Web Services Management Agent with TLS

Last Updated: July 23, 2012

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 Transport Layer Security (TLS) 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 TLS to initiate secure connections from Cisco software to applications over trusted and untrusted networks.

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 WSMA over TLS

- WSMA over TLS requires a certificate authority (CA) server to be available on the network.

Restrictions for WSMA over TLS

- You must be running a crypto image on your device in order to configure Transport Layer Security (TLS).

Information About WSMA with TLS

WSMA over TLS

The Web Services Management Agent (WSMA) agent needs to be configured to use a service profile which is using Transport Layer Security (TLS) as a transport to run the WSMA over TLS feature. The TLS protocol uses endpoint authentication and encryption to provide secure connections over any network. Encryption protects against eavesdropping, and digital certificates (signed by a trusted CA) protect against tampering and message forgery by authenticating the endpoints.

The WSMA listener and initiator profiles use the TLS server and client adapters to create and accept TLS connections. The TLS server uses a default port (13000) to listen for incoming connections; similarly, the TLS client uses the same default port to initiate connections. You can change the default port setting by changing the profile configuration.

Trusted Certificates

The WSMA over TLS feature requires a CA server to be available on the network. The CA's public key is made known to the client, and the public key must correspond to the private key used to sign the server's certificate. The Cisco device and the remote WSMA application use the CA server to validate the certificates sent between them.

WSMA Profiles with TLS

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 communication 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 SOAP 1.1 and SOAP 1.2.
- The transportation mechanisms for WSMA are Secure Shell (SSH), HTTP, Secure HTTP (HTTPS), and TLS. This mechanism opens listening sockets for listeners on the device or connecting sockets for clients on the device.

Service Listener with TLS

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. The accepted addresses are configured by defining an access list.

Accepted user IDs are configured by defining the transport method that the service listener listens for. The Transport Layer Security (TLS) transport method enforces the specific user ID that is accepted.

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

WSMA over TLS Authentication and Authorization

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 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.

Unlike Secure Shell (SSH) or Secure HTTP (HTTPS) connections, TLS connections do not require that a user log in to a Cisco device. TLS certificates provide host-level authentication but do not always provide user-level authentication. Therefore, the Web Services Security Header (WSSE) header (if configured) is used to authenticate and authorize different users from a specified host.

For TLS listener profiles, all WSMA requests are authenticated using the Simple Object Access Protocol (SOAP) WSSE header. After the request is authenticated, the user is authorized to perform operations based on the configured privilege level. The user can be configured on the Cisco device or an the AAA server. The identity of the remote host is validated using the TLS client-side certificate.

For TLS initiator profiles, the identity of the remote endpoint is verified using the certificate authority (CA) server as part of the TLS connection setup. After a connection is established, all incoming WSMA requests are authenticated using the WSSE header. After the request is authenticated, the user is authorized to perform operations based on the configured privilege level. The user can be configured on the Cisco device or on the AAA server.

If the WSSE SOAP header is disabled for a TLS listener or initiator profile, user-level authentication is not possible, and the following process is used to decide the authorization level to assign to the profile:

- The authorization level set using the **no wsse authorization level** command is used for all agents associated with the profile.
- If no authorization level is set, the default privilege level is used. The default privilege level is set to 1 (the minimum level).

How to Configure WSMA with TLS

Configuring Certificate Validation on the TLS Client for WSMA Initiator Mode

To use the Transport Layer Security (TLS) protocol to connect to the remote host, the Cisco device (acting as the TLS client) must validate the signed certificate of the Web Services Management Agent (WSMA) application host (acting as the TLS server). To allow the device to validate the certificate and trust all certificates signed by the certificate authority (CA), you must configure a trustpoint for the CA on the device and instruct the device to download a self-signed certificate from the CA that authenticates the CA to the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint** *name*
4. **enrollment url** *url*
5. **exit**
6. **crypto pki authenticate** *name*
7. **end**
8. **show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**crypto pki trustpoint** *name* **Example:** `Device(config)# crypto pki trustpoint my_CA`	Declares the CA that the device should use and enters ca-trustpoint configuration mode.

	Command or Action	Purpose
Step 4	**enrollment url** *url* **Example:** `Device(ca-trustpoint)# enrollment url http://myCAurl:80`	Specifies the URL of the CA.
Step 5	**exit** **Example:** `Device(ca-trustpoint)# exit`	Exits ca-trustpoint configuration mode and returns to global configuration mode.

