Configuring SNMP Monitoring

This chapter describes how to configure Simple Network Management Protocol (SNMP) traps, recipients, community strings, group associations, user security model groups, and user access permissions.

Note Throughout this chapter, the term Cisco WAAS device is used to refer collectively to the WAAS Central Manager and WAEs in your network. The term WAE refers to WAE appliances, and WAE Network Modules (the Cisco Network Modules-WAE family of devices).

This chapter contains the following sections:

Understanding SNMP

SNMP is an interoperable standards-based protocol that allows for external monitoring of Cisco WAAS devices through an SNMP agent.

An SNMP-managed network consists of the following primary components:

- **Managed device**: A network node that contains an SNMP agent and resides on a managed network. Managed devices include routers, access servers, switches, bridges, hubs, computer hosts, and printers. Each WAAS device running the WAAS software has an SNMP agent.
- **SNMP agent**: A software module that resides on a managed device. An agent has local knowledge of management information and translates that information into a form that is compatible with SNMP. The SNMP agent gathers data from the MIB, which is the repository for information about device parameters and network data. The agent can also send traps, or notification of certain events, to the management system.
- **Management station**: Also known as the SNMP host, the management station uses SNMP to send the agent an SNMP Get request to obtain information from the WAAS device. The managed devices then collect and store management information and use SNMP to make this information available to the management station.

Before you can access this SNMP information, you must have deployed an SNMP management application on a management station. This SNMP management station is referred to as the SNMP host because it uses SNMP to send the device agent an SNMP Get request to obtain information from the WAAS device.

This section contains the following topics:

SNMP Communication Process

The SNMP management station and the SNMP agent that resides on a Cisco WAAS device use SNMP to communicate as follows:

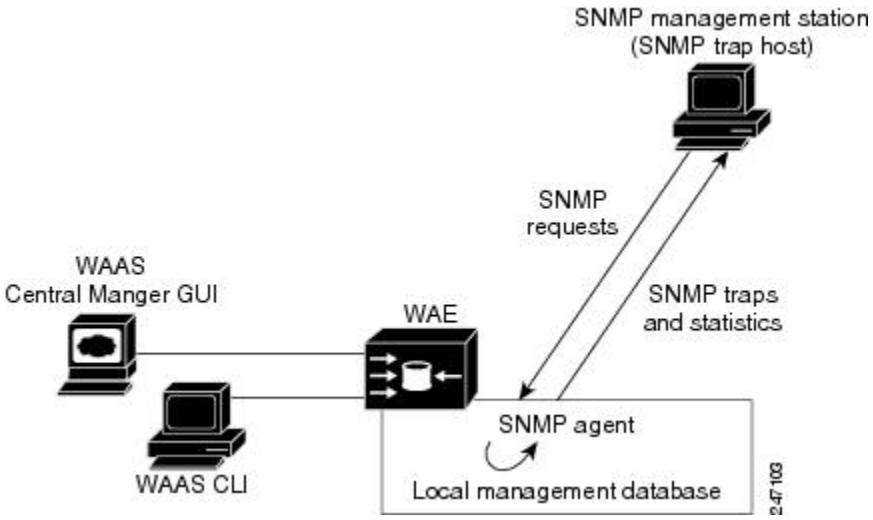
1. The SNMP management station (the SNMP host) uses SNMP to request information from the Cisco WAAS device.
2. After receiving these SNMP requests, the SNMP agent on the Cisco WAAS device accesses a table that contains information about the individual device. This table, or database, is called a MIB.

Note The SNMP agent on the Cisco WAAS device only initiates communication with the SNMP host under unusual conditions; it will initiate communication when it has a trap it needs to send to the host. For more information on this topic, see Enabling SNMP Traps, on page 37.

1. After locating the specified information in the MIB, the agent uses SNMP to send the information to the SNMP management station.

The following figure illustrates these SNMP operations for an individual Cisco WAAS device.

Figure 1: SNMP Components in a Cisco WAAS Network



Supported SNMP Versions

The Cisco WAAS software supports the following versions of SNMP:

- **Version 1 (SNMPv1)**: This is the initial implementation of SNMP. See RFC 1157 for a full description of its functionality.
- **Version 2 (SNMPv2c)**: This is the second release of SNMP, described in RFC 1902. It provides additions to data types, counter size, and protocol operations.
- **Version 3 (SNMPv3)**: This is the most recent version of SNMP, defined in RFC 2271 through RFC 2275.

Each Cisco device running Cisco WAAS software contains the software necessary to communicate information about device configuration and activity using SNMP.

SNMP Security Models and Security Levels

SNMPv1 and SNMPv2c do not have any security (that is, authentication or privacy) features to keep SNMP packet traffic confidential. As a result, packets on the wire can be detected and SNMP community strings compromised.

To solve the security shortcomings of SNMPv1 and SNMPv2c, SNMPv3 provides secure access to WAAS devices by authenticating and encrypting packets over the network. The SNMP agent in the WAAS software supports SNMPv3 as well as SNMPv1 and SNMPv2c.

The following security features are provided in SNMPv3:

- **Message integrity**: Ensures that nothing has interfered with a packet during transmission.
- **Authentication**: Determines that the message is from a valid source.
- **Encryption**: Scrambles the contents of a packet to prevent it from being seen by an unauthorized source.

SNMPv3 provides security models as well as security levels. A security model is an authentication process that is set up for a user and the group in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security process is used when an SNMP packet is handled. Three security models are available: SNMPv1, SNMPv2c, and SNMPv3.

The following table describes the combinations of security models and security levels.

Table 1: SNMP Security Models and Security Levels

Model	Level	Authentication	Encryption	Process
v1	noAuthNoPriv	Community string	No	Uses a community string match for user authentication.
v2c	noAuthNoPriv	Community string	No	Uses a community string match for user authentication.
v3	noAuthNoPriv	Username	No	Uses a username match for user authentication.
v3	AuthNoPriv	Message Digest 5 (MD5) or Secure Hash Algorithm (SHA)	No	Provides authentication based on the Hash-Based Message Authentication Code (HMAC)-MD5 or HMAC-SHA algorithms.
v3	AuthPriv	MD5 or SHA	Yes	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides Data Encryption Standard (DES) 56-bit encryption (packet authentication) based on the cipher block chaining (CBC)-DES (DES-56) standard.

The SNMPv3 agent can be used in the following modes:

- **noAuthNoPriv** mode (that is, no security mechanisms turned on for packets)
- **AuthNoPriv** mode (for packets that do not have to be encrypted using the privacy algorithm [DES 56])
- **AuthPriv** mode (for packets that must be encrypted; privacy requires that authentication be performed on the packet)

Using SNMPv3, users can securely collect management information from their SNMP agents without worrying that the data has been tampered with. Also, confidential information, such as SNMP set packets that change a Content Engine's configuration, can be encrypted to prevent their contents from being exposed on the wire. Also, the group-based administrative model allows different users to access the same SNMP agent with varying access privileges.

SNMPv3 with AES Encryption

This section containst the following topics:

About SNMPv3 with AES Encryption

This section describes DES and AES encryption for different Cisco WAAS versions.

- **SNMP in Cisco WAAS Version 6.4.3d and earlier**: Supports only Data Encryption Standard (DES) encryption.

- **SNMP in Cisco WAAS Version 6.4.3e and later**: Supports Advanced Encryption Standard (AES) encryption as well as DES encryption, which provides strong encryption capability for SNMPv3 messages. AES encryption uses the Cipher Feedback (CFB) mode with encryption key sizes of 128, 192, or 256 bits.

  The SNMPv3 User-based Security Model (USM) provides three modes of operation:

  - **noAuthNoPriv**: This mode is similar to the SNMPv1 and SNMPv2c model, in that a username is treated in a manner equivalent to the community string. This mode does not utilize strong authentication and does not encrypt SNMP traffic.
  - **authNoPriv**: This mode provides strong authentication via SHA or MD5, but does not encrypt SNMP traffic.
  - **authPriv**: This mode provides strong authentication via SHA or MD5, and encrypts SNMP messages using DES encryption algorithm.

  For Cisco WAAS Version 6.4.3e and later, Cisco provides support for AES as an additional option for message encryption under the SNMPv3 authPriv mode.

  Command modification for SNMPv3 with AES encryption:

  For Cisco WAAS Version 6.4.3e and later, the global configuration command **snmp-server user** *name group* contains the parameter **protocol AES {128 | 192 |256} | DES**. This parameter specifies the encryption method and key length. For more information, see the *Cisco Wide Area Application Services Command Reference*.

Operating Guidelines for SNMPv3 with AES Encryption

Consider the following operating guidelines for SNMPv3 with AES encryption:

- If one of the devices in a device group is running an earlier version than Cisco WAAS Version 6.4.3e, you must upgrade the device to Cisco WAAS Version 6.4.3e for the device group to create an AES encryption user for the group.
- If **Priv Password** is **Empty**, do not select Protocol Algorithm and AES Encryption.
- If **Priv Password** is entered, select **Priv Protocol** either (**DES** or **AES**).
- If Protocol is selected by AES, select **AES Encryption**.
- If **no-auth** is selected, do not select Protocol.
- If **Priv Password** is **Empty**, do not select AES Encryption.
- If the Cisco WAAS Central Manager is running Cisco WAAS Version 6.4.3e or later, and the device is part of a device group containing devices running 6.4.3e, then you can create the AES encryption user.

Upgrade and Downgrade Guidelines for SNMPv3 with AES Encryption

Consider the following upgrade and downgrade guidelines for SNMPv3 with AES encryption:

- Cisco WAAS 6.4.3d release and earlier Cisco WAAS versions support DES encryption only. The privacy protocol cannot be configured, and is not displayed in the **running-configuration** and **show** commands.

- If you downgrade from Cisco WAAS Version 6.4.3e to an earlier Cisco WAAS version, the downgrade will not proceed if there are SNMPv3 users configured in **authpriv mode** with **Priv Protocol AES**. A Warning message will be displayed in the CLI if AES encryption users are present in the running configuration. A pop-up message will also be displayed, if you attempt to downgrade from the WAAS Central Manager and if AES encryption users are present in the running configuration. You must remove the AES encryption users from configuration and downgrade again to complete the image installation.
- After upgrading to Cisco WAAS Version 6.4.3e or later, the existing SNMP users in **authpriv** mode, if present, will be added with **Priv Protocol DES**.

Cisco-Supported MIBs

This section contains the following topics:

About Cisco-Supported MIBs and CISCO-SMI

A Management Information Base (MIB) is a collection of managed objects, arranged in a hierarchical tree of MIB modules, groups, and objects:

- **MIB module**: Contains related MIB groups.

For example, CISCO-WAN-OPTIMIZATION-MIB contains many types of optimization groups, including cwoAoStats and cwoTfoStats.

- **MIB group**: Contains the prefix for a set of related MIB objects, such as cwoAoStats (AO statistics) and cwoTfoStats (TFO statistics).
- **MIB object**: Provides information about a specific aspect of the specified MIB group.

For example:

- The **cwoAoStatsIsConfigured** MIB object indicates if the AO is configured or not.
- The **cwoTfoStatsLoadStatus** displays the current TFO load status (such as "operating normally" or "overloaded").

The Structure of Management Information (SMI) defines the framework within which you can define or construct a MIB. The CISCO-SMI MIB group describes the structure of Cisco MIBs.

