP protocol entity and were for an unsupported SNMP version.

```plaintext
::= { snmp 3 }
```

### snmpInBadCommunityNames OBJECT-TYPE

- **SYNTAX**: Counter
- **ACCESS**: read-only
- **STATUS**: mandatory
- **DESCRIPTION**: The total number of SNMP Messages delivered to the SNMP protocol entity which used a SNMP community name not known to said entity.

```plaintext
::= { snmp 4 }
```

### snmpInBadCommunityUses OBJECT-TYPE

- **SYNTAX**: Counter
- **ACCESS**: read-only
- **STATUS**: mandatory
- **DESCRIPTION**: The total number of SNMP Messages delivered to the SNMP protocol entity which represented an SNMP operation which was not allowed by the SNMP community named in the Message.

```plaintext
::= { snmp 5 }
```

### snmpInASNParseErrs OBJECT-TYPE

- **SYNTAX**: Counter
- **ACCESS**: read-only
- **STATUS**: mandatory
- **DESCRIPTION**: The total number of SNMP Messages delivered to the SNMP protocol entity which used a Community Name not known to the entity.
The total number of ASN.1 or BER errors encountered by the SNMP protocol entity when decoding received SNMP Messages.

\[ \text{snmpInTooBigs} \]

\[
\text{snmpInNoSuchNames} \]

\[
\text{snmpInBadValues} \]

\[
\text{snmpInReadOnlys} \]

\[
\text{snmpInGenErrs} \]
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of SNMP PDUs which were delivered to the SNMP protocol entity and for which the value of the error-status field is genErr.
::= { snmp 12 }

snmpInTotalReqVars OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of MIB objects which have been retrieved successfully by the SNMP protocol entity as the result of receiving valid SNMP Get-Request and Get-Next PDUs.
::= { snmp 13 }

snmpInTotalSetVars OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of MIB objects which have been altered successfully by the SNMP protocol entity as the result of receiving valid SNMP Set-Request PDUs.
::= { snmp 14 }

snmpInGetRequests OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of SNMP Get-Request PDUs which have been accepted and processed by the SNMP protocol entity.
::= { snmp 15 }

snmpInGetNexts OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of SNMP Get-Next PDUs which have been accepted and processed by the SNMP protocol entity.
::= { snmp 16 }

snmpInSetRequests OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of SNMP Set-Request PDUs which have been accepted and processed by the SNMP protocol entity.
 ::= { snmp 17 }

snmpInGetResponses OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of SNMP Get-Response PDUs which have been accepted and processed by the SNMP protocol entity.
 ::= { snmp 18 }

snmpInTraps OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of SNMP Trap PDUs which have been accepted and processed by the SNMP protocol entity.
 ::= { snmp 19 }

snmpOutTooBigs OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of SNMP PDUs which were generated by the SNMP protocol entity and for which the value of the error-status field is tooBig.
 ::= { snmp 20 }

snmpOutNoSuchNames OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of SNMP PDUs which were generated by the SNMP protocol entity and for which the value of the error-status is noSuchName.
 ::= { snmp 21 }
snmpOutBadValues OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of SNMP PDUs which were generated by the SNMP protocol entity and for which
the value of the error-status field is badValue.
 ::= { snmp 22 }
-- { snmp 23 } is not used

snmpOutGenErrs OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of SNMP PDUs which were generated by the SNMP protocol entity and for which
the value of the error-status field is genErr.
 ::= { snmp 24 }

snmpOutGetRequests OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of SNMP Get-Request PDUs which have been generated by the SNMP protocol
entity.
 ::= { snmp 25 }

snmpOutGetNexts OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of SNMP Get-Next PDUs which have been generated by the SNMP protocol
entity.
 ::= { snmp 26 }

snmpOutSetRequests OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
The total number of SNMP Set-Request PDUs which have been generated by the SNMP protocol entity.
::= { snmp 27 }

**snmpOutGetResponses** OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of SNMP Get-Response PDUs which have been generated by the SNMP protocol entity.
::= { snmp 28 }

**snmpOutTraps** OBJECT-TYPE
SYNTAX  Counter
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
The total number of SNMP Trap PDUs which have been generated by the SNMP protocol entity.
::= { snmp 29 }

**snmpEnableAuthenTraps** OBJECT-TYPE
SYNTAX  Integer { enabled(1), disabled(2) }
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
Indicates whether the SNMP agent process is permitted to generate authentication-failure traps. The value of this object overrides any configuration information; as such, it provides a means whereby all authentication-failure traps may be disabled. Note that it is strongly recommended that this object be stored in non-volatile memory so that it remains constant between re-initializations of the network management system.
::= { snmp 30 }

**HOST-RESOURCES-MIB**


This MIB manages host systems. The term “host” means any computer that communicates with other similar computers attached to the internet and that is directly used by one or more human beings. Although this MIB does not necessarily apply to devices whose primary function is communications
services (terminal servers, routers, bridges, and monitoring equipment), such relevance is not explicitly precluded. This MIB contains attributes that are common to all internet hosts including, for example, both personal computers and systems that run variants of Unix.

Before you can compile HOST-RESOURCES-MIB, you need to compile the MIBs listed below in the order listed.

1. SNMPv2-SMI
2. SNMPv2-TC
3. SNMPv2-CONF
4. SNMPv2-MIB
5. IANAifType-MIB
6. IF-MIB
7. RFC1155-SMI
8. RFC-1212
9. SNMPv2-SMI-v1
10. SNMPv2-TC-v1

Additional downloads are:
- OID File: HOST-RESOURCES-MIB.oid

The following are contained in this section:
- Revisions, page 8-75
- Definitions, page 8-76
- Object Identifiers, page 8-76
- Textual Conventions, page 8-76
- Host Resources System Group, page 8-77
- Host Resources Storage Group, page 8-79
- Host Resources Device Group, page 8-81
- Host Resources Running Software Group, page 8-92
- Host Resources Running Software Performance Group, page 8-95
- Host Resources Installed Software Group, page 8-96
- Conformance Information, page 8-98
- Compliance Statements, page 8-98
- Cisco Unified CM Release 6.x Feature Services, page 8-100
- Cisco Unified CM Release 6.x Network Services, page 8-102
- Troubleshooting, page 8-103
Revisions

Table 8-2 lists the revisions to this MIB beginning with the latest revision.

Table 8-2  History of Revisions

<table>
<thead>
<tr>
<th>Date</th>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
</table>
| 03-06-2000 | Added and updated | Clarifications and bug fixes based on implementation experience. This revision was also reformatted in the SMIv2 format. The revisions made were:  
• Reformatted to new RFC document standards  
• Added copyright notice  
• Updated introduction to SNMP Framework  
• Updated references section  
• Added reference to RFC 2119  
• Added a meaningful security considerations section  
New IANA considerations section for registration of new types, conversion to new SMIv2 syntax for the following types and macros:  
• Counter32, Integer32, Gauge32, MODULE-IDENTITY, OBJECT-TYPE, TEXTUAL-CONVENTION, OBJECT-IDENTITY, MODULE-COMPLIANCE, OBJECT-GROUP  
• Used new Textual Conventions: TruthValue, DateAndTime, AutonomousType, InterfaceIndexOrZero  
• Fixed typo in hrPrinterStatus  
• Added missing error bits to hrPrinterDetectedErrorState  
• Clarified confusion resulting from suggested mappings to hrPrinterStatus.  
• Clarified that size of objects of type InternationalDisplayString is number of octets, not number of encoded symbols.  
• Clarified the use of the following objects based on implementation experience: hrSystemInitialLoadDevice, hrSystemInitialLoadParameters, hrMemorySize, hrStorageSize, hrStorageAllocationFailures, hrDeviceErrors, hrProcessorLoad, hrNetworkIfIndex, hrDiskStorageCapacity, hrSWRunStatus, hrSWRunPerfCPU, and hrSWInstalledDate.  
• Clarified implementation technique for hrSWInstalledTable.  
• Used new AUGMENTS clause for hrSWRunPerfTable.  
• Added Internationalization Considerations section. This revision published as RFC2790. |
| 10-20-1999 | Initial Version   | The original version of this MIB, published as RFC1514. ::= { hrMIBAdminInfo 1 }                                                                 |

Cisco Unified Communications Manager Managed Services Guide

OL-22523-01

8-75
Definitions

The following definitions are imported for HOST-RESOURCES-MIB:

- MODULE-IDENTITY, OBJECT-TYPE, mib-2, Integer32, Counter32, Gauge32, TimeTicks
- From SNMPv2-SMI—TEXTUAL-CONVENTION, DisplayString, TruthValue, DateAndTime, AutonomousType
- From SNMPv2-TC—MODULE-COMPLIANCE, OBJECT-GROUP
- From SNMPv2-CONF—InterfaceIndexOrZero
- From IF-MIB—hostResourcesMibModule MODULE-IDENTITY

Object Identifiers

host OBJECT IDENTIFIER ::= { mib-2 25 }
hrSystem OBJECT IDENTIFIER ::= { host 1 }
hrStorage OBJECT IDENTIFIER ::= { host 2 }
hrDevice OBJECT IDENTIFIER ::= { host 3 }
hSWRun OBJECT IDENTIFIER ::= { host 4 }
hSWRunPerf OBJECT IDENTIFIER ::= { host 5 }
hSWInstalled OBJECT IDENTIFIER ::= { host 6 }
hMIBAdminInfo OBJECT IDENTIFIER ::= { host 7 }

Textual Conventions

KBytes ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
Storage size, expressed in units of 1024 bytes.
SYNTAX Integer32 (0..2147483647)

ProductID ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
This textual convention is intended to identify the manufacturer, model, and version of a specific
hardware or software product. It is suggested that these OBJECT IDENTIFIERS are allocated such
that all products from a particular manufacturer are registered under a subtree distinct to that
manufacturer. In addition, all versions of a product should be registered under a subtree distinct to
that product. With this strategy, a management station may uniquely determine the manufacturer
and/or model of a product whose ProductID is unknown to the management station. Objects of this
type may be useful for inventory purposes or for automatically detecting incompatibilities or version
mismatches between various hardware and software components on a system.
For example, the product ID for the ACME 4860 66MHz clock doubled processor might be: enterprises.acme.acmeProcessors.a4860DX2.MHz66. A software product might be registered as: enterprises.acme.acmeOperatingSystems.acmeDOS.six(6).one(1).

SYNTAX OBJECT IDENTIFIER

UnknownProduct will be used for any unknown ProductID. UnknownProduct OBJECT IDENTIFIER ::= { 0 0 }.

InternationalDisplayString ::= TEXTUAL-CONVENTION

STATUS current
DESCRIPTION
This data type is used to model textual information in some character set. A network management station should use a local algorithm to determine which character set is in use and how it should be displayed. Note that this character set may be encoded with more than one octet per symbol, but will most often be NVT ASCII. When a size clause is specified for an object of this type, the size refers to the length in octets, not the number of symbols.

SYNTAX OCTET STRING

Host Resources System Group

hrSystemUptime OBJECT-TYPE
SYNTAX TimeTicks
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The amount of time since this host was last initialized. Note that this is different from sysUpTime in the SNMPv2-MIB [RFC1907] because sysUpTime is the uptime of the network management portion of the system.

::= { hrSystem 1 }

hrSystemDate OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The host's notion of the local date and time of day.

::= { hrSystem 2 }

hrSystemInitialLoadDevice OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The index of the hrDeviceEntry for the device from which this host is configured to load its initial operating system configuration (i.e., which operating system code and/or boot parameters). Note that writing to this object just changes the configuration that will be used the next time the operating system is loaded and does not actually cause the reload to occur.

::= { hrSystem 3 }

hrSystemInitialLoadParameters OBJECT-TYPE
SYNTAX InternationalDisplayString (SIZE (0..128))
MAX-ACCESS read-write
STATUS current
DESCRIPTION
This object contains the parameters (e.g. a pathname and parameter) supplied to the load device when requesting the initial operating system configuration from that device. Note that writing to this object just changes the configuration that will be used the next time the operating system is loaded and does not actually cause the reload to occur.

::= { hrSystem 4 }

hrSystemNumUsers OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of user sessions for which this host is storing state information. A session is a collection of processes requiring a single act of user authentication and possibly subject to collective job control.

::= { hrSystem 5 }

hrSystemProcesses OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of process contexts currently loaded or running on this system.

::= { hrSystem 6 }

hrSystemMaxProcesses OBJECT-TYPE
SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The maximum number of process contexts this system can support. If there is no fixed maximum, the value should be zero. On systems that have a fixed maximum, this object can help diagnose failures that occur when this maximum is reached.

::= { hrSystem 7 }
Host Resources Storage Group

Registration point for storage types, for use with hrStorageType. These are defined in the HOST-RESOURCES-TYPES module.

hrStorageTypes OBJECT IDENTIFIER ::= { hrStorage 1 }

hrMemorySize OBJECT-TYPE
SYNTAX KBytes
UNITS KBytes
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The amount of physical read-write main memory, typically RAM, contained by the host.
::= { hrStorage 2 }

hrStorageTable OBJECT-TYPE
SYNTAX Sequence of HrStorageEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The (conceptual) table of logical storage areas on the host. An entry shall be placed in the storage table for each logical area of storage that is allocated and has fixed resource limits. The amount of storage represented in an entity is the amount actually usable by the requesting entity, and excludes loss due to formatting or file system reference information.

These entries are associated with logical storage areas, as might be seen by an application, rather than physical storage entities which are typically seen by an operating system. Storage such as tapes and floppies without file systems on them are typically not allocated in chunks by the operating system to requesting applications, and therefore shouldn't appear in this table. Examples of valid storage for this table include disk partitions, file systems, RAM (for some architectures this is further segmented into regular memory, extended memory, and so on), backing store for virtual memory (‘swap space’).

This table is intended to be a useful diagnostic for “out of memory” and “out of buffers” types of failures. In addition, it can be a useful performance monitoring tool for tracking memory, disk, or buffer usage.
::= { hrStorage 3 }

hrStorageEntry OBJECT-TYPE
SYNTAX HrStorageEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A (conceptual) entry for one logical storage area on the host. As an example, an instance of the hrStorageType object might be named hrStorageType.3
INDEX { hrStorageIndex }
::= { hrStorageTable 1 }
hrStorageEntry ::= SEQUENCE { hrStorageIndex  Integer32, hrStorageTypeAutonomousType, hrStorageDescr   DisplayString, hrStorageAllocationUnits Integer32, hrStorageSizeInteger32, hrStorageUsedInteger32, hrStorageAllocationFailures  Counter32 }

hrStorageIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A unique value for each logical storage area contained by the host.
::= { hrStorageEntry 1 }

hrStorageType OBJECT-TYPE
SYNTAX AutonomousType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The type of storage represented by this entry.
::= { hrStorageEntry 2 }

hrStorageDescr OBJECT-TYPE
SYNTAX DisplayString
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A description of the type and instance of the storage described by this entry.
::= { hrStorageEntry 3 }

hrStorageAllocationUnits OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
UNITS Bytes
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The size, in bytes, of the data objects allocated from this pool. If this entry is monitoring sectors, blocks, buffers, or packets, for example, this number will commonly be greater than one. Otherwise this number will typically be one.
::= { hrStorageEntry 4 }

hrStorageSize OBJECT-TYPE
SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The size of the storage represented by this entry, in units of hrStorageAllocationUnits. This object is writable to allow remote configuration of the size of the storage area in those cases where such an operation makes sense and is possible on the underlying system. For example, the amount of main memory allocated to a buffer pool might be modified or the amount of disk space allocated to virtual memory might be modified.

```plaintext
hrStorageEntry 5
```

**hrStorageUsed OBJECT-TYPE**

SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The amount of the storage represented by this entry that is allocated, in units of hrStorageAllocationUnits.

```plaintext
hrStorageEntry 6
```

**hrStorageAllocationFailures OBJECT-TYPE**

SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of requests for storage represented by this entry that could not be honored due to not enough storage. It should be noted that as this object has a SYNTAX of Counter32, that it does not have a defined initial value. However, it is recommended that this object be initialized to zero, even though management stations must not depend on such an initialization.

```plaintext
hrStorageEntry 7
```

---

**Host Resources Device Group**

The device group is useful for identifying and diagnosing the devices on a system. The hrDeviceTable contains common information for any type of device. In addition, some devices have device-specific tables for more detailed information. More such tables may be defined in the future for other device types. Registration point for device types, for use with hrDeviceType. These are defined in the HOST-RESOURCES-TYPES module.

```plaintext
hrDeviceTypes OBJECT IDENTIFIER ::= { hrDevice 1 }
```

**hrDeviceTable OBJECT-TYPE**

SYNTAX Sequence of hrDeviceEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The (conceptual) table of devices contained by the host.

```plaintext
hrDeviceEntry 2
```

---
SYNTAX hrDeviceEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A (conceptual) entry for one device contained by the host. As an example, an instance of the
hrDeviceType object might be named hrDeviceType.3
INDEX { hrDeviceIndex } ::= { hrDeviceTable 1 }

HrDeviceEntry ::= SEQUENCE { hrDeviceIndex   Integer32, hrDeviceTypeAutonomousType,
hrDeviceDescr   DisplayString, hrDeviceID  ProductID, hrDeviceStatus  INTEGER, hrDeviceErrors
Counter32 }

hrDeviceIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A unique value for each device contained by the host. The value for each device must remain
constant at least from one re-initialization of the agent to the next re-initialization.
 ::= { hrDeviceEntry 1 }

hrDeviceType OBJECT-TYPE
SYNTAX AutonomousType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
An indication of the type of device. If this value is "hrDeviceProcessor { hrDeviceTypes 3 }" then
an entry exists in the hrProcessorTable which corresponds to this device. If this value is
"hrDeviceNetwork { hrDeviceTypes 4 }", then an entry exists in the hrNetworkTable which
corresponds to this device. If this value is "hrDevicePrinter { hrDeviceTypes 5 }", then an entry
exists in the hrPrinterTable which corresponds to this device.
If this value is "hrDeviceDiskStorage { hrDeviceTypes 6 }", then an entry exists in the
hrDiskStorageTable which corresponds to this device.
 ::= { hrDeviceEntry 2 } 

hrDeviceDescr OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..64))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A textual description of this device, including the device's manufacturer and revision, and
optionally, its serial number.
 ::= { hrDeviceEntry 3 }
hrDeviceID OBJECT-TYPE
SYNTAX ProductID
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The product ID for this device.
 ::= { hrDeviceEntry 4 }
hrDeviceStatus OBJECT-TYPE
SYNTAX INTEGER { unknown(1), running(2), warning(3), testing(4), down(5) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The current operational state of the device described by this row of the table. A value unknown(1) indicates that the current state of the device is unknown. running(2) indicates that the device is up and running and that no unusual error conditions are known. The warning(3) state indicates that agent has been informed of an unusual error condition by the operational software (e.g., a disk device driver) but that the device is still 'operational'. An example would be a high number of soft errors on a disk. A value of testing(4), indicates that the device is not available for use because it is in the testing state. The state of down(5) is used only when the agent has been informed that the device is not available for any use.
 ::= { hrDeviceEntry 5 }
hrDeviceErrors OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of errors detected on this device. It should be noted that as this object has a SYNTAX of Counter32, that it does not have a defined initial value. However, it is recommended that this object be initialized to zero, even though management stations must not depend on such an initialization.
 ::= { hrDeviceEntry 6 }
hrProcessorTable OBJECT-TYPE
SYNTAX Sequence of hrProcessorEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The (conceptual) table of processors contained by the host. Note that this table is potentially sparse: a (conceptual) entry exists only if the correspondent value of the hrDeviceType object is hrDeviceProcessor.
 ::= { hrDevice 3 }
hrProcessorEntry OBJECT-TYPE
SYNTAX hrProcessorEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A (conceptual) entry for one processor contained by the host. The hrDeviceIndex in the index represents the entry in the hrDeviceTable that corresponds to the hrProcessorEntry. As an example of how objects in this table are named, an instance of the hrProcessorFrwID object might be named hrProcessorFrwID.3
INDEX { hrDeviceIndex }
 ::= { hrProcessorTable 1 }

hrProcessorEntry ::= SEQUENCE { hrProcessorFrwIDProductID, hrProcessorLoad Integer32 }

hrProcessorFrwID OBJECT-TYPE
SYNTAX ProductID
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The product ID of the firmware associated with the processor.
 ::= { hrProcessorEntry 1 }

hrProcessorLoad OBJECT-TYPE
SYNTAX Integer32 (0..100)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The average, over the last minute, of the percentage of time that this processor was not idle. Implementations may approximate this one minute smoothing period if necessary.
 ::= { hrProcessorEntry 2 }

hrNetworkTable OBJECT-TYPE
SYNTAX Sequence of hrNetworkEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The (conceptual) table of network devices contained by the host. Note that this table is potentially sparse: a (conceptual) entry exists only if the correspondent value of the hrDeviceType object is hrDeviceNetwork.
 ::= { hrDevice 4 }

hrNetworkEntry OBJECT-TYPE
SYNTAX hrNetworkEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A (conceptual) entry for one network device contained by the host. The hrDeviceIndex in the index represents the entry in the hrDeviceTable that corresponds to the hrNetworkEntry. As an example of how objects in this table are named, an instance of the hrNetworkIfIndex object might be named hrNetworkIfIndex.3.
INDEX { hrDeviceIndex }
::= { hrNetworkTable 1 }
hrNetworkEntry ::= SEQUENCE { hrNetworkIfIndex InterfaceIndexOrZero }

hrNetworkIfIndex OBJECT-TYPE
SYNTAX InterfaceIndexOrZero
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The value of ifIndex which corresponds to this network device. If this device is not represented in the ifTable, then this value shall be zero.
::= { hrNetworkEntry 1 }

hrPrinterTable OBJECT-TYPE
SYNTAX Sequence of hrPrinterEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The (conceptual) table of printers local to the host. Note that this table is potentially sparse: a (conceptual) entry exists only if the correspondent value of the hrDeviceType object is hrDevicePrinter.
::= { hrDevice 5 }

hrPrinterEntry OBJECT-TYPE
SYNTAX hrPrinterEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A (conceptual) entry for one printer local to the host. The hrDeviceIndex in the index represents the entry in the hrDeviceTable that corresponds to the hrPrinterEntry.
As an example of how objects in the table are named an instance of the hrPrinterStatus object might be named hrPrinterStatus.3
INDEX { hrDeviceIndex }
::= { hrPrinterTable 1 }
hrPrinterEntry ::= SEQUENCE { hrPrinterStatus INTEGER, hrPrinterDetectedErrorState OCTET STRING }

hrPrinterStatus OBJECT-TYPE
SYNTAX INTEGER { other(1), unknown(2), idle(3), printing(4), warmup(5) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The current status of this printer device.
::= { hrPrinterEntry 1 }

hrPrinterDetectedErrorState OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-only
STATUS current
DESCRIPTION
This object represents any error conditions detected by the printer. The error conditions are encoded as bits in an octet string, with the following definitions (condition first then bit number):
- lowPaper 0
- noPaper  1
- lowToner 2
- noToner  3
- doorOpen 4
- jammed5
- offline  6
- serviceRequested 7
- inputTrayMissing 8
- outputTrayMissing 9
- markerSupplyMissing 10
- outputNearFull 11
- outputFull 12
- inputTrayEmpty 13
- overduePreventMaint 14

Bits are numbered starting with the most significant bit of the first byte being bit 0, the least significant bit of the first byte being bit 7, the most significant bit of the second byte being bit 8, and so on. A one bit encodes that the condition was detected, while a zero bit encodes that the condition was not detected.