Step 6	**crypto pki authenticate** *name*
	
	Example:
	`Device(config)# crypto pki authenticate my_CA`
	`Certificate has the following attributes:`
	`Fingerprint MD5: AC3B4A2B FD027F65 0B4650BF 018B1F79`
	`Fingerprint SHA1: BC183062 A013FFDC 1E8E79B3 0150DEBF B887CD15`
	`% Do you accept this certificate? [yes/no]: yes`
	`Trustpoint CA certificate accepted.`
		Authenticates the CA to the device by obtaining the self-signed certificate of the CA that contains the public key of the CA.
		• Because the CA signs its own certificate, you should manually authenticate the public key of the CA by contacting the CA administrator when you perform this command.
		• After the device obtains the certificate, it displays a prompt asking you to accept the certificate.

Step 7	**end** **Example:** `Device(config)# end`	Ends the current configuration session and returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** `Device# show running-config`	Displays the status of the server configuration, including CA and certificate details.

Enabling a WSMA Service Initiator over TLS

If you configure service initiator over Transport Layer Security (TLS), you must first configure the certificate authority (CA) settings on the Cisco device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **wsma profile initiator** *profile-name*
4. **encap** {**soap11** | **soap12**}
5. [**backup**] **transport tls** *remote-host* [*initiator-port-number*] [**localcert** *trustpoint-name*] [**remotecert** *trustpoint-name*] [**source** *source-interface*]}
6. **keepalive** *interval* [**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

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • 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	**encap** {**soap11**	**soap12**} **Example:** `Device(config-wsma-initiator)# encap soap12`	(Optional) Configures an encapsulation for the service listener profile.

	Command or Action	Purpose
Step 5	[**backup**] **transport tls** *remote-host* [*initiator-port-number*] [**localcert** *trustpoint-name*] [**remotecert** *trustpoint-name*] [**source** *source-interface*]} **Example:** `Device(config-wsma-initiator)# transport tls 192.2.1.10`	Defines a transport configuration for the WSMA profile. • The port that the remote WSMA TLS application is listening on must be known. By default this is port 13000. If the server is listening on a port other than 13000, then the correct port must be configured using the *initiator-port-number* argument.
Step 6	**keepalive** *interval* [**retries** *number*] **Example:** `Device(config-wsma-initiator)# keepalive 100 retries 10`	(Optional) Enables keepalive messages and configures interval and retry values for a WSMA profile.
Step 7	**idle-timeout** *minutes* **Example:** `Device(config-wsma-initiator)# idle-timeout 345`	(Optional) Specifies the amount of time (in minutes) to keep the session alive in the absence of any data traffic.
Step 8	**max-message** *message-size* **Example:** `Device(config-wsma-initiator)# max-message 290`	(Optional) Specifies the maximum receive message size (from 1 to 2000 kilobytes).
Step 9	**backup hold** *minutes* **Example:** `Device(config-wsma-initiator)# backup hold 233`	(Optional) Sets the time (in minutes) that the WSMA profile remains connected to the backup transport configuration.
Step 10	**backup excluded** *seconds* **Example:** `Device(config-wsma-initiator)# backup excluded 30`	(Optional) Sets the time that the WSMA profile must wait before attempting to connect to the backup transport configuration after a connection is lost.

	Command or Action	Purpose
Step 11	**reconnect** *seconds* **Example:** `Device(config-wsma-initiator)# reconnect 434`	(Optional) Specifies the time for the WSMA initiator profile to wait before attempting to reconnect a session.
Step 12	**stealth** **Example:** `Device(config-wsma-initiator)# stealth`	(Optional) Configures the service to not send Simple Object Access Protocol (SOAP) fault messages in response to corrupted XML messages.
Step 13	**wsse** **Example:** `Device(config-wsma-initiator)# wsse`	(Optional) Enables the Web Services Security Header (WSSE) for a WSMA profile. • By default, the WSSE is enabled. Enter the **no wsse** command to disable the WSSE.
Step 14	**end** **Example:** `Device(config-wsma-initiator)# end`	Ends the current configuration session and returns to privileged EXEC mode.

Configuring Certificates on the TLS Server for WSMA Listener Mode

To configure CA certificates for WSMA listener mode using the TLS protocol on the Cisco IOS device, you must configure a trustpoint for the CA on the device and instruct the device to download a self-signed certificate from the CA which authenticates the CA to the device. You must then instruct the device to request it’s own certificate signed by the CA.

