



Implementing IPv6 for Network Management

Last Updated: August 1, 2012

This document describes the concepts and commands used to manage Cisco applications over IPv6 and to implement IPv6 for network management.

- [Finding Feature Information, page 1](#)
- [Information About Implementing IPv6 for Network Management, page 1](#)
- [How to Implement IPv6 for Network Management, page 5](#)
- [Configuration Examples for Implementing IPv6 for Network Management, page 13](#)
- [Additional References, page 15](#)
- [Feature Information for Implementing IPv6 for Network Management, page 17](#)

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.

Information About Implementing IPv6 for Network Management

- [Telnet Access over IPv6, page 2](#)
- [TFTP IPv6 Support, page 2](#)
- [ping and traceroute Commands in IPv6, page 2](#)
- [SSH over an IPv6 Transport, page 2](#)
- [SNMP over an IPv6 Transport, page 2](#)
- [Cisco IOS XE IPv6 Embedded Management Components, page 3](#)



Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Telnet Access over IPv6

The Telnet client and server in Cisco software support IPv6 connections. A user can establish a Telnet session directly to the device using an IPv6 Telnet client, or an IPv6 Telnet connection can be initiated from the device. A vty interface and password must be created in order to enable Telnet access to an IPv6 device.

TFTP IPv6 Support

TFTP is designed to transfer files over the network from one host to another using the most minimal set of functionality possible. TFTP uses a client/server model in which clients can request to copy files to or from a server. TFTP uses UDP over IPv4 or IPv6 as its transport, and it can work over IPv4 and IPv6 network layers.

- [TFTP File Downloading for IPv6, page 2](#)

TFTP File Downloading for IPv6

IPv6 supports TFTP file downloading and uploading using the **copy** command. The **copy** command accepts a destination IPv6 address or IPv6 hostname as an argument and saves the running configuration of the device to an IPv6 TFTP server, as follows:

```
Device# copy running-config tftp://[3ffe:xxxx:c18:1:290:27ff:fe3a:9e9a]/running-config
```

ping and traceroute Commands in IPv6

The **ping** command accepts a destination IPv6 address or IPv6 hostname as an argument and sends Internet Control Message Protocol version 6 (ICMPv6) echo request messages to the specified destination. The ICMPv6 echo reply messages are reported on the console. Extended ping functionality is also supported in IPv6.

The **traceroute** command accepts a destination IPv6 address or IPv6 hostname as an argument and will generate IPv6 traffic to report each IPv6 hop used to reach the destination address.

SSH over an IPv6 Transport

Secure shell (SSH) SSH in IPv6 functions the same and offers the same benefits as SSH in IPv4. The SSH server feature enables an SSH client to make a secure, encrypted connection to a Cisco device, and the SSH client feature enables a Cisco device to make a secure, encrypted connection to another Cisco device or to any other device running an SSH server. IPv6 enhancements to SSH consist of support for IPv6 addresses that enable a Cisco device to accept and establish secure, encrypted connections with remote IPv6 nodes over an IPv6 transport.

SNMP over an IPv6 Transport

Simple Network Management Protocol (SNMP) can be configured over IPv6 transport so that an IPv6 host can perform SNMP queries and receive SNMP notifications from a device running IPv6 software. The SNMP agent and related MIBs have been enhanced to support IPv6 addressing. This feature uses the data encryption standard (3DES) and advanced encryption standard (AES) message encryption.

- [Cisco IOS XE IPv6 MIBs, page 3](#)

- [MIBs Supported for IPv6, page 3](#)

Cisco IOS XE IPv6 MIBs

Cisco has long supported IP-MIB and IP-FORWARD-MIB in IPv4. CISCO-IETF-IP-MIB and CISCO-IETF-IP-FORWARDING-MIB are IPv6 MIBs that are defined as being protocol-independent, but are implemented only for IPv6 objects and tables. IP-MIB and IP-FORWARD-MIB were updated to RFC 4293 and RFC 4292 standards, as follows:

- The upgrade is backward-compatible; all IP-MIB and IP-FORWARD-MIB objects and tables still appear.
- IP-MIB and IP-FORWARD-MIB include new IPv6-only, IPv4-only, and protocol-version independent (PVI) objects and tables. However, IPv6 supports IPv6-only and the new IPv6 part of the PVI objects and tables in these MIBs.

