CONTENTS

CHAPTER 1

Configuring the Web Services Management Agent 1

Finding Feature Information 1

Prerequisites for Configuring WSMA 2

Restrictions for Configuring WSMA 2

Information About Configuring WSMA 2

Configuration WSMA Service 2

EXEC WSMA Service 3

Filesystem WSMA Service 4

Notification WSMA Service 5

Hello WSMA Service 6

Keepalive WSMA Service 6

WSMA Profiles 6

Service Listener WSMA Profile 7

Service Initiator WSMA Profile 7

SOAP 7

WSMA over SSHv2 7

WSMA over HTTP 8

WSMA ID 9

WSMA Security 10

WSMA Schema 10

Wireless WebUI 10

How to Configure WSMA 10

Enabling SSHv2 Using a Hostname and Domain Name 10

Enabling the HTTP Server 12

Enabling the HTTPS Server 13

Verifying the Status of the SSH Connection 15

Enabling a WSMA Service Initiator 16

Enabling a WSMA Service Listener 19
Enabling WSMA Services 21

Enabling WebUI 22

Configuring WSMA Exec Agent 22
Configuring WSMA Configuration Agent 24
Configuring WSMA File System 25
Configuring WSMA Notify System Agent 26
Configuring WSMA Listener Profile 27

Assigning WSMA IDs 28

Monitoring and Maintaining WSMA Services 29
Monitoring and Maintaining WSMA Profiles 31
Delivering WSMA Payloads 32

Configuration Examples for WSMA 36
Example: Enabling SSHv2 Using a Hostname and Domain Name 36
Example: Enabling SSHv2 Using RSA Keys 36
Example: Configuring a WSMA Service 37
Example: Configuring the WSMA Initiator Profile 37
Example: Configuring the WSMA Listener Profile 37
Example: Displaying WSMA Profile Parameters 37

Additional References 39

Feature Information for Web Services Management Agent 40

Glossary 42

CHAPTER 2

WSMA Enhancements for Wireless Management 43

Finding Feature Information 43

Restrictions for WSMA Enhancements for Wireless Management 43
Information About WSMA Enhancements for Wireless Management 44

Smart Back-off Reconnect Support 44
EXEC Agent–Response Payload Compression Support 44

How to Configure Smart Back-off Reconnect 44
Configuring Smart Back-off Reconnect 44

Configuration Examples for WSMA Enhancements for Wireless Management 45
Example: Configuring the Smart Back-off Reconnect 45

Additional References for WSMA Enhancements 46
Feature Information for WSMA Enhancements for Wireless Management 46
Configuring the Web Services Management Agent

The Web Services Management Agent (WSMA) defines a set of web services through which a network device can be managed, configuration data information can be retrieved, and new configuration data can be uploaded and manipulated. WSMA uses XML-based data encoding, that is transported by the Simple Object Access Protocol (SOAP) for the configuration data and protocol messages.

You can use WSMA over Secure Shell Version 2 (SSHv2), HTTP, or Secure HTTP (HTTPS) to access the entire Cisco CLI. Multiple WSMA clients can connect to the WSMA server running on Cisco software.

You can also use WSMA over SSHv2, HTTP, or HTTPS to initiate secure connections from Cisco software to applications over trusted and untrusted networks.

• Finding Feature Information, page 1
• Prerequisites for Configuring WSMA, page 2
• Restrictions for Configuring WSMA, page 2
• Information About Configuring WSMA, page 2
• How to Configure WSMA, page 10
• Configuration Examples for WSMA, page 36
• Additional References, page 39
• Feature Information for Web Services Management Agent, page 40
• Glossary, page 42

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.
Prerequisites for Configuring WSMA

• Every Web Services Management Agent (WSMA) agent must be associated with a WSMA profile to perform any operations. Otherwise, the WSMA agents cannot send or receive any messages.

• WSMA over Secure Shell Version 2 (SSHv2) requires that a vty line be available for each WSMA session.

Restrictions for Configuring WSMA

• Secure Shell Version 1 (SSHv1) is not supported; only SSHv2 is supported.

• You must be running a crypto image in order to configure SSH or Secure HTTP (HTTPS).

• Notification services are not supported for Web Services Management Agent (WSMA) over HTTP in listener mode.

• WSMA keepalive messages must be configured for config, exec, and filesys services for WSMA over HTTP in initiator mode.

Information About Configuring WSMA

Configuration WSMA Service

Web Services Management Agent (WSMA) is a family of embedded agents used by an external point-to-point management application to fully manage a Cisco networking device.

The Configuration WSMA agent provides services to change the configuration on Cisco devices and validates and applies a set of configuration commands to Cisco software. Any noninteractive configuration CLI command that can be applied using the Cisco console can also be applied using this WSMA. This service is available for all configuration CLI commands on the Cisco device. It treats a set of commands as a single operation.

Three types of configuration requests can occur:

• configTest—validates the syntax of the configuration data but does not apply the data to the running configuration.

• configApply—modifies the running configuration with the supplied configuration data. Use the action-on-fail attribute to specify the error handling to perform if an error is encountered when applying the configuration. The level of error information returned in the response can be controlled using the details attribute.

• configPersist—copies the running configuration to the startup configuration so that it persists across reloads.

The configuration WSMA service allows you to specify the CLI commands using either the XML Programmatic Interface (XML-PI) mode, or as direct CLI commands. Configuration WSMA service requests use the following modes and attributes:
• block mode—use the `<cli-config-data-block>` tag to encapsulate a multiline block of CLI commands.

• cmd mode—use the `<cli-config-data>` tag to encapsulate a block of configuration settings where each CLI line is individually delimited by `<cmd>` tags.

• XML-PI mode—use the `<xml-config-data>` tag to encapsulate processing instructions. This format is compatible with Cisco Enhanced Device Interface (EDI).

• action-on-fail—use this attribute to specify the action to perform when an error is encountered. You can specify the following action values:
  • stop—stops the execution on the first error but preserves the system state. If the execution is stopped, the configuration could be partially applied.
  • continue—ignores errors and continues implementing instructions.
  • rollback—stops processing at the first error and restores the configuration to the state before any configuration was applied. The rollback action value is enabled only if the `archive` command is configured.

• details—Use this attribute to control the level of error details. You can specify one of the following values:
  • brief—provides minimal detail in error responses.
  • errors—provides details on all error encountered.
  • all—provides the maximum level of details on errors.

EXEC WSMA Service

Web Services Management Agent (WSMA) is a family of embedded agents used by the point-to-point management application to fully manage a device.

The EXEC WSMA provides services to retrieve operational data from the Cisco device and handles EXEC mode command-line operations such as `show` commands and other diagnostic commands on Cisco devices. Interactive EXEC commands have Expect and Response tags to allow you to configure the exchange sequence. The service can retrieve `show` command operational data in XML-Programmatic Interface (PI) format and it allows remote reloading of the Cisco device.

EXEC WSMA service requests consist of a single EXEC mode command encapsulated in an `<execCLI>` tag with the following tags and attributes:

• `execTest`—validates the syntax of the EXEC command but does not run the command.

• `maxWait`—the maximum time to accumulate data and wait for the EXEC command to complete. Once the interval expires the operation stops and all accumulated data is sent in the response.

• `maxResponseSize`—the maximum number of bytes allowed in the body of the response. The default is 0 (infinity), and the range is 0 to $2^{31} - 1$. If the response exceeds the specified size, the operation stops and all accumulated data is sent in the response.

• `format`—returns the results of EXEC commands in XML-PI format. You must specify the path to the spec file on the Cisco file system. To use the global spec file command in the Cisco file system and still return XML-PI format results use the attribute `format=""`. 

EXEC WSMA Service
• xsd—sends the XML schema of the EXEC command instead of the output of the EXEC command when this value is set to 1.

