Extreme SLX-OS NETCONF Operations Guide, 17s.1.02

Supporting the ExtremeSwitching SLX 9140 and SLX 9240 Switches

53-1005316-02
22 September 2017

Contents

Preface

Document conventions

The document conventions describe text formatting conventions, command syntax conventions, and important notice formats used in Extreme technical documentation.

Notes, cautions, and warnings

Notes, cautions, and warning statements may be used in this document. They are listed in the order of increasing severity of potential hazards.

NOTE
A Note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

ATTENTION
An Attention statement indicates a stronger note, for example, to alert you when traffic might be interrupted or the device might reboot.

CAUTION
A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER
A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Text formatting conventions

Text formatting conventions such as boldface, italic, or Courier font may be used to highlight specific words or phrases.

Format	Description
bold text	Identifies command names.
	Identifies keywords and operands.
	Identifies the names of GUI elements.
	Identifies text to enter in the GUI.
italic text	Identifies emphasis.
	Identifies variables.
	Identifies document titles.
`Courier font`	Identifies CLI output.

Format	Description
	Identifies command syntax examples.

Command syntax conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

Convention	Description
bold text	Identifies command names, keywords, and command options.
italic text	Identifies a variable.
[ ]	Syntax components displayed within square brackets are optional. Default responses to system prompts are enclosed in square brackets.
{ **x** | **y** | **z** }	A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options.
x | **y**	A vertical bar separates mutually exclusive elements.
< >	Nonprinting characters, for example, passwords, are enclosed in angle brackets.
...	Repeat the previous element, for example, *member*[*member*...].
\	Indicates a "soft" line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

Extreme resources

Visit the Extreme website to locate related documentation for your product and additional Extreme resources.

White papers, data sheets, and the most recent versions of Extreme software and hardware manuals are available at www.extremenetworks.com. Product documentation for all supported releases is available to registered users at www.extremenetworks.com/support/documentation.

Document feedback

Quality is our first concern at Extreme, and we have made every effort to ensure the accuracy and completeness of this document. However, if you find an error or an omission, or you think that a topic needs further development, we want to hear from you.

You can provide feedback in two ways:

- Use our short online feedback form at http://www.extremenetworks.com/documentation-feedback-pdf/
- Email us at internalinfodev@extremenetworks.com

Provide the publication title, part number, and as much detail as possible, including the topic heading and page number if applicable, as well as your suggestions for improvement.

Contacting Extreme Technical Support

As an Extreme customer, you can contact Extreme Technical Support using one of the following methods: 24x7 online or by telephone. OEM customers should contact their OEM/solution provider.

If you require assistance, contact Extreme Networks using one of the following methods:

- GTAC (Global Technical Assistance Center) for immediate support
  - Phone: 1-800-998-2408 (toll-free in U.S. and Canada) or +1 408-579-2826. For the support phone number in your country, visit: www.extremenetworks.com/support/contact.
  - Email: support@extremenetworks.com. To expedite your message, enter the product name or model number in the subject line.
- GTAC Knowledge - Get on-demand and tested resolutions from the GTAC Knowledgebase, or create a help case if you need more guidance.
- The Hub - A forum for Extreme customers to connect with one another, get questions answered, share ideas and feedback, and get problems solved. This community is monitored by Extreme Networks employees, but is not intended to replace specific guidance from GTAC.
- Support Portal - Manage cases, downloads, service contracts, product licensing, and training and certifications.

Before contacting Extreme Networks for technical support, have the following information ready:

- Your Extreme Networks service contract number and/or serial numbers for all involved Extreme Networks products
- A description of the failure
- A description of any action(s) already taken to resolve the problem
- A description of your network environment (such as layout, cable type, other relevant environmental information)
- Network load at the time of trouble (if known)
- The device history (for example, if you have returned the device before, or if this is a recurring problem)
- Any related RMA (Return Material Authorization) numbers

About This Document

Supported hardware and software

In those instances in which procedures or parts of procedures documented here apply to some devices but not to others, this guide identifies exactly which devices are supported and which are not.

Although many different software and hardware configurations are tested and supported by Extreme Networks, Inc. for this SLX-OS release, documenting all possible configurations and scenarios is beyond the scope of this document.

The following hardware platforms are supported by this release:

- ExtremeSwitching SLX 9140
- ExtremeSwitching SLX 9240

NOTE
Some of the commands in this document use a slot/port designation. Because the SLX 9140 and the SLX 9240 do not contain line cards, the slot designation must always be "0" (for example, 0/1 for port 1).

What's new in this document

This document is released in conjunction with SLX-OS 17s.1.02.

On October 30, 2017, Extreme Networks, Inc. acquired the data center networking business from Brocade Communications Systems, Inc. This document has been updated to remove or replace references to Brocade Communications, Inc. with Extreme Networks., Inc., as appropriate

NETCONF Overview

NETCONF and YANG

Extreme SLX-OS provides support for the Network Configuration Protocol (NETCONF) and the YANG data modeling language. Using Extensible Markup Language (XML) constructs, the NETCONF protocol provides the ability to manipulate configuration data and view state data modeled in YANG. NETCONF uses a client/server architecture in which remote procedure calls (RPCs) manipulate the modeled data across a secure transport, such as Secure Shell version 2 (SSHv2).

NETCONF provides mechanisms through which you can perform the following operations:

- Manage network devices
- Retrieve configuration data and operational state data
- Upload and manipulate configurations

NETCONF is partitioned conceptually into four layers, as shown in

NETCONF in client/server architecture

The NETCONF protocol uses RPCs to facilitate communication between the client (NETCONF Manager or application) and the server (NETCONF Agent or managed device).

A client encodes an RPC request in XML and sends it to a server using a secure, connection-oriented session. The server responds with a reply encoded in XML.

The communication between the client and server consists of a series of alternating request and reply messages. The NETCONF peers use <rpc> and <rpc-reply> elements to provide transport protocol-independent framing of NETCONF requests and responses. The NETCONF server processes the RPC requests sequentially in the order in which they are received.

> **NOTE**
> As of Extreme SLX-OS 17s.1.02 release, a NETCONF <edit-config> RPC which maps to a single (CLI) configuration command is supported. A NETCONF <edit-config> RPC which maps to multiple configuration (CLI) commands might fail.

RPC request

The <rpc> element is used for enclosing a NETCONF request sent from the client to the server.

Every <rpc> element contains a mandatory attribute, the message-id. This attribute has a unique value for every RPC request, and is used to associate every RPC request with the corresponding response. The message-id value is a monotonically increasing integer string. The maximum length of the string is 4095 characters. If the message-id is not present in the RPC request, the server rejects the request by returning an <rpc-error> with an <error-tag> element set to “missing-attribute”.

If there are any additional attributes present in the RPC request, the NETCONF server returns them unmodified in the corresponding RPC reply.

RPC reply

An <rpc-reply> element is sent in response to every RPC request.

The <rpc-reply> element contains the mandatory attribute message-id copied from the corresponding RPC request, along with any additional attributes that are present in the RPC request.

For successfully processed <get> or <get-config> requests, the response data is encoded as the content of the <rpc-reply> element.

For successfully processed <edit-config> or <close-session> requests, the <ok> element is encoded as the content of the <rpc-reply> element.

For unsuccessful RPC requests, one or more <rpc-error> elements are encoded inside the <rpc-reply> element.

RPC and error handling

If the RPC request fails, an <rpc-error> element is encoded inside the <rpc-reply> element and sent to the client.

The <rpc-error> element indicates the first detected error. The server is not required to detect or report multiple errors. If the server detects multiple errors then the order of the error detection and reporting is at the discretion of the server.

SSH subsystem

The NETCONF client must use Secure Shell Version 2 (SSHv2) as the network transport to connect to the NETCONF server. Only the SSHv2 protocol is supported as the NETCONF transport protocol.

To run NETCONF over SSHv2, the client establishes an SSH transport connection using the SSH transport protocol to the NETCONF port. The default NETCONF port is 830. The underlying SSH client and server exchange keys for message integrity and encryption.

The SSHv2 client invokes the ssh-userauth service to authenticate the user. All currently supported SSH user authentication methods such as the public-key, password, and keyboard-interactive authentications are supported for a NETCONF session also. If the SSH user authentication is disabled, the user is allowed full access.

On successful user authentication, the client invokes the ssh-connection service, also known as the SSH connection protocol. After the SSH session is established, the NETCONF client invokes NETCONF as an SSH subsystem called netconf.

RFC references

For details about NETCONF and YANG as defined by the Internet Engineering Task Force (IETF), refer to the following documents:

- RFC 6241, “NETCONF Configuration Protocol.”
- RFC 4742 “Using the NETCONF Configuration Protocol over Secure Shell (SSH).”
- RFC 6020, “YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)”
- RFC 6021, “Common YANG Data Types”

Using NETCONF

Establishing a NETCONF session

Up to 16 concurrent sessions can be established with a NETCONF server. A session times out if it is idle for 30 minutes.

Each NETCONF session begins with a handshake in which the NETCONF server and the client specify the NETCONF capabilities they support. The following sections describe the message exchange on starting a NETCONF session.

Hello messages exchange

After establishing a secure transport connection, both the NETCONF server and client send a <hello> element simultaneously to announce their capabilities and session identifier.

The NETCONF server must include the <session-id> element in the <hello> element. The <session-id> element contains the unique session value for the NETCONF session. If the client receives the <hello> element without the <session-id>, the client aborts the NETCONF session by closing the underlying SSH session.

The NETCONF client must not include the <session-id> element in the <hello> element. If the server receives the <hello> element with the <session-id>, the server aborts the NETCONF session by closing the underlying SSH session.

The NETCONF client must include a valid xmlns attribute in the <hello> element. If the server receives the <hello> element without a valid xmlns attribute, the server aborts the NETCONF session by closing the underlying SSH session.

The NETCONF client must include a base capability. The server receiving the <hello> element without a NETCONF base capability aborts the NETCONF session by closing the underlying SSH session.

The server receiving an <rpc> element without first receiving a <hello> element aborts the NETCONF session by closing the underlying SSH session.

The following example shows a <hello> element from the NETCONF server.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <capabilities>
        <capability>urn:ietf:params:netconf:base:1.0</capability>
        <capability>urn:ietf:params:netconf:capability:writable-running:1.0    </capability>
        <capability>urn:ietf:params:netconf:capability:startup:1.0</capability>
        <capability>urn:ietf:params:netconf:capability:xpath:1.0</capability>
        <capability>urn:ietf:params:netconf:capability:validate:1.0</capability>
        <capability>http://tail-f.com/ns/netconf/actions/1.0</capability>
        <capability>http://tail-f.com/ns/aaa/1.1?revision=2010-06-17&module=tailfaaa</capability>
        <capability>urn:brocade.com:mgmt:brocade-aaa?revision=2010-10-21&module=brocade-aaa</capability>
        <capability>urn:brocade.com:mgmt:brocade-aaa-ext?revision=2010-09-21&module=brocade-aaa-ext</
capability>
        <capability>urn:brocade.com:mgmt:brocade-cdp?revision=2010-08-17&module=brocade-cdp</capability>
        <capability>urn:brocade.com:mgmt:brocade-cee-map?revision=2011-04-18&module=brocade-cee-map</
capability>
    </capabilities>
    (output truncated)
    <session-id>4</session-id>
</hello>
```

Server capabilities

A NETCONF capability is a set of protocol extensions that supplements the base NETCONF specification.

A NETCONF capability is identified with a Uniform Resource Identifier (URI). Capabilities augment the base operations of the NETCONF server, describing both the additional operations and the contents allowed inside the operations. To support a capability, the NETCONF server must support all the dependent capabilities.

The following capabilities are supported on SLX-OS switches:

- Base capability—The set of operations and contents that any NETCONF implementation must support. The URI for the base capability is urn:ietf:param:xml:ns:netconf:base:1.0. Both the NETCONF client and server must support the base capability.
- Writable-running capability—Indicates that the device supports <edit-config> and <copy-config> operations where the <running> configuration is the target. The URI is urn:ietf:params:netconf:capability:writable-running:1.0.
- Startup capability—Supports separate datastores for the running and startup configuration. Operations performed on the running-config datastore do not affect the startup configuration until a <copy-config> operation is performed to explicitly copy the running configuration to the startup configuration. The URI for the startup capability is urn:ietf:params:netconf:capability:startup:1.0.
- Xpath capability—Supports XPath expressions in <filter> elements. <filter> elements are used in <get> and <get-config> operations to limit the scope of the retrieved data. The URI for the xpath capability is urn:ietf:params:netconf:capability:xpath:1.0.
- Validate capability—Allows validation to be performed on a configuration. The URI for the validate capability is urn:ietf:params:netconf:capability:validate:1.0.
- Actions capability—Allows operations to be performed on the datastore using the custom action mechanism for features that are supported by this mechanism in the YANG code. Refer to “Using the custom action mechanism” on page 17 for details. The URI for the actions capability is http://tail-f.com/ns/netconf/actions/1.0.
- tailf-aaa capability—Supports proprietary authentication, authorization, and accounting (AAA). The URI for the tailf-aaa capability is http://tail-f.com/ns/aaa/1.1?revision=2010-06-17&module=tailf-aaa.
- Extreme proprietary capabilities—A set of capabilities that support Extreme Network OS features. Each capability references a namespace containing instance data. Each namespace corresponds to a file containing the YANG module that models the data. For example the brocade-cee-map capability at URI urn:brocade.com:mgmt:brocade-cee-map?revision=2011-04-18&module=brocade-cee-map provides support for the features modeled in the brocade-cee-map module.

For an overview of each YANG module and structural details, refer to the Extreme SLX-OS YANG Reference Manual. For element definitions, refer to the specific YANG file .

NOTE
The Candidate Configuration capability and Confirmed Commit capability are not supported.

Client capabilities

The client must support the base capability.

In addition, we recommend that the client specify the identification capability with URI http://tail-f.com/ns/netconf/identification/1.0 while establishing a session with the server. This capability provides client information to the server, including the vendor, product name, and version of the client application in addition to user information. Server administrators can subsequently gather information about who is accessing the server using the show netconf client-capabilities command or the <get-netconf-client-capabilities>custom RPC. Refer to Appendix A, “Managing NETCONF,” for details.

The following example shows a <hello> element from the NETCONF client.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <capabilities>
        <capability>urn:ietf:params:netconf:base:1.0</capability>
        <capability>http://tail-f.com/ns/netconf/identification/1.0?
vendor=extreme&amp;product=bna&amp;version=3.0&amp;client-identity=adminUser</capability>
    </capabilities>
</hello>
```

Retrieving configuration data

You can retrieve configuration data using either the <get-config> or <get> RPC. RFC 4741, NETCONF Configuration Protocol specifies that the <get-config> RPC returns only configuration data while the <get> RPC returns configuration data and operational state data.