MIBs Supported for IPv6

The following MIBs are supported for IPv6:

- CISCO-CONFIG-COPY-MIB
- CISCO-CONFIG-MAN-MIB
- CISCO-DATA-COLLECTION-MIB
- CISCO-FLASH-MIB
- CISCO-SNMP-TARGET-EXT-MIB
- ENTITY-MIB
- IP-FORWARD-MIB
- IP-MIB
- NOTIFICATION-LOG-MIB
- SNMP-TARGET-MIB

CISCO-CONFIG-COPY-MIB and CISCO-FLASH-MIB support IPv6 addressing when TFTP, remote copy protocol (rcp), or FTP is used.

Cisco IOS XE IPv6 Embedded Management Components

This section describes Cisco IOS XE software embedded management components that have IPv6-compliant operability in IPv6 and dual-stack IPv6 and IPv4 networks.

- [Syslog, page 3](#)
- [TCL, page 4](#)
- [CNS Agents, page 4](#)
- [Config Logger, page 5](#)
- [IP SLAs for IPv6, page 5](#)

Syslog

The Cisco system message logging (syslog) process in IPv6 allows users to log syslog messages to external syslog servers and hosts with IPv6 addresses. This implementation allows user to specify an IPv4-based logging host (syslog server) by providing the host's IP address in IPv4 format (for example, 192.168.0.0) or IPv6 format (for example, 2001:DB8:A00:1::1/64).

TCL

Tool command language (TCL) is used in Cisco software for IPv6 to support features such as embedded syslog manager (ESM), embedded event manager (EEM), interactive voice response (IVR), and tcsh parser mode. TCL supports both initiating (client) and listening (server) sockets.

CNS Agents

IPv6 addressing is supported in the Cisco Networking Services (CNS) subsystem. CNS is a foundation technology for linking users to networking services, and it provides the infrastructure for the automated configuration of large numbers of network devices. Many IPv6 networks are complex, with many devices, and each device must be configured individually. When standard configurations do not exist or have been modified, the time involved in initial installation and subsequent upgrading is considerable. ISPs need a method for sending out partial configurations to introduce new services.

To address all these issues, CNS was designed to provide "plug-and-play" network services using a central directory service and distributed agents. CNS features include CNS agents and a flow-through provisioning structure. CNS flow-through provisioning uses the CNS configuration and event agents to provide an automated workflow, eliminating the need for an onsite technician.

IPv6 addressing supports the CNS agents described in the following sections:

- [CNS Configuration Agent, page 4](#)
- [CNS Event Agent, page 4](#)
- [CNS EXEC Agent, page 4](#)
- [CNS Image Agent, page 4](#)

CNS Configuration Agent

The CNS configuration agent is involved in the initial configuration and subsequent partial configurations on a Cisco device. The configuration agent uses a CNS configuration engine to provide methods for automating initial Cisco device configurations, incremental configurations, and synchronized configuration updates, and the configuration engine reports the status of the configuration load as an event to which a network monitoring or workflow application can subscribe.

CNS Event Agent

The CNS event agent provides a transport connection to the CNS event bus for all other CNS agents. No event can be sent to the device by the configuration engine until the CNS event agent is operational and has successfully built a connection between the configuration engine and the device.

The event agent uses a CNS configuration engine to provide methods for automating initial Cisco device configurations, incremental configurations, and synchronized configuration updates.

CNS EXEC Agent

The CNS EXEC agent allows a remote application to execute a CLI command in EXEC mode on a Cisco device by sending an event message that contains the command.

CNS Image Agent

Administrators maintaining large networks of Cisco devices need an automated mechanism to load image files onto large numbers of remote devices. Network management applications are useful to determine

which images to run and how to manage images received from the Cisco online software center. Other image distribution solutions do not scale to cover thousands of devices and cannot distribute images to devices behind a firewall or using Network Address Translation (NAT). The CNS image agent enables the managed device to initiate a network connection and request an image download allowing devices using NAT, or behind firewalls, to access the image server.