Types of MIB Output for SNMP Monitoring

This section contains the following topics:

MIB Output for Statistical Data

MIB output can provide information about a device, interface, or process at a specified moment in time. The following figure shows an example of MIB output of statistical data for the MIB object **cwoDreCacheStats**. This object displays DRE cache information, such as the current operational status, the portion of the disk space allocated for DRE cache, the age of the oldest data unit the data block, and the amount of data units replaced in the last hour.

Sample MIB Output for DRE Cache Information with cwoDreCacheStats

```
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsStatus.0 = STRING: Usable
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsTotal.0 = Counter64: 77822 MB
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsUsed.0 = Gauge32: 96 percent
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsDataUnitUsage.0 = Counter64: 0 MB
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsReplacedOneHrDataUnit.0 = Counter64: 0 MB
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsDataUnitAge.0 = STRING: 0s
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsSigblockUsage.0 = Counter64: 1695 MB
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsReplacedOneHrSigblock.0 = Counter64: 0 MB
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsSigblockAge.0 = STRING: 14d17h
```

MIB Output for Trend Data

The greatest value provided by MIBs may be in enabling SNMP monitoring to use the external MIB tool to gather statistics — and then provide trend data from these statistics, in either text or graphical format. This enables you to more easily identify anomalies in your WAAS network, and therefore to more effectively plan or modify your network.

For example, in the output shown in the above Figure 17-2 , the MIB cwoDreCacheStatsUsed provides information on the percentage of DRE disk space currently being used:

```
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsUsed.0 = Gauge32: 96 percent
```

If you want to monitor the trend of how DRE disk space is being used over a particular period of time, you could run the cwoDreCacheStatsUsed MIB for a specified time range. As shown below in Figure 17-3 , you could view data for a specified time range that displays the usage trend for the DRE cache disk space.

Sample MIB Output for Percentage of DRE Disk Space Being Used

```
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsUsed.0 = Gauge32: 85 percent
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsUsed.0 = Gauge32: 91 percent
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsUsed.0 = Gauge32: 96 percent
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsUsed.0 = Gauge32: 98 percent
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsUsed.0 = Gauge32: 93 percent
```

For more information on MIB usage with SNMP monitoring, and for more examples of MIB output, see Using MIBs to Monitor Cisco WAAS, on page 31.

Types of Cisco-Supported MIBs

This section describes the types of Cisco-supported MIBs, in alphabetical order by topic:

Akamai Connect (CISCO-WAN-OPTIMIZATION-MIB)

The following table shows the Akamai Connect MIB objects associated with the cwoAoHttpxStatsAKC MIB object, for a specified caching mode: Standard, Basic, Bypass, or Advanced. For each Akamai Connect caching mode, there are MIB objects that provide the following types of information:

- **Cache transactions**: The Akamai Connect cache statistics for the total number of cache-hit transactions that were served from cache in the specified caching mode.
- **Cache transactions percent**: The percentage of total number of cache-hit HTTP transactions in the specified caching mode.
- **Cache response time saved**: The total response time saved for cache-hit HTTP transactions in the specified Akamai Connect cache mode, in milliseconds.

- **Average cache response time saved**: The average response time saved per cache-hit HTTP transaction in the specified Akamai Connect cache mode, in milliseconds.
- **Response in bytes**: The total number of response bytes saved for cache-hit HTTP transactions in the specified Akamai Connect cache mode.
- **Response bytes percent**: The percentage of total number of response bytes saved for cache-hit HTTP transactions in the specified Akamai Connect cache mode.
- **Response time saved percent**: The percentage of total response time saved for cache-hit HTTP transactions in the specified Akamai Connect cache mode.

Table 2: Akamai Connect MIB Objects

MIB Object	Associated MIB Objects
cwoAoHttpxStatsAKCStdEntry	Provides information about the Akamai Connect cache in Standard mode (default), in which Akamai Connect caches objects marked as cacheable, as well as objects with no explicit cache marker and with a last-modified date. • cwoAoHttpxStatsAKCStdCacheTrans • cwoAoHttpxStatsAKCStdCacheTransPercent • cwoAoHttpxStatsAKCStdCacheRespTimeSaved • cwoAoHttpxStatsAKCStdAvgCacheRespTimeSaved • cwoAoHttpxStatsAKCStdRespBytes • cwoAoHttpxStatsAKCStdRespBytesPercent • cwoAoHttpxStatsAKCStdRespTimeSavedPercent
cwoAoHttpxStatsAKCBasicEntry	Provides information about the Akamai Connect cache in Basic mode, in which Akamai Connect caches only objects explicitly marked as cacheable. • cwoAoHttpxStatsAKCBasicCacheTrans • cwoAoHttpxStatsAKCBasicCacheTransPercent • cwoAoHttpxStatsAKCBasicRespBytes • cwoAoHttpxStatsAKCBasicRespBytesPercent • cwoAoHttpxStatsAKCBasicCacheRespTimeSaved • cwoAoHttpxStatsAKCBasicAvgCacheRespTimeSaved • cwoAoHttpxStatsAKCBasicRespTimeSavedPercent

MIB Object	Associated MIB Objects
cwoAoHttpxStatsAKCBypassEntry	Provides information about the Akamai Connect cache in Bypass mode, in which Akamai Connect caching is turned off for a configured site or sites. • cwoAoHttpxStatsAKCBypassCacheTrans • cwoAoHttpxStatsAKCBypassCacheTransPercent • cwoAoHttpxStatsAKCBypassCacheRespTimeSaved • cwoAoHttpxStatsAKCBypassAvgCacheRespTimeSaved • cwoAoHttpxStatsAKCBypassCacheRespTimeSavedPercent • cwoAoHttpxStatsAKCBypassRespBytes • cwoAoHttpxStatsAKCBypassRespBytesPercent
cwoAoHttpxStatsAKCAdvEntry	Provides information about the Akamai Connect cache in Advanced mode, in which Akamai Connect caches media types more aggressively, and caches all object types for longer times, when there is no explicit expiration time. • cwoAoHttpxStatsAKCAdvCacheTrans • cwoAoHttpxStatsAKCAdvRespBytes • cwoAoHttpxStatsAKCAdvCacheTransPercent • cwoAoHttpxStatsAKCAdvRespBytesPercent • cwoAoHttpxStatsAKCAdvCacheRespTimeSaved • cwoAoHttpxStatsAKCAdvAvgCacheRespTimeSaved • cwoAoHttpxStatsAKCAdvRespTimeSavedPercent
cwoAoHttpxStatsAKCTotalEntry	Provides summary information about the Akamai Connect cache, from all caching modes. • cwoAoHttpxStatsAKCTotalCacheTrans • cwoAoHttpxStatsAKCTotalRespBytes • cwoAoHttpxStatsAKCTotalCacheTransPercent • cwoAoHttpxStatsAKCTotalRespBytesPercent • cwoAoHttpxStatsAKCTotalCacheRespTimeSaved • cwoAoHttpxStatsAKCTotalAvgCacheRespTimeSaved • cwoAoHttpxStatsAKCTotalRespTimeSavedPercent

Alarms (CISCO-CONTENT-ENGINE-MIB)

The following table describes CISCO-CONTENT-ENGINE-MIB objects that are used to verify if there are critical, major, or minor alarms raised on the system.

Table 3: Alarms MIB Objects

MIB Object	Description
cceAlarmMinorCount	The number of alarms currently raised with a severity level of Minor.
cceAlarmMajorCount	The number of alarms currently raised with a severity level of Major.
cceAlarmCriticalCount	The number of alarms currently raised with a severity level of Critical.

AOs (CISCO-WAN-OPTIMIZATION-MIB)

The CISCO-WAN-OPTIMIZATION-MIB group displays information about the status and statistics associated with application optimizers.

The Application Optimizers (AOs), also known as Application Acclerators, statistics MIB group displays status information such as configuration or license information for AOs including HTTP, SSL, MAPI, SMB, and ICA.

This section contains the following tables for the **cwoAoStats** MIB objects:

- AO Name, Configuration, and License MIB Objects
- AO Operational Status, Startup Time, and Reset Time MIB Objects
- AO Summary Connection Information MIB Objects
- AO Current Connection Information MIB Objects
- AO Load Status and Bandwidth Information MIB Objects

Table 4: AO Name, Configuration, and License MIB Objects

MIB Object	Description
cwoAoStatsName	The name of the AO, such as HTTP, SSL, MAPI, SMB, and ICA.
cwoAoStatsIsConfigured	Indicates if the AO is configured or not. **Note** If the AO is not configured, then the cwoAoStatsOperationalState for this AO is Shutdown.
cwoAoStatsIsLicensed	Indicates if the license for the AO is valid or not. **Note** If the license for the AO is not valid, then the cwoAoStatsOperationalState for this AO is Shutdown.

Table 5: AO Operational Status, Startup Time, and Reset Time MIB Objects

cwoAoStatsOperationalState	The operational state of the AO: • shutdown (1) • initializing (2) • normalRunning (3) • normalDisabled (4) • licenseExpired (5) • cleaningup (6) • error (7) **Note** If the AO is not configured or if the license for this AO is not valid, the operational state is Shutdown.
cwoAoStatsStartUpTime	The date and time when the AO was started.
cwoAoStatsLastResetTime	The date and time of the last time the statistics of the AO were reset. When the specified AO's statistics are reset, then all statistics counters are also reset. **Note** When the specified AO is in the Shutdown state, the value of cwoAoStatsStartUpTime and cwoAoStatsLastResetTime is Null.

cwoAoStats MIB Objects for AO Summary Connection Information

Table 6: AO Summary Connection Information MIB Objects

MIB Object	Description
cwoAoStatsTotalHandledConn	Total number of connections handled by the AO since it was started or since its statistics were last reset.
cwoAoStatsTotalOptConn	Total number of connections optimized by the AO since it was started or since its statistics were last reset.
cwoAoStatsTotalHandedOffConn	Total number of connections handed off to generic optimization by the AO since it was started or since its statistics were last reset.
cwoAoStatsTotalDroppedConn	Total number of connections dropped by the AO since it was started or since its statistics were last reset.

Table 7: AO Current Connection Information MIB Objects

MIB Object	Description
cwoAoStatsActiveOptConn	The number of active connections that are getting optimized by the AO.
cwoAoStatsMaxActiveOptConn	The maximum number of active TCP connections the AO can optimize.

MIB Object	Description
cwoAoStatsPendingConn	The number of connections currently pending in the queue of connections to be optimized by the AO.

Table 8: AO Load Status and Bandwidth Information MIB Objects

MIB Object	Description
cwoAoStatsLoadStatus	The load status of the AO.
cwoAoStatsBwOpt	The percentage bandwidth optimization achieved due to optimization done by the AO.

Applications (CISCO-WAN-OPTIMIZATION-MIB)

The cwoAppStats MIB object displays information about application optimization and traffic.

Table 9: Applications Information MIB Objects

MIB Object	Description
cwoAppStatsAppName	The name of a particular application that is configured for optimization.
cwoAppStatsOriginalBytes	The total original traffic (uncompressed) in bytes of a particular application that has entered into the system.
cwoAppStatsOptimizedBytes	The total optimized traffic, in bytes, of a particular application.
cwoAppStatsPTBytes	The total pass-through traffic, in bytes, of a particular application.

AppNav (CISCO-APPNAV-MIB)

The CISCO-APPNAV-MIB group displays information about AppNavwhen the WAAS device is in AppNav Controller mode.

