53-1003778-01
25 June 2015

Network OS

NETCONF Operations Guide

Supporting Network OS 6.0.1

BROCADE

Brocade Communications Systems, Incorporated

Corporate and Latin American Headquarters
Brocade Communications Systems, Inc.
130 Holger Way
San Jose, CA 95134
Tel: 1-408-333-8000
Fax: 1-408-333-8101
E-mail: info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems China HK, Ltd.
No. 1 Guanghua Road
Chao Yang District
Units 2718 and 2818
Beijing 100020, China
Tel: +8610 6588 8888
Fax: +8610 6588 9999
E-mail: china-info@brocade.com

European Headquarters
Brocade Communications Switzerland Sàrl
Centre Swissair
Tour B - 4ème étage
29, Route de l'Aéroport
Case Postale 105
CH-1215 Genève 15
Switzerland
Tel: +41 22 799 5640
Fax: +41 22 799 5641
E-mail: emea-info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems Co., Ltd. (Shenzhen WFOE)
Citic Plaza
No. 233 Tian He Road North
Unit 1308 – 13th Floor
Guangzhou, China
Tel: +8620 3891 2000
Fax: +8620 3891 2111
E-mail: china-info@brocade.com

Document History

Title	Publication number	Summary of changes	Date
Network OS NETCONF Operations Guide	53-1002565-01	New document	September 2012
Network OS NETCONF Operations Guide	53-1003050-01	Updated for Network OS v4.0.0	October 2013
Network OS NETCONF Operations Guide	53-1003173-01	Updated for Network OS v4.1.0	January 2014
Network OS NETCONF Operations Guide	53-1003231-01	Updated for Network OS v4.1.1	March 2014
Network OS NETCONF Operations Guide	53-1003231-02	Corrections in headers for examples	April 2014
Network OS NETCONF Operations Guide	53-1003320-01	Updated for Network OS v5.0.0	August 2014
Network OS NETCONF Operations Guide	53-1003461-01	Updated for Network OS v5.0.1	September 2014
Network OS NETCONF Operations Guide	53-1003461-02	Updated for Network OS v5.0.1a	November 2014
Network OS NETCONF Operations Guide	53-1003665-01	Updated for Network OS v6.0.0	February 2015
Network OS NETCONF Operations Guide	53-1003778-01	Updated for Network OS v6.0.1	June 2015

Contents (High Level)

Section I Appendixes

Contents (Detailed)

Chapter 40 Configuring 802.1x Port Authentication

Chapter 41 Configuring sFlow

Chapter 42 Configuring Switched Port Analyzer

Chapter 47 Configuring OSPFv3

Chapter 48 Configuring VRRP

Figures

Tables

About This Document

In this chapter

Document conventions

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

Text formatting

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

Format	Description
bold text	Identifies command names Identifies keywords and operands Identifies the names of user-manipulated GUI elements Identifies text to enter at the GUI
italic text	Identifies emphasis Identifies variables Identifies document titles
`Courier font`	Identifies CLI output 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.
value	In Fibre Channel products, a fixed value provided as input to a command option is printed in plain text, for example, –**show** WWN.

Convention	Description
[ ]	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. In Fibre Channel products, square brackets may be used instead for this purpose.
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.

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.

Brocade resources

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

You can download additional publications supporting your product at www.brocade.com. Select the Brocade Products tab to locate your product, then click the Brocade product name or image to open the individual product page. The user manuals are available in the resources module at the bottom of the page under the Documentation category.

To get up-to-the-minute information on Brocade products and resources, go to MyBrocade. You can register at no cost to obtain a user ID and password.

Release notes are available on MyBrocade under Product Downloads.

White papers, online demonstrations, and data sheets are available through the Brocade website.

Contacting Brocade Technical Support

As a Brocade customer, you can contact Brocade Technical Support 24x7 online, by telephone, or by email. Brocade OEM customers contact their OEM/Solutions provider.

Brocade customers

For product support information and the latest information on contacting the Technical Assistance Center, go to http://www.brocade.com/services-support/index.html.

If you have purchased Brocade product support directly from Brocade, use one of the following methods to contact the Brocade Technical Assistance Center 24x7.

Online	Telephone	E-mail
Preferred method of contact for non urgent issues: • My Cases through MyBrocade • Software downloads and licensing tools • Knowledge Base	Required for Sev 1-Critical and Sev 2-High issues: • Continental US: 1-800-752-8061 • Europe, Middle East, Africa, and Asia Pacific: +800-AT FIBREE (+800 28 34 27 33) • For areas unable to access toll free number: +1-408-333-6061 • Toll-free numbers are available in many countries.	support@brocade.com Please include: • Problem summary • Serial number • Installation details • Environment description

Brocade OEM customers

If you have purchased Brocade product support from a Brocade OEM/Solution Provider, contact your OEM/Solution Provider for all of your product support needs.

- OEM/Solution Providers are trained and certified by Brocade to support Brocade® products.
- Brocade provides backline support for issues that cannot be resolved by the OEM/Solution Provider.
- Brocade Supplemental Support augments your existing OEM support contract, providing direct access to Brocade expertise. For more information, contact Brocade or your OEM.
- For questions regarding service levels and response times, contact your OEM/Solution Provider.

Document feedback

To send feedback and report errors in the documentation you can use the feedback form posted with the document or you can e-mail the documentation team.

Quality is our first concern at Brocade 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:

- Through the online feedback form in the HTML documents posted on www.brocade.com.
- By sending your feedback to documentation@brocade.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.

About This Document

In this chapter

How this document is organized

This document is organized to help you find the information that you want as quickly and easily as possible.

Section 1, Network OS Administration contains the following components:

- Chapter 1, "NETCONF Overview" provides an overview of the basic features of NETCONF.
- Chapter 2, "Basic NETCONF Operations" provides instructions for performing basic NETCONF operations such as establishing a session, editing the configuration, and retrieving operational data.
- Chapter 3, "Basic Switch Management" provides procedures for connecting to a switch, setting switch attributes, enabling or disabling a chassis, rebooting, managing slots, modules and ports, configuring the switch banner, uploading SupportSave data, and managing system logs.
- Chapter 4, "Network Time Protocol" provides instructions and examples for using NETCONF operations to configure NTP servers, and to set the date, time, and time zone on a switch.
- Chapter 5, "Installing and Maintaining Firmware" provides preparations and procedures for performing firmware downloads.
- Chapter 6, "Administering Licenses" provides procedures for verifying and activating Brocade licenses.
- Chapter 7, "SNMP" provides procedures for setting community strings and other SNMP configurations.
- Chapter 8, "Fabric" provides procedures for configuring fabric parameters.
- Chapter 9, "Metro VCS" provides procedures for configuring Metro VCS.
- Chapter 10, "Administering Zones" provides procedures for administering fabric-based zoning.
- Chapter 11, "Configuring Fibre Channel Ports" provides procedures for configuring Fibre Channel ports.
- Chapter 12, "Configuring Access Gateway" provides procedures for configuring Access Gateway ports.

- Chapter 13, “System Monitor Configuration” provides procedures monitoring the health of each fan, power supply, temperature sensor, chassis identification (CID) card, small form-factor pluggable (SFP) device, management module (MM), line card, or switch fabric module (SFM), or compact flash of the switch.
- Chapter 14, “VMware vCenter” provides procedures for configuring VMware vCenter.
- Chapter 15, “Configuring Remote Monitoring” provides procedures for configuring Remote Monitoring.
- Chapter 16, “Python Event-Management and Scripting” provides procedures for phython scripting.
- Chapter 17, “Configuring DHCP Dynamic Auto Deployment” provides procedures for configuring DHCP dynamic auto deployment.

Section 2, Network OS Security Configuration, contains the following components:

- Chapter 18, “Managing User Accounts” provides procedures for creating, modifying, and unlocking user accounts, creating and managing user-defined roles, defining role-based command access rules, and managing passwords.
- Chapter 19, “External Server Authentication” provides procedures for configuring an external RADIUS, TACACS+, or LDAP server for remote user authentication.
- Chapter 20, “Fabric Authentication” provides procedures to configure fabric authentication and Switch Connection Control (SCC) policies.

Section 3, Network OS Layer 2 Switch Features, contains the following components:

- Chapter 21, “Administering Edge-Loop Detection” provides procedures for administering edge-loop detection.
- Chapter 22, “Configuring AMPP” provides procedures for configuring the Auto Migrating Port Profile (AMPP) profiles.
- Chapter 23, “Configuring FCoE Interfaces” provides procedures for configuring Fibre Channel over Ethernet (FCoE) interfaces.
- Chapter 24, “Configuring FlexPort” provides procedures for configuring Flexport.
- Chapter 25, “Configuring VLANs” provides procedures for configuring Virtual LANs.
- Chapter 26, “Resolving Repeated MAC-Moves” provides procedures for configuring mac-moves.
- Chapter 27, “Configuring Link-State Tracking (LST)” provides procedures for configuring Link-State Tracking.
- Chapter 28, “Configuring VXLAN Overlay Gateways for NSX Controller Deployments” provides procedures for configuring VXLAN overlay gateways with NSX Controller.
- Chapter 29, “Configuring Distributed VXLAN Gateways” provides procedures for configuring distributed VXLAN gateways.
- Chapter 30, “Configuring Virtual Fabrics” provides procedures for configuring Virtual Fabrics.
- Chapter 31, “Configuring Spanning Tree Protocols” provides procedures for configuring the Spanning Tree Protocol (STP), Rapid STP (RSTP), Multiple STP (MSTP), and Per-VLAN Spanning Tree (PVST).
- Chapter 32, “Configuring UDLD” provides procedures for configuring UDLD.
- Chapter 33, “Configuring Link Aggregation” provides procedures for configuring Link Aggregation and the Link Aggregation Control Protocol (LACP).

- Chapter 34, “Configuring LLDP” provides procedures for configuring the Link Layer Discovery Protocol (LLDP) and the DCB Capability Exchange Protocol (DCBX).
- Chapter 35, “HTTPS Crypto Certificates” provides procedures for configuring HTTPS crypto certificates.
- Chapter 36, “Configuring ACLs” provides procedures for configuring Access Control Lists (ACLs).
- Chapter 37, “Configuring Dynamic ARP Inspection (DAI)” provides procedures for configuring Dynamic ARP Inspection.
- Chapter 38, “Router Advertisement (RA) Guard” provides procedures to enable or disable RA Guards.
- Chapter 39, “Configuring QoS” provides procedures for configuring Quality of Service (QoS), including the Policer feature.
- Chapter 40, “Configuring 802.1x Port Authentication” provides procedures for configuring the 802.1x Port Authentication protocol.
- Chapter 41, “Configuring sFlow” provides procedures for configuring sFlow.
- Chapter 42, “Configuring Switched Port Analyzer” provides procedures for configuring Switched Port Analyzer (SPAN).
- Chapter 43, “Configuring SFP Breakout Mode” provides procedures for configuring breakout mode.

Section 4, Network OS Layer 3 Routing Features, contains the following components:

- Chapter 44, “IP Route Policy” provides procedures for configuring IP prefix lists and route maps that are used for controlling IP subnet transportation between subsystems.
- Chapter 45, “IP Route Management” provides procedures for configuring the route manager to optimize forwarding of IP packets.
- Chapter 46, “Configuring OSPF” provides procedures for configuring Open Shortest Path First (OSPF).
- Chapter 47, “Configuring OSPFv3” provides procedures for configuring Open Shortest Path First Version3 (OSPFv3).
- Chapter 48, “Configuring VRRP” provides procedures for configuring the Virtual Router Redundancy Protocol (VRRP).
- Chapter 49, “Configuring VRRPv3” provides procedures for configuring the Virtual Router Redundancy Protocol Version3 (VRRPv3).
- Chapter 50, “Configuring Bidirectional Forwarding Detection (BFD)” provides procedures for configuring Bidirectional Forwarding Detection.
- Chapter 51, “Configuring Fabric-Virtual-Gateway” provides procedures for configuring Fabric-Virtual-Gateway.
- Chapter 52, “Configuring VRF” provides procedures for configuring remote monitoring (RMON).
- Chapter 53, “Configuring Multi-VRF” provides procedures for configuring Multi-VRF.
- Chapter 54, “Configuring BGP” provides procedures for configuring BGP.
- Chapter 55, “Configuring BGP4+” provides procedures for configuring BGP4+.
- Chapter 56, “Configuring and Managing IPv6 ACLs” provides procedures for configuring IPv6 access control lists (ACLs).
- Chapter 57, “Configuring Protocol-independent multicast (PIM)” provides procedures for configuring PIM.

- Chapter 58, “Configuring Dual Stack Support” provides procedures for configuring the Dual stack support.
- Chapter 59, “Configuring IGMP” provides procedures for configuring IGMP snooping.
- Chapter 60, “Configuring DHCP Relay” provides procedures for configuring DHCP Relay.
- Chapter 61, “Configuring IP DHCPv6 Relay” provides procedures for configuring IP DHCPv6 relay.
- Chapter 62, “Configuring Monitoring and Alerting Policy Suite” provides procedures for configuring MAPS.

Section 5, Appendixes, contains the following component:

- Appendix A, “Managing NETCONF” provides procedures for viewing NETCONF client capabilities and for monitoring NETCONF statistics and session information.

Supported hardware and software

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

Although many different software and hardware configurations are tested and supported by Brocade Communications Systems, Inc. for Network OS v5.0.0, documenting all possible configurations and scenarios is beyond the scope of this document.

The following hardware platforms are supported by this release of Network OS:

- Brocade VDX 2740 Converged Ethernet Switch Module

NOTE
The Brocade VDX 2740 is the equivalent of the Lenovo Flex System EN4023 10Gb Scalable Switch. This platform is identified in the system as "EN4023."

- Brocade VDX 6740
  - Brocade VDX 6740-48
  - Brocade VDX 6740-64
- Brocade VDX 6740T
  - Brocade VDX 6740T-48
  - Brocade VDX 6740T-64
  - Brocade VDX 6740T-1G
- Brocade VDX 6940-36Q
- Brocade VDX 6940-144S
- Brocade VDX 8770
  - Brocade VDX 8770-4
  - Brocade VDX 8770-8

To obtain information about a Network OS version other than this release, refer to the documentation specific to that version.

Using the Network OS CLI

For complete instructions and support for using the Network OS v6.0.1 command line interface (CLI), refer to the *Network OS Command Reference.*

What's new in this document

This document is released in conjunction with Network OS 6.0.1.

New chapters

- Configuring Bidirectional Forwarding Detection (BFD).
- Configuring Dynamic ARP Inspection (DAI).
- Configuring Fabric-Virtual-Gateway.
- Configuring Link-State Tracking (LST).
- Configuring Monitoring and Alerting Policy Suite.
- Configuring Multi-VRF.
- Configuring Distributed VXLAN Gateways.

Modified chapters

- SNMP, modified sections Setting the SNMP version 1 or 2c server host and Setting the SNMP version 3 host to include the parameter *use-vrf*.
- Configuring Access Gateway, added section Enabling Access Gateway mode.
- Configuring FCoE Interfaces, added section Configuring FCoE logical SANs.
- Configuring sFlow, modified section Configuring the sFlow protocol globally to include the parameter *use-vrf*.
- Configuring PIM, added sections Enabling IGMP snooping on access-layer switches, Enabling PIM on aggregation-layer switches and Restricting unknown multicast.

Section

I

Network OS Administration

This section describes basic Network OS administration features, and includes the following chapters:

Chapter

NETCONF Overview 1

In this chapter

NETCONF and YANG

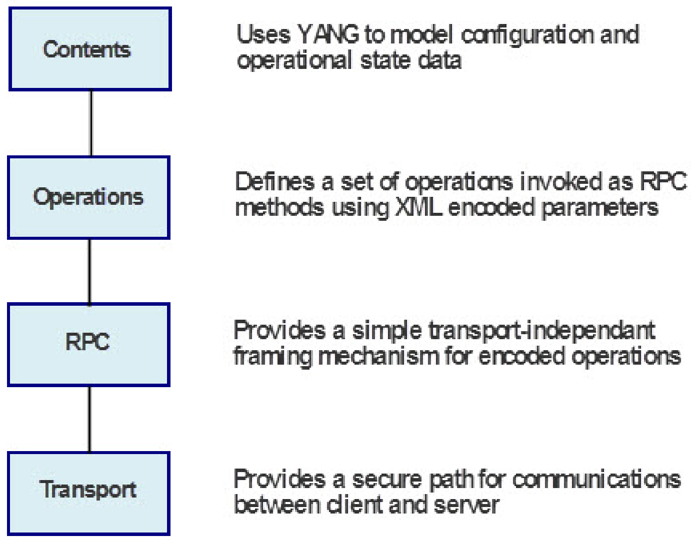
Brocade Network OS provides support for the Network Configuration Protocol (NETCONF) and the YANG data modeling language. Using Extensible Markup Language (XML) constructs, the NETCONF protocol provides the ability to manipulate configuration data and view state data modeled in YANG. NETCONF uses a client/server 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 Figure 1.

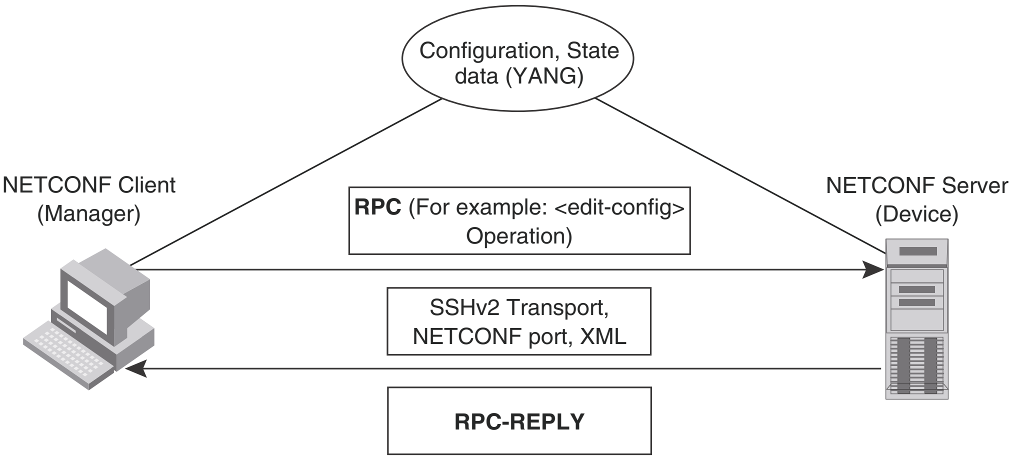
FIGURE 1 **Four layers of NETCONF**



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, as shown in Figure 2.

FIGURE 2 NETCONF communication



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.

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.

Partial success behavior in logical chassis

NETCONF clients should explicitly handle an <rpc-error> in logical chassis mode as shown in the following example. NETCONF clients consider it as a warning, but do not stop operation. The database is committed with the new configuration. The following example is a sample of an <rpc-reply> that is partially successful.

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2">
    <rpc-error>
        <error-type>application</error-type>
            <error-tag>operation-failed</error-tag>
                <error-severity>error</error-severity>
                    <error-message
unknown:lang="en">FRAMEWORK_CLUSTER_PARTIAL_SUCCESS | Warning: Operation context
"/qos:nas/server-ip[server-ip="10.0.0.0/24"]/vrf[vrf-name="Testing_Vrf"]".
Cluster wide operation failed on Rbridge-id(s): 6. Succeeded on Rbridge-id(s): 1.
Rbridge-id(s): 6 Reason: %Error: Command is not supported on this
platform.</error-message>
    </rpc-error>
</rpc-reply>
```

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*"

NETCONF support in Network OS

This section describes the support in Network OS for NETCONF features.

Table 1 describes the degree of support in Network OS for each NETCONF RPC. For details of the RPCs listed in Table 1, refer to RFC 4741.

TABLE 1 NETCONF RPCs supported in Network OS

RPC	Function	Support in Network OS
<copy-config>	Copies the startup configuration to the running configuration, copies the running configuration to the startup configuration, copies the startup or running configuration to a remote file, or copies the remote file to the startup or running configuration.	Use <bna-config-cmd> custom RPC instead.
<close-session>	Terminates the current NETCONF session gracefully.	Supported
<delete-config>	Deletes a configuration datastore.	Supported
<edit-config>	Makes changes to a configuration datastore.	The merge and delete operations are supported. The replace and create operations are not supported. The <running> target is supported. The <candidate> target is not supported. The <error-option> element supports only the *stop-on-error* value. It does not support the *continue-on-error* or *rollback-on-error* values.
<get>	Retrieves running configuration and device state information.	Retrieval of configuration data is supported. Retrieval of operational state data is not supported through the <get> RPC. Operational state data is retrieved using the Brocade Custom RPC and the custom action mechanism. Configuration state data is not modeled in the data models.
<get-config>	Retrieves the entire or partial configuration data.	Supported
<kill-session>	Forces the termination of a NETCONF session.	Supported
<lock>	Locks a configuration datastore.	Not supported
<unlock>	Unlocks a configuration datastore.	Not supported

To retrieve operational state data, Network OS supports two mechanisms: the Brocade Custom RPCs and the custom action mechanism. Refer to Chapter 2, "Basic NETCONF Operations," for details about Brocade customized RPCs and the custom action mechanism.

Chapter

Basic NETCONF Operations 2

In this chapter

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=tailf-
aaa</capability>
        <capability>urn:brocade.com:mgmt:brocade-aaa?revision=2010-10-21&module=br
ocade-aaa</capability>
        <capability>urn:brocade.com:mgmt:brocade-aaa-ext?revision=2010-09-21&modul
e=brocade-aaa-ext</capability>
        <capability>urn:brocade.com:mgmt:brocade-cdp?revision=2010-08-17&module=br
ocade-cdp</capability>
        <capability>urn:brocade.com:mgmt:brocade-cee-map?revision=2011-04-18&modul
e=brocade-cee-map</capability>
        <capability>
urn:brocade.com:mgmt:brocade-chassis?revision=2011-04-11&module=brocade-chassis
         </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 Network 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.
- Brocade proprietary capabilities—A set of capabilities that support Brocade 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 *Network OS YANG Reference Manual*. For element definitions, refer to the YANG file itself.

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

Client capabilities

The client must support the base capability. In addition, Brocade recommends 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=brocade&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 Brocade implementation, the <get> RPC does not return operational state data; Brocade instead provides a set of Custom RPCs and actions for returning operational state data. In the Brocade 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, Brocade 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 Fibre Channel configuration for all Fibre Channel interfaces configured on the switch, use the following filter. This operation returns all configuration data for all Fibre Channel ports on the managed device.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="201" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fc-port/>
            </interface>
        </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 <fc-port> element further constrains the output to the information under the <fc-port> node. Used in this way, <fc-port> is termed a *containment node*.

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

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="202" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
```

```
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fc-port>
                    <name>8/0/1</name>
                </fc-port>
            </interface>
        </filter>
    </get-config>
</rpc>
```

If all you want to know is the setting of one specific Fibre Channel port attribute, such as the configured speed, use a filter such as the following. In this case, <fc-speed-cfg> suppresses the inclusion of all its sibling nodes. It is termed a *selection node*.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="203" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fc-port>
                    <name>8/0/1</name>
                    <fc-speed-cfg/>
                </fc-port>
            </interface>
        </filter>
    </get-config>
</rpc>
```

The following example retrieves the configuration for the Fibre Channel port 1 on routing bridge 8.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="204" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fc-port>
                    <name>8/0/1</name>
                </fc-port>
            </interface>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="204" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <fc-port>
            <name>8/0/1</name>
            <ifindex>1</ifindex>
            <fc-speed-cfg>8gbps</fc-speed-cfg>
            <long-distance>ld</long-distance>
            <vc-link-init>arb</vc-link-init>
            <desire-distance>0</desire-distance>
```

```
            <trunk-enable></trunk-enable>
        </fc-port>
    </interface>
</rpc-reply>
```

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 tengigabitethernet [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 0/59 and 0/60. 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/tengigabitethernet/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">
            <tengigabitethernet>
                <name>0/59</name>
                <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                    <default-cos>0</default-cos>
                    <cos-mutation>test</cos-mutation>
                </qos>
            </tengigabitethernet>
            <tengigabitethernet>
                <name>0/60</name>
                <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                    <default-cos>0</default-cos>
                    <cos-mutation>test</cos-mutation>
                </qos>
```

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

Retrieving operational data

In the Brocade Network OS implementation of NETCONF, two mechanisms are used for retrieving operational data: Brocade 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 Brocade Custom RPCs and actions, and their locations, refer to the *Network OS YANG Reference Manual*.

Brocade Network OS does not support retrieving operational data using the standard <get> RPC.

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-port-profile-for-intf> RPC defined in the brocade-port-profile-ext.yang file.

The brocade-port-profile-ext.yang file defines the structure of the <get-port-profile-for-intf> RPC as follows:

```
+---x get-port-profile-for-intf
      +--ro input
         +--ro interface-type?   enumeration
         +--ro interface-name?   union
      +--ro output
         +--ro interface
            +--ro interface-type?   enumeration
            +--ro interface-name?   union
            +--ro port-profile
               +--ro name?   common-def:name-string64
```

The following example shows the <rpc> message and reply. The <get-port-profile-for-intf> element contains an xmlns attribute that identifies the corresponding namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="206">
    <get-port-profile-for-intf
        xmlns="urn:brocade.com:mgmt:brocade-interface-ext"/>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="206">
    <interface xmlns="urn:brocade.com:mgmt:brocade-port-profile-ext">
        <interface-type>tengigabitethernet</interface-type>
        <interface-name>9/0/53</interface-name>
        <port-profile>
            <name>auto-VM_Network</name>
        </port-profile>
    </interface>
    <interface xmlns="urn:brocade.com:mgmt:brocade-port-profile-ext">
        <interface-type>tengigabitethernet</interface-type>
        <interface-name>9/0/54</interface-name>
        <port-profile>
```

```
            <name>auto-for_iscsi</name>
        </port-profile>
    </interface>
</rpc-reply>
```

Refer to the *Network OS YANG Reference Manual* 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>tengigabitethernet</interface-type>
            <interfce-name>66/0/10</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>tengigabitethernet</interface-type>
            <interfce-name>66/0/12</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 Network OS implementation for retrieving operational information. The following structure is defined in the brocade-zone.yang module for displaying operational data related to zoning.

```
+--rw common-def:show
   +--rw brocade-zone:zoning
         +--action brocade-zone:operation-info
            +--input
            +--output
               +--ro brocade-zone:db-max
               +--ro brocade-zone:db-avail
               +--ro brocade-zone:db-committed
               +--ro brocade-zone:db-transaction
               +--ro brocade-zone:transaction-token
               +--ro brocade-zone:last-zone-changed-timestamp
               +--ro brocade-zone:last-zone-committed-timestamp
```

The following example shows use of the <zoning>/<operation-info> action.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="209">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
```

```
        <nca:data>
            <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <zoning xmlns="urn:brocade.com:mgmt:brocade-zone"/>
            </show>
        </nca:data>
    </nca:action>
</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="209">
    <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
        <db-max>1045274</db-max>
        <db-avail>1043895</db-avail>
        <db-committed>367</db-committed>
        <db-transaction>373</db-transaction>
        <transaction-token>1</transaction-token>
        <last-zone-changed-timestamp>2011-11-16 16:54:31 GMT-7:00
        </last-zone-changed-timestamp>
        <last-zone-committed-timestamp>2011-11-16 16:23:44 GMT-7:0
        </last-zone-committed-timestamp>
    </zoning>
</rpc-reply>
```

For a list of available actions and their locations, refer to the *Network 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 Brocade 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:brocade.com:mgmt:brocade-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:brocade.com:mgmt:brocade-mac-address-table">
                <static>
                    <mac-address>0011.2222.3333</mac-address>
                    <forward>forward</forward>
                    <interface-type>tengigabitethernet</interface-type>
                    <interface-name>66/0/1</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:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-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

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

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

Some simple examples are provided here. Refer to the *Network 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 Brocade VCS Fabric
- How to perform management configuration using the Brocade Network OS command line interface (CLI)

The most common configuration management operation is to copy the *running-config* file to the *startup-config* file. You must perform this operation to save configuration changes across reboots. To copy the running-config file to the startup-config file, issue the following RPC.

```
<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>running-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>
```

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:brocade.com:mgmt:brocade-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:brocade.com:mgmt:brocade-ras">completed</status>
</rpc-reply>
```

To archive or back up the *running-config* or *startup-config* file, specify <running/> or <startup/> 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:brocade.com:mgmt:brocade-ras">
      <src>running-config</src>
      <dest>https://user@brocade.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:brocade.com:mgmt:brocade-ras">6</session-id>
    <status xmlns="urn:brocade.com:mgmt:brocade-ras">in-progress</status>
</rpc-reply>
```

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

```
<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>https://user@brocade.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:brocade.com:mgmt:brocade-ras">6</session-id>
    <status xmlns="urn:brocade.com:mgmt:brocade-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.

Chapter

Basic Switch Management 3

In this chapter

Basic switch management with NETCONF overview

This chapter provides procedures for performing some basic switch operations using the NETCONF interface.

Refer to the *Network OS Administrator's Guide* for the following related information:

- Conceptual and overview information
- Using DHCP Automatic Deployment (DAD)
- Procedures for configuring the Ethernet management interface
- Basic switch configuration using the Network OS command line interface (CLI)

Using the NETCONF interface, you can perform the following basic switch configuration operations described in this chapter:

- Use the <edit-config> RPC to set host attributes, configure a line card type on a chassis slot, configure a switch banner, enable or disable first failure data capture (FFDC), and configure logging.

- Use custom actions to enable or disable a chassis, reboot a switch, power on/off a line card, obtain slot and module status, and upload supportSave data.
- Use the <show-raslog> custom RPC to return RASlog messages.

Switch management parameters described in this chapter are defined mostly in the brocade-ras, brocade-linecard-management, and brocade-chassis YANG modules. For structural maps of these YANG modules, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of parameters, refer to the corresponding.yang file.

Connecting to the switch through an SSH session

For NETCONF operations, you musty connect to the switch using SSH.