The CNS image agent can be configured to use the CNS event bus. To use the CNS event bus, the CNS event agent must be enabled and connected to the CNS event gateway in the CNS Configuration Engine. The CNS image agent can also use an HTTP server that understands the CNS image agent protocol. Deployment of CNS image agent operations can use both the CNS event bus and an HTTP server.

Config Logger

Config logger tracks and reports configuration changes. Config logger supports two content types:

- Plain text--With plain-text format, the config logger reports configuration changes only.
- XML--The config logger uses XML to report the configuration change details (for example, what changed, who changed it, when changes were made, parser return code [PRC] values, and incremental NVGEN results).

IP SLAs for IPv6

Cisco IP Service Level Agreements (SLAs) are a portfolio of technology embedded in most devices that run Cisco software that allows Cisco customers to analyze IPv6 service levels for IPv6 applications and services, increase productivity, lower operational costs, and reduce the frequency of network outages. IP SLAs uses active traffic monitoring--the generation of traffic in a continuous, reliable, and predictable manner--for measuring network performance.

The following Cisco IP SLAs are supported for IPv6:

- Internet Control Message Protocol (ICMP) echo operation--Used to monitor end-to-end response time between a Cisco device and other devices using IPv4 or IPv6. ICMP echo is useful for troubleshooting network connectivity issues.
- TCP connect operation--Used to measure the response time taken to perform a TCP Connect operation between a Cisco device and other devices using IPv4 or IPv6.
- User Datagram Protocol (UDP) echo operation--Used to monitor end-to-end response time between a Cisco router and devices using IPv4 or IPv6 .
- UDP jitter operation--Used to analyze round-trip delay, one-way delay, one-way jitter, one-way packet loss, and connectivity in networks that carry UDP traffic in IPv4 or IPv6 networks.
- UDP jitter operation--Used to monitor VoIP quality levels in your network, allowing you to guarantee VoIP quality levels to your users in IPv4 or IPv6 networks.

How to Implement IPv6 for Network Management

- [Enabling Telnet Access to an IPv6 Device and Establishing a Telnet Session, page 6](#)
- [Enabling SSH on an IPv6 Router, page 7](#)
- [Configuring an SNMP Notification Server over IPv6, page 9](#)
- [Configuring Cisco IOS XE IPv6 Embedded Management Components, page 12](#)

Enabling Telnet Access to an IPv6 Device and Establishing a Telnet Session

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 host** *name* [*port*] *ipv6-address*
4. **line** [**aux** | **console** | **tty** | **vty**] *line-number* [*ending-line-number*]
5. **password** *password*
6. **login** [**local** | **tacacs**]
7. **ipv6 access-class** *ipv6-access-list-name* {**in** | **out**}
8. **telnet** *host* [*port*] [*keyword*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 host <i>name</i> [<i>port</i>] <i>ipv6-address</i> Example: Device(config)# ipv6 host cisco-sj 2001:DB8:20:1::12	Defines a static hostname-to-address mapping in the hostname cache.
Step 4	line [aux console tty vty] <i>line-number</i> [<i>ending-line-number</i>] Example: Device(config)# line vty 0 4	Creates a vty interface.

Command or Action	Purpose
Step 5 <code>password <i>password</i></code> Example: <code>Device(config)# password hostword</code>	Creates a password that enables Telnet.
Step 6 <code>login [<i>local</i> <i>tacacs</i>]</code> Example: <code>Device(config)# login tacacs</code>	(Optional) Enables password checking at login.
Step 7 <code>ipv6 access-class <i>ipv6-access-list-name</i> {<i>in</i> <i>out</i>}</code> Example: <code>Device(config)# ipv6 access-list hostlist</code>	(Optional) Adds an IPv6 access list to the line interface. <ul style="list-style-type: none"> Using this command restricts remote access to sessions that match the access list.
Step 8 <code>telnet <i>host</i> [<i>port</i>] [<i>keyword</i>]</code> Example: <code>Device(config)# telnet cisco-sj</code>	Establishes a Telnet session from a device to a remote host using either the hostname or the IPv6 address. <ul style="list-style-type: none"> The Telnet session can be established to a device name or to an IPv6 address.

Enabling SSH on an IPv6 Router

If you do not configure SSH parameters, then the default values will be used.