This object is useful for alerting an operator to specific warning or error conditions that may occur, especially those requiring human intervention.
::= { hrPrinterEntry 2 }

hrDiskStorageTable OBJECT-TYPE
SYNTAX Sequence of hrDiskStorageEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The (conceptual) table of long-term storage devices contained by the host. In particular, disk devices accessed remotely over a network are not included here. Note that this table is potentially sparse: a (conceptual) entry exists only if the correspondent value of the hrDeviceType object is hrDeviceDiskStorage.

:= { hrDevice 6 }

**hrDiskStorageEntry OBJECT-TYPE**

SYNTAX hrDiskStorageEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

A (conceptual) entry for one long-term storage device contained by the host. The hrDeviceIndex in the index represents the entry in the hrDeviceTable that corresponds to the hrDiskStorageEntry. As an example, an instance of the hrDiskStorageCapacity object might be named hrDiskStorageCapacity.3

INDEX { hrDeviceIndex }
:= { hrDiskStorageTable 1 }

hrDiskStorageEntry ::= SEQUENCE { hrDiskStorageAccess INTEGER, hrDiskStorageMedia INTEGER, hrDiskStorageRemoveble TruthValue, hrDiskStorageCapacity KBytes }

**hrDiskStorageAccess OBJECT-TYPE**

SYNTAX INTEGER { readWrite(1), readOnly(2) }

MAX-ACCESS read-only

STATUS current

DESCRIPTION

An indication if this long-term storage device is readable and writable or only readable. This should reflect the media type, any write-protect mechanism, and any device configuration that affects the entire device.

:= { hrDiskStorageEntry 1 }

**hrDiskStorageMedia OBJECT-TYPE**

SYNTAX INTEGER { other(1), unknown(2), hardDisk(3), floppyDisk(4), opticalDiskROM(5), opticalDiskWORM(6), --Write Once Read Many-- opticalDiskRW(7), ramDisk(8) }

MAX-ACCESS read-only

STATUS current

DESCRIPTION

An indication of the type of media used in this long-term storage device.

:= { hrDiskStorageEntry 2 }

**hrDiskStorageRemoveble OBJECT-TYPE**

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION
Denotes whether or not the disk media may be removed from the drive.
::= { hrDiskStorageEntry 3 }

hrDiskStorageCapacity OBJECT-TYPE
SYNTAX KBytes
UNITS KBytes
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total size for this long-term storage device. If the media is removable and is currently removed, this value should be zero.
::= { hrDiskStorageEntry 4 }

hrPartitionTable OBJECT-TYPE
SYNTAX Sequence of hrPartitionEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The (conceptual) table of partitions for long-term storage devices contained by the host. In particular, partitions accessed remotely over a network are not included here.
::= { hrDevice 7 }

hrPartitionEntry OBJECT-TYPE
SYNTAX hrPartitionEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A (conceptual) entry for one partition. The hrDeviceIndex in the index represents the entry in the hrDeviceTable that corresponds to the hrPartitionEntry.

As an example of how objects in this table are named, an instance of the hrPartitionSize object might be named hrPartitionSize.3.1
INDEX { hrDeviceIndex, hrPartitionIndex }
::= { hrPartitionTable 1 }

hrPartitionEntry ::= SEQUENCE { hrPartitionIndexInteger32,
hrPartitionLabelInternationalDisplayString, hrPartitionID OCTET STRING, hrPartitionSize Bytes, hrPartitionFSIndex Integer32 }

hrPartitionIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A unique value for each partition on this long-term storage device. The value for each long-term
storage device must remain constant at least from one re-initialization of the agent to the next
re-initialization.

::= { hrPartitionEntry 1 }

hrPartitionLabel OBJECT-TYPE
SYNTAX InternationalDisplayString (SIZE (0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A textual description of this partition.

::= { hrPartitionEntry 2 }

hrPartitionID OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A descriptor which uniquely represents this partition to the responsible operating system. On some
systems, this might take on a binary representation.

::= { hrPartitionEntry 3 }

hrPartitionSize OBJECT-TYPE
SYNTAX KBytes
UNITS KBytes
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The size of this partition.

::= { hrPartitionEntry 4 }

hrPartitionFSIndex OBJECT-TYPE
SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The index of the file system mounted on this partition. If no file system is mounted on this partition,
then this value shall be zero. Note that multiple partitions may point to one file system, denoting
that that file system resides on those partitions. Multiple file systems may not reside on one
partition.

::= { hrPartitionEntry 5 }
File System Table

Registration point for popular File System types, for use with hrFSType. These are defined in the HOST-RESOURCES-TYPES module.

hrFSTypes OBJECT IDENTIFIER ::= { hrDevice 9 }

hrFSTable OBJECT-TYPE
SYNTAX Sequence of hrFSEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The (conceptual) table of file systems local to this host or remotely mounted from a file server. File systems that are in only one user's environment on a multi-user system will not be included in this table.
::= { hrDevice 8 }

hrFSEntry OBJECT-TYPE
SYNTAX hrFSEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A (conceptual) entry for one file system local to this host or remotely mounted from a file server. File systems that are in only one user's environment on a multi-user system will not be included in this table.
As an example of how objects in this table are named, an instance of the hrFSMountPoint object might be named hrFSMountPoint.3
INDEX { hrFSIndex }
::= { hrFSTable 1 }

hrFSEntry ::= SEQUENCE { hrFSIndex   Integer32, hrFSMountPoint  InternationalDisplayString, hrFSRemoteMountPointInternationalDisplayString, hrFSTypeAutonomousType, hrFSAccess INTEGER, hrFSBootableTruthValue, hrFSStorageIndexInteger32, hrFSLastFullBackupDate   DateAndTime, hrFSLastPartialBackupDate   DateAndTime }

hrFSIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A unique value for each file system local to this host. The value for each file system must remain constant at least from one re-initialization of the agent to the next re-initialization.
::= { hrFSEntry 1 }

hrFSMountPoint OBJECT-TYPE
SYNTAX InternationalDisplayString (SIZE(0..128))
MAX-ACCESS read-only
HOST-RESOURCES-MIB

STATUS current
DESCRIPTION
The path name of the root of this file system.
 ::= { hrFSEntry 2 }

hrFSRemoteMountPoint OBJECT-TYPE
SYNTAX InternationalDisplayString (SIZE(0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A description of the name and/or address of the server that this file system is mounted from. This may also include parameters such as the mount point on the remote file system. If this is not a remote file system, this string should have a length of zero.
 ::= { hrFSEntry 3 }

hrFSType OBJECT-TYPE
SYNTAX AutonomousType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The value of this object identifies the type of this file system.
 ::= { hrFSEntry 4 }

hrFSAccess OBJECT-TYPE
SYNTAX Integer { readWrite(1), readOnly(2) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
An indication if this file system is logically configured by the operating system to be readable and writable or only readable. This does not represent any local access-control policy, except one that is applied to the file system as a whole.
 ::= { hrFSEntry 5 }

hrFSBootable OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A flag indicating whether this file system is bootable.
 ::= { hrFSEntry 6 }

hrFSStorageIndex OBJECT-TYPE
SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-only
HOST-RESOURCES-MIB

STATUS current
DESCRIPTION
The index of the hrStorageEntry that represents information about this file system. If there is no such information available, then this value shall be zero. The relevant storage entry will be useful in tracking the percent usage of this file system and diagnosing errors that may occur when it runs out of space.
::= { hrFSEntry 7 }

hrFSLastFullBackupDate OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The last date at which this complete file system was copied to another storage device for backup. This information is useful for ensuring that backups are being performed regularly. If this information is not known, then this variable shall have the value corresponding to January 1, year 0000, 00:00:00.0, which is encoded as (hex) 00 00 01 01 00 00 00 00.
::= { hrFSEntry 8 }

hrFSLastPartialBackupDate OBJECT-TYPE
SYNTAX DateAndTime
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The last date at which a portion of this file system was copied to another storage device for backup. This information is useful for ensuring that backups are being performed regularly. If this information is not known, then this variable shall have the value corresponding to January 1, year 0000, 00:00:00.0, which is encoded as (hex) 00 00 01 01 00 00 00 00.
::= { hrFSEntry 9 }

Host Resources Running Software Group

The hrSWRunTable contains an entry for each distinct piece of software that is running or loaded into physical or virtual memory in preparation for running. This includes the host’s operating system, device drivers, and applications.

hrSWOSIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The value of the hrSWRunIndex for the hrSWRunEntry that represents the primary operating system running on this host. This object is useful for quickly and uniquely identifying that primary operating system.
::= { hrSWRun 1 }
hrSWRunTable OBJECT-TYPE
  SYNTAX Sequence of hrSWRunEntry
  MAX-ACCESS not-accessible
  STATUS current
  DESCRIPTION
  The (conceptual) table of software running on the host.
  ::= { hrSWRun 2 }

hrSWRunEntry OBJECT-TYPE
  SYNTAX hrSWRunEntry
  MAX-ACCESS not-accessible
  STATUS current
  DESCRIPTION
  A (conceptual) entry for one piece of software running on the host.
  Note that because the installed software table only contains information for software stored locally on this host, not every piece of running software will be found in the installed software table. This is true of software that was loaded and run from a non-local source, such as a network-mounted file system.
  As an example of how objects in this table are named, an instance of the hrSWRunName object might be named hrSWRunName.1287
  INDEX { hrSWRunIndex }
  ::= { hrSWRunTable 1 }

  HrSWRunEntry ::= SEQUENCE { hrSWRunIndex   Integer32,
                              hrSWRunNameInternationalDisplayString, hrSWRunID  ProductID,
                              hrSWRunPathInternationalDisplayString, hrSWRunParameters  InternationalDisplayString,
                              hrSWRunTypeINTEGER, hrSWRunStatus  INTEGER }

hrSWRunIndex OBJECT-TYPE
  SYNTAX Integer32 (1..2147483647)
  MAX-ACCESS read-only
  STATUS current
  DESCRIPTION
  A unique value for each piece of software running on the host. Wherever possible, this should be the system's native, unique identification number.
  ::= { hrSWRunEntry 1 }

hrSWRunName OBJECT-TYPE
  SYNTAX InternationalDisplayString (SIZE (0..64))
  MAX-ACCESS read-only
  STATUS current
  DESCRIPTION
  A textual description of this running piece of software, including the manufacturer, revision, and the name by which it is commonly known. If this software was installed locally, this should be the same string as used in the corresponding hrSWInstalledName.
::= { hrSWRunEntry 2 }

**hrSWRunID OBJECT-TYPE**
SYNTAX ProductID
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The product ID of this running piece of software.
::= { hrSWRunEntry 3 }

**hrSWRunPath OBJECT-TYPE**
SYNTAX InternationalDisplayString (SIZE(0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A description of the location on long-term storage (e.g. a disk drive) from which this software was loaded.
::= { hrSWRunEntry 4 }

**hrSWRunParameters OBJECT-TYPE**
SYNTAX InternationalDisplayString (SIZE(0..128))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A description of the parameters supplied to this software when it was initially loaded.
::= { hrSWRunEntry 5 }

**hrSWRunType OBJECT-TYPE**
SYNTAX INTEGER { unknown(1), operatingSystem(2), deviceDriver(3), application(4) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The type of this software.
::= { hrSWRunEntry 6 }

**hrSWRunStatus OBJECT-TYPE**
SYNTAX INTEGER { running(1), runnable(2), waiting for resource -- (i.e., CPU, memory, IO) notRunnable(3), loaded but waiting for event invalid(4) -- not loaded } 
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The status of this running piece of software. Setting this value to invalid(4) shall cause this software to stop running and to be unloaded. Sets to other values are not valid.
::= { hrSWRunEntry 7 }

Host Resources Running Software Performance Group

The hrSWRunPerfTable contains an entry corresponding to each entry in the hrSWRunTable.

hrSWRunPerfTable OBJECT-TYPE
SYNTAX Sequence of hrSWRunPerfEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The (conceptual) table of running software performance metrics.
::= { hrSWRunPerf 1 }

hrSWRunPerfEntry OBJECT-TYPE
SYNTAX hrSWRunPerfEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A (conceptual) entry containing software performance metrics. As an example, an instance of the hrSWRunPerfCPU object might be named hrSWRunPerfCPU.1287. This table augments information in the hrSWRunTable.
AUGMENTS { hrSWRunEntry }
::= { hrSWRunPerfTable 1 }

hrSWRunPerfEntry ::= SEQUENCE { hrSWRunPerfCPU  Integer32, hrSWRunPerfMem  KBytes }

hrSWRunPerfCPU OBJECT-TYPE
SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of centi-seconds of the total system's CPU resources consumed by this process. Note that on a multi-processor system, this value may increment by more than one centi-second in one centi-second of real (wall clock) time.
::= { hrSWRunPerfEntry 1 }

hrSWRunPerfMem OBJECT-TYPE
SYNTAX KBytes
UNITS KBytes
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total amount of real system memory allocated to this process.
Host Resources Installed Software Group

The hrSWInstalledTable contains an entry for each piece of software installed in long-term storage (e.g. a disk drive) locally on this host. Note that this does not include software loadable remotely from a network server. Different implementations may track software in varying ways. For example, while some implementations may track executable files as distinct pieces of software, other implementations may use other strategies such as keeping track of software packages (e.g., related groups of files) or keeping track of system or application patches.

This table is useful for identifying and inventoring software on a host and for diagnosing incompatibility and version mismatch problems between various pieces of hardware and software.

hrSWInstalledLastErrorChange OBJECT-TYPE

SYNTAX TimeTicks
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The value of sysUpTime when an entry in the hrSWInstalledTable was last added, renamed, or deleted. Because this table is likely to contain many entries, polling of this object allows a management station to determine when re-downloading of the table might be useful.

::= { hrSWInstalled 1 }

hrSWInstalledLastUpdateTime OBJECT-TYPE

SYNTAX TimeTicks
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The value of sysUpTime when the hrSWInstalledTable was last completely updated. Because caching of this data will be a popular implementation strategy, retrieval of this object allows a management station to obtain a guarantee that no data in this table is older than the indicated time.

::= { hrSWInstalled 2 }

hrSWInstalledTable OBJECT-TYPE

SYNTAX SEQUENCE OF HrSWInstalledEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The (conceptual) table of software installed on this host.

::= { hrSWInstalled 3 }

hrSWInstalledEntry OBJECT-TYPE

SYNTAX HrSWInstalledEntry
MAX-ACCESS not-accessible
STATUS current
HOST-RESOURCES-MIB

DESCRIPTION
A (conceptual) entry for a piece of software installed on this host. As an example of how objects in
this table are named, an instance of the hrSWInstalledName object might be named
hrSWInstalledName.96

INDEX { hrSWInstalledIndex }
 ::= { hrSWInstalledTable 1 }

hrSWInstalledEntry ::= SEQUENCE { hrSWInstalledIndex   Integer32,
hrSWInstalledNameInternationalDisplayS tring, hrSWInstalledID  ProductID,
hrSWInstalledTypeINTEGER, hrSWInstalledDateDateAndTime }

hrSWInstalledIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A unique value for each piece of software installed on the host. This value shall be in the range from
1 to the number of pieces of software installed on the host.
 ::= { hrSWInstalledEntry 1 }

hrSWInstalledName OBJECT-TYPE
SYNTAX InternationalDisplayString (SIZE (0..64))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A textual description of this installed piece of software, including the manufacturer, revision, the
name by which it is commonly known, and optionally, its serial number.
 ::= { hrSWInstalledEntry 2 }

hrSWInstalledID OBJECT-TYPE
SYNTAX ProductID
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The product ID of this installed piece of software.
 ::= { hrSWInstalledEntry 3 }

hrSWInstalledType OBJECT-TYPE
SYNTAX INTEGER { unknown(1), operatingSystem(2), deviceDriver(3), application(4) }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The type of this software.
 ::= { hrSWInstalledEntry 4 }
hrSWInstalledDate OBJECT-TYPE
   SYNTAX DateAndTime
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
   The last-modification date of this application as it would appear in a directory listing.
   If this information is not known, then this variable shall have the value corresponding to January 1,
   year 0000, 00:00:00.0, which is encoded as (hex) 00 00 01 01 00 00 00 00.
   ::= { hrSWInstalledEntry 5 }

Conformance Information

hrMIBCompliances OBJECT IDENTIFIER ::= { hrMIBAdminInfo 2 }
hrMIBGroups OBJECT IDENTIFIER ::= { hrMIBAdminInfo 3 }

Compliance Statements

hrMIBCompliance MODULE-COMPLIANCE
   STATUS current
   DESCRIPTION
   The requirements for conformance to the Host Resources MIB.
   MANDATORY-GROUPS { hrSystemGroup, hrStorageGroup, hrDeviceGroup }

OBJECT hrSystemDate
   MIN-ACCESS read-only
   DESCRIPTION
   Write access is not required.

OBJECT hrSystemInitialLoadDevice
   MIN-ACCESS read-only
   DESCRIPTION
   Write access is not required.

OBJECT hrSystemInitialLoadParameters
   MIN-ACCESS read-only
   DESCRIPTION
   Write access is not required.

OBJECT hrStorageSize
   MIN-ACCESS read-only
   DESCRIPTION
   Write access is not required.
OBJECT hrFSLastFullBackupDate
   MIN-ACCESS read-only
   DESCRIPTION Write access is not required.

OBJECT hrFSLastPartialBackupDate
   MIN-ACCESS read-only
   DESCRIPTION
   Write access is not required.

GROUP hrSWSRunGroup
   DESCRIPTION
   The Running Software Group. Implementation of this group is mandatory only when the
   hrSWSRunPerfGroup is implemented.

OBJECT hrSWSRunStatus
   MIN-ACCESS read-only
   DESCRIPTION
   Write access is not required.

GROUP hrSWSRunPerfGroup
   DESCRIPTION
   The Running Software Performance Group. Implementation of this group is at the discretion of the
   implementor.