In the Extreme implementation, the <get> RPC does not return operational state data; Extreme instead provides a set of Custom RPCs and actions for returning operational state data. In the Extreme implementation, the <get-config> and <get> operations are essentially the same. This document will typically refer to the <get-config> operation, though <get> can be used equally.

The following example shows a client message that issues the <get-config> operation in its most basic form. It retrieves the entire running configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="200" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
            <source>
            <running/>
        </source>
    </get-config>
</rpc>
```

Such a request, however, typically results in an unwanted or unmanageable amount of output. To restrict the output to the portion of the configuration you want, Extreme supports two types of filtering: subtree filtering and xpath filtering.

For complete details about subtree filtering and xpath filtering, refer to the RFC 4741, The NETCONF Protocol. The following sections provide some examples.

Subtree filtering

Subtree filtering defines a point in the configuration hierarchy that limits the returned configuration data.

Only data at this point and the subtrees below it are returned. For example, to retrieve the loopback configuration for all loopback interfaces configured on the device, use the following filter. This operation returns all configuration data for all loop ports on the managed device.

```
<?xml version="1.0" ?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <routing-system xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <loopback xmlns="urn:brocade.com:mgmt:brocade-intf-loopback">
                        <id>1</id>
                    </loopback>
                </interface>
            </routing-system>
        </filter>
```

```
    </get-config>
</rpc>
```

The purpose of each filter element is as follows:

- The <filter> element tag contains a type statement that identifies the filter type as a subtree filter.
- The <interface> element constrains the output to the interface configuration in the urn:brocade.com:mgmt:brocade-interface namespace.
- The <loopback> element further constrains the output to the information under the <loopback> node. Used in this way, <loopback> is termed a containment node.

To further restrict the output and retrieve loopback configuration data for only one specific loopback interface, use the following filter. In this example, the <id> element is termed a content match node; the filter returns the values of all loopback attributes for the specified port.

```
<?xml version="1.0" ?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <routing-system xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <loopback xmlns="urn:brocade.com:mgmt:brocade-intf-loopback">
                        <id>1</id>
                        <vrf/>
                    </loopback>
                </interface>
            </routing-system>
        </filter>
    </get-config>
</rpc>
```

If all you want to know is the setting of one specific loopback port attribute, such as the name of VRF, use a filter such as the following. In this case, <vrf> suppresses the inclusion of all its sibling nodes. It is termed a selection node.

```
<?xml version="1.0" ?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <routing-system xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <loopback xmlns="urn:brocade.com:mgmt:brocade-intf-loopback">
                        <id>1</id>
                        <vrf>
                            <forwarding/>
                        </vrf>
                    </loopback>
                </interface>
            </routing-system>
        </filter>
    </get-config>
</rpc>
```

The following example retrieves the configuration for the loopback interface.

```
<?xml version="1.0" ?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
```

```
            <routing-system xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <loopback xmlns="urn:brocade.com:mgmt:brocade-intf-loopback">
                        <id>1</id>
                        <shutdown/>
                    </loopback>
                </interface>
            </routing-system>
        </filter>
    </get-config>
</rpc>
```

xpath filtering

Sometimes the data element that qualifies the information you want is at a lower level in the data hierarchy than the information you need.

For example, if you want to return a list of interfaces that are bound to a CoS-to-CoS mutation QoS map, the element to be used for the selection criteria (<cos-mutation>name</cos-mutation>) resides at a lower level in the hierarchy than the information to be retrieved (the interface name), as shown in the following representation of the QoS map structure. In such cases, you must use an xpath filter and not a subtree filter.

```
| +--rw ethernet [name]
+--rw name                        interface-type
.
.
.
+--rw qos:qos
+--rw qos:default-cos?            int32
+--rw qos:cos-mutation?            map-name-type
+--rw qos:cos-traffic-class?    map-name-type
+--rw qos:dscp-mutation?        map-name-type
```

The following example returns the interface names to which the CoS-to-CoS mutation QoS map named “test” is bound. In this case, the map named “test” is bound to interfaces 2/5 and 2/6. The <filter> element tag specifies that the filter type is xpath and also specifies the data path and selection criteria.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="205">
    <get-config>
        <source>
            <running></running>
        </source>
            <filter type="xpath" select="/interface/ethernet/qos[cos-mutation='test']"></filter>
    </get-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="205">
    <data>
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
            <ethernet>
                <name>0/5</name>
                <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                    <default-cos>0</default-cos>
                    <cos-mutation>test</cos-mutation>
                </qos>
            </ethernet>
            <ethernet>
                <name>0/6</name>
                <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                    <default-cos>0</default-cos>
                    <cos-mutation>test</cos-mutation>
                </qos>
            </ethernet>
        </interface>
```

```
    </data>
</rpc-reply>
```

Retrieving operational data

In the Extreme SLX-OS implementation of NETCONF, two mechanisms are used for retrieving operational data: Extreme custom RPCs and custom actions.

Custom RPC and action support is added to some of the YANG modules to support the return of specific operational data.

For a complete list of the Extreme Custom RPCs and actions, and their locations, refer to the *Extreme SLX-OS YANG Reference Manual*.

Using custom RPCs

If an RPC is defined in a YANG module, you can use that RPC to return the associated namespace information defined in its output elements.

For example, to return information about port-profiles to which interfaces are applied, you can use the <get-ip-interface> RPC defined in the brocade-interface-ext.yang file.

The brocade-interface-ext.yang file file defines the structure of the <get-ip-interface> RPC as follows:

```
+---x get-ip-interface
|  +--ro input
|  |  +--ro (request-type)?
|  |     +--:(get-request)
|  |        +--ro interface-type?   enumeration
|  |        +--ro interface-name?   union
|  +--ro output
|     +--ro interface [interface-type interface-name]
|     |  +--ro interface-type         enumeration
|     |  +--ro interface-name         union
|     |  +--ro if-name?               string
|     |  +--ro ip-address [ipv4]
|     |  |  +--ro ipv4         string
|     |  |  +--ro ipv4-type?   enumeration
|     |  |  +--ro broadcast?   string
|     |  |  +--ro ip-mtu?      interface:mtu-type
|     |  +--ro if-state?              enumeration
|     |  +--ro line-protocol-state?   enumeration
|     |  +--ro proxy-arp?             string
|     |  +--ro vrf?                   string
|     +--ro has-more?    boolean
```

The following example shows the <rpc> message and reply. The <get-ip-interface> element contains an xmlns attribute that identifies the corresponding namespace.

```
<rpc-reply message-id="307" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <interface-type>ethernet</interface-type>
        <interface-name>2/4</interface-name>
        <if-name>ethernet 2/4</if-name>
        <if-state>up</if-state>
        <line-protocol-state>down</line-protocol-state>
        <ip-address>
            <ipv4>unassigned</ipv4>
        </ip-address>
    </interface>
</rpc-reply>
```

Refer to the *Brocade SLX-OS YANG Reference Guide, 17r.1.00* for a list of Custom RPCs, a brief description of their function, and their location.

Retrieving operational data with pagination

Some RPCs return operational data that consists of lists of entities. For example, an RPC might return detailed information about every interface. For these kinds of applications, to make the output manageable, pagination is supported by providing a <has-more> element in the output of the RPC.

The following example shows how the <has-more> element works to provide pagination for the <get-vlan-brief> RPC. In the input, you can request information about a specific VLAN, or about all VLANs by not providing an input parameter. If you request input about all VLANs, you will first receive information about the VLAN with the lowest VLAN ID. You can then check the <has-more> element in the output to determine whether information is available for additional VLANs. If <has-more> is true, use the value returned in <last-vlan-id> as the <last-rcvd--vlan-id> input parameter to the next call to <get-vlan-brief>. The <get-vlan-brief> RPC then returns the next available VLAN. Continue until <has-more> returns false.

```
+---x get-vlan-brief
    +--ro input
    | +--ro (request-type)?
    | +--:(get-request)
    | | +--ro vlan-id? interface:vlan-type
    | +--:(get-next-request)
    | +--ro last-rcvd-vlan-id? interface:vlan-type
    +--ro output
      +--ro vlan [vlan-id]
    | +--ro vlan-id interface:vlan-type
    | +--ro vlan-type? enumeration
    | +--ro vlan-name? string
    | +--ro vlan-state? enumeration
    | +--ro interface [interface-type interface-name]
    | +--ro interface-type enumeration
    | +--ro interface-name union
    | +--ro tag? enumeration
    +--ro last-vlan-id? interface:vlan-type
    +--ro has-more? boolean
```

The following example uses the <get-interface-brief> RPC to return information about the first VLAN. In this case, the first VLAN is VLAN 20.

```
<rpc message-id="207" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-vlan-brief xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
    </get-vlan-brief>
</rpc>

<rpc-reply message-id="207" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <vlan xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <vlanid>20</vlanid>
        <vlan-type>static</vlan-type>
        <vlan-name>vlan-20</vlan-name>
        <vlan-state>active</valan-state>
        <interface>
            <interface-type>ethernet</interface-type>
            <interfce-name>2/5</interface-name>
            <tag>tagged</tag>
        </interface>
    </vlan>
    <last-vlan-id>20</last-vlan-id>
    <has-more>true</has-more>
</rpc-reply>
```

The <has-more> field is true, so use the value returned in <last-vlan-id> as the <last-rcvd-vlan-id> in the next call to <get-vlan-brief> to return information about the next VLAN.

```
<rpc message-id="208" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-vlan-brief xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <last-rcvd-vlan-id>20</last-rcvd-vlan-id>
    </get-vlan-brief>
</rpc>

<rpc-reply message-id="208" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <vlan xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <vlanid>30</vlanid>
        <vlan-type>static</vlan-type>
        <vlan-name>vlan-30</vlan-name>
        <vlan-state>active</valan-state>
        <interface>
            <interface-type>ethernet</interface-type>
            <interfce-name>2/5</interface-name>
            <tag>tagged</tag>
        </interface>
    </vlan>
    <last-vlan-id>30</last-vlan-id>
    <has-more>false</has-more>
</rpc-reply>
```

If the <has-more> field returns false, no more VLAN data can be retrieved.

Using the custom action mechanism

An action is a proprietary mechanism used for implementing operations that do not affect the configuration datastore. Several implementations of actions exist in the Extreme SLX-OS implementation for retrieving operational information.

The following structure is defined in the brocade-arp.yang module for displaying operational data related to arp.

```
+--rw arp-entry [arp-ip-address]
      +--rw arp-ip-address       inet:ipv4-address
      +--rw mac-address-value?   mac-access-list:mac-address-type
      +--rw interfacename?       enumeration
      +--rw (interfacetype)?
         +--:(Ethernet)
         |  +--rw Ethernet?            interface:interface-type
         +--:(Ve)
            +--rw Ve?                 interface:vlan-type
```

The following example shows use of an action.

```
<get-arp xmlns="urn:brocade.com:mgmt:extreme-arp"></get-arp>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <arp-entry>
        <ip-address>20.0.0.122</ip-address>
        <mac-address>0005.3379.407a</mac-address>
        <interface-type>unknown</interface-type>
        <interface-name></interface-name>
        <is-resolved>true</is-resolved>
        <age>03:16:05</age>
        <entry-type>dynamic</entry-type>
    </arp-entry>
</rpc-reply>
```

For a list of available actions and their locations, refer to the *Extreme SLX-OS YANG Reference Manual*.