• cmd—this mandatory tag contains the EXEC command to run.

• dialogue—this optional tag is used only for interactive EXEC commands. It specifies an expect and reply sequence. It includes a repeat attribute that is used if there are multiple identical expect and reply sequences.

• expect—the prompt the system expects. The value need not be an exact match to the specified string. The string match has two attributes:
  • caseSensitive—set this attribute to true to do case-sensitive match. The default is to be case-sensitive.
  • match—set to leading, trailing, embedded, or exact. The default is an exact match.

• reply—the answer to the prompt if it matches.

The order and number of the dialog elements must match the actual prompts seen or the EXEC call will fail. All dialogs must be run, otherwise an error message is seen.

Filesystem WSMA Service

Web Services Management Agent (WSMA) is a family of embedded agents used by an external point-to-point management application to fully manage a Cisco networking device.

The Filesystem WSMA service provides services to manage files on the Cisco device. It copies and validates files between local and remote file systems. This agent can be used to list directories, upgrade the software image running on the device, and delete files. File copies can be validated using a Message Digest 5 (MD5) checksum if available.

There are three types of Filesystem requests:

• fileList—provides a directory listing.

• fileDelete—specifies a list of files to be deleted using the deleteFileList tag.

• fileCopy—enables the copying of files to and from the local file system. The file is copied outside of the WSMA transport mechanism using the protocol specified in the srcURL and dstURL attributes. This copy process is similar to copying a file using the EXEC CLI shell. However, this process performs additional validation checks that are not available in the EXEC shell.

The fileCopy request option has the following attributes:

• filesize—This mandatory attribute is the number of bytes to be copied. If the filesize attribute does not match the size of the copied file, the operation fails.

• erase—This mandatory attribute is a Boolean True/False. If this attribute is set to TRUE, the file system is erased before the filecopy operation is performed. This attribute is useful where the new image does not fit on the disk.

• overwrite—This mandatory attribute is a Boolean True/False. If this attribute is set to TRUE, the current file on the file system is overwritten by the new file being copied.

• retries—This optional attribute specifies the number of times the file copy is attempted in the absence of a permanent failure.
retry-interval—This optional attribute specifies the time interval between a fileCopy retry. The default is 10 seconds.

The fileCopy request option has the following tags:

- srcURL—Specifies the URL and protocol to use for the file transfer.
- dstURL—Specifies the URL of the location to which the file is copied.
- validationInfo—Specifies an optional MD5 checksum to provide additional security during downloading.
- deleteFilelist—Specifies an optional tag that is the same as fileDelete. The file list is deleted before the copy starts.

There are three types of Filesystem responses:

- filesystemList—This response is a listing of every disk, directory, and file on the device. This response also includes additional information such as name, size, and flags. You can use this information to calculate the used and free space on the device to assist image download.
- fileDeleteStatus—This response provides an itemized response to the delete file request and displays the status of each file in the list and whether or not the file was deleted.
- fileCopyStatus—This response provides a report on whether the copy succeeded or failed. Success is indicated by an empty body with the success attribute set to 1 if the request succeeded.

The errorInfo response is seen only if the operation fails. The errorInfo response returns an error string of the error encountered that contains two error fields: errorCode and errorMessage.

The errorCode response details the possible error types and can include:

- BAD_PARAMETER—Indicates that at least one of the parameters on the request is invalid.
- INTERNAL_ERROR—Indicates that an unknown internal API error occurred.
- NO_MEMORY—Indicates that the system has run out of memory.
- OPERATION_FAILED—Indicates that the operation did not complete. The error message provides details.
- NO_MEMORY—Indicates that memory allocation failed.
- PERMISSION_DENIED—Indicates that authorization failed.

## Notification WSMA Service

Web Services Management Agent (WSMA) is a family of embedded agents used by an external point-to-point management application to fully manage a Cisco networking device.

The Notification WSMA service collects configuration-change events and forwards the details to the management application that has subscribed to get the notifications.

Multiple management applications can receive the notifications by connecting to a listener profile. Each management application must explicitly subscribe to the notifications and can turn notification on or off on the profile without affecting the operation of other connected management applications. If a connection drops notifications are turned off.
Notifications are not cached or stored. If no management application is connected when an event happens then there is no record of that event.

Notification requests have three attributes:

- correlator—used to coordinate the acknowledgment to the request.
- type—a string representing the types of notifications to enable on the session. The only supported string is configChange.
- activate—turns notification on or off by sending the value 0 (off) or 1 (on).

Notification responses have the following attributes:

- correlator—used to coordinate the acknowledgment to the request.
- success—this attribute is set to True if the requested notification type is successfully enabled or disabled.

**Hello WSMA Service**

Web Services Management Agent (WSMA) is a family of embedded agents used by the point-to-point management application to fully manage a device.

When a new WSMA session is established, the Cisco device sends a Hello message containing the WSMA ID and a list of WSMA services available on the session. The remote management application can query this information by sending a WSMA Hello request to the Cisco device.

This service is enabled by default on every WSMA profile.

**Keepalive WSMA Service**

Web Services Management Agent (WSMA) is a family of embedded agents, used by the point-to-point management application to fully manage a device.

If a WSMA profile is configured to use keepalive messages, and if no WSMA service request has been received for the configured keepalive interval, the Cisco device sends a Keepalive request on the WSMA session. If the number of keepalive requests sent exceeds the configured retries, the WSMA session is closed.

A keepalive request has one attribute, correlator. The correlator attribute is a number that starts at 1 and increments each time a keepalive request is sent on a session. The correlator value used in a keepalive response must match the value in a keepalive request.

**WSMA Profiles**

Web Services Management Agent (WSMA) needs input from external management applications to cause actions on the device. A physical transport protocol must be configured and associated to a WSMA to allow the WSMA to communicate with external management applications. The transport protocol and an encapsulation together form a WSMA profile. Any WSMA agent must be associated with a specific WSMA profile to perform valid operations. WSMA profiles demultiplex requests to the appropriate WSMA.

WSMA profiles work as a transport termination point and allow transport and XML encapsulation parameters to be configured:
• The configurable encapsulations for WSMA are Simple Object Access Protocol (SOAP) 1.1 and SOAP 1.2.

• The transportation mechanisms for WSMA are Secure Shell (SSH), HTTP, and Secure HTTP (HTTPS). This mechanism opens listening sockets for listeners on the device or connecting sockets for clients on the device.

Service Listener WSMA Profile

The service listener is a type of Web Services Management Agent (WSMA) profile that listens for incoming connections and accepts devices from allowed addresses or accepted user IDs. Define an access list to configure accepted addresses.

Configure accepted user IDs by defining the transport method that the service listener listens for. The transport method (Secure Shell (SSH), HTTP, or Secure HTTP [HTTPS]) enforces the specific user ID that is accepted.

WSMA listener profiles cannot access Cisco devices that are located behind a firewall.

Service Initiator WSMA Profile

The service initiator is a type of Web Services Management Agent (WSMA) profile that initiates secure connections from Cisco devices to management applications over trusted and untrusted networks.

The service initiator creates a dynamic socket that attempts to stay connected to a configured server address. Each initiator can be configured with retry, keepalive, timeout, and reconnect settings. In addition, each initiator can specify a backup connection to use if the primary connection fails.

The service initiator allows WSMA to initiate connections to devices behind a firewall or Network Address Translation (NAT), and in Zero Touch Deployment (ZTD) networks.

SOAP

Simple Object Access Protocol (SOAP) is an industry-standard protocol to exchange XML data between applications. It defines a common mechanism to handle corrupted XML messages. It has a header mechanism to collate metadata associated with a transaction.

SOAP 1.1 and SOAP 1.2 have different schema definitions. They can coexist with no impact on the other. Cisco software has both SOAP 1.1 and SOAP 1.2 libraries. SOAP has mechanisms to handle XML framing and operational errors in a generic manner, allowing greater interoperability of XML-based applications.