This section contains the following topics:

AppNav Controller MIB Objects

An AppNav Controller is a device that intercepts network traffic and, based on a flow policy, distributes that traffic to one more WAAS nodes for optimization. The following table displays AppNav Controller MIB objects.

Table 10: AppNav Controller Group MIB Objects

MIB Object	Descripton
cAppNavACIndex	An index of the cAppNavACTable. The unique integer value generated for each entry must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

MIB Object	Descripton
cAppNavACIpAddrType	The address type of the cAppNavACIpAddr object. The cAppNavACEntries are only valid for address types of IPv4 and IPv6.
cAppNavACIpaddr	The IP address of the AppNav Controller.
cAppNavACServContextName	The name of the service context to which the specified AppNav Contoller belongs.
cAppNavACACGName	The name of the AppNav Controller Group to which the specified AppNav Controller belongs.
cAppNavACCurrentCMState	The current cluster membership state of the specified AppNav Controller. • **Green (1)**: Operational with no error conditions • **Yellow (2)**: Degraded (overloaded, joining cluster, or has other noncritical operational issues) • **Red (3)**:Critical (one or more processes is in a critical state) • **Gray (4)**: Disabled • **Black (5)**: Unknown status

AppNav Controller Group MIB Objects

An AppNav Controller Group is a group of AppNav Controllers that together provide the necessary intelligence for handling asymmetric flows and high availability. Table 17-11 displays AppNav Controller Group MIB objects.

Table 11: AppNav Controller Group MIB Objects

MIB Object	Description
cAppNavACGIndex	An index of the AppNavACGTable. The unique integer value generated for each entry must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.
cAppNavACGName	The name of the AppNav Controller Group.
cAppNavACGServContextName	The service context to which the specified AppNav Controller Group belongs.

AppNav Service Node MIB Objects

A WAAS node is also known as a service node. The following table displays AppNav service node MIB objects.

Table 12: AppNav Service Node MIB Objects

MIB Object	Description
cAppNavSNIndex	An index of the **cAppNavSNTable**. The unique integer value generated for each entry must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.
cAppNavSNIpAddrType	The address type of **cacSNIpAddr**. The **cacSNEntries** are valid for address types IPv4 and IPv6 only.
cAppNavSNIpAddr	The IP address of the specified service node.
cAppNavSNServContextName	The name of the service context to which the specified service node belongs.
cAppNavSNSNGName	The name of the service node group to which the specified service node belongs.
cAppNavSNCurrentCMState	The current cluster membership state of the specified service node. • **Green (1)**: Operational with no error conditions • **Yellow (2)**: Degraded (overloaded, joining cluster, or has other noncritical operational issues) • **Red (3)**: Critical (one or more processes is in a critical state) • **Gray (4)**: Disabled • **Black (5)**: Unknown status

AppNav Service Node Group MIB Objects

A WAAS node is also known as a service node. The following table displays AppNav service node MIB objects.

Table 13: AppNav Service Node MIB Objects

MIB Object	Description
cAppNavSNIndex	An index of the **cAppNavSNTable**. The unique integer value generated for each entry must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.
cAppNavSNIpAddrType	The address type of **cacSNIpAddr**. The cacSNEntries are valid for address types IPv4 and IPv6 only.
cAppNavSNIpAddr	The IP address of the specified service node.
cAppNavSNServContextName	The name of the service context to which the specified service node belongs.

MIB Object	Description
cAppNavSNSNGName	The name of the service node group to which the specified service node belongs.
cAppNavSNCurrentCMState	The current cluster membership state of the specified service node. • **Green (1)**: Operational with no error conditions • **Yellow (2)**: Degraded (overloaded, joining cluster, or has other noncritical operational issues) • **Red (3)**: Critical (one or more processes is in a critical state) • **Gray (4)**: Disabled • **Black (5)**: Unknown status

AppNav Service Node Group MIB Objects

A Service Node Group is also known as a WAAS Node Group. The following table displays AppNav Service Node Group MIB objects.

Table 14: AppNav Service Node Group Information MIB Objects

MIB Object	Description
cAppNavSNGIndex	An index of the **cAppNavSNGTable**. The unique integer value generated for each entry must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.
cAppNavSNGName	The name of the Service Node Group.
cAppNavSNGServContextName	The service context to which the specified Service Node Group belongs.

AppNav Service Context MIB Objects

A service context is used to tie the AppNav Controller group, service node group, and AppNav policy map together. The following table displays the AppNav Service Context MIB objects.

Table 15: AppNav Service Context Information MIB Objects

MIB Object	Description
cAppNavServContextIndex	An index of the **cAppNavServiceContextTable**. The unique integer value generated for each entry must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.
cAppNavServContextName	The name of the service context.

MIB Object	Description
cAppNavServContextCurrOpState	The current operational state of the service context.
cAppNavServContextLastOpState	The last operational state of the service context.
cAppNavServContextIRState	The **Interception Readiness** (IR) state of the service context.
cAppNavServContextJoinState	The **Join** state of the service context.

Class Maps (CISCO-WAN-OPTIMIZATION-MIB)

AppNav class maps classify traffic according to one or more match conditions, such as peer device ID, or a mix of one peer device ID and the source IP, or destination IP, or destination port. The following table shows class map information MIB objects.

Table 16: Class Map Information MIB Objects

MIB Object	Description
cwoCmapStatsType	The class map type, such as HTTP, MAPI, NFS, or a custom class map.
cwoCmapStatsName	The name of the class map.
cwoCmapStatsDescr	The descriptive information of the class map configured on the WAN optimization system. If the description is not configured for a given class map, then this string will be a NULL string.
cwoCmapStatsTotalConns	The total number of connections processed by the class map.
cwoCmapStatsTotalBytes	The total number of bytes processed by the class map.
cwoCmapStatsTotalPTConns	The total connections made as pass-through, due to some reason by the class map.
cwoCmapStatsTotalPTBytes	The total number of bytes made pass-through by the class map.

Configuration (CISCO-CONFIG-MAN-MIB)

The CISCO-CONFIG-MAN-MIB group represents a model of configuration data that exists in various locations:

- **Running**: In use by the running system
- **Terminal**: Saved to whatever hardware is attached as the terminal
- **Local**: Saved locally in NVRAM or in flash memory
- **Remote**: Saved to a server on the network

Note The CISCO-CONFIG-MAN-MIB group includes only operations that are specifically related to configuration, although some of the system functions can be used for general file storage and transfer.

CPU and Memory (CISCO-PROCESS-MIB)

The CISCO-PROCESS-MIB group displays memory and CPU usage on the device and also describes active system processes.

CPU utilization presents a status of how busy the system is. The numbers are a ratio of the current idle time over the longest idle time. (This information should be used as an estimate only.)

Table 17: CPU and Memory Information MIB Objects

MIB Object	Description
cpmCPUTotal1minRev	The overall CPU percentage showing how busy the system was in the last 1 minute.
cpmCPUTotal5minRev	The overall CPU percentage showing how busy the system was in the last 5 minutes.

Devices (CISCO-CDP-MIB and CISCO-ENTITY-ASSET-MIB)

This section describes two MIB groups:

CISCO-CDP-MIB Group

The CISC-CDP-MIB group displays the ifIndex value of the local interface.

For example:

- 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 is greater than any ifIndex value supported by the repeater.
- In this example, the specific port is indicated by the corresponding values of cdpInterfaceGroup and cdpInterfacePort, where these values correspond to the group number and the port number values of RFC 1516.

CISCO-ENTITY-ASSET-MIB Group

The CISCO-ENTITY-ASSET-MIB group provides information about items in the entPhysicalTable MIB object, including part number, serial number, hardware version, firmware ID and software ID. A full description of these is provided in RFC 2037.

Note the following about information listed in entPhysicalTable:

- Displayed information includes the orderable part number, serial number, hardware revision, manufacturing assembly number and revision, firmware ID and revision (if any), and software ID and revision (if any) of relevant entities listed in entPhysicalTable. Entities that have none of this data available are not listed in this MIB.
- The entPhysicalTable is sparsely populated. Therefore, some variables may not exist for a particular entity at a particular time.

For example, a row that represents a powered-off module may have no values for software ID (ceAssetSoftwareID) and revision (ceAssetSoftwareRevision). Similarly, a power supply would probably never have firmware or software information listed in the table.

- The data may have other items encoded in it.

For example, a manufacturing date in the serial number, consider all data items to be a single unit. Do not decompose the items or parse them. Use only string equal and unequal operations on them.

DRE Cache (CISCO-WAN-OPTIMIZATION-MIB)

The following table displays optimization DRE cache statistics MIB objects, which provide information such as the portion of disk space allocated for DRE cache or the percentage of DRE disk space currently being used.

Table 18: DRE Cache Statistics MIB Objects

MIB Object	Description
cwoDreCacheStatsStatus	The status of the portion of the disk allocated for DRE cache: **Initializing**, **Usable**, or **Failed**.
cwoDreCacheStatsAge	The age of the oldest data present in the DRE cache. When new data is written to the DRE cache portion of the disk, it replaces the oldest data in the DRE cache.
cwoDreCacheStatsTotal	The portion of disk space allocated for DRE cache, in MB. For example, if the total cache disk space is 708 MB, and the portion allocated for DRE cache is 10%, then the value of cwoDreCacheStatsTotal, as shown below in the sample output, is 70800 MB.
cwoDreCacheStatsUsed	The percentage of DRE disk space currently being used. For example, if the disk space allocated for DRE is 70800 MB, and the value of **cwoDreCacheStatsUsed** is 85%, as shown below in the sample output, this indicates that 60,180 MB of the DRE cache disk space is being used, and 10,620 MB of the DRE cache disk space is free.
cwoDreCacheStatsDataUnitUsage	The DRE cache disk space currently being used, by data unit.
cwoDreCacheStatsReplacedOneHrDataUnit	The amount of data units replaced in the DRE cache in the last hour. Data is replaced on a First In/First Out (FIFO) order, and is stored in the DRE cache data block.
cwoDreCacheStatsDataUnitAge	The age of the oldest data unit in the data block. When new data is written to the data block when the data block is full, the oldest data unit is removed.
cwoDreCacheStatsSigblockUsage	The DRE disk space currently used by the signature block.
cwoDreCacheStatsReplacedOneHrSigblock	The amount of cache replaced within the last hour by the signature block.

MIB Object	Description
cwoDreCacheStatsSigblockAge	The time that the DRE Sigblock has been in the cache in days (d),hours (h), minutes (m), and seconds (s). For example, “1d1h” means 1 day, 1 hour.

DRE Performance (CISCO-WAN-OPTIMIZATION-MIB)

The following table displays DRE performance MIB objects, which provide information such as DRE compression ratio during decoding or the decoding average message size.

Table 19: DRE Performance Statistics MIB Objects

MIB Object	Description
cwoDrePerfStatsEncodeCompressionRatio	The DRE compression ratio during encoding.
cwoDrePerfStatsEncodeCompressionLatency	The Encoding average latency introduced to compress a message.
cwoDrePerfStatsEncodeAvgMsgSize	The Encoding average message size.
cwoDrePerfStatsDecodeCompressionRatio	The DRE compression ratio during decoding.
cwoDrePerfStatsDecodeCompressionLatency	The Decoding average latency introduced to compress a message.
cwoDrePerfStatsDecodeAvgMsgSize	The Decoding average message size.

HTTP (CISCO-WAN-OPTIMIZATION-MIB)

The following table shows the HTTP AO information MIB objects, which provide information such as the percentage estimated time saved due to optimizations done by HTTP AO since it was started or the total number of SharePoint Optimized HTTP sessions.