Editing the configuration

All configuration editing is done using the merge or delete operations of the <edit-config> RPC. The create and replace operations are not supported.

Refer to RFC 4741, The NETCONF Protocol, for details about these operations.

> NOTE
> Every NETCONF <edit-config> request should have a one-to-one mapping with a Extreme SLX-OS CLI command. You cannot combine two CLI operations into one NETCONF request.

The following example of the default merge operation adds a static address to the MAC address table. The operation is performed on the running configuration and configures the <mac-address-table> node in the urn:Extreme.com:mgmt:Extreme-mac-address-table namespace.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="210" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac-address-table xmlns="urn:extreme.com:mgmt:extreme-mac-address-table">
                <static>
                    <mac-address>0011.2222.3333</mac-address>
                    <forward>forward</forward>
                    <interface-type>ethernet</interface-type>
                    <interface-name>2/5</interface-name>
                    <vlan>vlan</vlan>
                    <vlanid>100</vlanid>
                </static>
            </mac-address-table>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="210" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The delete operation is used to remove or disable part of the configuration. The following example disables MSTP on the managed device.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="211" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:extreme.com:mgmt:extreme-interface">
                <spanning-tree xmlns="urn:extreme.com:mgmt:extreme-xstp">
                    <mstp xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" Operation="delete"/>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="211" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Managing the configuration

Exteme SLX-OS provides the custom <bna-config-cmd> PRC for performing any of the following operations:

- Copy the *running-config* file to a remote file.
- Copy a remote file to the *running-config* file.

Some simple examples are provided here. Refer to the *Extreme SLX-OS Administrator's Guide* for the following related information:

- General configuration management concepts
- Details and recommendations about how to apply these operations in a modular chassis or a Extreme VCS Fabric or a IP Fabric
- How to perform management configuration using the Extreme SLX-OS command line interface (CLI)

To monitor the progress of the copy operation, issue the <bna-config-cmd-status> custom RPC. Provide the session-ID returned by the corresponding <bna-config-cmd> as the input parameter.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="212">
    <bna-config-cmd-status xmlns="urn:extreme.com:mgmt:extreme-ras">
        <session-id>5</session-id>
    </bna-config-cmd-status>
</rpc>

<rpc-reply message-id="212" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <status xmlns="urn:extreme.com:mgmt:extreme-ras">completed</status>
</rpc-reply>
```

To archive or back up the *running-config* file, specify <running/> as the <src> parameter, and the URL of the archive as the <dest> parameter. The following example archives the running-config file.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="212">
    <bna-config-cmd xmlns="urn:extreme.com:mgmt:Extreme-ras">
        <src>running-config</src>
        <dest>https://user@extreme.com:passphrase/cfg/archiveMay7.txt</dest>
    </bna-config-cmd>
</rpc>

<rpc-reply message-id="212" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <session-id xmlns="urn:extreme.com:mgmt:extreme-ras">6</session-id>
    <status xmlns="urn:extreme.com:mgmt:extreme-ras">in-progress</status>
</rpc-reply>
```

To restore an archived configuration, specify the archive URL as the <source> parameter and <running/> as the <target>.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="212">
    <bna-config-cmd xmlns="urn:extreme.com:mgmt:extreme-ras">
        <src>https://user@Extreme.com:passphrase/cfg/archiveMay7.txt</src>
        <dest>running-config</dest>
    </bna-config-cmd>
</rpc>

<rpc-reply message-id="212" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <session-id xmlns="urn:extreme.com:mgmt:extreme-ras">6</session-id>
    <status xmlns="urn:extreme.com:mgmt:extreme-ras">in-progress</status>
</rpc-reply>
```

Disconnecting from a NETCONF session

To disconnect from a NETCONF session, issue the standard <close-session> RPC.

This operation causes the server to release any resources associated with the session and gracefully close any associated connections.

```
<rpc message-id="215" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <close-session/>
</rpc>

<rpc-reply message-id="215" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The <kill-session> RPC is also supported. Issuing <kill-session> aborts all operations and closes the session.

Sample use cases for SLX-OS NETCONF

This chapter discusses common use cases for the Extreme SLX-OS NETCONF.

NOTE

The information provided in this chapter may not cover the end-to-end configuration. Refer to the *Extreme SLX-OS Administrator's Guide* for the complete set of configuration tasks.

VRF configuration

VRF (Virtual Routing and Forwarding) is a technology that controls information flow within a network by isolating the traffic by partitioning the network into different logical VRF domains.

Every VRF-capable router supports one routing table for each VRF instance. Each VRF-capable router can function as a group of multiple virtual routers on the same physical router. VRF, in conjunction with virtual private network (VPN) solutions, guarantees privacy of information and isolation of traffic within its logical VRF domain.

This chapter provides procedures and examples for configuring VRF using the NETCONF interface.

Using the NETCONF interface, you can perform the following VRF configuration operations:

- Use the *<edit-config>* remote procedure call (RPC) to activate and deactivate VRF globally, set global VRF parameters, activate and deactivate VRF on a port, and to set interface parameters on a specific port.
- Use the <get-config> RPC to verify all or part of the VRF configuration.

VRF parameters are defined in the *extreme-vrf* YANG module. For a structural map of the YANG module, refer to the *Extreme SLX-OS YANG Reference Manual*. For definitions and explanations of all VRF parameters, refer to the *extreme-vrf.yang file*.

Configuring VRF

This chapter provides procedures and examples for configuring VRF using the NETCONF interface.

1. Configure VRF "red".

```
<vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
        <vrf-name>red</vrf-name>
    </vrf>
```

2. Enable the IPv4 or IPv6 address-family support to configure a variety of VRF unicast routing options.

   The below example shows how to enable IPv4 address-family support

```
<vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
        <vrf-name>Red</vrf-name>
        <address-family>
            <ip>
                <unicast></unicast>
            </ip>
        </address-family>
    </vrf>
```

3. Configure the maximum number of routes to be used for the VRF

```
<vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
        <vrf-name>red</vrf-name>
        <address-family>
            <ip>
                <unicast>
                    <max-route>200</max-route>
                </unicast>
            </ip>
        </address-family>
    </vrf>
```

4. Enable the Open Shortest Path First (OSPF) routing protocol over virtual forward and routing (VRF).

```
<router>
        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf”>
            <vrf>red</vrf>
        </ospf>
    </router>
```

5. Assign it to an area

```
<router>
        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
            <vrf>red</vrf>
            <area>
                <area-id>0</area-id>
            </area>
        </ospf>
    </router>
```

6. Bind the interface to the VRF instance

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <ve>
            <name>1</name>
            <vrf xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                <forwarding>red</forwarding>
            </vrf>
        </ve>
    </interface>
```

STP overview

A network topology of bridges typically contains redundant connections to provide alternate paths in case of link failures. However, because there is no concept of TTL in Ethernet frames, this could result in the permanent circulation of frames if there are loops in the network. To prevent loops, a spanning tree connecting all the bridges is formed in real time.

The redundant ports are put in a blocking (nonforwarding) state. They are enabled when required. In order to build a spanning tree for the bridge topology, the bridges must exchange control frames (BPDUs - Bridge Protocol Data Units). The protocols define the semantics of the BPDUs and the required state machine. The first Spanning Tree Protocol (STP) became part of the IEEE 802.1d standard.

The STP interface states for every Layer 2 interface running STP are as follows:

- *Blocking* - The interface does not forward frames.
- *Listening* - The interface is identified by the spanning tree as one that should participate in frame forwarding. This is a transitional state after the blocking state.
- *Learning* - The interface prepares to participate in frame forwarding.
- *Forwarding* - The interface forwards frames.

- *Disabled* - The interface is not participating in spanning tree because of a shutdown port, no link on the port, or no spanning tree instance running on the port.

A port participating in spanning tree moves through these states:

- From initialization to blocking
- From blocking to listening or to disabled
- From listening to learning or to disabled
- From learning to forwarding, blocking, or disabled
- From forwarding to disabled

Configuring STP

The process for configuring STP is as follows:

1. Enable STP using the below NETCONF statement.

```
<protocol xmlns="urn:extreme.com:mgmt:extreme-interface">
    <spanning-tree xmlns="urn:extreme.com:mgmt:extreme-xstp”>
        <stp></stp>
    </spanning-tree>
</protocol>
```

2. Designate the root switch by using the bridge-priority command. The range is 0 through 61440 and the priority values can be set only in increments of 4096.

```
<protocol xmlns="urn:extreme.com:mgmt:extreme-interface">
    <spanning-tree xmlns="urn:extreme.com:mgmt:extreme-xstp">
        <stp>
            <bridge-priority>32768</bridge-priority>
        </stp>
    </spanningtree>
</protocol>
```

3. Enable port fast on switch ports by using the **spanning-tree portfast** command.

   NOTE
   Note the following conditions:

   - Port fast only needs to be enabled on ports that connect to workstations or PCs. Repeat these commands for every port connected to workstations or PCs. Do not enable port fast on ports that connect to other switches.
   - If BPDUs are received on a port fast enabled interface, the interface loses the edge port status unless it receives a shut/no shut.
   - Enabling port fast on ports can cause temporary bridging loops, in both trunking and nontrunking mode.

```
<interface xmlns="urn:extreme.com:mgmt:extreme-interface">
    <ethernet>
        <name>2/5</name>
        <spanning-tree xmlns="urn:extreme.com:mgmt:extreme-xstp">
            <portfast>
                <portfastbasic></portfastbasic>
            </portfast>
        </spanning-tree>
    </ethernet>
</interface>
```

Configuring RSTP

The process for configuring RSTP is as follows.

1. Enable RSTP by using the global **protocol spanning-tree** command.

```
<protocol xmlns="urn:extreme.com:mgmt:extreme-interface">
    <spanning-tree xmlns="urn:extreme.com:mgmt:extreme-xstp">
        <rstp></rstp>
    </spanning-tree>
</protocol>
```

2. Designate the root switch by using the **bridge-priority** command. The range is 0 through 61440 and the priority values can be set only in increments of 4096.

```
<protocol xmlns="urn:extreme.com:mgmt:extreme-interface">
    <spanning-tree xmlns="urn:extreme.com:mgmt:extreme-xstp">
        <rstp>
            <bridge-priority>32768</bridge-priority>
        </rstp>
    </spanning-tree>
</protocol>
```

3. Configure the bridge forward delay value to set the time an interface spends in each of the listening and learning states.

```
<protocol xmlns="urn:extreme.com:mgmt:extreme-interface">
    <spanning-tree xmlns="urn:extreme.com:mgmt:extreme-xstp">
        <rstp>
            <forward-delay>30</forward-delay>
        </rstp>
    </spanning-tree>
</protocol>
```

4. Configure the bridge maximum aging time value to set the interval time in seconds between messages that the spanning tree receives from the interface.

```
<protocol xmlns="urn:extreme.com:mgmt:extreme-interface">
    <spanning-tree xmlns="urn:extreme.com:mgmt:extreme-xstp">
        <rstp>
            <max-age>40</max-age>
        </rstp>
    </spanning-tree>
</protocol>
```

5. Enable the error-disable-timeout timer.

```
<protocol xmlns="urn:extreme.com:mgmt:extreme-interface">
    <spanning-tree xmlns="urn:extreme.com:mgmt:extreme-xstp">
        <rstp>
            <error-disable-timeout>
                <enable></enable>
            </error-disable-timeout>
        </rstp>
    </spanning-tree>
</protocol>
```

6. Configure the error-disable-timeout interval value to set the timeout for errors on an interface.