To enable certificates for WSMA listener mode, perform the following tasks:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint** *name*
4. **enrollment** {**url***url* | **terminal**}
5. **exit**
6. **crypto pki authenticate** *name*
7. **crypto pki enroll** *name*
8. **crypto pki import** *name* **certificate**
9. **end**
10. **show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**crypto pki trustpoint** *name* **Example:** `Device(config)# crypto pki trustpoint my_CA`	Declares the CA that the device should use and enter ca-trustpoint configuration mode.
Step 4	**enrollment** {**url***url*	**terminal**} **Example:** `Device(ca-trustpoint)# enrollment url http://myCAurl:80`	Specifies the URL of the CA. • Use the **enrollment terminal** command to specify manual cut-and-paste certificate enrollment.
Step 5	**exit** **Example:** `Device(ca-trustpoint)# exit`	Exits ca-trustpoint configuration mode and returns to global configuration mode.

	Command or Action	Purpose

Step 6	**crypto pki authenticate** *name*
	
	Example:
	`Device(config)# crypto pki authenticate my_CA`
	
	`Certificate has the following attributes:`
	`Fingerprint MD5: AC3B4A2B FD027F65 0B4650BF 018B1F79`
	`Fingerprint SHA1: BC183062 A013FFDC 1E8E79B3 0150DEBF B887CD15`
	`% Do you accept this certificate? [yes/no]: yes`
	`Trustpoint CA certificate accepted.`
		Authenticates the CA to the device by obtaining the self-signed certificate of the CA that contains the public key of the CA.
		• Because the CA signs its own certificate, you should manually authenticate the public key of the CA by contacting the CA administrator when you perform this command.
		• If you specified manual cut-and-paste certificate enrollment in step 4, you will now be prompted to enter the encoded CA certificate.
		• After the device obtains the certificate, it displays a prompt asking you to accept the certificate.

Step 7	**crypto pki enroll** *name*
	
	Example:
	`Device(config)# crypto pki enroll my_CA`
	
	`% Start certificate enrollment ..`
	`% Create a challenge password. You will need to verbally provide this password to the CA Administrator in order to revoke your certificate. For security reasons your password will not be saved in the configuration. Please make a note of it.`
	`Password:`
	`Re-enter password:`
	`% The subject name in the certificate will include: devicename.cisco.com`
	`% Include the router serial number in the subject name? [yes/no]: yes`
	`% The serial number in the certificate will be: 34835646`
	`% Include an IP address in the subject name? [no]:`
	`Request certificate from CA? [yes/no]: yes`
	`% Certificate request sent to Certificate Authority`
	`% The 'show crypto pki certificate verbose my_CA' command will show the fingerprint.`
		Enrolls the device with the CA and requests certificates for this device from the CA.
		• The device prompts you to enter a challenge password and to select configuration options during the enrollment process.

Step 8	**crypto pki import** *name* **certificate** **Example:** `Device(config)# crypto pki import my_CA certificate`	(Optional) Manually imports a certificate to the device. • This command is required only if you selected manual cut-and-paste in step 4. • The device displays a certificate request on the console terminal. The certificate request must be copied to the CA. • The CA creates a signed certificate for the device. • The signed certificate is imported into the device using this command.

	Command or Action	Purpose
Step 9	**end** **Example:** `Device(config)# end`	Ends the current configuration session and returns to privileged EXEC mode.
Step 10	**show running-config** **Example:** `Device# show running-config`	Displays the status of the server configuration, including CA and certificate details.

Enabling a WSMA Service Listener over TLS

If you configure service listener over Transport Layer Security (TLS), you must first configure the certificate authority (CA) settings on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **wsma profile listener** *profile-name*
4. **encap** {**soap11** | **soap12**}
5. **transport tls** [*listener-port-number*] [**localcert** *trustpoint-name*] [**disable-remotecert-validation** | **remotecert** *trustpoint-name*]
6. **idle-timeout** *minutes*
7. **max-message** *message-size*
8. **keepalive** *interval* [**retries** *number*]
9. **acl** *acl-number*
10. **stealth**
11. **wsse**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**wsma profile listener** *profile-name* **Example:** `Device(config)# wsma profile listener prof1`	Creates a service listener and enters the Web Services Management Agent (WSMA) listener configuration mode.
Step 4	**encap** {**soap11** | **soap12**} **Example:** `Device(config-wsma-listen)# encap soap12`	(Optional) Configures an encapsulation for the service listener profile.
Step 5	**transport tls** [*listener-port-number*] [**localcert** *trustpoint-name*] [**disable-remotecert-validation** | **remotecert** *trustpoint-name*] **Example:** `Device(config-wsma-listen)# transport tls 65534`	Defines a transport configuration for the WSMA profile.
Step 6	**idle-timeout** *minutes* **Example:** `Device(config-wsma-listen)# idle-timeout 345`	(Optional) Specifies the amount of time (in minutes) to keep the session alive in the absence of any data traffic.
Step 7	**max-message** *message-size* **Example:** `Device(config-wsma-listen)# max-message 290`	(Optional) Specifies the maximum receive message size (from 1 to 2000 kilobytes).
Step 8	**keepalive** *interval* [**retries** *number*] **Example:** `Device(config-wsma-listen)# keepalive 100 retries 10`	(Optional) Enables keepalive messages and configures interval and retry values for a WSMA profile. • Keepalive messages are not sent on HTTP or Secure HTTP (HTTPS) listener connections.