Table 20: HTTP AO Information MIB Objects

MIB Object	Description
cwoAoHttpxStatsTotalSavedTime	The total time saved due to optimizations done by HTTP AO since it was started.
cwoAoHttpxStatsTotalRTT	The total Round Trip Time (RTT) for all the connections going through HTTP AO since it was started.
cwoAoHttpxStatsTotalMDCMTime	The Meta Data Cache Misses (MDCM) for HTTP AO since it was started.
cwoAoHttpxStatsEstSavedTime	The percentage estimated time saved due to optimizations done by HTTP AO since it was started.

MIB Object	Description
cwoAoHttpxStatsTotalSPSessions	The total number of SharePoint Optimized HTTP sessions. This counter is incremented for every session on which SharePoint optimization can be performed. An HTTP session is tagged as a SharePoint Session based on the information present in the HTTP request.
cwoAoHttpxStatsTotalSPPFSessions	The total number of SharePoint Pre-fetch optimized HTTP sessions. • This counter is incremented for every session on which SharePoint pre-fetch optimization can be performed. • An HTTP session is tagged as a SharePoint pre-fetch Session based on the information present in the HTTP request. • A pre-fetch operation is one where the edge WAAS device fetches the next set of data (which it anticipates the client will request later) from the server based on the current HTTP Request information.
cwoAoHttpxStatsTotalSPPFObjects	The total number of pre-fetched objects served locally for SharePoint pre-fetch sessions. • The edge WAAS device maintains a local cache where the pre-fetched responses are saved. • This object is incremented whenever the SharePoint client request is served from the pre-fetch cache.
cwoAoHttpxStatsTotalSPRTTSaved	The total Round Trip Time (RTT) saved due to SharePoint pre-fetch optimizations since SharePoint pre-fetch optimization was started.
cwoAoHttpxStatsTotalSPPFMissTime	The total time for SharePoint pre-fetch Cache Misses since SharePoint pre-fetch optimization was started.

Note Discontinuities in the value of these HTTP counters can occur at re-initialization of the HTTP AO. The last discontinuity time is indicated by the value of **cwoAoStatsLastResetTime** for the HTTP AO.

Interfaces (IF-MIB)

The IF-MIB group supports querying for interface-related statistics including 64-bit interface counters. These counters include received and sent octets, unicast, multicast, and broadcast packets on the device interfaces. All the objects from **ifXEntry** are supported except for ifCounterDiscontinuityTime. This MIB is documented in RFC 2233.

Loopback interface interface information are not reported.

A transmission error or discard can point to Layer 1 or Layer 2 problems, such as a bad cable or a speed/duplex mismatch on a connected switch or router.

This section contains the following types of MIB objects for the IF-MIB group:

Interface Description MIB Object

The ifDescr MIB object displays information about the interface, including the name of the manufacturer, the product name, and the version of the hardware or software interface.

Interface Status MIB Objects

This section describes two interface status MIB objects:

- **ifAdminStatus**: Displays the desired (specified) status of the interface:
  - **up (1)**: The interface is up and ready to transmit and receive network traffic.
  - **down (2)**: The interface is down.
  - **testing (3)**: In the Testing state, no operational packets can be passed.

Note At system startup, all interfaces start with ifAdminStatus down. After either management action or configuration information, ifAdminStatus is changed to either up or testing, or remains down.

- **ifOperStatus**: Displays the current operational status of the interface:
  - **up (1)**: The interface is up and ready to transmit and receive network traffic.
  - **down (2)**: The interface is down.
  - **testing(3)**: In the Testing state, no operational packets can be passed.
  - **unknown(4)**: The status of the interface cannot be determined.
  - **dormant(5)**: The interface is waiting for an external action.
  - **notPresent(6)**: The interface has a missing component; usually a missing hardware component.
  - **lowerLayerDown(7)**: The interface is down due to a lower-layer interface.

Note If **ifAdminStatus** is down, then **ifOperStatus** should also be down. If **ifAdminStatus** is up, then **ifOperStatus** should also be up.

Interface Discards MIB Objects

Table 21: Interface Discards MIB Objects

MIB Object	Description
ifInDiscards	Displays the number of inbound packets selected to be discarded, even though no errors have been detected to prevent their being deliverable to a higher-layer protocol. A possible reason for discarding such a packet is to free up buffer space.
ifOutDiscards	The number of outbound packets selected to be discarded, even though no errors had been detected to prevent them from being transmitted. A possible reason for discarding such a packet is to free up buffer space. The **ifInDiscards** MIB object is usually a subset of the **locIfInputQueueDrops** MIB object.

Note Discontinuities in the value of **ifInDiscards** or of **ifOutDiscards** can occur at re-initialization of the management system and at other times, as indicated by the value **ifCounterDiscontinuityTime**.

Interface Errors MIB Objects

Table 22: Interface Errors MIB Objects

MIB Object	Description
ifInErrors	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.
ifOutErrors	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.

Note Discontinuities in the value of **ifInErrors** or of **ifOutErrors** can occur at re-initialization of the management system and at other times, as indicated by the value **ifCounterDiscontinuityTime**.

IP Routing (IP-MIB, IP-FORWARD-MIB, MIB-II)

This section contains the following MIB groups:

IP-MIB Group

The IP-MIB group manages IP and ICMP implementations, excluding their management of IP routes.

IP-FORWARD-MIB Group

Displays Classless Inter-Domain Routing (CIDR) multi-path IP Routes.

MIB-II Group

The MIB-II group is the Internet Standard MIB, and is used with network management protocols in TCP/IP-based internets. The MIB-II is documented in RFC 1213, and is found in the RFC1213-MIB file in the v1 directory on the download site (other MIBs are in the v2 directory).

The following objects from this MIB are not supported:

- ifInUnknownProtos
- ifOutNUcastPkts
- ipRouteAge
- TcpConnEntry group
- egpInMsgs
- egpInErrors
- egpOutMsgs
- egpOutErrors
- EgpNeighEntry group
- egpAs
- atTable
- ipRouteTable

MAPI (CISCO-WAN-OPTIMIZATION-MIB)

The following table displays the Message Application Programming Interface (MAPI) AO MIB objects.

Note For these MIB objects, discontinuities in the value of the counter can occur at re-initialization of the MAPI AO. The last discontinuity time is indicated by the value of cwoAoStatsLastResetTime for the MAPI AO.

Table 23: MAPI AO MIB Objects

MIB Object	Description
cwoAoMapixStatsUnEncrALRT	The Average Local Response Time (ALRT) for unencrypted connections of MAPI AO since it was started.
cwoAoMapixStatsUnEncrARRT	The Average Remote Response Time (ARRT) for unencrypted connections of MAPI AO since it was started.
cwoAoMapixStatsTotalUnEncrLRs	The total requests served locally for unencrypted connections by MAPI AO since it was started.

MIB Object	Description
cwoAoMapixStatsTotalUnEncrRRs	The total Remote Requests(RR) served by remote servers for unencrypted connections of MAPI AO since it was started.
cwoAoMapixStatsUnEncrAvgRedTime	The average time reduced for unencrypted connections due to optimizations done by MAPI AO since it was started.
cwoAoMapixStatsEncrALRT	The Average Local Response Time (ALRT) for encrypted connections of MAPI AO since it was started.
cwoAoMapixStatsEncrARRT	The Average Remote Response Time (ARRT) for encrypted connections of MAPI AO since it was started.
cwoAoMapixStatsTotalEncrLRs	The total requests served locally for encrypted connections by MAPI AO since it was started.
cwoAoMapixStatsTotalEncrRRs	The total Remote Requests (RR) served by remote servers for encrypted connections by MAPI AO since it was started.
cwoAoMapixStatsEncrAvgRedTime	The average time reduced for encrypted connections due to optimizations done by MAPI AO since it was started.

Network Management (EVENT-MIB, HOST-RESOURCES-MIB)

This section contains the following MIB groups:

EVENT-MIB Group

The EVENT-MIB group defines the event triggers and actions for network management purposes. This MIB is described in RFC 2981.

HOST-RESOURCES-MIB Group

This MIB manages host systems. The term “host” implies any computer that communicates with other similar computers connected to the Internet.

The HOST-RESOURCES-MIB provides attributes that are common to all Internet hosts, for example, personal computers and systems that run variants of UNIX. It does not apply to devices whose primary function is communications services (terminal servers, routers, bridges, monitoring equipment).

The following objects from this MIB are not supported:

- HrPrinterEntry
- hrSWOSIndex
- hrSWInstalledGroup

Policy Maps (CISCO-WAN-OPTIMIZATION-MIB)

Policy maps associate policy actions with class maps. The following table shows the policy maps MIB objects, which display information such as the type of policy map or the total number of connections processed by the policy map since it has been active.

Table 24: Policy Maps MIB Objects

MIB Object	Description
cwoPmapStatsType	The type of policy map.
cwoPmapStatsName	The name of the policy map.
cwoPmapStatsDescr	The description of the policy map configured on the WAN optimization system. If a description is not configured for a particular policy map, this string will contain a NULL string.
cwoPmapStatsTotalConns	The total number of connections processed by the policy map since it has been active.
cwoPmapStatsTotalBytes	The total bytes processed by the policy map since it has been active.
cwoPmapStatsTotalPTConns	The total connections made as pass-through connections, due to some reason by the policy map, since it has been active.
cwoPmapStatsTotalPTBytes	The total bytes made as pass-through, due to some reason by the policy map, since it has been active.

SMB (CISCO-WAN-OPTIMIZATION-MIB)

The CISCO-WAN-OPTIMIZATION-MIB group displays information about the status and statistics associated with optimization and application accelerators.

Note For these MIB objects, discontinuities in the value of the counter can occur at re-initialization of the SMB AO. The last discontinuity time is indicated by the value of **cwoAoStatsLastResetTime** for the SMB AO.

This section describes the cwoAoSmbxStats MIB objects, and contains the following topics:

About SMB Statistics MIB Objects

The Server Message Block (SMB) application accelerator (AO) transparently accelerates traffic and supports prepositioning of files. It relies on automatic discovery. You can fine-tune this accelerator for specific traffic needs.

cwoAoSmbxStats MIB Objects for Cache Information

Table 25: SMB AO Cache MIB Objects

MIB Object	
cwoAoSmbxStatsBytesReadCache	The total number of bytes read from the SMB AO cache (Read-ahead and Metadata cache) since it was started.
cwoAoSmbxStatsBytesWriteCache	The total number of bytes written to SMB AO cache (Read-ahead and Metadata) since it was started.