```
<protocol xmlns="urn:extreme.com:mgmt:extreme-interface">
    <spanning-tree xmlns="urn:extreme.com:mgmt:extreme-xstp">
        <rstp>
            <error-disable-timeout>
                <interval>500</interval>
            </error-disable-timeout>
        </rstp>
    </spanning-tree>
</protocol>
```

NETCONF Statements

aaa

Configures AAA authentication.

Usage

```
<aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
    <aaa/>
</aaa-config>
```

History

Release version	History
17s.1.00	This NETCONF call was introduced.

aaa/accounting

Configures login or command accounting; either commands or login information are forwarded to accounting servers.

Usage

```
<aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
    <aaa>
        <accounting/>
    </aaa>
</aaa-config>
```

History

Release version	History
17s.1.00	This NETCONF call was introduced.

aaa/authentication/login

Configures the AAA login sequence.

Usage

```
<aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
    <aaa>
        <authentication>
            <login/>
        </authentication>
    </aaa>
</aaa-config>
```

Parameters

authentication

Specifies the authentication, authorization, and accounting (AAA) on the switch.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

aaa/authentication/login/first

Configures the primary source of authentication.

Usage

```
<aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
    <aaa>
        <authentication>
            <login>
                <first/>
            </login>
        </authentication>
    </aaa>
</aaa-config>
```

History

Release version	History
17s.1.00	This NETCONF call was introduced.

aaa/authentication/login/second

Configures the secondary source of authentication.

Usage

```
<aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
    <aaa>
        <authentication>
            <login>
                <second/>
            </login>
        </authentication>
    </aaa>
</aaa-config>
```

Parameters

authentication

Specifies authentication.

login

Specifies login.

second

Specifies second.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

aaa/username

Configures a username for AAA login.

Usage

```
<username xmlns="urn:brocade.com:mgmt:brocade-aaa">
    <name>{req_val}</name>
</username>
```

History

Release version	History
17s.1.00	This NETCONF call was introduced.

aaa/username/desc

Adds describes for the username.

Usage

```
<username xmlns="urn:brocade.com:mgmt:brocade-aaa">
    <name>{req_val}</name>
    <desc/>
</username>
```

Parameters

name

Specifies the username.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

aaa/username/encryption-level

Configures the encryption level for a username.

Usage

```
<username xmlns="urn:brocade.com:mgmt:brocade-aaa">
    <name>{req_val}</name>
    <encryption-level/>
</username>
```

Parameters

name

Specifies the username.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

aaa/username/role

Configures the role of the user.

Usage

```
<username xmlns="urn:brocade.com:mgmt:brocade-aaa">
    <name>{req_val}</name>
    <role/>
</username>
```

Parameters

name

Specifies the username.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

arp

Configures, modifies, or retrieves Address Resolution Protocol (ARP).

Usage

```
<arp-entry xmlns="urn:brocade.com:mgmt:brocade-arp">
    <arp-ip-address>{req_val}</arp-ip-address>
</arp-entry>
```

Parameters

arp-ip-address

The IP adress of the ARP entry.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

bgp

Provides a data model to manage Border Gateway Protocol (BGP).

Usage

```
<routing-system xmlns="urn:brocade.com:mgmt:brocade-common-def">
    <router>
        <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
            <router-bgp-attributes>
                <local-as/>
            </router-bgp-attributes>
        </router-bgp>
    </router>
</routing-system>
```

Parameters

local-as

Specifies local AS.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

bridge-domain

Configures a bridge domain.

Usage

```
<bridge-domain xmlns="urn:brocade.com:mgmt:brocade-bridge-domain">
    <bridge-domain-id>{req_val}</bridge-domain-id>
    <bridge-domain-type>{req_val}</bridge-domain-type>
</bridge-domain>
```

Parameters

bridge-domain-id

Specifies the bridge domian ID.

bridge-domain-type

Specifies the bridge domain type..

peer

Specifies the peer.

ip-address

Specifies the peer IP address.

load-balance

Specifies load-balance.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

bridge-domain/(bridge-domain-id)/(bridge-domain-type)/local-switching

Configures local switching.

Usage

```
<bridge-domain xmlns="urn:brocade.com:mgmt:brocade-bridge-domain">
    <bridge-domain-id>{req_val}</bridge-domain-id>
    <bridge-domain-type>{req_val}</bridge-domain-type>
    <local-switching/>
</bridge-domain>
```

Parameters

bridge-domain-id

Specifies the bridge-domain ID.

bridge-domain-type

Specifies the bridge domain type.

local-switching

Configures local switching.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

bridge-domain/(bridge-domain-id)/(bridge-domain-type)/logical-interface

Configures the logical interface.

Usage

```
<bridge-domain xmlns="urn:brocade.com:mgmt:brocade-bridge-domain">
    <bridge-domain-id>{req_val}</bridge-domain-id>
    <bridge-domain-type>{req_val}</bridge-domain-type>
    <logical-interface>
        <port-channel>
            <pc-lif-bind-id>{req_val}</pc-lif-bind-id>
        </port-channel>
    </logical-interface>
</bridge-domain>
```

Parameters

port-channel

Specifies Port Channel as the logical interface.

pc-lif-bind-id

Specifies the Port Channel LIF bind ID.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

bridge-domain/(bridge-domain-id)/(bridge-domain-type)/peer/(peer-ip)/cos

Sets the cos value in the range 0 to 7.

Usage

```
<bridge-domain xmlns="urn:brocade.com:mgmt:brocade-bridge-domain">
    <bridge-domain-id>{req_val}</bridge-domain-id>
    <bridge-domain-type>{req_val}</bridge-domain-type>
    <peer>
        <peer-ip>{req_val}</peer-ip>
        <cos/>
    </peer>
</bridge-domain>
```

Parameters

cos

Specifies the CoS. The range is from 0 through 7.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

bridge-domain-id/(bridge-domain-type)/peer/(peer-ip)/load-balance

Retrieves load-balancing details.

Usage

```
<bridge-domain xmlns="urn:brocade.com:mgmt:brocade-bridge-domain">
    <bridge-domain-id>{req_val}</bridge-domain-id>
    <bridge-domain-type>{req_val}</bridge-domain-type>
    <peer>
        <peer-ip>{req_val}</peer-ip>
        <load-balance/>
    </peer>
</bridge-domain>
```

Parameters

peer

Specifies the peer.

peer-ip

Specifies the peer IP address.

load-balance

Specifies load balance.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

bridge-domain/(bridge-domain-id)/(bridge-domain-type)/peer/(peer-ip)/lsp

Configures label-switched paths (LSPs).

Usage

```
<bridge-domain xmlns="urn:brocade.com:mgmt:brocade-bridge-domain">
    <bridge-domain-id>{req_val}</bridge-domain-id>
    <bridge-domain-type>{req_val}</bridge-domain-type>
    <peer>
        <peer-ip>{req_val}</peer-ip>
        <lsp/>
    </peer>
</bridge-domain>
```

Parameters

bridge-domain-id
: Specifies the bridge domain ID.

bridge-domain-id
: Specifies the bridge domain type.

peer
: Specifies the peer.

peer-ip
: Specifies the peer IP address.

lsp
: Specifies the LSP.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

bridge-domain/(bridge-domain-id)/(bridge-domain-type)/pw-profile

Sets the Pw-profile name.

Usage

```
<bridge-domain xmlns="urn:brocade.com:mgmt:brocade-bridge-domain">
    <bridge-domain-id>{req_val}</bridge-domain-id>
    <bridge-domain-type>{req_val}</bridge-domain-type>
    <pw-profile-name/>
</bridge-domain>
```

History

Release version	History
17s.1.00	This NETCONF call was introduced.

bridge-domain/(bridge-domain-id)/(bridge-domain-type)/statistics

Configures statistics for a bridge domain?

Usage

```
<bridge-domain xmlns="urn:brocade.com:mgmt:brocade-bridge-domain">
    <bridge-domain-id>{req_val}</bridge-domain-id>
    <bridge-domain-type>{req_val}</bridge-domain-type>
    <statistics/>
</bridge-domain>
```

History

Release version	History
17s.1.00	This NETCONF call was introduced.

bridge-domain/(bridge-domain-id)/(bridge-domain-type)/vc-id

Configures the VC ID for the bridge domain.

Usage

```
<bridge-domain xmlns="urn:brocade.com:mgmt:brocade-bridge-domain">
    <bridge-domain-id>{req_val}</bridge-domain-id>
    <bridge-domain-type>{req_val}</bridge-domain-type>
    <vc-id-num/>
</bridge-domain>
```

Parameters

vc-id-num

Specifies the VC ID.

bridge-domain-type

Specifies the bridge domain type.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

clock

Configures, modifies, or retrieves the system time zone.

Usage

```
<clock-sa xmlns="urn:brocade.com:mgmt:brocade-clock">
    <clock/>
</clock-sa>
```

History

Release version	History
17s.1.00	This NETCONF call was introduced.

clock/time-zone

Configures, modifies, or retrieves the system time zone.

Usage

```
<clock-sa xmlns="urn:brocade.com:mgmt:brocade-clock">
    <clock>
        <timezone/>
    </clock>
</clock-sa>
```

Parameters

timezone

Specifies the time zone.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

cluster

Configures, modifies, or retrieves a Multi-Chassis Trunking (MCT) cluster.

Usage

```
<node-id xmlns="urn:brocade.com:mgmt:brocade-node">
    <node-id>{req_val}</node-id>
    <cluster xmlns="urn:brocade.com:mgmt:brocade-cluster">
        <management>
            <principal-priority/>
        </management>
    </cluster>
</node-id>

<mgmt-cluster xmlns="urn:brocade.com:mgmt:brocade-cluster">
    <cluster>
        <management>
            <virtual>
                <ipv6>
                    <address>
                        <ipv6address>{req_val}</ipv6address>
                    </address>
                </ipv6>
            </virtual>
        </management>
    </cluster>
</mgmt-cluster>
```

Parameters

node-id
: Specifies the node ID.

management
: Specifies management.

virtual
: Specifies virtual.

principal-priority
: Specifies the principal priority.

ip*address*
: Specifies the IPv4 address or IPv6 address.

History

Release version	History
17s.1.01	This NETCONF call was introduced.

dot1x

Configures, retrieves, and modifies 802.1X authentication.

Usage

```
<dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x">
    <test>
        <timeout/>
    </test>
</dot1x>
```

Parameters

test timeout

The readiness test interval value in seconds. Valid values range from 1 through 65535. The default readiness test interval is 10 seconds.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

hardware

Configures, modifies, or retrieves the hardware management configuration.

Usage

```
<hardware xmlns="urn:brocade.com:mgmt:brocade-hardware">
    <connector>
        <name>{req_val}</name>
        <breakout>
            <cage-mode/>
        </breakout>
    </connector>
</hardware>
```

Parameters

connector *name*

Specifies the hadrware connnector name.

breakout *name*

Specifies breakout and enters cage-mode.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

interface/{interface-type}/{interface-name}/ipv6/ra-dns-server

Configures, modifies, or retrieves the Domain Name System (DNS) server address and the lifetime multiplier information to IPv6 hosts in the Router Advertisement (RA) message.

Usage

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <ethernet>
                    <name>{req_val}</name>
                    <ipv6>
                        <ipv6-nd-ra xmlns="urn:brocade.com:mgmt:brocade-ipv6-nd-ra">
                            <ipv6-intf-cmds>
                                <nd>
                                    <ra-dns-server>
                                        <dns-server-prefix>{req_val}</dns-server-prefix>
                                    </ra-dns-server>
                                </nd>
                            </ipv6-intf-cmds>
                        </ipv6-nd-ra>
                    </ipv6>
                </ethernet>
            </interface>
```

Parameters

ethernet
: Specifies Ethernet interface.

name
: Specifies the interface name.

ipv6
: Specifies IPv6 address.

nd
: Specifies the neighbor discovery protocol.

dns-server-prefix
: Specifies the prefix of the DNS server.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

interface/{interface-type}/{interface-name}/ipv6/ra-domain-name

Configures the domain name of the Domain Name System (DNS) suffix and the lifetime multiplier information to IPv6 hosts in the Router Advertisement (RA) message.

Usage

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <ethernet>
                    <name>{req_val}</name>
                    <ipv6>
                        <ipv6-nd-ra xmlns="urn:brocade.com:mgmt:brocade-ipv6-nd-ra">
                            <ipv6-intf-cmds>
                                <nd>
                                    <ra-domain-name>
                                        <domain-name-string>{req_val}</domain-name-string>
                                        <domain-name-lifetime-multiplier/>
                                    </ra-domain-name>
                                </nd>
                            </ipv6-intf-cmds>
                        </ipv6-nd-ra>
                    </ipv6>
                </ethernet>
            </interface>
```

Parameters

ethernet
Specifies Ethernet interface.

name
Specifies the interface name.

ipv6
Specifies IPv6.

ra-domain-name
Specifies the RA domain name.

domain-name-string
Specifies the domain name.

domain-name-lifetime-multiplier
Specifies domain name option and lifetime multiplier for DNS search list option.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

interface/{interface-type}/{interface-name}/link-error-disable

Configures port link dampening (PLD).

Usage

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <ethernet>
                    <name>{req_val}</name>
                    <link-error-disable xmlns="urn:brocade.com:mgmt:brocade-pld">
                        <wait-time-in-sec/>
                    </link-error-disable>
                </ethernet>
            </interface>
```

Parameters

ethernet

Specifies Ethernet interface.

name

Specifies the interface name.

link-error-disable-entry

Specifies the link error disable entry.

wait-time-in-sec

Specifies the wait time.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

interface/{interface-type}/{interface-name}/link-fault-signaling

Configures, retrieves, and modifies Link Fault Signaling (LFS).