Prior to configuring SSH over an IPv6 transport, ensure that the following conditions exist:

- An IPsec (Data Encryption Standard [DES] or 3DES) encryption software image is loaded on your router. IPv6 transport for the SSH server and SSH client requires an IPsec encryption software image.
- A hostname and host domain are configured for your router. Refer to the "Mapping Hostnames to IPv6 Addresses" section of Implementing IPv6 Addressing and Basic Connectivity for information on assigning hostnames to IPv6 addresses and specifying default domain names that can be used by both IPv4 and IPv6.
- A Rivest, Shamir, and Adelman (RSA) key pair, which automatically enables SSH, is generated for your router. RSA is the public key cryptographic system developed by Ron Rivest, Adi Shamir, and Leonard Adelman. RSA keys come in pairs: one public key and one private key.
- A user authentication mechanism for local or remote access is configured on your router.

**Note**

The basic restrictions for SSH over an IPv4 transport listed in the "Configuring Secure Shell" chapter of *Cisco IOS XE Security Configuration Guide* apply to SSH over an IPv6 transport. In addition to the restrictions listed in that chapter, the use of locally stored usernames and passwords is the only user authentication mechanism supported by SSH over an IPv6 transport; the TACACS+ and RADIUS user authentication mechanisms are not supported over an IPv6 transport.

**Note**

To authenticate SSH clients, configure TACACS+ or RADIUS over an IPv4 transport and then an SSH server over an IPv6 transport.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ssh [timeout *seconds* | authentication-retries *integer*]**
4. **exit**
5. **ssh [-v {1 | 2}] [-c {3des | aes128-cbc | aes192-cbc | aes256-cbc}] [-l *userid* | -I *userid* : {*number*} {*ip-address*} | -l *userid* :rotary {*number*} {*ip-address*}] [-m {hmac-md5 | hmac-md5-96 | hmac-sha1 | hmac-sha1-96}] [-o *numberofpasswordprompts* n] [-p *port-num*] {*ip-addr* | *hostname*} [*command*]**

DETAILED STEPS

Command or Action	Purpose
Step 1 enable Example: <pre>Router> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
Step 2 configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.

Command or Action	Purpose
<p>Step 3 <code>ip ssh [timeout <i>seconds</i> authentication-retries <i>integer</i>]</code></p> <p>Example:</p> <pre>Router(config)# ip ssh timeout 100 authentication-retries 2</pre>	<p>Configures SSH control variables on your router.</p> <ul style="list-style-type: none"> You can specify the timeout in seconds, not to exceed 120 seconds. The default is 120. This setting applies to the SSH negotiation phase. Once the EXEC session starts, the standard timeouts configured for the vty apply. <p>By default, five vty lines are defined (0-4); therefore, five terminal sessions are possible. After the SSH executes a shell, the vty timeout starts. The vty timeout defaults to 10 minutes.</p> <ul style="list-style-type: none"> You can also specify the number of authentication retries, not to exceed five authentication retries. The default is three.
<p>Step 4 <code>exit</code></p> <p>Example:</p> <pre>Router(config)# exit</pre>	<p>Exits configuration mode, and returns the router to privileged EXEC mode.</p>
<p>Step 5 <code>ssh [-v {1 2}] [-c {3des aes128-cbc aes192-cbc aes256-cbc}] [-l <i>userid</i> -I <i>userid</i> : {<i>number</i>} {<i>ip-address</i>} -I <i>userid</i> :rotary {<i>number</i>} {<i>ip-address</i>}] [-m {hmac-md5 hmac-md5-96 hmac-sha1 hmac-sha1-96}] [-o <i>numberofpasswordprompts</i><i>n</i>] [-p <i>port-num</i>] {<i>ip-addr</i> <i>hostname</i>} [<i>command</i>]</code></p> <p>Example:</p> <pre>Router# ssh</pre>	<p>Starts an encrypted session with a remote networking device.</p>

Configuring an SNMP Notification Server over IPv6

Use an SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to regulate access to the agent on the router. Optionally, you can specify one or more of the following characteristics associated with the string:

- An access list of IP addresses of the SNMP managers that are permitted to use the community string to gain access to the agent.
- A MIB view, which defines the subset of all MIB objects accessible to the given community.
- Read and write or read-only permission for the MIB objects accessible to the community.