WSMA over SSHv2

To run the WSMA over SSHv2 feature, the Web Services Management Agent (WSMA) agent must be configured to use a service profile that is using Secure Shell (SSH) as a transport method. The figure below shows a basic WSMA over SSHv2 network configuration. The client and server exchange keys for security and password encryption. The user ID and password of the SSHv2 session running WSMA are used for authorization and authentication purposes. The user privilege level is enforced and the client session may not
have full access to the WSMA operations if the privilege level is not high enough. If authentication, authorization, and accounting (AAA) is configured, the AAA service is used as if a user had established an SSH session directly to the device. Using the existing security configuration makes the transition to WSMA almost seamless. Once the client has been successfully authenticated, the client invokes the SSH connection protocol and the SSH session is established. After the SSH session is established, the user or application invokes WSMA as an SSH subsystem. The default name for the subsystem is "wsma."

Figure 1: WSMA over SSHv2

SSHv2 runs on top of a reliable transport layer and provides strong authentication and encryption capabilities. SSHv2 provides a means to securely access and securely execute commands on another computer over a network.

Service listeners do not support SSHv1. The configuration for the SSHv2 server is similar to the configuration for SSHv1. Use the `ip ssh version` command to specify which version of SSH you want to configure. If you do not configure this command, SSH by default runs in compatibility mode; that is, both SSHv1 and SSHv2 connections are honored.

Note

SSHv1 is a protocol that has never been defined in a standard. If you do not want your device to fall back to the undefined protocol (version 1), you should use the `ip ssh version` command and specify version 2.

Use the `ip ssh rsa keypair-name` command to enable an SSH connection using Rivest, Shamir, and Adelman (RSA) keys that you have configured. If you configure the `ip ssh rsa keypair-name` command with a key-pair name, SSH is enabled if the key pair exists, or SSH will be enabled if the key pair is generated later. If you use this command to enable SSH, you need not configure a hostname and a domain name.

WSMA over HTTP

To run the WSMA over HTTP feature, you must configure the Web Services Management Agent (WSMA) agent to use a service profile that is using either HTTP or Secure HTTP (HTTPS) as a transport. For HTTPS, the client and server exchange keys for security and password encryption. The user ID and password of the HTTP or HTTPS session running WSMA are used for authorization and authentication purposes. The user privilege level is enforced and the client session may not have full access to the WSMA operations if the privilege level is not high enough. If authentication, authorization, and accounting (AAA) is configured, the AAA service is used as if a user had established an SSH session directly to the device. Using the existing security configuration makes the transition to WSMA almost seamless. After the HTTP or HTTPS session is established, the user or application invokes WSMA as an HTTP path. The default name for the path is "/wsma."
When you use HTTP as the transport for a initiator profile, the WSMA Notification service is available without additional configuration. However, to use the config, exec, and filesys services, you must first configure keepalive messages on the initiator profile. When keepalive messages are configured, the Cisco device can periodically send a request to the remote WSMA application, which allows the remote HTTP server the opportunity to send a WSMA request.

When you use HTTP as the transport for a listener profile, the WSMA Notification service is not supported since the Cisco device acting as a HTTP server cannot send HTTP requests, it can only respond to HTTP requests.

**HTTP**

HTTP is a reliable request/response protocol that runs on top of a reliable transport layer. HTTPS provides strong authentication and encryption capabilities.

HTTP is configured with the `ip http server` command and HTTPS is configured using the `ip http secure-server` command.

**Access Lists**

You can configure access lists for use with a service listener. An access list is a sequential collection of permit and deny conditions that applies to IP addresses. The Cisco software tests addresses against the conditions in an access list one by one. The first match determines whether the software accepts or rejects the address. Because the software stops testing conditions after the first match, the order of the conditions is critical. If no conditions match, the software rejects the address.

The two main tasks involved in using access lists are as follows:

1. Creating an access list by specifying an access list number or name and access conditions.
2. Applying the access list to interfaces or terminal lines.

**WSMA ID**

The Web Services Management Agent (WSMA) IDs allow Cisco networking devices to have unique IDs. Unique IDs are important in a Network Address Translation (NAT) or Dynamic Host Configuration Protocol (DHCP) network where all the device IP addresses are locally significant. In this type of deployment, the WSMA ID can be used to give each device a globally unique ID.

The WSMA ID can be explicitly configured based on other properties of the device such as:

- The hardware serial number
- The hostname
- The IP address of an interface
- The MAC address of an interface
- A user-defined string

Whenever the WSMA ID changes, all WSMA sessions are disconnected to protect the management applications from synchronizing the state dynamically.
WSMA Security

Web Services Management Agent (WSMA) security is integrated with authentication, authorization, and accounting (AAA) configuration of Cisco software. The AAA associations configured on the transport layer are used by the WSMA.

WSMA is designed for point-to-point operation and works over an encrypted transport. The security on the transport layer identifies and authenticates the users.

WSSE

The Web Services Security Header (WSSE) is the Simple Object Access Protocol (SOAP) security extension. The WSMA profiles can be configured to expect or ignore additional security headers in the SOAP messages depending on the deployment mode. If WSMA is configured to contain a security header, the format of the header is as per the SOAP security extension, WSSE.

SOAP enforces authentication using the WSSE header. Any authentication errors are reported as SOAP faults. The authenticated message is passed on to the WSMA, which checks for the authorization level of the user before applying any operation. Authorization errors are reported as a WSMA error response.

If WSMA profiles are configured without the WSSE, then the security header is ignored and the transport login credentials are used for authentication. If the WSSE is expected, then the details of the security header are used to authenticate the user. If the security header is missing, the incoming message is discarded and a SOAP fault is issued.

WSMA Schema

Each Web Services Management Agent (WSMA) service publishes its XML schema. The schema describe the XML messages that the specific WSMA service can understand and execute. The WSMA schema define the entire data required to execute an operation and ensure operations can be performed identically regardless of the type of transport used to carry the message.

Wireless WebUI

Wireless WebUI allows you to monitor and configure the wireless features. WSMA configurations are required to enable wireless WebUI. When WebUI sends a request, WSMA, parser, and AAA components process and authorize WebUI requests.

How to Configure WSMA

Enabling SSHv2 Using a Hostname and Domain Name

Perform this task to enable Secure Shell Version 2 (SSHv2) on your device using a hostname and domain name.
**SUMMARY STEPS**

1. enable
2. configure terminal
3. hostname hostname
4. ip domain-name name
5. crypto key generate rsa
6. ip ssh \[timeout seconds | authentication-retries integer\]
7. ip ssh version 2
8. end

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>enable</td>
</tr>
<tr>
<td>Example:</td>
<td>Device&gt; enable</td>
</tr>
<tr>
<td></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>configure terminal</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# configure terminal</td>
</tr>
<tr>
<td></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>hostname hostname</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config)# hostname host1</td>
</tr>
<tr>
<td></td>
<td>Configures a hostname for your device.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>ip domain-name name</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config)# ip domain-name example.com</td>
</tr>
<tr>
<td></td>
<td>Configures a domain name for your device.</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>crypto key generate rsa</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config)# crypto key generate rsa</td>
</tr>
<tr>
<td></td>
<td>Enables the SSH server for local and remote authentication.</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>ip ssh [timeout seconds</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config)# ip ssh timeout 120</td>
</tr>
<tr>
<td></td>
<td>(Optional) Configures SSH control variables on your device.</td>
</tr>
</tbody>
</table>
### Enabling the HTTP Server

Perform this task to enable the HTTP server. The HTTP server is disabled by default. Once the HTTP server is enabled, you can configure optional server characteristics.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. ip http server
4. ip http authentication {aaa | local}

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable   | Enables privileged EXEC mode.  
| Example:            |         |
| Device> enable      | • Enter your password if prompted. |
| **Step 2** configure terminal | Enters global configuration mode. |
| Example:            |         |
| Device# configure terminal | |
| **Step 3** ip http server | Enables the HTTP 1.1 server, including the Cisco web browser user interface. |
| Example:            |         |
| Device(config)# ip http server | |
Enabling the HTTPS Server

To disable the standard HTTP server and configure the HTTPS server with Secure Socket Layer (SSL) version 3.0, complete this task.