Usage

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
               <ethernet>
                   <name>{req_val}</name>
                   <link-fault-signaling xmlns="urn:brocade.com:mgmt:brocade-lfs">
                       <tx/>
                   </link-fault-signaling>
               </ethernet>
           </interface>
```

Parameters

ethernet

Specifies Ethernet interface.

name

Specifies the interface name.

tx

Specifies the tx direction.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

interface/{interface-type}/{interface-name}/ra-dns-server/hoplimit

Configures the number of hops to be advertised in IPv6 neighbor discovery router advertisement (RA) messages.

Usage

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <ethernet>
                    <name>{req_val}</name>
                    <ipv6>
                        <ipv6-nd-ra xmlns="urn:brocade.com:mgmt:brocade-ipv6-nd-ra">
                            <ipv6-intf-cmds>
                                <nd>
                                    <hoplimit/>
                                </nd>
                            </ipv6-intf-cmds>
                        </ipv6-nd-ra>
                    </ipv6>
                </ethernet>
            </interface>
```

Parameters

ethernet

Specifies Ethernet interface.

name

Specifies the interface name.

ipv6

Specifies IPv6 address.

number

Specifies the number of hops to be advertised.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

interface/{interface-type}/{interface-name}/ra-dns-server/mtu

Sets the size of the maximum transmission unit (MTU) that is advertised in Neighbor Discovery Router Advertisement (RA) messages.

Usage

```
<routing-system xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>{req_val}</name>
                        <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <ipv6-nd-ra xmlns="urn:brocade.com:mgmt:brocade-ipv6-nd-ra">
                                <ipv6-intf-cmds>
                                    <nd>
                                        <mtu/>
                                    </nd>
                                </ipv6-intf-cmds>
                            </ipv6-nd-ra>
                        </ipv6>
                    </ve>
                </interface>
            </routing-system>
```

Parameters

ve

Specifies virtual Ethernet interface.

name

Specifies the interface name.

ipv6

Specifies IPv6 address.

mtu *number*

Specifies the size, in bytes, of the MTU that is advertised.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

interface/{interface-type}/{interface-name}/ra-dns-server/other-config-flag

In IPv6 Neighbor Discovery, indicates to hosts on a local link that they can use the stateful autoconfiguration feature to obtain configuration settings other than IPv6 address information for their interfaces.

Usage

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>{req_val}</name>
                        <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <ipv6-nd-ra xmlns="urn:brocade.com:mgmt:brocade-ipv6-nd-ra">
                                <ipv6-intf-cmds>
                                    <nd>
                                        <other-config-flag/>
                                    </nd>
                                </ipv6-intf-cmds>
                            </ipv6-nd-ra>
                        </ipv6>
                    </ve>
                </interface>
```

Parameters

ve

Specifies virtual Ethernet interface.

name

Specifies the interface name.

ipv6

Specifies IPv6 address.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

ip/dhcp/relay/servers

Configures DHCP relay servers.

Usage

```
<ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
    <interface-ve-dhcp-conf xmlns="urn:brocade.com:mgmt:brocade-dhcp">
        <dhcp>
            <relay>
                <servers>
                    <relay-ip-addr>{req_val}</relay-ip-addr>
                    <server-vrf-name>{req_val}</server-vrf-name>
                </servers>
            </relay>
        </dhcp>
    </interface-ve-dhcp-conf>
</ip>
```

Parameters

relay-ip-addr

Specifies the IP address of the relay server.

server-vrf-name

Specifies the VRF name of the server.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

node

Configures, modifies, or retrieves a node.

Usage

```
<node-id xmlns="urn:brocade.com:mgmt:brocade-node">
    <node-id>{req_val}</node-id>
</node-id>
```

Parameters

node-id

Specifies the node ID.

History

Release version	History
17s.1.01	This NETCONF call was introduced.

overlay-policy

Configures, modifies, or retrieves an overlay-policy.

Usage

```
<overlay-policy-map xmlns="urn:brocade.com:mgmt:brocade-overlay-policy">
    <pmap-name>{req_val}</pmap-name>
    <pmap-seq>
        <pmap-seq-num>{req_val}</pmap-seq-num>
        <overlay-class>{req_val}</overlay-class>
        <ip-acl/>
    </pmap-seq>
</overlay-policy-map>
```

Parameters

cmap-name
: Spcifies the class map name.

pmap-name
: Specifies the policy map name

cmap-seq-num
: Spcifies the class map sequence number.

pmap-seq-num
: Spcifies the policy map sequence number.

ipv4/*access-group*
: Spcifies the IPV4 access group

ipv6/*access-group*
: Spcifies the IPV6 access group

mac/*access-group*
: Spcifies the MAC access group

transit-name
: Specifies the transit name.

overlay-sp-pmap-name
: Spcifies the policy map name of the overlay service policy .

History

Release version	History
17s.1.01	This NETCONF call was introduced.

overlay-policy/overlay-class-map

Configures, modifies, or retrieves a class-map policy name.

Usage

```
<overlay-class-map xmlns="urn:brocade.com:mgmt:brocade-overlay-policy">
    <cmap-name>{req_val}</cmap-name>
    <cmap-seq>
        <cmap-seq-num>{req_val}</cmap-seq-num>
        <match>
            <contn-src-dst/>
        </match>
    </cmap-seq>
</overlay-class-map>
```

Parameters

cmap-name

Spcifies the class map name.

cmap-seq-num

Spcifies the class map sequence number.

match

Spcifies match.

contn-src-dst

Spcifies the source and destination information.

History

Release version	History
17s.1.01	This NETCONF call was introduced.

overlay-policy/overlay-policy-map

Configures, modifies, or retrieves an overlay-policy.

Usage

```
<overlay-policy-map xmlns="urn:brocade.com:mgmt:brocade-overlay-policy">
    <pmap-name>{req_val}</pmap-name>
    <pmap-seq>
        <pmap-seq-num>{req_val}</pmap-seq-num>
        <overlay-class>{req_val}</overlay-class>
        <ip-acl/>
    </pmap-seq>
</overlay-policy-map>
```

Parameters

pmap-name

Specifies the policy map name

pmap-seq-num

Spcifies the policy map sequence number.

overlay-class

Spcifies the overlay class.

ip-acl

Spcifies the IP ACL.

History

Release version	History
17s.1.01	This NETCONF call was introduced.

overlay-policy/overlay-policy-map

Configures, modifies, or retrieves an overlay service policy map.

Usage

```
<overlay-policy-map xmlns="urn:brocade.com:mgmt:brocade-overlay-policy">
    <pmap-name>{req_val}</pmap-name>
    <pmap-seq>
        <pmap-seq-num>{req_val}</pmap-seq-num>
        <overlay-class>{req_val}</overlay-class>
        <ip-acl/>
    </pmap-seq>
</overlay-policy-map>
```

Parameters

pmap-name
Specifies the policy map name

pmap-seq-num
Spcifies the policy map sequence number.

overlay-class
Spcifies the overlay class.

pmap-seq-num
Spcifies the policy map sequence number.

ip-acl
Spcifies the IP ACL.

History

Release version	History
17s.1.01	This NETCONF call was introduced.

snmp

Provides a data model for configuring the Simple Network Management Protocol (SNMP) server.

Usage

```
<snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
    <context>
        <context-name>{req_val}</context-name>
    </context>
</snmp-server>
```

Parameters

context-name

Specifies the context name.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

system-monitor

Configures, modifies, or retrieves FRU threshold and alert setting.

Usage

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <system-monitor xmlns="urn:brocade.com:mgmt:brocade-system-monitor"/>
        </filter>
    </get-config>
</rpc>
```

Parameters

action

Specifies the response type. Supported types are:

all

Specifies that e-mail and RASLog messaging are used.

email

Specifies that an e-mail message is sent.

none

Specifies that no message is sent.

raslog

Specifies RASLog messaging.

state

Specifies the hardware state to be monitored. Supported states are:

all

Specifies that all hardware states are monitored.

faulty

Specifies that hardware is monitored for faults.

inserted

Specifies that the insertion state of hardware is monitored.

none

Specifies that no hardware states are monitored.

on

Specifies that the hardware on/off state is monitored.

removed

Specifies that the removal of hardware is monitored.

down-threshold

Specifies an integer value that, when exceeded, indicates when hardware is down.

marginal-threshold

Specifies an integer value that, when exceeded, indicates when hardware is operating marginally.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

tvf-domain

Configures, modifies, or retrieves a Transparent VLAN Flooding (TVF) domain.

Usage

```
<tvf-domain xmlns="urn:brocade.com:mgmt:brocade-tvf-domain">
    <name>{req_val}</name>
</tvf-domain>
```

Parameters

name

Spcifies the TVF domain name.

History

Release version	History
17s.1.01	This NETCONF call was introduced.

vlan/{vlan-name}/mac

Configures a MAC access group.

Usage

```
<interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <vlan>
                    <name>{req_val}</name>
                    <mac xmlns="urn:brocade.com:mgmt:brocade-mac-access-list">
                        <access-group>
                            <mac-access-list>{req_val}</mac-access-list>
                            <mac-direction>{req_val}</mac-direction>
                            <traffic-type/>
                        </access-group>
                    </mac>
                </vlan>
            </interface-vlan>
```

Parameters

vlan

Specifies vlan.

name

Specifies the VLAN name.

access-group

Specifies the MAC acess- group.

mac-access-list

Specifies the mac-access list.

mac-direction

Specifies the MAC direction.

traffic-type

Specifies the traffic type.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

vrf

Configures the virtual rounding and forwarding (VRF).

Usage

```
<vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
    <vrf-name>{req_val}</vrf-name>
</vrf>
```

Parameters

vrf-name

Specifies the VRF name.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

vrf/{vrf-name}/address-family/ipv4/unicast

Configures the IPv4 address family configurations.

Usage

```
<vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
    <vrf-name>{req_val}</vrf-name>
    <address-family>
        <ip>
            <unicast/>
        </ip>
    </address-family>
</vrf>
```

Parameters

vrf-name
: Specifies the VRF name.

address-family
: Specifies address family.

ip
: Specifies the IP address.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

vrf/{vrf-name}/address-family/ipv4/unicast/max-route

Configures IPv4 address family maximum route.

Usage

```
<vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
    <vrf-name>{req_val}</vrf-name>
    <address-family>
        <ip>
            <unicast>
                <max-route/>
            </unicast>
        </ip>
    </address-family>
</vrf>
```

Parameters

vrf-name
: Specifies the VRF name.

address-family
: Specifies address family.

ip
: Specifies the IP address.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

vrf/{vrf-name}/address-family/ipv6/unicast

Configures the IPv6 address family configurations.

Usage

```
<vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
    <vrf-name>{req_val}</vrf-name>
    <address-family>
        <ipv6>
            <unicast/>
        </ipv6>
    </address-family>
</vrf>
```

Parameters

vrf-name

Specifies the VRF name.

address-family

Specifies address family.

ipv6

Specifies the IP address.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

vrf/{vrf-name}/address-family/ipv6/unicast/max-route

Configures IPv6 address family maximum route.

Usage

```
<vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
    <vrf-name>{req_val}</vrf-name>
    <address-family>
        <ipv6>
            <unicast>
                <max-route/>
            </unicast>
        </ipv6>
    </address-family>
</vrf>
```

Parameters

vrf-name
: Specifies the VRF name.

address-family
: Specifies address family.

ipv6
: Specifies the IP address.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

vrf/{vrf-name}/ip/router-id

Configures the IP route details..

Usage

```
<vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
    <vrf-name>{req_val}</vrf-name>
    <ip>
        <vrf-router-id/>
    </ip>
</vrf>
```

Parameters

vrf-name

Specifies the VRF name.

vrf-router-id

Specifies the VRF router ID..

History

Release version	History
17s.1.00	This NETCONF call was introduced.

bna-config-cmd

Copies configuration data to and from the system.