1. Connect through a serial port to the switch.
2. Verify that the switch's network interface is configured and that it is connected to the IP network through the RJ-45 Ethernet port.
3. Log off the switch's serial port.
4. From a management station, open an SSH connection using the management IP address of the switch to which you want to connect.
5. Enter the account user name at the login prompt.
6. Enter the password.

   Brocade recommends that you change the default account password when you log in for the first time. For more information on changing the default password, refer to the Brocade VDX Hardware Reference Manuals.
7. Verify that the login was successful.

   The prompt displays the host name followed by a pound sign (#).

```
login as: admin
admin@10.20.49.112's password:******

-----------------------------------------------------
WARNING: The default password of 'admin' and 'user' accounts have not been
changed.
Welcome to the Brocade Network Operating System Software
admin connected from 10.110.100.92 using ssh on VDX6740
```

Setting host attributes

The following example names the host and the chassis for routing bridge 27.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="300" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <system xmlns="urn:brocade.com:mgmt:brocade-ras">
```

```
                <switch-attributes>
                    <rbridge-id>
                        <rbridge-id>27</rbridge-id>
                        <host-name>lab1_vdx0023</host-name>
                        <chassis-name>lab1_vdx0023</chassis-name>
                    </rbridge-id>
                </switch-attributes>
            </system>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="300" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Obtaining host attribute information

To return the configured host attribute information, issue the <get-config> RPC with a subtree filter to return only the information under the <system>/<switch-attributes> node in the urn:brocade.com:mgmt:brocade-ras namespace, as shown in the following example. Include the <rbridge-id> leaf element under the <switch-attributes> node to restrict output to a specific switch.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="301" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <system xmlns="urn:brocade.com:mgmt:brocade-ras">
                <switch-attributes>
                    <rbridge-id>
                        <rbridge-id>27</rbridge-id>
                    </rbridge-id>
                </switch-attributes>
            </system>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="301" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <system xmlns="urn:brocade.com:mgmt:brocade-ras">
        <switch-attributes>
            <rbridge-id>
                <rbridge-id>27</rbridge-id>
                <host-name>lab1_vdx0023</host-name>
                <chassis-name>lab1_vdx0023</chassis-name>
            </rbridge-id>
        </switch-attributes>
    </system>
</rpc-reply>
```

Disabling or enabling a chassis

To enable a chassis, issue the <chassis>/<enable> custom action located in the urn:brocade.com:mgmt:brocade-chassis namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="302">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <chassis xmlns="urn:brocade.com:mgmt:brocade-chassis">
                <enable/>
            </chassis>
        </nca:data>
    </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="302">
    <ok/>
</rpc-reply>
```

To re-enable a disabled chassis, issue the <chassis>/<disable> custom action also located in the urn:brocade.com:mgmt:brocade-chassis namespace.

```
rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="303">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <chassis xmlns="urn:brocade.com:mgmt:brocade-chassis">
                <disable/>
            </chassis>
        </nca:data>
    </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="303">
    <ok/>
</rpc-reply>
```

Rebooting a Brocade switch

CAUTION

Do not perform a reload operation between a disable operation and an enable operation on a chassis. Your ports will be closed.

To perform a reboot of the entire modular chassis, issue the <ha>/<chassisreboot> custom action located in the urn:brocade.com:mgmt:brocade-ha namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="304">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <ha xmlns="urn:brocade.com:mgmt:brocade-ha">
                <chassisreboot/>
            </ha>
        </nca:data>
    </nca:action>
</rpc>
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="304">
    <ok/>
</rpc-reply>
```

To perform a fastboot operation, issue the <reboot>/<fastboot> custom action located in the urn:brocade.com:mgmt:brocade-firmware namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="305">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <reboot xmlns="urn:brocade.com:mgmt:brocade-firmware">
                <fastboot/>
            </reboot>
        </nca:data>
    </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="305">
    <ok/>
</rpc-reply>
```

NOTE
Both reboot operations are disruptive, and you are prompted for confirmation before executing. When you reboot a switch connected to a fabric, all traffic to and from that switch stops. All ports on that switch remain inactive until the switch comes back online.

Obtaining interface configuration information

To obtain interface information, issue the <get-config> RPC with a subtree filter to return only information under the <interface> node located in the urn:brocade.com:mgmt:brocade-interface namespace.

The following example further restricts the output to 10 Gigabit Ethernet interfaces by specifying the <tengigabitethernet> node in the subtree filter.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="306" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet/>
            </interface>
        </filter>
    </get-config>
</rpc>
```

Retrieving the IP interface information

Use the <get-ip-interface> custom RPC to retrieve the IP interface information.

```
<rpc message-id="307" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-ip-interface xmlns="urn:brocade.com:mgmt:brocade-interface">
```

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

<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>fortygigabitethernet</interface-type>
        <interface-name>2/0/49</interface-name>
        <if-name>FortyGigabitEthernet 2/0/49</if-name>
        <if-state>up</if-state>
        <line-protocol-state>down</line-protocol-state>
        <ip-address>
            <ipv4>unassigned</ipv4>
        </ip-address>
    </interface>
    <interface>
        <interface-type>fortygigabitethernet</interface-type>
        <interface-name>2/0/50</interface-name>
        <if-name>FortyGigabitEthernet 2/0/50</if-name><if-state>up</if-state>
        <line-protocol-state>down</line-protocol-state>
        <ip-address>
            <ipv4>unassigned</ipv4>
        </ip-address>
    </interface>
</rpc-reply>
```

Obtaining slot and module status information

To show information about all slots in the chassis, issue the <slotsinfor>/<slots> custom action located in the urn:brocade.com:mgmt:brocade-linecard-management namespace. The <slotsinfor> node is included in the <show> node of the urn:brocade.com:mgmt:brocade-common-def namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="307">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <slotsinfor
                 xmlns="urn:brocade.com:mgmt:brocade-linecard-management">
                    <slots/>
                </slotsinfor>
            </show>
        </nca:data>
    </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="307">
    <ok/>
</rpc-reply>
```

Similarly, you can show information about specific module types by issuing the following custom actions:

- Issue the <mminfor>/<mm> custom action to display information for the management modules.
- Issue the <sfminfo>/<sfm> custom action to display information for the switch fabric modules.
- Issue the <linecardinfo>/<linecard> custom action to display information for the interface modules.

Replacing an interface module

You can remove an interface module without powering it off. However, doing so will not remove the configuration. When you replace a module with a different type, you must first remove the configuration and then reconfigure the slot for the new interface module type.

Removing the configuration requires the interface module to be powered off.

The example RPCs shown in the following procedure replace the card in slot 1 with a LC48x10G module. These examples assume VCS Fabric mode.

1. Power off the interface module by issuing the <linecardservice>/<power-off> custom action located in the urn:brocade.com:mgmt:brocade-linecard-management namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="308">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <linecardservice
             xmlns="urn:brocade.com:mgmt:brocade-linecard-management">
                <power-off>
                    <linecard>1</linecard>
                </power-off>
            </linecardservice>
        </nca:data>
    </nca:action>
</rpc>

<rpc-reply message-id="308" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. Clear the slot configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="309" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>5</rbridge-id>
                <global-lc-holder
                 xmlns="urn:brocade.com:mgmt:brocade-linecard-management">
                    <linecard>
                        <linecards
                         xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete">
                           <linecardName>1</linecardName>
                        </linecards>
                    </linecard>
                </global-lc-holder>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="309" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Remove the interface module.
4. Specify the new line card type.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="310" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
            <rbridge-id>5</rbridge-id>
                <global-lc-holder
                 xmlns="urn:brocade.com:mgmt:brocade-linecard-management">
                    <linecard>
                        <linecards>
                            <linecardName>1</linecardName>
                            <linecardType>LC48x10G</linecardType>
                        </linecards>
                    </linecard>
                </global-lc-holder>
            </rbridgeid>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="310" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

5. Insert the new interface module into the configured slot.
6. To power on the new line card, reissue the <linecardservice>/<power-off> custom action located in the urn:brocade.com:mgmt:brocade-linecard-management namespace.
7. Issue the <bna-config-cmd> custom RPC in the urn:brocade.com:mgmt:brocade-ras namespace to copy the *running-config* file to the *startup-config* file and save your configuration changes.
8. To verify the configuration change, issue the <get-config> RPC with a subtree filter to return only the contents of the <linecard>/<linecards> node in the urn:brocade.com:mgmt:brocade-linecard-management namespace.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="311" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <rbridgeid xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>5</rbridge-id>
                <global-lc-holder
                 xmlns="urn:brocade.com:mgmt:brocade-linecard-management">
                    <linecard>
                        <linecards>
                            <linecardName>1</linecardName>
                        </linecards>
                    </linecard>
                </global-lc-holder>
```

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

<rpc-reply message-id="311" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <rbridgeid xmlns="urn:brocade.com:mgmt:brocade-rbridge">
        <rbridge-id>5</rbridge-id>
        <global-lc-holder
         xmlns="urn:brocade.com:mgmt:brocade-linecard-management">
            <linecard>
                <linecards>
                    <linecardName>1</linecardName>
                    <linecardType>LC48x10G</linecardType>
                </linecards>
            </linecard>
        </global-lc-holder>
    </rbridgeid>
</rpc-reply>
```

Configuring a switch banner

The following procedure sets and verifies a switch banner.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="312" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <banner xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <login>Please do not disturb the setup on this switch</login>
                <motd>TPS reports are due every Thursday.</motd>
            </banner>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="312" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To verify the banner, issue the <get-config> RPC with a subtree filter to return the contents of the <banner> node in the urn:brocade.com:mgmt:brocade-aaa namespace.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="313" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <banner xmlns="urn:brocade.com:mgmt:brocade-aaa">
        </filter>
    </get-config>
</rpc>
```

```
<rpc-reply message-id="313" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <banner xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <login>Please do not disturb the setup on this switch</login>
    </banner>
</rpc-reply>
```

Configuring BP rate limit

The following example shows how to configure the bp rate limit slot range.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <bp-rate-limit xmlns="urn:brocade.com:mgmt:brocade-bprate-limit">
                    <heavy>
                        <bp-rate-limit-slot-range>1-4</bp-rate-limit-slot-range>
                    </heavy>
                </bp-rate-limit>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>
<rpc-reply message-id="3" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring hardware profiles

The following example selects a route table profile to optimize resources for the maximum number of IPv6 Neighbor Discovery entries and selects a TCAM profile to optimize resources for the maximum number of IPv4/IPv6 multicast entries:

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <hardware-profile xmlns="urn:brocade.com:mgmt:brocade-hardware">
                    <tcam>
                        <predefined>
                            <tcam_profiletype>openflow</tcam_profiletype>
                        </predefined>
                    </tcam>
                    <route-table>
                        <predefined>
```

```
<routing_profiletype>ipv6-max-route</routing_profiletype>
                         <routing_parameter>
                             <maximum_paths>8</maximum_paths>
                             <openflow_enable>on</openflow_enable>
                         </routing_parameter>
                     </predefined>
                 </route-table>
                 <vlan-classification>
                     <predefined>
                         <vlan_profiletype>default</vlan_profiletype>
                     </predefined>
                 </vlan-classification>
                 <kap>
                     <predefined>
                         <kap_profiletype>default</kap_profiletype>
                     </predefined>
                 </kap>
             </hardware-profile>
         </rbridge-id>
     </config>
   </edit-config>
</rpc>

<rpc-reply message-id="3" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Uploading supportSave data to an external host interactively

To upload supportSave data interactively, issue the <copy>/<support-interactive> action located in the urn:brocade.com:mgmt:brocade-ras namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="314">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <copy xmlns="urn:brocade.com:mgmt:brocade-ras">
                <support-interactive/>
            </copy>
        </nca:data>
    </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="314">
    <ok/>
</rpc-reply>
```

The switch responds with a dialog for accessing and uploading to the external host.

Uploading supportSave to an external host using FTP

To upload supportSave data to an external host using FTP, issue the <copy>/<support>/<ftp> custom action located in the urn:brocade.com:mgmt:brocade-ras namespace. Under the <ftp> node, include the following leaf elements:

- In the <user> and <password> elements, provide valid login credentials for an account on the FTP server.

- In the <host> field, specify the IP address of the FTP server.

  IPv6 addresses are valid only on Network OS 3.0.0 platforms.
- In the <directory> field, specify the path to the directory on the FTP server where you want to store the supportSave data.
- (VCS Fabric mode only) In the <rbridge-id> field, specify the RBridge ID of the switch whose supportSave data you want to save.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="315">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <copy xmlns="urn:brocade.com:mgmt:brocade-ras">
                <support>
                    <ftp>
                        <user>admin</user>
                        <host>10.38.33.131</host>
                        <directory>/home/admin/support</directory>
                        <password>h8F!@m</password>
                        <rbridge-id>5</rbridgeid>
                    </ftp>
                </support>
            </copy>
        </nca:data>
    </nca:action>
</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="315">
    <copy xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
        <support>
            <ftp>
                <supportSaveResult>
                    <rbridge-id>5</rbridge-id>
                    <status-code>success</status-code>
                </supportSaveResult>
            </ftp>
        </support>
    </copy>
</rpc-reply>
```

Uploading supportSave to an external host using SCP

To upload supportSave data to an external host using SCP, issue the <copy>/<support>/<scp> custom action located in the urn:brocade.com:mgmt:brocade-ras namespace. Under the <scp> node, include the following leaf elements:

- In the <user> and <password> elements, provide valid login credentials for an account on the SCP server.
- In the <host> field, specify the IP address of the SCP server.

  IPv6 addresses are valid only on Network OS 3.0.0 platforms.
- In the <directory> field, specify the path to the directory on the SCP server where you want to store the supportSave data.
- (VCS Fabric mode only) In the <rbridge-id> field, specify the RBridge ID of the switch whose supportSave data you want to save.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="316">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <copy xmlns="urn:brocade.com:mgmt:brocade-ras">
                <support>
                    <scp>
                        <user>admin</user>
                        <host>10.38.33.131</host>
                        <directory>/home/admin/support</directory>
                        <password>h8F!@m</password>
                        <rbridge-id>5</rbridgeid>
                    </scp>
                </support>
            </copy>
        </nca:data>
    </nca:action>
</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="316">
    <copy xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
        <support>
            <scp>
                <supportSaveResult>
                    <rbridge-id>5</rbridge-id>
                    <status-code>success</status-code>
                </supportSaveResult>
            </scp>
        </support>
    </copy>
</rpc-reply>
```

Saving supportSave data to an attached USB device

You can use a Brocade-branded USB device to save the support data. The Brocade-branded USB device comes with factory-configured default directories and interacts with the Network OS CLI.

1. To enable the USB device, issue the <system>/<usb>/<on> custom action located in the urn:brocade.com:mgmt:brocade-ras namespace.

   In the VCS Fabric mode only, include the <rbridge-id> leaf element under the <on> node and specify the routing bridge on which you want to enable the USB device.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="317">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <system xmlns="urn:brocade.com:mgmt:brocade-ras">
                <usb>
                    <on>
                        <rbridge-id>27</rbridge-id>
                    </on>
                </usb>
            </system>
        </nca:data>
    </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="317">
    <ok/>
```

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

2. To display the default directories, issue the <system>/<usb>/<dir> custom action located in the urn:brocade.com:mgmt:brocade-ras namespace.

   In the VCS Fabric mode only, include the <rbridge-id> leaf element under the <dir> node and specify the routing bridge for which you want to display the default directories.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="318">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <system xmlns="urn:brocade.com:mgmt:brocade-ras">
                <usb>
                    <dir>
                        <rbridge-id>27</rbridge-id>
                    </dir>
                </usb>
            </system>
        </nca:data>
    </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="318">
    <ok/>
</rpc-reply>
```

3. Issue the <copy>/<support>/<usb> custom action located in the urn:brocade.com:mgmt:brocade-ras namespace to copy the supportSave information to the USB device. Under the <usb> node, include the following leaf elements:

   - In the <directory> element, specify the directory where the supportSave data will be copied.
   - (VCS Fabric mode only) In the <rbridge-id> element, specify the RBridge ID of the switch whose data you want copied. Specify “all” to copy supportSave data for all switches in the Fabric cluster.
   - (Optional) In the <timeout> element, provide a supportSave timeout multiplier. This value increases timeout values associated with supportSave operations. For example, a value of 2 doubles timeouts.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="319">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <copy xmlns="urn:brocade.com:mgmt:brocade-ras">
                <support>
                    <usb>
                        <directory>support</directory>
                        <rbridge-id>5</rbridgeid>
                        <timeout>2</timeout>
                    </usb>
                </support>
            </copy>
        </nca:data>
    </nca:action>
</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="319">
    <copy xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
        <support>
            <usb>
```

```
            <supportSaveResult>
                <rbridge-id>5</rbridge-id>
                <status-code>success</status-code>
            </supportSaveResult>
        </usb>
    </support>
</copy>
</rpc-reply>
```

Enabling or disabling FFDC

The following example enables FFDC on routing bridge 56.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="320" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <support xmlns="urn:brocade.com:mgmt:brocade-ras">
                <rbridge-id>
                    <rbridge-id>56</rbridge-id>
                    <ffdc/>
                </rbridge-id>
            </support>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="320" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To disable FFDC, use the same RPC, but include the delete operation in the leading tag of the <rbridge-id> node element.

Adding syslog servers

The following example adds four syslog servers. It sets the secure mode on servers 192.168.163.233 and fec0:60:69bc:92:218:8bff:fe40:15c4 and specifies a port number for each server. It also adds 192.168.163.235 and 192.168.162.326 in non-secure mode with the default port value of 514.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="321" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <logging xmlns="urn:brocade.com:mgmt:brocade-ras">
                <syslog-server>
                    <syslogip>192.168.163.233</syslogip>
                    <secure/>
                    <port>2000</port>
                </syslog-server>
```

```
                <syslog-server>
                    <syslogip>fec0:60:69bc:92:218:8bff:fe40:15c4</syslogip>
                    <secure/>
                    <port>1999</port>
                </syslog-server>
                <syslog-server>
                    <syslogip>192.168.163.235</syslogip>
                </syslog-server>
                <syslog-server>
                    <syslogip>192.168.163.236</syslogip>
                </syslog-server>
            </logging>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="321" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To verify the syslog server configuration, issue the <edit-config> RPC with a subtree filter to return only information under the <logging> node in the urn:brocade.com:mgmt:brocade-ras namespace.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="322" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <logging xmlns="urn:brocade.com:mgmt:brocade-ras">
                <syslog-server/>
            </logging>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="322" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <logging xmlns="urn:brocade.com:mgmt:brocade-ras">
        <syslog-server>
            <syslogip>192.168.163.233</syslogip>
            <secure/>
            <port>2000</port>
        </syslog-server>
        <syslog-server>
            <syslogip>fec0:60:69bc:92:218:8bff:fe40:15c4</syslogip>
            <secure/>
            <port>1999</port>
        </syslog-server>
        <syslog-server>
            <syslogip>192.168.163.235</syslogip>
        </syslog-server>
        <syslog-server>
            <syslogip>192.168.163.236</syslogip>
        </syslog-server>
    </logging>
</rpc-reply>
```

Modifying the syslog server configuration

You can change the secure mode and the port number of a configured syslog server. The following example disables secure mode for the syslog server 192.168.163.233, enables secure mode for the syslog server 192.168.163.236. It also sets the port number for the newly secured server to 2001.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="323" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <logging xmlns="urn:brocade.com:mgmt:brocade-ras">
                <syslog-server>
                    <syslogip>192.168.163.233</syslogip>
                    <secure xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                </syslog-server>
                <syslog-server>
                    <syslogip>192.168.163.236</syslogip>
                    <secure/>
                    <port>2001</port>
                </syslog-server>
            </logging>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="323" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Importing a syslog CA certificate

The following example shows how to import a syslog CA certificate.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="324">
   <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <data>
            <certutil xmlns="urn:brocade.com:mgmt:brocade-certutil">
                <import>
                    <syslogca>
                        <protocol>SCP</protocol>
                        <user>testuser</user>
                        <password>password</password>
                        <host>10.70.4.101</host>
                        <directory>/users/home40/testuser</directory>
                        <file>ca.cert</file>
                    </syslogca>
                </import>
            </certutil>
        </data>
    </action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="324">
```

```
    <ok/>
</rpc-reply>
```

Removing a syslog CA certificate

To delete the CA certificate, issue the <syslogca> action located in the <no>/<certutil> node, where the <no> element resides in the urn:brocade.com:mgmt:brocade-common-def namespace and the <certutil> node resides in the urn:brocade.com:mgmt:brocade-certutil namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="325">
   <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
       <data>
           <no xmlns="urn:brocade.com:mgmt:brocade-common-def">
               <certutil xmlns="urn:brocade.com:mgmt:brocade-certutil">
                   <syslogca/>
               </certutil>
           </no>
       </data>
    </action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="325">
    <ok/>
</rpc-reply>
```

Removing a syslog server

The following example removes a syslog server.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="326" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <logging xmlns="urn:brocade.com:mgmt:brocade-ras">
                <syslog-server xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete">
                    <syslogip>192.168.163.236</syslogip>
                </syslog-server>
            </logging>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="326" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Displaying the RASlog messages

To display the RASlog messages, issue the <show-raslog> custom RPC located in the urn:brocade.com:mgmt:brocade-ras-ext namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="327">
    <show-raslog xmlns="urn:brocade.com:mgmt:brocade-ras-ext"/>
```

```
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="327">
    <show-all-raslog xmlns="urn:brocade.com:mgmt:brocade-ras-ext">
        <rbridge-id>tengigabitethernet</interface-type>
        <number-of-entries>237</number-of-entries>
        <raslog-entries>
            <index>13187</index>
            <message-id>NSM-2006</message-id>
            <date-and-time-info>2000-03-11/20:12:03.1234</date-and-time-info>
            <severity>informational</severity>
            <message>Port-profile aa1 removed successfully on TenGigabitEthernet/
             2/0/17</message>
            <message-flag>other</message-flag>
            <switch-or-chassis-name>switchA</switch-or-chassis-name>
        </raslog-entries>
        <raslog-entries>
(output truncated)
```

Setting the RASlog severity filter

You can choose one of the following severity levels to filter RASlog messages: INFO (default), ERROR, WARNING, or CRITICAL. Input values are case-sensitive. The configured severity level marks the reporting threshold. All messages with the configured severity or higher are displayed.

The following example show how to set the RASlog severity filter in the VCS Fabric mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="328" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <logging xmlns="urn:brocade.com:mgmt:brocade-ras">
                <rbridge-id>
                    <rbridge-id>23</rbridge-id>
                    <raslog>
                        <console>WARNING</console>
                    </raslog>
                </rbridge-id>
            </logging>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="328" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Audit log configuration

Audit log messages contain user information such as login name and login IP address. The audit log's purpose is to enable tracking of important user-originated events in the cluster; this is in contrast to RASlog messages, which are primarily used for abnormal or error-related events.

When an audit log message is generated on a switch, it is forwarded to the syslog server. To limit the audit log messages to the syslog server and facilitate monitoring of the audit log messages, three audit log classes are defined: FIRMWARE, SECURITY, and CONFIGURATION.

You must enable the audit log class to generate the audit log messages for that class. The classes are enabled by default.

The following example enables SECURITY and WARNING messages, but disables CONFIGURATION messages.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="329" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <logging xmlns="urn:brocade.com:mgmt:brocade-ras">
                <auditlog>
                    <class xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete">
                        <class>CONFIGURATION</class>
                    </class>
                    <class>
                        <class>WARNING</class>
                    </class>
                    <class>
                        <class>SECURITY</class>
                    </class>
                </auditlog>
            </logging>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="325" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Retrieving the system uptime

Use the <get-system-uptime> custom RPC to retrieve the system uptime information.

```
<rpc message-id="307" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-system-uptime xmlns="urn:brocade.com:mgmt:brocade-system">
    </get-system-uptime>
</rpc>

<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">rbridge-id>
        <days>0</days>
        <hours>5</hours>
        <minutes>53</minutes>
        <seconds>4</seconds>
    </show-system-uptime>
</rpc-reply>
```

Auto Fabric

Auto Fabric is a feature that allow plug and play for Brocade VDX switches.

Configuring Auto fabric for bare metal

The following example shows how to create a bare-metal configuration.:

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <preprovision xmlns="urn:brocade.com:mgmt:brocade-preprovision">
                <rbridge-id>
                    <rbridge-id>3</rbridge-id>
                    <wwn>11:11:11:11:11:11:11:15</wwn>
                </rbridge-id>
            </preprovision>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Obtaining bare-metal state on the system

<show-bare-metal-state> returns the bare-metal state on the system.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:message-id="3">
    <show-bare-metal-state
xmlns="urn:brocade.com:mgmt:brocade-preprovision"></show-bare-metal-state>
</nc:rpc>

<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="3">
    <bare-metal-state
xmlns="urn:brocade.com:mgmt:brocade-preprovision">disable</bare-metal-state>
</rpc-reply>
```

Retrieving VCS RBridge detail

<vcs-rbridge-config> returns the VCS RBridge details.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:message-id="3">
    <vcs-rbridge-config xmlns="urn:brocade.com:mgmt:brocade-vcs">
        <vcs-id>50</vcs-id>
        <rbridge-id>4</rbridge-id>
    </vcs-rbridge-config>
</nc:rpc>
```

Chapter

Network Time Protocol 4

In this chapter

Time management with NETCONF overview

Through the NETCONF interface, you can perform the following operations for managing time:

- Use the <clock-set-datetime> action to set the local clock date and time.
- Use the <clock-set-timezone> action to set the time zone.
- Use the <no>/<clock>/<timezone> action to clear the time zone data.
- Use the <show-clock> RPC to return the local time, date, and time zone.
- Use the <edit-config> RPC to configure an NTP server.
- Use the <show-ntp> custom RPC to obtain the NTP server address.
- Use the <get-config> RPC to validate configuration settings.

NTP parameters are defined in the brocade-ntp YANG module. Date and time parameters are defined in the brocade-clock YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Setting the date and time

The following example sets the local time to 2:15 in the afternoon of May 17, 2012.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" nc:message-id="304">
   <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
      <nca:data>
         <clock-set-datetime xmlns="urn:brocade.com:mgmt:brocade-clock">
            <clock>
               <set>2012-05-17T14:15:00</set>
            </clock>
         </clock-set-datetime>
      </nca:data>
   </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="304">
    <ok/>
</rpc-reply>
```

Setting the time zone

The following example sets the time zone for Los Angeles, California.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="305">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <clock-set-timezone xmlns="urn:brocade.com:mgmt:brocade-clock">
                <clock>
                    <timezone>America/Los_Angeles</timezone>
                </clock>
            </clock-set-timezone>
        </nca:data>
    </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
     message-id="305">
    <ok/>
</rpc-reply>
```

Retrieving the current local clock and time zone

The following example returns the clock and time zone information for the switch with routing bridge ID 66.

```
<rpc message-id="307" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-clock xmlns="urn:brocade.com:mgmt:brocade-clock">
        <rbridge-id>66</rbridge-id>
    </show-clock>
</rpc>

<rpc-reply message-id="307" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <clock-time xmlns="urn:brocade.com:mgmt:brocade-clock">
        <rbridge-id-out>66</rbridge-id-out>
        <current-time>2012-05-17T12:15:00</current-time>
        <timezone>America/Los_Angeles</</timezone>
    </clock-time>
</rpc-reply>
```

Removing the time zone setting

The following example removes the time zone setting.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="306">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <no xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <clock xmlns="urn:brocade.com:mgmt:brocade-clock">
                    <timezone>
                    </timezone>
                </clock>
            </no>
        </nca:data>
    </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="306">
```

```
    <ok/>
</rpc-reply>
```

Synchronizing the local time with an external source

To add an NTP server IP address to the list of server IP addresses, issue the <edit-config> RPC to configure the <ntp>/<server> node in the urn:brocade.com:mgmt:brocade-ntp namespace. Under the <server> node, set the value of the <ip> element to the IPv4 or IPv6 address, as shown in the following example.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="301" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ntp xmlns="urn:brocade.com:mgmt:brocade-ntp" >
                <server>
                    <ip>192.168.10.1</ip>
                </server>
            </ntp>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="301" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Retrieving an NTP server IP address

Use the <show-ntp> custom RPC located in the urn:brocade.com:mgmt:brocade-ntp namespace to return the IP address of the currently active NTP server. If no server is configured or no server can be reached, “LOCL” is returned instead (for local switch time). The request is for the local switch unless a switch ID is specified in the <rbridge-id> element.

NOTE
Specifying “all” in the <rbridge-id> element returns only local information.

The response includes either an <ip> element containing the IPv4 or IPv6 address, or a Boolean <LOCL> element, with its value set to “true”.