**Before You Begin**

If a certificate authority is to be used for certification, you should declare the certificate authority (CA) trustpoint on the routing device before enabling the secure HTTP server.

**SUMMARY STEPS**

1. enable
2. show ip http server status
3. configure terminal
4. no ip http server
5. ip http secure-server
6. ip http secure-port port-number
7. ip http secure-ciphersuite [3des-ede-cbc-sha] [rc4-128-sha] [rc4-128-md5] [des-cbc-sha]
8. ip http secure-client-auth
9. ip http secure-trustpoint name
10. end
11. show ip http server secure status

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
</tbody>
</table>
## Enabling the HTTPS Server

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>(Optional) Displays the status of the HTTP server.</td>
</tr>
<tr>
<td>show ip http server status</td>
<td>• If you are unsure whether the secure HTTP server is supported in the software image you are running, enter this command and look for the line “HTTP secure server capability: {Present</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>• This command displays the status of the standard HTTP server (enabled or disabled).</td>
</tr>
<tr>
<td>Device# show ip http server status</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Disables the standard HTTP server.</td>
</tr>
<tr>
<td>no ip http server</td>
<td><strong>Note</strong> When enabling the HTTPS server you should always disable the standard HTTP server to prevent insecure connections to the same services. This is a precautionary step (typically, the HTTP server is disabled by default).</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device(config)# no ip http server</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>Enables the HTTPS server.</td>
</tr>
<tr>
<td>ip http secure-server</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device(config)# ip http secure-server</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>(Optional) Specifies the port number that should be used for the HTTPS server.</td>
</tr>
<tr>
<td>ip http secure-port <em>port-number</em></td>
<td>• The default port number is 443.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>• Valid options are 443 or any number in the range 1025 to 65535.</td>
</tr>
<tr>
<td>Device(config)# ip http secure-port 1025</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td>(Optional) Specifies the CipherSuites (encryption algorithms) that should be used for encryption over the HTTPS connection.</td>
</tr>
<tr>
<td>ip http secure-ciphersuite [3des-ede-cbc-sha] [rc4-128-sha] [rc4-128-md5] [des-cbc-sha]</td>
<td>• This command allows you to restrict the list of CipherSuites that the server offers the connecting clients. For example, you may want to allow only the most secure CipherSuite to be used.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>• Unless you have a reason to specify the CipherSuites that should be used, or you are unfamiliar with the details of these CipherSuites, you should leave this command unconfigured and let the server and client negotiate the CipherSuite that they both support (this is the default).</td>
</tr>
<tr>
<td>Device(config)# ip http secure-ciphersuite rc4-128-sha rc4-128-md5</td>
<td></td>
</tr>
<tr>
<td>Step 8</td>
<td>Command or Action</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------</td>
</tr>
<tr>
<td></td>
<td><code>ip http secure-client-auth</code></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config)# ip http secure-client-auth</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 9</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>ip http secure-trustpoint name</code></td>
<td>Specifies the CA trustpoint that should be used to obtain an X.509v3 security certificate and to authenticate the connecting client’s certificate.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config)# ip http secure-trustpoint trustpoint-01</td>
<td>- Use of this command assumes you have already declared a CA trustpoint using the <code>crypto pki trustpoint</code> command and associated submode commands.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Use the same trustpoint name that you used in the associated <code>crypto pki trustpoint</code> command.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 10</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>end</code></td>
<td>Ends the current configuration session and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config)# end</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 11</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>show ip http server secure status</code></td>
<td>Displays the status of the HTTP secure server configuration.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device# show ip http server secure status</td>
<td></td>
</tr>
</tbody>
</table>

## Verifying the Status of the SSH Connection

To display the status of the Secure Shell (SSH) connection on your device, use the `show ssh` and `show ip ssh` commands.

The `ip ssh version` command can be used for troubleshooting your SSH configuration. By changing versions, you can determine which SSH version has a problem.

### SUMMARY STEPS

1. `enable`
2. `show ssh`
3. `show ip ssh`
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><code>Device&gt; enable</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>show ssh</code></td>
<td>Displays the status of SSH server connections.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>Device# show ssh</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td><code>show ip ssh</code></td>
<td>Displays the version and configuration data for SSH.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>Device# show ip ssh</code></td>
<td></td>
</tr>
</tbody>
</table>

### Examples

The following sample output from the `show ssh` command displays status about SSHv2 connections:

```
Device# show ssh
Connection Version Mode Encryption Hmac State
Username
1 2.0 IN aes128-cbc hmac-md5 Session started lab
1 2.0 OUT aes128-cbc hmac-md5 Session started lab
%No SSHv1 server connections running.
```

The following sample output from the `show ip ssh` command displays the version of SSH that is enabled, the authentication timeout values, and the number of authentication retries:

```
Device# show ip ssh
SSH Enabled - version 2.0
Authentication timeout: 120 secs; Authentication retries: 3
```

## Enabling a WSMA Service Initiator

### Before You Begin

If you configure service initiator over HTTP or Secure HTTP (HTTPS), you must configure keepalive settings so that the Cisco device can periodically send a HTTP Request to the remote Web Services Management Agent (WSMA) application, thus giving the remote WSMA application a chance to send WSMA requests.
SUMMARY STEPS

1. enable
2. configure terminal
3. wsma profile initiator profile-name
4. encap {soap11 | soap12}
5. [backup] transport {http | https | ssh} remote-host [initiator-port-number] path path-name [user] username [0 | 6] password | source source-interface
6. keepalive seconds [retries number]
7. idle-timeout minutes
8. max-message message-size
9. backup hold minutes
10. backup excluded seconds
11. reconnect seconds
12. stealth
13. wsse
14. end

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Device&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> wsma profile initiator profile-name</td>
<td>Creates a service initiator and enters WSMA initiator configuration mode.</td>
</tr>
<tr>
<td>Example: Device(config)# wsma profile initiator prof1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> encap {soap11</td>
<td>soap12}</td>
</tr>
<tr>
<td>Example: Device(config-wsma-init)# encap soap12</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> [backup] transport {http</td>
<td>https</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
</tr>
<tr>
<td>username [0</td>
<td>6</td>
</tr>
<tr>
<td>password source-interface</td>
<td>• To ensure that the Cisco device allows the remote WSMA application to send WSMA requests, keepalive messages must be enabled on HTTP and HTTPS initiator connections.</td>
</tr>
</tbody>
</table>

**Step 6**

**keepalive seconds [retries number]**

Example:

Device(config-wsma-init)# keepalive 100 retries 10

(Optional) Specifies the amount of time (in minutes) to keep the session alive in the absence of any data traffic.

**Step 7**

**idle-timeout minutes**

Example:

Device(config-wsma-init)# idle-timeout 345

(Optional) Specifies the maximum receive message size (from 1 to 2000 kilobytes).

**Step 8**

**max-message message-size**

Example:

Device(config-wsma-init)# max-message 290

(Optional) Sets the time (in minutes) that the WSMA profile remains connected to the backup transport configuration.

**Step 9**

**backup hold minutes**

Example:

Device(config-wsma-init)# backup hold 233

(Optional) Sets the time (in seconds) that the WSMA profile must wait before attempting to connect to the backup transport configuration after a connection is lost.