Usage

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="212">
    <bna-config-cmd xmlns="urn:brocade.com:mgmt:brocade-ras">
        <src>default-config</src>
        <dest>startup-config</dest>
    </bna-config-cmd>
</rpc>

<rpc-reply message-id="212" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <session-id xmlns="urn:brocade.com:mgmt:brocade-ras">5</session-id>
    <status xmlns="urn:brocade.com:mgmt:brocade-ras">in-progress</status>
</rpc-reply>
```

Parameters

session-id

This id is used along with bna-config-cmd-status API to get the status of this operation (inprogress/complete).

status

Displays the status of this operation (inprogress/complete).

History

Release version	History
17s.1.00	This NETCONF call was introduced.

dad-status

Displays the current status of firmware download.

Usage

```
<dad-status></dad-status>
```

```
<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <dad-status xmlns="urn:brocade.com:mgmt:brocade-firmware">
        <dad-status-entries>
            <index>1</index>
            <date-and-time-info>Fri Nov 25 21:01:12 GMT 2016</date-and-time-info>
            <message>DHCP Auto-deployment enabled.</message>
        </dad-status-entries>
    </dad-status>
</rpc-reply>
```

Parameters

index

Displays the Index number

date-and-time-info

Displays the Date and time information

message

Displays the status message

History

Release version	History
17s.1.00	This NETCONF call was introduced.

fwdl-status

Returns the status of the firmware download operation.

Usage

```
<fwdl-status></fwdl-status>
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="71">
   <fwdl-state xmlns="urn:brocade.com:mgmt:brocade-firmware">completed</fwdl-state>
   <number-of-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">24</number-of-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>1</index>
      <blade-name>SW/0</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:20:20</date-and-time-info>
      <message>Firmware install begins.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>2</index>
      <blade-name>SW/0</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:23:25</date-and-time-info>
      <message>Firmware install ends.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>3</index>
      <blade-name>SW/1</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:23:25</date-and-time-info>
      <message>Firmware install begins.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>4</index>
      <blade-name>SW/1</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:26:27</date-and-time-info>
      <message>Firmware install ends.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>5</index>
      <blade-name>SW/0</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:26:28</date-and-time-info>
      <message>Firmware starts to swap.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>6</index>
      <blade-name>SW/1</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:26:28</date-and-time-info>
      <message>Firmware starts to swap.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>7</index>
      <blade-name>SW/1</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:26:34</date-and-time-info>
      <message>Firmware is swapped.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>8</index>
      <blade-name>SW/0</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:26:36</date-and-time-info>
      <message>Firmware is swapped.</message>
   </fwdl-entries>
```

```
<fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
   <index>9</index>
   <blade-name>SW/0</blade-name>
   <message-id>0</message-id>
   <date-and-time-info>2016-11-29/01:26:36</date-and-time-info>
   <message>Firmware is downloaded successfully.</message>
</fwdl-entries>
<fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
   <index>10</index>
   <blade-name>SW/1</blade-name>
   <message-id>0</message-id>
   <date-and-time-info>2016-11-29/01:26:37</date-and-time-info>
   <message>Firmware is downloaded successfully.</message>
</fwdl-entries>
<fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
   <index>11</index>
   <blade-name>SW/1</blade-name>
   <message-id>0</message-id>
   <date-and-time-info>2016-11-29/01:34:17</date-and-time-info>
   <message>The DB/filesystem starts shutting down.</message>
</fwdl-entries>
<fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
   <index>12</index>
   <blade-name>SW/0</blade-name>
   <message-id>0</message-id>
   <date-and-time-info>2016-11-29/01:34:17</date-and-time-info>
   <message>The DB/filesystem starts shutting down.</message>
</fwdl-entries>
<fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
   <index>13</index>
   <blade-name>SW/1</blade-name>
   <message-id>0</message-id>
   <date-and-time-info>2016-11-29/01:34:35</date-and-time-info>
   <message>The DB/filesystem has been shut down.</message>
</fwdl-entries>
<fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
   <index>14</index>
   <blade-name>SW/0</blade-name>
   <message-id>0</message-id>
   <date-and-time-info>2016-11-29/01:34:37</date-and-time-info>
   <message>The DB/filesystem has been shut down.</message>
</fwdl-entries>
<fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
   <index>15</index>
   <blade-name>SW/1</blade-name>
   <message-id>0</message-id>
   <date-and-time-info>2016-11-29/01:34:38</date-and-time-info>
   <message>The blade begins to reboot.</message>
</fwdl-entries>
<fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
   <index>16</index>
   <blade-name>SW/0</blade-name>
   <message-id>0</message-id>
   <date-and-time-info>2016-11-29/01:34:38</date-and-time-info>
   <message>The blade begins to reboot.</message>
</fwdl-entries>
<fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
   <index>17</index>
   <blade-name>SW/1</blade-name>
   <message-id>0</message-id>
   <date-and-time-info>2016-11-29/01:44:23</date-and-time-info>
   <message>The blade is rebooted.</message>
</fwdl-entries>
<fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
   <index>18</index>
   <blade-name>SW/1</blade-name>
   <message-id>0</message-id>
   <date-and-time-info>2016-11-29/01:44:23</date-and-time-info>
   <message>Firmware commit begins.</message>
</fwdl-entries>
<fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
   <index>19</index>
```

```
      <blade-name>SW/0</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:44:23</date-and-time-info>
      <message>The blade is rebooted.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>20</index>
      <blade-name>SW/0</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:44:23</date-and-time-info>
      <message>Firmware commit begins.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>21</index>
      <blade-name>SW/0</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:48:42</date-and-time-info>
      <message>Firmware commit ends.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>22</index>
      <blade-name>SW/0</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:48:42</date-and-time-info>
      <message>Firmware is downloaded successfully.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>23</index>
      <blade-name>SW/1</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:48:51</date-and-time-info>
      <message>Firmware commit ends.</message>
   </fwdl-entries>
   <fwdl-entries xmlns="urn:brocade.com:mgmt:brocade-firmware">
      <index>24</index>
      <blade-name>SW/1</blade-name>
      <message-id>0</message-id>
      <date-and-time-info>2016-11-29/01:48:51</date-and-time-info>
      <message>Firmware is downloaded successfully.</message>
   </fwdl-entries>
</rpc-reply>
```

Parameters

fwdl-state

Specifies the firmware download state.

number-of-entries

Specifies the number of status entries.

index

Specifies the sequence number for the message.

blade-name

Specifies the name of the blade.

message-id

Specifies the message indentifier.

date-and-time-info

Specifies the date and time of the message. The format is YYYY-MM-DD/HH:MM:SS.SSSS.

message

Displays the textual description of the status.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-arp

Retrieves the ARP cache information.

Usage

```
<get-arp xmlns="urn:brocade.com:mgmt:brocade-arp"></get-arp>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <arp-entry>
        <ip-address>20.0.0.122</ip-address>
        <mac-address>0005.3379.407a</mac-address>
        <interface-type>unknown</interface-type>
        <interface-name></interface-name>
        <is-resolved>true</is-resolved>
        <age>03:16:05</age>
        <entry-type>dynamic</entry-type>
    </arp-entry>
</rpc-reply>
```

Parameters

ip-address
: Displays the IP address of the ARP entry.

mac-address
: Displays the MAC address of the ARP entry.

interface-type
: Displays the interface type.

interface-name
: Displays the interface name.

is-resolved
: Indicates whether the ARP entry is resolved or not.

age
: Displays the age of the ARP entry.

entry-type
: Displays the type of the ARP entry.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-contained-in-ID

Retrieves enclosure related information on embedded platforms.

Usage

```
<get-contained-in-ID xmlns="urn:brocade.com:mgmt:brocade-entity"></get-contained-in-ID>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <contained-in-ID>Bay 7</contained-in-ID>
</rpc-reply>
```

Parameters

contained-in-ID

Displays present slot ID of switch.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-interface-detail

Returns operational details of all the possible interfaces of the managed entity. Use this RPC to discover basic characteristics of all the interfaces in the system. Each sublayer below the internetwork layer of a network interface is considered to be an interface.

Usage

```
<get-interface-detail></get-interface-detail>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="67">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type>ethernet</interface-type>
        <interface-name>2/4</interface-name>
        <port-role>edge</port-role>
        <port-mode>unknown</port-mode>
        <if-name>ethernet 2/4</if-name>
        <if-state>up</if-state>
        <line-protocol-state>down</line-protocol-state>
        <line-protocol-state-info> (link protocol down)</line-protocol-state-info>
        <hardware-type>ethernet</hardware-type>
        <current-hardware-address>50:eb:1a:17:40:28</current-hardware-address>
        <logical-hardware-address>50:eb:1a:17:40:28</logical-hardware-address>
        <ifindex>8791662784</ifindex>
        <mtu>2500</mtu>
        <actual-line-speed>nil</actual-line-speed>
        <configured-line-speed>auto</configured-line-speed>
        <line-duplex-state>full</line-duplex-state>
        <flow-control></flow-control>
        <queuing-strategy>fifo</queuing-strategy>
        <ifHCInOctets>0</ifHCInOctets>
        <ifHCInUcastPkts>0</ifHCInUcastPkts>
        <ifHCInMulticastPkts>0</ifHCInMulticastPkts>
        <ifHCInBroadcastPkts>0</ifHCInBroadcastPkts>
        <ifHCInErrors>0</ifHCInErrors>
        <ifHCOutOctets>0</ifHCOutOctets>
        <ifHCOutUcastPkts>0</ifHCOutUcastPkts>
        <ifHCOutMulticastPkts>0</ifHCOutMulticastPkts>
        <ifHCOutBroadcastPkts>0</ifHCOutBroadcastPkts>
        <ifHCOutErrors>0</ifHCOutErrors>
    </interface>
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type>ethernet</interface-type>
        <interface-name>2/3</interface-name>
        <port-role>isl</port-role>
        <port-mode>unknown</port-mode>
        <if-name>ethernet 2/3</if-name>
        <if-state>up</if-state>
        <line-protocol-state>up</line-protocol-state>
        <line-protocol-state-info> (connected)</line-protocol-state-info>
        <hardware-type>ethernet</hardware-type>
        <current-hardware-address>50:eb:1a:17:3f:f8</current-hardware-address>
        <logical-hardware-address>50:eb:1a:17:3f:f8</logical-hardware-address>
        <media-type>sfp</media-type>
        <wavelength>1310</wavelength>
        <ifindex>8791269376</ifindex>
        <mtu>9216</mtu>
        <actual-line-speed>10Gbps</actual-line-speed>
        <configured-line-speed>auto</configured-line-speed>
        <line-duplex-state>full</line-duplex-state>
        <flow-control></flow-control>
        <queuing-strategy>fifo</queuing-strategy>
        <ifHCInOctets>303455437</ifHCInOctets>
        <ifHCInUcastPkts>301429</ifHCInUcastPkts>
        <ifHCInMulticastPkts>79743</ifHCInMulticastPkts>
        <ifHCInBroadcastPkts>0</ifHCInBroadcastPkts>
```

```
            <ifHCInErrors>0</ifHCInErrors>
            <ifHCOutOctets>300765428</ifHCOutOctets>
            <ifHCOutUcastPkts>301347</ifHCOutUcastPkts>
            <ifHCOutMulticastPkts>56906</ifHCOutMulticastPkts>
            <ifHCOutBroadcastPkts>0</ifHCOutBroadcastPkts>
            <ifHCOutErrors>0</ifHCOutErrors>
        </interface>
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
            <interface-type>ethernet</interface-type>
            <interface-name>2/4</interface-name>
            <port-role>edge</port-role>
            <port-mode>unknown</port-mode>
            <if-name>ethernet 2/4</if-name>
            <if-state>up</if-state>
            <line-protocol-state>down</line-protocol-state>
            <line-protocol-state-info> (link protocol down)</line-protocol-state-info>
            <hardware-type>ethernet</hardware-type>
            <current-hardware-address>50:eb:1a:17:40:1d</current-hardware-address>
            <logical-hardware-address>50:eb:1a:17:40:1d</logical-hardware-address>
            <ifindex>8791572480</ifindex>
            <mtu>2500</mtu>
            <actual-line-speed>nil</actual-line-speed>
            <configured-line-speed>auto</configured-line-speed>
            <line-duplex-state>full</line-duplex-state>
            <flow-control></flow-control>
            <queuing-strategy>fifo</queuing-strategy>
            <ifHCInOctets>0</ifHCInOctets>
            <ifHCInUcastPkts>0</ifHCInUcastPkts>
            <ifHCInMulticastPkts>0</ifHCInMulticastPkts>
            <ifHCInBroadcastPkts>0</ifHCInBroadcastPkts>
            <ifHCInErrors>0</ifHCInErrors>
            <ifHCOutOctets>0</ifHCOutOctets>
            <ifHCOutUcastPkts>0</ifHCOutUcastPkts>
            <ifHCOutMulticastPkts>0</ifHCOutMulticastPkts>
            <ifHCOutBroadcastPkts>0</ifHCOutBroadcastPkts>
            <ifHCOutErrors>0</ifHCOutErrors>
        </interface>
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
            <interface-type>l2vlan</interface-type>
            <interface-name>1</interface-name>
            <if-name>Vlan 1</if-name>
            <ifindex>1207959553</ifindex>
            <queuing-strategy>fifo</queuing-strategy>
        </interface>
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
            <interface-type>l2vlan</interface-type>
            <interface-name>10</interface-name>
            <if-name>Vlan 10</if-name>
            <current-hardware-address>00:27:f8:fd:00:03</current-hardware-address>
            <logical-hardware-address>00:27:f8:fd:00:03</logical-hardware-address>
            <ifindex>1207959562</ifindex>
            <queuing-strategy>fifo</queuing-strategy>
        </interface>
       <has-more xmlns="urn:brocade.com:mgmt:brocade-interface-ext">false</has-more>
</rpc-reply>
```

Parameters

interface-type

Specifies the interface type.

interface-name

Specifies the nterface name.

port-role

Displays the current role that the particular interface is playing. This is applicable only for physical interfaces.

port-mode
: Displays the operational mode of the particular interface. This is applicable only for physical interfaces or port-channel interfaces.

if-name
: Displays the interface display name as in MIB-II's ifTable. However interface-name and interface-type values of this instance forms fully qualified name for this interface.

if-state
: Displays the current operational state of this interface.

line-protocol-state
: Displays the 'Line protocol' state of the interface.

line-protocol-state-info
: Displays thwe reason for the current line protocol state of the interface.

hardware-type
: Dispalys the hardware type

current-hardware-address
: Displays the address of the interface at its protocol sub-layer.

logical-hardware-address
: Displays the address of the interface at its protocol sub-layer.

ifindex
: Displays a unique value, greater than zero, for each interface.

mtu
: Displays the IP MTU value of the interface.

actual-line-speed
: Displays the actual line speed of this interface.

configured-line-speed
: Displays the administratively configured line speed of the interface.

line-duplex-state
: Displays the 'Line duplex state' of the interface.

flow-control
: Displays the 'Flow control' for the interface.

queuing-strategy
: Displays the 'Queuing strategy' for the interface.

ifHCInOctets
: Displays the total number of octets received on the interface, including framing characters.

ifHCInUcastPkts
: Displays the The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were not addressed to a multicast or broadcast address at this sub-layer.

ifHCInMulticastPkts
: Displays The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a multicast address at the sub-layer. For a MAC layer protocol, this includes both Group and Functional addresses.

ifHCInBroadcastPkts
: Displays the The number of packets, delivered by the sub-layer to a higher (sub-)layer, which were addressed to a broadcast address at the sub-layer.

ifHCInErrors
: For packet-oriented interfaces, the number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol. For character-oriented or fixed-length interfaces, the number of inbound transmission units that contained errors preventing them from being deliverable to a higher-layer protocol.

ifHCOutOctets
: Displays the total number of octets transmitted out of the interface, including framing characters

ifHCOutUcastPkts
: Displays the total number of packets that higher-level protocols requested be transmitted, and which were not addressed to a multicast or broadcast address at the sub-layer, including those that were discarded or not sent.

ifHCOutMulticastPkts
: Dispalys the total number of packets that higher-level protocols requested be transmitted, and which were addressed to a multicast address at this sub-layer, including those that were discarded or not sent. For a MAC layer protocol, this includes both Group and Functional addresses.

ifHCOutBroadcastPkts
: Displays the total number of packets that higher-level protocols requested be transmitted, and which were addressed to a broadcast address at this sub-layer, including those that were discarded or not sent.

ifHCOutErrors
: For packet-oriented interfaces, the number of outbound packets that could not be transmitted because of errors. For character-oriented or fixed-length interfaces, the number of outbound transmission units that could not be transmitted because of errors.

ip-mtu
: Displays the IP MTU value of this interface.

line-protocol-exception-info
: Displays the 'Exception information' of line protocol.

media-type
: Displays the media type.

wavelength
: Displays the wavelength of pluggable media.

if-description
: Displays the textual string containing information about the interface.

queuing-strategy
: Displays the 'Queuing strategy' for this interface.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-interface-switchport

Returns switch-port or Layer 2 characteristics of all the interfaces in the managed device.

Usage

