



# CHAPTER 1

## Using the XML Management Interface

---

This chapter describes how to use the XML management interface to configure devices.

This chapter includes the following sections:

- [Information About the XML Management Interface, page 1-1](#)
- [Licensing Requirements for the XML Management Interface, page 1-2](#)
- [Prerequisites to Using the XML Management Interface, page 1-3](#)
- [Using the XML Management Interface, page 1-3](#)
- [Example XML Instances, page 1-10](#)
- [Additional References, page 1-12](#)

## Information About the XML Management Interface

You can use the XML management interface to configure a device. The interface uses the XML-based Network Configuration Protocol (NETCONF) that allows you to manage devices and communicate over the interface with an XML management tool or a program.

NETCONF is implemented with an XML Schema (XSD) that allows you to enclose device configuration elements within a remote procedure call (RPC) message. From within an RPC message, you select one of the NETCONF operations that matches the type of command that you want the device to execute. You can configure the entire set of CLI commands on the device with NETCONF. For information about using NETCONF, see the [“Creating NETCONF XML Instances” section on page 1-6](#).

For more information about NETCONF, see [RFC 4741](#).

For more information about using the NETCONF protocol over the Secure Shell (SSH), see [RFC 4742](#).

This section includes the following topics:

- [NETCONF Layers, page 1-2](#)
- [SSH xmlagent, page 1-2](#)

**Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)**

## NETCONF Layers

Table 1-1 shows the NETCONF layers.

**Table 1-1** NETCONF Layers

Layer	Example
Transport protocol	SSH Version 2
RPC	<rpc>, <rpc-reply>
Operations	<get-config>, <edit-config>
Content	show or configuration command

A description of the four NETCONF layers is as follows:

- SSH transport protocol—Provides a secure, encrypted connection between a client and server.
- RPC tag—Introduces a configuration command from the requestor and the corresponding reply from the XML server.
- NETCONF operation tag—Indicates the type of configuration command.
- Content—Indicates the XML representation of the feature that you want to configure.

## SSH xmlagent

The device software provides a Secure Shell (SSH) service called xmlagent that supports NETCONF over SSH Version 2.



### Note

The xmlagent service is referred to as the XML server in the Cisco NX-OS software.

NETCONF over SSH is initiated by the exchange of a hello message between the client and the XML server. After the initial exchange, the client sends XML requests, which the server responds to with XML responses. The client and server terminate requests and responses with the character sequence ]]>]]>. Because this character sequence is not valid in XML, the client and the server can interpret when the messages end, which keeps communication synchronized.

The XML schemas that define XML configuration instances you can use are described in the [“Creating NETCONF XML Instances”](#) section on page 1-6.

## Licensing Requirements for the XML Management Interface

The following table shows the licensing requirements for this feature:

Product	License Requirement
NX-OS	The XML management interface requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the NX-OS licensing scheme, see the <i>Cisco Nexus 7000 Series NX-OS Licensing Guide, Release 4.0</i> .

***Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)***

## Prerequisites to Using the XML Management Interface

The XML management interface has the following prerequisites:

- SSH Version 2 software is installed on the client PC.
- If desired, an XML management tool of your choice that supports NETCONF over SSH and is accessible from the client PC.

## Using the XML Management Interface

This section describes how to manually configure and use the XML management interface.

You can use the XML management interface with the default settings on the device. You must have the SSH Version 2 software installed on the client PC.

Before you begin exchanging XML documents between a client PC and the device, you can verify the following conditions:

- SSH Version 2 is enabled on the device, which is the default device setting.
- Options are set for the XML server on the device.

This section includes the following topics:

- [Configuring SSH and the XML Server Options through the CLI, page 1-3](#)
- [Starting an SSH Session, page 1-5](#)
- [Sending the Hello Message, page 1-5](#)
- [Obtaining the XSD Files, page 1-6](#)
- [Creating NETCONF XML Instances, page 1-6](#)
- [Sending an XML Document to the XML Server, page 1-10](#)

## Configuring SSH and the XML Server Options through the CLI

By default, the SSH server is enabled on the device. If you disable SSH, you must enable it before you start an SSH session on the client PC.