```
<rpc message-id="303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-ntp xmlns="urn:brocade.com:mgmt:brocade-ntp">
        <rbridge-id>66</rbridge-id>
    </show-ntp>
</rpc>

<rpc-reply message-id="303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <node-active-server xmlns="urn:brocade.com:mgmt:brocade-ntp">
        <rbridge-id-out>66</rbridge-id-out>
        <LOCL>true</LOCL>
    </node-active-server>
</rpc-reply>
```

Removing an NTP server IP address

Use this operation to remove an NTP server IP address from a list of server IP addresses. At least one IP address in the remaining list must be a reachable, configured NTP server or the remove request fails.

The following example shows how to remove an NTP server IP address from the list of server IP addresses.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="302" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ntp xmlns="urn:brocade.com:mgmt:brocade-ntp" >
                <server xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                operation="delete">
                    <ip>192.168.10.1</ip>
                </server>
            </ntp>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="302" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Installing and Maintaining Firmware

5

In this chapter

Firmware upgrade with NETCONF overview

Brocade firmware upgrades consist of multiple firmware packages listed in a .plist file. The .plist file contains specific firmware information (time stamp, platform code, version, and so forth) and the names of the firmware packages to be downloaded. These packages are made available periodically to add features or to remedy defects in the firmware.

Firmware upgrades are performed incrementally. The firmware download operation compares the new firmware packages against the current installation and only downloads the packages that contain new features or have been modified.

You can download the firmware from a remote server using the File Transfer Protocol (FTP), Secure Copy Protocol (SCP), Secure File Transfer Protocol (SFTP), or you can download the firmware from an attached Brocade-branded USB device.

This chapter describes procedures for installing and maintaining firmware using the NETCONF interface. Refer to the *Network OS Administrator's Guide* for the following information:

- Firmware download concepts
- Overview information about how Network OS performs firmware upgrade on compact switches and on modular switches
- Upgrade and downgrade considerations
- Error handling

Through the NETCONF interface, you can perform the following operations on firmware:

- Use the <show-firmware-version> custom RPC to obtain firmware version information.
- Use the <download>/<ftp> action to download firmware from an FTP server.
- Use the <download>/<scp> action to download firmware from a Secure Copy Protocol.
- Use the <download>/<sftp> action to download firmware from a Secure FTP server.
- Use the <download>/<usb> action to load firmware from a USB device.
- Use the <usb>/<on> device to gain access to a USB device.

- Use the <fwdl-status> custom RPC to query the status of a download operation.
- Use the <firmware-commit> action to commit a firmware upgrade.
- Use the <firmware-restore> action to restore a previous firmware version.

Firmware download parameters, Custom RPCs, and actions are defined in the brocade-firmware YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Preparing for a firmware download

To prepare for a firmware upgrade, perform the tasks listed in this section. In the unlikely event of a failure or timeout, you will be able to provide your switch support provider the information required to troubleshoot the firmware download.

1. Verify the current firmware version. Refer to "Obtaining the switch firmware version" on page 50 for details.
2. Decide on a migration path. Check the connected devices to ensure firmware compatibility and that any older versions are supported. Refer to the Network OS Compatibility section of the *Brocade Network OS Release Notes* for the recommended firmware version.
3. Back up your switch configuration prior to the firmware download. Refer to "Managing the configuration" on page 19 for details.
4. *Optional:* For additional support, connect the switch to a computer with a serial console cable. Ensure that all serial consoles and any open network connection sessions, such as Telnet, are logged and included with any trouble reports.
5. Perform the copy support operation to collect all current core files before executing the firmware download. This information helps to troubleshoot the firmware download process in the event of a problem.
6. *Optional:* Clear RASlog messages to erase all existing messages in addition to internal messages.

Obtaining the switch firmware version

Use the <show-firmware-version> custom RPC to obtain the following information:

- Network Operating System version—The firmware version number
- Build time—The build date and time of the firmware
- Firmware name—The label of the firmware image
- Control Processor—CP model and memory

To retrieve switch firmware information, issue the <show-firmware-version> custom RPC from the urn:brocade.com:mgmt:brocade-firmware-ext namespace, and specify the routing bridge ID of the switch you want to query in the <switchid> input parameter.

```
<rpc message-id="401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
       <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
           <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
       </show-firmware-version>
</rpc>

<rpc-reply message-id="401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
        <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
            <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
            <os-name>Network Operating system Software</os-name>
            <os-version>4.0.0</os-version>
            <copy-right-info>1995-2010 Brocade Communications Systems, Inc.
             </copy-right-info>
            <build-time>19:18:58 Jun 23, 2012</build-time>
            <firmware-full-version>v4.0.0_bldg56</firmware-full-version>
            <control-processor-vendor>Freescale Semiconductor
             </control-processor-vendor>
            <control-processor-chipset>8548E</control-processor-chipset>
            <control-processor-memory>2000 MB</control-processor-memory>
            <node-info>
                <slot-no>1</slot-no>
                <node-instance-no>1</node-instance-no>
                <node-type>type-mm</node-type>
                <is-active-cp>ytue</is-active-cp>
                <firmware-version-info>
                    <primary-version>v4.0.0_bldg56</primary-version>
                    <secondary-version>v4.0.0_bldg56</secondary-version>
                </firmware-version-info>
            </node-info>
        </show-firmware-version>
</rpc>
```

Obtaining and decompressing firmware

Firmware upgrades are available for customers with support service contracts and for partners on the Brocade website at *http://www.mybrocade.com*.

You must download the firmware package either to an FTP server or to a USB device and decompress the package before you can use the download operations described in this chapter to update the firmware on your equipment. Use the UNIX **tar** command for .tar files, the **gunzip** command for all .gz files, or a Windows unzip program for all .zip files.

When you unpack the downloaded firmware, it expands into a directory that is named according to the firmware version. The firmware download operations, when issued with the path to the directory where the firmware is stored, perform an automatic search for the correct package file type associated with the device.

Connecting to the switch

When you upgrade firmware in default mode, you connect to the switch through the management IP address. Modular switches have one management IP address for the chassis and separate IP addresses for each management module. To upgrade both management modules, you can either connect to the chassis management IP address or to the IP address of the active management module. If you want to upgrade a single management module only, you must connect to the IP address of that management module and perform the firmware download operation in manual mode. In manual mode, only the local management module is upgraded.

Downloading the firmware from a remote server

Under normal circumstances, it is recommended to perform firmware download in the default mode. Do not disable autocommit mode unless you want to evaluate a firmware upgrade before committing to it. Refer to "Evaluating a firmware upgrade" on page 55 for details about overriding the autocommit mode.

When upgrading multiple switches, complete the following steps on each switch before you upgrade the next one.

1. Verify that the FTP or SSH server is running on the remote server and that you have a valid user ID and password on that server.
2. Download the firmware package from the Brocade website and store the file on the FTP or SSH server.

   To download the firmware from an attached USB device, refer to "Downloading firmware from a USB device" on page 54.
3. Decompress the firmware archive.
4. Connect to the switch or management module you are upgrading.
5. Obtain the switch firmware version information. Refer to "Obtaining the switch firmware version" on page 50 for details.
6. Issue the <download>/<ftp> action in the urn:brocade.com:mgmt:brocade-firmware namespace to perform the firmware download operation. Provide the following input elements:
   - <user>—The user ID on the remote server
   - <password>—The user password
   - <host>—The IPv4 or IPv6 IP address
   - <directory>—The directory on the remote server where the firmware file is located
   - <file>—The firmware filename

   The reply message contains a session ID in the <fwdl-tid> element.

   NOTE
   To be able to mention the FTP server by name, a Domain Name System (DNS) entry must exist for the server.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="402">
<action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
    <data>
        <firmware xmlns="urn:brocade.com:mgmt:brocade-firmware">
            <download>
                <ftp>
                    <user>fvt</user>
                    <password>party4green</password>
                    <host>10.1.2.30</host>
                    <directory>/</directory>
                    <file>release.plist</file>
                </ftp>
            </download>
        </firmware>
    </data>
</action>
</rpc>
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="402">
    <data>
        <firmware xmlns="urn:brocade.com:mgmt:brocade-firmware">
            <download>
                <ftp>
                    <fwdl-tid>34</fwdl-tid>
                    <fwdl-status>0</fwdl-status>
                </ftp>
            </download>
        </firmware>
    </data>
</rpc-reply>
```

CAUTION

Do not interrupt the firmware download process. If you encounter a problem, wait for the timeout (30 minutes for network problems) before attempting the firmware download operation again. Disrupting the process (for example, by disconnecting the switch from the power source) can render the switch inoperable and may require you to seek help from your switch service provider.

7. While the upgrade is proceeding, you can use the <fwdl-status> custom RPC with the value returned in the <fwdl-tid> element in step 6 to query the status of the download, as shown in the following example.

```
<rpc message-id="403" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <fwdl-status xmlns="urn:brocade.com:mgmt:brocade-firmware">
        <fwdl-tid>34</fwdl-tid>
    </fwdl-status>
</rpc>

<rpc-reply message-id="403" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <data>
        <number-of-entries>1</number-of-entries>
        <fwdl-state>complete</fwdl-state>
        <fwdl-entries>
            <index>54</index>
            <message-id>1</message-id>
            <date-and-time-info>2012-07-23/14:32:24:1234</date-and-time-info>
            <message>Firmware has been downloaded successfully</message>
            <blade-slot>1</blade-slot>
            <blade-swbd>v4.0.0_bldg56</blade-swdb>
            <blade-name>A1</blade-name>
            <blade-state>active</blade-state>
            <blade-app>BFOS</blade-app>
        </fwdl-entries>
        <fwdl-entries>
            <index>55</index>
            <message-id>2</message-id>
            <date-and-time-info>2012-07-23/14:32:24:1234</date-and-time-info>
            <message>The commit operation has completed successfully</message>
        </fwdl-entries>
    </data>
</rpc>
```

8. After the switch reboots, issue the <show-firmware-version> custom RPC located in the urn:brocade.com:mgmt:brocade-firmware-ext namespace to verify the firmware upgrade.

```
<rpc message-id="404" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
        <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
    </show-firmware-version>
</rpc>

<rpc-reply message-id="404" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
        <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
        <os-name>Network Operating system Software</os-name>
        <os-version>4.0.0</os-version>
        <copy-right-info>1995-2010 Brocade Communications Systems, Inc.
         </copy-right-info>
        <build-time>19:18:58 Jun 23, 2012</build-time>
        <firmware-full-version>v4.0.0_bldg56</firmware-full-version>
        <control-processor-vendor>Freescale Semiconductor
         </control-processor-vendor>
        <control-processor-chipset>8548E</control-processor-chipset>
        <control-processor-memory>2000 MB</control-processor-memory>
        <node-info>
            <firmware-version-info>
                <primary-version>v4.0.0_bldg56</primary-version>
            </firmware-version-info>
        </node-info>
    </show-firmware-version>
</rpc>
```

CAUTION

Do not interrupt the firmware download process. If you encounter a problem, wait for the timeout (30 minutes for network problems) before attempting the download operation again. Disrupting the process (for example, by disconnecting the switch from the power source) can render the switch inoperable and may require you to seek help from your switch service provider.

Downloading firmware from a USB device

The Brocade VDX 6740, and 8770 switches support firmware download from a Brocade-branded USB device. Third-party USB devices are not supported. Before you can access the USB device, you must enable the device and mount it as a file system. The firmware images to be downloaded must be stored in the factory-configured firmware directory. Multiple images can be stored under this directory.

1. Ensure that the USB device is connected to the switch.
2. Issue the <usb>/<on> action located in the brocade.com:mgmt:brocade-ras namespace and specify the routing bridge ID in the <switchid> input element, as shown in the following example.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="405">
    <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <data>
            <usb xmlns="urn:brocade.com:mgmt:brocade-ras">
                <on>
                    <switchid>23</switchid>
                </on>
            </usb>
```

```
        </data>
    </action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="405">
    <ok/>
</rpc-reply>
```

3. Issue the <download>/<usb> action located in the urn:brocade.com:mgmt:brocade-firmware namespace to perform the firmware download operation. In the <directory> element, provide the directory on the remote server where the firmware file is located.

   The reply message contains a session ID in the <fwdl-tid> element.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="406">
    <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <data>
            <firmware xmlns="urn:brocade.com:mgmt:brocade-firmware">
                <download>
                    <usb>
                        <directory>NOS_v4.0.0</directory>
                    </usb>
                </download>
            </firmware>
        </data>
    </action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
 message-id="406">
    <data>
        <firmware xmlns="urn:brocade.com:mgmt:brocade-firmware">
            <download>
                <ftp>
                    <fwdl-tid>34</fwdl-tid>
                    <fwdl-status>0</fwdl-status>
                </ftp>
            </download>
        </firmware>
    </data>
</rpc-reply>
```

Evaluating a firmware upgrade

CAUTION

Because of potential compatibility issues, Brocade does not recommend restoring Network OS v2.1.x after you upgraded to Network OS v3.0.0.

You can restore a previous firmware version after downloading and evaluating a newer (or older) version by downloading the firmware to a single partition only. The previous version is preserved on the secondary partition and you can restore it if necessary.

- To enable firmware restoration on a compact switch, you perform the firmware download operation with the <nocommit> option. This option prevents the firmware download from copying the firmware to both partitions and committing the upgrade.

- To enable firmware restoration on a modular switch with two management modules, you update the firmware on each management module separately by performing the firmware download operation with both the <manual> and <nocommit> options. This sequence or operations preserves the previous firmware on the secondary partitions of all system components and ensures that you will be able to restore the previous firmware version.

ATTENTION
When you evaluate a firmware upgrade, make sure you disable all features that are supported only by the upgraded firmware before restoring the original version.

Downloading firmware to a single partition

1. Verify that the SFTP, FTP, or SSH server is running on the host server and that you have a user ID on that server.
2. Obtain the firmware file from the Brocade website at *http://www.mybrocade.com* or from your switch support provider and store the file on the FTP or SSH server.
3. Unpack the compressed firmware archive.
4. Issue the <show-firmware-version> custom RPC located in the urn:brocade.com:mgmt:brocade-firmware-ext namespace to view the current firmware version.

```
<rpc message-id="407" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
        <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
    </show-firmware-version>
</rpc>

<rpc-reply message-id="407" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
        <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
        <os-name>Network Operating system Software</os-name>
        <os-version>4.0.0</os-version>
        <copy-right-info>1995-2010 Brocade Communications Systems, Inc.
         </copy-right-info>
        <build-time>19:18:58 Jun 23, 2012</build-time>
        <firmware-full-version>v4.0.0_bldg56</firmware-full-version>
        <control-processor-vendor>Freescale Semiconductor
         </control-processor-vendor>
        <control-processor-chipset>8548E</control-processor-chipset>
        <control-processor-memory>2000 MB</control-processor-memory>
        <node-info>
            <firmware-version-info>
                <primary-version>v4.0.0_bldg56</primary-version>
                <secondary-version>v4.0.0_bldg56</primary-version>
            </firmware-version-info>
        </node-info>
    </show-firmware-version>
</rpc>
```

5. Issue the <download>/<ftp> action located in the urn:brocade.com:mgmt:brocade-firmware namespace to perform the firmware download operation. Provide the following input elements:
   - <user>—The user ID on the remote server.
   - <password>—The user password.
   - <host>—The IPv4 or IPv6 IP address.

- <directory>—The directory on the remote server where the firmware file is located.
- <file>—The firmware filename.
- <nocomit>—Ensures the firmware image is downloaded only to the primary partition.

The reply message contains a session ID in the <fwdl-tid> element.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="408">
    <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <data>
            <firmware xmlns="urn:brocade.com:mgmt:brocade-firmware">
                <download>
                    <ftp>
                        <user>fvt</user>
                        <password>pary4green</password>
                        <host>10.1.2.30</host>
                        <directory>/</directory>
                        <file>release.plist</file>
                        <nocommit/>
                    </ftp>
                </download>
            </firmware>
        </data>
    </action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="408">
    <data>
        <firmware xmlns="urn:brocade.com:mgmt:brocade-firmware">
            <download>
                <ftp>
                    <fwdl-tid>34</fwdl-tid>
                    <fwdl-status>0</fwdl-status>
                </ftp>
            </download>
        </firmware>
    </data>
</rpc-reply>
```

The switch will perform a reboot and come up with the new firmware. Your current switch session will automatically disconnect.

6. Issue the <show-firmware-version> custom RPC to confirm that the primary partition of the switch contains the new firmware, and the secondary does not.

```
<rpc message-id="409" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
        <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
    </show-firmware-version>
</rpc>

<rpc-reply message-id="409" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
        <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
        <os-name>Network Operating system Software</os-name>
        <os-version>4.0.0</os-version>
        <copy-right-info>1995-2010 Brocade Communications Systems, Inc.
         </copy-right-info>
        <build-time>19:18:58 Jun 23, 2012</build-time>
        <firmware-full-version>v4.0.0_bldg56</firmware-full-version>
```

```
        <control-processor-vendor>Freescale Semiconductor
         </control-processor-vendor>
        <control-processor-chipset>8548E</control-processor-chipset>
        <control-processor-memory>2000 MB</control-processor-memory>
        <node-info>
            <firmware-version-info>
                <primary-version>v3.0.1_bldg57</primary-version>
                <secondary-version>v4.0.0_bldg56</primary-version>
            </firmware-version-info>
        </node-info>
    </show-firmware-version>
</rpc>
```

ATTENTION

If you want to *restore* the firmware, stop here and skip ahead to "Restoring the previous firmware version" on page 59; otherwise, continue to "Commiting the firmware upgrade" on page 58 to complete the firmware download process.

You are now ready to evaluate the new version of firmware.

Commiting the firmware upgrade

If you decide to keep the firmware upgrade, use the firmware commit operation to update the secondary partition with new firmware. On modular switches you must perform this operation on both management modules. It may take several minutes to complete the commit operation.

1. Issue the <firmware> action and specify the <commit> element.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="410">
    <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <data>
            <firmware xmlns="urn:brocade.com:mgmt:brocade-firmware">
                <commit/>
            </firmware>
        </data>
    </action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="410">
    <data>
        <firmware xmlns="urn:brocade.com:mgmt:brocade-firmware">
            <commit>
                <result></result>
            </commit>
        </firmware>
    </data>
</rpc-reply>
```

2. Issue the <show-firmware-version> custom RPC to confirm that both the primary partition and the secondary partition of the switch contain the new firmware.

```
<rpc message-id="411" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
        <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
    </show-firmware-version>
</rpc>

<rpc-reply message-id="411" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
        <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
        <os-name>Network Operating system Software</os-name>
        <os-version>4.0.0</os-version>
        <copy-right-info>1995-2010 Brocade Communications Systems, Inc.
         </copy-right-info>
        <build-time>19:18:58 Jun 23, 2012</build-time>
        <firmware-full-version>v4.0.0_bldg56</firmware-full-version>
        <control-processor-vendor>Freescale Semiconductor
         </control-processor-vendor>
        <control-processor-chipset>8548E</control-processor-chipset>
        <control-processor-memory>2000 MB</control-processor-memory>
        <node-info>
            <firmware-version-info>
                <primary-version>v4.0.1_bldg67</primary-version>
                <secondary-version>v4.0.1_bldg67</primary-version>
            </firmware-version-info>
        </node-info>
    </show-firmware-version>
</rpc>
```

Restoring the previous firmware version

Use the firmware restore operation to back out of a firmware upgrade. This option works only if autocommit mode was disabled during the firmware download. On modular switches you must perform this operation on both management modules.

1. Issue the <firmware> action and specify the <restore> element.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="412">
    <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <data>
            <firmware xmlns="urn:brocade.com:mgmt:brocade-firmware">
                <restore/>
            </firmware>
        </data>
    </action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="412">
    <data>
        <firmware xmlns="urn:brocade.com:mgmt:brocade-firmware">
            <commit>
                <result></result>
            </commit>
        </firmware>
    </data>
</rpc-reply>
```

   The switch will reboot and come up with the original firmware.

   The firmware commit operation will begin to copy the original firmware from the secondary partition to the primary partition. When this process completes, both partitions will have the original firmware. It may take several minutes to complete the operation.

2. Wait until all processes have completed and the switch is fully up and operational.
3. Issue the <show-firmware-version> custom RPC and verify that both partitions on the switch have the original firmware.

```
<rpc message-id="413" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
        <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
    </show-firmware-version>
</rpc>

<rpc-reply message-id="413" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-firmware-version xmlns="urn:brocade.com:mgmt:brocade-firmware-ext">
        <switchid xmlns="urn:brocade.com:mgmt:brocade-ras-ext">24</switchid>
        <os-name>Network Operating system Software</os-name>
        <os-version>4.0.0</os-version>
        <copy-right-info>1995-2010 Brocade Communications Systems, Inc.
         </copy-right-info>
        <build-time>19:18:58 Jun 23, 2012</build-time>
        <firmware-full-version>v4.0.0_bldg56</firmware-full-version>
        <control-processor-vendor>Freescale Semiconductor
         </control-processor-vendor>
        <control-processor-chipset>8548E</control-processor-chipset>
        <control-processor-memory>2000 MB</control-processor-memory>
        <node-info>
            <firmware-version-info>
                <primary-version>v4.0.0_bldg56</primary-version>
                <secondary-version>v4.0.0_bldg56</secondary-version>
            </firmware-version-info>
        </node-info>
    </show-firmware-version>
</rpc-reply>
```

Obtaining the firmware download status

Use the <dad-status> to obtain the firmware download status.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <dad-status xmlns="urn:brocade.com:mgmt:brocade-firmware">
    </dad-status>
</rpc>

<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 Oct 25 21:01:12 GMT 2013</date-and-time-info>
            <message>DHCP Auto-deployment enabled.</message>
        </dad-status-entries>
    </dad-status>
</rpc-reply>
```

Firmware upgrade in Brocade VCS Fabric mode

As of Network OS v4.1.1 release, the firmware download operation supports local switch upgrades only. To upgrade the entire cluster, you must issue the firmware download operation on each switch separately. For each switch in the fabric, complete the firmware download on the current switch before initiating a firmware download on the next switch. This process minimizes traffic disruption between switches.

Issue the <fwdl-status> custom RPC to verify that the download process is complete, and then move on to the next switch.

Chapter

Administering Licenses

6

In this chapter

Licensing with NETCONF overview

This chapter describes NETCONF operations that install, activate, remove, and verify licences. Refer to the *Network OS Licensing guide* for more information.

Retrieving the switch license ID

The following example returns the license ID for all switches in the fabric. To return the licence ID of a specific switch, replace the <all/> element with an <rbridge-id> element containing the routing bridge ID.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="701">
   <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
       <nca:data>
           <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
               <license xmlns="urn:brocade.com:mgmt:brocade-license">
                   <id>
                       <all/>
                   </id>
               </license>
           </show>
       </nca:data>
    </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="701">
   <data>
        <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
           <license xmlns="urn:brocade.com:mgmt:brocade-license">
               <id>
                   <licenseid-list>
                       <licenseid-rbridge-id>2</licenseid-rbridge-id>
```

```
                    <license-id>10:00:00:05:33:54:C6:3E</license-id>
                </licenseid-list>
            </id>
        </license>
    </show>
  </data>
</rpc-reply>
```

Obtaining a license key

License upgrade orders are fulfilled either through a license activation paperpack, or by an e-mail message containing a transaction key and a link to the Brocade software portal. A device-specific license file is generated in the software portal when you enter the transaction key along with the switch license ID. Use the <show>/<license>/<id> action to obtain the switch license ID.

Follow the instructions in the paperpack or the e-mail message as described for your platform and license type. The transaction key and the associated license string when the transaction key is exercised/redeemed is case-sensitive; you must enter the key exactly as it appears in the paperpack. To lessen the chance of an error, copy and paste the transaction key when you redeem the license using the transaction key and also when you install the license on your switch.

You will receive an e-mail message with the software license keys embedded in an XML file along with installation instructions.

NOTE
Store the license key in a safe place for future reference. You cannot retrieve the license key from the configuration datastore.

Installing or removing a license

Refer to the *Network OS Software Licensing Guide* for procedures for installing and removing licenses. You cannot install or remove licenses using the NETCONF interface.

Activating the Dynamic POD feature

The Dynamic POD feature is used with both the POD licenses as well as the 10G and 40G Port Upgrade licenses. To activate the Dynamic POD feature, complete the following steps.

1. Verify the current states of the ports with the <get-interface-detail> custom RPC located in the urn:brocade.com:mgmt:brocade-interface-ext namespace.

   The <line-protocol-state> and <line-protocol-state-info> fields shown in bold typeface in the following example indicate whether a port is licensed.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="714">
    <get-interface-detail xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type xmlns="urn:brocade.com:mgmt:brocade-interface">
            TenGigabitEthernet</interface-type>
        <interface-name xmlns="urn:brocade.com:mgmt:brocade-interface">
            22/0/1</interface-name>
    </get-interface-detail>
</rpc>
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="714">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type xmlns="urn:brocade.com:mgmt:brocade-interface">
            TenGigabitEthernet</interface-type>
        <interface-name xmlns="urn:brocade.com:mgmt:brocade-interface">1/0/1
            </interface-name>
        <ifindex>27</ifindex>
        <mtu xmlns="urn:brocade.com:mgmt:brocade-interface">2500</mtu>
        <ip-mtu xmlns="urn:brocade.com:mgmt:brocade-interface">2500</ip-mtu>
        <if-name>1/0/1</if-name>
        <if-state
xmlns="urn:brocade.com:mgmt:brocade-interface">down</if-state>
        <line-protocol-state
xmlns="urn:brocade.com:mgmt:brocade-interface">down
            </line-protocol-state>
        <line-protocol-state-info>No DPOD License</line-protocol-state-info>
(output truncated)
```

2. Install the Brocade Dynamic POD or Port Upgrade license.

   For instructions on how to install a license, refer to "Installing or removing a license" on page 64.

3. Disable and re-enable the ports.

   Alternatively, you can disable and re-enable the chassis to activate ports.

4. Issue the <get-interface-detail> custom RPC again to verify the newly activated ports and port details.

Obtaining the Dynamic POD assignments

To display the Dynamic POD assignments, issue the <show>/<dpod> action located in the urn:brocade.com:mgmt:brocade-license namespace. The reply provides a summary of the POD license status.

In the following example, all 24 ports are licensed and potentially available. Currently, the three unassigned ports are disabled persistently, and therefore are not assigned to any Dynamic POD license port set.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="715">
   <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
       <data>
           <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
               <dpod xmlns="urn:brocade.com:mgmt:brocade-license">
                   <rbridge-id>2</rbridge-id>>
               </dpod>
           </show>
       </data>
   </action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="715">
   <data>
        <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
           <dpod xmlns="urn:brocade.com:mgmt:brocade-license">
               <show-dpod-list>
                   <showdpod-rbridge-id>2</showdpod-rbridge-id>
```

```
                    <dpod-details>
                        <dpod-ports>24</dpod-ports>
                        <dpod-licenses-installed>1</dpod-licenses-installed>
                        <num-ports-provisioned>24</num-ports-provisioned>
                        <num-ports-reserved></num-ports-reserved>
                        <num-ports-license-available>3</num-ports-license-available>
                    </dpod-details>
                </show-dpod-list>
            </dpod>
        </show>
    </data>
</rpc>
```

Overriding Dynamic POD assignments

You can override the automatic port license assignments by releasing Dynamic POD assignments from a port and by reserving an assignment for a specific port.

Reserving a port assignment

Reserving an assignment for a port assigns that port to a POD license regardless of whether the port is online or offline. Reserving assignments allocates the POD license to specified ports. This operation overrides automatic port assignments. The reserved assignment will not be available to other ports that come online. To reserve an assignment for a port, a free assignment must be available.

If all ports are assigned, select a port to release its POD assignment. Follow the instructions in "Releasing a port from a POD set" on page 68 to release a port from its POD assignment. Once the port is released, you can reuse the assignment for another port.

1. Select the port for which you want to reserve an assignment and issue the <edit-config> RPC to configure the <dpod> node to reserve the desired ports. Set the <port-id>/<operation> element for each port to "reserve".

   The following example reserves ports 5/0/10 and 5/0/11.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="716">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <dpod xmlns="urn:brocade.com:mgmt:brocade-license">
                <port-id>
                    <port-id>5/0/10</port-id>
                    <operation>reserve</operation>
                </port-id>
                <port-id>
                    <port-id>5/0/11</port-id>
                    <operation>reserve</operation>
                </port-id>
            </dpod>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="716">
    <ok/>
</rpc-reply>
```

NOTE
In Fabric Cluster mode, license reservations or removals do not persist across switch reboots and power cycles. To make them persistent, save the configuration changes by copying the running configuration to the startup configuration before you reboot the switch. In Logical Chassis mode, the license reservations are automatically saved in the cluster - you do not need to explicitly save them.

2. Issue the <bna-config-cmd> RPC to save the configuration changes to the startup configuration.

3. Issue the <reboot>/<fastboot> custom action located in the urn:brocade.com:mgmt:brocade-firmware namespace to reboot the switch.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="718">
    <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <data>
            <reboot xmlns="urn:brocade.com:mgmt:brocade-firmware">
                <fastboot/>
            </reboot>
        </data>
    </action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="718">
    <ok/>
</rpc-reply>
```

4. Issue the <get-config> RPC to retrieve the DPOD configuration for the ports you reserved in step 1 to verify that the ports are reserved.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="719">
    <get-config>
        <target>
            <running/>
        </target>
        <filter type="subtree">
            <dpod xmlns="urn:brocade.com:mgmt:brocade-license>
                <port-id>
                    <port-id>5/0/10</port-id>
                    <port-id>5/0/11</port-id>
                </port-id>
            </dpod>
        </filter>
    </get-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="719">
    <dpod xmlns="urn:brocade.com:mgmt:brocade-logical-switch">
        <port-id>
            <port-id>5/0/10</port-id>
            <operation>reserve</operation>
        </port-id>
        <port-id>
            <port-id>5/0/11</port-id>
```

```
            <operation>reserve</operation>
        </port-id>
    </dpod>
</rpc-reply>
```

Releasing a port from a POD set

Once a port has been assigned to a Dynamic POD license port set, it remains licensed (or “reserved”) until you remove the port from the port set. You remove a port from the port set by editing the DPOD configuration with the <edit-config> RPC. Releasing a port removes it from the Dynamic POD license port set; the port appears as unassigned until it comes back online.

To prevent a port from coming back online and taking a POD assignment, disable the port and save the running configuration. This action will disable the port persistently.

A port POD assignment can only be released if the port is currently offline. Shut the port down to disable the port or disable the switch if you plan to release multiple ports.

1. Issue the <edit-config> RPC to configure the <interface> node in the urn:brocade.com:mgmt:brocade-interface namespace to shut down the interface.