```
<get-interface-switchport></get-interface-switchport>


<rpc-reply message-id="303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <switchport xmlns="urn:brocade.com:mgmt:brocade-brocade-interface-ext">
        <interface-name>2/1</interface-name>
        <interface-type>ethernet</interface-type>
        <mode>access</mode>
        <ingress-filter-enabled>true</ingress-filter-enabled>
        <acceptable-frame-type>admit-all</acceptable-frame-type>
        <default-vlan>1</default-vlan>
        <active-vlans>
            <vlanid>1</vlanid>
        </active-vlans>
    </switchport>
</rpc-reply>
```

Parameters

interface-name
: Specifies the interface value.

interface-type
: Displays the type of the interface.

mode
: Displays the mode of the port-channel.

ingress-filter-enabled
: Indicates if the 'Ingress filtering' is enabled for the interface.

acceptable-frame-type
: The switch-port ingress Frame admission policy - whether only tagged Frames are allowed or all.

default-vlan
: Displays 'default vlan' identifier value for this switch-port.

vlanid
: Displays the list of active VLAN identifiers.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-ip-interface

Returns brief details of all interfaces, loopback and VE interface details of particular managed entity.

Usage

```
<get-ip-interface></get-ip-interface>

<rpc-reply message-id="307" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <interface-type>ethernet</interface-type>
        <interface-name>2/4</interface-name>
        <if-name>ethernet 2/4</if-name>
        <if-state>up</if-state>
        <line-protocol-state>down</line-protocol-state>
        <ip-address>
            <ipv4>unassigned</ipv4>
        </ip-address>
    </interface>
</rpc-reply>
```

Parameters

interface-type

Displays the network interface name in a VCS environment in the format: slot/port .

interface-name

Displays the interface value.

if-name

The interface display name as in MIB-II's ifTable. However interface-name and interface-type values of this instance forms fully qualified name for this interface.

if-state

Displays the current operational state of the interface.

line-protocol-state

Displays the 'Line protocol' state of the interface.

ipv4

Displays the IP address in dotted decimal/Mask (A.B.C.D/M).

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-last-config-update-time

Returns the time stamp of the last configuration change done on the managed device.

Usage

```
<get-last-config-update-time></get-last-config-update-time>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="64">
    <last-config-update-time xmlns="urn:brocade.com:mgmt:brocade-vcs">1401804078</last-config-update-
time>
</rpc-reply>
```

Parameters

last-config-update-time

Displays the time stamp of the last configuration change.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-lldp-neighbor-detail

Returns the details of all the neighboring interfaces of the managed entity.

Usage

```
<get-lldp-neighbor-detail></get-lldp-neighbor-detail>


<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="4">
    <lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
        <local-interface-name>ethernet 1/3</local-interface-name>
        <local-interface-ifindex>203448320</local-interface-ifindex>
        <local-interface-mac>0005.3379.6de0</local-interface-mac>
        <remote-interface-name>port1</remote-interface-name>
        <remote-interface-mac>0005.3348.3043</remote-interface-mac>
        <dead-interval>120</dead-interval>
        <remaining-life>114</remaining-life>
        <remote-chassis-id>0005.3348.3043</remote-chassis-id>
        <lldp-pdu-transmitted>16159</lldp-pdu-transmitted>
        <lldp-pdu-received>15846</lldp-pdu-received>
    </lldp-neighbor-detail>
    <has-more xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">false</has-more>
</rpc-reply>
```

Parameters

local-interface-name
: Indicates the local interface display name.

local-interface-ifindex
: Indicates the local interface index.

local-interface-mac
: Indicates the local interface MAC address.

remote-interface-name
: Indicates the remote interface display name .

remote-interface-mac
: Indicates the remote interface MAC address.

dead-interval
: Indicates the dead interval.

remaining-life
: Indicates the remaining life period.

remote-chassis-id
: Indicates the remote chassis ID.

lldp-pdu-transmitted
: Number of LLDP PDUs transmitted from the interface.

lldp-pdu-received
: Number of LLDP PDUs received by the interface.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-mac-acl-for-intf

Returns information about the MAC ACL applied on the specified interfaces.

Usage

```
<get-mac-acl-for-intf></get-mac-acl-for-intf>


<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2407">
    <get-mac-acl-for-intf xmlns="urn:brocade.com:mgmt:brocade-mac-access-list">
        <interface>
            <interface-type>l2vlan</interface-type>
            <interface-name>50</interface-name>
            <ingress-policy>
                <policy-name>test_02</policy-name>
            </ingress-policy>
            <egress-policy>
                <policy-name>test_01</egress-policy>
            </egress-policy>
        </interface>
    </get-mac-acl-for-intf>
</rpc-reply>
```

Parameters

interface-type

Displays the interface type.

interface-name

Displays the interface name.

policy-name

Displays the MAC ACL policy name.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-mac-address-table

Returns the MAC address table for a given MAC address.

Usage

```
<get-mac-address-table> <forwarding-interface> <interface-type>ethernet</interface-type>
<interface-name>2/4</interface-name> </forwarding-interface> <mac-type>static</mac-type>
</get-mac-address-table>


<rpc-reply xmlns=""urn:ietf:params:xml:ns:netconf:base:
1.0""xmlns:nc=""urn:ietf:params:xml:ns:netconf:base:1.0"" message-id=""2"">
       <mac-address-table xmlns=""urn:brocade.com:mgmt:brocade-mac-address-table"">
       <vlanid>10</vlanid>
       <mac-address>00:11:11:11:44:44</mac-address>
       <mac-type>static</mac-type>
       <mac-state>active</mac-state>
       <forwarding-interface>
       <interface-type>ethernet</interface-type>
       <interface-name>2/4</interface-name>
       </forwarding-interface>
      </mac-address-table>
      <has-more xmlns=""urn:brocade.com:mgmt:brocade-mac-address-table"">false</has-more>
      </rpc-reply>

<get-mac-address-table>
       <last-mac-address-details>
       <last-mac-address>00:11:11:82:12:92</last-mac-address>
       <last-vlan-id>10</last-vlan-id>
       <last-mac-type>static</last-mac-type>
       </last-mac-address-details>
       <forwarding-interface-type>ethernet</forwarding-interface-type>
       <forwarding-interface-name>2/4</forwarding-interface-name>
     <mac-address-type>static</mac-address-type>
    </get-mac-address-table>

<rpc-reply xmlns=""urn:ietf:params:xml:ns:netconf:base:
1.0""xmlns:nc=""urn:ietf:params:xml:ns:netconf:base:1.0"" message-id=""2"">
       <mac-address-table xmlns=""urn:brocade.com:mgmt:brocade-mac-address-table"">
       <vlanid>10</vlanid>
       <mac-address>00:11:11:82:12:92</mac-address>
       <mac-type>static</mac-type>
       <mac-state>active</mac-state>
       <forwarding-interface>
       <interface-type>ethernet</interface-type>
       <interface-name>2/4</interface-name>
       </forwarding-interface>
      </mac-address-table>
      <has-more xmlns=""urn:brocade.com:mgmt:brocade-mac-address-table"">false</has-more>
      </rpc-reply>
```

Parameters

vlanid

Displays the VLAN ID.

mac-address

Displays the MAC address.

mac-type

Displays the MAC type.

mac-state
Displays the MAC state.

interface-type
Displays the interface type.

interface-name
Displays the interface name.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-media-detail

Returns the media properties of all the interfaces of the managed entity.

Usage

```
<get-media-detail></get-media-detail>


<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="59">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type>ethernet</interface-type>
        <interface-name>2/5</interface-name>
        <qsfpp>
            <speed>40Gbps</speed>
            <connector>lc</connector>
            <encoding>ieee-802-3ab</encoding>
            <vendor-name>BROCADE         </vendor-name>
            <vendor-oui>00:05:1e</vendor-oui>
            <vendor-pn>57-1000263-01   </vendor-pn>
            <vendor-rev>A </vendor-rev>
            <distance>long-dist</distance>
            <media-form-factor>unknown</media-form-factor>
            <wavelength>26020</wavelength>
            <serial-no>LDF113390001CBS </serial-no>
            <date-code>130928  </date-code>
            <temperature>31</temperature>
            <voltage>3305.7</voltage>
            <current>37.364</current>
            <tx-power>0.0</tx-power>
            <rx-power>2.7</rx-power>
        </qsfpp>
    </interface>
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type>ethernet</interface-type>
        <interface-name>2/5</interface-name>
        <sfp>
            <speed>10Gbps</speed>
            <connector>lc</connector>
            <encoding>unknown</encoding>
            <vendor-name>BROCADE</vendor-name>
            <vendor-oui>00:05:1e</vendor-oui>
            <vendor-pn>57-0000076-01</vendor-pn>
            <vendor-rev>A</vendor-rev>
            <distance>unknown</distance>
            <media-form-factor>unknown</media-form-factor>
            <wavelength>1310</wavelength>
            <serial-no>ADF21346000071B </serial-no>
            <date-code>131110</date-code>
            <temperature>36</temperature>
            <voltage>3292.0</voltage>
            <current>38.602</current>
            <tx-power>700.5</tx-power>
            <rx-power>741.6</rx-power>
        </sfp>
    </interface>
</rpc-reply>
```

Parameters

interface-type

Displays the interface type.

interface-name
Displays the interface name.

speed
Specifies the speed.

connector
Specifies the connector.

encoding
Displays the type of encoding used to transmit the data on this interface.

vendor-name
Displays the vendor of the interface.

vendor-oui
Displays the vendor IEEE company ID.

vendor-pn
Displays the vendor part number.

vendor-rev
Displays the vendor revision level.

distance
Displays the SFP distance.

media-form-factor
Displays the media form factor.

wavelength
Displays the wavelength of pluggable media.

serial-no
Displays the serial number.

date-code
Displays the vendor's manufactoring date code.

temperature
Displays the module temperature (degrees C).

voltage
Indicates the supply voltage (Volts).

current
Displays the laser diode drive current (milliAmps).

tx-power
Displays the transmitted optical power (microWatts).

rx-power
Displays the received optical power (microWatts).

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-netconf-client-capabilities

Returns the vendor information for all NETCONF clients.

Usage