You can configure XML server options to control the number of concurrent sessions and the timeout for active sessions. You can also enable XML document validation and terminate XML sessions.



### Note

---

The XML server timeout applies only to active sessions.

---

For more information about configuring SSH, see the *Cisco NX-OS Security CLI Configuration Guide*.

For more information about the XML commands, see the *Cisco NX-OS System Management Configuration Guide*.

### SUMMARY STEPS

1. **show xml server status**
2. **xml server validate session | all**

**Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)**

3. `xml server terminate session`
4. `config t`
5. `no ssh server enable`
6. `ssh server enable`
7. `show ssh server`
8. `xml server max-session sessions`
9. `xml server timeout seconds`
10. `show xml server status`
11. `copy running-config startup-config`

**DETAILED STEPS**

	<b>Command</b>	<b>Purpose</b>
<b>Step 1</b>	<code>show xml server status</code>  <b>Example:</b> switch# show xml server status	(Optional) Displays information about XML server settings and active XML server sessions. You can find session numbers in this output.
<b>Step 2</b>	<code>xml server validate session   all</code>  <b>Example:</b> switch# xml server validate all	Causes validation of XML documents for the specified server session.
<b>Step 3</b>	<code>xml server terminate session</code>  <b>Example:</b> switch# xml server terminate 8665	Terminates the specified XML server session.
<b>Step 4</b>	<code>config t</code>  <b>Example:</b> switch# config t switch(config)#	Enters configuration mode.
<b>Step 5</b>	<code>no ssh server enable</code>  <b>Example:</b> switch(config)# no ssh server enable	(Optional) Disables the SSH server so that you can generate keys. For information about generating keys, see the <i>Cisco NX-OS Security CLI Configuration Guide</i> .
<b>Step 6</b>	<code>ssh server enable</code>  <b>Example:</b> switch(config)# ssh server enable	Enables the SSH server. The default is enabled.
<b>Step 7</b>	<code>show ssh server</code>  <b>Example:</b> switch(config)# show ssh server	(Optional) Displays the status of the SSH server.

## Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)

	Command	Purpose
Step 8	<b>xml server max-session sessions</b>  <b>Example:</b> switch(config)# xml server max-session 6	Sets the number of allowed XML server sessions. The default is 8. The range is from 1 to 8.
Step 9	<b>xml server timeout seconds</b>  <b>Example:</b> switch(config)# xml server timeout 2400	Sets the number of seconds after which the XML server session is terminated. The default is 1200 seconds. The range is from 1 to 9223372036854775807.
Step 10	<b>show xml server status</b>  <b>Example:</b> switch(config)# show xml server status	(Optional) Displays information about the XML server settings and active XML server sessions.
Step 11	<b>copy running-config startup-config</b>  <b>Example:</b> switch(config)# copy running-config startup-config	(Optional) Saves the running configuration to the startup configuration.

## Starting an SSH Session

You can start an SSH Version 2 session on the client PC with a command similar to the following:

```
ssh2 username@ip-address -s xmlagent
```

You enter the login username, the IP address of the device, and the service to which to connect. The xmlagent service is referred to as the XML server in the device software.



### Note

The SSH command syntax may differ for the SSH software on the client PC.

If you do not receive the hello message from the XML server, verify the following conditions:

- The SSH server is enabled on the device.
- The XML server max-sessions option is adequate to support the number of SSH connections to the device.
- The active XML server sessions on the device are not all in use.

## Sending the Hello Message

When you start an SSH session to the XML server, the server responds immediately with a hello message that informs the client of the server's capabilities. You must advertise your capabilities to the server with a hello message before the server will process any other requests. The XML server supports only the base capabilities and expects support only for the base capabilities from the client.

[Example 1-1](#) and [Example 1-2](#) show sample hello messages from the server and client.



### Note

You must end all XML documents with `]]>]]>` to support synchronization in NETCONF over SSH.

## Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)

### Example 1-1 Hello Message from the Server

```
<?xml version="1.0"?>
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>urn:ietf:params:xml:ns:netconf:base:1.0</capability>
  </capabilities>
</hello>]]>]]>
```

### Example 1-2 Hello Message from the Client

```
<?xml version="1.0"?>
<nc:hello xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:capabilities>
    <nc:capability>urn:ietf:params:xml:ns:netconf:base:1.0</nc:capability>
  </nc:capabilities>
</nc:hello>]]>]]>
```

## Obtaining the XSD Files

To obtain a copy of the XSD files, follow these steps:

- 
- Step 1** From your browser, navigate to the Cisco software download site at the following URL: <http://www.cisco.com/kobayashi/sw-center/index.shtml>  
The Download Software page opens.
  - Step 2** In the Select a Software Product Category list, choose **Switches Software**.
  - Step 3** If you are not already logged in as a registered Cisco user, you are prompted to log in now.
  - Step 4** Under the Select a Device list, expand the Data Center Switches by clicking the plus (+).
  - Step 5** Expand a product series by clicking the plus (+).
  - Step 6** Choose the product.
  - Step 7** From the Select a Software Type list, choose **NX-OS XML Schema Definition**.
  - Step 8** Under Select a Release, expand the releases by clicking the plus (+).
  - Step 9** Choose the release.
  - Step 10** Choose the NX-OS XSD gzip file to the right of the release.
  - Step 11** If you are requested, follow the instructions to apply for eligibility to download strong encryption software images.
  - Step 12** Click **Download**.  
Cisco's End User License Agreement opens.
  - Step 13** Click **Agree** and follow the instructions to download the file to your PC.
- 

## Creating NETCONF XML Instances

You can create NETCONF XML instances by enclosing XML configuration elements within an RPC tag and NETCONF operation tags. The XML configuration elements are defined in feature-based XML schema definition (XSD) files, which represent the available CLI commands in an XML format.

## Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)

Example 1-3 shows the tags used in the NETCONF XML framework. Tag lines are marked with the following letter codes:

- X—XML declaration
- R—RPC request tag
- N—NETCONF operation tags
- D—Device tags

### Example 1-3 NETCONF XML Framework

```
X <?xml version="1.0"?>
R <nc:rpc message-id="1" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
R  xmlns="http://www.cisco.com/nxos:1.0:xml">
N  <nc:get>
N    <nc:filter type="subtree">
D      <show>
D        <xml>
D          <server>
D            <status/>
D          </server>
D        </xml>
D      </show>
N    </nc:filter>
N  </nc:get>
R </nc:rpc>]]>]]>
```



#### Note

You will need to use your own XML editor or XML management interface tool to create XML instances.

This section includes the following topics:

- [RPC Request Tag <rpc>](#), page 1-7
- [RPC Response Tag <rpc-reply>](#), page 1-7
- [NETCONF Operations](#), page 1-8
- [Multiple Configuration Elements in a Single XML Instance](#), page 1-9

## RPC Request Tag <rpc>

All NETCONF XML instances must begin with the RPC request tag <rpc>. Example 1-4 shows the <rpc> element with its required message-id attribute. The XML namespace is shown as it is used in device responses. You enclose the NETCONF operation tags and the XML configuration tags within the RPC tag. You follow the </rpc> end-tag with the XML termination character sequence.

### Example 1-4 RPC Request Tag <rpc>

```
<nc:rpc message-id="315" xmlns xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
...
</nc:rpc>]]>]]>
```

## RPC Response Tag <rpc-reply>

For every XML request sent by the client, the XML server sends an XML response enclosed in the RPC response tag <rpc-reply>. Example 1-5 shows the <rpc-reply> element.

## Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)

### Example 1-5 RPC Response Tag <rpc-reply>

```
<nc:rpc-reply message-id="315" xmlns xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</nc:rpc-reply>]]>]]>
```

The elements <ok>, <data>, or <rpc-error> can appear in the RPC response. [Table 1-2](#) describes the RPC response elements that can appear in the <rpc-reply> element.