**Step 10**

**backup excluded seconds**

Example:

Device(config-wsma-init)# backup excluded 30

(Optional) Specifies the time for the WSMA initiator profile to wait before attempting to reconnect a session.
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 12</strong> stealth</td>
<td>(Optional) Configures the service to not send Simple Object Access Protocol (SOAP) fault messages in response to corrupted XML messages.</td>
</tr>
<tr>
<td>Example: Device(config-wsma-init)# stealth</td>
<td></td>
</tr>
<tr>
<td><strong>Step 13</strong> wsse</td>
<td>(Optional) Enables the Web Services Security Header (WSSE) for a WSMA profile.</td>
</tr>
<tr>
<td>Example: Device(config-wsma-init)# wsse</td>
<td></td>
</tr>
<tr>
<td>• By default, the WSSE is enabled. Enter the no wsse command to disable the WSSE.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 14</strong> end</td>
<td>Ends the current configuration session and returns you to privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Device(config-wsma-init)# end</td>
<td></td>
</tr>
</tbody>
</table>

### Enabling a WSMA Service Listener

**Before You Begin**

Before you configure service listener over SSH, you must first configure SSH. For more information, see the "Enabling SSHv2 Using a Hostname and Domain Name" section.

Before you configure service listener over HTTP, you must first configure HTTP. For more information, see the "Enabling HTTP Server" section and the "Enabling the HTTPS Server" section.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. wsma profile listener *profile-name*
4. encap {soap11 | soap12}
5. transport {http | https [path *pathname*] | ssh [subsys *subsys-name*]}
6. idle-timeout minutes
7. max-message message-size
8. keepalive seconds [retries number]
9. acl *acl-number*
10. stealth
11. wsse
12. end
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Step 1 | enable | Enables privileged EXEC mode.  
  **Example:**  
  Device> enable |
| Step 2 | configure terminal | Enters global configuration mode. |
| Step 3 | wsma profile listener `profile-name` | Creates a service listener and enters WSMA listener configuration mode.  
  **Example:**  
  Device(config)# wsma profile listener prof1 |
| Step 4 | `encap {soap11 | soap12}` | (Optional) Configures an encapsulation for the service listener profile.  
  **Example:**  
  Device(config-wsma-listen)# encap soap12 |
| Step 5 | `transport {http | https [path pathname] | ssh [subsys subsys-name]}` | Defines a transport configuration for the Web Services Management Agent (WSMA) profile.  
  **Example:**  
  Device(config-wsma-listen)# transport ssh subsys wsma |
| Step 6 | `idle-timeout minutes` | (Optional) Specifies the amount of time (in minutes) to keep the session alive in the absence of any data traffic.  
  **Example:**  
  Device(config-wsma-listen)# idle-timeout 345 |
| Step 7 | `max-message message-size` | (Optional) Specifies the maximum receive message size (from 1 to 2000 kilobytes).  
  **Example:**  
  Device(config-wsma-listen)# max-message 290 |
| Step 8 | `keepalive seconds [retries number]` | (Optional) Enables keepalive messages and configures interval and retry values for a WSMA profile.  
  **Example:**  
  Device(config-wsma-listen)# keepalive 100 retries 10  
  - Keepalive messages are not sent on HTTP or HTTPS listener connections. |
<table>
<thead>
<tr>
<th>Step 9</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>acl acl-number</code></td>
<td>(Optional) Defines the access control list (ACL) group to use.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-wsma-listen)# acl 34</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 10</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>stealth</code></td>
<td>(Optional) Configures the service to not send Simple Object Access Protocol (SOAP) fault messages in response to corrupted XML messages.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-wsma-listen)# stealth</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 11</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>wsse</code></td>
<td>(Optional) Enables the Web Services Security Header (WSSE) for a WSMA profile.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-wsma-listen)# wsse</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• By default, the WSSE is enabled. Enter the <code>no wsse</code> command to disable the WSSE.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 12</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>end</code></td>
<td>Ends the current configuration session and returns you to privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-wsma-listen)# end</td>
<td></td>
</tr>
</tbody>
</table>

### Enabling WSMA Services

Perform this task to enable a specific Web Services Management Agent (WSMA) service and associate it with a profile.

**Before You Begin**

A WSMA initiator or listener profile must be configured and enabled.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `wsma agent {config | exec | filesys | notify} profile profile-name`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
</tbody>
</table>
### Enabling WebUI

**Note**

The WebUI is configured by default on the device.

HTTP and Secure HTTP servers must be enabled.

To enable WebUI, perform the tasks described in the following sections:

### Configuring WSMA Exec Agent

**SUMMARY STEPS**

1. enable
2. configure terminal
3. wsma agent exec
4. profile profile-name
5. profile httpslistener
6. end
### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>wsma agent exec</td>
<td>Starts the WSMA EXEC agent and enters WSMA EXEC agent configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config)# wsma agent exec</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>profile profile-name</td>
<td>Creates a profile for WSMA EXEC agent.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(wsma-exec-agent)# profile httplistener</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>profile httplistener</td>
<td>Creates a profile for WSMA EXEC agent.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(wsma-exec-agent)# profile httplistener</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>end</td>
<td>Exits WSMA EXEC agent configuration mode and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(wsma-exec-agent)# end</td>
<td></td>
</tr>
</tbody>
</table>
Configuring WSMA Configuration Agent

SUMMARY STEPS

1. enable
2. configure terminal
3. wsma agent config
4. profile httplistener
5. profile httpslistener
6. end

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable | Enables privileged EXEC mode.  
   - Enter your password if prompted. |
| Example: Device> enable | |
| **Step 2** configure terminal | Enters global configuration mode. |
| Example: Device# configure terminal | |
| **Step 3** wsma agent config | Starts the WSMA configuration agent and enters WSMA configuration agent mode. |
| Example: Device(config)# wsma agent config | |
| **Step 4** profile httplistener | Creates a profile for WSMA config agent. |
| Example: Device(wsma-config-agent)# profile httplistener | |
| **Step 5** profile httpslistener | Creates a profile to use for WSMA config agent. |
| Example: Device(wsma-config-agent)# profile httpslistener | |
### Purpose

Command or Action | Purpose
---|---
**Step 6**  
*Example:*  
Device(wsma-config-agent)# end | Exits WSMA configuration agent mode and returns to privileged EXEC mode.

---

## Configuring WSMA File System

### SUMMARY STEPS

1. `enable`  
2. `configure terminal`  
3. `wsma agent filesys`  
4. `profile httplistener`  
5. `profile httpslistener`  
6. `exit`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**  
*Example:*  
Device> enable | Enables privileged EXEC mode.  
- Enter your password if prompted.  
| **Step 2**  
*Example:*  
Device# configure terminal | Enters global configuration mode. |
| **Step 3**  
*Example:*  
Device(config)# wsma agent filesys | Starts the WSMA file system agent and enters WSMA file system agent mode. |
| **Step 4**  
*Example:*  
Device(wsma-filesys-agent)# profile httplistener | Creates a profile to use for WSMA filesys agent. |
### Command or Action | Purpose
--- | ---
**Step 5** | **profile httpslistener**
*Example:*
Device(wsma-filesys-agent)# profile httpslistener

**Step 6** | **exit**
*Example:*
Device(wsma-filesys-agent)# exit

---

### Configuring WSMA Notify System Agent

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `wsma agent notify`
4. `profile httpslistener`
5. `profile httpslistener`
6. `exit`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | **enable**
*Example:*
Device> enable

- Enter your password if prompted. |
| **Step 2** | **configure terminal**
*Example:*
Device# configure terminal |
| **Step 3** | **wsma agent notify**
*Example:*
Device(config)# wsma agent notify |
| **Step 4** | **profile httpslistener**
*Example:*
Device(wsma-filesys-agent)# profile httpslistener |