GROUP hrSWSInstalledGroup
   DESCRIPTION
   The Installed Software Group. Implementation of this group is at the discretion of the implementor.
   ::= { hrMIBCompliances 1 }

hrSystemGroup OBJECT-GROUP
   OBJECTS { hrSystemUptime, hrSystemDate, hrSystemInitialLoadDevice,  
               hrSystemInitialLoadParameters, hrSystemNumUsers, hrSystemProcesses, hrSystemMaxProcesses
               }
   STATUS current
   DESCRIPTION
   The Host Resources System Group.
   ::= { hrMIBGroups 1 }

hrStorageGroup OBJECT-GROUP
   OBJECTS { hrMemorySize, hrStorageIndex, hrStorageType, hrStorageDescr,  
               hrStorageAllocationUnits, hrStorageSize, hrStorageUsed, hrStorageAllocationFailures  
               }
   STATUS current
   DESCRIPTION
   The Host Resources Storage Group.
   ::= { hrMIBGroups 2 }

hrDeviceGroup OBJECT-GROUP
HOST-RESOURCES-MIB

OBJECTS { hrDeviceIndex, hrDeviceType, hrDeviceDescr, hrDeviceID, hrDeviceStatus, hrDeviceErrors, hrProcessorFrmID, hrProcessorLoad, hrNetworkIfIndex, hrPrinterStatus, hrPrinterDetectedErrorState, hrDiskStorageAccess, hrDiskStorageMedia, hrDiskStorageRemoveble, hrDiskStorageCapacity, hrPartitionIndex, hrPartitionLabel, hrPartitionID, hrPartitionSize, hrPartitionFSIndex, hrFSSIndex, hrFSMountPoint, hrFSRemoteMountPoint, hrFSType, hrFSAccess, hrFSBootable, hrFSStorageIndex, hrFSLastFullBackupDate, hrFSLastPartialBackupDate }

STATUS current
DESCRIPTION
The Host Resources Device Group.
::= { hrMIBGroups 3 }

hrSWRunGroup OBJECT-GROUP

OBJECTS { hrSWOSIndex, hrSWRunIndex, hrSWRunName, hrSWRunID, hrSWRunPath, hrSWRunParameters, hrSWRunType, hrSWRunStatus }

STATUS current
DESCRIPTION
The Host Resources Running Software Group.
::= { hrMIBGroups 4 }

hrSWRunPerfGroup OBJECT-GROUP

OBJECTS { hrSWRunPerfCPU, hrSWRunPerfMem }

STATUS current
DESCRIPTION
The Host Resources Running Software Performance Group.
::= { hrMIBGroups 5 }

hrSWInstalledGroup OBJECT-GROUP

OBJECTS { hrSWInstalledLastChange, hrSWInstalledLastUpdateTime, hrSWInstalledIndex, hrSWInstalledName, hrSWInstalledID, hrSWInstalledType, hrSWInstalledDate }

STATUS current
DESCRIPTION
The Host Resources Installed Software Group.
::= { hrMIBGroups 6 }

Cisco Unified CM Release 6.x Feature Services

Table 8-3 lists the Cisco Unified Serviceability feature services in Cisco Unified Communications Manager Release 6.x. It also lists the applicable HOST-RESOURCES-MIB OIDs, clearing values, and object responses.
Table 8-3  Cisco Unified CM Release 6.x Feature Services and HOST-RESOURCES-MIB

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 6.x Feature Services</th>
<th>hrSWRunName OIDs</th>
<th>Clearing Values (Positive String)</th>
<th>Object Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Unified CM Attendant Console Server Service</td>
<td>1.3.6.1.2.1.25.4.2.1.2</td>
<td>acserver</td>
<td>Cisco CallManager Attendant Console Server Service Failure</td>
</tr>
<tr>
<td>Cisco Extended Functions Service</td>
<td></td>
<td>cef</td>
<td>Cisco Extended Functions Service Failure</td>
</tr>
<tr>
<td>Cisco Serviceability Reporter service</td>
<td></td>
<td>rmtreporter</td>
<td>Cisco Serviceability Reporter service failure</td>
</tr>
<tr>
<td>Compaq Insite Manager Service</td>
<td></td>
<td>cmascsid</td>
<td>Compaq Insite Manager Service Failure</td>
</tr>
<tr>
<td>Cisco Messaging Interface Service</td>
<td></td>
<td>cmi</td>
<td>Cisco Messaging Interface Service Failure</td>
</tr>
<tr>
<td>CSA service</td>
<td></td>
<td>ciscosecd</td>
<td>Cisco Security Agent Service Failure</td>
</tr>
<tr>
<td>CISCO-CCM-MIB activation on system</td>
<td>1.3.6.1.4.1.9.9.156</td>
<td>ccmAgt</td>
<td>CCM MIB Query Capabilities Disabled</td>
</tr>
<tr>
<td>IP Voice Media Streaming Service IF ACTIVATED</td>
<td>1.3.6.1.2.1.25.4.2.1.2</td>
<td>ipvmsd</td>
<td>IP Voice Media Streaming Service Failure</td>
</tr>
<tr>
<td>Cisco Unified CM Service If Activated</td>
<td></td>
<td>ccm</td>
<td>Cisco CallManager Service Failure</td>
</tr>
<tr>
<td>TFTP Service If Activated</td>
<td></td>
<td>ctftp</td>
<td>TFTP Service Failure</td>
</tr>
<tr>
<td>CTIManager Service If Activated</td>
<td></td>
<td>CTIManager</td>
<td>CTIManager Service Failure</td>
</tr>
<tr>
<td>Syslog Service</td>
<td></td>
<td>syslogd</td>
<td>Syslog Service Failure</td>
</tr>
<tr>
<td>DHCP Monitor Service If Activated</td>
<td></td>
<td>DHCP Monitor</td>
<td>DHCPMonitor Service Failure</td>
</tr>
<tr>
<td>Certificate Trust List Service Availability If Activated</td>
<td></td>
<td>CTLProvider</td>
<td>CTLProvider Service Failure</td>
</tr>
<tr>
<td>Certificate Authority Proxy Function Service Availability If Activated</td>
<td></td>
<td>capf</td>
<td>Certificate Authority Proxy Function Failure</td>
</tr>
<tr>
<td>DirSync Service Availability If Activated</td>
<td></td>
<td>CCMDirSync</td>
<td>CCMDirSync Service Failure</td>
</tr>
<tr>
<td>HOST-RESOURCES MIB activation on system</td>
<td>1.3.6.1.2.1.25</td>
<td>host_agent.pl</td>
<td>Host MIB Query Capabilities Disabled</td>
</tr>
</tbody>
</table>
Table 8-3  Cisco Unified CM Release 6.x Feature Services and HOST-RESOURCES-MIB (continued)

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 6.x Feature Services</th>
<th>hrSWRunName OIDs</th>
<th>Clearing Values (Positive String)</th>
<th>Object Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIB2 (RFC1213) activation on system</td>
<td>1.3.6.1.2.1</td>
<td>mib2_agent.pl</td>
<td>MIB2 MIB Query Capabilities</td>
</tr>
<tr>
<td>SYSAPPL-MIB activation on system</td>
<td>1.3.6.1.2.1.54</td>
<td>sapp_agent.pl</td>
<td>SysApp MIB Query Capabilities</td>
</tr>
</tbody>
</table>

Cisco Unified CM Release 6.x Network Services

Table 8-4 lists the Cisco Unified Serviceability network services in Cisco Unified Communications Manager Release 6.x. It also lists the applicable HOST-RESOURCES-MIB OIDs, clearing values, and object responses.

Table 8-4  Cisco Unified CM Release 6.x Network Services and HOST-RESOURCES-MIB

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 6.x Network Services</th>
<th>hrSWRunName OIDs</th>
<th>Clearing Values (Positive String)</th>
<th>Object Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco AMC Service Service</td>
<td>1.3.6.1.2.1.25.4.2.1.2</td>
<td>amc</td>
<td>Cisco AMC Service Service Failure</td>
</tr>
<tr>
<td>Cisco CAR Scheduler Service</td>
<td>1.3.6.1.2.1.54</td>
<td>sappagt</td>
<td>SysApp MIB Query Capabilities</td>
</tr>
<tr>
<td>Cisco Trace Collection Service</td>
<td>1.3.6.1.2.1.54</td>
<td>tracecollection</td>
<td>Cisco Trace Collection Service</td>
</tr>
<tr>
<td>HOST-RESOURCES MIB activation on system</td>
<td>1.3.6.1.2.1.54</td>
<td>hostagt</td>
<td>Host MIB Query Capabilities</td>
</tr>
<tr>
<td>MIB2 (RFC1213) activation on system</td>
<td>1.3.6.1.2.1.54</td>
<td>sappagt</td>
<td>MIB2 MIB Query Capabilities</td>
</tr>
<tr>
<td>SNMP activation on system</td>
<td>1.3.6.1.2.1.54</td>
<td>snmp_master_age</td>
<td>System SNMP Capabilities</td>
</tr>
<tr>
<td>SNMP activation on system</td>
<td>1.3.6.1.2.1.54</td>
<td>snmpd</td>
<td>System SNMP Capabilities</td>
</tr>
<tr>
<td>Native Agent Adaptor activation on system</td>
<td>1.3.6.1.2.1.54</td>
<td>naaagt</td>
<td>Native Adaptor Agent Capabilities</td>
</tr>
<tr>
<td>RIS Data Collector Service</td>
<td>1.3.6.1.2.1.54</td>
<td>RisDC</td>
<td>RIS Data Collector Service Failure</td>
</tr>
<tr>
<td>CDR Agent Service</td>
<td>1.3.6.1.2.1.54</td>
<td>cdragent</td>
<td>CDR Agent Service</td>
</tr>
<tr>
<td>CDR Replication Service</td>
<td>1.3.6.1.2.1.54</td>
<td>cdrrep</td>
<td>CDR Replication Service Failure</td>
</tr>
</tbody>
</table>
**Troubleshooting**

The following logs and information needs to be collected for troubleshooting purpose:

- The hostagt log files by executing the `file get activelog /platform/snmp/hostagt/` command.
- The syslog files by executing the `file get activelog /syslog/` command.
- Master SNMP Agent log files by executing the `file get activelog /platform/snmp/snmpdm/` command.
- Sequence of operations performed.
Frequently Asked Questions

Can the HOST-RESOURCES-MIB be used for process monitoring?
The HOST RESOURCES MIB retrieves the information about the processes running on the system in hrSwRunTable, by monitoring all of the processes running in the system. However, to monitor only the installed Cisco Application, use the SYSAPPL-MIB.

How is the memory usage values shown by RTMT mapped to the HOST-RESOURCES-MIB?

Table 8-5 lists the memory usage values.

<table>
<thead>
<tr>
<th>Memory Usages</th>
<th>RTMT Counter</th>
<th>HOST-RESOURCES-MIB</th>
</tr>
</thead>
<tbody>
<tr>
<td>SWAP memory Usage</td>
<td>Memory\Used Swap Kbytes</td>
<td>hrStorageUsed.2 (whose description is Virtual Memory)</td>
</tr>
<tr>
<td>Physical Memory Usage</td>
<td>Memory\Used Kbytes</td>
<td>hrStorageUsed.1 (whose description is Physical RAM)</td>
</tr>
<tr>
<td>Total memory (physical + swap) usage</td>
<td>Memory\Used VM Kbytes</td>
<td>No equivalent. Basically need to add hrStorageUsed.2 and hrStorageUsed.1</td>
</tr>
</tbody>
</table>

Since swap memory may not be used at all on lightly used servers, HR Virtual Memory may return 0. To validate HR VM is returning correctly, that value needs to be compared against RTMT Memory\Used Swap KBytes. The hrStorageUsed for physical memory shows the data in terms of used - (buffers + cache).

The hrStorageUsed for physical memory shows the data in terms of used that is buffers + cache.

The shared memory info that is exposed by the MIB is HOST-RESOURCES-MIB::hrStorageDescr.10 = STRING: /dev/shm. The virtual memory reported by HOST-RESOURCES-MIB is what is considered as swap memory by RTMT.

For HOST RESOURCES MIB, the following is used:

- P9Physical memory usage = (Physical RAM hrStorageUsed + /dev/shm hrStorageUsed) / (Physical RAM hrStorageSize)
- %VM used = (Physical RAM hrStorageUsed + /dev/shm hrStorageUsed + Virtual Memory hrStorageUsed) / (Physical RAM hrStorageSize + Virtual Memory hrStorageSize)

Why do the disk space values shown by RTMT and the HOST-RESOURCES-MIB differ?

In general the df size will not match the used and available disk space data shown. This is because of minfree percentage of reserved filesystem disk blocks. The minfree value for a Cisco Unified Communication Manager in Releases 6.x and 7.0 systems is 1%. So there will be difference of 1% between the disk space used value shown in RTMT and HOST-RESOURCES-MIB.

In RTMT, the disk space used value is shown from df reported values: [(Total Space - Available Space) /Total Space] * 100 where the Total Space includes the minfree also. For the HOST-RESOURCES-MIB, this is calculated by [hrStorageUsed/hrStorageSize] * 100 wherein the hrStorageSize does not include the minfree.
How does the Host Agent display the value in hrStorageUsed?
The hrStorageUsed for physical RAM was corrected to show the data in terms of used (buffers + cache). To check if the host agent version is correct, collect the snmp-rpm version installed in the system by using the show packages active snmp command.

How the memory capacity/usage values compare to those of HOST-RESOURCES-MIB?
In the HOST-RESOURCES-MIB the size and storage used are represented in terms of hrStorageUnits. If for that storage type, the hrStorageUnits is 4096 bytes then the hrStorageUsed or hrStorageSize value queried in the MIB value should be multiplied by 4096. For example, the show status command displays the Total Memory as 4090068K for Physical RAM.

If hrStorageUnits for physicalRAM storage type is 4096 bytes, then hrStorageSize for Physical RAM will be shown as 1022517 which is 4090078K \[ (1022517 \times 4096)/1024 = 4090068K \].

An SNMP query on hrSWRunName in HOST-RESOURCES-MIB intermittently returns incorrect entries in Windows.
The Microsoft SNMP extension agent (hostmib.dll) supports the HOST-RESOURCE-MIB. So Microsoft support may be able to help on this. If the problem is persistent then following is recommended:

- Use the tlist snmp.exe file to verify the hostmib.dll is listed in the output.
- Verify there are no error/warning messages from SNMP, in the event viewer, when SNMP service is started.
- Make sure the community string used has been configured with read privilege under snmp service properties.
- Use MSSQL-MIB (MssqlSrvInfoTable) to confirm sql process status.

Monitoring Processes
HOST-RESOURCES-MIB retrieves information about all the processes that are running on the system from hrSWRunTable. Use this MIB for monitoring all the processes that are running in the system. To monitor the only the installed Cisco application, use SYSAPPL-MIB. Disk Space and RTMT.

The used and available disk space values that are shown by HOST-RESOURCES-MIB may not match the disk space values that are shown by RTMT due to the minfree percentage of reserved file system disk blocks. Because the minfree value for Cisco Unified Communications Manager in 6.x and 7.0 systems is 1 percent, you will see a 1 percent difference between the used disk space value that is shown by RTMT and HOST-RESOURCES-MIB.

- In RTMT, the disk space used value gets shown from df reported values: \([(\text{Total Space} – \text{Available Space}) / \text{Total Space}] \times 100\) where the Total Space includes the minfree also.
- For Host Resources MIB, the disk space used value gets calculated by \([\text{hrStorageUsed}/\text{hrStorageSize}] \times 100\) where the hrStorageSize does not include the minfree.
Before you can compile IF-MIB, you need to compile the MIBs listed below in the order listed.

1. SNMPv2-SMI
2. SNMPv2-TC
3. SNMPv2-CONF
4. SNMPv2-MIB
5. IANAifType-MIB
6. RFC1155-SMI
7. RFC-1212
8. SNMPv2-SMI-v1
9. RFC-1215
10. SNMPv2-TC-v1
11. IF-MIB

Additional downloads are:
- OID File: IF-MIB.oid

The following are contained in this section:
- Revisions, page 8-106
- Definitions, page 8-107
- Objects, page 8-107
- Textual Conventions, page 8-107
- Interface Index, page 8-108
- Interfaces Table, page 8-109
- Extension to the Interface Table, page 8-115
- High Capacity Counter Objects, page 8-117
- Interface Stack Group, page 8-121
- Generic Receive Address Table, page 8-123
- Definition of Interface-Related Traps, page 8-124
- Conformance Information, page 8-125
- Compliance Statements, page 8-125
- Units of Conformance, page 8-127
- Deprecated Definitions - Objects, page 8-129
- Deprecated Definitions - Groups, page 8-133
- Deprecated Definitions - Compliance, page 8-134

Revisions

Table 8-2 lists the revisions to this MIB beginning with the latest revision.
The following definitions are imported for IF-MIB:

- MODULE-IDENTITY, OBJECT-TYPE, Counter32, Gauge32, Counter64, Integer32, TimeTicks, mib-2, NOTIFICATION-TYPE
- From SNMPv2-SMI—TEXTUAL-CONVENTION, DisplayString, PhysAddress, TruthValue, RowStatus, TimeStamp, AutonomousType, TestAndIncr
- From SNMPv2-TC—MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
- From SNMPv2-CONF—snmpTraps
- From SNMPv2-MIB—IANAifType
- From IANAIfType-MIB;

**Objects**

ifMIBObjects OBJECT IDENTIFIER ::= {ifMIB 1}

interfaces OBJECT IDENTIFIER ::= {mib-2 2}

**Textual Conventions**

OwnerString has the same semantics as used in RFC 1271.

OwnerString ::= TEXTUAL-CONVENTION

  DISPLAY-HINT 255a
  STATUS  deprecated
  DESCRIPTION

This data type is used to model an administratively assigned name of the owner of a resource. This information is taken from the NVT ASCII character set. It is suggested that this name contain one or more of the following: ASCII form of the manager station's transport address, management...
station name (e.g., domain name), network management personnel's name, location, or phone number. In some cases the agent itself will be the owner of an entry. In these cases, this string shall be set to a string starting with agent.

A value which indicates the set of services that this entity may potentially offers. The value is a sum. This sum initially takes the value zero. Then, for each layer, L, in the range 1 through 7, that this node performs transactions for, 2 raised to (L - 1) is added to the sum. For example, a node which performs only routing functions would have a value of 4 (2^3). In contrast, a node which is a host offering application services would have a value of 72 (2^4 + 2^7). Note that in the context of the Internet suite of protocols, values should be calculated accordingly:

Layer functionality:
- 1—physical (e.g., repeaters)
- 2—datalink/subnetwork (e.g., bridges)
- 3—internet (e.g., supports the IP)
- 4—end-to-end (e.g., supports the TCP)
- 7—applications (e.g., supports the SMTP)

For systems including OSI protocols, layers 5 and 6 may also be counted.

SYNTAX Octet String (SIZE(0..255))

Interface Index

The Interface Index contains the semantics of ifIndex and should be used for any objects defined in other MIB modules that need these semantics.

InterfaceIndex ::= TEXTUAL-CONVENTION

DISPLAY-HINT d
STATUS current
DESCRIPTION
A unique value, greater than zero, for each interface or interface sub-layer in the managed system. It is recommended that values are assigned contiguously starting from 1. The value for each interface sub-layer must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

SYNTAX Integer32 (1..2147483647)

InterfaceIndexOrZero ::= TEXTUAL-CONVENTION

DISPLAY-HINT d
STATUS current
DESCRIPTION
This textual convention is an extension of the InterfaceIndex convention. The latter defines a greater than zero value used to identify an interface or interface sub-layer in the managed system. This extension permits the additional value of zero. The value zero is object-specific and must therefore be defined as part of the description of any object which uses this syntax. Examples of the usage of zero might include situations where interface was unknown, or when none or all interfaces need to be referenced.

SYNTAX Integer32 (0..2147483647)

ifNumber OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of network interfaces (regardless of their current state) present on this system.
::= {interfaces 1 }

ifTableLastChange OBJECT-TYPE
SYNTAX TimeTicks
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The value of sysUpTime at the time of the last creation or deletion of an entry in the ifTable. If the number of entries has been unchanged since the last re-initialization of the local network management subsystem, then this object contains a zero value.
::= {ifMIBObjects 5}

Interfaces Table

The Interfaces table contains information on the entity's interfaces. Each sub-layer below the internetwork-layer of a network interface is considered to be an interface.

ifTable OBJECT-TYPE
SYNTAX Sequence of IfEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A list of interface entries. The number of entries is given by the value of ifNumber.
::= {interfaces 2}

ifEntry OBJECT-TYPE
SYNTAX IfEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry containing management information applicable to a particular interface.
INDEX {ifIndex}
::= {ifTable 1}
IfEntry ::= SEQUENCE {ifIndex InterfaceIndex, ifDescr DisplayString, ifType IANAifType, ifMtu Integer32, filespec Gauge32, ifPhysAddress PhysAddress, ifAdminStatus INTEGER, ifOperStatus INTEGER, ifLastChangeTimeTicks, ifInOctets Counter32, ifInUcastPkts Counter32, ifInNUcastPkts Counter32, -- deprecated ifInDiscardsCounter32, ifInErrors Counter32, ifInUnknownProtos
Counter32, ifOutOctets Counter32, ifOutUcastPkts Counter32, ifOutNUcastPkts Counter32, --
deprecated ifOutDiscards Counter32, ifOutErrors Counter32, ifOutQLen Gauge32,-- deprecated
ifSpecific OBJECT IDENTIFIER -- deprecated

**ifIndex** OBJECT-TYPE
SYNTAX InterfaceIndex
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A unique value, greater than zero, for each interface. It is recommended that values are assigned
contiguously starting from 1. The value for each interface sub-layer must remain constant at least
from one re-initialization of the entity's network management system to the next re-initialization.
::= {ifEntry 1}

**ifDescr** OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..255))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
A textual string containing information about the interface. This string should include the name of
the manufacturer, the product name and the version of the interface hardware/software.
::= {ifEntry 2}

**ifType** OBJECT-TYPE
SYNTAX IANAifType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The type of interface. Additional values for ifType are assigned by the Internet Assigned Numbers
Authority (IANA), through updating the syntax of the IANAifType textual convention.
::= {ifEntry 3}

**ifMtu** OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The size of the largest packet which can be sent/received on the interface, specified in octets. For
interfaces that are used for transmitting network datagrams, this is the size of the largest network
datagram that can be sent on the interface.
::= {ifEntry 4}

**ifSpeed** OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
An estimate of the interface current bandwidth in bits per second. For interfaces which do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth. If the bandwidth of the interface is greater than the maximum value reportable by this object then this object should report its maximum value (4,294,967,295) and ifHighSpeed must be used to report the interface speed. For a sub-layer which has no concept of bandwidth, this object should be zero.