**Table 1-2** RPC Response Elements

Element	Description
<ok>	The RPC request completed successfully. Used when no data is returned in the response.
<data>	The RPC request completed successfully. The data associated with the RPC request is enclosed in the <data> element.
<rpc-error>	The RPC request failed. Error information is enclosed in the <rpc-error> element.

## NETCONF Operations

NETCONF provides the configuration operations listed in [Table 1-3](#).

**Table 1-3** NETCONF Operations in Cisco NX-OS

NETCONF Operation	Description	Example
close-session	Closes the current XML server session.	<a href="#">NETCONF close-session Operation Example, page 1-11</a>
commit	Operation not supported.	—
copy-config	Operation not supported.	—
delete-config	Performs the equivalent of the <b>write erase</b> command on the startup configuration.	<a href="#">NETCONF delete-config Operation Example, page 1-11</a>
edit-config	Configures features in the running configuration of the device. You use this operation for configuration commands.	<a href="#">Interface Configuration Example, page 1-12</a>
get	Receives configuration information from the device. You use this operation for <b>show</b> commands. The source of the data is the running configuration.	<a href="#">Example 1-3 on page 1-7</a>
get-config	Operation not supported.	—
kill-session	Closes the specified XML server session. You cannot close your own session. See the close-session NETCONF operation.	<a href="#">NETCONF kill-session Operation Example, page 1-11</a>
lock	Operation not supported.	—
unlock	Operation not supported.	—
validate	Operation not supported.	—

***Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)***

## Multiple Configuration Elements in a Single XML Instance

You can specify multiple configuration elements in a single XML instance. The restriction is that the same NETCONF operation must be used for all the configuration elements. The XML server ensures that all elements in the request parse without errors before applying configuration elements. Any error causes the entire request to fail.

[Example 1-6](#) shows the XML instance that represents the combination of the following commands:

```
switch# show ssh server
switch# show xml server status
```

### **Example 1-6 Multiple Configuration Elements in a Single XML Instance**

```
<?xml version="1.0"?>
<nc:rpc message-id="1" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns="http://www.cisco.com/nxos:1.0:security_tree">
  <nc:get>
    <nc:filter type="subtree">
      <show>
        <ssh>
          <server>
          </server>
        </ssh>
      </show>
      <show>
        <xml>
          <server>
            <status/>
          </server>
        </xml>
      </show>
    </nc:filter>
  </nc:get>
</nc:rpc>]]>]]>
```

[Example 1-7](#) shows the XML server reply to [Example 1-6](#).



#### **Note**

The XML server may send replies to configuration elements in a different order than you define them in the XML instance.

### **Example 1-7 Response to Multiple Configuration Elements in a Single XML Instance**

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns="http://www.cisco.com/nxos:1.0:security_tree" message-id="1">
  <nc:data>
    <show>
      <xml>
        <server>
          <status>
            <__XML__OPT_Cmd_show_xml__readonly__>
              <__readonly__>
                <operational_status>
                  <o_status>enabled</o_status>
                </operational_status>
                <maximum_sessions_c>
                  <max_session>8</max_session>
                </maximum_sessions_c>
                <TABLE_sessions>
```

## Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)

```

<ROW_sessions>
  <session_id>24559</session_id>
  <user_name>admin</user_name>
  <start_time>Tue Mar 25 16:14:17 2008</start_time>
  <sap_id>10519</sap_id>
  <timeout>1200</timeout>
  <time_remaining_to_timeout>587</time_remaining_to_timeout>
  <ip_addr>171.71.55.134</ip_addr>
</ROW_sessions>
</TABLE_sessions>
</__readonly__>
</__XML_OPT_Cmd_show_xml__readonly__>
</status>
</server>
</xml>
</show>
</nc:data>
</nc:rpc-reply>
]]>]]><?xml version="1.0"?>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns="http://www.cisco.com/nxos:1.0:security_tree" message-id="1">
  <nc:data>ssh is enabled
version 2 enabled
</nc:data>
</nc:rpc-reply>
]]>]]>

```

## Sending an XML Document to the XML Server

To send an XML document to the XML server through an SSH session that you opened in a command shell, you can copy the XML text from an editor and paste it into the SSH session. Although typically, you will use an automated method to send XML documents to the XML server, you can verify the SSH connection to the XML server with this method.

Follow these guidelines with this method:

- Verify that the XML server sent the hello message immediately after you started the SSH session by looking for the hello message text in the command shell output.
- Send the client hello message before you send any XML requests. Because the XML server sends the hello response immediately, no additional response is sent after you send the client hello message.
- Always terminate the XML document with the character sequence `]]>]]>`.

## Example XML Instances

This section provides examples of XML instances of various types.

This section includes the following sections:

- [NETCONF delete-config Operation Example, page 1-11](#)
- [NETCONF close-session Operation Example, page 1-11](#)
- [NETCONF kill-session Operation Example, page 1-11](#)
- [Interface Configuration Example, page 1-12](#)

***Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)***

## NETCONF delete-config Operation Example

The following example shows the use of the NETCONF delete-config operation:

```
<?xml version="1.0"?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <delete-config>
    <target>
      <running/>
    </target>
  </delete-config>
</rpc>]]>]]>
```

The XML server reply to this example is as follows:

```
<?xml version="1.0"?>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns="http://www.cisco.com/nxos:1.0" message-id="101">
  <nc:ok/>
</nc:rpc-reply>]]>]]>
```

## NETCONF close-session Operation Example

The following example shows the use of the NETCONF close-session operation:

```
<?xml version="1.0"?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <close-session/>
</rpc>]]>]]>
```

The XML server reply to this example is as follows:

```
<?xml version="1.0"?>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns="http://www.cisco.com/nxos:1.0" message-id="101">
  <nc:ok/>
</nc:rpc-reply>]]>]]>
```

## NETCONF kill-session Operation Example

The following example shows the use of the NETCONF kill-session operation:

```
<?xml version="1.0"?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <kill-session>
    <session-id>31990</session-id>
  </kill-session>
</rpc>]]>]]>
```

The XML server reply to this example is as follows:

```
<?xml version="1.0"?>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns="http://www.cisco.com/nxos:1.0" message-id="101">
  <nc:ok/>
</nc:rpc-reply>]]>]]>
```

***Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)***

## Interface Configuration Example

This example shows the equivalent of the following CLI commands:

```
switch# config t
switch(config)# interface ethernet 2/30
switch(config-if)# description Marketing Network
```

The following example shows the use of the NETCONF operation edit-config:

```
<?xml version="1.0"?>
<nc:rpc message-id="16" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns="http://www.cisco.com/nxos:1.0:if_manager">
  <nc:edit-config>
    <nc:target>
      <nc:running/>
    </nc:target>
    <nc:config>
      <configure>
        <__XML__MODE__exec_configure>
          <interface>
            <ethernet>
              <interface>2/30</interface>
              <__XML__MODE_if-ethernet>
                <__XML__MODE_if-ethernet>
                  <description>
                    <desc_line>Marketing Network</desc_line>
                  </description>
                </__XML__MODE_if-ethernet>
              </__XML__MODE_if-ethernet>
            </ethernet>
          </interface>
        </__XML__MODE__exec_configure>
      </configure>
    </nc:config>
  </nc:edit-config>
</nc:rpc>]]>]]>
```

The XML server reply to this example is as follows:

```
<?xml version="1.0"?>
<nc:rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns="http://www.cisco.com/nxos:1.0:if_manager" message-id="16">
  <nc:ok/>
</nc:rpc-reply>]]>]]>
```

## Additional References

For additional information related to implementing the XML management interface, see the following sections:

- [Standards, page 1-13](#)
- [RFCs, page 1-13](#)

***Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)***

## 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.	—

## RFCs

RFCs	Title
<a href="#">RFC 4741</a>	<i>NETCONF Configuration Protocol</i>
<a href="#">RFC 4742</a>	<i>Using the NETCONF Configuration Protocol over Secure SHell (SSH)</i>

***Send document comments to [nexus7k-docfeedback@cisco.com](mailto:nexus7k-docfeedback@cisco.com)***