MIB Object	
cwoAoSmbxStatsMDCacheHitCount	The SMB AO Metadata cache hit count since SMB AO was started.
cwoAoSmbxStatsMDCacheHitRate	The SMB AO Metadata cache hit rate since it was started.
cwoAoSmbxStatsMaxRACacheSize	The maximum disk space that can be allocated for Read Ahead data in the SMB AO cache.
cwoAoSmbxStatsMaxMDCacheSize	The maximum disk space that can be allocated for Metadata in the SMB AO cache
cwoAoSmbxStatsRAEvictedAge	The amount of time spent in the SMB AO Read Ahead cache by the resource that was last evicted since last update. **Note** If this amount is too short or too long, we recommend that you modify the size of the cache.
cwoAoSmbxStatsTotalFilesInRACache	The total number of files in the SMB AO Read Ahead cache.

cwoAoSmbxStats MIB Objects for Client and Server Information

Table 26: SMB AO Client and Server MIB Objects

MIB Object	Description
cwoAoSmbxStatsBytesReadServer	The total number of bytes read from file servers by SMB AO since it was started.
cwoAoSmbxStatsBytesWriteServer	The total number of bytes written to file servers by SMB AO since it was started.
cwoAoSmbxStatsBytesReadClient	The total number of bytes read by SMB AO clients since it was started.
cwoAoSmbxStatsBytesWriteClient	The total number of bytes written by SMB AO clients since it was started.

cwoAoSmbxStats MIB Objects for LAN and WAN Information

Table 27: SMB AO LAN and WAN MIB Objects

MIB Object	Description
cwoAoSmbxStatsRdL4SignWANBytes	The total number of Layer 4 (L4) optimized signed bytes read from WAN by SMB AO since the SMB AO was started. L4 optimization includes TFO, DRE and LZ optimizations.
cwoAoSmbxStatsWrL4SignWANBytes	The total number of Layer 4 (L4) optimized signed bytes written to WAN by SMB AO since SMB AO was started. L4 optimization includes TFO, DRE and LZ optimizations.
cwoAoSmbxStatsRdSignLANBytes	The total number of signed bytes read from LAN by SMB AO since the SMB AO was started.

MIB Object	Description
cwoAoSmbxStatsWrSignLANBytes	The total number of original signed bytes written to LAN by SMB AO since SMB AO was started. **Note** Discontinuities in the values of these counters can occur at re-initialization of the SMB AO. The last discontinuity time is indicated by the value of cwoAoStatsLastResetTime for the SMB AO.

cwoAoSmbxStats MIB Objects for RTT, Response Time, and File Information

Table 28: SMB RTT, Response Time, and File Information MIB Objects

MIB Object	Description
cwoAoSmbxStatsRTT	The total round trip time (RTT) for all SMB connections since it was started.
cwoAoSmbxStatsTotalRespTimeSaving	The total response time saved due to SMB AO optimizations since it was started.
cwoAoSmbxStatsOpenFiles	The number of files currently opened by the SMB AO.

cwoAoSmbxStats MIB Objects for SMB Requests Information

Table 29: SMB Requests MIB Objects

MIB Object	Description
cwoAoSmbxStatsProcessedReqs	The total number of requests processed by the SMB AO since it was started.
cwoAoSmbxStatsActiveReqs	The total number of active requests getting processed by the SMB AO.
cwoAoSmbxStatsTotalRemoteReqs	The total number of SMB requests sent to the remote file server since the SMB AO was started.
cwoAoSmbxStatsTotalLocalReqs	The total number of SMB requests served locally by the SMB AO since it was started.
cwoAoSmbxStatsRemoteAvgTime	The average duration of time taken by the SMB AO to process all remote requests since it was started.
cwoAoSmbxStatsLocalAvgTime	The average duration of time taken by the SMB AO to process all local requests since it was started.

SNMP (ENTITY, ISNMP, and SNMP MIB Groups)

This section describes the following SNMP MIB groups:

- ENTITY-MIB: Represents multiple logical entities supported by a single SNMP agent. This MIB is documented in RFC 2737. The following objects are supported:

- entityPhysicalGroup
- entityLogicalGroup
- entConfigChange

- SNMP-FRAMEWORK-MIB: Facilitates remote configuration and administration of the SNMP entity. This MIB is documented in RFC 2571.
- SNMP-NOTIFICATION-MIB: Contains objects for the remote configuration of the parameters used by an SNMP entity for the generation of notifications. This MIB is documented in RFC 3413.
- SNMP-TARGET-MIB: Provides information about specifying targets of management operations for notification filtering and for proxy forwarding. This MIB is documented in RFC 3413.
- SNMP-USM-MIB: Provides information on the User-based Security Model.
- SNMP-VACM-MIB: Provides information on the View-based Access Control Model.
- SNMPv2-MIB: For this MIB group, WAAS supports the following MIB objects:
  - **coldStart**: Signifies that the SNMPv2 entity, acting in an agent role, is reinitializing itself and that its configuration may have been altered.
  - **linkUp**: The link up trap/notification.
  - **linkDown**: The link down trap/notification.
  - **authenticationFailure**: Signifies that the SNMPv2 entity, acting in an agent role, has received a protocol message that is not properly authenticated. While all implementations of the SNMPv2 must be capable of generating this trap, the snmpEnableAuthenTraps object indicates whether this trap will be generated.

This MIB is documented in RFC 1907.

TFO (CISCO-WAN-OPTIMIZATION-MIB)

The CISCO-WAN-OPTIMIZATION-MIB group displays information about the status and statistics associated with optimization and application accelerators.

This section describes the **cwoTfoStats** MIB objects, and contains the following topics:

About TFO Statistics MIB Objects

Cisco WAAS uses a variety Transport Flow Optimization (TFO) features to optimize TCP traffic intercepted by the Cisco WAAS devices. TFO protects communicating clients and servers from negative WAN conditions, such as bandwidth constraints, packet loss, congestion, and retransmission.

cwoTfoStats MIB Object for TFO Load Status

Table 30: TFO Load Status MIB Object

MIB Object	Description
cwoTfoStatsLoadStatus	Displays the current TFO load status: • **Unknown (1)**: TFO is in an unknown state, not active or disabled. • **Green (2)**: TFO is operating normally, within acceptable load limits. • **Yellow (3)**: TFO is overloaded, and new connections received may not be optimized. • **Red (4)**: TFO is not working properly, and both existing and new connections may not be optimized. **Note** If **cwoTfoStatsLoadStatus** shows **Unknown (1)**, **Yellow (3)** or **Red (4)**, then the TFO is either overloaded or has some other error condition, and no optimization can occur at any other level, such as DRE, LZ, or AO.
cwoTfoStatsTotalOptConn	The total number of connections optimized by the specified AO since it was started or since its statistics were last reset.
cwoTfoStatsTotalNormalClosedConn	The total number of optimized TCP connections that were closed normally since TFO was started, or since its statistics were last reset.
cwoTfoStatsResetConn	The total number of optimized TCP connections that have been reset since TFO was started or since its statistics were last reset.

cwoTfoStats MIB Objects for TFO Summary Connection Information

Table 31: TFO Summary Connection MIB Objects

MIB Object	Description
cwoTfoStatsLoadStatus	Displays the current TFO load status: • **Unknown (1)**: TFO is in an unknown state, not active or disabled. • **Green (2)**: TFO is operating normally, within acceptable load limits. • **Yellow (3)**: TFO is overloaded, and new connections received may not be optimized. • **Red (4)**: TFO is not working properly, and both existing and new connections may not be optimized. **Note** If **cwoTfoStatsLoadStatus** shows **Unknown (1)**, **Yellow (3)** or **Red (4)**, then the TFO is either overloaded or has some other error condition, and no optimization can occur at any other level, such as DRE, LZ, or AO.
cwoTfoStatsTotalOptConn	The total number of connections optimized by the specified AO since it was started or since its statistics were last reset.
cwoTfoStatsTotalNormalClosedConn	The total number of optimized TCP connections that were closed normally since TFO was started, or since its statistics were last reset.
cwoTfoStatsResetConn	The total number of optimized TCP connections that have been reset since TFO was started or since its statistics were last reset.

cwoTfoStats MIB Objects for TFO Current Connection Information

Table 32: TFO Current Connection MIB Objects

MIB Object	Description
cwoTfoStatsActiveOptConn	The number of active TCP connections that are getting optimized.
cwoTfoStatsMaxActiveConn	The maximum number of active TCP connections that the specified device can optimize.
cwoTfoStatsActivePTConn	The number of active pass-through TCP connections.
cwoTfoStatsActiveOptTCPPlusConn	The number of active TCP connections going through TCP plus other optimization.
cwoTfoStatsActiveOptTCPOnlyConn	The number of active TCP connections going through TCP optimization only.

MIB Object	Description
cwoTfoStatsActiveOptTCPPrepConn	The number of active TCP connections that were originated by an accelerator to acquire data in anticipation of its future use.
cwoTfoStatsStatsActiveADConn	The number of current active TCP connections in the auto-discovery state.
cwoTfoStatsReservedConn	The number of TCP connections that are reserved for the MAPI accelerator.
cwoTfoStatsPendingConn	The number of TCP connections that are pending in the queue of connections to be optimized.

Downloading MIB Files

You can download the MIB files for most of the MIBS that are supported by a device that is running the WAAS software from the following Cisco FTP site:

ftp://ftp.cisco.com/pub/mibs/v2

You can download the RFC1213-MIB file (for MIB-II) from the following Cisco FTP site:

ftp://ftp.cisco.com/pub/mibs/v1

The MIB objects that are defined in each MIB are described in the MIB files at the above FTP sites and are self-explanatory.

Using MIBs to Monitor Cisco WAAS

This section contains usage examples and sample output for using MIB files to monitor WAAS:

Using MIBs to Display Alarm Status

This section provides usage examples and sample output for Cisco WAAS alarm information. For more information on these MIBs, see Alarms (CISCO-CONTENT-ENGINE-MIB), on page 10.

- To verify that there are no alarms on the system, use **cceAlarm**:

```
CISCO-CONTENT-ENGINE-MIB::cceAlarmMinorCount.0 = Gauge32: 0
CISCO-CONTENT-ENGINE-MIB::cceAlarmMajorCount.0 = Gauge32: 0
CISCO-CONTENT-ENGINE-MIB::cceAlarmCriticalCount.0 = Gauge32: 0
```

Using MIBs to Display AO Information and Status

This section provides usage examples and sample MIB output for WAAS AO information and status. For more information on these MIBs, see AOs (CISCO-WAN-OPTIMIZATION-MIB), on page 10.

- To verify the configuration status of WAAS AOs, use **cwoAoStatsIsConfigured**:

```
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsIsConfigured."epm" = INTEGER: true(1)
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsIsConfigured."ica" = INTEGER: false(2)
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsIsConfigured."nfs" = INTEGER: true(1)
```

```
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsIsConfigured."smb" = INTEGER: true(1)
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsIsConfigured."ssl" = INTEGER: true(1)
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsIsConfigured."http" = INTEGER: true(1)
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsIsConfigured."mapi" = INTEGER: true(1)
```

- To verify the operational state of configured WAAS AOs, use **cwoAoStatsOperationState**:

```
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsOperationalState."epm" = INTEGER: normalRunning(3)
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsOperationalState."ica" = INTEGER: shutdown(1)
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsOperationalState."nfs" = INTEGER: normalRunning(3)
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsOperationalState."smb" = INTEGER: normalRunning(3)
CISCO-WAN-OPTIMIZATION-MIB::cwoAoStatsOperationalState."ssl" = INTEGER: normalRunning(3)
```

Using MIBs to Display DRE Cache and Performance Information

This section provides usage examples and sample MIB output for DRE cache and performance information.

For overview information on these MIBs, see DRE Cache (CISCO-WAN-OPTIMIZATION-MIB), on page 18 andDRE Performance (CISCO-WAN-OPTIMIZATION-MIB), on page 19.

- To verify if DRE is operational and the DRE is in a usable state (the DRE states are **Initializing**, **Usable**, **Failed**), use **cwoDreCacheStatsStatus**:

```
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsStatus.0 = STRING: Usable
```

- To display DRE cache age, use **cwoDreCacheStatsAge**:

```
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsAge.0 = STRING: 5d17h
```

Note On both branch and datacenter devices, the cache age should provide an effective capacity-to-reduction ratio. It is important that you baseline this value and set triggers according to your specific use case.For a datacenter device, the cache age should be approximately 5 to 7 days. However, there are scenarios where your cache age could be much lower and Cisco WAAS is still providing a very good reduction ratio; for example, in replication or backup scenarios. For a branch device, the cache age in practice will likely be more than 5 to 7 days.