You can configure one or more community strings. To remove a specific community string, use the **no snmp-server community** command.

The **snmp-server host** command specifies which hosts will receive SNMP notifications and whether you want the notifications sent as traps or inform requests. The **snmp-server enable traps** command globally

enables the production mechanism for the specified notification types (such as Border Gateway Protocol [BGP] traps, config traps, entity traps, and Hot Standby Router Protocol [HSRP] traps).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [**ipv6** *nacl*] [*access-list-number*]
4. **snmp-server engineID remote** {*ipv4-ip-address* | *ipv6-address*} [**udp-port** *udp-port-number*] [**vrf** *vrf-name*] *engineid-string*
5. **snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**context** *context-name*] [**read** *read-view*] [**write** *write-view*] [**notify** *notify-view*] [**access** [**ipv6** *named-access-list*]{*acl-number* | *acl-name*}]
6. **snmp-server host** {*hostname* | *ip-address*} [**vrf** *vrf-name*] [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port** *port*] [*notification-type*]
7. **snmp-server user** *username* *group-name* [**remote** *host* [**udp-port** *port*]] {**v1** | **v2c** | **v3** [**encrypted**] [**auth** {**md5** | **sha**} *auth-password*]} [**access** [**ipv6** *nacl*] [**priv** {**des** | **3des** | **aes**{**128** **192** | **256**}}] *privpassword*] {*acl-number* | *acl-name*}]
8. **snmp-server enable traps** [*notification-type*] [**vrrp**

DETAILED STEPS

Command or Action	Purpose
Step 1 enable Example: <pre>Router> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3 snmp-server community <i>string</i> [view <i>view-name</i>] [ro rw] [ipv6 <i>nacl</i>] [<i>access-list-number</i>] Example: <pre>Router(config)# snmp-server community mgr view restricted rw ipv6 mgr2</pre>	Defines the community access string.

Command or Action	Purpose
<p>Step 4 snmp-server engineID remote {<i>ipv4-ip-address</i> <i>ipv6-address</i>} [udp-port <i>udp-port-number</i>] [vrf <i>vrf-name</i>] <i>engineid-string</i></p> <p>Example:</p> <pre>Router(config)# snmp-server engineID remote 3ffe:b00:c18:1::3/127 remotev6</pre>	<p>(Optional) Specifies the name of the remote SNMP engine (or copy of SNMP).</p>
<p>Step 5 snmp-server group <i>group-name</i> {v1 v2c v3 {auth noauth priv}} [context <i>context-name</i>] [read <i>read-view</i>] [write <i>write-view</i>] [notify <i>notify-view</i>] [access [ipv6 <i>named-access-list</i>]{<i>acl-number</i> <i>acl-name</i>}]</p> <p>Example:</p> <pre>Router(config)# snmp-server group public v2c access ipv6 public2</pre>	<p>(Optional) Configures a new SNMP group, or a table that maps SNMP users to SNMP views.</p>
<p>Step 6 snmp-server host {<i>hostname</i> <i>ip-address</i>} [vrf <i>vrf-name</i>] [traps informs] [version {1 2c 3 {auth noauth priv}}] <i>community-string</i> [udp-port <i>port</i>] [<i>notification-type</i>]</p> <p>Example:</p> <pre>Router(config)# snmp-server host host1.com 2c vrf trap- vrf</pre>	<p>Specifies the recipient of an SNMP notification operation.</p> <p>Specifies whether you want the SNMP notifications sent as traps or informs, the version of SNMP to use, the security level of the notifications (for SNMPv3), and the recipient (host) of the notifications.</p>
<p>Step 7 snmp-server user <i>username</i> <i>group-name</i> [remote <i>host</i>] [udp-port <i>port</i>] {v1 v2c v3 [encrypted] [auth {md5 sha} <i>auth-password</i>]} [access [ipv6 <i>nacl</i>] [priv {des 3des aes{128 192 256}}] <i>privpassword</i>] {<i>acl-number</i> <i>acl-name</i>}]</p> <p>Example:</p> <pre>Router(config)# snmp-server user user1 bldg1 remote 3ffe:b00:c18:1::3/127 v2c access ipv6 public2</pre>	<p>(Optional) Configures a new user to an existing SNMP group.</p> <p>Note You cannot configure a remote user for an address without first configuring the engine ID for that remote host. This is a restriction imposed in the design of these commands; if you try to configure the user before the host, you will receive a warning message and the command will not be executed</p>
<p>Step 8 snmp-server enable traps [<i>notification-type</i>] [vrrp]</p> <p>Example:</p> <pre>Router(config)# snmp-server enable traps bgp</pre>	<p>Enables sending of traps or informs, and specifies the type of notifications to be sent.</p> <ul style="list-style-type: none"> • If a <i>notification-type</i> is not specified, all supported notification will be enabled on the router. • To discover which notifications are available on your router, enter the snmp-server enable traps ? command.