   The following example shuts down interface 1/0/10.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="720">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/10</name>
                    <shutdown/>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="720">
    <ok/>
</rpc-reply>
```

2. Issue the <edit-config> RPC to configure the <dpod> node located in the urn:brocade.com:mgmt:brocade-license namespace and set the <operation> element to “release”.

   The following example releases ports 5/0/10 and 5/0/11.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="721">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <dpod xmlns="urn:brocade.com:mgmt:brocade-license">
                <port-id>
```

```
                    <port-id>5/0/10</port-id>
                    <operation>release</operation>
                </port-id>
                <port-id>
                    <port-id>1/0/10</port-id>
                    <operation>release</operation>
                </port-id>
            </dpod>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="721">
    <ok/>
</rpc-reply>
```

3. Issue the <edit-config> RPC to configure the <interface>/<tengigabitethernet> node in the urn:brocade.com:mgmt:brocade-interface namespace and include the following elements to re-enable the port:

   a. In the <name> element, specify the port name in [*rbridge-id/*]*slot/port* format.

   b. In the <shutdown> element, include the delete operation in the element tag to enable the port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="722">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/10</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete"/>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="722">
    <ok/>
</rpc-reply>
```

4. Issue the <bna-config-cmd> RPC to save the configuration changes by copying the running configuration to the startup configuration.

NOTE
Do not release a port unless you plan to disconnect the optical link or disable the port persistently. If you leave the link in a state where the port could be brought online, the POD mechanism will detect this unassigned port and attempt to reassign it to a port set.

Chapter

SNMP

7

In this chapter

SNMP management with NETCONF overview

This chapter provides procedures and examples for Brocade SNMP management using the NETCONF interface. Refer to the *Network OS Administrator's Guide* for the following related information:

- A conceptual overview of concepts related to SNMP
- Guidelines and restrictions regarding SNMP
- How to perform SNMP management using the Network OS command line interface

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

- Use the <edit-config> remote procedure call (RPC) to perform the following operations:
  - Enable and disable SNMP community strings
  - Configure SNMP server hosts
  - Enable and disable fabric trunking
  - Configure multiple SNMP server contexts
  - Configure the VCS Fabric virtual IP address
  - Configure password encryption for SNMPv3 users
- Use the <get-config> RPC to verify all or part of the SNMP configuration.
- Use the <show-vcs> custom RPC to return configuration state information about SNMP.

Brocade SNMP parameters are defined in the brocade-snmp YANG module. For structural maps of this YANG module, refer to the *Network OS YANG Reference Manual*.

Adding an SNMP community string

The following example adds a community string named “private” to the SNMP configuration and sets its groupname to group1.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp" >
                <community>
                    <community>private</community>
                    <groupname>group1</groupname>
                </community>
            </snmp-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Removing an SNMP community string

The following example removes the community string named “private” from the SNMP configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id=”1003" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
                <community xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                operation="delete">
                    <community>private</community>
                </community>
            </snmp-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1003" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring SNMP user

The following example shows how to configure an user in SNMP configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<rpc message-id="1002" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
                <user>
                    <username>snmp</username>
                </user>
                <user>
                    <username>snmpadmin2</username>
                    <groupname>snmpadmin</groupname>
                </user>
                <user>
                    <username>snmpadmin3</username>
                    <groupname>snmpadmin</groupname>
                </user>
                <user>
                    <username>snmpuser2</username>
                </user>
                <user>
                    <username>snmpuser3</username>
                    <auth>md5</md5>
                    <priv>DES</priv>
                </user>
            </snmp-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply
```

The following example shows how to configure an user in local RBridge in SNMP configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1002" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-snmp">
                <rbridge-id>2</rbridge-id>
                <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
                    <user>
                        <username>snmp</username>
                    </user>
                    <user>
                        <username>snmpadmin2</username>
                        <groupname>snmpadmin</groupname>
                    </user>
                </snmp-server>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="1001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply
```

Obtaining SNMP user names

To obtain the configured user names, issue the <get-config> RPC and provide a subtree filter to return only the <snmp-server>/<user> node information from the urn:brocade.com:mgmt:brocade-snmp namespace in the running configuration. For each configured SNMPv3 user name, the reply message contains a <user> node containing the user name and any assigned group, authentication protocol, or privacy protocol. If an authentication password or privacy password has been assigned, the encrypted password is returned in the corresponding leaf element with the encrypted flag.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1006" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
                <user/>
            </snmp-server>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="1006" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
        <user>
            <username>snmp</username>
        </user>
        <user>
            <username>snmpadmin2</username>
            <groupname>snmpadmin</groupname>
        </user>
        <user>
            <username>snmpadmin3</username>
            <groupname>snmpadmin</groupname>
        </user>
        <user>
            <username>snmpuser2</username>
        </user>
        <user>
            <username>snmpuser3</username>
            <auth>md5</md5>
            <priv>DES</priv>
        </user>
    </snmp-server>
</rpc-reply>
```

Setting the SNMP version 1 or 2c server host

The following example sets up “commaccess” as a read-only user and specifies the SNMP version 2c host using an IPv6 address. The host will receive SNMP traps on target port 162.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1007" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp" >
                <host>
                    <ip>1050:0:0:0:5:600:300c:326b</ip>
                    <community>commaccess</community>
                    <version>2c</version>
                    <udp-port>161</udp-port>
                    <severity-level>Info</severity-level>
                    <use-vrf>mgmt-vrf</use-vrf>
                </host>
            </snmp-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1007" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Setting the SNMP version 3 host

The following example configures dns1.mycorp.com as an SNMPv3 host and associates the snmpuser3 version 3 user with it. In this case, a DNS address is used to identify the server.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1008" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
                <v3host>
                    <hostip>dns1.mycorp.com</hostip>
                    <username>user1</username>
                    <udp-port>160</udp-port>
                    <notifytype>informs</notifytype>
                    <engineid>00:00:00:00:00:00</engineid>
                    <severity-level>Info</severity-level>
                    <use-vrf>mgmt-vrf</use-vrf>
                </host>
            </snmp-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1008" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example configures dns2.mycorp.com as an SNMPv3 host in rbridge-id 1 and associates the snmpuser4 version 3 user with it. In this case, a DNS address is used to identify the server.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
                    <v3host>
                        <hostip>dns2.mycorp.com</hostip>
                        <username>snmpuser4</username>
                        <udp-port>160</udp-port>
                        <notifytype>informs</notifytype>
                        <engineid>00:00:00:00:00:00</engineid>
                        <severity-level>Info</severity-level>
                        <use-vrf>mgmt-vrf</use-vrf>
                    </v3host>
                </snmp-server>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Removing the SNMP server host

The following example shows how to remove version 2c from the host, deleting the version restores the default value of version 1.

```
<rpc message-id="1009" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
                <host>
                    <ip>10.32.147.6</ip>
                    <community>public</community>
                    <version xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete">2c</version>
                </host>
            </snmp-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1009" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To remove the SNMP host from the switch configuration altogether, place the delete operation in the opening tag of the <host> element.

Setting the SNMP server contact

The default contact string is *Field Support*.

The following example changes the default contact string to “Operator 12345.”

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp" >
                <agtconfig>
                    <contact>Operator 12345</contact>
                </agtconfig>
            </snmp-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Setting the SNMP server location

The default server location string is *End User Premise*.

The following example changes the server location string to “Building 3 Room 214”.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1012" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp" >
                <agtconfig>
                    <location>Building 3 Room 214</location>
                </agtconfig>
            </snmp server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1012" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Returning the SNMP configuration

To display the current SNMP configuration for the SNMP host, community strings, user names, contact, and location, issue the <get-config> RPC and provide a subtree filter to return the <snmp-server> node from the urn:brocade.com:mgmt:brocade-snmp workspace in the running configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="117" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp"/>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="117" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
        <community>
            <community>ConvergedNetwork</community>
            <access>ro</access>
        </community>
        <community>
            <community>OrigEquipMfg</community>
            <access>rw</access>
        </community>
        <community>
            <community>Secret COde</community>
            <access>rw</access>
        </community>
        <community>
            <community>common</community>
            <access>ro</access>
        </community>
        <community>
            <community>private</community>
            <access>rw</access>
        </community>
        <community>
            <community>public</community>
            <access>ro</access>
        </community>
        <user>
            <username>snmp</username>
        </user>
        <user>
            <username>snmpadmin2</username>
            <groupname>snmpadmin</groupname>
        </user>
        <user>
            <username>snmpadmin3</username>
            <groupname>snmpadmin</groupname>
        </user>
        <user>
            <username>snmpuser2</username>
        </user>
        <user>
            <username>snmpuser3</username>
            <auth>md5</md5>
            <priv>DES</priv>
        </user>
        <host>
            <ip>10.17.37.107</ip>
            <community>public</community>
```

```
        </host>
        <agtconfig>
            <contact>Field Support</contact>
            <location>End User Premise</location>
        </agtconfig>
    </snmp-server>
</rpc-reply>
```

Setting the SNMP server context

The following example shows how to map a context to the name of a Virtual Routing and Forwarding (VRF) instance.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp" >
                <context>
                    <context>mycontext</context>
                    <vrf-name>myvrf</vrf-name>
                </context>
            </snmp-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Support for password encryption for SNMPv3 users

For SNMPv3 user, the passwords for <auth-password> and <priv-password> are encrypted. You can configure either with plain text password or encrypted password. In both the cases, the passwords are shown as encrypted.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp" >
                <user>snmpadmin2<user/>
                <groupname>snmpadmin</groupname>
                <auth>md5</auth>
                <auth-password>MVb+360X3kcfBzug5Vo6dQ==\n</auth-password>
                <priv>DES</priv>
                <priv-password>ckJFoHbzVvhR0xFRPjsMTA==\n</priv-password>
                    </encrypted>
            </snmp-server>
        </config>
```

```
    </edit-config>
</rpc>

<rpc-reply message-id="1001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

NOTE
This process may not be successful where encrypted passwords are generated by third-party or open-source tools.

SNMP traps

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp" >
                <enable>
                    <trap>
                        <trap-flag></trap-flag>
                    </trap>
                </enable>
            </snmp-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

SNMP server engine ID

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
                    <engineID>
                        <local>10:20:30:40:50:60:70:80:90:10:30:12</local>
                    </engineID>
                </snmp-server>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="1010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

SNMP server view

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
                <view>
                    <viewname>all</viewname>
                    <mibtree>1</mibtree>
                    <mibtree-access>included</mibtree-access>
                </view>
            </snmp-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

SNMP server group

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <snmp-server xmlns="urn:brocade.com:mgmt:brocade-snmp">
                <group>
                    <group-name>user</group-name>
                    <group-version>v1</group-version>
                    <group-auth-mode>noauth</group-auth-mode>
                    <read>All</read>
                    <write>-</write>
                    <notify>All</notify>
                </group>
            </snmp-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Chapter

Fabric 8

In this chapter

Fabric management with NETCONF overview

This chapter provides procedures and examples for Brocade VCS Fabric management using the NETCONF interface. Refer to the *Network OS Administrator's Guide* for the following related information:

- A conceptual overview of concepts related to Brocade VCS Fabric technology, such as the Transparent Interconnection of Lots of Links (TRILL) protocol, routing bridges, neighbor discovery, trunks, and fabric formation
- How to perform fabric management using the Network OS command line interface

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

- Use the <edit-config> remote procedure call (RPC) to perform the following operations:
  - Enable and disable VCS Fabric mode
  - Enable and disable fabric ISL and configure long distance ISL ports
  - Enable and disable fabric trunking
  - Configure the routing bridge priority for broadcast, unknown unicast, and multicast forwarding
  - Configure the VCS Fabric virtual IP address
  - Perform ECMP load balancing

- Use the <get-config> RPC to verify all or part of the VCS Fabric management configuration.
- Use the <show-vcs> custom RPC to return configuration state information about the VCS Fabric.

Brocade VCS Fabric parameters are defined in the brocade-vcs YANG module. Fabric interface management parameters are defined in the brocade-fabric-service and brocade-fcoe YANG modules. For structural maps of these YANG modules, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all VCS Fabric management parameters, refer to the brocade-fabric-service.yang, and brocade-fcoe.yang files.

Enabling VCS Fabric mode

The following example RPC enables VCS Fabric mode.

```
<rpc message-id="1200" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <vcsmode xmlns="urn:brocade.com:mgmt:brocade-vcs">
                <vcs-mode/>
            </vcsmode>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1200" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling VCS Fabric mode

To disable VCS Fabric mode, include the delete operation in the <vcs-mode> leaf element tag.

```
rpc message-id="1201" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <vcsmode xmlns="urn:brocade.com:mgmt:brocade-vcs">
                <vcs-mode xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete"/>
            </vcsmode>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1201" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Retrieving the VCS Fabric configuration details

Use the <get-vcs-details> to retrieve the VCS Fabric configuration information.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <get-vcs-details xmlns="urn:brocade.com:mgmt:brocade-vcs">
    </get-vcs-details>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <vcs-details xmlns="urn:brocade.com:mgmt:brocade-vcs">
        <node-vcs-mode>true</node-vcs-mode>
        <local-switch-wwn>10:00:00:27:F8:54:4F:98</local-switch-wwn>
        <node-vcs-type>vcs-management-cluster</node-vcs-type>
        <node-vcs-id>1</node-vcs-id>
        <principal-switch-wwn>10:00:00:27:F8:54:4F:98</principal-switch-wwn>
        <co-ordinator-wwn>10:00:00:27:F8:54:4F:98</co-ordinator-wwn>
    </vcs-details>
</rpc-reply>
```

Enabling a fabric ISL

The following example enables fabric ISL on 10-Gigabit Ethernet port 1/0/2.

```
<rpc message-id="1202" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/2</name>
                    <fabric xmlns=”urn:brocade.com:mgmt:brocade-fcoe”>
                        <fabric-isl>
                            <fabric-isl-enable/>
                        </fabric-isl>
                    </fabric>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1202" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

NOTE
Using an XGIG analyzer between switches in a Brocade VCS Fabric may cause a signal detection timeout, causing the fabric ISL link to fail.

Enabling a fabric trunk

The following example enables trunking on 10-Gigabit Ethernet port 1/0/4.

```
<rpc message-id="1205" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/4</name>
                    <fabric xmlns=”urn:brocade.com:mgmt:brocade-fcoe”>
                        <fabric-trunk>
                            <fabric-trunk-enable/>
                        </fabric-trunk>
                    </fabric>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1205" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling a fabric trunk

Fabric trunking is enabled by default.

The following example disables fabric trunking on port 1/0/4.

```
<rpc message-id="1206" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/4</name>
                    <fabric xmlns=”urn:brocade.com:mgmt:brocade-fcoe”>
                        <fabric-trunk>
                            <fabric-trunk-enable
                             xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                             operation="delete"/>
                        </fabric-trunk>
                    </fabric>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1206" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
```

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

Broadcast, unknown unicast, and multicast forwarding

All switches in a Brocade VCS Fabric share a single multicast tree rooted at the routing bridge with the lowest RBridge ID (domain ID). All broadcast, unknown unicast, and multicast traffic between two edge routing bridges is forwarded on this multicast tree inside the Brocade VCS Fabric. The multicast tree includes all routing bridges in the Brocade VCS Fabric.

Multicast distribution tree-root selection

Network OS software supports the following distribution tree behaviors.

- The root of the distribution tree is the switch with the lowest RBridge ID. The automated selection process does not require any user intervention.
- Each switch in the cluster optionally carries a multicast root priority. This priority setting overrides the automatically-selected multicast root. In deployments where a multicast root is required to be a specific switch that does not have the lowest RBridge ID, then the priority setting on that switch can override the root selection. If there are two switches with the same priority, then the switch with the lower RBridge ID prevails.
- A backup multicast root is pre-selected, which is the switch with the next lowest RBridge ID. The backup multicast root is automatically selected by all switches should the current multicast root fail.

Priorities

As stated previously, the root of the tree is auto-selected as the switch with the lowest RBridge ID. For example, if you had a cluster with RBridge IDs 5, 6, 7, and 8, then 5 would be the root. If you then added RBridge ID 1 to this fabric, the tree would be re-calculated with 1 as the root.

In order to avoid this behavior, you can set a priority (default is 1). The highest priority overrides the lowest RBridge ID and becomes the root.

For example, to build a fabric with RBridge ID 7 or 8 as the root, set their priority to something higher than 1 (priority values are 1 through 255). If there is a tie in priority, the lower RBridge ID is still chosen. If RBridge ID 7 and 8 are both set to priority 1, 7 becomes the root.

Changing the priority

The following example sets the priority of routing bridge 12 to 10.

```
<rpc message-id="1207" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <fabric xmlns="urn:brocade.com:mgmt:brocade-fabric-service">
                <route>
                    <mcast>
                        <rbridge-id>
                            <rbridge-id>12</rbridge-id>
```

```
                    <priority>10</priority>
                </rbridge-id>
            </mcast>
        </route>
      </fabric>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1207" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Obtaining the running configuration

You can use the <get-config> RPC to return fabric route multicast configuration information. The configuration currently effective on the switch is referred to as the running configuration. Any configuration change you make while the switch is online is made to the running configuration. The running configuration is nonpersistent.

NOTE
To save configuration changes, you must save the running-config file to a file, or you can apply the changes by copying the running configuration to the startup configuration.

To obtain the route multicast configuration, issue the <get-config> RPC with a subtree filter that restricts the output to the contents of the <fabric>/</route>/<mcast> node, as show in the following example.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1208" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <fabric xmlns="urn:brocade.com:mgmt:brocade-fabric-service">
                <route>
                    <mcast/>
                </route>
            </fabric>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="1208" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <fabric xmlns="urn:brocade.com:mgmt:brocade-fabric-service">
        <route>
            <mcast>
                <rbridge-id>
                    <rbridge-id>12</rbridge-id>
                    <priority>10</priority>
                </rbridge-id>
                <rbridge-id>
                    <rbridge-id>13</rbridge-id>
                    <priority>1</priority>
                </rbridge-id>
            </mcast>
```

```
        </route>
    </fabric>
</rpc>
```

Configuring the VCS Fabric virtual IP address

The following example shows how to configure the Virtual IP address.

NOTE
The virtual IP address must be an IPv4 address.

```
<rpc message-id="1209" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <vcs xmlns="urn:brocade.com:mgmt:brocade-vcs">
                <virtual>
                    <ip>
                        <address>10.0.0.23</address>
                    </ip>
                </virtual>
            </vcs>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1209" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To see the currently configured virtual IP address, issue the <show-vcs> custom RPC located in the urn:brocade.com:mgmt:brocade-vcs namespace as shown in the following example.

```
<rpc message-id="1210" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <show-vcs xmlns="urn:brocade.com:mgmt:brocade-vcs"/>
</rpc>

<rpc-reply message-id="1210" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <principal-switch-wwn>10:00:00:05:33:52:9F:A0</principal-switch-wwn>
    <co-cordinator-wwn>10:00:00:05:33:4B:0B:8C</co-cordinator-wwn>
    <vcs-cluster-type-info>vcs-fabric-cluster</vcs-cluster-type-info>
    <vcs-guid>01234567890123456789012345678901</vcs-guid>
    <virtual-ip-address>10.37.36.218</virtual-ip-address>
(output truncated)
```

To remove the currently configured virtual IP address, issue the following <edit-config> RPC.

```
<rpc message-id="1211" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <vcs xmlns="urn:brocade.com:mgmt:brocade-vcs">
                <virtual>
                    <ip>
```

```
                        <address xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    </ip>
                </virtual>
            </vcs>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1211" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Virtual IP address configuration scenarios

A virtual IP address may be assigned to a switch whenever it is the principal switch in the cluster. Refer to the *Network OS Administrator's Guide* for a list and explanation of the configuration scenarios that may occur.

Fabric ECMP load balancing

The following example uses the Destination MAC address and VID-based load balancing flavor and swaps every fourth bit pair of the input fields before feeding them to the hash function.

```
<rpc message-id="1212" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>2</rbridge-id>
                <fabric xmlns="urn:brocade.com:mgmt:brocade-fabric-service">
                    <ecmp>
                        <ecmp-load-balance>dst-mac-vid</ecmp-load-balance>
                        <ecmp-load-balance-hash-swap>4</ecmp-load-balance-hash-swap>
                    </ecmp>
                </fabric>
            </rbridge>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1212" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To verify the hash field selection configuration and hash swap configuration, issue the <get-config> RPC with a subtree filter to return the <ecmp> configuration node for the desired routing bridge. The following example returns the <ecmp> configuration for routing bridge 2.

```
<rpc message-id="1213" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
```

```
            <rbridge xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>2</rbridge-id>
                <fabric xmlns="urn:brocade.com:mgmt:brocade-fabric-service">
                    <ecmp/>
                </fabric>
            </rbridge>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="1213" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <rbridge xmlns="urn:brocade.com:mgmt:brocade-rbridge">
        <rbridge-id>2</rbridge-id>
        <fabric xmlns="urn:brocade.com:mgmt:brocade-fabric-service">
            <ecmp>
                <ecmp-load-balance>dst-mac-vid</ecmp-load-balance>
                <ecmp-load-balance-hash-swap>4</ecmp-load-balance-hash-swap>
            </ecmp>
        </fabric>
    </rbridge>
</rpc-reply>
```

Enabling Duplicate WWN login mode

Th following section shows how to enable duplicate WWN login mode.

Enabling Duplicate WWN login mode for a particular node/switch in a fabric

The following example enables duplicate WWN login mode for a particular node/switch in a fabric

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <fabric xmlns="urn:brocade.com:mgmt:brocade-fabric-service">
                    <login-policy>
                        <duplicateWWN>
                            <precedence>new-login</precedence>
                        </duplicateWWN>
                    </login-policy>
                </fabric>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Metro VCS 9

In this chapter

Metro VCS configuration with NETCONF overview

Metro VCS allows you to interconnect different locations and form clusters of Data Centers (DCs) over long distance in order to provide Disaster Protection/Recovery and load sharing.

Refer to the *Network OS Administrator's Guide* for information on Metro VCS and for the following related information:

- Metro VCS using long distance ISL
- Metro VCS using standard ISL
- Metro VCS and distributed Ethernet VLAGs

Through the NETCONF interface, you can perform the following operations that affect the functioning of Metro VCS:

- Use the <edit-config> RPC to activate, configure, or deactivate the Metro VCS Fabric ISL.
- Use the <edit-config> RPC to activate, configure, or deactivate Metro VCS using long distance ISL on specific 10-Gigabit, 40-Gigabit, or Gigabit Ethernet interfaces.
- Use the <edit-config> RPC to activate, configure, or deactivate Metro VCS using standard ISL on specific 10-Gigabit, 40-Gigabit, or Gigabit Ethernet interfaces.
- Use the <edit-config> RPC to activate, configure, or deactivate Metro VCS and distributed Ethernet VLAGs on specific 10-Gigabit, 40-Gigabit, or Gigabit Ethernet interfaces.
- Use the <get-config> RPC to verify all or part of the global or per-port Metro VCS configuration.

Metro VCS must be enabled globally before it can be enabled on a specific interface.

Metro VCS parameters are defined in the brocade-vcs YANG module. For information about the brocade-vcs YANG module, refer to the *Network OS YANG Reference Manual*.

Configuring Metro VCS using the long-distance-isl element

Each long distance ISL port of a VCS must be connected to a long distance ISL port on the remote VCS.

The following example shows how to configure a long distance ISL port.

```
<rpc message-id="1203" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/2</name>
                    <long-distance-isl>5000</long-distance-isl>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1203" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling a fabric ISL

To disable a fabric ISL, issue the following RPC.

```
<rpc message-id="1204" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/2</name>
                    <fabric xmlns=”urn:brocade.com:mgmt:brocade-fcoe”>
                        <fabric-isl>
                            <fabric-isl-enable
                             xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                             operation="delete"/>
                        </fabric-isl>
                    </fabric>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1204" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring Metro VCS using standard ISL

In order to deploy Metro VCS using standard ISL, no configuration is required on the standard fabric ISL. The default configuration on the 10 Gbps interface allows ISL formation with other Brocade VDX switches in the same VCS Cluster automatically. BLDP negotiation takes place to form standard ISLs for distances up to 30 m. Refer to the *Network OS Administrator's Guide* for details.

Configuring vLAGs for distributed Ethernet Fabrics

The following example configures physical interface 11/0/2 to port channel 4.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1907" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
            <target>
                <running/>
            </target>
        <config>
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
            <tengigabitethernet>
                <name>11/0/2</name>
                <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                operation="delete"/>
                <switchport>
                    <basic/>
                        <channel-group>
                        <port-int>4</port-int>
                        <mode>active</mode>
                        <type>standard</type>
                        </channel-group>
                </switchport>
            </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1907" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

Retrieving Metro VCS configuration

To retrieve the switch-port/Layer 2 characteristics of the interfaces configured as switchport, use the <get-interface-switchport> custom RPC.

```
<rpc message-id="303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-interface-switchport xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
    </get-interface-switchport>
</rpc>

<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>195/2/1</interface-name>
        <interface-type>fortygigabitethernet</interface-type>
        <mode>access</mode>
```

```
        <fcoe-port-enabled>false</fcoe-port-enabled>
        <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>
```

Chapter

Administering Zones 10

In this chapter

Zoning with NETCONF overview

NETCONF interfaces in Network OS provide support for partitioning your network into logical groups of devices or zones for security and for relief from Registered State Change Notification (RSCN) storms. In addition to conventional zones supported on Network OS fabrics, logical SAN (LSAN) zones that span heterogeneous networks of Network OS switches and Fabric OS switches are also supported.

This chapter describes the NETCONF operations that can be performed on zone objects and provides examples. Refer to the *Network OS Administrator's Guide* for conceptual information and general operational guidelines about zones, including:

- General zoning concepts
- LSAN zones
- Explanations of terminology
- Recommended approaches to zoning
- Effects on zoning of adding switches to a fabric or merging fabrics
- Supported firmware and firmware upgrade and downgrade considerations

Through the NETCONF interface, you can perform the following operations on zones:

- Use the <edit-config> RPC to create and manage zone configurations.
- Use the <zoning> action to obtain zoning database size and enabled-configuration information.
- Use the <get-config> RPC to validate configuration settings.

Zoning parameters are defined in the brocade-zone YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Setting the default zoning mode

The following example shows how to set the default zoning mode and saves the modified zoning configuration to nonvolatile memory.

```
<rpc message-id="601" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
                <enabled-configuration>
                    <default-zone-access>noaccess</default-zone-access>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="601" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Viewing database size information

To retrieve database size information, issue the <show>/<zoning> custom action that resides in the urn:brocade.com:mgmt:brocade-zone namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="602">
    <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <data>
            <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <zoning xmlns="urn:brocade.com:mgmt:brocade-zone"/>
                    <operation-info></operation-info>
                </zoning>
            </show>
        </data>
    </action>
</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="602">
    <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
        <db-max>1045274</db-max>
        <db-avail>1043895</db-avail>
        <db-committed>367</db-committed>
        <db-transaction>373</db-transaction>
        <transaction-token>1</transaction-token>
        <last-zone-changed-timestamp>2011-11-16 16:54:31 GMT-7:00
        </last-zone-changed-timestamp>
        <last-zone-committed-timestamp>2011-11-16 16:23:44 GMT-7:0
        </last-zone-committed-timestamp>
    </zoning>
```

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

Zone aliases

A zone alias is user-defined name for a logical group of ports or WWNs. You can simplify the process of creating and managing zones by first specifying aliases for zone members. Aliases facilitate tracking and eliminate the need for long lists of individual zone member names. An alias can be a member of a zone, but it cannot be a member of a zoning configuration.

Creating an alias

1. Issue the <show>/<name-server>/<detail> custom action that resides in the urn:brocade.com:mgmt:brocade-nameserver namespace to list the WWNs of devices and targets available in the Brocade VCS Fabric. The available WWNs appear in the <name-server>/<name-server-portname> and <name-server>/<nameserver-nodename> fields in the reply message.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="603">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <name-server xmlns="urn:brocade.com:mgmt:brocade-nameserver">
                    <detail>
                        <rbridge-id>66</rbridge-id>
                    </detail>
                </name-server>
            </show>
        </nca:data>
    </nca:action>
</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="603">
    <name-server xmlns="urn:brocade.com:mgmt:brocade-nameserver">
        <nameserver-portid>013100</nameserver-portid>
        <nameserver-portname>20:00:00:00:00:00:00:01</nameserver-portname>
        <nameserver-nodename>10:00:00:05:00:00:00:01</nameserver-nodename>
(output truncated)
```

2. The following example shows how to configure an alias.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="604" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration>
                    <alias>
                        <alias-name>alias1</alias-name>
                        <member-entry>
                            <alias-entry-name>10:00:00:00:00:00:00:01
                            </alias-entry-name>
                        </member-entry>
```

```
                </alias>
            </defined-configuration>
            <enabled-configuration>
                <cfg-action>cfg-save</cfg-action>
            </enabled-configuration>
        </zoning>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="604" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Adding additional members to an existing alias

1. Issue the <show>/<name-server>/<detail> custom action mechanism that resides in the urn:brocade.com:mgmt:brocade-nameserver namespace to list the WWNs of devices and targets available in the Brocade VCS Fabric. The available WWNs appear in the <name-server>/<name-server-portname> and <name-server>/<nameserver-nodename> fields in the reply message.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="605">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <name-server xmlns="urn:brocade.com:mgmt:brocade-nameserver">
                    <detail>
                        <rbridge-id>66</rbridge-id>
                    </detail>
                </name-server>
            </show>
        </nca:data>
    </nca:action>
</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="605">
    <name-server xmlns="urn:brocade.com:mgmt:brocade-nameserver">
        <nameserver-portid>013200</nameserver-portid>
        <nameserver-portname>20:00:00:00:00:00:00:02</nameserver-portname>
        <nameserver-nodename>10:00:00:05:00:00:00:02</nameserver-nodename>
(output truncated)
    <name-server xmlns="urn:brocade.com:mgmt:brocade-nameserver">
        <nameserver-portid>013300</nameserver-portid>
        <nameserver-portname>20:00:00:00:00:00:00:03</nameserver-portname>
        <nameserver-nodename>10:00:00:05:00:00:00:03</nameserver-nodename>
(output truncated)
```

2. The following example adds two member nodes to an alias.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="606" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
```

```
                <defined-configuration>
                    <alias>
                        <alias-name>alias1</alias-name>
                        <member-entry>
                            <alias-entry-name>10:00:00:00:00:00:00:02
                            </alias-entry-name>
                        </member-entry>
                        <member-entry>
                            <alias-entry-name>10:00:00:00:00:00:00:03
                            </alias-entry-name>
                        </member-entry>
                    </alias>
                </defined-configuration>
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="606" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Removing a member from an alias

1. Issue the <get-config> RPC to return the member information of the alias for which you want to remove a member. To limit the reply to a specific alias, use a subtree filter to view only the contents of the <zoning>/<defined-configuration>/<alias>/<alias-name> node.