::= {ifEntry 5}

ifPhysAddress OBJECT-TYPE
SYNTAX PhysAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The interface's address at its protocol sub-layer. For example, for an 802.x interface, this object normally contains a MAC address. The interface's media-specific MIB must define the bit and byte ordering and the format of the value of this object. For interfaces which do not have such an address (e.g., a serial line), this object should contain an octet string of zero length.

::= {ifEntry 6}

ifAdminStatus OBJECT-TYPE
SYNTAX Integer {up(1), -- ready to pass packets down(2), testing(3) -- in some test mode}
MAX-ACCESS read-write
STATUS current
DESCRIPTION
The desired state of the interface. The testing(3) state indicates that no operational packets can be passed. When a managed system initializes, all interfaces start with ifAdminStatus in the down(2) state. As a result of either explicit management action or per configuration information retained by the managed system, ifAdminStatus is then changed to either the up(1) or testing(3) states (or remains in the down(2) state).

::= {ifEntry 7}

ifOperStatus OBJECT-TYPE
SYNTAX INTEGER {up(1), -- ready to pass packets down(2), testing(3), -- in some test mode unknown(4), -- status cannot be determined -- for some reason. dormant(5), notPresent(6),-- some component is missing lowerLayerDown(7) -- down due to state of -- lower-layer interface(s)}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The current operational state of the interface. The testing(3) state indicates that no operational packets can be passed. If ifAdminStatus is down(2) then ifOperStatus should be down(2). If ifAdminStatus is changed to up(1) then ifOperStatus should change to up(1) if the interface is ready to transmit and receive network traffic; it should change to dormant(5) if the interface is waiting for
external actions (such as a serial line waiting for an incoming connection); it should remain in the
down(2) state if and only if there is a fault that prevents it from going to the up(1) state; it should
remain in the notPresent(6) state if the interface has missing (typically, hardware) components.

::= {ifEntry 8}

ifLastChange OBJECT-TYPE
SYNTAX TimeTicks
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The value of sysUpTime at the time the interface entered its current operational state. If the current
state was entered prior to the last re-initialization of the local network management subsystem, then
this object contains a zero value.

::= {ifEntry 9}

ifInOctets OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total number of octets received on the interface, including framing characters. Discontinuities
in the value of this counter can occur at re-initialization of the management system, and at other
times as indicated by the value of ifCounterDiscontinuityTime.

::= {ifEntry 10}

ifInUcastPkts OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were not addressed
to a multicast or broadcast address at this sub-layer. Discontinuities in the value of this counter can
occur at re-initialization of the management system, and at other times as indicated by the value of
ifCounterDiscontinuityTime.

::= {ifEntry 11}

ifInNUcastPkts OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to
a multicast or broadcast address at this sub-layer. Discontinuities in the value of this counter can
occur at re-initialization of the management system, and at other times as indicated by the value of
ifCounterDiscontinuityTime.
This object is deprecated in favour of ifInMulticastPkts and ifInBroadcastPkts.

::= { ifEntry 12 }

ifInDiscards OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol. One possible reason for discarding such a packet could be to free up buffer space. Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifEntry 13 }

ifInErrors OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
For packet-oriented interfaces, the number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol. For character-oriented or fixed-length interfaces, the number of inbound transmission units that contained errors preventing them from being deliverable to a higher-layer protocol.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifEntry 14 }

ifInUnknownProtos OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
For packet-oriented interfaces, the number of packets received via the interface which were discarded because of an unknown or unsupported protocol. For character-oriented or fixed-length interfaces that support protocol multiplexing the number of transmission units received via the interface which were discarded because of an unknown or unsupported protocol. For any interface that does not support protocol multiplexing, this counter will always be 0.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifEntry 15 }

ifOutOctets OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total number of octets transmitted out of the interface, including framing characters.
Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifEntry 16 }

ifOutUcastPkts OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total number of packets that higher-level protocols requested be transmitted, and which were not addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent. Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifEntry 17 }

ifOutNUcastPkts OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
The total number of packets that higher-level protocols requested be transmitted, and which were addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent. Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

This object is deprecated in favour of ifOutMulticastPkts and ifOutBroadcastPkts.

::= { ifEntry 18 }

ifOutDiscards OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space. Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifEntry 19 }

ifOutErrors OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS  current
DESCRIPTION
For packet-oriented interfaces, the number of outbound packets that could not be transmitted
because of errors. For character-oriented or fixed-length interfaces, the number of outbound
transmission units that could not be transmitted because of errors. Discontinuities in the value of
this counter can occur at re-initialization of the management system, and at other times as indicated
by the value of ifCounterDiscontinuityTime.
::= { ifEntry 20 }

ifOutQLen OBJECT-TYPE
SYNTAX  Gauge32
MAX-ACCESS  read-only
STATUS  deprecated
DESCRIPTION
The length of the output packet queue (in packets).
::= { ifEntry 21 }

ifSpecific OBJECT-TYPE
SYNTAX  OBJECT IDENTIFIER
MAX-ACCESS  read-only
STATUS  deprecated
DESCRIPTION
A reference to MIB definitions specific to the particular media being used to realize the interface.
It is recommended that this value point to an instance of a MIB object in the media-specific MIB,
i.e., that this object have the semantics associated with the InstancePointer textual convention
declared in RFC 2579. In fact, it is recommended that the media-specific MIB specify what value
ifSpecific should/can take for values of ifType. If no MIB definitions specific to the particular
media are available, the value should be set to the OBJECT IDENTIFIER { 0 0 }.
::= { ifEntry 22 }

Extension to the Interface Table

This table replaces the ifExtnsTable table.

ifXTableOBJECT-TYPE
SYNTAX  Sequence of IfXEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
A list of interface entries. The number of entries is given by the value of ifNumber. This table
contains additional objects for the interface table.
::= { ifMIBObjects 1 }

ifXEntryOBJECT-TYPE
SYNTAX  IfXEntry
Chapter 8    Industry-Standard Management Information Base

MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
An entry containing additional management information applicable to a particular interface.
AUGMENTS { ifEntry }
 ::= { ifXTable 1 }

IFXEntry ::= SEQUENCE {ifName DisplayString, ifInMulticastPkts Counter32, ifInBroadcastPkts Counter32, ifOutMulticastPkts Counter32, ifOutBroadcastPkts Counter32, ifHCInOctetsCounter64, ifHCInUcastPkts Counter64, ifHCInMulticastPkts Counter64, ifHCInBroadcastPkts Counter64, ifHCOutOctets Counter64, ifHCOutUcastPkts Counter64, ifHCOutMulticastPkts Counter64, ifHCOutBroadcastPkts Counter64, ifLinkUpDownTrapEnable INTEGER, ifHighSpeed Gauge32, ifPromiscuousMode TruthValue, ifConnectorPresent TruthValue, ifAlias DisplayString, ifCounterDiscontinuityTime TimeStamp }

ifName OBJECT-TYPE
SYNTAX DisplayString
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The textual name of the interface. The value of this object should be the name of the interface as assigned by the local device and should be suitable for use in commands entered at the device's 'console'. This might be a text name, such as 'lo0' or a simple port number, such as '1', depending on the interface naming syntax of the device. If several entries in the ifTable together represent a single interface as named by the device, then each will have the same value of ifName. Note that for an agent which responds to SNMP queries concerning an interface on some other (proxied) device, then the value of ifName for such an interface is the proxied device's local name for it.

If there is no local name, or this object is otherwise not applicable, then this object contains a zero-length string.
 ::= { ifXEntry 1 }

ifInMulticastPkts OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a multicast address at this sub-layer. For a MAC layer protocol, this includes both Group and Functional addresses. Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.
 ::= { ifXEntry 2 }

ifInBroadcastPkts OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to
a broadcast address at this sub-layer. Discontinuities in the value of this counter can occur at
re-initialization of the management system, and at other times as indicated by the value of
ifCounterDiscontinuityTime.
::= { ifXEntry 3 }

ifOutMulticastPkts OBJECT-TYPE
SYNTAX  Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total number of packets that higher-level protocols requested be transmitted, and which were
addressed to a multicast address at this sub-layer, including those that were discarded or not sent.
For a MAC layer protocol, this includes both Group and Functional addresses.
Discontinuities in the value of this counter can occur at re-initialization of the management system,
and at other times as indicated by the value of ifCounterDiscontinuityTime.
::= { ifXEntry 4 }

ifOutBroadcastPkts OBJECT-TYPE
SYNTAX  Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total number of packets that higher-level protocols requested be transmitted, and which were
addressed to a broadcast address at this sub-layer, including those that were discarded or not sent.
Discontinuities in the value of this counter can occur at re-initialization of the management system,
and at other times as indicated by the value of ifCounterDiscontinuityTime.
::= { ifXEntry 5 }

High Capacity Counter Objects

These objects are all 64bit versions of the basic ifTable counters. These objects all have the same basic
semantics as their 32-bit counterparts, however, their syntax has been extended to 64 bits.

ifHCInOctets OBJECT-TYPE
SYNTAX  Counter64
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total number of octets received on the interface, including framing characters. This object is a 64-bit version of ifInOctets. Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifXEntry 6 }

**ifHCInUcastPkts** OBJECT-TYPE
SYNTAX  Counter64
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were not addressed to a multicast or broadcast address at this sub-layer. This object is a 64-bit version of ifInUcastPkts.
Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifXEntry 7 }

**ifHCInMulticastPkts** OBJECT-TYPE
SYNTAX  Counter64
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a multicast address at this sub-layer. For a MAC layer protocol, this includes both Group and Functional addresses. This object is a 64-bit version of ifInMulticastPkts.
Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifXEntry 8 }

**ifHCInBroadcastPkts** OBJECT-TYPE
SYNTAX  Counter64
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a broadcast address at this sub-layer. This object is a 64-bit version of ifInBroadcastPkts.
Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifXEntry 9 }

**ifHCOOutOctets** OBJECT-TYPE
SYNTAX  Counter64
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The total number of octets transmitted out of the interface, including framing characters. This object is a 64-bit version of ifOutOctets.
Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifXEntry 10 }

ifHCOutUcastPkts OBJECT-TYPE
SYNTAX Counter64
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total number of packets that higher-level protocols requested be transmitted, and which were not addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent. This object is a 64-bit version of ifOutUcastPkts.
Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifXEntry 11 }

ifHCOutMulticastPkts OBJECT-TYPE
SYNTAX Counter64
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total number of packets that higher-level protocols requested be transmitted, and which were addressed to a multicast address at this sub-layer, including those that were discarded or not sent. For a MAC layer protocol, this includes both Group and Functional addresses. This object is a 64-bit version of ifOutMulticastPkts.
Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifXEntry 12 }

ifHCOutBroadcastPkts OBJECT-TYPE
SYNTAX Counter64
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The total number of packets that higher-level protocols requested be transmitted, and which were addressed to a broadcast address at this sub-layer, including those that were discarded or not sent. This object is a 64-bit version of ifOutBroadcastPkts.
Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

::= { ifXEntry 13 }

ifLinkUpDownTrapEnable OBJECT-TYPE
SYNTAX  Integer { enabled(1), disabled(2) }
MAX-ACCESS  read-write
STATUS  current
DESCRIPTION
Indicates whether linkUp/linkDown traps should be generated for this interface. By default, this object should have the value enabled(1) for interfaces which do not operate on 'top' of any other interface (as defined in the ifStackTable), and disabled(2) otherwise.
 ::= { ifXEntry 14 }

ifHighSpeed  OBJECT-TYPE
SYNTAX  Gauge32
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
An estimate of the interface's current bandwidth in units of 1,000,000 bits per second. If this object reports a value of `n' then the speed of the interface is somewhere in the range of `n-500,000' to `n+499,999'. For interfaces which do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth. For a sub-layer which has no concept of bandwidth, this object should be zero.
 ::= { ifXEntry 15 }

ifPromiscuousMode  OBJECT-TYPE
SYNTAX  TruthValue
MAX-ACCESS  read-write
STATUS  current
DESCRIPTION
This object has a value of false(2) if this interface only accepts packets/frames that are addressed to this station. This object has a value of true(1) when the station accepts all packets/frames transmitted on the media. The value true(1) is only legal on certain types of media. If legal, setting this object to a value of true(1) may require the interface to be reset before becoming effective.

The value of ifPromiscuousMode does not affect the reception of broadcast and multicast packets/frames by the interface.
 ::= { ifXEntry 16 }

ifConnectorPresent  OBJECT-TYPE
SYNTAX  TruthValue
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
This object has the value 'true(1)' if the interface sublayer has a physical connector and the value 'false(2)' otherwise.
 ::= { ifXEntry 17 }

ifAlias  OBJECT-TYPE
SYNTAX  DisplayString (SIZE(0..64))
MAX-ACCESS  read-write
STATUS  current
DESCRIPTION
This object is an alias name for the interface as specified by a network manager, and provides a non-volatile handle for the interface.

On the first instantiation of an interface, the value of ifAlias associated with that interface is the zero-length string. As and when a value is written into an instance of ifAlias through a network management set operation, then the agent must retain the supplied value in the ifAlias instance associated with the same interface for as long as that interface remains instantiated, including across all re-initializations/reboots of the network management system, including those which result in a change of the interface's ifIndex value.

An example of the value which a network manager might store in this object for a WAN interface is the (Telco's) circuit number/identifier of the interface.

Some agents may support write-access only for interfaces having particular values of ifType. An agent which supports write access to this object is required to keep the value in non-volatile storage, but it may limit the length of new values depending on how much storage is already occupied by the current values for other interfaces.

 ::= { ifXEntry 18 }

\textbf{ifCounterDiscontinuityTime OBJECT-TYPE}

SYNTAX  TimeStamp
MAX-ACCESS  read-only
STATUS  current
DESCRIPTION
The value of sysUpTime on the most recent occasion at which any one or more of this interface's counters suffered a discontinuity. The relevant counters are the specific instances associated with this interface of any Counter32 or Counter64 object contained in the ifTable or ifXTable. If no such discontinuities have occurred since the last re-initialization of the local management subsystem, then this object contains a zero value.

 ::= { ifXEntry 19 }

\section*{Interface Stack Group}

Implementation of this group is optional, but strongly recommended for all systems.

\textbf{ifStackTable OBJECT-TYPE}

SYNTAX  Sequence of IfStackEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
The table containing information on the relationships between the multiple sub-layers of network interfaces. In particular, it contains information on which sub-layers run 'on top of' which other sub-layers, where each sub-layer corresponds to a conceptual row in the ifTable. For example when the sub-layer with ifIndex value x runs over the sub-layer with ifIndex value y, then this table contains ifStackStatus.x.y=active.
For each ifIndex value, I, which identifies an active interface, there are always at least two
instantiated rows in this table associated with I. For one of these rows, I is the value of
ifStackHigherLayer; for the other, I is the value of ifStackLowerLayer. (If I is not involved in
multiplexing, then these are the only two rows associated with I.)

For example, two rows exist even for an interface which has no others stacked on top or below it:

- ifStackStatus.0.x=active
- ifStackStatus.x.0=active

::= { ifMIBObjects 2 }  

ifStackEntry OBJECT-TYPE
SYNTAX IfStackEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
Information on a particular relationship between two sub-layers, specifying that one sub-layer runs
on 'top' of the other sub-layer. Each sub-layer corresponds to a conceptual row in the ifTable.
INDEX { ifStackHigherLayer, ifStackLowerLayer }
::= { ifStackTable 1 }

IfStackEntry ::= SEQUENCE { ifStackHigherLayer InterfaceIndexOrZero, ifStackLowerLayer InterfaceIndexOrZero, ifStackStatus RowStatus }

ifStackHigherLayer OBJECT-TYPE
SYNTAX InterfaceIndexOrZero
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The value of ifIndex corresponding to the higher sub-layer of the relationship, i.e., the sub-layer
which runs on 'top' of the sub-layer identified by the corresponding instance of ifStackLowerLayer.
If there is no higher sub-layer (below the internetwork layer), then this object has the value 0.
::= { ifStackEntry 1 }

ifStackLowerLayer OBJECT-TYPE
SYNTAX InterfaceIndexOrZero
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
The value of ifIndex corresponding to the lower sub-layer of the relationship, i.e., the sub-layer
which runs 'below' the sub-layer identified by the corresponding instance of ifStackHigherLayer. If
there is no lower sub-layer, then this object has the value 0.
::= { ifStackEntry 2 }

ifStackStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION
The status of the relationship between two sub-layers. Changing the value of this object from 'active' to 'notInService' or 'destroy' will likely have consequences up and down the interface stack. Thus, write access to this object is likely to be inappropriate for some types of interfaces, and many implementations will choose not to support write-access for any type of interface.

::= { ifStackEntry 3 }

ifStackLastChange OBJECT-TYPE
SYNTAX TimeTicks
MAX-ACCESS read-only
STATUS current
DESCRIPTION
The value of sysUpTime at the time of the last change of the (whole) interface stack. A change of the interface stack is defined to be any creation, deletion, or change in value of any instance of ifStackStatus. If the interface stack has been unchanged since the last re-initialization of the local network management subsystem, then this object contains a zero value.

::= { ifMIBObjects 6 }

**Generic Receive Address Table**

This group of objects is mandatory for all types of interfaces which can receive packets/frames addressed to more than one address. This table replaces the ifExtnsRcvAddr table. The main difference is that this table makes use of the RowStatus textual convention, while ifExtnsRcvAddr did not.

ifRcvAddressTable  OBJECT-TYPE
SYNTAX  Sequence of IfRcvAddressEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
This table contains an entry for each address (broadcast, multicast, or uni-cast) for which the system will receive packets/frames on a particular interface, except as follows:

– For an interface operating in promiscuous mode, entries are only required for those addresses for which the system would receive frames were it not operating in promiscuous mode.
– For 802.5 functional addresses, only one entry is required, for the address which has the functional address bit ANDed with the bit mask of all functional addresses for which the interface will accept frames.

A system is normally able to use any unicast address which corresponds to an entry in this table as a source address.

::= { ifMIBObjects 4 }

ifRcvAddressEntry  OBJECT-TYPE
SYNTAX  IfRcvAddressEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
A list of objects identifying an address for which the system will accept packets/frames on the particular interface identified by the index value ifIndex.

INDEX  { ifIndex, ifRcvAddressAddress }
::= { ifRcvAddressTable 1 }
IfRcvAddressEntry ::= SEQUENCE { ifRcvAddressAddress  PhysAddress, ifRcvAddressStatusRowStatus, ifRcvAddressType  INTEGER }

ifRcvAddressAddress OBJECT-TYPE
SYNTAX  PhysAddress
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
An address for which the system will accept packets/frames on this entry's interface.
::= { ifRcvAddressEntry 1 }

ifRcvAddressStatus OBJECT-TYPE
SYNTAX  RowStatus
MAX-ACCESS  read-create
STATUS  current
DESCRIPTION
This object is used to create and delete rows in the ifRcvAddressTable.
::= { ifRcvAddressEntry 2 }

ifRcvAddressType OBJECT-TYPE
SYNTAX  INTEGER {other(1), volatile(2), nonVolatile(3) }
MAX-ACCESS  read-create
STATUS  current
DESCRIPTION
This object has the value nonVolatile(3) for those entries in the table which are valid and will not be deleted by the next restart of the managed system. Entries having the value volatile(2) are valid and exist, but have not been saved, so that will not exist after the next restart of the managed system. Entries having the value other(1) are valid and exist but are not classified as to whether they will continue to exist after the next restart.
DEFVAL  { volatile }
::= { ifRcvAddressEntry 3 }

Definition of Interface-Related Traps

linkDown NOTIFICATION-TYPE
OBJECTS { ifIndex, ifAdminStatus, ifOperStatus }
STATUS  current
DESCRIPTION
A linkDown trap signifies that the SNMP entity, acting in an agent role, has detected that the
ifOperStatus object for one of its communication links is about to enter the down state from some
other state (but not from the notPresent state). This other state is indicated by the included value
of ifOperStatus.
::= { snmpTraps 3 }

linkUp NOTIFICATION-TYPE
OBJECTS { ifIndex, ifAdminStatus, ifOperStatus }
STATUS current
DESCRIPTION
A linkUp trap signifies that the SNMP entity, acting in an agent role, has detected that the
ifOperStatus object for one of its communication links left the down state and transitioned into some
other state (but not into the notPresent state). This other state is indicated by the included value of
ifOperStatus.
::= { snmpTraps 4 }

Conformance Information

ifConformance OBJECT IDENTIFIER ::= { ifMIB 2 }
ifGroups OBJECT IDENTIFIER ::= { ifConformance 1 }
ifCompliances OBJECT IDENTIFIER ::= { ifConformance 2 }

Compliance Statements

ifCompliance3 MODULE-COMPLIANCE
STATUS current
DESCRIPTION
The compliance statement for SNMP entities which have network interfaces.
MODULE -- this module
MANDATORY-GROUPS { ifGeneralInformationGroup, linkUpDownNotificationsGroup }
The groups:
  – ifFixedLengthGroup
  – ifHCFixedLengthGroup
  – ifPacketGroup
  – ifHCPacketGroup
  – ifVHCPacketGroup
Mutually exclusive; at most one of these groups is implemented for a particular interface. When
any of these groups is implemented for a particular interface, then ifCounterDiscontinuityGroup
must also be implemented for that interface.
GROUP ifFixedLengthGroup
DESCRIPTION
This group is mandatory for those network interfaces which are character-oriented or transmit data in fixed-length transmission units, and for which the value of the corresponding instance of ifSpeed is less than or equal to 20,000,000 bits/second.