---
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 4</strong></td>
<td>Creates a profile to use for WSMA notify agent.</td>
</tr>
<tr>
<td>profile <code>httplistener</code></td>
<td>Example: <code>Device(wsma-notify-agent)# profile httplistener</code></td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>Creates a profile to use for WSMA notify agent.</td>
</tr>
<tr>
<td>profile <code>httpslistener</code></td>
<td>Example: <code>Device(wsma-notify-agent)# profile httpslistener</code></td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>Exits WSMA notify agent mode and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td>exit</td>
<td>Example: <code>Device(wsma-notify-agent)# exit</code></td>
</tr>
</tbody>
</table>

### Configuring WSMA Listener Profile

**SUMMARY STEPS**

1. enable
2. configure terminal
3. `wsma profile listener httplistener`
4. `transport http`
5. exit
6. `wsma profile listener httpslistener`
7. `transport https`
8. exit

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>enable</td>
<td>Example: <code>Device&gt; enable</code></td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> wsma profile listener httplistener</td>
<td>Enables WSMA listener profile and enters WSMA listener configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device(config)# wsma profile listener httplistener</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> transport http</td>
<td>Defines a transport configuration for a WSMA profile</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device(config-wsma-listen)# transport http</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> exit</td>
<td>Exits WSMA listener configuration mode and returns to global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device(config-wsma-listen)# exit</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> wsma profile listener httpslistener</td>
<td>Enables WSMA listener profile and enters WSMA listener configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device(config)# wsma profile listener httpslistener</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> transport https</td>
<td>Defines a transport configuration for a WSMA profile</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device(config-wsma-listen)# transport https</td>
<td></td>
</tr>
<tr>
<td><strong>Step 8</strong> exit</td>
<td>Exits WSMA listener configuration mode and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Device(config-wsma-listen)# end</td>
<td></td>
</tr>
</tbody>
</table>

**Assigning WSMA IDs**

Perform this task to assign unique Web Services Management Agent (WSMA) IDs to Cisco networking devices.
SUMMARY STEPS

1. enable
2. configure terminal
3. wsma id {hardware-serial | hostname | ip-address interface-type | mac-address interface-type | string value}

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
</tbody>
</table>
| Example:
  Device> enable   |         |
| Step 2 configure terminal | Enters global configuration mode. |
| Example:
  Device# configure terminal |         |
| Step 3 wsma id {hardware-serial | hostname | ip-address interface-type | mac-address interface-type | string value} | Assigns unique WSMA IDs to Cisco networking devices. |
| Example:
  Device(config)# wsma id ip-address fastethernet 0/1 |         |

Monitoring and Maintaining WSMA Services

SUMMARY STEPS

1. enable
2. show wsma agent [config | exec | filesys | notify] [counters | schema]
3. debug wsma agent [config | exec | filesys | notify]
4. clear wsma agent [config | exec | filesys | notify] counters

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device&gt; enable</td>
<td></td>
</tr>
</tbody>
</table>

### Step 2
show wsma agent [config | exec | filesys | notify] [counters | schema]

Example:
Device# show wsma agent config counters

### Step 3
depbug wsma agent [config | exec | filesys | notify]

Example:
Device# debug wsma agent config

### Step 4
clear wsma agent [config | exec | filesys | notify] counters

Example:
Device# clear wsma agent filesys counters

#### Examples
The following example shows how to display the WSMA configuration agent counters. The counters return the following information:

- messages received—The total number of messages that were passed from the service profile into the WSMA.
- replies sent—The total number of reply messages sent to the services profile.
- faults—The number of faults that prevented a received message producing a reply.
- notifications—The total number of notification messages sent to the services profile.

Device# show wsma agent counters

WSMA Exec Agent Statistics:
messages received 0, replies sent 0, faults 0
WSMA Config Agent Statistics:
messages received 4, replies sent 4, faults 0
WSMA Filesys Agent Statistics:
messages received 1, replies sent 1, faults 0
WSMA Notification Agent Statistics:
config silent
messages received 0, replies sent 0, notifications sent 0, faults 0

The following example shows how to display the WSMA configuration schema:

Device# show wsma agent config schema
Monitoring and Maintaining WSMA Profiles

Perform this task to monitor and maintain Web Services Management Agent (WSMA) profiles.

SUMMARY STEPS

1. enable
2. show wsma profile [name profile-name] {connections | counters | schema}
3. debug wsma profile [listener | initiator]
4. clear wsma profile [profile-name] {connections | counters}

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>enable</td>
</tr>
<tr>
<td>Example:</td>
<td>Device&gt; enable</td>
</tr>
<tr>
<td></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>show wsma profile [name profile-name] {connections</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# show wsma profile connections</td>
</tr>
<tr>
<td></td>
<td>Displays the specified service profile connections, statistics counters, or schema.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>debug wsma profile [listener</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# debug wsma profile listener</td>
</tr>
<tr>
<td></td>
<td>Enables debugging of WSMA profiles.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>clear wsma profile [profile-name] {connections</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# clear wsma profile prof1 counters</td>
</tr>
<tr>
<td></td>
<td>Clears WSMA profile sessions or statistic counters.</td>
</tr>
</tbody>
</table>
Delivering WSMA Payloads

An XML payload is typically wrapped in a Simple Object Access Protocol (SOAP) message for data transportation. Without a correct design of SOAP messages, an XML payload may not be exchanged properly even if the payload follows a common XML schema. The XML payload over all transports is identical. Web Services Management Agent (WSMA) supports both SOAP1.1 and SOAP1.2. The SOAP header supports two modes of security, no wsse and wsse.

Use the following XML schema to deliver WSMA payloads:

**WSMA EXEC Request: Ping**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/
 xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/
 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 <SOAP:Body>
<request xmlns="urn:cisco:wsma-exec" correlator="01">
<execCLI>
<cmd>ping oz-dirt</cmd>
</execCLI>
</request>
</SOAP:Body>
</SOAP:Envelope>
```

**WSMA EXEC Response: Ping**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/
 xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/
 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 <SOAP:Body>
<response xmlns="urn:cisco:wsma-exec" correlator="01" success="1">
<execLog>
<dialogueLog>
<sent>ping oz-dirt</sent>
<received>Type escape sequence to abort. Sending 5, 100-byte ICMP Echos to 10.3.1.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms</received>
</dialogueLog>
</execLog>
</response>
</SOAP:Body>
</SOAP:Envelope>
```