Configuring Cisco IOS XE IPv6 Embedded Management Components

Most IPv6 embedded management components are enabled automatically when IPv6 is enabled and do not need further configuration. To configure syslog over IPv6 or disable HTTP access to a router, refer to the tasks in the following sections:

- [Configuring Syslog over IPv6, page 12](#)
- [Disabling HTTP Access to an IPv6 Device, page 12](#)

Configuring Syslog over IPv6

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **logging host** *{ {ip-address | hostname} | {ipv6 ipv6-address | hostname} }* **[transport {udp [port port-number] | tcp [port port-number] [audit]}]** **[xml | filtered [stream stream-id]] [alarm [severity]]**

DETAILED STEPS

Command or Action	Purpose
Step 1 enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2 configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3 logging host <i>{ {ip-address hostname} {ipv6 ipv6-address hostname} }</i> [transport {udp [port port-number] tcp [port port-number] [audit]}] [xml filtered [stream stream-id]] [alarm [severity]] Example: <pre>Device(config)# logging host ipv6 AAAA:BBBB:CCCC:DDDD::FFFF</pre>	Logs system messages and debug output to a remote host.

Disabling HTTP Access to an IPv6 Device

HTTP access over IPv6 is automatically enabled if an HTTP server is enabled and the device has an IPv6 address. If the HTTP server is not required, it should be disabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no ip http server**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
	Example: Device> enable	
Step 2	configure terminal	Enters global configuration mode.
	Example: Device# configure terminal	
Step 3	no ip http server	Disables HTTP access.
	Example: Device(config)# no ip http server	

Configuration Examples for Implementing IPv6 for Network Management

- [Examples: Enabling Telnet Access to an IPv6 Device, page 13](#)
- [Examples: Configuring an SNMP Notification Server over IPv6, page 15](#)

Examples: Enabling Telnet Access to an IPv6 Device

The following examples provide information on how to enable Telnet and start a session to or from an IPv6 device. In the following example, the IPv6 address is specified as 2001:DB8:20:1::12, and the hostname is specified as cisco-sj. The **show host** command is used to verify this information.

```
Device# configure terminal
Device(config)# ipv6 host cisco-sj 2001:DB8:20:1::12
Device(config)# end
Device# show host
Default domain is not set
Name/address lookup uses static mappings
Codes:UN - unknown, EX - expired, OK - OK, ?? - revalidate
      temp - temporary, perm - permanent
```

```

      NA - Not Applicable  None - Not defined
Host      Port  Flags      Age Type  Address(es)
cisco-sj  None  (perm, OK)  0   IPv6  2001:DB8:20:1::12

```

To enable Telnet access to a device, create a vty interface and password:

```

Device(config)# line vty 0 4
password lab
login

```

To use Telnet to access the device, you must enter the password:

```

Device# telnet cisco-sj
Trying cisco-sj (2001:DB8:20:1::12)... Open
User Access Verification
Password:
cisco-sj
.
.
.
verification

```

It is not necessary to use the **telnet** command. Specifying either the hostname or the address is sufficient, as shown in the following examples:

```
Device# cisco-sj
```

or

```
Device# 2001:DB8:20:1::12
```

To display the IPv6 connected user (line 130) on the device to which you are connected, use the **show users** command:

```

Device# show users
      Line      User      Host(s)      Idle      Location
*   0 con 0      idle      idle      00:00:00
  130 vty 0      idle      idle      00:00:22  8800::3