   The following example returns the alias member information for the alias named alias1.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="607" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuation>
                    <alias>
                        <alias-name>alias1</alias-name>
                    </alias>
                </defined-configuration>
            </zoning>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="607" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <zoning xmlns="urn:brocade.com:mgmt:brocade-qos">
        <defined-configuration>
            <alias>
                <alias-name>alias1</alias-name>
                    <member-entry>
                        <alias-entry-name>10:00:00:00:00:00:00:01
                        </alias-entry-name>
```

```
                    </member-entry>
                    <member-entry>
                        <alias-entry-name>10:00:00:00:00:00:00:02
                        </alias-entry-name>
                    </member-entry>
                    <member-entry>
                        <alias-entry-name>10:00:00:00:00:00:00:03
                        </alias-entry-name>
                    </member-entry>
                </alias-name>
            </alias>
        </defined-configuration>
    </zoning>
</rpc-reply>
```

2. The following example removes WWNs 10:00:00:00:00:00:00:02 and 10:00:00:00:00:00:00:03 from alias1.

NOTE
When the last member of an alias is removed, the alias is also deleted.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="608" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration>
                    <alias>
                        <alias-name>alias1</alias-name>
                        <member-entry xmlns="urn:brocade.com:mgmt:brocade-zone"
                    xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete"/>
                            <alias-entry-name>10:00:00:00:00:00:00:02
                            </alias-entry-name>
                        </member-entry>
                        <member-entry xmlns="urn:brocade.com:mgmt:brocade-zone"
                    xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete"/>
                            <alias-entry-name>10:00:00:00:00:00:00:03
                            </alias-entry-name>
                        </member-entry>
                    </alias>
                </defined-configuration>
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="608" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting an alias

1. Issue the <get-config> RPC to return configuration information about zone aliases in the defined configuration. To limit the reply to alias information, use a subtree filter to view only the contents of the <zoning>/<defined-configuration>/<alias> node in the urn:brocade.com:mgmt:brocade-zone namespace.

   The following example returns zone alias information.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="609" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuation>
                    <alias/>
                </defined-configuration>
            </zoning>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="609" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
        <defined-configuration>
            <alias>
                <alias-name>alias1</alias-name>
                    <member-entry>
                        <alias-entry-name>10:00:00:00:00:00:00:01
                        </alias-entry-name>
                    </member-entry>
                    <member-entry>
                        <alias-entry-name>10:00:00:00:00:00:00:02
                        </alias-entry-name>
                    </member-entry>
                    <member-entry>
                        <alias-entry-name>10:00:00:00:00:00:00:03
                        </alias-entry-name>
                    </member-entry>
                </alias-name>
            </alias>
(output truncated)
```

2. The following example removes alias1 and saves the defined configuration to nonvolatile storage.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="610" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration>
                    <alias>
                        <alias-name
```

```
                        xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                        operation="delete">alias1</alias-name>
                    </alias>
                </defined-configuration>
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="610" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Zoning information

The following sections provide procedures for querying the defined zones configuration and the enabled zone configuration:

- "Retrieving the defined configuration" on page 104
- "Retrieving the enabled configuration" on page 106

Retrieving the defined configuration

To retrieve the entire defined configuration, use the following filter.

```
<filter type="subtree">
    <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
        <defined-configuration/>
    </zoning>
</filter>
```

To retrieve the zone membership details of a specific zone configuration, use a filter such as the following.

```
<filter type="subtree">
    <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
        <defined-configuration>
            <cfg>
                <cfg-name>cfg2</cfg-name>
            </cfg>
        </defined-configuration>
    </zoning>
</filter>
```

To retrieve the membership details of a specific zone, use a filter such as the following.

```
<filter type="subtree">
    <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
        <defined-configuration>
            <zone>
                <zone-name>ZoneB</zone-name>
            </zone>
        </defined-configuration>
    </zoning>
</filter>
```

To retrieve alias membership information, use a filter such as the following.

```
<filter type="subtree">
    <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
        <defined-configuration>
            <alias/>
        </defined-configuration>
    </zoning>
</filter>
```

The following example retrieves the entire defined configuration. For each configuration, the output lists each member zone. For each zone, the output lists the WWN of each member. In this case, the defined configuration contains two zoning configurations (cfg1 and cfg2). cfg1 contains three zones (zoneA, zoneB, and zoneC). cfg2 contains two zones (zoneA and zoneB). Each zone has two members. One device is shared among all three zones. This defined configuration contains no alias definitions.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="611" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration/>
            </zoning>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="611" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
        <defined-configuration>
            <cfg>
                <cfg-name>cfg1</cfg-name>
                <member-zone>
                    <zone-name>zoneA</zone-name>
                </member-zone>
                <member-zone>
                    <zone-name>zoneB</zone-name>
                </member-zone>
                <member-zone>
                    <zone-name>zoneC</zone-name>
                </member-zone>
            </cfg>
            <cfg>
                <cfg-name>cfg2</cfg-name>
                <member-zone>
                    <zone-name>zoneA</zone-name>
                </member-zone>
                <member-zone>
                    <zone-name>zoneB</zone-name>
                </member-zone>
            </cfg>
            <zone>
                <zone-name>zoneA</zone-name>
                <member-entry>
                    <entry-name>11:22:33:44:55:66:77:80</entry-name>
                </member-entry>
```

```
                <member-entry>
                    <entry-name>11:22:33:44:55:66:77:81</entry-name>
                </member-entry>
            </zone>
            <zone>
                <zone-name>zoneB</zone-name>
                <member-entry>
                    <entry-name>11:22:33:44:55:66:77:80</entry-name>
                </member-entry>
                <member-entry>
                    <entry-name>11:22:33:44:55:66:77:82</entry-name>
                </member-entry>
            </zone>
            <zone>
                <zone-name>zoneC</zone-name>
                <member-entry>
                    <entry-name>11:22:33:44:55:66:77:80</entry-name>
                </member-entry>
                <member-entry>
                    <entry-name>11:22:33:44:55:66:77:83</entry-name>
                </member-entry>
            </zone>
        </defined-configuration>
    </zoning>
</rpc-reply>
```

Retrieving the enabled configuration

The following example shows how to retrieve the entire enabled zone configuration.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <show-zoning-enabled-configuration
xmlns="urn:brocade.com:mgmt:brocade-zone"></show-zoning-enabled-configuration>
</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="1">
<enabled-configuration xmlns="urn:brocade.com:mgmt:brocade-zone">
    <cfg-name>cfg1</cfg-name>
        <enabled-zone>
            <zone-name>zone1</zone-name>
                <member-entry>
                    <entry-name>10:00:00:00:00:00:00:01</entry-name>
                </member-entry>
                <member-entry>
                    <entry-name>10:00:00:00:00:00:00:02</entry-name>
                </member-entry>
        </enabled-zone>
        <enabled-zone>
            <zone-name>zone2</zone-name>
                <member-entry>
                    <entry-name>10:00:00:00:00:00:00:03</entry-name>
                </member-entry>
                <member-entry>
                    <entry-name>10:00:00:00:00:00:00:04</entry-name>
                </member-entry>
        </enabled-zone>
    <has-more>false</has-more>
</enabled-configuration>
```

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

To retrieve a single zone and its underlying members, specify the zone-name-pattern field and issue the following:

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3">
   <show-zoning-enabled-configuration xmlns="urn:brocade.com:mgmt:brocade-zone">
      <zone-name-pattern>zone1</zone-name-pattern>
   </show-zoning-enabled-configuration>
</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="3">
   <enabled-configuration xmlns="urn:brocade.com:mgmt:brocade-zone">
      <cfg-name>cfg1</cfg-name>
      <enabled-zone>
         <zone-name>zone1</zone-name>
         <member-entry>
            <entry-name>10:00:00:00:00:00:00:01</entry-name>
         </member-entry>
      </enabled-zone>
      <has-more>false</has-more>
   </enabled-configuration>
</rpc-reply>
```

To support pagination, users must specify the last zone that was received and the zone plugin will return a block of the database immediately following the specified zone object. For example, for a zone configuration containing 200 zones and containing 1 WWN member each, the initial zone request procedure would look like the following:

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <show-zoning-enabled-configuration
xmlns="urn:brocade.com:mgmt:brocade-zone"></show-zoning-enabled-configuration>
</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="1">
   <enabled-configuration xmlns="urn:brocade.com:mgmt:brocade-zone">
      <cfg-name>cfg1</cfg-name>
      <enabled-zone>
         <zone-name>zone1</zone-name>
         <member-entry>
            <entry-name>10:00:00:00:00:00:00:01</entry-name>
         </member-entry>
      </enabled-zone>
..
      <enabled-zone>
         <zone-name>zone100</zone-name>
         <member-entry>
            <entry-name>10:00:00:00:00:00:00:64</entry-name>
         </member-entry>
      </enabled-zone>
      <has-more>true</has-more>
   </enabled-configuration>
</rpc-reply>
```

The next zone request would be:

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <show-zoning-enabled-configuration xmlns="urn:brocade.com:mgmt:brocade-zone">
      <last-rcvd-zone-name>zone100</last-rcvd-zone-name >
```

```
   </show-zoning-enabled-configuration>
</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="1">
   <enabled-configuration xmlns="urn:brocade.com:mgmt:brocade-zone">
      <cfg-name>cfg1</cfg-name>
      <enabled-zone>
         <zone-name>zone101</zone-name>
         <member-entry>
            <entry-name>10:00:00:00:00:00:00:65</entry-name>
         </member-entry>
      </enabled-zone>
..
      <enabled-zone>
         <zone-name>zone200</zone-name>
         <member-entry>
            <entry-name>10:00:00:00:00:00:00:C8</entry-name>
         </member-entry>
      </enabled-zone>
      <has-more>false</has-more>
   </enabled-configuration>
</rpc-reply>
```

Creating a zone

The following example adds a new zone to the defined configuration.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="613" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration>
                    <zone>
                        <zone-name>zoneD</zone-name>
                        <member-entry>
                            <entry-name>11:22:33:44:55:66:77:80</entry-name>
                        </member-entry>
                        <member-entry>
                            <entry-name>11:22:33:44:55:66:77:84</entry-name>
                        </member-entry>
                    </zone>
                </defined-configuration>
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="613" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Adding a member to a zone

The following example adds a WWN to an existing zone.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="614" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration>
                    <zone>
                        <zone-name>zoneD</zone-name>
                        <member-entry>
                            <entry-name>11:22:33:44:55:66:77:85</entry-name>
                        </member-entry>
                    </zone>
                </defined-configuration>
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="614" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Removing a member from a zone

The following example removes WWN 11:22:33:44:55:66:77:84 from zoneD.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="615" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration>
                    <zone>
                        <zone-name>zoneD</zone-name>
                        <member-entry
                            xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                            operation="delete">
                            <entry-name>11:22:33:44:55:66:77:84</entry-name>
                        </member-entry>
                    </zone>
                </defined-configuration>
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
```

```
    </edit-config>
</rpc>

<rpc-reply message-id="615" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting a zone

The following example deletes zoneD from the defined configuration and saves the defined configuration to nonvolatile storage.

```
<rpc message-id="616" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration>
                    <zone xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete">
                        <zone-name>zoneD</zone-name>
                    </zone>
                </defined-configuration>
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>
<rpc-reply message-id="616" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Zone configuration management

The following sections describe zoning configuration and management.

Creating a zone configuration

The following example creates cfg3 with zoneA and zoneD as member zones and saves the defined configuration to nonvolatile memory.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="617" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration>
                    <cfg>
                        <cfg-name>cfg3</cfg-name>
```

```
                    <member-zone>
                        <zone-name>zoneA</zone-name>
                    </member-zone>
                    <member-zone>
                        <zone-name>zoneD</zone-name>
                    </member-zone>
                </cfg>
            </defined-configuration>
            <enabled-configuration>
                <cfg-action>cfg-save</cfg-action>
            </enabled-configuration>
        </zoning>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="617" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

NOTE
Zone aliases are not valid zone configuration members. Adding an alias to an existing zone configuration will not be blocked. However, the attempt to enable a zone configuration that contains aliases will fail with an appropriate error message.

Adding a zone to a zone configuration

The following example adds zoneC to cfg3.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="618" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration>
                    <cfg>
                        <cfg-name>cfg3</cfg-name>
                        <member-zone>
                            <zone-name>zoneC</zone-name>
                        </member-zone>
                    </cfg>
                </defined-configuration>
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="618" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Removing a zone from a zone configuration

The following example removes zoneA from cfg3 and saves the configuration to nonvolatile memory.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="619" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration>
                    <cfg>
                        <cfg-name>cfg3</cfg-name>
                        <member-zone
                         xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete">
                            <zone-name>zoneA</zone-name>
                        </member-zone>
                    </cfg>
                </defined-configuration>
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="619" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling a zone configuration

Only one zone configuration can be enabled.

The following example enables cfg3.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="620" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <enabled-configuration>
                    <cfg-name>cfg3</cfg-name>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="620" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
```

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

If the configuration you are trying to enable contains a zone with no member, the server will return an <rpc-reply> with an error similar to the following message.

```
<rpc-reply message-id="620" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <rpc-error>
        <error-type>application</error-type>
        <error-tag>Command Failed</error-tag>
        <error-severity>error</error-severity>
        <error-message xml:lang="en">
            Cfg contains empty zone object "zoneB"
        </error-message>
    </rpc-error>
</rpc-reply>
```

Disabling a zone configuration

The following example disables the currently enabled configuration.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="621" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <enabled-configuration>
                    <cfg-name xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete">
                    </cfg-name>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="621" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting a zone configuration

The following example deletes cfg3 and saves the defined configuration to nonvolatile memory.

```
    <rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="622">
        <edit-config>
            <target>
                <running></running>
            </target>
            <config>
                <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                    <defined-configuration>
                        <cfg xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
    operation="delete">
                            <cfg-name>cfg1</cfg-name>
                        </cfg>
                    </defined-configuration>
```

```
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="622">
 <ok></ok>
</rpc-reply>

<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="623">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="623">
   <ok></ok>
</rpc-reply>
```

Clearing changes to a zone configuration

The following example removes all uncommitted operations from the zoning database.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="623" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <enabled-configuration>
                    <cfg-action>cfg-transaction-abort</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="623" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Clearing all enabled-zone configurations

The following example clears the zoning database and enables the default configuration.

```
<rpc message-id="624" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <edit-config>
        <target>
            <running/>
        </target>
    <config>
        <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
            <enabled-configuration>
                <cfg-action>cfg-clear</cfg-action>
            </enabled-configuration>
        </zoning>
    </config>
    </edit-config>
</rpc>

<rpc-reply message-id="624" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>

?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="625" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <cfg-name xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
operation="delete"/>
                </cfg-name>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="625" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Saving a copy of the zone configuration

The following example commits the zoning transaction and saves the defined configuration to a remote file.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="626" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <enabled-configuration>
                    <cfg-action>cfg-save</cfg-action>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="626" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>

<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>running-config</src>
      <dest>https://user@brocade.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:brocade.com:mgmt:brocade-ras">6</session-id>
    <status xmlns="urn:brocade.com:mgmt:brocade-ras">in-progress</status>
</rpc-reply>
```

Restoring a configuration from backup

The following example shows how to add a saved configuration to the running configuration.

```
<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>https://user@brocade.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:brocade.com:mgmt:brocade-ras">6</session-id>
    <status xmlns="urn:brocade.com:mgmt:brocade-ras">in-progress</status>
</rpc-reply>
```

Zone configuration scenario

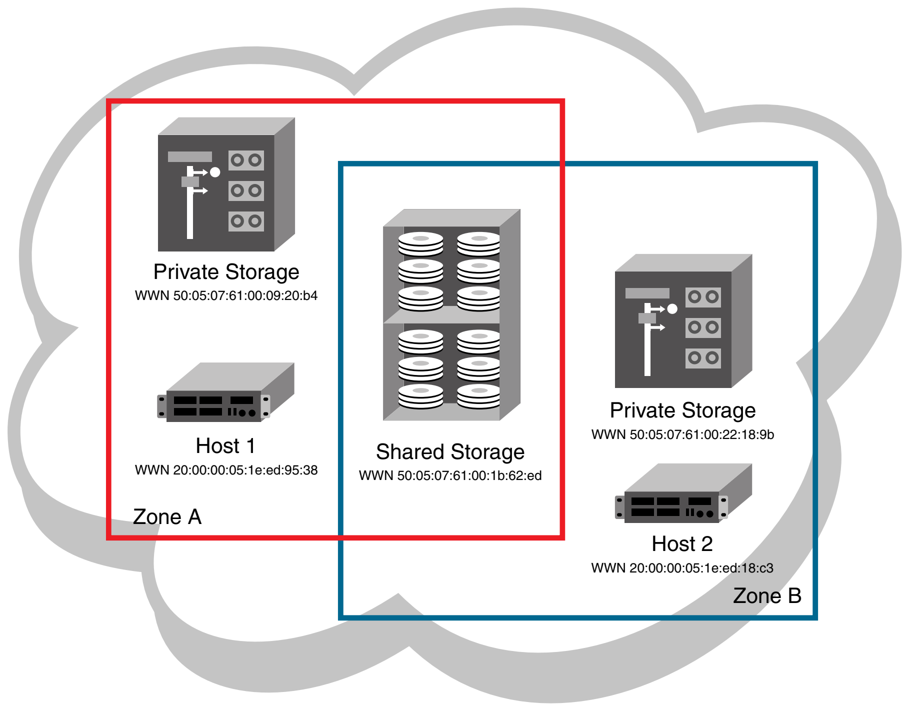


FIGURE 3 Zone configuration example

The following example creates the zone configuration shown in Figure 3. The example assumes that two hosts need access to the same storage device, while each host needs private storage of its own. You create two zones: Zone A contains Host 1, its private storage device, and the shared storage device; Zone B contains Host 2, its private storage device, and the shared storage device. In addition, you create two zone configurations: cfg1 in which only Zone A is effective; cfg2, in which both zones are effective.

The following example RPC retrieves available WWNs.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="629">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <name-server xmlns="urn:brocade.com:mgmt:brocade-nameserver">
                    <detail>
                        <rbridge-id>66</rbridge-id>
                    </detail>
                </name-server>
            </show>
        </nca:data>
    </nca:action>
</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="629">
    <name-server xmlns="urn:brocade.com:mgmt:brocade-nameserver">
        <nameserver-portid>016400</nameserver-portid>
        <nameserver-portname>10:00:00:05:1E:ED:95:38</nameserver-portname>
```

```
        <nameserver-nodename>20:00:00:05:1E:ED:95:38</nameserver-nodename>
(output truncated)
```

The following example RPC creates the defined configuration.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="630" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration>
                    <zone>
                        <zone-name>zoneA</zone-name>
                        <member-entry>
                            <entry-name>20:00:00:05:1e:ed:95:38</entry-name>
                        </member-entry>
                        <member-entry>
                            <entry-name>50:05:07:61:00:09:20:b4</entry-name>
                        </member-entry>
                        <member-entry>
                            <entry-name>50:05:07:61:00:1b:62:ed</entry-name>
                        </member-entry>
                    </zone>
                    <zone>
                        <zone-name>zoneB</zone-name>
                        <member-entry>
                            <entry-name>20:00:00:05:1e:ed:18:c3</entry-name>
                        </member-entry>
                        <member-entry>
                            <entry-name>50:05:07:61:00:22:18:9b</entry-name>
                        </member-entry>
                        <member-entry>
                            <entry-name>50:05:07:61:00:1b:62:ed</entry-name>
                        </member-entry>
                    </zone>
                    <cfg>
                        <cfg-name>cfg1</cfg-name>
                        <member-zone>
                            <zone-name>zoneA</zone-name>
                        </member-zone>
                    </cfg>
                    <cfg>
                        <cfg-name>cfg2</cfg-name>
                        <member-zone>
                            <zone-name>zoneA</zone-name>
                        </member-zone>
                        <member-zone>
                            <zone-name>zoneB</zone-name>
                        </member-zone>
                    </cfg>
                </defined-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="630" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <ok/>
</rpc-reply>
```

The following example RPC retrieves the defined configuration for verification.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="631" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <target>
            <running/>
        </target>
        <filter type="subtree">
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <defined-configuration/>
            </zoning>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="631" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
        <defined-configuration>
            <cfg>
                <cfg-name>cfg1</cfg-name>
                <member-zone>
                    <zone-name>zoneA</zone-name>
                </member-zone>
            </cfg>
            <cfg>
                <cfg-name>cfg2</cfg-name>
                <member-zone>
                    <zone-name>zoneA</zone-name>
                </member-zone>
                <member-zone>
                    <zone-name>zoneB</zone-name>
                </member-zone>
            </cfg>
            <zone>
                <zone-name>zoneA</zone-name>
                <member-entry>
                    <entry-name>20:00:00:05:1e:ed:95:38</entry-name>
                </member-entry>
                <member-entry>
                    <entry-name>50:05:07:61:00:09:20:b4</entry-name>
                </member-entry>
                <member-entry>
                    <entry-name>50:05:07:61:00:1b:62:ed</entry-name>
                </member-entry>
            </zone>
            <zone>
                <zone-name>zoneB</zone-name>
                <member-entry>
                    <entry-name>20:00:00:05:1e:ed:18:c3</entry-name>
                </member-entry>
                <member-entry>
                    <entry-name>50:05:07:61:00:22:18:9b</entry-name>
                </member-entry>
                <member-entry>
                    <entry-name>50:05:07:61:00:1b:62:ed</entry-name>
                </member-entry>
```

```
            </zone>
        </defined-configuration>
    </zoning>
</rpc-reply>
```

The following example RPC enables cfg2.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="632" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <zoning xmlns="urn:brocade.com:mgmt:brocade-zone">
                <enabled-configuration>
                    <cfg-name>cfg2</cfg-name>
                </enabled-configuration>
            </zoning>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="632" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Configuring Fibre Channel Ports 11

In this chapter

Fibre Channel ports configuration with NETCONF overview

This chapter provides procedures for configuring Fibre Channel ports using the NETCONF interface. Refer to the *Network OS Administrator's Guide* for related conceptual and overview information.

Through the NETCONF interface, you can perform the following operations on Fibre Channel ports:

- Use the <edit-config> RPC to activate and deactivate a Fibre Channel port and set the following port attributes:
  - desire-distance
  - fill-word
  - isl-r_rdy
  - long-distance
  - speed
  - trunk-enable
  - vc-link-init
- Use the <get-config> RPC to view all or part of the Fibre Channel port configuration.
- Use the <show-fibrechannel-interface> custom RPC to return operational information about the interface.

Fibre Channel port parameters are defined in the brocade-interface YANG module. The <show-fibrechannel-interface> RPC is defined in the brocade-fabric-service YANG module. For information about these YANG modules, refer to the *Network OS YANG Reference Manual*.

Retrieving the Fibre Channel port configuration

To retrieve the Fibre Channel configuration for all Fibre Channel interfaces configured on the switch, use the following filter:

```
<filter type="subtree">
   <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
```

```
        <fc-port/>
    </interface>
</filter>
```

To retrieve Fibre Channel configuration data for a specific Fibre Channel interface, use the following filter.

```
<filter type="subtree">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <fc-port>
            <name>8/0/1</name>
        </fc-port>
    </interface>
</filter>
```

To retrieve the settings of specific attributes for a given Fibre Channel port, use a filter such as the following. In this case, just the configured port speed is retrieved.

```
<filter type="subtree">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <fc-port>
            <name>8/0/1</name>
            <fc-speed-cfg/>
        </fc-port>
    </interface>
</filter>
```

The following example retrieves the configuration for the Fibre Channel port 1 on routing bridge 8.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1300" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fc-port>
                    <name>8/0/1</name>
                </fc-port>
            </interface>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="1300" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <fc-port>
            <name>8/0/1</name>
            <fc-speed-cfg>8gbps</fc-speed-cfg>
            <long-distance>ld</long-distance>
            <vc-link-init>arb</vc-link-init>
            <desire-distance>0</desire-distance>
            <trunk-enable></trunk-enable>
            <config-mode>4</config-mode>
        </fc-port>
    </interface>
</rpc-reply>
```

Enabling a Fibre Channel port

The following example shows how to enable a Fibre Channel port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1301" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fc-port>
                    <name>8/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete"/>
                </fc-port>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1301" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling a Fibre Channel port

The following example shows how to disable a Fibre Channel port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1302" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fc-port>
                    <name>8/0/1</name>
                    <shutdown/>
                </fc-port>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1302" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Setting Fibre Channel port speed

The following example shows how to set the speed of a Fibre Channel port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fc-port>
                    <name>8/0/1</name>
                    <fc-port-speed>4gbps</fc-port-speed>
                </fc-port>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring a Fibre Channel port for trunking

The following example shows how to enable a Fibre Channel port for trunking.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1305" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface>
                <fc-port>
                    <name>8/0/1</name>
                    <trunk-enable/>
                </fc-port>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1305" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Retrieving Fibre Channel interface information

The following example shows how to retrieve information about a fibre channel interface

NOTE
If you retrieve the <port-index> leaf on a slot without a line card, the value returned is 4294967295, which is actually -1. This leaf is an unsigned integer, and returns -1 when no card is found.

```
<rpc message-id="502" xmlns="urn:ietf:params:xml:ns:NETCONF:base:1.0">
    <show-fibrechannel-interface-info
     xmlns="urn:brocade.com:mgmt:brocade-fabric-service">
```

```
        <all/>
    </show-fibrechannel-interface-info>
</rpc>

<rpc-reply message-id="502" xmlns="urn:ietf:params:xml:ns:NETCONF:base:1.0">
    <show-fibrechennel-interface
     xmlns="urn:brocade.com:mgmt:brocade-fabric-service">
        <portsgroup-rbridgeid>24</portsgroup-rbridgid>
        <show-fibrechannel-info>
            <port-interface>24/0/1</port-interface>
            <port-index>1</port-index>
            <port-type>E-Port</port-type
            <port-wwn>20:79:00:05:33:67:26:78</port-wwn>
            <remote-port-wwn>20:79:00:05:22:58:26:73</remote-port-wwn>
            <remote-node-wwn>10:79:00:05:22:58:26:73</remote-node-wwn>
            <port-state>Online</port-state>
            <port-status>In_Sync</port-status>
            <port-status-message>trunk port</port-status-message>
            <port-health></port-health>
            <port-trunked>True</port-trunked>
            <port-trunk-master>0</port-trunk-master>
            <port-actual-distance>100</port-actual-distance>
            <port-desired-credit>10</port-desired-credit>
            <port-buffer-allocated>0</port-buffer-allocated>
            <port-licensed>True</port-licensed>
            <port-address>427900</port-address>
        </show-fibrechannel-info>
    </show-fibrechannel-interface>
</rpc-reply>
```

Chapter

Configuring Access Gateway 12

In this chapter

Access Gateway configuration with NETCONF overview

This chapter provides procedures for configuring Access Gateway ports using the NETCONF interface. Refer to the *Network OS Administrator's Guide* for related conceptual and overview information.

Through the NETCONF interface, you can perform the following operations on Access Gateway ports:

- Use the <edit-config> RPC to configure access gateway ports.
- Use the <get-config> RPC to view all or part of the Fibre Channel port configuration.
- Use the <show-fibrechannel-interface> custom RPC to return operational information about the interface.

Access Gateway parameters are defined in the brocade-ag YANG module. For information about these YANG modules, refer to the *Network OS YANG Reference Manual*.

Enabling Access Gateway mode

The following example shows how to enable Access Gateway mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
```

```
                <ag xmlns="urn:brocade.com:mgmt:brocade-ag">
                    <enable></enable>
                </ag>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring port mapping

The following example shows how to configure VF_Port to N_Port mapping:

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ag xmlns="urn:brocade.com:mgmt:brocade-ag">
                    <nport-menu>
                        <nport-interface>
                            <nport>
                                <agNPortNb>1/0/5</agNPortNb>
                                <map>
                                    <map-fport>
                                        <map-fport-interface></map-fport-interface>
                                    </map-fport>
                                </map>
                            </nport>
                        </nport-interface>
                    </nport-menu>
                </ag>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Creating port groups

The following example shows how to create a port group:

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
```

```
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ag xmlns="urn:brocade.com:mgmt:brocade-ag">
                    <pg>
                        <pgid>0</pgid>
                    </pg>
                </ag>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Naming a port group

The following example shows how to rename a port group:

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ag xmlns="urn:brocade.com:mgmt:brocade-ag">
                    <pg>
                        <pgid>0</pgid>
                        <rename>pg0</rename>
                    </pg>
                </ag>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Adding N_Ports in a port group

The following example shows how to configure n_ports in a port group:

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ag xmlns="urn:brocade.com:mgmt:brocade-ag">
                    <pg>
```

```
                        <pgid>0</pgid>
                        <nport>
                            <pg-nport-interface>
                                <nports>1/0/5</nports>
                            </pg-nport-interface>
                        </nport>
                    </pg>
                </ag>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling Login Balancing mode

The following example shows how to enable login balancing mode:

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ag xmlns="urn:brocade.com:mgmt:brocade-ag">
                    <pg>
                        <pgid>1</pgid>
                        <modes>lb</modes>
                    </pg>
                </ag>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Setting and displaying the fabric name monitoring TOV

The following example shows how to set the time out value (TOV):

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
```

```
                <ag xmlns="urn:brocade.com:mgmt:brocade-ag">
                    <timeout>
                        <fnmtovalue>120</fnmtovalue>
                    </timeout>
                </ag>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Setting and displaying the reliability counter for N_Port monitoring

The following example shows how to set the reliability count of static change notifications:

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ag xmlns="urn:brocade.com:mgmt:brocade-ag">
                    <counter>
                        <reliabilitycountervalue>11</reliabilitycountervalue>
                    </counter>
                </ag>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

System Monitor Configuration 13

In this chapter

System Monitor configuration with NETCONF overview

System Monitor provides customizable monitoring thresholds, which allow you to monitor the health of each fan, power supply, temperature sensor, CID card, small form-factor pluggable (SFP) device, management module (MM), line card, switch fabric module (SFM), or compact flash of the switch.

This chapter provides procedures for configuring System Monitor using NETCONF operations. Refer to the *Network OS Administrator's Guide* for the following related information:

- An overview of System Monitor
- An explanation and lists of default FRU threshold settings for each supported hardware platform
- An overview of resource monitoring
- SFP thresholds values, including defaults per SFP type
- An overview of monitored interface error types and related concepts
- Procedures for configuring System Monitor with the Network OS command line interface (CLI)

Through the NETCONF interface, you can perform the following operations for configuring System Monitor:

- Use the <edit-config> RPC to configure thresholds and notifications.
- Use the <get-config> RPC to validate configuration settings.
- Use the <show-system-monitor> custom RPC to obtain the health status of the switch.
- Use the <sfp>, <interface>, and <security> custom actions located in the <threshold> node of the urn:brocade.com:mgmt:brocade-threshold-monitor-ext namespace to display SFP, interface, and security default monitoring settings.

System Monitor parameters are defined in the brocade-system-monitor, brocade-system-monitor-ext, brocade-threshold-monitor, and brocade-threshold-monitor-ext YANG modules. For an overview and structural map of the YANG modules, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all user management parameters, refer to the brocade-system-monitor.yang, brocade-system-monitor-ext.yang, brocade-threshold-monitor.yang, and brocade-threshold-monitor-ext.yang files.

FRU monitoring

System Monitor monitors the absolute state of the fans, power supplies, CID card, line cards, and SFPs. For a description of possible states, and hardware platform default threshold settings, refer to the *Network OS Administrator's Guide*.

Setting system thresholds

The following example shows how to set system threshold values.