```
<get-netconf-client-capabilities></get-netconf-client-capabilities>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="57">
    <session xmlns="urn:extreme.com:mgmt:extreme-netconf-ext">
        <session-id>30</session-id>
        <user-name>admin</user-name>
        <vendor>Extreme</vendor>
        <product>SLX Netconf Client</product>
        <version>0.8 beta</version>
        <identity>sgajaraj</identity>
        <af-type>IPV4</af-type>
        <host-ip>172.22.8.111</host-ip>
        <time>2018-06-04T11:00:35+00:00</time>
    </session>
</rpc-reply>
```

Parameters

session-id
: Displays the session ID of the NETCONF client session.

user-name
: Displays the login name of the user for the NETCONF client session.

vendor
: Displays the vendor name of the NETCONF client session.

product
: Displays the product name of the NETCONF client session.

version
: Displays the product version of the NETCONF client session.

identity
: Displays the identity of the NETCONF client session.

af-type
: Specifies the address family type.

host-ip
: Displays IP address of NETCONF client session.

time
: Displays the login time of NETCONF client session.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-port-channel-detail

Returns link aggregation control configuration parameters for all the port-channels in the system.

Usage

```
<get-port-channel-detail></get-port-channel-detail>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1002">
    <get-port-channel-detail xmlns="urn:brocade.com:mgmt:brocade-lag">
        <lacp>
            <aggregator-id>27</aggregator-id>
            <aggregator-type>standard</aggregator-type>
            <isvlag>false</isvlag>
            <aggregator-mode>none</aggregator-mode>
            <admin-key>0027</admin-key>
            <oper-key>0027</oper-key>
            <actor-system-id>00-05-33-6f-18-18</actor-system-id>
            <partner-system-id>00-05-1e-cd-6e-9f</partner-system-id>
            <system-priority>32768</system-priority>
            <partner-oper-priority>32768</partner-oper-priority>
            <rx-link-count>4</rx-link-count>
            <tx-link-count>4</tx-link-count>
            <individual-agg>0</individial-agg>
            <ready-agg>1</ready-agg>
            <partner-oper-key>0027</partner-oper-key>
            <aggr-member>
                <interface-type>ethernet</interface-type>
                <interface-name>231/0/22</interface-name>
                <actor-port>0xE718160201</actor-port>
                <sync>1</sync>
            </aggr-member>
        </lacp>
        <has-more>true</has-more>
    </get-port-channel-detail>
</rpc-reply>
```

Parameters

aggregator-id

Displays the aggregator ID.

aggregator-type

Displays the aggregator type.

isvlag

Specifies if agreegator is VLAG.

aggregator-mode

Displays aggregator mode.

admin-key

Displays the admin key.

oper-key

Displays the opertional key.

actor-system-id

Displays the actor system ID.

partner-system-id
Displays the partner system ID.

system-priority
Displays the system priority.

partner-oper-priority
Displays the partner operational priority.

rx-link-count
Displays the RX link counter.

tx-link-count
Displays the TX link counter.

individual-agg
Displays the Individual aggregator.

ready-agg
Displays the Ready aggregator.

partner-oper-key
Displays the Partner Opertional key.

interface-type
Displays the interface type .

interface-name
Displays the interface name.

actor-port
Displays the actor port number.

sync
Displays the sync information.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-stp-brief-info

Returns Spanning Tree Protocol (STP) information.

Usage

```
<get-stp-brief-info></get-stp-brief-info>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2025">
    <get-stp-brief-info xmlns="urn:brocade.com:mgmt:brocade-xstp-ext">
        <spanning-tree-info>
            <stp-mode>STP</stp-mode>
            <stp>
                <route-bridge>
                    <priority>32768<priority>
                    <bridge-id>22<bridge-id>
                    <hello-time>2<hello-time>
                    <max-age>20</max-age>
                    <forward-delay>15</forward-delay>
                </route-bridge>
                <bridge>
                    <priority>32768</priority>
                            <hello-time>2<hello-time>
                    <max-age>20</max-age>
                    <forward-delay>15</forward-delay>
                    <transmit-hold-count>6</transit-hold-count>
                    <migrate-time>3</migrate-time>
                    <port>
                        <interface-type>ethernet</interface-type>
                        <interface-name>2/5</interface-name>
                        <spanningtree-enabled>true<spanningtree-enabled>
                        (output truncated)
        </spanning-tree-info>
        <has-more>true</has-more>
        <last-instance>
            <instance-id>91</instance-id>
        </last-instance>
    </get-stp-brief-info>
</rpc-reply>
```

Parameters

stp-mode

Displays the type of the Spanning Tree Protocol configured on the switch.

priority

Displays the Bridge priority.

hello-time

Displays the interval between two transmissions of BPDU packets sent by the Root Bridge to tell all other switches that it is indeed the Root Bridge (1 to 10 sec).

max-age

The Max Age may be set to ensure that old information does not endlessly circulate through redundant paths in the network, preventing the effective propagation of new information (6 to 40 sec).

forward-delay

Port on the Switch spends this time in the listening state while moving from the blocking state to the forwarding state (4 to 30 sec).

transmit-hold-count
: Displays the transmin hold count.

migrate-time
: Displays the migrate time.

interface-type
: Displays the interface type.

interface-name
: Displays the interface name.

spanningtree-enabled
: Displays if the spnanning tree is enabled.

instance-id
: Specifies the instance ID.

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-system-uptime

Returns the time since the managed entity was last reinitialized.

Usage

```
<get-system-uptime></get-system-uptime>


<rpc-reply message-id="307" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-system-uptime xmlns="urn:brocade.com:mgmt:brocade-system">
        <days>0</days>
        <hours>5</hours>
        <minutes>53</minutes>
        <seconds>4</seconds>
    </show-system-uptime>
</rpc-reply>
```

Parameters

days

Displays the number of days the managed node is up since its last re-initialization

hours

Displays the number of hours the managed node is up since its last re-initialization

minutes

Displays the number of minutes the managed node is up since its last re-initialization

seconds

Displays the number of seconds the managed node is up since its last re-initialization

History

Release version	History
17s.1.00	This NETCONF call was introduced.

get-vlan-brief

Returns operational data for a given VLAN and enumeration of all the interfaces belonging to this VLAN.

Usage

```
<get-vlan-brief></get-vlan-brief>


<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="47">
    <configured-vlans-count xmlns="urn:brocade.com:mgmt:brocade-interface-ext">13</configured-vlans-
count>
    <provisioned-vlans-count xmlns="urn:brocade.com:mgmt:brocade-interface-ext">13</provisioned-vlans-
count>
    <unprovisioned-vlans-count xmlns="urn:brocade.com:mgmt:brocade-interface-ext">0</unprovisioned-
vlans-count>
    <vlan xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <vlan-id>1</vlan-id>
        <vlan-type>static</vlan-type>
        <vlan-name>default</vlan-name>
        <vlan-state>members-down</vlan-state>
        <interface>
            <interface-type>unknown</interface-type>
            <interface-name></interface-name>
            <tag>tagged</tag>
            <classification>
                <classification-type>vni</classification-type>
                <classification-value>2</classification-value>
            </classification>
        </interface>
    </vlan>
    <last-vlan-id xmlns="urn:brocade.com:mgmt:brocade-interface-ext">200</last-vlan-id>
    <has-more xmlns="urn:brocade.com:mgmt:brocade-interface-ext">true</has-more>
</rpc-reply>
```

Parameters

configured-vlans-count
: Displays the number of VLANs configured.

provisioned-vlans-count
: Displays the number of VLANs provisioned.

unprovisioned-vlans-count
: Displays the number of VLANs unprovisioned.

vlan-id
: Displays the VLAN ID

vlan-type
: Displays the VLAN type

vlan-name
: Displays the administrative name of the VLAN

vlan-state
: Displays the operational state of the VLAN

interface-type
: Displays the interface type

interface-name
Displays the interface name

tag
Displays the state of the interface - untagged, tagged, or converged

classification-type
Displays the type of classification

classification-value
Displays the value of the VLAN classification

last-vlan-id
Displays the last VLAN record that has been fetched

History

Release version	History
17s.1.00	This NETCONF call was introduced.

show-clock

Returns the date, time, and time zone.

Usage

```
<show-clock></show-clock>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="7">
    <clock-time xmlns="urn:brocade.com:mgmt:brocade-clock">
        <current-time>2014-06-04T11:03:31+00:00</current-time>
        <timezone>Etc/GMT</timezone>
    </clock-time>
</rpc-reply>
```

Parameters

current-time

Displays the switch date and time

timezone

Displays the region/city or region/state/city

History

Release version	History
17s.1.00	This NETCONF call was introduced.

show-ntp

Returns the active NTP server for the Extreme VCS fabric or specified switch.

Usage

```
<show-ntp></show-ntp>


<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="12">
    <node-active-server xmlns="urn:extreme.com:mgmt:brocade-ntp">
            <LOCL>true</LOCL>
    </node-active-server>
</rpc-reply>
```

Parameters

LOCL

Indicates whether the LOCL is true or false

History

Release version	History
17s.1.00	This NETCONF call was introduced.

show-raslog

Returns RAS Log entries.

Usage

```
<show-raslog></show-raslog>


<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <show-all-raslog xmlns="urn:brocade.com:mgmt:brocade-ras-ext">
                <number-of-entries>2842</number-of-entries>
        <raslog-entries>
            <index>8288</index>
            <message-id>NSM-1019</message-id>
            <date-and-time-info>2013/12/07-03:21:17:69</date-and-time-info>
            <severity>informational</severity>
            <log-type>dce</log-type>
            <repeat-count>1</repeat-count>
            <message> Interface Ve 4093 is administratively up.</message>
            <message-flag>unknown</message-flag>
            <switch-or-chassis-name>sw0</switch-or-chassis-name>
        </raslog-entries>
        <raslog-entries>
            <index>13584</index>
            <message-id>SEC-3022</message-id>
            <date-and-time-info>2014/06/03-14:03:52:25</date-and-time-info>
            <severity>informational</severity>
            <log-type>system</log-type>
            <repeat-count>1</repeat-count>
            <message>Event: logout, Status: success, Info: Successful logout by user [admin].</message>
            <message-flag>unknown</message-flag>
            <switch-or-chassis-name>sw0</switch-or-chassis-name>
        </raslog-entries>
    </show-all-raslog>
</rpc-reply>
```

Parameters

number-of-entries

Displays the number of recent events to be fetched from the RASLOG entries

index

Displays the sequence number for the message

message-id

Displays the message identifier

date-and-time-info

Displays the date and time of the message. The format is: YYYY-MM-DD/HH:MM:SS.SSSS

severity

Displays the severity of the message. Valid values include: INFO, WARNING, ERROR, and CRITICAL

log-type

Specifies if the message is a SYSTEM or DCE log

repeat-count

Displays the number of times the particular event has occurred

message
: Displays the textual description of the event

message-flag
: Displays the type of the message

switch-or-chassis-name
: Displays the switch name for the generator of the message, or chassis

History

Release version	History
17s.1.00	This NETCONF call was introduced.

show-system-monitor

Returns system status information.

Usage

```
<show-system-monitor></show-system-monitor>


<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="17">
    <switch-status xmlns="urn:brocade.com:mgmt:brocade-system-monitor-ext">
                <switch-name>sw0</switch-name>
        <switch-ip>10.25.224.18</switch-ip>
        <switch-state>state-marginal</switch-state>
        <switch-state-reason>Switch Status is MARGINAL. Contributors: * Power Supply: 1 bad.
(MARGINAL).</switch-state-reason>
        <report-time>2014-06-04T11:10:5711.668484+31:03</report-time>
        <component-status>
            <component-name>Power supplies monitor</component-name>
            <component-state>state-marginal</component-state>
        </component-status>
        <component-status>
            <component-name>Temperatures monitor</component-name>
            <component-state>state-healthy</component-state>
        </component-status>
        <component-status>
            <component-name>Fans monitor</component-name>
            <component-state>state-healthy</component-state>
        </component-status>
    </switch-status>
</rpc-reply>
```

Parameters

switch-name
: Displays the name of the switch

switch-ip
: Displays the IP address of the switch

switch-state
: Displays the switch status based on components

switch-state-reason
: Displays the component reason for switch status

report-time
: Displays the switch report time stamp

component-name
: Displays the component name

component-state
: Displays the component status based on thresholds

History

Release version	History
17s.1.00	This NETCONF call was introduced.

user-session-info

Returns user role information.

Usage

```
<user-session-info></user-session-info>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="20">
    <user-role xmlns="urn:brocade.com:mgmt:brocade-aaa-ext">admin</user-role>
</rpc-reply>
```

Parameters

user-role

Displays the user role.

History

Release version	History
17s.1.00	This NETCONF call was introduced.