GROUP   ifHCFixedLengthGroup
DESCRIPTION
This group is mandatory for those network interfaces which are character-oriented or transmit data in fixed-length transmission units, and for which the value of the corresponding instance of ifSpeed is greater than 20,000,000 bits/second.

GROUP   ifPacketGroup
DESCRIPTION
This group is mandatory for those network interfaces which are packet-oriented, and for which the value of the corresponding instance of ifSpeed is less than or equal to 20,000,000 bits/second.

GROUP   ifHCPacketGroup
DESCRIPTION
This group is mandatory only for those network interfaces which are packet-oriented and for which the value of the corresponding instance of ifSpeed is greater than 20,000,000 bits/second but less than or equal to 650,000,000 bits/second.

GROUP   ifVHCPacketGroup
DESCRIPTION
This group is mandatory only for those network interfaces which are packet-oriented and for which the value of the corresponding instance of ifSpeed is greater than 650,000,000 bits/second.

GROUP   ifCounterDiscontinuityGroup
DESCRIPTION
This group is mandatory for those network interfaces that are required to maintain counters (i.e., those for which one of the ifFixedLengthGroup, ifHCFixedLengthGroup, ifPacketGroup, ifHCPacketGroup, or ifVHCPacketGroup is mandatory).

GROUP   ifRcvAddressGroup
DESCRIPTION
The applicability of this group MUST be defined by the media-specific MIBs. Media-specific MIBs must define the exact meaning, use, and semantics of the addresses in this group.

OBJECT   ifLinkUpDownTrapEnable
MIN-ACCESS   read-only
DESCRIPTION
Write access is not required.

OBJECT   ifPromiscuousMode
MIN-ACCESS   read-only
DESCRIPTION
Write access is not required.

OBJECT   ifAdminStatus
SYNTAX   INTEGER { up(1), down(2) }
MIN-ACCESS   read-only
DESCRIPTION
Write access is not required, nor is support for the value testing(3).

OBJECT ifAlias
MIN-ACCESS read-only
DESCRIPTION
Write access is not required.
::= { ifCompliances 3 }

Units of Conformance

ifGeneralInformationGroup OBJECT-GROUP
OBJECTS { ifIndex, ifDescr, ifType, ifSpeed, ifPhysAddress, ifAdminStatus, ifOperStatus, ifLastChange, ifLinkUpDownTrapEnable, ifConnectorPresent, ifHighSpeed, ifName, ifNumber, ifAlias, ifTableLastChange }
STATUS current
DESCRIPTION
A collection of objects providing information applicable to all network interfaces.
::= { ifGroups 10 }

Note
The following five groups are mutually exclusive; at most one of these groups is implemented for any interface.

- ifFixedLengthGroup OBJECT-GROUP
  OBJECTS { ifInOctets, ifOutOctets, ifInUnknownProtos, ifInErrors, ifOutErrors }
  STATUS current
  DESCRIPTION
  A collection of objects providing information specific to non-high speed (non-high speed interfaces transmit and receive at speeds less than or equal to 20,000,000 bits/second) character-oriented or fixed-length-transmission network interfaces.
  ::= { ifGroups 2 }

ifHCFixedLengthGroup OBJECT-GROUP
OBJECTS { ifHCInOctets, ifHCOutOctets, ifInOctets, ifOutOctets, ifInUnknownProtos, ifInErrors, ifOutErrors }
STATUS current
DESCRIPTION
A collection of objects providing information specific to high speed (greater than 20,000,000 bits/second) character-oriented or fixed-length-transmission network interfaces.
 ::= { ifGroups 3 }

ifPacketGroup OBJECT-GROUP
Chapter 8  Industry-Standard Management Information Base

OBJECTS { ifInOctets, ifOutOctets, ifInUnknownProtos, ifInErrors, ifOutErrors, ifMtu, ifInUcastPkts, ifInMulticastPkts, ifInBroadcastPkts, ifInDiscards, ifOutUcastPkts, ifOutMulticastPkts, ifOutBroadcastPkts, ifOutDiscards, ifPromiscuousMode }

STATUS current

DESCRIPTION
A collection of objects providing information specific to non-high speed (non-high speed interfaces transmit and receive at speeds less than or equal to 20,000,000 bits/second) packet-oriented network interfaces.

::= { ifGroups 4 }

ifHCPacketGroupOBJECT-GROUP

OBJECTS { ifHCInOctets, ifHCOutOctets, ifInOctets, ifOutOctets, ifInUnknownProtos, ifInErrors, ifOutErrors, ifMtu, ifInUcastPkts, ifInMulticastPkts, ifInBroadcastPkts, ifInDiscards, ifOutUcastPkts, ifOutMulticastPkts, ifOutBroadcastPkts, ifOutDiscards, ifPromiscuousMode }

STATUS current

DESCRIPTION
A collection of objects providing information specific to high speed (greater than 20,000,000 bits/second but less than or equal to 650,000,000 bits/second) packet-oriented network interfaces.

::= { ifGroups 5 }

ifVHCPacketGroupOBJECT-GROUP

OBJECTS { ifHCInUcastPkts, ifHCOutUcastPkts, ifHCInMulticastPkts, ifHCOutMulticastPkts, ifHCInBroadcastPkts, ifHCOutBroadcastPkts, ifHCInOctets, ifHCOutOctets, ifInOctets, ifOutOctets, ifInUnknownProtos, ifInErrors, ifOutErrors, ifMtu, ifInUcastPkts, ifInMulticastPkts, ifInBroadcastPkts, ifInDiscards, ifOutUcastPkts, ifOutMulticastPkts, ifOutBroadcastPkts, ifOutDiscards, ifPromiscuousMode }

STATUS current

DESCRIPTION
A collection of objects providing information specific to higher speed (greater than 650,000,000 bits/second) packet-oriented network interfaces.

::= { ifGroups 6 }

ifRcvAddressGroupOBJECT-GROUP

OBJECTS { ifRcvAddressStatus, ifRcvAddressType }

STATUS current

DESCRIPTION
A collection of objects providing information on the multiple addresses which an interface receives.

::= { ifGroups 7 }

ifStackGroup2OBJECT-GROUP

OBJECTS { ifStackStatus, ifStackLastChange }

STATUS current

DESCRIPTION
A collection of objects providing information on the layering of MIB-II interfaces.

::= { ifGroups 11 }
ifCounterDiscontinuityGroup  OBJECT-GROUP
  OBJECTS { ifCounterDiscontinuityTime }
  STATUS  current
  DESCRIPTION
  A collection of objects providing information specific to interface counter discontinuities.
  ::= { ifGroups 13 }

linkUpDownNotificationsGroup  NOTIFICATION-GROUP
  NOTIFICATIONS { linkUp, linkDown }
  STATUS  current
  DESCRIPTION
  The notifications which indicate specific changes in the value of ifOperStatus.
  ::= { ifGroups 14 }

Deprecated Definitions - Objects

Interface Test Table

This group of objects is optional and deprecated. However, a media-specific MIB may make implementation of this group mandatory. This table replaces the ifExtnsTestTable.

ifTestTable  OBJECT-TYPE
  SYNTAX  SEQUENCE OF IfTestEntry
  MAX-ACCESS  not-accessible
  STATUS  deprecated
  DESCRIPTION
  This table contains one entry per interface. It defines objects which allow a network manager to instruct an agent to test an interface for various faults. Tests for an interface are defined in the media-specific MIB for that interface. After invoking a test, the object ifTestResult can be read to determine the outcome. If an agent cannot perform the test, ifTestResult is set to so indicate. The object ifTestCode can be used to provide further test-specific or interface-specific (or even enterprise-specific) information concerning the outcome of the test. Only one test can be in progress on each interface at any one time. If one test is in progress when another test is invoked, the second test is rejected. Some agents may reject a test when a prior test is active on another interface.

Before starting a test, a manager-station must first obtain 'ownership' of the entry in the ifTestTable for the interface to be tested. This is accomplished with the ifTestId and ifTestStatus objects as follows:

try_again:
get (ifTestId, ifTestStatus)
while (ifTestStatus != notInUse)
/*
* Loop while a test is running or some other
* manager is configuring a test.
/*
short delay
get (ifTestId, ifTestStatus)
}
/*
* Is not being used right now -- let's compete
* to see who gets it.
*/
lock_value = ifTestId
if ( set(ifTestId = lock_value, ifTestStatus = inUse,
ifTestOwner = 'my-IP-address') == FAILURE)
/*
* Another manager got the ifTestEntry -- go
* try again
*/
goto try_again;
/*
* I have the lock
*/
set up any test parameters.
/*
* This starts the test
*/
set(ifTestType = test_to_run);
Wait for test completion by polling ifTestResult when test completes, agent sets ifTestResult agent
also sets ifTestStatus = 'notInUse' retrieve any additional test results, and ifTestId if (ifTestId ==
lock_value+1) results are valid.
A manager station first retrieves the value of the appropriate ifTestId and ifTestStatus objects,
periodically repeating the retrieval if necessary, until the value of ifTestStatus is 'notInUse'. The
manager station then tries to set the same ifTestId object to the value it just retrieved, the same
ifTestStatus object to 'inUse', and the corresponding ifTestOwner object to a value indicating itself.
If the set operation succeeds then the manager has obtained ownership of the ifTestEntry, and the
value of the ifTestId object is incremented by the agent (per the semantics of TestAndIncr). Failure
of the set operation indicates that some other manager has obtained ownership of the ifTestEntry.
Once ownership is obtained, any test parameters can be setup, and then the test is initiated by setting
ifTestType. On completion of the test, the agent sets ifTestStatus to 'notInUse'. Once this occurs,
the manager can retrieve the results. In the (rare) event that the invocation of tests by two network
managers were to overlap, then there would be a possibility that the first test's results might be
overwritten by the second test's results prior to the first results being read. This unlikely
circumstance can be detected by a network manager retrieving ifTestId at the same time as retrieving
the test results, and ensuring that the results are for the desired request.
If ifTestType is not set within an abnormally long period of time after ownership is obtained, the
agent should time-out the manager, and reset the value of the ifTestStatus object back to 'notInUse'.
It is suggested that this time-out period be 5 minutes.

In general, a management station must not retransmit a request to invoke a test for which it does not
receive a response; instead, it properly inspects an agent's MIB to determine if the invocation was
successful. Only if the invocation was unsuccessful, is the invocation request retransmitted.

Some tests may require the interface to be taken off-line in order to execute them, or may even
require the agent to reboot after completion of the test. In these circumstances, communication with
the management station invoking the test may be lost until after completion of the test. An agent is
not required to support such tests. However, if such tests are supported, then the agent should make
every effort to transmit a response to the request which invoked the test prior to losing
communication. When the agent is restored to normal service, the results of the test are properly
made available in the appropriate objects.

Note that this requires that the ifIndex value assigned to an interface must be unchanged even if the
test causes a reboot. An agent must reject any test for which it cannot, perhaps due to resource
constraints, make available at least the minimum amount of information after that test completes.

```::= { ifMIBObjects 3 }
```

**ifTestEntry OBJECT-TYPE**

SYNTAX   IfTestEntry
MAX-ACCESS not-accessible
STATUS   deprecated
DESCRIPTION
An entry containing objects for invoking tests on an interface.
AUGMENTS { ifEntry }
::= { ifTestTable 1 }
IfTestEntry ::= 
SEQUENCE { ifTestId   TestAndIncr, ifTestStatus   INTEGER, ifTestType AutonomousType,
ifTestResult   INTEGER, ifTestCode OBJECT  IDENTIFIER, ifTestOwnerOwnerString }

**ifTestId OBJECT-TYPE**

SYNTAX   TestAndIncr
MAX-ACCESS read-write
STATUS   deprecated
DESCRIPTION
This object identifies the current invocation of the interface's test.
::= { ifTestEntry 1 }

**ifTestStatus OBJECT-TYPE**

SYNTAX   INTEGER { notInUse(1), inUse(2) }
MAX-ACCESS read-write
STATUS   deprecated
DESCRIPTION
This object indicates whether or not some manager currently has the necessary 'ownership' required to invoke a test on this interface. A write to this object is only successful when it changes its value from 'notInUse(1)' to 'inUse(2)'. After completion of a test, the agent resets the value back to 'notInUse(1)'.

::= { ifTestEntry 2 }

ifTestType OBJECT-TYPE
SYNTAX AutonomousType
MAX-ACCESS read-write
STATUS deprecated
DESCRIPTION
A control variable used to start and stop operator-initiated interface tests. Most OBJECT IDENTIFIER values assigned to tests are defined elsewhere, in association with specific types of interface. However, this document assigns a value for a full-duplex loopback test, and defines the special meanings of the subject identifier:

noTest OBJECT IDENTIFIER ::= { 0 0 }

When the value noTest is written to this object, no action is taken unless a test is in progress, in which case the test is aborted. Writing any other value to this object is only valid when no test is currently in progress, in which case the indicated test is initiated.

When read, this object always returns the most recent value that ifTestType was set to. If it has not been set since the last initialization of the network management subsystem on the agent, a value of noTest is returned.

::= { ifTestEntry 3 }

ifTestResult OBJECT-TYPE
SYNTAX INTEGER { none(1), -- no test yet requested success(2), inProgress(3), notSupported(4), unAbleToRun(5), -- due to state of system aborted(6), failed(7) }
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
This object contains the result of the most recently requested test, or the value none(1) if no tests have been requested since the last reset. Note that this facility provides no provision for saving the results of one test when starting another, as could be required if used by multiple managers concurrently.

::= { ifTestEntry 4 }

ifTestCode OBJECT-TYPE
SYNTAX OBJECT IDENTIFIER
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
This object contains a code which contains more specific information on the test result, for example an error-code after a failed test. Error codes and other values this object may take are specific to the type of interface and/or test. The value may have the semantics of either the AutonomousType or InstancePointer textual conventions as defined in RFC 2579. The identifier is testCodeUnknown OBJECT IDENTIFIER ::= { 0 0 } and defined for use if no additional result code is available.
::= {ifTestEntry 5}

ifTestOwner OBJECT-TYPE
SYNTAX    OwnerString
MAX-ACCESS  read-write
STATUS     deprecated
DESCRIPTION
The entity which currently has the 'ownership' required to invoke a test on this interface.
::= {ifTestEntry 6}

** Deprecated Definitions - Groups **

ifGeneralGroup OBJECT-GROUP
OBJECTS {ifDescr, ifType, ifSpeed, ifPhysAddress, ifAdminStatus, ifOperStatus, ifLastChange, ifLinkUpDownTrapEnable, ifConnectorPresent, ifHighSpeed, ifName}
STATUS     deprecated
DESCRIPTION
A collection of objects deprecated in favour of ifGeneralInformationGroup.
::= {ifGroups 1}

ifTestGroup OBJECT-GROUP
OBJECTS {ifTestId, ifTestStatus, ifTestType, ifTestResult, ifTestCode, ifTestOwner}
STATUS     deprecated
DESCRIPTION
A collection of objects providing the ability to invoke tests on an interface.
::= {ifGroups 8}

ifStackGroup OBJECT-GROUP
OBJECTS {ifStackStatus}
STATUS     deprecated
DESCRIPTION
The previous collection of objects providing information on the layering of MIB-II interfaces.
::= {ifGroups 9}

ifOldObjectsGroup OBJECT-GROUP
OBJECTS {ifInNUcastPkts, ifOutNUcastPkts, ifOutQLen, ifSpecific}
STATUS     deprecated
DESCRIPTION
The collection of objects deprecated from the original MIB-II interfaces group.
::= {ifGroups 12}
**Deprecated Definitions - Compliance**

```plaintext
ifCompliance MODULE-COMPLIANCE

  STATUS  deprecated
  DESCRIPTION
  A compliance statement defined in a previous version of this MIB module, for SNMP entities which have network interfaces.
  MODULE  -- this module
  MANDATORY-GROUPS { ifGeneralGroup, ifStackGroup }
  GROUP   ifFixedLengthGroup
  DESCRIPTION
  This group is mandatory for all network interfaces which are character-oriented or transmit data in fixed-length transmission units.
  GROUP   ifHCFixedLengthGroup
  DESCRIPTION
  This group is mandatory only for those network interfaces which are character-oriented or transmit data in fixed-length transmission units, and for which the value of the corresponding instance of ifSpeed is greater than 20,000,000 bits/second.
  GROUP   ifPacketGroup
  DESCRIPTION
  This group is mandatory for all network interfaces which are packet-oriented.
  GROUP   ifHCpacketGroup
  DESCRIPTION
  This group is mandatory only for those network interfaces which are packet-oriented and for which the value of the corresponding instance of ifSpeed is greater than 650,000,000 bits/second.
  GROUP   ifTestGroup
  DESCRIPTION
  This group is optional. Media-specific MIBs which require interface tests are strongly encouraged to use this group for invoking tests and reporting results. A medium specific MIB which has mandatory tests may make implementation of this group mandatory.
  GROUP   ifRcvAddressGroup
  DESCRIPTION
  The applicability of this group MUST be defined by the media-specific MIBs. Media-specific MIBs must define the exact meaning, use, and semantics of the addresses in this group.

OBJECT  ifLinkUpDownTrapEnable
MIN-ACCESS  read-only
DESCRIPTION
Write access is not required.
OBJECT  ifPromiscuousMode
MIN-ACCESS  read-only
```
DESCRIPTION
Write access is not required.

OBJECT  ifStackStatus
SYNTAX  INTEGER { active(1) } -- subset of RowStatus
MIN-ACCESS  read-only

DESCRIPTION
Write access is not required, and only one of the six enumerated values for the RowStatus textual
convention need be supported, specifically: active(1).

OBJECT  ifAdminStatus
SYNTAX  INTEGER { up(1), down(2) }
MIN-ACCESS  read-only

DESCRIPTION
Write access is not required, nor is support for the value testing(3).

::= { ifCompliances 1 }

ifCompliance2 MODULE-COMPLIANCE

STATUS  deprecated

DESCRIPTION
A compliance statement defined in a previous version of this MIB module, for SNMP entities which
have network interfaces.

MODULE  -- this module

MANDATORY-GROUPS { ifGeneralInformationGroup, ifStackGroup2,
ifCounterDiscontinuityGroup }

GROUP  ifFixedLengthGroup
DESCRIPTION
This group is mandatory for all network interfaces which are character-oriented or transmit data in
fixed-length transmission units.

GROUP  ifHCFixedLengthGroup
DESCRIPTION
This group is mandatory only for those network interfaces which are character-oriented or transmit
data in fixed-length transmission units, and for which the value of the corresponding instance of
ifSpeed is greater than 20,000,000 bits/second.

GROUP  ifPacketGroup
DESCRIPTION
This group is mandatory for all network interfaces which are packet-oriented.

GROUP  ifHCPacketGroup
DESCRIPTION
This group is mandatory only for those network interfaces which are packet-oriented and for which
the value of the corresponding instance of ifSpeed is greater than 650,000,000 bits/second.

GROUP  ifRevAddressGroup
DESCRIPTION
The applicability of this group MUST be defined by the media-specific MIBs. Media-specific MIBs must define the exact meaning, use, and semantics of the addresses in this group.

OBJECT ifLinkUpDownTrapEnable
MIN-ACCESS read-only
DESCRIPTION
Write access is not required.

OBJECT ifPromiscuousMode
MIN-ACCESS read-only
DESCRIPTION
Write access is not required.

OBJECT ifStackStatus
SYNTAX INTEGER { active(1) } -- subset of RowStatus
MIN-ACCESS read-only
DESCRIPTION
Write access is not required, and only one of the six enumerated values for the RowStatus textual convention need be supported, specifically: active(1).

OBJECT ifAdminStatus
SYNTAX INTEGER { up(1), down(2) }
MIN-ACCESS read-only
DESCRIPTION
Write access is not required, nor is support for the value testing(3).

OBJECT ifAlias
MIN-ACCESS read-only
DESCRIPTION
Write access is not required.

::= { ifCompliances 2 }
Vendor-Specific Management Information Base

This chapter describes the vendor-specific Management Information Base (MIB) text documents that Cisco Unified Communications Manager (Cisco Unified CM) supports and that are used with Simple Network Management Protocol (SNMP). The chapter contains the following sections:

- Vendor-Specific Management Information Base, page 9-1
- Supported Servers in Cisco Unified CM Releases, page 9-1
- IBM MIBs, page 9-17
- Hewlett Packard MIBs, page 9-20
- Intel MIBs, page 9-26

Vendor-Specific Management Information Base

The MIBs described in this chapter exist on various Cisco Media Convergence Servers (MCS), depending on vendor and model number. To query these MIBS, you can use the standard MIB browsers provided by the vendor. Go to the following URLs:

- For HP, go to http://h18013.www1.hp.com/products/servers/management/hpsim/index.html to download HP SIM.
- For IBM, go to http://www-03.ibm.com/systems/management/director/index.html to download IBM Systems Director.