```

Note that the address displayed is the IPv6 address of the source of the connection. If the hostname of the source is known (either through a domain name server [DNS] or locally in the host cache), then it is displayed instead:

```

Device# show users
      Line      User      Host(s)      Idle      Location
*   0 con 0      idle      idle      00:00:00
  130 vty 0      idle      idle      00:02:47  cisco-sj

```

If the user at the connecting device suspends the session with ^6x and then enters the **show sessions** command, the IPv6 connection is displayed:

```

Device# show sessions
Conn Host      Address      Byte  Idle Conn Name
*   1 cisco-sj  2001:DB8:20:1::12    0    0 cisco-sj

```

The Conn Name field shows the hostname of the destination only if it is known. If it is not known, the output might look similar to the following:

```

Device# show sessions
Conn Host      Address      Byte  Idle Conn Name
*   1 2001:DB8:20:1::12 2001:DB8:20:1::12    0    0 2001:DB8:20:1::12

```

Examples: Configuring an SNMP Notification Server over IPv6

The following example permits any SNMP to access all objects with read-only permission using the community string named public. The device also will send Border Gateway Protocol (BGP) traps to the IPv4 host 172.16.1.111 and IPv6 host 3ffe:b00:c18:1::3/127 using SNMPv1 and to the host 172.16.1.27 using SNMPv2c. The community string named public will be sent with the traps.

```
Device(config)# snmp-server community public
Device(config)# snmp-server enable traps bgp
Device(config)# snmp-server host 172.16.1.27 version 2c public
Device(config)# snmp-server host 172.16.1.111 version 1 public
Device(config)# snmp-server host 3ffe:b00:c18:1::3/127 public
```

Example: Associate an SNMP Server Group with Specified Views

In the following example, the SNMP context A is associated with the views in SNMPv2c group GROUP1 and the IPv6 named access list public2:

```
Device(config)# snmp-server context A
Device(config)# snmp mib community-map commA context A target-list commAVpn
Device(config)# snmp mib target list commAVpn vrf CustomerA
Device(config)# snmp-server view viewA ciscoPingMIB included
Device(config)# snmp-server view viewA ipForward included
Device(config)# snmp-server group GROUP1 v2c context A read viewA write viewA notify
access ipv6 public2
```

Example: Create an SNMP Notification Server

The following example configures the IPv6 host as the notification server:

```
Device> enable
Device# configure terminal
Device(config)# snmp-server community mgr view restricted rw ipv6 mgr2
Device(config)# snmp-server engineID remote 3ffe:b00:c18:1::3/127 remotev6
Device(config)# snmp-server group public v2c access ipv6 public2
Device(config)# snmp-server host host1.com 2c vrf trap-vrf
Device(config)# snmp-server user user1 bldg1 remote 3ffe:b00:c18:1::3/127 v2c access ipv6
public2
Device(config)# snmp-server enable traps bgp
Device(config)# exit
```

Additional References

Related Documents

Related Topic	Document Title
IPv6 supported features	" Start Here: Cisco IOS XE Software Release Specifics for IPv6 Features ," <i>Cisco IOS XE IPv6 Configuration Guide</i>
Basic IPv6 configuration tasks	" Implementing IPv6 Addressing and Basic Connectivity ," <i>Cisco IOS XE IPv6 Configuration Guide</i>

Related Topic	Document Title
IPv6 commands: complete command syntax, command mode, defaults, usage guidelines, and examples	<i>Cisco IOS IPv6 Command Reference</i>
SSH configuration information	<i>Cisco IOS Security Command Reference</i>
Cisco IOS master command list, all releases	Cisco IOS Master Command List, All Releases
IP SLAs for IPv6	<ul style="list-style-type: none"> • IP SLAs--Analyzing IP Service Levels Using the ICMP Echo Operation • IP SLAs--Analyzing IP Service Levels Using the TCP Connect Operation • IP SLAs--Analyzing IP Service Levels Using the UDP Echo Operation • IP SLAs--Analyzing IP Service Levels Using the UDP Jitter Operation • IP SLAs--Analyzing VoIP Service Levels Using the UDP Jitter Operation