- To display DRE cache information, including the portion of the disk space allocated for DRE cache, the age of the oldest data unit the data block, and the amount of data units replaced in the last hour, use **cwoDreCacheStats** MIB objects:

```
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsTotal.0 = Counter64: 77822 MB
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsUsed.0 = Gauge32: 96 percent
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsDataUnitUsage.0 = Counter64: 0 MB
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsReplacedOneHrDataUnit.0 = Counter64: 0 MB
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsDataUnitAge.0 = STRING: 0s
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsSigblockUsage.0 = Counter64: 1695 MB
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsReplacedOneHrSigblock.0 = Counter64: 0 MB
CISCO-WAN-OPTIMIZATION-MIB::cwoDreCacheStatsSigblockAge.0 = STRING: 14d17h
```

- To display compression ratio values, use **cwoDrePerfStats** MIB objects. For datacenter devices, it is especially useful to view Encode compression ratio values, and for branch devices, it is especially useful to view Decode compression ratio values.

```
CISCO-WAN-OPTIMIZATION-MIB::cwoDrePerfStatsEncodeCompressionRatio.0 = Gauge32: 9 percent
CISCO-WAN-OPTIMIZATION-MIB::cwoDrePerfStatsDecodeCompressionRatio.0 = Gauge32: 51 percent
```

- To display compression latency values, use **cwoDrePerfStats** MIB objects. For datacenter devices, it is especially useful to view Encode compression latency values, and for branch devices, it is especially useful to view Decode compression latency values.

```
CISCO-WAN-OPTIMIZATION-MIB::cwoDrePerfStatsEncodeCompressionLatency.0 = Counter64: 0 ms
CISCO-WAN-OPTIMIZATION-MIB::cwoDrePerfStatsDecodeCompressionLatency.0 = Counter64: 0 ms
```

Note Set a baseline for the latency value. If the latency value begins to deviate higher than normal, it could indicate a potential disk problem or failing disk, or it could indicate that a new traffic pattern is driving higher than normal disk input/output.

- To display the average size of all the messages handled by DRE during encoding or decoding, use **cwoDrePerfStats** MIB objects:

```
CISCO-WAN-OPTIMIZATION-MIB::cwoDrePerfStatsEncodeAvgMsgSize.0 = STRING: 1991 B
CISCO-WAN-OPTIMIZATION-MIB::cwoDrePerfStatsDecodeAvgMsgSize.0 = STRING: 1082 B
```

Using MIBs to Display Interface Information

This section provides usage examples and sample MIB output for interface information—description, status, and transmission errors and discards. For more information on these MIBs, see Interfaces (IF-MIB), on page 20.

- To check the up/down status of your interfaces, use **ifDescr**, **ifAdminStatus**, and **ifOperStatus**.

```
IF-MIB::ifDescr.1 = STRING: GigabitEthernet 0/0
IF-MIB::ifDescr.2 = STRING: GigabitEthernet 0/1
IF-MIB::ifAdminStatus.1 = INTEGER: up(1)
IF-MIB::ifAdminStatus.2 = INTEGER: up(1)
IF-MIB::ifOperStatus.1 = INTEGER: up(1)
IF-MIB::ifOperStatus.2 = INTEGER: down(2)
```

- To check if there are any transmission-related errors which could point to L1 and L2 problems (e.g. bad cable or speed/duplex mismatch on connected switch/router), use **ifInErrors** and **ifInDiscards**.

```
IF-MIB::ifInErrors.1 = Counter32: 0
IF-MIB::ifInErrors.2 = Counter32: 0
IF-MIB::ifOutErrors.1 = Counter32: 0
IF-MIB::ifOutErrors.2 = Counter32: 0
IF-MIB::ifInDiscards.1 = Counter32: 0
IF-MIB::ifInDiscards.2 = Counter32: 0
IF-MIB::ifOutDiscards.1 = Counter32: 0
IF-MIB::ifOutDiscards.2 = Counter32: 0
```

Using MIBs to Display TFO Information

This section provides usage examples and sample MIB output for TFO information and status. For more information on these MIBs, see TFO (CISCO-WAN-OPTIMIZATION-MIB), on page 28.

This section contains the following topics:

Performing Trend and Baseline Analysis with TFO MIBs

Before you begin

To be able to assess what normal load and benefits Cisco WAAS provides for your network, we recommend that you perform some trend and baseline analysis. Then, based on the results, you can create traps and alerts if the counters are above or below your defined thresholds, whichever is appropriate for the specific counter.

Procedure

Step 1 To verify key connection information, use the following MIB to verify the maximum number of connections the system can optimize.

```
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsMaxActiveConn.0 = Counter64: 750
```

Step 2 Use the following MIB object to verify the total number of active optimized connections:

```
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsActiveOptConn.0 = Counter64: 21
```

Step 3 After verifying the maximum number of connections and the total active optimized connections, you can do one of the following:

- Set an alert in your monitoring tool.

Or

- Set an SNMP trap if the number gets close to the limit on a consistent basis.

For example, the WAAS poll interval is every 5 minutes. An alert is triggered if within a 1-hour or 4-hour period the total number of active optimized connections crosses 90% of the maximum number of connections the system can optimize 10 times.

For how to set an SNMP trap, see Enabling SNMP Traps, on page 37.

Displaying Connection Information Using cwoTfoStats

To display connection information, use **cwoTfoStats** MIB objects:

```
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsTotalNormalClosedConn.0 = 0
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsPendingConn.0 = 0
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsReservedConn.0 = 0
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsResetConn.0 = 0
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsOptConn.0 = 0
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsMaxActiveConn.0 = 0
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsTotalOptConn.0 = 0
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsActivePTConn.0 = 0
```

Displaying TFO Auto-Discovery and Load Status Information Using cwoTfoStats

To display TFO auto-discovery and load status information, use **cwoTfoStats**:

```
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsActiveADConn.0 = 0
CISCO-WAN-OPTIMIZATION-MIB::cwoTfoStatsLoadStatus.0 = INTEGER: green(2)
```

Note If the TFO load status shows **Unknown(1)**, **Yellow(3)** or **Red(4)**, then the TFO is overloaded or has some other error condition, and no optimization can occur at any other level, such as DRE, LZ, or AO.

Enabling the SNMP Agent on a WAAS Device

By default, the SNMP agent on WAAS devices is disabled and an SNMP community string is not defined. The SNMP community string is used as a password for authentication when accessing the SNMP agent on a WAAS device. To be authenticated, the Community Name field of any SNMP message sent to the WAAS device must match the SNMP community string defined on the WAAS device.

The SNMP agent on a WAAS device is enabled when you define the SNMP community string on the device. The WAAS Central Manager GUI allows you to define the SNMP community string on a device or device group.

If the SNMPv3 protocol is going to be used for SNMP requests, the next step is to define an SNMP user account that can be used to access a WAAS device through SNMP. For more information on how to create an SNMPv3 user account on a WAAS device, see Creating an SNMP User, on page 47.

Checklist for Configuring SNMP

The following is a checklist for enabling SNMP monitoring on a Cisco WAAS device or device group.

Procedure

	Command or Action	Purpose
Step 1	Prepare for SNMP monitoring.	For more information, see Preparing for SNMP Monitoring, on page 36.
Step 2	Select the SNMP traps that you want to enable.	The Cisco WAAS Central Manager provides a wide-range of traps that you can enable on a Cisco WAAS device or device group. To define additional traps, see Defining SNMP Triggers to Generate User-Defined Traps, on page 40.
Step 3	Specify the SNMP host that receives the SNMP traps.	Specify the SNMP host to that the WAAS device or device group should send their traps to. You can specify multiple hosts so different WAAS devices send traps to different hosts.

	Command or Action	Purpose
		For more information, see Specifying the SNMP Host, on page 42.
Step 4	Specify the SNMP community string.	Specify the SNMP community string so external users can read or write to the MIB. For more information, see Specifying the SNMP Community String, on page 43.
Step 5	Set up SNMP views.	To restrict an SNMP group to a specific view, you must create a view that specifies the MIB subtree that you want the group to view. For more information, see Creating SNMP Views, on page 45.
Step 6	Create an SNMP group.	You must set up an SNMP group if are going to create any SNMP users or want to restrict a group to view a specific MIB subtree. For more information, see Creating an SNMP Group, on page 46.
Step 7	Create an SNMP user.	If the SNMPv3 protocol is going to be used for SNMP requests, you must create at least one SNMPv3 user account on the Cisco WAAS device in order for the WAAS device to be accessed through SNMP. For more information, see Creating an SNMP User, on page 47.
Step 8	Configure SNMP contact settings.	For more information, see Configuring SNMP Contact Settings, on page 49.

Preparing for SNMP Monitoring

Before you configure your Cisco WAAS network for SNMP monitoring, complete the following preparation tasks:

- Set up the SNMP host (management station) that the Cisco WAAS devices will use to send SNMP traps.
- Determine if all your Cisco WAAS devices will be sending traps to the same host, or to different hosts. Write down the IP address or hostname of each SNMP host.
- Obtain the community string used to access the SNMP agents.
- Determine if you want to create SNMP groups so you can restrict views by group.
- Determine what additional SNMP traps you need.
- Clock synchronization between the devices in a Cisco WAAS network is important. On each Cisco WAAS device, be sure to set up a Network Time Protocol (NTP) server to keep the clocks synchronized.

Enabling SNMP Traps

Procedure

Step 1 From the Cisco WAAS Central Manager menu, choose **Devices >** *device-name* or **Device Groups >** *device-group-name*.

Step 2 Choose **Configure > Monitoring > SNMP > General Settings**.

The **SNMP General Settings** window appears. The "SNMP General Settings" table desribes the fields in this window.