Supported Servers in Cisco Unified CM Releases

This section lists the supported server models and unsupported server models by MIB and by Cisco Unified CM Release. It contains the following subsections:

- Cisco Unified CM Release 8.5(1), page 9-2
- Cisco Unified CM Release 8.0(2), page 9-4
- Cisco Unified CM Release 8.0(1), page 9-5
- Cisco Unified CM Release 7.1(2), page 9-8
- Cisco Unified CM Release 7.1(1), page 9-9
Cisco Unified CM Release 8.5(1)

Table 9-1 provides the list of servers available in Cisco Unified CM release 8.5(1).

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 8.5(1)</th>
<th>IBM Server Models</th>
<th>HP Server Models</th>
<th>Cisco Unified Computing System</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• MCS-7816-I3-IPC1</td>
<td>• MCS-7816-H3-IPC1</td>
<td>• UCS B200 M1</td>
</tr>
<tr>
<td></td>
<td>• MCS-7816-I4-IPC1</td>
<td>• MCS-7825-H3-IPC1</td>
<td>• UCS C210 M1</td>
</tr>
<tr>
<td></td>
<td>• MCS-7816-I5-IPC1</td>
<td>• MCS-7825-H4-IPC1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7825-I3-IPC1</td>
<td>• MCS-7828-H3-IPC1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7825-I4-IPC1</td>
<td>• MCS-7835-H2-IPC1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7825-I5-IPC1</td>
<td>• MCS-7835-H2-IPC2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7828-I3-SS1</td>
<td>• MCS-7845-H2-IPC1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7828-I4-SS1</td>
<td>• MCS-7845-H2-IPC2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7828-I5-SS1</td>
<td>• DL380G6 SW only</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7835-I2-IPC1</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7835-I2-IPC2</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7835-I3-IPC1</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7845-I2-IPC1</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7845-I2-IPC2</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MCS-7845-I3-IPC1</td>
<td>—</td>
<td></td>
</tr>
</tbody>
</table>

Inapplicable MIBs in Cisco Unified CM Release 8.5(1)

IBM-SYSTEM-POWER MIB does not apply to the following IBM server models:

- MCS-7815-I2-IPC1
- MCS-7816-I3-IPC1
- MCS-7816-I4-IPC1
- MCS-7816-I5-IPC1
- MCS-7825-I2-IPC1
- MCS-7825-I3-IPC1
- MCS-7825-I4-IPC1
- MCS-7825-I5-IPC1
• MCS-7828-I3-SS1
• MCS-7828-I4-SS1
• MCS-7828-I5-SS1

IBM-SYSTEM-RAID MIB does not apply to the following IBM server models:
• MCS-7815-I2-IPC1
• MCS-7816-I3-IPC1
• MCS-7816-I4-IPC1
• MCS-7816-I5-IPC1
• MCS-7825-I4-IPC1
• MCS-7825-I5-IPC1
• MCS-7828-I4-SS1
• MCS-7828-I5-SS1
• MCS-7835-I3-IPC1
• MCS-7845-I3-IPC1

IBM-SYSTEM-STORAGE-MIB does not apply to the following IBM server models:
• MCS-7815-I2-IPC1
• MCS-7816-I3-IPC1
• MCS-7816-I4-IPC1
• MCS-7816-I5-IPC1
• MCS-7825-I2-IPC1
• MCS-7825-I3-IPC1
• MCS-7828-I3-SS1
• MCS-7835-I2-IPC1
• MCS-7845-I2-IPC2
• MCS-7845-I2-IPC1
• MCS-7845-I2-IPC2

HP CPQSCSI MIB does not apply to the following HP server model:
• MCS-7816-H3-IPC1
• MCS-7825-H3-IPC1
• MCS-7825-H4-IPC1
• MCS-7828-H3-IPC1
• MCS-7835-H2-IPC1
• MCS-7835-H2-IPC2
• MCS-7845-H2-IPC1
• MCS-7845-H2-IPC2
• DL380G6 SW only
Cisco Unified CM Release 8.0(2)

Table 9-2 provides the list of servers available in Cisco Unified CM release 8.0(2).

<table>
<thead>
<tr>
<th>Table 9-2</th>
<th>Servers Available in Cisco Unified CM Release 8.0(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cisco Unified CM Release 8.0(2)</strong></td>
<td><strong>HP Server Models</strong></td>
</tr>
<tr>
<td>IBM Server Models</td>
<td></td>
</tr>
<tr>
<td>• MCS-7815-I2-IPC1</td>
<td>• MCS-7816-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7816-I3-IPC1</td>
<td>• MCS-7825-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7816-I4-IPC1/CCX1</td>
<td>• MCS-7825-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I2-IPC1</td>
<td>• MCS-7825-H4-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I3-IPC1</td>
<td>• MCS-7828-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I4-IPC1</td>
<td>• MCS-7835-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7828-I2-SS1</td>
<td>• MCS-7835-H2-IPC2</td>
</tr>
<tr>
<td>• MCS-7828-I4-SS1</td>
<td>• DL380G6 (Single E5504 CPU)</td>
</tr>
<tr>
<td>• MCS-7835-I2-IPC1</td>
<td>• MCS-7845-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7835-I2-IPC2</td>
<td>• MCS-7845-H2-IPC2</td>
</tr>
<tr>
<td>• MCS-7835-I3-IPC1</td>
<td>• DL380G6 (Single E5540 CPU)</td>
</tr>
<tr>
<td>• MCS-7845-I2-IPC1</td>
<td></td>
</tr>
<tr>
<td>• MCS-7845-I2-IPC2</td>
<td></td>
</tr>
<tr>
<td>• MCS-7845-I3-IPC1</td>
<td></td>
</tr>
</tbody>
</table>

Inapplicable MIBs in Cisco Unified CM Release 8.0(2)

IBM-SYSTEM-POWER MIB does not apply to the following IBM server models:
- MCS-7815-I2-IPC1
- MCS-7816-I3-IPC1
- MCS-7816-I4-IPC1/CCX1
- MCS-7825-I2-IPC1
- MCS-7825-I3-IPC1
- MCS-7825-I4-IPC1
- MCS-7828-I3-SS1
- MCS-7828-I4-SS1

IBM-SYSTEM-RAID MIB does not apply to the following IBM server models:
- MCS-7815-I2-IPC1
- MCS-7816-I3-IPC1
- MCS-7816-I4-IPC1/CCX1
- MCS-7825-I4-IPC1
Supported Servers in Cisco Unified CM Releases

- MCS-7828-I4-SS1
- MCS-7835-I3-IPC1
- MCS-7845-I3-IPC1

IBM-SYSTEM-STORAGE-MIB does not apply to the following IBM server models:
- MCS-7815-I2-IPC1
- MCS-7816-I3-IPC1
- MCS-7816-I4-IPC1/CCX1
- MCS-7825-I2-IPC1
- MCS-7825-I3-IPC1
- MCS-7828-I3-SS1
- MCS-7835-I2-IPC1
- MCS-7835-I2-IPC2
- MCS-7845-I2-IPC1
- MCS-7845-I2-IPC2

HP CPQSCSI MIB does not apply to the following HP server models:
- MCS-7816-H3-IPC1
- MCS-7825-H2-IPC1
- MCS-7825-H3-IPC1
- MCS-7825-H4-IPC1
- MCS-7828-H3-IPC1
- MCS-7835-H2-IPC1
- MCS-7835-H2-IPC2
- DL380G6 (Single E5504 CPU)
- MCS-7845-H2-IPC1
- MCS-7845-H2-IPC2
- DL380G6 (Single E5540 CPU)

Cisco Unified CM Release 8.0(1)

Table 9-3 provides the list of servers available in Cisco Unified CM release 8.0(1).

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 8.0(1)</th>
<th>HP Server Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Server Models</td>
<td></td>
</tr>
<tr>
<td>- MCS-7815-I2-IPC1(^1)</td>
<td>- MCS-7816-H3-IPC1(^1)</td>
</tr>
<tr>
<td>- MCS-7816-I3-IPC1(^1)</td>
<td>- MCS-7825-H2-IPC1(^1)</td>
</tr>
<tr>
<td>- MCS-7816-I4-IPC1(^1)</td>
<td>- MCS-7825-H2-IPC2(^1)</td>
</tr>
<tr>
<td>- MCS-7825-I2-IPC1(^1)</td>
<td>- MCS-7825-H3-IPC1(^1)</td>
</tr>
</tbody>
</table>
### Supported, but note that Cisco Unified Communications Manager 6.1 and higher requires memory of minimum 2GB for MCS 7815/16/25/35, and 4GB for MCS 7845, and hard drive capacity of 72/80 GB or higher. This will result in mandatory memory and hard drive upgrades, if older supported servers are desired for use with the new software versions.

1. Supported, but note that servers running Cisco Unified Communications Manager (CallManager) 4.0 and later require a minimum of 2 GB of memory for Cisco MCS 7815, MCS 7816, MCS 7825, and MCS 7835 and 4 GB of memory for Cisco MCS 7845.

### Inapplicable MIBs in Cisco Unified CM Release 8.0(1)

**IBM-SYSTEM-POWER MIB does not apply to the following IBM server models:**
- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7815-I3-IPC1
- MCS-7816-I3-IPC1
- MCS-7816-I4-IPC1
- MCS-7825I-3.0-IPC1
- MCS-7825-I1-IPC1
- MCS-7825-I2-IPC1
- MCS-7825-I3-IPC1
- MCS-7825-I4-IPC1
- MCS-7828-I3-IPC1
- MCS-7828-I4-IPC1
- MCS-7835-I2-IPC1
- MCS-7835-I3-IPC1
- MCS-7835-I4-IPC1
- MCS-7835-I2-IPC2
- MCS-7835-I3-IPC2
- MCS-7835-I4-IPC2
- MCS-7845-I2-IPC1
- MCS-7845-I2-IPC2
- MCS-7845-I3-IPC1
- MCS-7845-I3-IPC2
- MCS-7845-I4-IPC1
- MCS-7845-I4-IPC2

**IBM-SYSTEM-RAID MIB does not apply to the following IBM server models:**
- MCS-7815-I1-IPC1

---

1. Supported, but note that Cisco Unified Communications Manager 6.1 and higher requires memory of minimum 2GB for MCS 7815/16/25/35, and 4GB for MCS 7845, and hard drive capacity of 72/80 GB or higher. This will result in mandatory memory and hard drive upgrades, if older supported servers are desired for use with the new software versions.

2. Supported, but note that servers running Cisco Unified Communications Manager (CallManager) 4.0 and later require a minimum of 2 GB of memory for Cisco MCS 7815, MCS 7816, MCS 7825, and MCS 7835 and 4 GB of memory for Cisco MCS 7845.

• MCS-7815-I2-IPC1
• MCS-7815-I3-IPC1
• MCS-7816-I3-IPC1
• MCS-7816-I4-IPC1
• MCS-7825-I4-IPC1
• MCS-7828-I4-IPC1

IBM-SYSTEM-STORAGE-MIB does not apply to the following IBM server models:
• MCS-7815-I1-IPC1
• MCS-7815-I2-IPC1
• MCS-7815-I3-IPC1
• MCS-7816-I3-IPC1
• MCS-7816-I4-IPC1
• MCS-7825I-3.0-IPC1
• MCS-7825-I1-IPC1
• MCS-7825-I2-IPC1
• MCS-7825-I3-IPC1
• MCS-7828-I3-IPC1
• MCS-7835I-3.0-IPC1
• MCS-7835-I1-IPC1
• MCS-7835-I2-IPC1
• MCS-7835-I2-IPC2
• MCS-7845I-3.0-IPC1
• MCS-7845-I1-IPC1
• MCS-7845-I2-IPC1
• MCS-7845-I2-IPC2

HP CPQSCSI MIB does not apply to the following HP server model:
• MCS-7816-H4-IPC1
• MCS-7825H-3.0-IPC1
• MCS-7825-H1-IPC1
• MCS-7825-H2-IPC1
• MCS-7825-H3-IPC1
• MCS-7825-H4-IPC1
• MCS-7828-H3-IPC1
• MCS-7835H-3.0-IPC1
• MCS-7835-H1-IPC1
• MCS-7835-H2-IPC1
• MCS-7835-H2-IPC2
• MCS-7845H-3.0-IPC1
Cisco Unified CM Release 7.1(2)

Table 9-4 provides the list of servers available in Cisco Unified CM release 7.1(2).

Table 9-4  Servers Available in Cisco Unified CM Release 7.1(2)

<table>
<thead>
<tr>
<th>IBM Server Models</th>
<th>HP Server Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>• MCS-7815-I1-IPC1</td>
<td>• MCS-7816-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7815-I2-IPC1</td>
<td>• MCS-7816-H4-IPC1/CCX1</td>
</tr>
<tr>
<td>• MCS-7815-I3-IPC1</td>
<td>• MCS-7825H-3.0-IPC1</td>
</tr>
<tr>
<td>• MCS-7816-I3-IPC1</td>
<td>• MCS-7825-H1-IPC1</td>
</tr>
<tr>
<td>• MCS-7816-I4-IPC1/CCX1</td>
<td>• MCS-7825-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7825I-3.0-IPC1</td>
<td>• MCS-7825-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I1-IPC1</td>
<td>• MCS-7825-H4-IPC1/CCE1/CCX1/ECS1/RC1</td>
</tr>
<tr>
<td>• MCS-7825-I2-IPC1</td>
<td>• MCS-7828-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I3-IPC1</td>
<td>• MCS-7835H-3.0-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I4-IPC1/CCX1/CCX1/ECS1/RC1</td>
<td>• MCS-7835-H1-IPC1</td>
</tr>
<tr>
<td>• MCS-7828-I3-IPC1</td>
<td>• MCS-7835-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7835I-3.0-IPC1</td>
<td>• MCS-7835-H2-IPC2/CCE2/CCX2/RC2/ECS2</td>
</tr>
<tr>
<td>• MCS-7835-I1-IPC1</td>
<td>• MCS-7845H-3.0-IPC1</td>
</tr>
<tr>
<td>• MCS-7835-I2-IPC1</td>
<td>• MCS-7845-H1-IPC1</td>
</tr>
<tr>
<td>• MCS-7835-I2-IPC2/CCE2/CCX2/RC2/ECS2</td>
<td>• MCS-7845-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7845I-3.0-IPC1</td>
<td>• MCS-7845-H2-IPC2/CCE2/CCX2/RC2/ECS</td>
</tr>
<tr>
<td>• MCS-7845-I1-IPC1</td>
<td>—</td>
</tr>
<tr>
<td>• MCS-7845-I2-IPC1</td>
<td>—</td>
</tr>
<tr>
<td>• MCS-7845-I2-IPC2/CCE2/CCX2/RC2/ECS</td>
<td>—</td>
</tr>
</tbody>
</table>

HP CPQSM2 MIB does not apply to the following HP server model:
• MCS-7825H-3.0-IPC1
Inapplicable MIBs in Cisco Unified CM Release 7.1(2)

IBM-SYSTEM-POWER MIB does not apply to the following IBM server models:
- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7815-I3-IPC1
- MCS-7816-I3-IPC1
- MCS-7816-I4-IPC1/CCX1
- MCS-7825I-3.0-IPC1
- MCS-7825-I1-IPC1
- MCS-7825-I2-IPC1
- MCS-7825-I3-IPC1
- MCS-7825-I4-IPC1/CCX1
- MCS-7828-I3-IPC1

IBM-SYSTEM-RAID MIB does not apply to the following IBM server models:
- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7815-I3-IPC1
- MCS-7816-I3-IPC1
- MCS-7816-I4-IPC1/CCX1

HP CPQSM2 MIB does not apply to the following HP server model:
- MCS-7825H-3.0-IPC1

Cisco Unified CM Release 7.1(1)

Table 9-5 provides the list of servers available in Cisco Unified CM release 7.1(1).

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 7.1(1)</th>
<th>HP Server Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Server Models</td>
<td>HP Server Models</td>
</tr>
<tr>
<td>MCS-7815-I1-IPC1</td>
<td>MCS-7816-H3-IPC1</td>
</tr>
<tr>
<td>MCS-7815-I2-IPC1</td>
<td>MCS-7816-H4-IPC1/CCX1</td>
</tr>
<tr>
<td>MCS-7815-I3-IPC1</td>
<td>MCS-7825H-3.0-IPC1</td>
</tr>
<tr>
<td>MCS-7816-I3-IPC1</td>
<td>MCS-7825-H1-IPC1</td>
</tr>
<tr>
<td>MCS-7816-I4-IPC1/CCX1</td>
<td>MCS-7825-H2-IPC1</td>
</tr>
<tr>
<td>MCS-7825I-3.0-IPC1</td>
<td>MCS-7825-H3-IPC1</td>
</tr>
<tr>
<td>MCS-7825-I1-IPC1</td>
<td>MCS-7825-H4-IPC1/CCX1/CCX1/ECS1/RC1</td>
</tr>
<tr>
<td>MCS-7825-I2-IPC1</td>
<td>MCS-7828-H3-IPC1</td>
</tr>
<tr>
<td>MCS-7825-I3-IPC1</td>
<td>MCS-7835H-3.0-IPC1</td>
</tr>
</tbody>
</table>
Inapplicable MIBs

IBM-SYSTEM-POWER MIB does not apply to the following IBM server models:
- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7815-I3-IPC1
- MCS-7816-I3-IPC1
- MCS-7816-I4-IPC1/CCX1
- MCS-7825I-3.0-IPC1
- MCS-7825I-3.0-IPC1
- MCS-7825-I1-IPC1
- MCS-7825-I2-IPC1
- MCS-7825-I3-IPC1
- MCS-7825-I4-IPC1/CCX1/CCX1/EC1/RC1
- MCS-7828-I3-IPC1

IBM-SYSTEM-RAID MIB does not apply to the following IBM server models:
- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7815-I3-IPC1
- MCS-7816-I3-IPC1
- MCS-7816-I4-IPC1/CCX1

HP CPQSM2 MIB does not apply to the following HP server model:
- MCS-7825H-3.0-IPC1
Cisco Unified CM Release 7.0(1)

Table 9-6 provides the list of servers available in Cisco Unified CM release 7.0(1).

Table 9-6  Servers Available in Cisco Unified CM Release 7.0(1)

<table>
<thead>
<tr>
<th>IBM Server Models</th>
<th>HP Server Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>• MCS-7815-I1-IPC1</td>
<td>• MCS-7816-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7815-I2-IPC1</td>
<td>• MCS-7825H-3.0-IPC1</td>
</tr>
<tr>
<td>• MCS-7815-I3-IPC1</td>
<td>• MCS-7825-H1-IPC1</td>
</tr>
<tr>
<td>• MCS-7816-I3-IPC1</td>
<td>• MCS-7825-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7825I-3.0-IPC1</td>
<td>• MCS-7825-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I1-IPC1</td>
<td>• MCS-7828-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I2-IPC1</td>
<td>• MCS-7835-3.0-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I3-IPC1</td>
<td>• MCS-7835-H1-IPC1</td>
</tr>
<tr>
<td>• MCS-7828-I3-IPC1</td>
<td>• MCS-7835-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7835I-3.0-IPC1</td>
<td>• MCS-7845H-3.0-IPC1</td>
</tr>
<tr>
<td>• MCS-7835-I1-IPC1</td>
<td>• MCS-7845-H1-IPC1</td>
</tr>
<tr>
<td>• MCS-7835-I2-IPC1/IPC2</td>
<td>• MCS-7845-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7845I-3.0-IPC1</td>
<td>—</td>
</tr>
<tr>
<td>• MCS-7845-I1-IPC1</td>
<td>—</td>
</tr>
<tr>
<td>• MCS-7845-I2-IPC1/IPC2</td>
<td>—</td>
</tr>
<tr>
<td>• MCS-7815-I1-IPC1</td>
<td>—</td>
</tr>
</tbody>
</table>

Note  IBM Model MCS-7835I-2.4-EVV1 is discontinued in this release.

Note  HP MCS-7825H-2.2-EVV1, MCS-7835H-2.4-EVV1, and MCS-7845H-2.4-EVV1 are discontinued in this release.

Unsupported Servers by MIB

IBM-SYSTEM-POWER MIB does not apply to the following IBM server models:

• MCS-7815-I1-IPC1
• MCS-7815-I2-IPC1
• MCS-7815-I3-IPC1
• MCS-7816-I3-IPC1
• MCS-7825I-3.0-IPC1
• MCS-7825-I1-IPC1
- MCS-7825-I2-IPC1
- MCS-7825-I3-IPC1
- MCS-7828-I3-IPC1

IBM-SYSTEM-RAID MIB does not apply to the following IBM server models:
- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7815-I3-IPC1
- MCS-7816-I3-IPC1

HP CPQSM2 MIB does not apply to the following HP server model:
- MCS-7825H-3.0-IPC1

**Cisco Unified CM Release 6.1(3)**

Table 9-7 provides the list of servers available in Cisco Unified CM release 6.1(3).