```
<rpc message-id="1100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <system-monitor xmlns="urn:brocade.com:mgmt:brocade-system-monitor">
                <fan>
                    <threshold>
                        <marginal-threshold>1</down-threshold>
                        <down-threshold>2</down-threshold>
                    </threshold>
                </fan>
                <power>
                    <threshold>
                        <marginal-threshold>1</down-threshold>
                        <down-threshold>2</down-threshold>
                    </threshold>
                </power>
                <temp>
                    <threshold>
                        <marginal-threshold>1</down-threshold>
                        <down-threshold>2</down-threshold>
                    </threshold>
                </temp>
                <cid-card>
                    <threshold>
                        <marginal-threshold>1</down-threshold>
                        <down-threshold>0</down-threshold>
                    </threshold>
                </cid-card>
                <compact-flash>
                    <threshold>
                        <marginal-threshold>1</down-threshold>
                        <down-threshold>0</down-threshold>
                    </threshold>
                </compact-flash>
                <MM>
                    <threshold>
                        <marginal-threshold>1</down-threshold>
                        <down-threshold>0</down-threshold>
                    </threshold>
                </MM>
                <LineCard>
                    <threshold>
                        <marginal-threshold>1</down-threshold>
                        <down-threshold>0</down-threshold>
                    </threshold>
```

```
                </LineCard>
                <SFM>
                    <threshold>
                        <marginal-threshold>1</down-threshold>
                        <down-threshold>0</down-threshold>
                    </threshold>
                </SFM>
            </system-monitor>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Setting FRU state alerts and actions

The following example shows how to set FRU state alerts and actions.

```
<rpc message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <system-monitor xmlns="urn:brocade.com:mgmt:brocade-system-monitor">
                <fan>
                    <alert>
                        <state>removed</state>
                        <action>raslog</action>
                    </alert>
                </fan>
                <power>
                    <alert>
                        <state>removed</state>
                        <action>raslog</action>
                    </alert>
                </power>
                <cid-card>
                    <alert>
                        <state>inserted faulty</state>
                        <action>email</action>
                    </alert>
                </cid-card>
                <sfp>
                    <alert>
                        <state>none</state>
                        <action>none</action>
                    </alert>
                </sfp>
                <LineCard>
                    <alert>
                        <state>removed</state>
                        <action>raslog</action>
                    </alert>
                </LineCard>
            </system-monitor>
        </config>
```

```
    </edit-config>
</rpc>

<rpc-reply message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Obtaining the switch health status

The following example shows how to obtain the switch health status.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1102">
    <show-system-monitor
        xmlns="urn:brocade.com:mgmt:brocade-system-monitor-ext">
        <rbridge-id>101</rbridge-id>
    </show-system-monitor>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1102">
    <switch-status xmlns="urn:brocade.com:mgmt:brocade-system-monitor-ext">
        <rbridge-id-out>101</rbridge-id-out>
        <switch-name>prodSwitchB</switch-name>
        <switch-ip>154.56.1.0</switch-ip>
        <report-time></report-time>
        <switch-state>state-healthy</switch-state>
        <switch-state-reason></switch-state-reason>
        <component-status>
            <component-name>fan</component-name>
            <component-state>state-marginal</component-state>
        </component-status>
        <component-status>
(output truncated)
```

Obtaining the system monitoring configuration

The following example shows how to retrieve the system monitor configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1103" 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">
            </system-monitor>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="1103" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <system-monitor xmlns="urn:brocade.com:mgmt:brocade-system-monitor">
        <fan>
            <threshold>
                <marginal-threshold>1</down-threshold>
                <down-threshold>2</down-threshold>
            </threshold>
            <alert>
```

```
                <state>1</state>
                <action>2</action>
            </alert>
        </fan>
        <power>
            <threshold>
                <marginal-threshold>1</down-threshold>
                <down-threshold>2</down-threshold>
            </threshold>
            <alert>
                <state>1</state>
                <action>2</action>
            </alert>
        </power>
        <temp>
            <threshold>
                <marginal-threshold>1</down-threshold>
                <down-threshold>2</down-threshold>
            </threshold>
        </temp>
        <cid-card>
            <threshold>
                <marginal-threshold>1</down-threshold>
                <down-threshold>0</down-threshold>
            </threshold>
            <alert>
                <state>1</state>
                <action>0</action>
            </alert>
        </cid-card>
(output truncated)
```

Alert notifications

The processes in this section configure the alert notifications.

Configuring e-mail alerts

The following example shows how to configure e-mail recipients of FRU alerts.

```
<rpc message-id="1104" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <system-monitor-mail
             xmlns="urn:brocade.com:mgmt:brocade-system-monitor">
                <fru>
                    <enable>
                    <email-list>
                        <email>admin@customer.com</email>
                    </email-list>
                </fru>
            </system-monitor-email>
        </config>
    </edit-config>
```

```
</rpc>

<rpc-reply message-id="1104" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Forwarding e-mail messages to a relay server

This procedure allows the sendmail agent on the switch to resolve the domain name and forward all e-mail messages to a relay server.

To create a mapping:

```
<rpc message-id="1105" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <system-monitor-mail
             xmlns="urn:brocade.com:mgmt:brocade-system-monitor">
                <relay>
                    <host-ip>1.2.3.4</host-ip>
                    <domain-name>englab.brocade.com</domain-name>
                </relay>
            </system-monitor-email>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1105" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To delete the mapping:

```
<rpc message-id="1106" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <system-monitor-mail
             xmlns="urn:brocade.com:mgmt:brocade-system-monitor">
                <relay xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete">
                    <host-ip xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete">1.2.3.4</host-ip>
                </relay>
            </system-monitor-email>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1106" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To change the domain name:

```
<rpc message-id="1107" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <system-monitor-mail
             xmlns="urn:brocade.com:mgmt:brocade-system-monitor">
                <relay>
                    <host-ip>1.2.3.4</host-ip>
                    <domain-name>customer.com</domain-name>
                </relay>
            </system-monitor-email>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1107" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To delete the domain name and return to the default:

```
<rpc message-id="1108" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <system-monitor-mail
             xmlns="urn:brocade.com:mgmt:brocade-system-monitor">
                <relay xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete">
                    <host-ip>1.2.3.4</host-ip>
                    <domain-name xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete">customer.com</domain-name>
                </relay>
            </system-monitor-email>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1108" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Resource monitoring

For a conceptual overview of resource monitoring, refer to the *Network OS Administrator's Guide*.

Configuring memory monitoring

NOTE
E-mail is not a supported action for threshold monitoring.

The following example configures memory monitoring.

```
<rpc message-id="1109" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <threshold-monitor-hidden
             xmlns="urn:brocade.com:mgmt:brocade-threshold-monitor">
                <threshold-monitor>
                    <memory>
                        <poll>30</poll>
                        <retry>3</retry>
                        <limit>75<limit>
                        <high-limit>80</high-limit>
                        <low-limit>50</high-limit>
                        <actions>raslog</actions>
                    </memory>
                </threshold-monitor>
            </threshold-monitor-hidden>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1109" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring CPU monitoring

The following example configures CPU monitoring.

```
<rpc message-id="1110" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <threshold-monitor-hidden
             xmlns="urn:brocade.com:mgmt:brocade-threshold-monitor">
                <threshold-monitor>
                    <Cpu>
                        <poll>30</poll>
                        <retry>3</retry>
                        <limit>75<limit>
                        <actions>all</actions>
                    </Cpu>
                </threshold-monitor>
            </threshold-monitor-hidden>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1110" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Obtaining the threshold monitoring configuration

The following example returns the CPU threshold monitoring configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1111" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <threshold-monitor-hidden
             xmlns="urn:brocade.com:mgmt:brocade-system-monitor">
                <threshold-monitor>
                    <Cpu>
                </threshold-monitor>
            </threshhold-monitor-hidden>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="1111" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <threshold-monitor-hidden
     xmlns="urn:brocade.com:mgmt:brocade-system-monitor">
        <threashold-monitor>
            <Cpu>
                <poll>30</poll>
                <retry>3</retry>
                <limit>75</limit>
                <actions>all</actions>
            </Cpu>
        </threshold-monitor>
    </threshold-monitor-hidden>
</rpc-reply>
```

Displaying security monitoring default values

The following example shows how to display the default values of security threshold and alert options.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1116">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <defaults>
                    <threshold
                     xmlns="urn:brocade.com:mgmt:brocade-threshold-monitor-ext"/>
                        <security/>
                    </threshold>
                </defaults>
            </show>
        </nca:data>
    </nca:action>
</rpc>
```

Configuring security monitoring

The following example shows how to configure security monitoring on a switch.

The following example configures a security policy that generates a RASlog message when a high threshold value of 10 telnet violations is breached.

```
<rpc message-id="1117" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <threshold-monitor-hidden
             xmlns="urn:brocade.com:mgmt:brocade-threshold-monitor">
                <threshold-monitor>
                    <security>
                        <policy>
                            <sec_policy_name>cusotm</sec_policy_name>
                            <area>
                                <sec_area_value>telnet-violation</sec_area_value>
                                <timebase>hour</timebase>
                                <threshold>
                                    <sec-high-threshold>10</sec-high-threshold>
                                    <sec-buffer>3</sec-buffer>
                                </threshold>
                                <alert>
                                    <above>
                                        <sec-above-highthresh-action>raslog
                                         </sec-above-highthresh-action>
                                    </above>
                                </alert>
                            </area>
                        </policy>
                    </security>
                </threshold-monitor>
            </threshold-monitor-hidden>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1117" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Applying security monitoring policies

The following example shows how to toggle between default settings and saved custom configuration settings and to apply actions and thresholds separately.

```
switch(config)# threshold-monitor security apply custom
<rpc message-id="1118" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <threshold-monitor-hidden
             xmlns="urn:brocade.com:mgmt:brocade-threshold-monitor">
                <threshold-monitor>
                    <security>
                        <apply>custom</apply>
                    </security>
                </threshold-monitor>
            </threshold-monitor-hidden>
        </config>
```

```
        </edit-config>
    </rpc>

    <rpc-reply message-id="1118" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
       <ok/>
    </rpc-reply>
```

Displaying interface monitoring default values

The following example shows how to display the default values of Interface threshold and alert options.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1119">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <defaults>
                    <threshold
                     xmlns="urn:brocade.com:mgmt:brocade-threshold-monitor-ext"/>
                        <interface>
                            <type>Ethernet</type>
                    </threshold>
                </defaults>
            </show>
        </nca:data>
    </nca:action>
</rpc>
```

Configuring interface monitoring

The following example shows how to configure interface monitoring on a switch.

```
<rpc message-id="1120" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <threshold-monitor-hidden
             xmlns="urn:brocade.com:mgmt:brocade-threshold-monitor">
                <threshold-monitor>
                    <interface>
                        <policy>
                            <policy_name>custom</policy_name>
                            <area>
                                <type>Ethernet</type>
                                <area_value>IFG</area_value>
                                <threshold>
                                    <timebase_value>hour</timebase_value>
                                    <high-threshold>80</high-threshold>
                                    <low-threshold>10<low-threshold>
                                    <buffer>10</buffer>
                                </threshold>
                                <alert>
                                    <above>
                                        <above-highthresh-action>fence raslog
                                         </above-highthresh-action>
                                    </above>
```

```
                                </alert>
                            </area>
                        </policy>
                    </interface>
                </threshold-monitor>
            </threshold-monitor-hidden>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1120" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Applying interface monitoring policies

This procedure allows you to toggle between default settings and saved custom configuration settings and to apply actions and thresholds separately. For example, you can choose to use default threshold settings with a customized subset of available actions, or you can modify some of the threshold settings and use the default action settings.

```
<rpc message-id="1121" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <threshold-monitor-hidden
             xmlns="urn:brocade.com:mgmt:brocade-threshold-monitor">
                <threshold-monitor>
                    <interface>
                        <apply>custom</apply>
                    </interface>
                </threshold-monitor>
            </threshold-monitor-hidden>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1121" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Pausing interface monitoring

The following example shows how to pause the monitoring of all ports and retain the ability to resume port monitoring at a later time.

```
rpc message-id="1122" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <threshold-monitor-hidden
             xmlns="urn:brocade.com:mgmt:brocade-threshold-monitor">
                <threshold-monitor>
                    <interface>
                        <pause/>
```

```
                </interface>
            </threshold-monitor>
        </threshold-monitor-hidden>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="1122" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Continuing interface monitoring

The following example shows how to resume port monitoring.

```
rpc message-id="1123" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <threshold-monitor-hidden
             xmlns="urn:brocade.com:mgmt:brocade-threshold-monitor">
                <threshold-monitor>
                    <interface>
                        <pause xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    </interface>
                </threshold-monitor>
            </threshold-monitor-hidden>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1123" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

VMware vCenter 14

In this chapter

vCenter management with NETCONF overview

This chapter provides procedures and examples for Brocade VCS Fabric management using the NETCONF interface. Refer to the *Network OS Administrator's Guide* for the following related information:

- A conceptual overview of concepts related to Brocade vCenter technology
- Brocade vCenter guidelines and restrictions
- Brocade vCenter discovery principles
- How to perform Brocade vCenter management using the Network OS command line interface (CLI)

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

- Use the <edit-config> remote procedure call (RPC) to perform the following operations:
  - Enable and disable vCenter
  - Configure discovery time interval
  - Configure user-triggered vCenter discovery
- Use the <get-config> RPC to verify all or part of the vCenter configuration.
- Use the <show-vcs> custom RPC to return configuration state information about vCenter.

Brocade vCenter parameters are defined in the brocade-vswitch YANG module. For structural maps of this YANG module, refer to the *Network OS YANG Reference Manual*.

Configuring vCenter

The processes in this section configure vCenter.

Step 1: Enabling QoS

You must edit the network resource pool settings and set QoS priorities. Refer to the latest VMware vSphere Networking documentation.

Step 2: Enabling CDP/LLDP

In order for an Ethernet Fabric to detect the ESX/ESXi hosts, you must first enable Cisco Discovery Protocol (CDP) and Link Layer Discovery Protocol (LLDP) on all the virtual switches (vSwitches) and distributed vSwitches (dvSwitches) in the vCenter Inventory.

For more information, refer to the VMware KB article 1003885.

Enabling CDP/LLDP on vSwitches

Complete the following steps to enable CDP/LLDP on virtual switches (vSwitches).

1. Log in as root to the ESX/ESXi Host.
2. Use the following command to verify the current CDP/LLDP settings.

   ```
   [root@server root]# esxcfg-vswitch -b vSwitch1
   ```

3. Use the following command to enable CDP/LLDP for a given virtual switch. Possible values here are **advertise** or **both**.

   ```
   [root@server root]# esxcfg-vswitch -B both vSwitch1
   ```

Enabling CDP/LLDP on dvSwitches

Complete the following steps to enable CDP on distributed virtual switches (dvSwitches).

1. Connect to the vCenter server by using the vSphere Client.
2. In the vCenter Server home page, click **Networking**.
3. Right-click the distributed virtual switches (dvSwitches) and click **Edit Settings**.
4. Select **Advanced** under **Properties.**
5. Use the check box and the drop-down list to change the CDP/LLDP settings.

Step 3: Adding and activating vCenter

After enabling CDP on all the vSwitches and dvSwitches in the vCenter, the Network OS-side configuration is a two step process: adding the vCenter and activating the vCenter.

Adding vCenter

The following example enables vCenter globally.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
            <target>
                <running/>
            </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <vswitch xmlns="urn:brocade.com:mgmt:brocade-vswitch"/>
                        <vcenter>myvcenter</vcenter>
                        <url>https://10.2.2.2</url>
                        <name>user</name>
                        <password>pass</password>
```

```
                        <interval>4</interval>
                    </vcenter>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

NOTE
By default, the vCenter server accepts only HTTPS connection requests.

Activating the vCenter

After adding the vCenter, you must activate the configured vCenter instance.

NOTE
In VCS mode, you can configure the vCenter by using any node. Discovery is initiated by the primary node.

```
switch(config)# vcenter myvcenter activate
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
            <target>
                <running/>
            </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <vswitch xmlns="urn:brocade.com:mgmt:brocade-vswitch"/>
                        <vcenter>myvcenter</vcenter>
                        </activate>
                    </vcenter>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Immediately following first-time vCenter activation, Network OS starts the virtual asset discovery process.

When the discovery process completes, the status displays as “Success.” Network OS has performed all the necessary configurations needed for the vCenter Server. Network OS is now ready for CDP transmissions from the virtual switches to identify which ESX/ESXi host is connected to which physical interface in the Ethernet Fabric.

Step 4: Retrieving the discovered virtual assets

Use the <get-config> RPC to retrieve the current configuration data and operational state data. Refer to "Retrieving configuration data" on page 11 and "Retrieving operational data" on page 15 for detailed instructions.

Custom RPCs

- **get-vmpolicy-macaddr** - Use the <get-vmpolicy-macaddr> to retrieve the vnics/vmknics to portgroup to port-profile association.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <get-vmpolicy-macaddr xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <vcenter>VC6</vcenter>
   </get-vmpolicy-macaddr>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
<vmpolicy-macaddr xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <mac>00:21:5e:c6:0e:c8</mac>
      <datacenter>datacenter-4381</datacenter>
      <dvpg-nn>Management Network</dvpg-nn>
      <port-prof>auto_VC6_datacenter-4381_Management+Network</port-prof>
   </vmpolicy-macaddr>
   <vmpolicy-macaddr xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <mac>00:21:5e:c6:b6:ec</mac>
      <datacenter>datacenter-2</datacenter>
      <dvpg-nn>Management Network</dvpg-nn>
      <port-prof>auto_VC6_datacenter-2_Management+Network</port-prof>
   </vmpolicy-macaddr>
   <vmpolicy-macaddr xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <mac>00:50:56:62:ab:88</mac>
      <datacenter>datacenter-2</datacenter>
      <dvpg-nn>VMkernel</dvpg-nn>
      <port-prof>auto_VC6_datacenter-2_VMkernel</port-prof>
   </vmpolicy-macaddr>
   <vmpolicy-macaddr xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <mac>00:50:56:63:da:67</mac>
      <datacenter>datacenter-2</datacenter>
      <dvpg-nn>VMkernel</dvpg-nn>
      <port-prof>auto_VC6_datacenter-2_VMkernel</port-prof>
   </vmpolicy-macaddr>
   <instance-id xmlns="urn:brocade.com:mgmt:brocade-vswitch">0</instance-id>
   <has-more xmlns="urn:brocade.com:mgmt:brocade-vswitch">false</has-more>
</rpc-reply>
```

- **get-vnetwork-dvpgs** - Use the <get-vnetwork-dvpgs> to retrieve details of the discovered distributed virtual port-groups.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <get-vnetwork-dvpgs xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <vcenter>VC6</vcenter>
   </get-vnetwork-dvpgs>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <vnetwork-dvpgs xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <name>dvPortGroup</name>
      <datacenter>datacenter-2</datacenter>
```

```
        <dvs-nn>dvSwitch</dvs-nn>
        <vlan>0,</vlan>
    </vnetwork-dvpgs>
    <vnetwork-dvpgs xmlns="urn:brocade.com:mgmt:brocade-vswitch">
        <name>dvSwitch-DVUplinks-4504</name>
        <datacenter>datacenter-2</datacenter>
        <dvs-nn>dvSwitch</dvs-nn>
        <vlan>0-4094,</vlan>
    </vnetwork-dvpgs>
    <instance-id xmlns="urn:brocade.com:mgmt:brocade-vswitch">0</instance-id>
    <has-more xmlns="urn:brocade.com:mgmt:brocade-vswitch">false</has-more>
</rpc-reply>
```

- get-vnetwork-dvs - Use the <get-vnetwork-dvs> to retrieve the details of the discovered Distributed Virtual Switches.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <get-vnetwork-dvs xmlns="urn:brocade.com:mgmt:brocade-vswitch">
        <vcenter>VC6</vcenter>
    </get-vnetwork-dvs>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <vnetwork-dvs xmlns="urn:brocade.com:mgmt:brocade-vswitch">
        <name>dvSwitch</name>
        <datacenter>datacenter-2</datacenter>
        <host>ESX5-1-74.englab.brocade.com</host>
        <pnic>vmnic4</pnic>
        <interface-type>unknown</interface-type>
        <interface-name></interface-name>
    </vnetwork-dvs>
    <vnetwork-dvs xmlns="urn:brocade.com:mgmt:brocade-vswitch">
        <name>dvSwitch</name>
        <datacenter>datacenter-2</datacenter>
        <host>ESX5-1-74.englab.brocade.com</host>
        <pnic>vmnic5</pnic>
        <interface-type>unknown</interface-type>
        <interface-name></interface-name>
    </vnetwork-dvs>
    <vnetwork-dvs xmlns="urn:brocade.com:mgmt:brocade-vswitch">
        <name>dvSwitch</name>
        <datacenter>datacenter-2</datacenter>
        <host>ESX5-1-74.englab.brocade.com</host>
        <pnic>vmnic8</pnic>
        <interface-type>unknown</interface-type>
        <interface-name></interface-name>
    </vnetwork-dvs>
    <vnetwork-dvs xmlns="urn:brocade.com:mgmt:brocade-vswitch">
        <name>dvSwitch</name>
        <datacenter>datacenter-2</datacenter>
        <host>ESX5-1-74.englab.brocade.com</host>
        <pnic>vmnic9</pnic>
        <interface-type>unknown</interface-type>
        <interface-name></interface-name>
    </vnetwork-dvs>
    <instance-id xmlns="urn:brocade.com:mgmt:brocade-vswitch">0</instance-id>
    <has-more xmlns="urn:brocade.com:mgmt:brocade-vswitch">false</has-more>
</rpc-reply>
```

- **get-vnetwork-hosts** - Use the <get-vnetwork-hosts> to retrieve the details of the discovered hosts.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <get-vnetwork-hosts xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <vcenter>VC7</vcenter>
   </get-vnetwork-hosts>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <vnetwork-hosts xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <name>ESXi5-1-76.englab.brocade.com</name>
      <datacenter>datacenter-21</datacenter>
      <vmnic>vmnic0</vmnic>
      <mac>00:21:5e:c6:22:00</mac>
      <vswitch>vSwitch0</vswitch>
      <interface-type>unknown</interface-type>
      <interface-name></interface-name>
   </vnetwork-hosts>
   <vnetwork-hosts xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <name>ESXi5-1-76.englab.brocade.com</name>
      <datacenter>datacenter-21</datacenter>
      <vmnic>vmnic3</vmnic>
      <mac>00:05:33:26:43:db</mac>
      <vswitch>vSwitch2</vswitch>
      <interface-type>unknown</interface-type>
      <interface-name></interface-name>
   </vnetwork-hosts>
   <vnetwork-hosts xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <name>ESXi5-1-76.englab.brocade.com</name>
      <datacenter>datacenter-21</datacenter>
      <vmnic>vmnic4</vmnic>
      <mac>00:05:1e:b1:12:86</mac>
      <vswitch>vSwitch2</vswitch>
      <interface-type>unknown</interface-type>
      <interface-name></interface-name>
   </vnetwork-hosts>
   <instance-id xmlns="urn:brocade.com:mgmt:brocade-vswitch">0</instance-id>
   <has-more xmlns="urn:brocade.com:mgmt:brocade-vswitch">false</has-more>
</rpc-reply>
```

- **get-vnetwork-portgroups** - Use the <get-vnetwork-portgroups> to retrieve the details of the discovered port groups.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <get-vnetwork-portgroups xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <vcenter>VC7</vcenter>
   </get-vnetwork-portgroups>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <vnetwork-pgs xmlns="urn:brocade.com:mgmt:brocade-vswitch">
    <name>Management Network</name>
    <datacenter>datacenter-2</datacenter>
    <vlan>0</vlan>
    <host-nn>ESX5-0-72.englab.brocade.com</host-nn>
   </vnetwork-pgs>
   <vnetwork-pgs>
    <name>Management Network</name>
    <datacenter>datacenter-2</datacenter>
```

```
      <vlan>0</vlan>
      <host-nn>ESX5-1-74.englab.brocade.com</host-nn>
     </vnetwork-pgs>
     <vnetwork-pgs>
      <name>Management Network</name>
      <datacenter>datacenter-4381</datacenter>
      <vlan>0</vlan>
      <host-nn>ESX5-1-75.englab.brocade.com</host-nn>
     </vnetwork-pgs>
     <vnetwork-pgs>
      <name>Management Network</name>
      <datacenter>datacenter-4381</datacenter>
      <vlan>0</vlan>
      <host-nn>ESXi5-0-71.englab.brocade.com</host-nn>
     </vnetwork-pgs>
     <instance-id xmlns="urn:brocade.com:mgmt:brocade-vswitch">0</instance-id>
     <has-more xmlns="urn:brocade.com:mgmt:brocade-vswitch">false</has-more>
</rpc-reply>
```

- **get-vnetwork-vms** - Use the <get-vnetwork-vms> to retrieve the details of the discovered VMs.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <get-vnetwork-vms xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <vcenter>VC7</vcenter>
   </get-vnetwork-vms>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <vnetwork-vms xmlns="urn:brocade.com:mgmt:brocade-vswitch">
    <name>KVM_Hyperv_101_castor_castor</name>
    <datacenter>datacenter-2</datacenter>
    <mac>00:50:56:b3:5e:25</mac>
    <host-nn>ESX5-1-74.englab.brocade.com</host-nn>
  </vnetwork-vms>
  <vnetwork-vms>
    <name>KVM_Hyperv_101_castor_castor</name>
    <datacenter>datacenter-2</datacenter>
    <mac>00:50:56:b3:6b:19</mac>
    <host-nn>ESX5-1-74.englab.brocade.com</host-nn>
  </vnetwork-vms>
  <vnetwork-vms>
    <name>KVM_Hyperv_102_castor_nexus</name>
    <datacenter>datacenter-2</datacenter>
    <mac>00:50:56:b3:37:c6</mac>
    <host-nn>ESX5-1-74.englab.brocade.com</host-nn>
  </vnetwork-vms>
</rpc-reply>
```

- **get-vnetwork-vswitches** - Use the <get-vnetwork-vswitches> to retrieve the details of the discovered VSwitches.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" nc:message-id="1">
   <get-vnetwork-vswitches xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <vcenter>VC6</vcenter>
   </get-vnetwork-vswitches>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <vnetwork-vswitches xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <name>vSwitch0</name>
      <datacenter>datacenter-2</datacenter>
```

```
      <host>ESX5-0-72.englab.brocade.com</host>
      <pnic>vmnic0</pnic>
      <interface-type>unknown</interface-type>
      <interface-name></interface-name>
   </vnetwork-vswitches>
   <vnetwork-vswitches xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <name>vSwitch0</name>
      <datacenter>datacenter-2</datacenter>
      <host>ESX5-1-74.englab.brocade.com</host>
      <pnic>vmnic0</pnic>
      <interface-type>unknown</interface-type>
      <interface-name></interface-name>
   </vnetwork-vswitches>
   <vnetwork-vswitches xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <name>vSwitch0</name>
      <datacenter>datacenter-4381</datacenter>
      <host>ESX5-1-75.englab.brocade.com</host>
      <pnic>vmnic0</pnic>
      <interface-type>unknown</interface-type>
      <interface-name></interface-name>
   </vnetwork-vswitches>
   <vnetwork-vswitches xmlns="urn:brocade.com:mgmt:brocade-vswitch">
      <name>vSwitch0</name>
      <datacenter>datacenter-4381</datacenter>
      <host>ESX5-1-75.englab.brocade.com</host>
      <pnic>vmnic9</pnic>
      <interface-type>unknown</interface-type>
      <interface-name></interface-name>
   </vnetwork-vswitches>
</rpc-reply>
```

Chapter

Configuring Remote Monitoring 15

In this chapter

RMON configuration with NETCONF overview

This chapter provides procedures for configuring remote monitoring (RMON) events and alarms using the NETCONF interface. Refer to the *Network OS Administrator's Guide* for the following related information:

- A conceptual overview of RMON
- Procedures for configuring RMON using the Network OS command line interface (CLI)

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

- Use the <edit-config> remote procedure call (RPC) to configure RMON.
- Use the <get-config> RPC to verify all or part of the RMON configuration.

RMON parameters are defined in the brocade-rmon YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all RMON parameters, refer to the brocade-rmon.yang file.

RMON configuration and management

Alarms and events are configurable RMON parameters:

- Events—Determines the action to take when an event is triggered by an alarm. The action can be to generate a log entry, an SNMP trap, or both. You must define the events before an alarm can be configured. If you do not configure the RMON event first, you will receive an error when you configure the alarm settings.
- Alarms—Monitors a specific management information base (MIB) object for a specified interval, triggers an alarm at a specified value (rising threshold), and resets the alarm at another value (falling threshold). Alarms are paired with events; the alarm triggers an event, which can generate a log entry or an SNMP trap.

Default RMON configuration

By default, no RMON alarms and events are configured and RMON collection statistics are not enabled.

Configuring RMON alarm settings

The following example shows how to configure RMON alarms and events.