Figure 2: SNMP General Settings Window

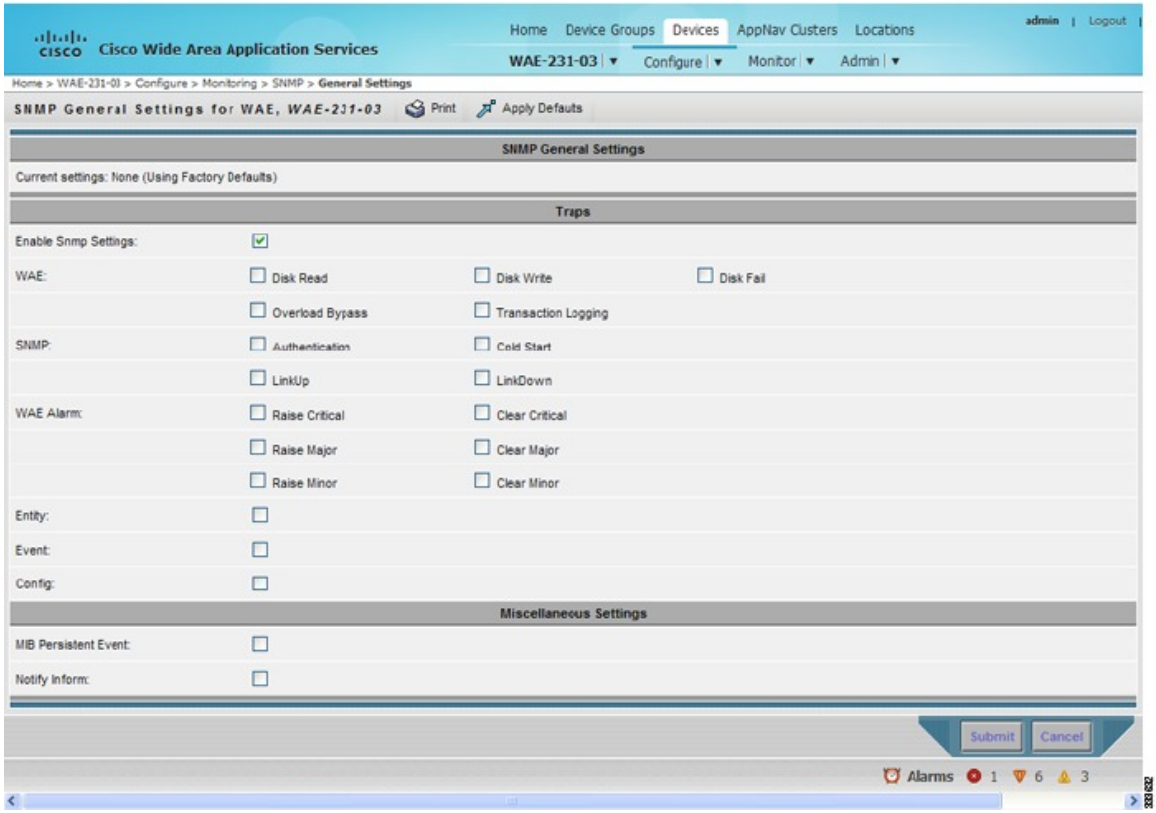


Table 33: SNMP General Settings

GUI Parameter	Function
Traps	
Enable Snmp Settings	Enables SNMP traps.

GUI Parameter	Function
WAE	Enables SNMP WAE traps: • **Disk Read**: Enables disk read error trap. • **Disk Write**: Enables disk write error trap. • **Disk Fail**: Enables disk failure error trap. • **Overload Bypass**: Enables WCCP overload bypass error trap. • **Transaction Logging**: Enables transaction log write error trap.
SNMP	Enables SNMP-specific traps: • **Authentication**: Enables authentication trap. • **Cold Start**: Enables cold start trap. • **LinkUp**: Link up trap. • **LinkDown**: Link down trap.
WAE Alarm	Enables WAE alarm traps: • **Raise Critical**: Enables raise-critical alarm trap • **Clear Critical**: Enables clear-critical alarm trap • **Raise Major**: Enables raise-major alarm trap • **Clear Major**: Enables clear-major alarm trap • **Raise Minor**: Enables raise-minor alarm trap • **Clear Minor**: Enables clear-minor alarm trap
Entity	Enables SNMP entity traps.
Event	Enables the Event MIB.
Config	Enables CiscoConfigManEvent error traps.
Miscellaneous Settings	
MIB Persistent Event	Enables persistence for the SNMP Event MIB. (This check box is not shown when the selected device is a Central Manager.)

GUI Parameter	Function
Notify Inform	Enables the SNMP notify inform request. Inform requests are more reliable than traps but consume more resources in the router and in the network. Traps are unreliable because the receiver does not send any acknowledgment when it receives a trap. The sender cannot determine if the trap was received. However, an SNMP manager that receives an inform request acknowledges the message with an SNMP response. If the sender never receives a response, the inform request can be sent again. Thus, informs are more likely to reach their intended destinations. **Note** To send an SNMPv3 inform message, you must configure at least one SNMPv3 user with a remote SNMP ID option on the WAAS device. The SNMP ID is entered in octet string form. For example, if the IP address of a remote SNMP entity is 192.147.142.129, then the octet string would be 00:00:63:00:00:00:a1:c0:93:8e:81. (Colons will be removed in the show running-config command output.)

Step 3 Check the appropriate check boxes to enable SNMP traps.

Step 4 Click **Submit**.

A **Click Submit to Save** message appears in red next to the current settings when there are pending changes to be saved after you have applied default or device group settings. You can also revert to the previously configured window settings by clicking **Reset**. The **Reset** button is visible only when you apply default or device group settings to change the current device settings but the settings have not yet been submitted.

What to do next

To enable SNMP traps from the CLI, run the **snmp-server enable traps** global configuration command.

To control access to the SNMP agent by an external SNMP server, run the **snmp-server access-list** global configuration command to apply an SNMP ACL.

Consider the following guidelines:

- If you are using an SNMP server ACL, you must permit the loopback interface.
- If you override the device group settings from the **SNMP General Settings** window, the Cisco WAAS Central Manager deletes the SNMP community, SNMP group, SNMP user, SNMP view, and SNMP host settings. You are asked to confirm this behavior.
- To define additional SNMP traps for other MIB objects of interest to your particular configuration, see Defining SNMP Triggers to Generate User-Defined Traps, on page 40.

Defining SNMP Triggers to Generate User-Defined Traps

Procedure

Step 1 From the Cisco WAAS Central Manager menu, choose **Devices** > *device-name* or **Device Groups** > *device-group-name*.

Step 2 Choose **Configure** > **Monitoring** > **SNMP** > **Trigger**.

The **SNMP Trigger List Entries** window appears. The columns in this window are the same as the parameters described in the "Creating New SNMP Trigger Settings" table

Step 3 In the taskbar, click the **Create New SNMP Trigger List Entry** icon.

The **Creating New SNMP Trigger** window appears. The "Creating New SNMP Trigger Settings" table describes the fields in this window.

Table 34: Creating New SNMP Trigger Settings

GUI Parameter	Function
Trigger Name	Custom-defined name for the notification trigger that you want to monitor.
MIB Name	MIB variable name of the object that you want to monitor.
Wild Card	(Optional) Check this check box if the MIB Name value is a wildcard. Note that this check box is disabled when editing the SNMP Trigger.
Frequency	Number of seconds (60–600) to wait between trigger samples.
Test	Test used to trigger the SNMP trap. Choose one of the following tests: • **absent**: A specified MIB object that was present at the last sampling is no longer present as of the current sampling. • **equal**: The value of the specified MIB object is equal to the specified threshold. • **greater-than**: The value of the specified MIB object is greater than the specified threshold value. • **less-than**: The value of the specified MIB object is less than the specified threshold value. • **on-change**: The value of the specified MIB object has changed since the last sampling. • **present**: A specified MIB object is present as of the current sampling that was not present at the previous sampling. • **threshold**: Configures a maximum and a minimum threshold for a MIB object.

GUI Parameter	Function
Sample Type	(Optional) Sample type, as follows: • **absolute**: The test is evaluated against a fixed integer value between zero and 2147483647. • **delta**: The test is evaluated against the change in the MIB object value between the current sampling and the previous sampling.
Threshold Value	Threshold value of the MIB object. This field is not used if absent, on-change, or present is chosen in the Test drop-down list.
MIB Var1MIB Var2MIB Var3	(Optional) Names of up to three alternate MIB variables to add to the notification. Validation of these names is not supported, so be sure to enter them correctly.
Comments	Description of the trap.

Step 4 In the appropriate fields, enter the MIB name, frequency, test, sample type, threshold value, and comments.

Note You can create valid triggers only on read-write and read-only MIB objects. If you create a trigger on a read-create MIB object, it is deleted from the Central Manager configuration after one one data feed poll cycle.

Step 5 Click **Submit**.

What to do next

Consider the following guidelines:

- The new SNMP trigger is listed in the **SNMP Trigger List** window.
- To edit an SNMP trigger, click the **Edit** icon next to the MIB name in the **SNMP Trigger List** Entries window.
- To delete an SNMP trigger, click the **Edit** icon next to the MIB name and then clicking the **Delete** taskbar icon.
- If you delete any of the default SNMP triggers, they will be restored after a reload.
- When you upgrade a WAE from an earlier version to the 6.0 version, all triggers are deleted.When you upgrade the Cisco WAAS Central Manager to Cisco WAAS Version 6.0, all the Device Group triggers will be copied to a WAE running a previous software version (if any) and all the Device Group triggers will be deleted. Also the Trigger Aggregate Settings will be set to false for all the WAES (running a version earlier than 6.0) that are being managed by the Cisco WAAS Central Manager (running Cisco WAAS Version 6.0 or later). This ensures that the DG triggers are no longer applied to any of the devices running a version earlier than Cisco WAAS Version 6.0.
- If you are using an SNMP server ACL, you must permit the loopback interface.
- When you downgrade a WAE from a Cisco WAAS Version 6.0 to an earlier release, all the IPv6 configurations will be removed. All the triggers and the monitor user configurations are deleted.
- To define SNMP traps from the CLI, run the **snmp trigger** global configuration command .

- To control access to the SNMP agent by an external SNMP server, run the **snmp-server access-list** global configuration command to apply an SNMP ACL.

Aggregating SNMP Triggers

An individual WAE device can have custom SNMP triggers defined and can belong to device groups that have other custom SNMP triggers defined.

In the SNMP Trigger List Entries window, the Aggregate Settings radio button controls how SNMP triggers are aggregated for an individual device, as follows:

- Choose **Yes** if you want to configure the device with all custom SNMP triggers that are defined for itself and for device groups to which it belongs.
- Choose **No** if you want to limit the device to just the custom SNMP triggers that are defined for itself.

When you change the setting, you get the following confirmation message: "This option will take effect immediately and will affect the device configuration. Do you wish to continue?" Click **OK** to continue.

Specifying the SNMP Host

Before you begin

Hosts are listed in the order in which they have been created. The maximum number of SNMP hosts that can be created is eight.

Procedure

Step 1 From the Cisco WAAS Central Manager menu, choose **Devices** > *device-name* or **Device Groups** > *device-group-name*.

Step 2 Choose **Configure** > **Monitoring** > **SNMP** > **Host**.

The **SNMP Hosts** window appears.

Step 3 In the taskbar, click the **Create New SNMP Host** icon.

The **Creating New SNMP Host** window appears. The following table describes the fields in this window.

Table 35: SNMP Host Settings

GUI Parameter	Function
Trap Host	Hostname or IP address of the SNMP trap host that is sent in SNMP trap messages from the WAE. This is a required field and now supports IPv6 addresses.
Community/User	Name of the SNMP community or user (64 characters maximum) that is sent in SNMP trap messages from the WAE. This is a required field.

GUI Parameter	Function
Authentication	Security model to use for sending notification to the recipient of an SNMP trap operation. Choose one of the following options from the drop-down list: • **No-auth**: Sends notification without any security mechanism. • **v2c**: Sends notification using Version 2c security. • **v3-auth**: Sends notification using SNMP Version 3 AuthNoPriv. • **v3-noauth**: Sends notification using SNMP Version 3 NoAuthNoPriv security. • **v3-priv**: Sends notification using SNMP Version 3 AuthPriv security.
Retry	Number of retries (1–10) allowed for the inform request. The default is 2 tries.
Timeout	Timeout for the inform request in seconds (1–1000). The default is 15 seconds.

Step 4 Enter the hostname or IP address of an SNMP trap host, SNMP community or user name, security model to send notification, and retry count and timeout for inform requests.

Step 5 Click **Submit**.

To specify the SNMP host from the CLI, run the **snmp-server host** global configuration command.