**WSMA Config Request: CMD Data Model**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/
 xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/
 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 <SOAP:Body>
<request xmlns="urn:cisco:wsma-config" correlator="4.1">
<configApply details="all">
<config-data>
<cli-config-data>
<cmd>no cns config partial mixy</cmd>
<cmd>no stupid</cmd>
<cmd>no cns exec 80</cmd>
```
WSMA Config Response: CMD Data Model

<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/
xmlns:xsd=http://www.w3.org/2001/XMLSchema"
xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance">
  <SOAP:Body>
    <response xmlns="urn:cisco:wsma-config" correlator="4.1" success="1">
      <resultEntry lineNumber="1" cliString="no cns config partial mixy">
        <success change="NO_CHANGE" mode="IMMEDIATE" />
      </resultEntry>
      <resultEntry lineNumber="2" cliString="no stupid">
        <failure errorType="TEMPORARY" errorCode="PARSE_ERROR_NOMATCH" />
      </resultEntry>
      <resultEntry lineNumber="3" cliString="no cns exec 80 ">
        <success change="NO_CHANGE" mode="IMMEDIATE" />
      </resultEntry>
    </response>
  </SOAP:Body>
</SOAP:Envelope>

WSMA Config Request: Block Data Model

<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/
xmlns:xsd=http://www.w3.org/2001/XMLSchema"
xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance">
  <SOAP:Body>
    <request xmlns="urn:cisco:wsma-config" correlator="5.1">
      <configApply details="all">
        <config-data>
          <cli-config-data-block>no cns config partial mixy
          no stupid
          no cns exec 80</cli-config-data-block>
        </config-data>
      </configApply>
    </request>
  </SOAP:Body>
</SOAP:Envelope>

WSMA Config Response: Block Data Model

<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/
xmlns:xsd=http://www.w3.org/2001/XMLSchema"
xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance">
  <SOAP:Body>
    <response xmlns="urn:cisco:wsma-config" correlator="5.1" success="1">
      <resultEntry lineNumber="1" cliString="no cns config partial mixy">
        <success change="NO_CHANGE" mode="IMMEDIATE" />
      </resultEntry>
      <resultEntry lineNumber="2" cliString="no stupid">
        <failure errorType="TEMPORARY" errorCode="PARSE_ERROR_NOMATCH" />
      </resultEntry>
      <resultEntry lineNumber="3" cliString="no cns exec 80 ">
        <success change="NO_CHANGE" mode="IMMEDIATE" />
      </resultEntry>
    </response>
  </SOAP:Body>
</SOAP:Envelope>
WSMA Config Request: Enhanced Device Interface (EDI) Data Model

<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP:Body>
    <request xmlns="urn:cisco:wsma-config" correlator="6.1">
      <configApply details="all">
        <config-data>
          <xml-config-data>
            <Device-Configuration><cns operation="delete" > <config><partial>
              <HostNameAddressConfigurationServer>mixy</HostNameAddressConfigurationServer>
              <PortNumberConfigServiceDefault80>80</PortNumberConfigServiceDefault80></partial></config></cns>
            <stupid operation="delete" /><cns operation="delete" ><exec><P>80</P></exec></cns>
          </Device-Configuration>
        </xml-config-data>
      </config-data>
    </request>
  </SOAP:Body>
</SOAP:Envelope>

WSMA Config Response: EDI Data Model

<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP:Body>
    <response xmlns="urn:cisco:wsma-config" correlator="6.1" success="1">
      <resultEntry lineNumber="1" cliString="no cns config partial mixy 80">
        <success change="NO_CHANGE" mode="IMMEDIATE" />
      </resultEntry>
      <resultEntry lineNumber="2" cliString="no stupid">
        <failure errorType="TEMPORARY" errorCode="PARSE_ERROR_NOMATCH" />
      </resultEntry>
      <resultEntry lineNumber="3" cliString="no cns exec 80">
        <success change="NO_CHANGE" mode="IMMEDIATE" />
      </resultEntry>
    </response>
  </SOAP:Body>
</SOAP:Envelope>

WSMA File List Request

<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP:Body>
    <request xmlns="urn:cisco:wsma-filesystem" correlator="2"><fileList/></request>
  </SOAP:Body>
</SOAP:Envelope>

WSMA File List Response

<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
WSMA File Copy Request

<SOAP:Body>
<fileCopy erase="0" overwrite="1" filesize="131739">
<srcURL>tftp://oz-dirt/jbalestr/spec.odm</srcURL>
<dstURL>test</dstURL>
</fileCopy>
</SOAP:Body>
</SOAP:Envelope>

WSMA File Copy Response

<SOAP:Body>
</SOAP:Envelope>
Configuration Examples for WSMA

Example: Enabling SSHv2 Using a Hostname and Domain Name

```
configure terminal
hostname host1
ip domain-name example.com
crypto key generate rsa
ip ssh timeout 120
ip ssh version 2
```

Example: Enabling SSHv2 Using RSA Keys