The following example shows an alarm that tests every sample for a rising threshold.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rmon xmlns="urn:brocade.com:mgmt:brocade-rmon">
                <alarm-entry>
                    <alarm-index>5</alarm-index>
                    <snmp-oid>1.3.6.1.2.1.16.1.1.1.5.65535</snmp-oid>
                    <alarm-interval>30</alarm-interval>
                    <alarm-sample>absolute</alarm-sample>
                    <alarm-rising-threshold>95</alarm-rising-threshold>
                    <alarm-rising-event-index>27</alarm-rising-event-index>
                    <alarm-owner>john_smith</alarm-owner>
                </alarm-entry>
            </rmon>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example shows an alarm that tests the delta between samples for a falling threshold.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rmon xmlns="urn:brocade.com:mgmt:brocade-rmon">
                <alarm-entry>
                    <alarm-index>5</alarm-index>
                    <snmp-oid>1.3.6.1.2.1.16.1.1.1.5.65535</snmp-oid>
                    <alarm-interval>10</alarm-interval
                    <alarm-sample>delta</alarm-sample>
                    <alarm-falling-threshold>65</alarm-falling-threshold>
                    <alarm-falling-event-index>42</alarm-falling-event-index>
                    <alarm-owner>john_smith</alarm-owner>
                </alarm-entry>
            </rmon>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Python Event-Management and Scripting 16

In this chapter

Python under Network OS

The Python interpreter installed with Network OS enables you to access a Python shell or to launch Python scripts. You can also define event handlers that run such scripts automatically upon specified conditions.

Python overview

Python is a high-level scripting language that also supports object-oriented programming.

Implementing Python event-management

Python event management enables you to specify a Python script that runs automatically upon specified conditions.

The conditions available for triggering a Python script are as follows:

- A specified RASlog message
- One of the following switch events:
  - Switch bootup
  - Completion of cluster formation
  - Completion of configuration-file replay

Configuring an event-handler profile

The following example shows how to configure and event-handler profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <event-handler xmlns="urn:brocade.com:mgmt:brocade-event-handler">
                <event-handler-list>
```

```
                    <name>eventHandler1</name>
                    <trigger>
                        <trigger-id>1</trigger-id>
                        <raslog>LOG-1001</raslog>
                    </trigger>
                    <action>
                        <python-script>example.py</python-script>
                    </action>
                </event-handler-list>
            </event-handler>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Activating an event-handler on a RBridge

The following example shows how to activate an event-handler profile on a RBridge.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <activate>
                    <event-handler
xmlns="urn:brocade.com:mgmt:brocade-event-handler">
                    <name>eventHandler1</name>
                </activate>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring event-handler options

The following example shows how to configure event-handler parameters.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
```

```
                <activate>
                    <event-handler
xmlns="urn:brocade.com:mgmt:brocade-event-handler">
                    <name>eventHandler1</name>
                    <delay>0</delay>
                    <iterations>1</iterations>
                    <interval>0</interval>
                    <run-mode>exclusive</run-mode>
                    <trigger-function>AND</trigger-function>
                    <trigger-mode>on-first-instance</trigger-mode>
                </activate>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Configuring DHCP Dynamic Auto Deployment 17

In this chapter

RMON configuration with NETCONF overview

This chapter provides procedures for configuring remote monitoring (RMON) events and alarms using the NETCONF interface. Refer to the *Network OS Administrator's Guide* for the following related information:

- A conceptual overview of RMON
- Procedures for configuring RMON using the Network OS command line interface (CLI)

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

- Use the <edit-config> remote procedure call (RPC) to configure RMON.
- Use the <get-config> RPC to verify all or part of the RMON configuration.

RMON parameters are defined in the brocade-rmon YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all RMON parameters, refer to the brocade-rmon.yang file.

RMON configuration and management

Alarms and events are configurable RMON parameters:

- Events—Determines the action to take when an event is triggered by an alarm. The action can be to generate a log entry, an SNMP trap, or both. You must define the events before an alarm can be configured. If you do not configure the RMON event first, you will receive an error when you configure the alarm settings.
- Alarms—Monitors a specific management information base (MIB) object for a specified interval, triggers an alarm at a specified value (rising threshold), and resets the alarm at another value (falling threshold). Alarms are paired with events; the alarm triggers an event, which can generate a log entry or an SNMP trap.

Default RMON configuration

By default, no RMON alarms and events are configured and RMON collection statistics are not enabled.

Configuring RMON alarm settings

The following example shows how to configure RMON alarms and events.

The following example shows an alarm that tests every sample for a rising threshold.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rmon xmlns="urn:brocade.com:mgmt:brocade-rmon">
                <alarm-entry>
                    <alarm-index>5</alarm-index>
                    <snmp-oid>1.3.6.1.2.1.16.1.1.1.5.65535</snmp-oid>
                    <alarm-interval>30</alarm-interval>
                    <alarm-sample>absolute</alarm-sample>
                    <alarm-rising-threshold>95</alarm-rising-threshold>
                    <alarm-rising-event-index>27</alarm-rising-event-index>
                    <alarm-owner>john_smith</alarm-owner>
                </alarm-entry>
            </rmon>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example shows an alarm that tests the delta between samples for a falling threshold.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rmon xmlns="urn:brocade.com:mgmt:brocade-rmon">
                <alarm-entry>
                    <alarm-index>5</alarm-index>
                    <snmp-oid>1.3.6.1.2.1.16.1.1.1.5.65535</snmp-oid>
                    <alarm-interval>10</alarm-interval
                    <alarm-sample>delta</alarm-sample>
                    <alarm-falling-threshold>65</alarm-falling-threshold>
                    <alarm-falling-event-index>42</alarm-falling-event-index>
                    <alarm-owner>john_smith</alarm-owner>
                </alarm-entry>
            </rmon>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Section

II

Network OS Security Configuration

This section describes security features, and includes the following chapters:

Chapter

Managing User Accounts 18

In this chapter

Managing user accounts with NETCONF overview

This chapter provides procedures for managing user accounts with the NETCONF interface. Refer to the *Network OS Security Configuration Guide* for the following related information:

- Related conceptual overview information
- Procedures and examples for managing user accounts using the Network OS command line interface (CLI)

Through the NETCONF interface, you can perform the following operations for managing user accounts:

- Use the <edit-config> RPC to configure user accounts, role-based access control, command access rules, and password policies.
- Use the <get-config> RPC to validate configuration settings.
- Use the <user>/<unlock> custom action to unlock a user account.

User management parameters are defined in the brocade-aaa YANG module. For an overview and structural map of the YANG module, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all user management parameters, refer to the brocade-aaa.yang file.

Creating a user account

The following example creates a new user account with the minimally required attributes: name, role, and password. The account name “brcdUser” has the default user privilege of accessing commands in the privileged EXEC mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <username xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <name>brcdUser</name>
                <role>user</role>
                <user-password>welcome</user-password>
            </username>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Verifying user account information

The user account information is saved in the switch configuration file.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="801" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <username xmlns="urn:brocade.com:mgmt:brocade-aaa">
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="801" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <username xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <name>brcdUser</name>
        <user-password>San5josE</user-password>
        <role>user</role>
    </username>
    <username xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <name>brcdUser2</name>
        <user-password>Broom6fielD</user-password>
        <role>user</role>
    </username>
    <username xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <name>brcdUser3</name>
        <user-password>Esoj3naS</user-password>
        <role>user</role>
    </username>
</rpc-reply>
```

Modifying an existing user account

The following example adds a description to the previously created “brcdUser” account.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="802" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <username xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <name>brcdUser</name>
                <desc>Brocade guest account</desc>
                <expire>never</expire>
            </username>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="802" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example changes the password for the account “testUser”. All active login sessions of a user are terminated if the user’s password or role is changed.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="803" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <username xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <name>testUser</name>
                <user-password>hellothere</user-password>
            </username>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="803" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling a user account

You can disable a user account by setting the enable parameter to “false”. All active login sessions for a user are terminated when a user account is disabled.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="804" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
```

```
            <username xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <name>testUser</name>
                <enable>false</enable>
            </username>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="804" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting a user account

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="805" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <username xmlns="urn:brocade.com:mgmt:brocade-aaa" operation="delete">
                <name>testUser</name>
            </username>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="805" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

All active login sessions for a user are terminated when a user account is deleted.

Obtaining user information

</user-session-info> returns user information.

```
<rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:message-id="7">
   <user-session-info
xmlns="urn:brocade.com:mgmt:brocade-aaa-ext"></user-session-info>
</rpc>

<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">
   <user-role xmlns="urn:brocade.com:mgmt:brocade-aaa-ext">admin</user-role>
</rpc-reply>
```

Unlocking a user account

The following example unlocks a locked user account.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="806">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
```

```
        <nca:data>
            <user xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <unlock>
                    <username>testUser</username>
                </unlock>
            </user>
        </nca:data>
    </nca:action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="806">
    <data>
        <user xmlns="urn:brocade.com:mgmt:brocade-ras">
            <unlock>
                <Result>Success</Result>
            </unlock>
        </user>
    </data>
    </rpc-reply>
```

Configuring a user alias

The global alias is accessible across all users. The user-level alias is accessible only when the respective user logs in.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <alias-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <alias>redwood</alias>
                <expansion>engineering</alias>
                <user>john smith</user>
                <expansion>manager</alias>
            </alias-config>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Creating or modifying a role

The following example creates a role named VLANAdmin and provides the description “Manages security.”

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="807" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <role xmlns="urn:brocade.com:mgmt:brocade-aaa">
```

```
                <name>
                    <name>VLANAdmin</name>
                    <desc>Manages security</desc>
                </name>
            </role>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="807" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Verifying a role configuration

To verify a role configuration, issue the <get-config> RPC with a subtree filter to return only the information under the <role> node in the urn:brocade.com:mgmt:brocade-aaa namespace.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="808" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <role xmlns="urn:brocade.com:mgmt:brocade-aaa">
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="808" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <role xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <name>
            <name>VLANAdmin</name>
            <desc>Manages security</desc>
        </name>
        <name>
            <name>NetworkAdmin</name>
            <desc>Manages networks</desc>
        </name>
        <name>
            <name>ClusterAdmin</name>
            <desc>Manages clusters</desc>
        </name>
    </role>
</rpc-reply>
```

Deleting a role

The following example shows how to delete a role.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="809" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <role xmlns="urn:brocade.com:mgmt:brocade-aaa">
```

```
                <name xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete">
                    <name>VLANAdmin</name>
                </name>
            </role>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="809" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring a placeholder rule

A rule created to allow the no-operation command does not enforce any authorization rules. Instead, you can use this instance as a placeholder for a valid command that is added later, as shown in the following example.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="810" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <index>75</index>
                <action>reject</action>
                <operation>read-write</operation>
                <role>NetworkAdmin</role>
                <command>
                    <enumList>no-operation</enumList>
                </command>
            </rule>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="810" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Rule processing

The following exception applies. When a match is found for a rule with the *read-only* operation, and the *accept* action, the system seeks to determine if there are any rules with the *read-write* operation and the *accept* action. If such rules are found, the rule with the *read-write* permission is applied.

Adding a rule

The following example shows how to add a rule.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="811" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
```

```
        <target>
            <running/>
        </target>
        <config>
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <index>150</index>
                <action>accept</action>
                <operation>read-write</operation>
                <role>SecAdminUser</role>
                <command>
                    <enumList>config</enunList>
                </command>
            </rule>
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <index>155</index>
                <action>accept</action>
                <operation>read-write</operation>
                <role>SecAdminUser</role>
                <command>
                    <enumList>username</enumList>
                </command>
            </rule>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="811" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Changing a rule

Changing a rule is like adding a rule, only the rule already exists. The following example changes the previously created rule (index number 155).

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="812" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <index>155</index>
                <command>
                    <enumList>role</enumList>
                </command>
            </rule>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="812" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

After changing the rule 155, SecAdminUser can log in to the switch and execute the **role** command and not the **username** command.

Deleting a rule

The following example shows how to delete a rule.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="813" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa"
             xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
             operation="delete">
                <index>155</index>
            </rule>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="813" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Verifying a rule

The following example verifies a rule.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="814" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="814" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <index>30</index>
        <action>accept</action>
        <operation>read-write</operation>
        <role>NetworkSecurityAdmin</role>
        <command>
            <enumList>role</enumList>
        </command>
    </rule>
    <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <index>31</index>
        <action>accept</action>
        <operation>read-write</operation>
        <role>NetworkSecurityAdmin</role>
        <command>
            <enumList>rule</enumList>
        </command>
    </rule>
```

```
    <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <index>32</index>
        <action>accept</action>
        <operation>read-write</operation>
        <role>NetworkSecurityAdmin</role>
        <commsand>
            <enumList>username</enumList>
        </command>
    </rule>
</rpc-reply>
```

Configuring a Brocade VCS Fabric security administrator account

The following example create a role for a Brocade VCS Fabric security administrator, creates a user account and associates it with the newly created role, and creates rules to specify the RBAC permissions for the NetworkSecurityAdmin role.

This example grants the secAdminUser account access to the configuration-level commands **role**, **rule**, **username**, **aaa**, and **radius-server**. Any account associated with the NetworkSecurityAdmin role can now create and modify user accounts, manage roles, and define rules. In addition, the role permits configuring a RADIUS server and setting the login sequence.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="815" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <role xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <name>
                    <name>NetworkSecurityAdmin</name>
                    <desc>Manages security</desc>
                </name>
            </role>
            <username xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <name>secAdminUser</name>
                <role>NetworkSecurityAdmin</role>
                <user-password>testpassword</user-password>
            </username>
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <index>30</index>
                <action>accept</action>
                <operation>read-write</operation>
                <role>NetworkSecurityAdmin</role>
                <command>
                    <enumList>role</enumList>
                </command>
            </rule>
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <index>31</index>
                <action>accept</action>
                <operation>read-write</operation>
                <role>NetworkSecurityAdmin</role>
                <command>
                    <enumList>rule</enumList>
                </command>
            </rule>
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
```

```
                <index>32</index>
                <action>accept</action>
                <operation>read-write</operation>
                <role>NetworkSecurityAdmin</role>
                <command>
                    <enumList>username</enumList>
                </command>
            </rule>
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <index>33</index>
                <action>accept</action>
                <operation>read-write</operation>
                <role>NetworkSecurityAdmin</role>
                <command>
                    <enumList>aaa</enumList>
                </command>
            </rule>
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <index>34</index>
                <action>accept</action>
                <operation>read-write</operation>
                <role>NetworkSecurityAdmin</role>
                <command>
                    <enumList>radius-server</enumList>
                </command>
            </rule>
            <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <index>35</index>
                <action>accept</action>
                <operation>read-write</operation>
                <role>NetworkSecurityAdmin</role>
                <command>
                    <enumList>configure</enumList>
                </command>
            </rule>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="815" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring a Brocade FCoE administrator account

The following example creates an FCOEAdmUser account that is associated with the FCoEAdmin role. It creates the access permissions rules that allow the user to perform FCoE operations.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="816" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <role xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <name>
                    <name>FCoEAdmin</name>
                    <desc>Manages FCoE</desc>
```

```
            </name>
        </role>
        <username xmlns="urn:brocade.com:mgmt:brocade-aaa">
            <name>FCoEAdmUser</name>
            <role>FCoEAdmin</role>
            <user-password>testpassword</user-password>
        </username>
        <rule xmlns="urn:brocade.com:mgmt:brocade-aaa">
            <index>40</index>
            <action>accept</action>
            <operation>read-write</operation>
            <role>FCoEAdmin</role>
            <command>
                <interface-fcoe>
                    <interface>
                        <fcoe/>
                    </interface>
                </interface-fcoe>
            </command>
        </rule>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="816" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>

</rpc-reply>
```

Password policies

The following is a list of the configurable password policies:

- Password strength policy. Refer the config guide
- Password encryption policy

Password strength policy

Refer to the config guide for configurable password policy parameters.

NOTE
Passwords can be a maximum of 40 characters in length.

Password encryption policy

In the following example, the testuser account password is created in clear-text after password encryption has been enabled. The global encryption policy overrides the account-level encryption settings. The password is stored as encrypted.

1. Issue the <edit-config> RPC to configure the <service> node in the urn:brocade.com:mgmt:brocade-aaa namespace.
2. Under the <service> node, include the empty <password-encryption> element to enforce password encryption.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="817" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <service xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <password-encryption/>
            </service>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="817" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. To verify the enforcement of password encryption, issue the <edit-config> RPC with a subtree filter to return information under the <service> node in the urn:brocade.com:mgmt:brocade-aaa namespace.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="818" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <service xmlns="urn:brocade.com:mgmt:brocade-aaa">
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="818" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <service xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <password-encryption/>
    </service>
</rpc-reply>
```

4. Issue the <edit-config> RPC to create the user account with a password.

   In this case, the <encryption-level> element specifies to save the password as clear text (encryption-level = 0).

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="819" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <username xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <name>testuser</name>
                <role>testrole</role>
                <desc>Test User</desc>
                <encryption-level>0</encryption-level>
                <user-password>Test User</user-password>
            </username>
        </config>
    </edit-config>
```

```
</rpc>

<rpc-reply message-id="819" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

5. To verify the form in which the password is stored, issue the <get-config> RPC with a subtree filter to retrieve the information under the <username> node in the urn:brocade.com:mgmt:brocade-aaa namespace.

   The output shows the password stored in encrypted form because the switch-level encryption level overrides the account level.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="820" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <username xmlns="urn:brocade.com:mgmt:brocade-aaa"/>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="820" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <username xmlns="urn:brocade.com:mgmt:brocade-aaa"/>
        <name>testuser</testuser>
        <user-password>cONW1RQ0nTV9Az42/9uCQg==\n</user-password>
        <encryption-level>7</encryption-level>
        <role>userrole</role>
        <desc>Test User</desc>
    </username>
</rpc-reply>
```

   When you disable the password encryption service, any new passwords added in clear-text will be stored as clear-text on the switch. Existing encrypted passwords remain encrypted.

Configuring the account lockout threshold

The following example shows how to configure the lockout threshold.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="821" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <password-attributes xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <max-retry>4</max-retry>
            </password-attributes>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="821" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

When a user account is locked, it can be unlocked using the procedure described in "Unlocking a user account" on page 168.

Creating a password policy

The following example defines a password policy that requires passwords to be at least eight characters long, contain at least two lowercase characters, at least one uppercase character, at least one numeric character, and at least one special character. The policy also enforces lockout after four attempts to enter the password. The <admin-lockout-enable> node enables the lockout policy for admin role accounts.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="822" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <password-attributes xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <min-length>8</min-length>
                <max-retry>4</max-retry>
                <character-retriction>
                    <lower>2</lower>
                    <upper>1</upper>
                    <numeric>1</numeric>
                    <special-char>1</special-char>
                    </admin-lockout-enable>
                </character-restriction>
            </password-attributes>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="822" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Displaying password attributes

To retrieve the current password policy.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="823" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <password-attributes xmlns="urn:brocade.com:mgmt:brocade-aaa"/>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="823" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <password-attributes xmlns="urn:brocade.com:mgmt:brocade-aaa"/>
        <max-retry>4</max-retry>
        <character-restriction>
            <upper>1</upper>
```

```
            <lower>2</lower>
            <numberic>1</numeric>
            <shpecial-char>1</special-char>
        </character-restriction>
    </password-attributes>
</rpc-reply>
```

Restoring the default password policy

The following example shows how to reset all password attributes to default values.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="824" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <password-attributes xmlns="urn:brocade.com:mgmt:brocade-aaa"
             xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
             operation="delete"/>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="824" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To restore the default value of just one password attribute, perform the same operation, but include the attribute element under the <password-attributes> node and apply the delete operation to that element instead of to the entire <password-attributes> node. All other password attributes remain unchanged.

Security event logging

Security event logging utilizes the RASlog audit infrastructure to record security-related audit events. Any user-initiated security event generates an auditable event. Audited events are generated for all Management interfaces. In Brocade VCS Fabric mode, for cluster-wide events, the audit is generated on all switches of the cluster. Refer to the *Network OS Message Reference* for information on how to configure and monitor security audit logging.

Chapter

External Server Authentication 19

In this chapter

Remote server authentication with NETCONF overview

This chapter provides procedures for configuring external AAA servers using the NETCONF interface. Refer to the *Network OS Security Configuration Guide* for the following related information:

- An overview of remote authentication server concepts, including the supported authentication modes:
  - Terminal Access Controller Access Control System Plus (TACACS+)
  - Remote Authentication Dial In User Service (RADIUS)
  - Lightweight Directory Access Protocol (LDAP)
  - Local
- Procedures for configuring remote authentication using the Network OS command line interface (CLI)
- Procedures for configuring server-side RADIUS

Through the NETCONF interface, you can perform the following operations on LDAP:

- Use the <edit-config> RPC to connect to or disconnect from a authentication server, or configure client-side TACACS+, RADIUS, or LDAP parameters.
- Use the <get-config> RPC to validate configuration settings.
- Use the <ldapca> action located in the urn:mgmt:brocade.com:mgmt:brocade-certutil namespace to import or delete an LDAP CA certificate.

Parameters for configure remote authentication are defined in the brocade-aaa YANG module. Refer to the *Network OS YANG Reference Manual* for details.

Login authentication mode

Using the NETCONF interfaces, you can configure primary and secondary authentication modes. The primary mode can be RADIUS, TACACS+, LDAP, or local. The secondary mode is optional and can only be local, and then only if the primary mode is RADIUS, TACACS+, or LDAP.

Setting and verifying the login authentication mode

The following example shows how to configure and verify the login authentication mode.

The following example configures TACACS+ as the primary source of authentication and the local user database as the secondary source.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="900" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <authentication>
                        <login>
                            <first>tacacs+</first>
                            <second>local</second>
                        </login>
                    </authentication>
                </aaa>
            </aaa-config>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="900" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To verify the configuration, issue the <get-config> RPC with a subtree filter to limit the returned information to the contents of the <aaa-config>/<aaa>/<authentication> node.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="901" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <authentication/>
                </aaa>
            </aaa-config>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="901" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <aaa>
            <authentication>
                <login>
                    <first>tacacs+</first>
                    <second>local</second>
                </login>
            </authentication>
```

```
        </aaa>
    </aaa-config>
</rpc>
```

Log in to the switch using an account with TACACS+ only credentials to verify that TACACS+ is being used to authenticate the user.

Resetting the login authentication mode

When you reset the login authentication mode, primary authentication reverts to local mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="902" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <authentication>
                        <login
                         xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    </authentication>
                </aaa>
            </aaa-config>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="902" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To verify the configuration, issue the <get-config> RPC with a subtree filter to limit the returned information to the contents of the <aaa-config>/<aaa>/<authentication> node.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="903" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <authentication/>
                </aaa>
            </aaa-config>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="903" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <aaa>
            <authentication>
                <login>
                    <first>local</first>
```

```
            </login>
        </authentication>
    </aaa>
</aaa-config>
</rpc>
```

6. Log in to the switch using an account with TACACS+ only credentials. The login should fail with an “access denied” error.
7. Log in to the switch using an account with local only credentials. The login should succeed.

Changing the login authentication mode

To change the authentication mode, you must first reset the configuration to the default local mode, and then set the authentication mode as desired. The following example resets the existing TACACS+ mode to local mode and then sets the authentication mode to RADIUS.

1. Reset the configuration to the default value.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <authentication>
                        <login
                         xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    </authentication>
                </aaa>
            </aaa-config>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. Specify the desired authentication mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="905" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <authentication>
                        <login>
                            <first>radius</first>
                            <second>local</second>
                        </login>
                    </authentication>
                </aaa>
```

```
            </aaa-config>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="905" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Verify the configuration with the <get-config> RPC.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="906" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <authentication/>
                </aaa>
            </aaa-config>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="906" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <aaa>
            <authentication>
                <login>
                    <first>radius</first>
                    <second>local</second>
                </login>
            </authentication>
        </aaa>
    </aaa-config>
</rpc>
```

4. Log in to the switch using an account with TACACS+ credentials. The login should fail with an “access denied” error.

5. Log in to the switch using an account with RADIUS credentials. The login should succeed.

Adding a RADIUS server to the client’s server list

The following example shows how to add a RADIUS server to the client server list.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="907" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <radius-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host>
                    <host-name>10.38.37.180</host-name>
                    <protocol>pap</protocol>
```

```
                <key>new#virgo*secret</key>
                <timeout>10</timeout>
            </host>
        </radius-server>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="907" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To validate the new configuration, issue the <get-config> RPC with a subtree filter to limit the returned information to RADIUS server 10.38.37.130.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="908" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <radius-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host>
                    <hostname>10.38.37.180</hostname>
                </host>
            </radius-server>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="908" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <radius-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <host>
            <hostname>10.38.37.180</hostname>
            <auth-port>1812</auth-port>
            <protocol>pap</protocol>
            <key>new#virgo*secret</key>
            <retries>5</retries>
            <timeout>10</timeout>
        </host>
    </radius-server>
</rpc>
```

Modifying the RADIUS server configuration

To modify the RADIUS server configuration on the client, perform the following steps.

1. Determine the configured RADIUS servers by Issuing the <get-config> RPC with a subtree filter to return only information about configured RADIUS servers.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="909" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <radius-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host/>
```

```
            </radius-server>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="909" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <radius-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <host>
            <hostname>10.38.37.180</hostname>
            <auth-port>1812</auth-port>
            <protocol>pap</protocol>
            <key>new#virgo*secret</key>
            <retries>5</retries>
            <timeout>10</timeout>
        </host>
        <host>
            <hostname>10.24.65.6</hostname>
            <auth-port>1812</auth-port>
            <protocol>pap</protocol>
            <key>changedesc</key>
            <retries>5</retries>
            <timeout>3</timeout>
        </host>
    </radius-server>
</rpc>
```

2. Issue the <edit-config> RPC to change the configuration of the RADIUS server.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="910" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <radius-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host>
                    <host-name>10.38.37.180</host-name>
                    <auth-port>1812</auth-port>
                    <key>changedesc</key>
                    <retries>5</retries>
                    <timeout>3</timeout>
                </host>
            </radius-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="910" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Re-issue the <get-config> RPC with a subtree filter to restrict the output to the modified RADIUS server and verify the configuration change.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="911" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
```

```
            <filter type="subtree">
                <radius-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                    <host>
                        <hostname>10.38.37.180</hostname>
                    </host>
                </radius-server>
            </filter>
        </get-config>
    </rpc>

    <rpc-reply message-id="911" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <radius-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
            <host>
                <hostname>10.38.37.180</hostname>
                <auth-port>1812</auth-port>
                <protocol>pap</protocol>
                <key>changedesc</key>
                <retries>5</retries>
                <timeout>3</timeout>
            </host>
        </radius-server>
    </rpc>
```

Removing a RADIUS server from a client's server list

The following example shows how to remove a RADIUS server from a client's server list.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="912" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <radius-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete">
                    <host-name>10.38.37.180</host-name>
                </host>
            </radius-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="912" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the client to use RADIUS for login authentication

After configuring the client-side RADIUS server list, you must set the authentication mode so that RADIUS is used as the primary source of authentication. Refer to "Login authentication mode" on page 181 for information on how to configure the login authentication mode.