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	--

MIBs

MIBs	MIBs Link
None	<p>To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL:</p> <p>http://www.cisco.com/go/mibs</p>

RFCs

RFCs	Title
RFC 1350	<i>The TFTP Protocol (Revision 2)</i>
RFC 2732	<i>Format for Literal IPv6 Addresses in URLs</i>
RFC 3414	<i>User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)</i>

RFCs	Title
RFC 3484	<i>Default Address Selection for Internet Protocol version 6 (IPv6)</i>
RFC 4292	IP Forwarding Table MIB
RFC 4293	<i>Management Information Base for the Internet Protocol (IP)</i>

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 Implementing IPv6 for Network 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 1 **Feature Information for Managing Cisco IOS XE Applications over IPv6**

Feature Name	Releases	Feature Information
CNS Agents for IPv6	Cisco IOS XE Release 2.1	CNS configuration and event agents use a CNS configuration engine to provide methods for automating initial Cisco IOS device configurations, incremental configurations, and synchronized configuration updates, and the configuration engine reports the status of the configuration load as an event to which a network monitoring or workflow application can subscribe.
IP SLAs for IPv6	Cisco IOS XE Release 2.1	IP SLAs are supported for IPv6.
IPv6 for Config Logger	Cisco IOS XE Release 2.1	Config logger tracks and reports configuration changes.
IPv6--Syslog over IPv6	Cisco IOS XE Release 2.1	<p>The Cisco IOS syslog process in IPv6 allows users to log syslog messages to external syslog servers and hosts with IPv6 addresses.</p> <p>The following command was modified by this feature: logging host</p>
IPv6 Services--IP-FORWARD-MIB Support	Cisco IOS XE Release 2.1	A MIB is a database of objects that can be managed on a device. The managed objects, or variables, can be set or read to provide information on the network devices and interfaces.
IPv6 Services--IP-MIB Support	Cisco IOS XE Release 2.1	A MIB is a database of objects that can be managed on a device. The managed objects, or variables, can be set or read to provide information on the network devices and interfaces.
IPv6 Services--RFC 4293 IP-MIB (IPv6 only) and RFC 4292 IP-FORWARD-MIB (IPv6 only)	Cisco IOS XE Release 2.1	IP-FORWARD-MIB and IP-MIB were updated to RFC 4292 and RFC 4293 standards, respectively.
IPv6 Support for TCL	Cisco IOS XE Release 2.1	IPv6 supports TCL.

Feature Name	Releases	Feature Information
SNMP over IPv6	Cisco IOS XE Release 2.1	<p>SNMP can be configured over IPv6 transport so that an IPv6 host can perform SNMP queries and receive SNMP notifications from a device running Cisco IOS IPv6.</p> <p>The following commands were modified by this feature: <code>snmp-server community</code>, <code>snmp-server engineID remote</code>, <code>snmp-server group</code>, <code>snmp-server host</code>, <code>snmp-server user</code></p>
SSH over an IPv6 Transport	Cisco IOS XE Release 2.1	<p>SSH in IPv6 functions the same and offers the same benefits as SSH in IPv4--the SSH Server feature enables an SSH client to make a secure, encrypted connection to a Cisco router and the SSH Client feature enables a Cisco router to make a secure, encrypted connection to another Cisco router or to any other device running an SSH server.</p> <p>The following command was modified by this feature: ssh</p>
Telnet Access over IPv6	Cisco IOS XE Release 2.1	<p>The Telnet client and server in the Cisco IOS software support IPv6 connections. A user can establish a Telnet session directly to the router using an IPv6 Telnet client, or an IPv6 Telnet connection can be initiated from the router.</p> <p>The following commands were modified by this feature: ipv6 access-class, <code>ipv6 host</code>, show host, show sessions, show users, <code>telnet</code></p>
TFTP File Downloading for IPv6	Cisco IOS XE Release 2.1	IPv6 supports TFTP file downloading and uploading.
TFTP IPv6 Support	Cisco IOS XE Release 3.4S	TFTP uses UDP over IPv4 or IPv6 as its transport, and can work over IPv4 and IPv6 network layers.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

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

© 2012 Cisco Systems, Inc. All rights reserved.