```
configure terminal
ip ssh rsa keypair-name sshkeys
```
crypto key generate rsa usage-keys label sshkeys modulus 768
ip ssh timeout 120
ip ssh version 2

Example: Configuring a WSMA Service

configure terminal
wsma agent config profile prof

Example: Configuring the WSMA Initiator Profile

configure terminal
wsma profile initiator ssh-test
transport ssh sshserver path /mypath/bin/mywsma-app.sh user user1 6 encrypted-password

Example: Configuring the WSMA Listener Profile

configure terminal
wsma profile listener mySession
encap soap12
transport ssh subsys wsma
acl 34
exit

Example: Displaying WSMA Profile Parameters

Device# show wsma profile connections
Listener Profile http: 0 open connections: 0 closing connections
Encap: soap11
WSSE header is required
Max message (RX) is 50 Kbytes
SOAP Faults are sent
Idle timeout infinite
Keepalive not configured
Listening via http
Listening to path /wsma. Max Idle 0 ms. Accepting post on plaintext connections.
Established at 01:11:04.207 UTC Tue Jan 12 2010
Tx 493475 bytes (90 msg), Tx 0 errors,
Last message sent at 05:18:08.539 UTC Sat Feb 20 2010
Rx 59457 bytes (90 msg), 0 empty msg
Last message received at 05:18:08.295 UTC Sat Feb 20 2010
Listener Profile ssh: 2 open connections: 0 closing connections
Encap: soap11
WSSE header is required
Max message (RX) is 50 Kbytes
SOAP Faults are sent
Idle timeout infinite
Keepalive not configured
Listening via ssh
SSH listener, 10 sessions accepted, 0 sessions rejected
Connected sessions...
Remote connection via SSH by user(cisco) from 172.16.29.134:44457, state connect
Established at 01:14:03.184 UTC Thu Mar 11 2010
Tx 1183 bytes (2 msg), Tx 0 errors,
Last message sent at 01:14:48.565 UTC Thu Mar 11 2010
Rx 10 bytes (1 msg), 0 empty msg
Last message received at 01:14:48.565 UTC Thu Mar 11 2010
Remote connection via SSH by user(cisco) from 172.16.154.90:45404, state connect
Established at 01:14:28.041 UTC Thu Mar 11 2010
Tx 1183 bytes (2 msg), Tx 0 errors,
Last message sent at 01:14:54.437 UTC Thu Mar 11 2010
Rx 7 bytes (1 msg), 1 empty msg
Last message received at 01:14:54.437 UTC Thu Mar 11 2010
Initiator Profile ssh-init: 0 open connections: 0 closing connections
Encap: soap11
WSSE header is required
Max message (RX) is 50 Kbytes
SOAP Faults are sent
Idle timeout infinite
Keepalive not configured
Reconnect time 60 seconds
No transport configured

The following example shows how to display information about Web Services Management Agent (WSMA)
profile counters:

Device# show wsma profile counters

Statistics for profile http
incoming total 90, bad XML 0, authentication errors 0, oversized 0
outgoing total 90, absorbed 0
message internal errors 0
Connection Accepts 90, local hangup 0, remote hangup 90, keepalive hangup 0
session internal errors 0
Statistics for profile ssh
incoming total 9, bad XML 2, authentication errors 0, oversized 0
outgoing total 20, absorbed 0
message internal errors 0
Connection Accepts 8, local hangup 0, remote hangup 8, keepalive hangup 0
session internal errors 0

The following example shows how to display information about WSMA profile schema:

Device# show wsma profile schema

Schema http
New Name Space ''
<VirtualRootTag> [0, 1] required
  New Name Space 'http://schemas.xmlsoap.org/soap/envelope/'
  <Envelope> 1+ required
    <Header> any subtree is allowed
    <Body> 1 required
      <Fault> [0, 1] required
        <faultcode> 1 required
        <faultstring> 1 required
        <faultactor> [0, 1] required
        <detail> any subtree is allowed
      New Name Space 'urn:cisco:exec'
        <request> [0, 1] required
          <execCLI> 1+ required
            <cmd> 1 required
            <dialogue> 0+ required
            <expect> 1 required
            <reply> 1 required
          New Name Space 'urn:cisco:wsma-config'
            <request> [0, 1] required
              <config-data> 1 required
                <cli-config-data> [0, 1] required
                  <cmd> 1 required
                  <cli-config-data-block> [0, 1] required
                <xml-config-data> [0, 1] required
                <Device-Configuration> [0, 1] required
                  <> any subtree is allowed
                New Name Space 'urn:cisco:wsma-filesystem'
                  <request> [0, 1] required
                    <fileList> [0, 1] required
                    <fileDelete> [0, 1] required
                    <deleteFileList> 1 required
                    <filename> 1+ required

Additional References

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>WSMA commands</td>
<td>Cisco IOS Web Services Management Agent Command Reference</td>
</tr>
<tr>
<td>Public Key Infrastructure</td>
<td>Public Key Infrastructure Configuration Guide in the Secure Connectivity Configuration Guide Library</td>
</tr>
<tr>
<td>Security and IP access lists commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
</tbody>
</table>

RFCs

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2132</td>
<td>DHCP Options and BOOTP Vendor Extensions</td>
</tr>
<tr>
<td>RFC 2246</td>
<td>The TLS Protocol Version 1.0</td>
</tr>
</tbody>
</table>
### Title

**RFC 4251**

The Secure Shell (SSH) Protocol Architecture

**RFC 4252**

The Secure Shell (SSH) Authentication Protocol

### Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

### Feature Information for Web Services Management Agent

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.
Table 1: Feature Information for Web Services Management Agent

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Services Management Agent</td>
<td>12.2(50)SY</td>
<td>The WSMA feature enables you to perform network configurations via the Cisco CLI over an encrypted transport.</td>
</tr>
<tr>
<td></td>
<td>12.4(24)T</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15.1(1)SG</td>
<td>The WSMA protocol defines a set of web services through which a network device can be managed, configuration data information can be retrieved, and new configuration data can be uploaded and manipulated. WSMA uses an XML-based data encoding for configuration data and protocol messages.</td>
</tr>
<tr>
<td></td>
<td>15.1(1)SY</td>
<td>In Cisco IOS 15.1(1)T this feature was modified to include support for both listener and initiator profiles.</td>
</tr>
<tr>
<td></td>
<td>15.1(1)T</td>
<td>The following commands were introduced: acl, clear wsma agent, clear wsma profile, debug wsma agent, debug wsma profile, encap, idle-timeout, max-message, show wsma agent, show wsma id, show wsma profile, stealth, transport, wsma agent, wsma id, wsma profile initiator, wsma profile listener.</td>
</tr>
<tr>
<td>WSMA Enhancements for WebUI</td>
<td>15.1(1)SY</td>
<td>The WSMA Enhancements for WebUI feature introduces support for the execTest tag in the EXEC WSMA service. The execTest tag validates the syntax of an EXEC command but does not run the command.</td>
</tr>
</tbody>
</table>
The WSMA and XML-PI Enhancement feature enables multiple profiles to be associated with any WSMA agent. Prior to the introduction of this feature, only one profile can be associated with a WSMA agent.

The following command was modified by this feature: `show wsma agent`.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>WSMA and XML-PI Enhancement</td>
<td>15.1(1)SY</td>
<td>The WSMA and XML-PI Enhancement feature enables multiple profiles to be associated with any WSMA agent. Prior to the introduction of this feature, only one profile can be associated with a WSMA agent.</td>
</tr>
<tr>
<td></td>
<td>15.2(1)S</td>
<td>The following command was modified by this feature: <code>show wsma agent</code>.</td>
</tr>
<tr>
<td></td>
<td>15.2(2)T</td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 2

WSMA Enhancements for Wireless Management

Web Services Management Agent (WSMA) Enhancements for Wireless Management feature allows you to reduce the reconnecting rates in an outage when device connections to the management server on wireless networks are disconnected unexpectedly. This feature also allows you to implement payload compression to optimize the bandwidth usage.

• Finding Feature Information, page 43
• Restrictions for WSMA Enhancements for Wireless Management, page 43
• Information About WSMA Enhancements for Wireless Management, page 44
• How to Configure Smart Back-off Reconnect, page 44
• Configuration Examples for WSMA Enhancements for Wireless Management, page 45
• Additional References for WSMA Enhancements, page 46
• Feature Information for WSMA Enhancements for Wireless Management, page 46

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for WSMA Enhancements for Wireless Management

• Only Web Services Management Agent (WSMA) EXEC response provides compression on demand.
• WSMA must provide support to all transport protocols including Secure Shell (SSH), Transport Layer Security (TLS), HTTP, HTTPS, and other TCP protocols.
• WSMA does not use the option to support gzip compression for HTTP.

Information About WSMA Enhancements for Wireless Management

Smart Back-off Reconnect Support

In a network outage, thousands of devices on the same wireless network (for example, Smartgrids) try to reconnect to the Network Management Server (NMS) at the same time. However, only limited number of devices can reconnect successfully on the first reconnect and subsequent retries. This results in high billings of some devices that fail to reconnect.

Web Services Management Agent (WSMA) enhancements for wireless management allows you to reduce the reconnecting rates with smart back-off reconnect policy by enhancing the existing `reconnect` command.

EXEC Agent–Response Payload Compression Support

The WSMA EXEC Agent enhancement allows you to optimize the payloads on wireless networks whenever possible to optimize bandwidth usage.

A new optional compression XML attribute in the WSMA EXEC request and WSMA EXEC response supports the new response payload compression feature. By default, there is no payload compression if the new compression XML attribute is not specified in the WSMA EXEC request.

How to Configure Smart Back-off Reconnect

Configuring Smart Back-off Reconnect

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `wsma profile initiator profile-name`
4. `reconnect [pause-time [exponential-backoff-factor [random]]]`
5. `end`

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
</tbody>
</table>
### Purpose Command or Action

**Example:**

```
Device> enable
```

- Enter your password if prompted.

### Step 2 configure terminal

**Example:**

```
Device# configure terminal
```

Enters global configuration mode.

### Step 3 wsma profile initiator profile-name

**Example:**

```
Device(config)# wsma profile initiator prof1
```

Creates a service initiator and enters WSMA initiator configuration mode.

### Step 4 reconnect [pause-time  [exponential-backoff-factor [random]]]

**Example:**

```
Device(config-wsma-init)# reconnect 130 3 random
```

(Optional) Specifies the time for the WSMA initiator profile to wait before attempting to reconnect a session.

### Step 5 end

**Example:**

```
Device(config-wsma-init)# end
```

Ends the current configuration session and returns you to privileged EXEC mode.

---

## Configuration Examples for WSMA Enhancements for Wireless Management

### Example: Configuring the Smart Back-off Reconnect

The following example shows how to configure a smart back-off reconnect policy:

```
Device> enable
Device# configure terminal
Device(config)# wsma profile initiator smartgrid
Device(config-wsma-init)# reconnect 130 4 random
Device(config-wsma-init)# end
```
Additional References for WSMA Enhancements

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
<tr>
<td>WSMA commands</td>
<td>Cisco IOS Web Services Management Agent Command Reference</td>
</tr>
</tbody>
</table>

Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

Feature Information for WSMA Enhancements for Wireless Management

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.
Table 2: Feature Information for Web Services Management Agent

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>WSMA Enhancements for Wireless Management</td>
<td>15.4(1)T</td>
<td>Web Services Management Agent (WSMA) Enhancements for Wireless Management feature allows you to reduce the reconnecting rates in an outage when device connections to the management server on wireless networks are disconnected unexpectedly. This feature also allows you to implement payload compression to optimize the bandwidth usage.</td>
</tr>
<tr>
<td></td>
<td>Cisco IOS 15.2(2)E</td>
<td></td>
</tr>
</tbody>
</table>

In Cisco IOS 15.2(2)E, this feature is supported on the following platforms:

- Catalyst 4500-X Series Switches
- Catalyst 3750
- Catalyst 3850
- Catalyst 2960
- Catalyst XE-4500 Series Switches
- Catalyst 5700 Wireless Controllers
- Catalyst 3650 Series Switches

The following command was modified: `reconnect`. 