Specifying the SNMP Community String

Before you begin

An SNMP community string is the password used to access an SNMP agent that resides on Cisco WAAS devices. There are two types of community strings: group and read-write. Community strings enhance the security of your SNMP messages.

Community strings are listed in the order in which they have been created. The maximum number of SNMP communities that can be created is ten. By default, an SNMP agent is disabled, and a community string is not configured. When a community string is configured, it permits read-only access to all agents by default.

Procedure

Step 1 From the Cisco WAAS Central Manager menu, choose **Devices** > *device-name* or **Device Groups** > *device-group-name*.

Step 2 Choose **Configure** > **Monitoring** > **SNMP** > **Community**.

The **SNMP Community Strings** window appears.

Step 3 In the taskbar, click the **Create New SNMP Community** String icon.

The **Creating New SNMP Community String** window appears. The "SNMP Community Settings" table describes the fields in this window.

Table 36: SNMP Community Settings

GUI Parameter	Function
Community	Community string used as a password for authentication when you access the SNMP agent of the WAE. The **Community Name** field of any SNMP message sent to the WAE must match the community string defined here in order to be authenticated. Entering a community string enables the SNMP agent on the WAE. You can enter a maximum of 64 characters in this field. This is a required field.
Group name/rw	Group to which the community string belongs. The **Read/Write** option allows a read or write group to be associated with this community string. The **Read/Write** option permits access to only a portion of the MIB subtree. Choose one of the following three options from the drop-down list: • **None**: Choose this option if you do not want to specify a group name to be associated with the community string. The Group Name field remains disabled if you select this option. • **Group**: Choose this option if you want to specify a group name. • **Read/Write**: Choose this option if you want to allow read-write access to the group associated with a community string. The Group Name field remains disabled if you select this option. This is a required field.
Group Name	Name of the group to which the community string belongs. You can enter a maximum of 64 characters in this field. This field is available only if you have chosen the **Group** option in the previous field.

Step 4 In the appropriate fields, enter the community string, choose whether or not read-write access to the group is allowed, and enter the group name.

Step 5 Click **Submit**.

To configure a community string from the CLI, run the **snmp-server community** global configuration command.

Creating SNMP Views

Before you begin

To restrict a group of users to view a specific MIB tree, you must create an SNMP view using the Cisco WAAS Central Manager GUI. Once you create the view, you need to create an SNMP group and SNMP users that belong to this group as described in later sections.

Views are listed in the order in which they have been created. The maximum number of views that can be created is ten.

Procedure

Step 1 To create a Version 2 SNMP (SNMPv2) MIB view: From the Cisco WAAS Central Manager menu, choose **Devices >** *device-name* or **Device Groups >** *device-group-name*.

Step 2 Choose **Configure > Monitoring > SNMP > View**. The SNMP Views window appears.

Step 3 In the taskbar, click the **Create New View** icon.

The **Creating New SNMP View** window appears. The "SNMPv2 View Settings" table describes the fields in this window.

Table 37: SNMPv2 View Settings

GUI Parameter	Function
Name	String representing the name of this family of view subtrees (64 characters maximum). The family name must be a valid MIB name such as ENTITY-MIB. This is a required field.
Family	Object identifier (64 characters maximum) that identifies a subtree of the MIB. This is a required field.
View Type	View option that determines the inclusion or exclusion of the MIB family from the view. Choose one of the following two options from the drop-down list: • **Included**: The MIB family is included in the view. • **Excluded**: The MIB family is excluded from the view.

Step 4 In the appropriate fields, enter the view name, the family name, and the view type.

Step 5 Click **Submit**.

Step 6 Create an SNMP group that will be assigned to this view as described in the section that follows.

To create an SNMP view from the CLI, run the **snmp-server view** global configuration command.

Creating an SNMP Group

Before you begin

You must set up an SNMP group if you are going to create any SNMP users or want to restrict a group of users to view a specific MIB subtree.

Groups are listed in the order in which they have been created. The maximum number of SNMP groups that can be created is ten.

Procedure

Step 1 From the Cisco WAAS Central Manager menu, choose **Devices** > *device-name* or **Device Groups** > *device-group-name*.

Step 2 Choose **Configure** > **Monitoring** > **SNMP** > **Group**.

The **SNMP Group Strings for WAE** window appears.

Step 3 In the taskbar, click the **Create New SNMP Group String** icon.

The **Creating New SNMP Group String for WAE** window appears. The "SNMP Group Settings" table describes the fields in this window.

Table 38: SNMP Group Settings

GUI Parameter	Function
Name	Name of the SNMP group. You can enter a maximum of 64 characters. This is a required field.
Sec Model	Security model for the group. Choose one of the following options from the drop-down list: • **v1**: Version 1 security model (SNMP Version 1 [noAuthNoPriv]). • **v2c**: Version 2c security model (SNMP Version 2 [noAuthNoPriv]). • **v3-auth**: User security level SNMP Version 3 AuthNoPriv. • **v3-noauth**: User security level SNMP Version 3 noAuthNoPriv. • **v3-priv**: User security level SNMP Version 3 AuthPriv. **Note** A group defined with the SNMPv1 or SNMPv2c security model should not be associated with SNMP users; they should only be associated with the community strings.
Read View	Name of the view (a maximum of 64 characters) that enables you only to view the contents of the agent. By default, no view is defined. In order to provide read access to users of the group, a view must be specified. For information on creating SNMP views, see Creating SNMP Views, on page 45.

GUI Parameter	Function
Write View	Name of the view (a maximum of 64 characters) that enables you to enter data and configure the contents of the agent. By default, no view is defined. For information on creating SNMP views, see Creating SNMP Views, on page 45.
Notify View	Name of the view (a maximum of 64 characters) that enables you to specify a notify, inform, or trap. By default, no view is defined. For information on creating SNMP views, see Creating SNMP Views, on page 45.

Step 4 In the appropriate fields, enter the SNMP group configuration name, the security model, and the names of the read, write, and notify views.

Step 5 Click **Submit**.

Step 6 Create SNMP users that belong to this new group as described in the section that follows.

To create an SNMP group from the CLI, run the **snmp-server group** global configuration command.

Creating an SNMP User

Before you begin

Users are listed in the order in which they have been created. The maximum number of users that can be created is ten.

Procedure

Step 1 From the Cisco WAAS Central Manager menu, choose **Devices** > *device-name* or **Device Groups** > *device-group-name*.

Step 2 Choose **Configure** > **Monitoring** > **SNMP** > **User**. A list of SNMP users for the device or device group appears.

Step 3 In the taskbar, click the **Create New SNMP User** icon.

The **Creating New SNMP User** window appears. The "SNMP User Settings" table describes the fields in this window.

Table 39: SNMP User Settings

GUI Parameter	Function
Name	String representing the name of the user (32 characters maximum) who can access the device or device group. This is a required field.
Group	Name of the group (64 characters maximum) to which the user belongs. This is a required field.

GUI Parameter	Function
Remote SNMP ID	Globally unique identifier for a remote SNMP entity (10 to 64 characters). To send an SNMPv3 message to the WAE, at least one user with a remote SNMP ID must be configured on the WAE. The SNMP ID must be entered in octet string format. Only hexadecimal characters and the colon (:) are allowed in this field. If any colons appear in the entered string, they are removed when the page is submitted.
Authentication Algorithm	Authentication algorithm that ensures the integrity of SNMP packets during transmission. Choose one of the following three options from the drop-down list: • **No-auth**: Requires no security mechanism to be turned on for SNMP packets. • **MD5**: Provides authentication based on the hash-based Message Authentication Code Message Digest 5 (HMAC-MD5) algorithm. • **SHA**: Provides authentication based on the hash-based Message Authentication Code Secure Hash (HMAC-SHA) algorithm.
Authentication Password	Alphanumeric string (256 characters maximum) that configures the user authentication (HMAC-MD5 or HMAC-SHA) password. The number of characters is adjusted to fit the display area if it exceeds the limit for display. The following special characters are not supported: space, backwards single quote (`), single quote ('), double quote ("), pipe (|), or question mark (?). This field is optional if the **no-auth** option is chosen for the authentication algorithm. Otherwise, this field must contain a value.
Confirmation Password	Authentication password for confirmation. The reentered password must be the same as the one entered in the previous field.
Private Password	Alphanumeric string (256 characters maximum) that configures the authentication (HMAC-MD5 or HMAC-SHA) parameters to enable the SNMP agent to receive packets from the SNMP host. The number of characters is adjusted to fit the display area if it exceeds the limit for display. The following special characters are not supported: space, backwards single quote (`), double quote ("), pipe (|), or question mark (?). **Note** For SNMPv3 users using Cisco WAAS Software Version 6.x and later, the private password must be a minimum of 8 alphanumeric characters and a maximum of 256 characters.
Confirmation Password	Private password for confirmation. The reentered password must be the same as the one entered in the previous field.

Step 4 In the appropriate fields, enter the username, the group to which the user belongs, the engine identity of the remote entity to which the user belongs, the authentication algorithm used to protect SNMP traffic from tampering, the user authentication parameters, and the authentication parameters for the packet.

Step 5 Click **Submit**.

What to do next

To create an SNMP user from the CLI, run the **snmp-server user** global configuration command.

Additionally, if you want to set up a monitor user to monitor the configured triggers, you can select it from the **Monitor User Settings** drop-down box. Any SNMP V3 user can be configured as a Monitor User. All the SNMP users created with a group having V3 authentication other than v3-private are eligible to be a Monitor User. A monitor user cannot be deleted, while being in that role. Similarly the corresponding monitor user group also cannot be deleted when a monitor user is configured with that group.

To create a monitor user from the CLI, run the **snmp-server monitor user** global configuration command.

Configuring SNMP Asset Tag Settings

Procedure

Step 1 To configure SNMP asset tag settings, which create values in the CISCO-ENTITY-ASSET-MIB: From the Cisco WAAS Central Manager menu, choose **Devices >** *device-name* or **Device Groups >** *device-group-name*.

Step 2 Choose **Configure > Monitoring > SNMP > Asset Tag**.

The **SNMP Asset Tag Settings** window appears.

Step 3 In the **Asset Tag Name** field, enter a name for the asset tag.

Step 4 Click **Submit**.

To configure SNMP asset tag settings from the CLI, run the **asset tag** global configuration command.

Configuring SNMP Contact Settings

Procedure

Step 1 From the Cisco WAAS Central Manager menu, choose **Devices >** *device-name* or **Device Groups >** *device-group-name*.

Step 2 Choose **Configure > Monitoring > SNMP > Contact Information.**

The **SNMP Contact Settings** window appears.

Step 3 Enter a contact name and location in the provided fields.

Step 4 Click **Submit**.

To configure SNMP contact settings from the CLI, run the **snmp-server contact** global configuration command.

Configuring SNMP Trap Source Settings

Procedure

Step 1 From the Cisco WAAS Central Manager menu, choose **Devices >** *device-name*. This setting is not supported from device groups.

Step 2 Choose **Configure > Monitoring > SNMP > Trap Source**.

The **SNMP Trap Source Settings** window appears.

Step 3 From the **Trap Source** drop-down list, choose the interface to be used as the trap source. From the available physical, standby, and port-channel interfaces, only those with IP addresses are shown in the list. For Cisco vWAAS devices, virtual interfaces with assigned IP addresses are shown in the list.

Note An interface assigned as a trap source cannot be removed until it is unassigned as a trap source.

Step 4 Click **Submit**.

To configure SNMP trap source settings from the CLI, run the **snmp-server trap-source** global configuration command.