<table>
<thead>
<tr>
<th>IBM Server Models</th>
<th>HP Server Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCS-7815-I1-IPC1</td>
<td>MCS-7816-H3-IPC1</td>
</tr>
<tr>
<td>MCS-7815-I2-IPC1</td>
<td>MCS-7825-H2.2-EVV1</td>
</tr>
<tr>
<td>MCS-7815-I3-IPC1</td>
<td>MCS-7825H-3.0-IPC1</td>
</tr>
<tr>
<td>MCS-7816-I3-IPC1</td>
<td>MCS-7825-H1-IPC1</td>
</tr>
<tr>
<td>MCS-7825I-3.0-IPC1</td>
<td>MCS-7825-H2-IPC1</td>
</tr>
<tr>
<td>MCS-7828-H4-BE</td>
<td></td>
</tr>
<tr>
<td>MCS-7828-I3-IPC1</td>
<td>MCS-7828-H3-IPC1</td>
</tr>
<tr>
<td>MCS-7828-I4-BE</td>
<td>MCS-7835H-2.4-EVV1</td>
</tr>
<tr>
<td>MCS-7835I-2.4-EVV1</td>
<td>MCS-7835-H1-IPC1</td>
</tr>
<tr>
<td>MCS-7835I-3.0-IPC1</td>
<td>MCS-7835-H2-IPC1</td>
</tr>
<tr>
<td>MCS-7835I-1-IPC1</td>
<td>MCS-7845-2.4-EVV1</td>
</tr>
<tr>
<td>MCS-7835I-2-IPC1/IPC2</td>
<td>MCS-7845H-3.0-IPC1</td>
</tr>
<tr>
<td>MCS-7845I-3.0-IPC1</td>
<td>MCS-7845H-1-IPC1</td>
</tr>
<tr>
<td>MCS-7845-I1-IPC1</td>
<td>MCS-7845-H2-IPC1</td>
</tr>
<tr>
<td>MCS-7845-I2-IPC1/IPC2</td>
<td></td>
</tr>
</tbody>
</table>
Unsupported Servers by MIB

IBM-SYSTEM-POWER MIB does not apply to the following IBM server models:

- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7815-I3-IPC1
- MCS-7816-I3-IPC1
- MCS-7825I-3.0-IPC1
- MCS-7825-I1-IPC1
- MCS-7825-I2-IPC1
- MCS-7825-I3-IPC1
- MCS-7828-I3-IPC1
- MCS-7828-I4-BE

IBM-SYSTEM-RAID MIB does not apply to the following IBM server models:

- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7815-I3-IPC1
- MCS-7816-I3-IPC1

HP CPQSCSI MIB does not apply to the following HP server models:

- MCS-7816-H3-IPC1
- MCS-7825H-2.2-EVV1
- MCS-7825H-3.0-IPC1
- MCS-7825-H1-IPC1
- MCS-7825-H2-IPC1
- MCS-7825-H3-IPC1
- MCS-7828-H3-IPC1
- MCS-7828-H4-BE
- MCS-7835H-2.4-EVV1
- MCS-7835H-3.0-IPC1
- MCS-7835-H1-IPC1
- MCS-7835-H2-IPC1
- MCS-7845H-2.4-EVV1
- MCS-7845H-3.0-IPC1
- MCS-7845-H1-IPC1
- MCS-7845-H2-IPC1

HP CPQS M2 MIB does not apply to the following HP server models:

- MCS-7825H-2.2-EVV1
- MCS-7825H-3.0-IPC1
Cisco Unified CM Release 6.1

Table 9-8 provides the list of servers available in Cisco Unified CM release 6.1.

### Table 9-8  Servers Available in Cisco Unified CM Release 6.1

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 6.1</th>
<th>HP Server Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Server Models</td>
<td></td>
</tr>
<tr>
<td>• MCS-7815-I1-IPC1</td>
<td>• MCS-7816-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7815-I2-IPC1</td>
<td>• MCS-7825H-2.2-EVV1</td>
</tr>
<tr>
<td>• MCS-7815-I3-IPC1</td>
<td>• MCS-7825H-3.0-IPC1</td>
</tr>
<tr>
<td>• MCS-7816-I3-IPC1</td>
<td>• MCS-7825-H1-IPC1</td>
</tr>
<tr>
<td>• MCS-7825I-3.0-IPC1</td>
<td>• MCS-7825-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I1-IPC1</td>
<td>• MCS-7825-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I2-IPC1</td>
<td>• MCS-7828-H3-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I3-IPC1</td>
<td>• MCS-7835H-2.4-EVV1</td>
</tr>
<tr>
<td>• MCS-7828-I3-IPC1</td>
<td>• MCS-7835H-3.0-IPC1</td>
</tr>
<tr>
<td>• MCS-7835I-2.4-EVV1</td>
<td>• MCS-7835-H1-IPC1</td>
</tr>
<tr>
<td>• MCS-7835I-3.0-IPC1</td>
<td>• MCS-7835-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7835-I1-IPC1</td>
<td>• MCS-7845H-2.4-EVV1</td>
</tr>
<tr>
<td>• MCS-7835-I2-IPC1/IPC2</td>
<td>• MCS-7845H-3.0-IPC1</td>
</tr>
<tr>
<td>• MCS-7845I-3.0-IPC1</td>
<td>• MCS-7845-H1-IPC1</td>
</tr>
<tr>
<td>• MCS-7845-I1-IPC1</td>
<td>• MCS-7845-H2-IPC1</td>
</tr>
</tbody>
</table>

### Unsupported Servers by MIB

IBM-SYSTEM-POWER MIB does not apply to the following IBM server models:

- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7815-I3-IPC1
- MCS-7816-I3-IPC1
- MCS-7825I-3.0-IPC1
- MCS-7825-I1-IPC1
- MCS-7825-I2-IPC1
- MCS-7825-I3-IPC1
- MCS-7828-13-IPC1

IBM-SYSTEM-RAID MIB does not apply to the following IBM server models:

- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
Supported Servers in Cisco Unified CM Releases

- MCS-7815-I3-IPC1
- MCS-7816-I3-IPC1

HP CPQSCSI MIB does not apply to the following HP server models:
- MCS-7816-H3-IPC1
- MCS-7825H-2.2-EVV1
- MCS-7825H-3.0-IPC1
- MCS-7825-H1-IPC1
- MCS-7825-H2-IPC1
- MCS-7825-H3-IPC1
- MCS-7828-H3-IPC1
- MCS-7828-H4-BE
- MCS-7835H-2.4-EVV1
- MCS-7835H-3.0-IPC1
- MCS-7835-H1-IPC1
- MCS-7835-H2-IPC1
- MCS-7845H-2.4-EVV1
- MCS-7845H-3.0-IPC1
- MCS-7845-H1-IPC1
- MCS-7845-H2-IPC1

HP CPQSM2 MIB does not apply to the following HP server models:
- MCS-7825H-2.2-EVV1
- MCS-7825H-3.0-IPC1

Cisco Unified CM Release 6.0

Table 9-9 provides the list of servers available in Cisco Unified CM release 6.0.

Table 9-9 Servers Available in Cisco Unified CM Release 6.0

<table>
<thead>
<tr>
<th>IBM Server Models</th>
<th>HP Server Models</th>
<th>Dell Server Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCS-7815-I1-IPC1</td>
<td>MCS-7816-H3-IPC1</td>
<td>PE2950</td>
</tr>
<tr>
<td>MCS-7815-I2-IPC1</td>
<td>MCS-7825H-2.2-EVV1</td>
<td></td>
</tr>
<tr>
<td>MCS-7816-I3-IPC1</td>
<td>MCS-7825H-3.0-IPC1</td>
<td></td>
</tr>
<tr>
<td>MCS-7825I-3.0-IPC1</td>
<td>MCS-7825-H1-IPC1</td>
<td></td>
</tr>
<tr>
<td>MCS-7825-I1-IPC1</td>
<td>MCS-7825-H2-IPC1</td>
<td></td>
</tr>
<tr>
<td>MCS-7825-I2-IPC1</td>
<td>MCS-7825-H3-IPC1</td>
<td></td>
</tr>
<tr>
<td>MCS-7828-I3-IPC1</td>
<td>MCS-7828-H3-IPC1</td>
<td></td>
</tr>
<tr>
<td>MCS-7835I-2.4-EVV1</td>
<td>MCS-7835H-2.4-EVV1</td>
<td></td>
</tr>
</tbody>
</table>
Table 9-9  Servers Available in Cisco Unified CM Release 6.0 (continued)

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 6.0</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>• MCS-7835I-3.0-IPC1</td>
<td>• MCS-7835H-3.0-IPC1</td>
</tr>
<tr>
<td>• MCS-7835-I1-IPC1</td>
<td>• MCS-7835-H1-IPC1</td>
</tr>
<tr>
<td>• MCS-7835-I2-IPC1</td>
<td>• MCS-7835-H2-IPC1</td>
</tr>
<tr>
<td>• MCS-7845I-3.0-IPC1</td>
<td>• MCS-7845H-2.4-EVV1</td>
</tr>
<tr>
<td>• MCS-7845-I1-IPC1</td>
<td>• MCS-7845H-3.0-IPC1</td>
</tr>
<tr>
<td>• MCS-7845-I2-IPC1</td>
<td>• MCS-7845-H1-IPC1</td>
</tr>
<tr>
<td>• MCS-7825-I3-IPC1</td>
<td>• MCS-7845-H2-IPC1</td>
</tr>
</tbody>
</table>

Unsupported Servers by MIB

IBM-SYSTEM-POWER (UMSPower) MIB does not apply to the following IBM server models:

- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7816-I3-IPC1
- MCS-7825I-3.0-IPC1
- MCS-7825-I1-IPC1
- MCS-7825-I2-IPC1
- MCS-7825-I3-IPC1
- MCS-7828-I3-IPC1

IBM-SERVERAID MIB does not apply to the following IBM server models:

- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7825I-3.0-IPC1
- MCS-7825-I1-IPC1
- MCS-7825-I2-IPC1
- MCS-7835-I2-IPC1
- MCS-7845-I2-IPC1

IBM-SYSTEM-RAID MIB does not apply to the following IBM server models:

- MCS-7815-I1-IPC1
- MCS-7815-I2-IPC1
- MCS-7816-I3-IPC1

HP CPQSCSI MIB does not apply to the following HP server models:

- MCS-7816-H3-IPC1
- MCS-7825H-2.2-EVV1
- MCS-7825H-3.0-IPC1
- MCS-7825H-1-IPC1
• MCS-7825-H2-IPC1
• MCS-7825-H3-IPC1
• MCS-7828-H3-IPC1
• MCS-7835H-2.4-EVV1
• MCS-7835H-3.0-IPC1
• MCS-7835-H1-IPC1
• MCS-7835-H2-IPC1
• MCS-7845H-2.4-EVV1
• MCS-7845H-3.0-IPC1
• MCS-7845-H1-IPC1
• MCS-7845-H2-IPC1

HP CPQSM2 MIB does not apply to the following HP server models:
• MCS-7825H-2.2-EVV1
• MCS-7825H-3.0-IPC1

IBM MIBs

Table 9-10 provides the list of IBM MIBs.

<table>
<thead>
<tr>
<th>MIB</th>
<th>OID</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Supported for browsing only</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IBM-SYSTEM-HEALTH-MIB</td>
<td>1.3.6.1.4.1.2.6.159.1.1.30</td>
<td>Provides temperature, voltage, and fan status</td>
</tr>
<tr>
<td>IBM-SYSTEM-ASSETID-MIB</td>
<td>1.3.6.1.4.1.2.6.159.1.1.60</td>
<td>Provides hardware component asset data</td>
</tr>
<tr>
<td>IBM-SYSTEM-LMSENSOR-MIB</td>
<td>1.3.6.1.4.1.2.6.159.1.1.80</td>
<td>Provides temperature, voltage, and fan details</td>
</tr>
<tr>
<td>IBM-SYSTEM-NETWORK-MIB</td>
<td>1.3.6.1.4.1.2.6.159.1.1.110</td>
<td>Provides Network Interface Card (NIC) status</td>
</tr>
<tr>
<td>IBM-SYSTEM-MEMORY-MIB</td>
<td>1.3.6.1.4.1.2.6.159.1.1.120</td>
<td>Provides physical memory details</td>
</tr>
<tr>
<td>IBM-SYSTEM-POWER-MIB</td>
<td>1.3.6.1.4.1.2.6.159.1.1.130</td>
<td>Provides power supply details</td>
</tr>
<tr>
<td>IBM-SYSTEM-PROCESSOR-MIB</td>
<td>1.3.6.1.4.1.2.6.159.1.1.140</td>
<td>Provides CPU asset/status data</td>
</tr>
<tr>
<td><strong>Supported for system traps</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IBM-SYSTEM-TRAP</td>
<td>1.3.6.1.4.1.2.6.159.1.1.0</td>
<td>Provides temperature, voltage, fan, disk, NIC, memory, power supply, and CPU details</td>
</tr>
<tr>
<td>IBM-SERVERAID-MIB</td>
<td>1.3.6.1.4.1.2.6.167.2</td>
<td>Provides RAID status</td>
</tr>
</tbody>
</table>
Table 9-10  IBM MIBs (continued)

<table>
<thead>
<tr>
<th>MIB</th>
<th>OID</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM-SYSTEM-RAID-MIB</td>
<td>1.3.6.1.4.1.2.6.159.1.1.200.1</td>
<td>Provides RAID status</td>
</tr>
<tr>
<td>IBM-SYSTEM-STORAGE-MIB</td>
<td>1.3.6.1.4.1.2.6.159.3.1</td>
<td>Provides RAID status</td>
</tr>
</tbody>
</table>

IBM Status Messages

Table 9-11 provides the IBM hardware status messages.

Table 9-11  IBM Hardware Status Messages, MIBs and Object Names, and Object Responses

Cisco Unified CM Release 6.x

<table>
<thead>
<tr>
<th>MCS-78xx Status</th>
<th>MIBS and Object Names</th>
<th>Object Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Fan</td>
<td>IBM-SYSTEM-LMSENSOR-MIB::ibmSystemTachometerStatus (also see ibmSystemTachometerKeyIndex)</td>
<td>This is a string indicating the current status of the object. Various operational and non-operational statuses can be defined. Operational statuses are OK, Degraded and Pred Fail. Pred Fail indicates that an element may be functioning properly but predicting a failure in the near future. An example is a SMART-enabled hard drive. Non-operational statuses are Error, Starting, Stopping and Service. Service can apply during mirror-resilvering of a disk, reload of a user permissions list, or other administrative work. Not all such work is on-line, yet the managed element is neither OK nor in one of the other states. OK = Normal; Error = Critical</td>
</tr>
</tbody>
</table>
### Table 9-11  IBM Hardware Status Messages, MIBs and Objects Names, and Object Responses (continued)

<table>
<thead>
<tr>
<th>MCS-78xx Status</th>
<th>MIBS and Object Names</th>
<th>Object Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage Sensor</td>
<td>IBM-SYSTEM-LMSENSOR-MIB::ibmSystemVoltageSensorStatus (also see ibmSystemVoltageSensorKeyIndex)</td>
<td>This is a string indicating the current status of the object. Various operational and non-operational statuses can be defined. Operational statuses are OK, Degraded and Pred Fail. Pred Fail indicates that an element may be functioning properly but predicting a failure in the near future. An example is a SMART-enabled hard drive. Non-operational statuses are Error, Starting, Stopping and Service. Service can apply during mirror-resilvering of a disk, reload of a user permissions list, or other administrative work. Not all such work is on-line, yet the managed element is neither OK nor in one of the other states. OK = Normal; Error = Critical</td>
</tr>
<tr>
<td>Thermal</td>
<td>IBM-SYSTEM-LMSENSOR-MIB::ibmSystemTemperatureSensorStatus (also see ibmSystemTemperatureSensorKeyIndex)</td>
<td>The Status property is a string indicating the current status of the object. Various operational and non-operational statuses can be defined. Operational statuses are OK, Degraded and Pred Fail. Pred Fail indicates that an element may be functioning properly but predicting a failure in the near future. An example is a SMART-enabled hard drive. Non-operational statuses can also be specified. These are Error, Starting, Stopping and Service. The latter, Service, could apply during mirror-resilvering of a disk, reload of a user permissions list, or other administrative work. Not all such work is on-line, yet the managed element is neither OK nor in one of the other states. OK = Normal; Error = Critical</td>
</tr>
<tr>
<td>Network Interface Card</td>
<td>IBM-SYSTEM-NETWORK-MIB::ibmSystemLogicalNetworkAdapterStatus (also see ibmSystemLogicalNetworkAdapterKeyIndex)</td>
<td>The online status of the adapter.</td>
</tr>
<tr>
<td>Logical Drive</td>
<td>IBM-SYSTEM-TRAP-MIB::ibmSystemRaidLogicalDriveStatus (also see ibmSystemRaidLogicalDriveKeyIndex)</td>
<td>The status of the logical drive</td>
</tr>
<tr>
<td>Physical Drive</td>
<td>IBM-SYSTEM-TRAP-MIB::ibmSystemRaidDiskDriveStatus &amp; ibmSystemRaidControllerStatus (also see ibmSystemRaidDiskDriveKeyIndex &amp; ibmSystemRaidControllerKeyIndex)</td>
<td></td>
</tr>
</tbody>
</table>
Table 9-12 provides the list of Hewlett Packard MIBs.

<table>
<thead>
<tr>
<th>MIB</th>
<th>OID</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supported for browsing and system traps</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPQSTDEQ-MIB</td>
<td>1.3.6.1.4.1.232.1</td>
<td>Provides hardware component configuration data</td>
</tr>
<tr>
<td>CPQSINFO-MIB</td>
<td>1.3.6.1.4.1.232.2</td>
<td>Provides hardware component asset data</td>
</tr>
<tr>
<td>CPQIDA-MIB</td>
<td>1.3.6.1.4.1.232.3</td>
<td>Provides RAID status/events</td>
</tr>
<tr>
<td>CPQHLTH-MIB</td>
<td>1.3.6.1.4.1.232.6</td>
<td>Provides hardware components status/events</td>
</tr>
<tr>
<td>CPQSTSYS-MIB</td>
<td>1.3.6.1.4.1.232.8</td>
<td>Provides storage (disk) systems status/events</td>
</tr>
<tr>
<td>CPQSM2-MIB</td>
<td>1.3.6.1.4.1.232.9</td>
<td>Provides iLO status/events</td>
</tr>
<tr>
<td>CPQTHRSH-MIB</td>
<td>1.3.6.1.4.1.232.10</td>
<td>Provides alarm threshold management</td>
</tr>
<tr>
<td>CPQHOST-MIB</td>
<td>1.3.6.1.4.1.232.11</td>
<td>Provides operating system information</td>
</tr>
<tr>
<td>CPQIDE-MIB</td>
<td>1.3.6.1.4.1.232.14</td>
<td>Provides IDE (CD-ROM) drive status/events</td>
</tr>
<tr>
<td>CPQNIC-MIB</td>
<td>1.3.6.1.4.1.232.18</td>
<td>Provides Network Interface Card (NIC) status/events</td>
</tr>
</tbody>
</table>
### HP Status Messages

Table 9-13 lists status messages, MIBs and OIDs, MIB object names and clearing values, and object responses.