	Command or Action	Purpose
Step 9	**acl** *acl-number* **Example:** `Device(config-wsma-listen)# acl 34`	(Optional) Defines the access control list (ACL) group to use.
Step 10	**stealth** **Example:** `Device(config-wsma-listen)# stealth`	(Optional) Configures the service to not send Simple Object Access Protocol (SOAP) fault messages in response to corrupted XML messages.
Step 11	**wsse** **Example:** `Device(config-wsma-listen)# wsse`	(Optional) Enables the Web Services Security Header (WSSE) for a WSMA profile. • By default, the WSSE is enabled. Enter the **no wsse** command to disable the WSSE.
Step 12	**end** **Example:** `Device(config-wsma-listen)# end`	Ends the current configuration session and returns to privileged EXEC mode.

Configuration Examples for WSMA with TLS

Example: Configuring Certificates on the TLS Server for WSMA Listener Mode

```
configure terminal
 crypto pki trustpoint my_CA
  enrollment terminal
  exit
crypto pki authenticate my_CA
 .
 .
 .
crypto pki import my_CA certificate
 .
 .
 .
```

```
end
```

Example: Enabling a WSMA Service Initiator over TLS

```
configure terminal
 wsma profile initiator profile1
  encap soap12
  keepalive 100 retries 10
  idle-timeout 120
  max-message 290
  backup hold 233
  backup excluded 30
  reconnect 434
  stealth
  wsse
```

Example: Enabling Certificate Validation on the TLS Client for WSMA Initiator Mode

```
configure terminal
 crypto pki trustpoint my_CA
  enrollment url http://myCAurl:80
  exit
crypto pki authenticate my_CA
```

Example: Enabling a WSMA Service Listener over TLS

```
configure terminal
 wsma profile listener profile1
  encap soap12
  transport tls 65534
  idle-timeout 345
  max-message 290
  keepalive 100 retries 10
  stealth
  wsse
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
WSMA commands	*Cisco IOS Web Services Management Agent Command Reference*

Related Topic	Document Title
IP access lists	*Security Configuration Guide: Access Control Lists* in the *Securing the Data Plan Configuration Guide Library*
IP access lists commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS Security Command Reference*
Public Key Infrastructure	*Public Key Infrastructure Configuration Guide* in the *Secure Connectivity Configuration Guide Library*
Secure Shell and Secure Shell Version 2	*Secure Shell Configuration Guide* in the *Securing User Services Configuration Guide Library*
Security commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS Security Command Reference*
WSMA schema files in XSD format	ftp://ftp.cisco.com/pub/wsma/schema/

RFCs

RFC	Title
RFC 2132	*DHCP Options and BOOTP Vendor Extensions*
RFC 2246	*The TLS Protocol Version 1.0*
RFC 4251	*The Secure Shell (SSH) Protocol Architecture*
RFC 4252	*The Secure Shell (SSH) Authentication Protocol*

Technical Assistance

Description	Link
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.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Web Services Management Agent with TLS

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 1 Feature Information for Web Services Management Agent with TLS

Feature Name	Releases	Feature Information
Web Services Management Agent with TLS	12.2(50)SY 15.1(1)T	This feature enables support for the TLS encryption protocol for WSMA initiator and listener profiles. The following commands were introduced or modified by this feature: **backup excluded**, **backup hold**, **debug wsma profile**, **encap**, **idle-timeout**, **keepalive**, **max-message**, **reconnect**, **stealth**, **transport**, **wsma profile initiator**, **wsma profile listener**, **wsse**.