<table>
<thead>
<tr>
<th>Chapter 9</th>
<th>Vendor-Specific Management Information Base</th>
</tr>
</thead>
<tbody>
<tr>
<td>OL-22523-01</td>
<td>Hewlett Packard MIBs</td>
</tr>
</tbody>
</table>

**Table 9-13 HP Hardware Status Messages, MIBs and OIDs, MIB Object Names and Clearing Values, and Object Responses**

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 6.x</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCS-78xx Status</td>
</tr>
<tr>
<td>Logical Drive</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
Table 9-13  
HP Hardware Status Messages, MIBs and OIDs, MIB Object Names and Clearing Values, and 
Object Responses (continued)

<table>
<thead>
<tr>
<th>MCS-78xx Status</th>
<th>MIB and OID</th>
<th>MIB Object Name and Clearing Value</th>
<th>Object Response</th>
</tr>
</thead>
</table>
| Physical Drive    | CPQIDA-MIB 1.3.6.1.4.1.232.3.2.5.1.1.6 | cpqDaPhyDrvStatus Clearing Value = 2 | • The following values are valid for the physical drive status:  
  • other (1) Indicates that the instrument agent does not recognize the drive. You may need to upgrade your instrument agent and/or driver software.  
  • ok (2) Indicates the drive is functioning properly.  
  • failed (3) Indicates that the drive is no longer operating and should be replaced.  
  • predictiveFailure(4) Indicates that the drive has a predictive failure error and should be replaced. |
| System Fan        | CPQHLTH-MIB 1.3.6.1.4.1.232.6.2.6.4  | cpqHeThermalSystem FanStatus Clearing Value = 2 | This value will be one of the following:  
  • other(1) Fan status detection is not supported by this system or driver.  
  • ok(2) The fan is operating properly.  
  • degraded(2) A redundant fan is not operating properly.  
  • failed(4) A non-redundant fan is not operating properly. |
| CPU Fan           | CPQHLTH-MIB 1.3.6.1.4.1.232.6.2.6.5  | cpqHeThermalCpuFan Status Clearing Value = 2 | This value will be one of the following:  
  • other(1) Fan status detection is not supported by this system or driver.  
  • ok(2) The fan is operating properly.  
  • degraded(2) A redundant fan is not operating properly.  
  • failed(4) A non-redundant fan is not operating properly. |
### Chapter 9      Vendor-Specific Management Information Base

### Hewlett Packard MIBs

#### Table 9-13   HP Hardware Status Messages, MIBs and OIDs, MIB Object Names and Clearing Values, and Object Responses (continued)

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 6.x</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MIB-78xx Status</strong></td>
</tr>
</tbody>
</table>
| Network Interface Card (NIC) | CPQNIC-MIB<br>1.3.6.1.4.1.232.18.2.3.1.1.13 | cpqNicIfPhysAdapter State<br>Clearing Value = 2 and 3 | The following values are valid—  
  • unknown(1) The instrument agent was not able to determine the status of the adapter. The instrument agent may need to be upgraded.  
  • ok(2) The physical adapter is operating properly.  
  • generalFailure(3) The physical adapter has failed.  
  • linkFailure(4) The physical adapter has lost link. Check the cable connections to this adapter. |
| Thermal                     | CPQHLTH-MIB<br>1.3.6.1.4.1.232.6.2.6.1 | cpqHeThermalCondition<br>Clearing Value = 2 | This value will be one of the following:  
  • other(1) Temperature could not be determined.  
  • ok(2) The temperature sensor is within normal operating range.  
  • degraded(3) The temperature sensor is outside of normal operating range.  
  • failed(4) The temperature sensor detects a condition that could permanently damage the system. |

**Note**  
The system automatically shuts down if the failed (4) condition occurs, so it is unlikely that 4 will ever be returned by the agent. If the cpqHeThermalDegradedAction is set to shut down (3), the system will shut down if the condition occurs.
### Table 9-13HP Hardware Status Messages, MIBs and OIDs, MIB Object Names and Clearing Values, and Object Responses (continued)

**Cisco Unified CM Release 6.x**

<table>
<thead>
<tr>
<th>MCS-78xx Status</th>
<th>MIB and OID</th>
<th>MIB Object Name and Clearing Value</th>
<th>Object Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power Supply¹</td>
<td>CPQHLTH-MIB 1.3.6.1.4.1.232.6.2.9.3.1.5</td>
<td>cpqHeFltTolPowerSupplyStatus&lt;br&gt;Clearing Value = 1</td>
<td>This value will be one of the following:&lt;br&gt;• other(1) The status could not be determined or not present.&lt;br&gt;• ok(2) The power supply is operating normally.&lt;br&gt;• degraded(3) A temperature sensor, fan or other power supply component is outside of normal operating range.&lt;br&gt;• failed(4) A power supply component detects a condition that could permanently damage the system.</td>
</tr>
<tr>
<td>NIC Errors</td>
<td>CPQNIC-MIB 1.3.6.1.4.1.232.18.2.3.1.1.16</td>
<td>cpqNicIfPhysAdapter&lt;br&gt;GoodTransmits&lt;br&gt;Clearing Value = &lt;0.5% for 1 hour</td>
<td>Interface is experiencing excessive errors</td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.232.18.2.3.1.1.18</td>
<td>cpqNicIfPhysAdapter&lt;br&gt;BadTransmits</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.232.18.2.3.1.1.17</td>
<td>cpqNicIfPhysAdapter&lt;br&gt;GoodReceives</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.232.18.2.3.1.1.19</td>
<td>cpqNicIfPhysAdapter&lt;br&gt;BadReceives</td>
<td></td>
</tr>
<tr>
<td>NIC Utilization</td>
<td>CPQNIC-MIB 1.3.6.1.4.1.232.18.2.3.1.1.16</td>
<td>cpqNicIfPhysAdapter&lt;br&gt;GoodTransmits&lt;br&gt;Clearing Value = &lt;50% for 1 hour</td>
<td>Interface is experiencing High Utilization</td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.232.18.2.3.1.1.18</td>
<td>cpqNicIfPhysAdapter&lt;br&gt;BadTransmits</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.232.18.2.3.1.1.17</td>
<td>cpqNicIfPhysAdapter&lt;br&gt;GoodReceives</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.232.18.2.3.1.1.19</td>
<td>cpqNicIfPhysAdapter&lt;br&gt;BadReceives</td>
<td></td>
</tr>
</tbody>
</table>
### Hewlett Packard MIBs

**Memory Module Trap**

<table>
<thead>
<tr>
<th>MCS-78xx Status</th>
<th>MIB and OID</th>
<th>MIB Object Name and Clearing Value</th>
<th>Object Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory Module Trap</td>
<td>1.3.6.1.4.1.232.6.3</td>
<td>cpqHe4CorrMem ReplaceMemModule</td>
<td>A correctable memory log entry indicates a memory module needs to be replaced. The errors have been corrected, but the memory module should be replaced. The error information is reported in the variable cpqHeCorrMemErrDesc</td>
</tr>
<tr>
<td></td>
<td></td>
<td>See CPQHOST-MIB for information on the following trap variables:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• sysName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• cpqHoTrapFlags</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• cpqHeResMemBoardIndex</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• cpqHeResMemModuleIndex</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• cpqHeResMemModuleSparePartNo</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• cpqSiMemModuleSize</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• cpqSiServerSystemId</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Trap number is 6056 which replaces 6029.</td>
<td></td>
</tr>
</tbody>
</table>

**78x5-H Insite Manager Service**

<table>
<thead>
<tr>
<th>78x5-H Insite Manager Service</th>
<th>MIB and OID</th>
<th>MIB Object Name and Clearing Value</th>
<th>Object Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host-RESOURCES-MIB</td>
<td>1.3.6.1.2.1.25.4.2.1.2</td>
<td>cmaeventd</td>
<td>Compaq Insite Manager Service Failure</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmafcad</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmamealhtd</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmahostd</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Positive String ID</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>forcmaidad</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmaidades</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmanicd</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmapierd</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmaperfd</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmasm2d</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmastdeqd</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmathreshd</td>
<td></td>
</tr>
</tbody>
</table>

1. Unavailable for MCS-7825H
Intel MIBs

Table 9-14 lists Intel MIBs, OID, and functions. See “Intel Status Messages” section on page 9-26 for descriptions of messages.

<table>
<thead>
<tr>
<th>MIB</th>
<th>OID</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supported for browsing and system traps</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INTEL-SERVER-BASEBOARD6</td>
<td>1.3.6.1.4.1.343.2.10.3.6.200</td>
<td>Denotes the power group and describes voltage probes, status, and readings</td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.343.2.10.3.6.300</td>
<td>Denotes the thermal group and describes cooling devices, fans, and temperature probes</td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.343.2.10.3.6.10</td>
<td>Denotes the instances of cooling devices</td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.343.2.10.3.6.20</td>
<td>Denotes the status, reading, and threshold for every cooling device and fan</td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.343.2.10.3.6.30</td>
<td>Denotes the instances of temperature probes</td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.343.2.10.3.6.40</td>
<td>Denotes the status, reading, thresholds for every temperature probe</td>
</tr>
<tr>
<td></td>
<td>1.3.6.1.4.1.343.2.10.3.6.100</td>
<td>Denotes the events group and describes power, thermal, and system events</td>
</tr>
</tbody>
</table>

Intel Status Messages

Table 9-15 lists status messages, MIBs and OIDs, MIB object names and clearing values, and object responses.

<table>
<thead>
<tr>
<th>Cisco Unified CM Release 7.x</th>
<th>MCS-78xx Status</th>
<th>MIBS and Object Names</th>
<th>Object Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Power</td>
<td>INTEL-SERVER-BASEBOARD6::powerEvents</td>
<td></td>
</tr>
<tr>
<td></td>
<td>System</td>
<td>INTEL-SERVER-BASEBOARD6::systemEvents</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Thermal</td>
<td>INTEL-SERVER-BASEBOARD6::thermalEvents</td>
<td></td>
</tr>
</tbody>
</table>
## Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>%IOwait</td>
<td>3-15</td>
</tr>
</tbody>
</table>

## A

### alarms

- alert-level severity 6-61
- critical-level severity 6-79
- debug-level severity 6-405
- emergency-level severity 6-52
- error-level severity 6-92
- informational-level severity 6-326
- notice-level severity 6-303
- overview 6-2
- pre-configured callmanager 6-32
- pre-configured system 6-19
- removed in Cisco Unified CM Release 8.0(1) 6-406
- warning-level severity 6-202

### alert notification

- configuring parameters for counter (table) 5-3
- alerts as syslog messages and traps 3-26

## B

### backup and restore

3-26

## C

### ccmProcess and cpu usage

3-20

### CDRs and CMRs

3-36

### Cisco Analog Access

perfmon object and counters 5-5

### Cisco Annunciator Device

perfmon object and counters 5-5

### Cisco CallManager

perfmon object and counters 5-5

### Cisco CallManager External Call Control

perfmon object and counters 5-13

### Cisco CallManager SAF

perfmon object and counters 5-14

### Cisco CallManager System Performance

perfmon object and counters 5-15

### CISCO-CCM_MIB

- Cisco Unified CM group mapping table 7-25
- Cisco Unified CM product type table 7-32
- Cisco Unified CM region pair table 7-27
- Cisco Unified CM region table 7-26
- Cisco Unified CM table 7-22
- Cisco Unified CM time zone table 7-29
- definitions 7-14
- device pool table 7-30
- objects 7-20
- phone failed table 7-40
- phone status update table 7-43
- phone table 7-34
- textual conventions 7-14

### CISCO-CCM-MIB alarms

7-67

---

Cisco Unified CM alarm enable 7-67

gateway alarm enable 7-69

malicious call alarm enable 7-69

phone failed config objects 7-67

phone status update config objects 7-68

all scalar objects 7-53

Cisco Unified CM alarms to enable 7-106
Index

Cisco Unified CM managed services and snmp traps 7-106
compliance statements 7-96
cti device directory number table 7-66
cti device table 7-62
dynamic table objects 7-109
enhanced phone extension table with combination index 7-45
gateway table 7-47
gateway trunk table 7-52
h323 device table 7-76
media device table 7-59
mib conformance statements 7-96
notifications and alarms 7-70
notification types 7-92
quality report alarm configuration information 7-89
sip device table 7-89
static object tables 7-110
traps to monitor 7-107
voice mail device table 7-85
voice mail directory number table 7-88
Cisco CTIManager
permon object and counters 5-17
Cisco Dual-Mode Mobility
permon object and counters 5-17
Cisco Extension Mobility
permon object and counters 5-19
Cisco Gatekeeper
permon object and counters 5-20
Cisco H.323
permon object and counters 5-20
Cisco Hunt Lists
permon object and counters 5-21
Cisco HW Conference Bridge Device
permon object and counters 5-22
Cisco IME Server 5-22
Cisco IP Manager Assistant
permon object and counters 5-22
Cisco Lines
permon object and counters 5-23
Cisco Locations
permon object and counters 5-23
CiscoLog
overview 6-2, 6-3, 6-4, 6-5, 6-6, 6-8, 6-10, 6-11, 6-13, 6-14, 6-17
Cisco Media Streaming Application
permon object and counters 5-24
Cisco Messaging Interface
permon object and counters 5-27
Cisco MGCP BRI Device
permon object and counters 5-28
Cisco MGCP FXO Device
permon object and counters 5-29
Cisco MGCP FXS Device
permon object and counters 5-29
Cisco MGCP Gateways
permon object and counters 5-30
Cisco MGCP PRI Device
permon object and counters 5-30
Cisco MGCP T1CAS Device
permon object and counters 5-31
Cisco MOH Device
permon object and counters 5-32, 5-33
Cisco MTP Device
permon object and counters 5-34
Cisco Phones
permon object and counters 5-34
Cisco Presence Feature
permon object and counters 5-34
Cisco QSIG Feature
permon object and counters 5-35
Cisco security agent support 3-32
Cisco Signaling Performance
permon object and counters 5-35
Cisco SIP
permon object and counters 5-35, 5-36
Cisco SIP Normalization
perfmon object and counters 5-36
Cisco SIP Stack
perfmon object and counters 5-43
Cisco SW Conf Bridge Device
perfmon object and counters 5-52
Cisco TFTP Server
perfmon object and counters 5-53
Cisco Tomcat Connector
perfmon object and counters 5-59
Cisco Transcode Device
perfmon object and counters 5-56
Cisco Unified CM Group Table
7-21
Cisco Unified Reporting 3-34
Cisco Video Conference Bridge
perfmon object and counters 5-57
Cisco WebDialer
perfmon object and counters 5-58
Cisco WSM Connector
perfmon object and counters 5-58
CLI 3-28
clock synchronization 6-4
code yellow 3-21
community strings 4-4
counters
alert notification parameters (table) 5-3
cpu usage 3-13
critical service 3-24
CTI
Cisco CTIManager
perfmon object and counters 5-17

Database Change Notification Client
perfmon object and counters 5-62
Database Change Notification Server
perfmon object and counters 5-63
Database Change Notification Subscription
perfmon object and counters 5-64

Database Local DSN
perfmon object and counters 5-64
Database Replication
Database Replication Does Not Occur When Connectivity Is Restored on Lost Node 3-41
Database Tables Out of Sync Do Not Trigger Alert 3-42
Replication Fails Between the Publisher and the Subscriber 3-39
Resetting Database Replication When Reverting to an Older Product Release 3-43
database replication 3-20
Database Replication Does Not Occur When Connectivity Is Restored on Lost Node 3-41
DB User Host Information Counters
perfmon object and counters 5-64
disk name mapping 3-18
disk usage 3-17

E

Enterprise Replication DBSpace Monitors
perfmon object and counters 5-64
Enterprise Replication Perfmon Counters
perfmon object and counters 5-65

F

format 6-2

G
general install/upgrade 3-33
generate report fields 2-22

H

hardware migration 3-31
HEADER field 6-10
historical information download 3-37
HOST field  6-6

-------------------------------

I

IMM

cisco unified CM server alerts 5-80
cisco unified CM server objects 5-78
IME server alerts  5-78
IME server objects  5-74
IMM performance monitoring  5-74
IME Client

cisco unified CM server objects 5-79
IME Client Instance

cisco unified CM server objects 5-80
IME Configuration Manager

IME server objects  5-74
IME Server

IME server objects  5-74
IME Server System Performance

IME server objects  5-77
informs

overview  4-5
IP

perfmon object and counters  5-65

-------------------------------

L

locked-down system  3-32
log file and syslog outputs  6-3

-------------------------------

M

Memory

perfmon object and counters  5-66
MESSAGE field  6-17
message format  6-5
message length  6-6
mibs
cisco-ccm-mib  7-1

MSGNAME field  6-13
multipart messages  6-4

-------------------------------

N

native hardware OOB management  3-37
Network Interface

perfmon object and counters  5-67
new and changed

Cisco Unified CM Release 8.0(1) 2-1, 2-25
Number of Replicates

perfmon object and counters  5-68

-------------------------------

O

object and counters

Database Change Notification Client 5-62
onboard agents  3-36
overview

alarms  6-2
CAR  1-7
CDRs and CMRs  1-7
CiscoLog messages  6-2
Cisco Unified CM  1-1
Cisco Unified Reporting  1-5
Cisco Unified Serviceability 1-4, 1-5
informs  4-5
managed services  1-3
MIBs  1-8
RTMT  1-6, 5-1
SNMP  4-3
support deployment models  1-2
trace collection  1-5
traps  4-5

-------------------------------

P

Partition

perfmon object and counters  5-69
perfmon

object and counters

Cisco Analog Access 5-5
Cisco Annunciator Device 5-5
Cisco CallManager 5-5
Cisco CallManager System Performance 5-15
Cisco CTIManager 5-17
Cisco Dual-Mode Mobility 5-17
Cisco Extension Mobility 5-19
Cisco Gatekeeper 5-20
Cisco H.323 5-20
Cisco Hunt Lists 5-21
Cisco HW Conference Bridge Device 5-22
Cisco IP Manager Assistant 5-22
Cisco Lines 5-23
Cisco Locations 5-23
Cisco Media Streaming Application 5-24
Cisco Messaging Interface 5-27
Cisco MGCP FXO Device 5-29
Cisco MGCP FXS Device 5-29
Cisco MGCP Gateways 5-30
Cisco MGCP PRI Device 5-30
Cisco MGCP T1CAS Device 5-31
Cisco MobilityManager 5-32
Cisco MOH Device 5-33
Cisco MTP Device 5-34
Cisco Phones 5-34
Cisco Presence Feature 5-34
Cisco QSIG Feature 5-35
Cisco Signaling Performance 5-35
Cisco SIP 5-35, 5-36
Cisco SIP Normalization 5-36
Cisco SIP Stack 5-43
Cisco SIP Station 5-51
Cisco SW Conf Bridge Device 5-52
Cisco TFTP Server 5-53
Cisco Tomcat Connector 5-59
Cisco Transcode Device 5-56
Cisco Video Conference Bridge 5-57

Cisco WebDialer 5-58
Cisco WSM Connector 5-58
Database Change Notification Server 5-63
Database Change Notification Subscription 5-64
Database Local DSN 5-64
DB User Host Information 5-64
Enterprise Replication 5-65
Enterprise Replication DS/Space Monitors 5-64
IP 5-65
Memory 5-66
Network Interface 5-67
Partition 5-69
Process 5-70
Processor 5-71
System 5-72
TCP 5-73
Thread 5-73
Tomcat JVM 5-61
Tomcat Web Application 5-61

perfmon counters 3-36

performance monitoring
cisco unified CM server objects

IME Client 5-79
IME Client Instance 5-80

IME server objects

IME Configuration Manager 5-74, 5-77

Number of Replicates 5-68

object and counters

Cisco Analog Access 5-5
Cisco Annunciator Device 5-5
Cisco CallManager 5-5
Cisco CallManager External Call Control 5-13
Cisco CallManager SAF 5-14
Cisco CallManager System Performance 5-15
Cisco CTIManager 5-17
Cisco Dual-Mode Mobility 5-17
Cisco Extension Mobility 5-19
Cisco Feature Control Policy 5-20
Cisco Gatekeeper 5-20
Cisco H.323  5-20
Cisco Hunt Lists  5-21
Cisco HW Conference Bridge Device  5-22
Cisco IP Manager Assistant  5-22
Cisco Lines  5-23
Cisco Locations  5-23
Cisco Media Streaming Application  5-24
Cisco Messaging Interface  5-27
Cisco MGCP BRI Device  5-28
Cisco MGCP FXO Device  5-29
Cisco MGCP FXS Device  5-29
Cisco MGCP Gateways  5-30
Cisco MGCP PRI Device  5-30
Cisco MGCP T1CAS Device  5-31
Cisco Mobility Manager  5-32
Cisco MOH Device  5-33
Cisco MTP Device  5-34
Cisco Phones  5-34
Cisco Presence Feature  5-34
Cisco QSIG Feature  5-35
Cisco Signaling Performance  5-35
Cisco SIP  5-35, 5-36
Cisco SIP Normalization  5-36
Cisco SIP Stack  5-43
Cisco SIP Station  5-51
Cisco SW Conf Bridge Device  5-52
Cisco TFTP Server  5-53
Cisco Tomcat Connector  5-59
Cisco Transcode Device  5-56
Cisco Video Conference Bridge  5-57
Cisco WebDialer  5-58
Cisco WSM Connector  5-58
Database Change Notification Server  5-63
Database Change Notification Subscription  5-64
Database Local DSN  5-64
DB User Host Information  5-64
Enterprise Replication  5-65
Enterprise Replication DBSpace Monitors  5-64
IP  5-65
Memory  5-66
Network Interface  5-67
Number of Replicates  5-68
Partition  5-69
Process  5-70
Processor  5-71
System  5-72
Thread  5-73
Tomcat JVM  5-61
Tomcat Web Application  5-61
phone registration status  3-37
Process
  perfmon object and counters  5-70
Processor
  perfmon object and counters  5-71

R

Replication Fails Between the Publisher and the Subscriber  3-39
report
  trunk utilization  2-21
Resetting Database Replication When Reverting to an Older Product Release  3-43
RIS data collector perfmonlog  3-23
role-based access control  3-32
RTMT
  callmanager perfmon objects and counters  5-5
  system perfmon objects and counters  5-59
RTMT reports  3-33

S

security patching and updating  3-32
SEQNUM field  6-6
serviceability reports  3-34
SEVERITY field  6-11
SNMP
  basics  4-3
Index

community strings 4-4
informs
  overview 4-5
SNMPv1 4-2
trace configuration 4-5
traps
  overview 4-5
troubleshooting tips for developers 4-5
users 4-4
snmp
  snmp/r MIBs 4-8
  troubleshooting 4-6
SNMP MIBs 3-27
standard syslog server implementations 6-4
summary 3-12
summary of CLI commands and GUI selections 3-43
syslog messages 3-25
System
  perfmon object and counters 5-72
system health
  critical processes to monitor 3-2
  miscellaneous information 3-33, 3-34, 3-36, 3-37
platform monitoring 3-27, 3-28
platform security 3-31, 3-32
recovery, migration, and backup/restore 3-26, 3-31
related documentation 3-44
RTMT monitoring 3-12
RTMTmonitoring 3-12, 3-13, 3-15, 3-16, 3-17, 3-18, 3-20, 3-21, 3-23, 3-24, 3-25, 3-26
software configuration management 3-32, 3-33
software configuration management detecting version and packages 3-33
supported interfaces 3-1

Thread
  perfmon object and counters 5-73
TIMESTAMP field 6-8
Tomcat JVM
  perfmon object and counters 5-61
Tomcat Web Application
  perfmon object and counters 5-61
trace
  collection 1-5
  recommendations for SNMP 4-5
  trace and log central 1-5
trace collection 1-5
trace tools 1-4
traps
  overview 4-5
troubleshooting
  database tables out of sync do not trigger alert 3-42
  for SNMP developers 4-5
troubleshooting trace 1-5
trunk utilization report 2-21

U

UPS integration 3-37
users (SNMP) 4-4
utilization
  trunk 2-21

V

virtual memory 3-16

T

TAGS field 6-14
TCP 5-73
  perfmon object and counters 5-73