Adding a TACACS+ server to the client's server list

The following example shows how to add a TACACS+ server to the client's server list.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <tacacs-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host>
                    <hostname>fec0:60:69bc:94:211:25ff:fec4:6010</hostname>
                    <protocol>chap</protocol>
                    <key>new#hercules*secret</key>
                </host>
            </tacacs-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Issue the <get-config> RPC with a subtree filter to limit the output to information about the TACACS+ server to verify the configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="914" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <tacacs-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host>
                    <hostname>fec0:60:69bc:94:211:25ff:fec4:6010</hostname>
                </host>
            </tacacs-server>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="914" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <tacacs-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <host>
            <hostname>fec0:60:69bc:94:211:25ff:fec4:6010</hostname>
            <port>49</port>
            <protocol>chap</protocol>
            <key>new#hercules*secret</key>
            <retries>5</retries>
            <timeout>5</timeout>
        </host>
    </tacacs-server>
</rpc>
```

Modifying the TACACS+ server configuration

To modify the TACACS+ configuration, perform the following steps.

1. Issue the <get-config> RPC with a subtree filter to return only information about configured TACACS+ servers.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="915" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <tacacs-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host/>
            </tacacs-server>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="915" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <tacacs-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <host>
            <hostname>fec0:60:69bc:94:211:25ff:fec4:6010</hostname>
            <port>49</port>
            <protocol>chap</protocol>
            <key>new#hercules*secret</key>
            <retries>5</retries>
            <timeout>5</timeout>
        </host>
    </tacacs-server>
</rpc>
```

2. Issue the <edit-config> RPC to change the configuration of the TACACS+ server.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="916" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <tacacs-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host>
                    <hostname>fec0:60:69bc:94:211:25ff:fec4:6010</hostname>
                    <key>changedesc</key>
                    <retries>100</retries>
                </host>
            </tacacs-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="916" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Issue the <get-config> RPC with a subtree filter to restrict the output to the modified TACACS+ server to verify the configuration change.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="917" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <tacacs-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host>
                    <hostname>fec0:60:69bc:94:211:25ff:fec4:6010</hostname>
                </host>
            </tacacs-server>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="917" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <tacacs-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <host>
            <hostname>fec0:60:69bc:94:211:25ff:fec4:6010</hostname>
            <port>49</port>
            <protocol>chap</protocol>
            <key>changedesc</key>
            <retries>100</retries>
            <timeout>5</timeout>
        </host>
    </tacacs-server>
</rpc>
```

Removing a TACACS+ server from a client’s server list

The following example shows how to remove a TACACS+ server from a client’s server list.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="918" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <tacacs-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete">
                    <host-name>10.54.37.170</host-name>
                </host>
            </tacacs-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="918" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the client to use TACACS+ for login authentication

After configuring the client-side TACACS+ server list, you must set the authentication mode so that TACACS+ is used as the primary source of authentication. Refer to "Login authentication mode" on page 181 for information on how to configure the login authentication mode.

TACACS+ accounting

This section provides procedures and examples for configuring TACACS+ accounting on the client. For related conceptual information, limitations, information about viewing TACACS+ accounting logs, and firmware downgrade considerations, refer to the *Network OS Security Configuration Guide*.

Enabling login accounting

The following example shows how to enable login accounting on a switch where accounting is disabled.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="919" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <accounting>
                        <exec>
                            <defaultacc>
                                <start-stop>
                                    <server-type>tacacs+</server-type>
                                </start-stop>
                            </defaultacc>
                        </exec>
                    </accounting>
                </aaa>
            </aaa-config>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="919" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Issue the <get-config> RPC with a subtree filter to limit the output to information under the <aaa-config>/<aaa>/<accounting> node to verify the configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="920" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
```

```
        </source>
        <filter type="subtree">
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <accounting/>
                </aaa>
            </aaa-config>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="920" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <aaa>
            <accounting>
                <exec>
                    <defaultacc>
                        <start-stop>
                            <server-type>taccacs+</server-type>
                        </start-stop>
                    </defaultacc>
                </exec>
            </accounting>
        </aaa>
    </aaa-config>
</rpc>
```

Enabling command accounting

The following example shows how to enable login accounting on a switch where login accounting is enabled and command accounting is disabled.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="921" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <accounting>
                        <commands>
                            <defaultacc>
                                <start-stop>
                                    <server-type>tacacs+</server-type>
                                </start-stop>
                            </defaultacc>
                        </commands>
                    </accounting>
                </aaa>
            </aaa-config>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="921" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Issue the <get-config> RPC with a subtree filter to limit the output to information under the <aaa-config>/<aaa>/<accounting> node to validate the configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="922" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <accounting/>
                </aaa>
            </aaa-config>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="922" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <aaa>
            <accounting>
                <commands>
                    <defaultacc>
                        <start-stop>
                            <server-type>taccacs+</server-type>
                        </start-stop>
                    </defaultacc>
                </commands>
            </accounting>
        </aaa>
    </aaa-config>
</rpc>
```

Disabling accounting

You must perform the disable operation separately for login accounting and for command accounting.

The following example shows how to disable command accounting.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="923" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <accounting>
                        <commands>
                            <defaultacc>
                                <start-stop>
                                    <server-type>none</server-type>
                                </start-stop>
                            </defaultacc>
                        </commands>
```

```
                    </accounting>
                </aaa>
            </aaa-config>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="923" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example shows how to disable login accounting.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="924" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <aaa-config xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <aaa>
                    <accounting>
                        <exec>
                            <defaultacc>
                                <start-stop>
                                    <server-type>none</server-type>
                                </start-stop>
                            </defaultacc>
                        </exec>
                    </accounting>
                </aaa>
            </aaa-config>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="924" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Server authentication

As a part of user authentication using LDAP, the Brocade switch can be configured to support server certificate authentication. Refer to the *Network OS Security Configuration Guide* for additional conceptual details.

Importing a CA certificate

The following example shows how to import a CA certificate.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="925">
   <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
       <data>
           <certutil xmlns="urn:brocade.com:mgmt:brocade-certutil">
               <import>
                   <ldapca>
                       <protocol>SCP</protocol>
                       <user>jane</user>
```

```
                <password>janepasswd</password>
                <host>10.23.24.56</host>
                <directory>/usr/ldapcacert</directory>
                <file>cacert.pam</file>
            </ldapca>
        </import>
      </certutil>
    </data>
  </action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="925">
  <ok/>
</rpc-reply>
```

Deleting CA certificates

This operation deletes the CA certificates of all the Active Directory (AD) servers.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="926">
   <action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
     <data>
       <no xmlns="urn:brocade.com:mgmt:brocade-common-def">
         <certutil xmlns="urn:brocade.com:mgmt:brocade-certutil">
           <ldapca/>
         </certutil>
       </no>
     </data>
  </action>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="926">
  <ok/>
</rpc-reply>
```

Adding an LDAP server to the client's server list

The following example shows how to add an LDAP server and configure access attributes.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="927" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <ldap-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <host>
          <hostname>10.24.65.6</hostname>
          <basedn>security.brocade.com</basedn>
          <port>3890</port>
          <timeout>8</timeout>
          <retries>3</retries>
        </host>
      </ldap-server>
    </config>
  </edit-config>
</rpc>
```

```
<rpc-reply message-id="927" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Confirm the LDAP settings by issuing a <get-config> RPC with a subtree filter to return configuration information for the LDAP server.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="928" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <ldap-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host>
                    <hostname>10.24.65.6</hostname>
                </host>
            </ldap-server>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="928" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ldap-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
        <host>
            <hostname>10.24.65.6</hostname>
            <port>3890</port>
            <retries>3</retries>
            <timeout>8</timeout>
            <basedn>security.brocade.com</basedn>
        </host>
    </ldap-server>
</rpc>
```

Optional: Use the delete operation on an attribute element to set the attribute back to the default value.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="929" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ldap-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host>
                    <hostname>10.24.65.6</hostname>
                    <retries xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                </host>
            </ldap-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="929" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Removing an LDAP server

The following example shows how to delete a connection to an LDAP server.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="930" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ldap-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete">
                    <hostname>10.24.65.6</hostname>
                </host>
            </ldap-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="930" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Active Directory groups

An Active Directory group defines access permissions for the LDAP server similar to Brocade roles.

Mapping an Active Directory group to a switch role

A maximum of 16 AD groups can be mapped to the switch roles.

The following example shows how to map an Active Directory (AD) group to a switch role.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="931" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ldap-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <maprole>
                    <group>
                        <ad-group>Administrator</ad-group>
                        <switch-role>admin</switch-role>
                    </group>
                </maprole>
            </ldap-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="931" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Removing the mapping of an Active Directory to a switch role

The following example shows how to remove an AD group mapping from a switch role.

The following example removes the mapping between the Brocade admin role and the Active Directory Administrator group. A Brocade user with the admin role can no longer perform the operations associated with the Active Directory Administrator group.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="932" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ldap-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <maprole>
                    <group>
                        <ad-group xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete">Administrator</ad-group>
                    </group>
                </maprole>
            </ldap-server>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="932" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the client to use LDAP/AD for login authentication

After configuring the switch LDAP server list, you must set the authentication mode so that LDAP is used as the primary source of authentication. Refer to "Login authentication mode" on page 181 for information on how to configure the login authentication mode.

Chapter

Fabric Authentication

20

In this chapter

Fabric authentication with NETCONF overview

When you connect a Brocade VCS Fabric to a Fabric OS fabric, the Network OS Fibre Channel E_Ports on the Brocade VDX hardware connect through Interswitch links (ISLs) to EX_Ports on an FC router, which in turn connects to the Fabric OS network. To ensure that no unauthorized devices can access the fabric, Network OS provides support for security policies and protocols capable of authenticating Network OS devices (E_Ports) to the EX_Ports on the FC router that provides access to the SAN storage and services.

This chapter describes how to use NETCONF remote procedure calls (RPCs) to configure fabric authentication and Switch Connection Control (SCC) policies. Refer to the *Network OS Security Configuration Guide* for the following related information:

- An overview of the Diffie Hellman - Challenge Handshake Authentication Protocol (DH-CHAP)
- An overview of how shared secret keys are used
- A overview of authentication policy configuration including details about each possible authentication state and the transitions among them
- An overview of SCC policies, including a discussion about defined and active policy sets
- How to configure fabric authentication and SCC using the Network OS command line interface (CLI)

Through the NETCONF interface, you can perform the following fabric authentication-related operations:

- Use the <edit-config> RPC to set authentication parameters and activate the FC-AUTH protocol.
- Use the <get-config> RPC to validate configuration settings.
- Use the <fcsp>/<auth-secret>/<dhchap> custom action to configure shared DH-CHAP shared secrets.
- Use the <show>/<fcsp>/<dhchap> custom action to return the device (WWN) for which the shared secret is configured.

Through the NETCONF interface, you can perform the following SCC policy-related operations:

- Use the <edit-config> RPC to configure the SCC policy.
- Use the <secpolicy>/<activate> custom action to activate the SCC policy.
- Use the <get-config> RPC to verify SCC policy configuration settings.

FC AUTH and SCC policy parameters are defined in the brocade-fc-auth YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Device authentication configuration

Configuring a Brocade VDX hardware switch to access a SAN fabric connected through an FC router involves the following steps.

1. Configure the matching shared secret pairs on the VDX hardware and on the FC router.
2. Configure the authentication policy on the VDX hardware switch (the FC router configuration is fixed).
3. Activate the authentication policy.

Setting up secret keys can quickly become an administrative challenge as your fabric size increases. As a minimum, key pairs must be installed on all connected fabric entities. However, when connections change, you must install new key pairs to accommodate these changes. If you anticipate this situation, you may install key pairs for all possible connections up front, thus enabling links to change arbitrarily while still maintaining a valid key pair for any new connection.

Configuring DH-CHAP shared secrets

The following example shows how to configure the DH-CHAP shared secrets.

NOTE
Only the following non-alphanumeric characters are valid for the secret key:
@, $, %, ^, &, *, (,), _, +, -, <, >, {, }, [,], ;, ', and :

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1400">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <fcsp xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
                <auth-sercret>
                    <dhchap>
                        <node>10:00:00:05:1e:7a:c3:00</node>
                        <peer-secret>12345678</peer-secret>
                        <local-secret>87654321</local-secret>
                    </dhchap>
                </auth-secret>
            </fcsp>
        </nca:data>
    </nca:action>
</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="1400">
    <fcsp xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
        <auth-secret>
            <dhchap>
                <result>Shared secret is configured successfully.</result>
            </dhchap>
        </auth-secret>
    </fcsp>
</rpc-reply>
```

Returning the device WWN for which a shared secret is configured

To return the device (WWN) for which the shared secret is configured, issue the <show>/<fcsp>/<auth-secret>/<dhchap> action, where the <show> node is located in the urn:brocade.com:mgmt:brocade-common-def namespace, and the <fcsp>/<auth-secret> nodes are located in the urn:brocade.com:mgmt:brocade-fc-auth namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1401">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <fcsp xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
                    <auth-sercret>
                        <dhchap/>
                    </auth-secret>
                </fcsp>
            </show>
        </nca:data>
    </nca:action>
</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="1401">
    <show xmlns="urn:brocade.com:mgmt:brocade-common-def">
        <fcsp xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
            <auth-secret>
                <dhchap>
                    <auth-show-secret>
                        <nodeid>10:00:00:05:1e:7a:c3:00</nodeid>
                    </auth-show-secret>
                </dhchap>
            </auth-secret>
        </fcsp>
    </show>
</rpc-reply>
```

Removing shared secrets

The following example shows how to remove the shared secrets.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1402">
    <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
        <nca:data>
            <no xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <fcsp xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
                    <auth-sercret>
                        <dhchap>
                            <node>10:00:00:05:1e:7a:c3:00</node>
                        </dhchap>
                    </auth-secret>
                </fcsp>
            </show>
        </nca:data>
    </nca:action>
</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="1402">
    <no xmlns="urn:brocade.com:mgmt:brocade-common-def">
```

```
        <fcsp xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
            <auth-secret>
                <dhchap>
                    <result>Shared secret successfully removed.</result>
                </dhchap>
            </auth-secret>
        </fcsp>
    </show>
</rpc-reply>
```

Setting the authentication policy parameters

The following example shows how to set the authentication policy parameters.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1403" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <fcsp xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
                <auth>
                    <proto>
                        <auth-type>dh-chap</auth-type>
                        <group>2</group>
                        <hash>md5</hash>
                    </proto>
                    <policy>
                        <switch>off</switch>
                    </policy>
                </auth>
            </fcsp>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Activating the authentication policy

The following example shows how to activate the authentication policy.

```
    <?xml version="1.0" encoding="UTF-8"?>
    <rpc message-id="1404" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <edit-config>
            <target>
                <running/>
            </target>
            <config>
                <fcsp xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
                    <auth>
                        <policy>
                            <switch>on</switch>
                        </policy>
                    </auth>
```

```
            </fcsp>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1404" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Issue the <get-config> RPC with a subtree filter to return the contents of the <fcsp>/<auth>/<policy> node in the urn:brocade.com:mgmt:brocade-fc-auth namespace to return and verify the switch policy state.

Switch Connection Control policy configuration

This section provides procedures to create, modify, activate, and remove a defined Switch Connection Control (SCC) policy.

Creating a defined SCC policy

The following example shows how to create an SCC policy, add members, and verify the configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1405" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <secpolicy xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
                <defined-policy>
                    <policies>
                        <policy>SCC_POLICY</policy>
                        <member-entry>
                            <member>10:00:00:05:1e:00:69:00</member>
                        </member-entry>
                        <member-entry>
                            <member>22:22:22:22:22:22:22:22</member>
                        </member-entry>
                    </policies>
                </defined-policy>
            </secpolicy>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1405" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Modifying the SCC policy

The following example adds member 10:00:00:08:2f:00:79:00 and removes member 22:22:22:22:22:22:22:22.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1406" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
      <target>
         <running/>
      </target>
      <config>
         <secpolicy xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
            <defined-policy>
               <policies>
                  <policy>SCC_POLICY</policy>
                  <member-entry>
                     <member>10:00:00:08:2f:00:79:00</member>
                  </member-entry>
                  <member-entry
                   xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                   operation="delete">
                     <member>22:22:22:22:22:22:22:22</member>
                  </member-entry>
               </policies>
            </defined-policy>
         </secpolicy>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="1406" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Activating the SCC policy

The following example shows how to activate an SCC policy and verify activation.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1407">
   <nca:action xmlns:nca="http://tail-f.com/ns/netconf/actions/1.0">
      <nca:data>
         <secpolicy xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
            <action>
               <activate/>
            </action>
         </secpolicy>
      </nca:data>
   </nca:action>
</rpc>

<rpc-reply message-id="1407" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

To verify the active configuration, issue a <get-config> RPC with a subtree filter to return only the <secpolicy>/<active-policy> node in the urn:brocade.com:mgmt:brocade-fc-auth namespace.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1408" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <get-config>
      <source>
         <running/>
      </source>
      <filter type="subtree">
```

```
            <secpolicy xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
                <active-policy>
                </active-policy>
            </secpolicy>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="1408" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <secpolicy xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
        <active-policy>
            <policies>
                <policy>
                <member-entry>
                    <member>10:00:00:05:1e:00:69:00</member>
                </member-entry>
                <member-entry>
                    <member>10:00:00:08:2f:00:79:00</member>
                </member-entry>
            </policies>
        </active-policy>
    </secpolicy>
</rpc-reply>
```

Removing the SCC policy

The following example shows how to remove the SCC policy.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1409" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <secpolicy xmlns="urn:brocade.com:mgmt:brocade-fc-auth">
                <defined-policy>
                    <policies xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete">
                        <policy>SCC_POLICY</policy>
                    </policies>
                </defined-policy>
            </secpolicy>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1409" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Section

III

Network OS Layer 2 Switch Features

This section describes the Layer 2 features of Network OS, and includes the following chapters:

Chapterhmm

Configuring AMPP

22

In this chapter

AMPP configuration with NETCONF overview

This chapter describes procedures for configuring and monitoring port-profiles using NETCONF interfaces. Refer to the *Network OS Layer 2 Switching Configuration Guide* for the following information:

- Conceptual information about how AMPP port-profiles work
- Conceptual information about how AMPP port-profile-domains work
- How AMPP port-profiles work with vLAG and Switched Port Analyzer (SPAN)
- Scalability information
- What a port-profile contains
- Definitions of port-profile states

Through the NETCONF interface, you can perform the following operations on AMPP:

- Use the <edit-config> RPC to configure AMPP port-profiles.
- Use the <edit-config> RPC to configure the port-profile-domain.
- Use the <get-config> RPC to validate configuration settings.
- Use the <get-port-profile-status> custom RPC to return the current status of AMPP profiles.
- Use the <get-port-profile-for-inf> custom RPC to return information about port-profiles to which interfaces are applied.

The AMPP configuration model is defined in the brocade-port-profile.yang module and the AMPP custom RPC is defined in the brocade-port-profile-ext YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Configuring AMPP port-profiles

This section contains procedures for configuring AMPP port-profiles using NETCONF interfaces.

Configuring a new port-profile

To support VM MAC address learning, the default port-profile is employed. The default profile is different from the other user-defined AMPP profiles:

- The port-profile ID (ppid) of the profile cannot be changed.
- The VLAN subprofile cannot be modified.
- The QoS subprofile and security-profile cannot be added.
- The default port-profile cannot be activated.
- When there are no port-profile-ports in the system, the default port-profile can be added with the FCoE profile.

The following example shows how to create and configure a new port-profile name.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1701" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>vm1-port-profile</name>
                <vlan-profile>
                    <switchport>
                        <trunk>
                            <native-vlan>300</native-vlan>
                            <allowed>
                                <vlan>
                                    <add>300</add>
                                </vlan>
                            </allowed>
                        </trunk>
                    </switchport>
                </vlan-profile>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1701" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example shows how to activate the profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1702" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <activate/>
                </port-profile>
```

```
            </port-profile-global>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1702" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example shows how to associate the profile with the MAC address for each host.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1703" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0001</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0002</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0003</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0004</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0005</mac-address>
                    </static>
                </port-profile>
            </port-profile-global>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1703" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring VLAN profiles

The VLAN profile defines the VLAN membership of the overall port-profile, which includes both the tagged and untagged VLANs.

NOTE
Private VLAN port mode commands are not available for AMPP VLAN profiles.

To configure the VLAN profile, perform the following steps.

1. Deactivate the port-profile before modifying the VLAN profile.

   AMPP profiles cannot be modified while active.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1704" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <activate xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                </port-profile>
            </port-profile-global>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1704" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. The following example configures the VLAN profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1705" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>vm1-port-profile</name>
                <vlan-profile>
                    <switchport>
                        <access>
                            <vlan>
                                <name>200</name>
                            </vlan>
                        </access>
                        <trunk>
                            <allowed>
                                <vlan>
                                    <add>10,20,30-40</add>
                                </vlan>
```

```
                    </allowed>
                    <native-vlan>300</native-vlan>
                </trunk>
            </switchport>
        </vlan-profile>
    </port-profile>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1705" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Activate the profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1706" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <activate/>
                </port-profile>
            </port-profile-global>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1706" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

4. Associate the profile with the MAC address for each host.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1707" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-adddress>0050.56bf:0001</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0002</mac-address>
                    </static>
                </port-profile>
```

```
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0003</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0004</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0005</mac-address>
                    </static>
                </port-profile>
            </port-profile-global>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1707" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Adding MAC classification to an access VLAN

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>PROFILE_1</name>
                <vlan-profile>
                    <switchport>
                        <access-mac-vlan-classification>
                        <access>
                            <vlan>
                                <access-vlan-id>5001</access-vlan-id>

<access-mac-address>0001.0010.0100</access-mac-address>
                            </vlan>
                        </access>
                        </access-mac-vlan-classification>
                    </switchport>
                </vlan-profile>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Adding MAC-group classification to an access VLAN

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>PROFILE_1</name>
                <vlan-profile>
                    <switchport>
                        <access-mac-group-vlan-classification>
                            <access>
                                <vlan>
                                    <access-vlan-id>5002</access-vlan-id>
                                    <access-mac-group>11</access-mac-group>
                                </vlan>
                            </access>
                        </access-mac-group-vlan-classification>
                    </switchport>
                </vlan-profile>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Adding a C-TAG classification to trunk VLAN port-profile

The following example configures a trunk VLAN with a C-TAG range.

```
<?xml version="1.0" encoding="UTF-8"?>
    <rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
        <edit-config>
            <target>
                <running/>
            </target>
            <config>
                <port-profile
xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                    <name>PROFILE_1</name>
                        <vlan-profile>
                    <switchport>
                        <trunk>
                            <trunk-vlan-classification>
                            <allowed>
                            <vlan>
                            <add>
                                <trunk-vlan-id>5111</trunk-vlan-id>
                                <trunk-ctag-id>111</trunk-ctag-id>
                            </add>
                            </vlan>
                            </allowed>
                            </trunk-vlan-classification>
```

```
                    </trunk>
                </switchport>
            </vlan-profile>
        </port-profile>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Adding a C-TAG classification to a native VLAN

The following example configures a native VLAN with a C-TAG range.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>PROFILE_1</name>
                <vlan-profile>
                    <switchport>
                        <trunk>
                            <native-vlan-classification>
                            <native-vlan-id>5112</native-vlan-id>
                            <native-vlan-ctag-id>112</native-vlan-ctag-id>
                            </native-vlan-classification>
                        </trunk>
                    </switchport>
                </vlan-profile>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring FCoE profiles

1. To globally configure the FCoE profile.

   The following example configures the FCoE profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1708" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <fcoe xmlns="urn:brocade.com:mgmt:brocade-fcoe">
```

```
                <fcoe-map>
                    <fcoe-map-name>default</fcoe-map-name>
                </fcoe-map>
            </fcoe>
            </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1708" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. The following example shows how to activate the FCoE port profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1709" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>default</name>
                <fcoe-profile>
                    <fcoeport>
                        <fcoe-map-name>default</fcoe-map-name>
                    </fcoeport>
                </fcoe-profile>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1709" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring QoS profiles

To configure the QoS profile, perform the following steps.

1. Deactivate the port-profile before modifying the VLAN profile.

   AMPP profiles cannot be modified while active.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1710" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <activate xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                </port-profile>
            </port-profile-global>
```

```
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1710" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. The following example configures the QoS profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1711" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>vm1-port-profile</name>
                <qos-profile>
                    <cee>default</cee>
                    <qos>
                        <cos>7</cos>
                        <trust>
                            <trust-cos/>
                        </trust>
                        <cos-mutation>vm1-cos2cos-map</cos-mutation>
                        <cos-traffic-calss>vm1-cos2traffic-map
                         </cos-traffic-class>
                        <flowcontrol>
                            <pfc>
                                <pfc-cos>1</pfc-cos>
                                <pfc-tx>on</pfc-tx>
                                <pfc-rx>on</pfc-rx>
                            </pfc>
                            <pfc>
                                <pfc-cos>2</pfc-cos>
                                <pfc-tx>on</pfc-tx>
                                <pfc-rx>on</pfc-rx>
                            </pfc>
                        </flowcontrol>
                    </qos>
                </qos-profile>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1711" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Activate the profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1712" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
```

```
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <activate/>
                </port-profile>
            </port-profile-global>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1712" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

4. Associate the profile with the MAC address for each host.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1713" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0001</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0002</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0003</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0004</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0005</mac-address>
                    </static>
                </port-profile>
            </port-profile-global>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="1713" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring security profiles

To configure the security profile, perform the following steps.

1. AMPP profiles cannot be modified while active. Deactivate the port-profile before modifying the security profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1714" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <activate xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                </port-profile>
            </port-profile-global>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1714" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. The following example shows how to apply an ACL to the security profile. Refer to Chapter 36, “Configuring ACLs” for details about modifying the ACL security attributes.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1716" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>vm1-port-profile</name>
                <security-profile>
                    <mac>
                        <access-group>
                            <access-group-name>vm1-acl</access-group-name>
                            <in/>
                        </access-group>
                    </mac>
                </security-profile>
            </port-profile>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="1716" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following examples shows how to apply IPv4/IPv6 ACL to the security profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1716" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>vm1-port-profile</name>
                <security-profile>
                    <ip>
                        <access-group>

<ipv4-access-group-name>vm2-acl</ipv4-access-group-name>
                            <ipv4-in></ipv4-in>
                        </access-group>
                    </ip>
                </security-profile>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1716" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>

<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1716" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>vm1-port-profile</name>
                <security-profile>
                    <ipv6>
                        <access-group>

<ipv4-access-group-name>vm3-acl</ipv6-access-group-name>
                            <ipv6-in></ipv6-in>
                        </access-group>
                    </ipv6>
                </security-profile>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1716" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Activate the profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1717" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <activate/>
                </port-profile>
            </port-profile-global>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1717" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

4. Associate the profile with the MAC address for each host.

```
<rpc message-id="1718" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0001</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0002</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0003</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0004</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static>
                        <mac-address>0050.56bf:0005</mac-address>
```

```
                </static>
            </port-profile>
        </port-profile-global>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="1718" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disassociating a port-profile from a MAC address

A significant step in creating a port-profile is associating the port-profile with the MAC address of host. In the event a MAC address must be changed, perform this task for the port profile, and then repeat the steps for associating the port-profile to the MAC address. Refer to "Configuring VLAN profiles", "Configuring FCoE profiles", "Configuring QoS profiles", and "Configuring security profiles".

NOTE
You may delete multiple MAC addresses from a single port-profile in a NETCONF cal.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1719" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                       <mac-address>0050.56bf:0001</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                       <mac-address>0050.56bf:0002</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                       <mac-address>0050.56bf:0003</mac-address>
                    </static>
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                       <mac-address>0050.56bf:0004</mac-address>
                    </static>
```

```
                </port-profile>
                <port-profile>
                    <name>vm1-port-profile</name>
                    <static xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                        <mac-address>0050.56bf:0005</mac-address>
                    </static>
                </port-profile>
            </port-profile-global>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1719" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting a port-profile

You cannot delete the default port-profile.

The following example deactivates the port-profile named vm1-port-profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1722" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <activate xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                </port-profile>
            </port-profile-global>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1722" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example deletes the port-profile named vm1-port-profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1723" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile
                 xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete">
```

```
                <name>vm1-port-profile</name>
            </port-profile>
        </port-profile-global>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="1723" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting a subprofile

To delete a subprofile, perform the following steps.

1. Deactivate the port-profile—Issue an <edit-config> RPC to configure the <port-profile> node in the urn:brocade.com:mgmt:brocade-port-profile namespace. Under the <port-profile> node, specify the following leaf elements.

   The following example deactivates the port-profile named vm1-port-profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1724" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-global
             xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile>
                    <name>vm1-port-profile</name>
                    <activate xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                </port-profile>
            </port-profile-global>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1724" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. To delete the VLAN subprofile, issue an <edit-config> RPC to configure the <port-profile> node in the urn:brocade.cpm:mgmt:brocade-port-profile namespace. Under the <port-profile> node, specify the following leaf elements.

   The following example deletes the VLAN subprofile from the port-profile named vm1-port-profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1725" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>vm1-port-profile</name>
```

```
                <vlan-profile
                 xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete"/>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1725" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. To delete the security subprofile, issue an <edit-config> RPC to configure the <port-profile> node in the urn:brocade.cpm:mgmt:brocade-port-profile namespace. Under the <port-profile> node, specify the following leaf elements.

   The following example deletes the security subprofile from the port-profile named vm1-port-profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1726" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>vm1-port-profile</name>
                <security-profile
                 xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete"/>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1726" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

4. To delete the FCoE subprofile, issue an <edit-config> RPC to configure the <port-profile> node in the urn:brocade.cpm:mgmt:brocade-port-profile namespace. Under the <port-profile> node, specify the following leaf elements.

   The following example deletes the FCoE subprofile from the port-profile named default.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1727" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>default</name>
                <fcoe-profile
                 xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete"/>
            </port-profile>
        </config>
    </edit-config>
```

```
</rpc>

<rpc-reply message-id="1727" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

5. To delete the QoS subprofile, issue an <edit-config> RPC to configure the <port-profile> node in the urn:brocade.cpm:mgmt:brocade-port-profile namespace. Under the <port-profile> node, specify the following leaf elements.

   The following example deletes the QoS subprofile from the port-profile named vm1-port-profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1728" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>vm1-port-profile</name>
                <qos-profile
                 xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete"/>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1728" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example shows how to allow non-profiled-macs on profiled ports. This only functions with the default port-profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1728" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>default</name>
                <allow>
                    <nonprofiledmacs></nonprofiledmacs>
                </allow>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1728" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <ok/>
</rpc-reply>
```

The following example shows how to restrict flooding. This is only applicable for default port-profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1728" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
```

```
        <target>
            <running/>
        </target>
        <config>
            <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <name>default</name>
                <restrict-flooding-container>
                    <restrict-flooding></restrict-flooding>
                </restrict-flooding-container>
            </port-profile>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1728" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <ok/>
</rpc-reply>
```

Obtaining the AMPP operational data

This section contains procedures for obtaining AMPP port-profiles operational data using NETCONF interfaces.

Obtaining the port-profile status

The following example returns status for the auto-VM_Network port-profile for all MAC associations in the activated state. In the first call, the <last-received-port-profile>/<profile-mac> element is set to 00:00:00:00:00. The output returns the first associated MAC address and the <has-more> element is set to true.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1730">
    <get-port-profile-status
     xmlns="urn:brocade.com:mgmt:brocade-port-profile-ext">
        <port-profile-name>auto-VM_Network</port-profile-name>
        <port-profile-status>activated</port-profile-status>
        <last-received-port-profile-info>
            <profile-name>auto-VM_Network</port-profile-name>
            <profile-mac>00:00:00:00:00:00</port-profile-mac>
        </last-received-port-profile-info>
    </get-port-profile-status>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1730">
    <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile-ext">
        <name>auto-VM_Network</name>
        <ppid>9</ppid>
        <is-active>true</is-active>
        <mac-association>
            <mac>00:50:56:b3:00:01</mac>
            <applied-interface>
                <interface-type>tengigabitethernet</interface-type>
                <interface-name>9/0/53</interface-name>
            </applied-interface>
        </mac-association>
        <has-more>true</has-more>
    </port-profile>
```

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

The following example takes as input the MAC association returned by the previous call. The output returns the next MAC association for this port-profile.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1731">
    <get-port-profile-status
     xmlns="urn:brocade.com:mgmt:brocade-port-profile-ext">
        <port-profile-name>auto-VM_Network</port-profile-name>
        <port-profile-status>activated</port-profile-status>
        <last-received-port-profile-info>
            <profile-name>auto-VM_Network</port-profile-name>
            <profile-mac>00:50:56:b3:00:01</port-profile-mac>
        </last-received-port-profile-info>
    </get-port-profile-status>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1731">
    <port-profile xmlns="urn:brocade.com:mgmt:brocade-port-profile-ext">
        <name>auto-VM_Network</name>
        <ppid>9</ppid>
        <is-active>true</is-active>
        <mac-association>
            <mac>00:50:56:b3:00:02</mac>
            <applied-interface>
                <interface-type>tengigabitethernet</interface-type>
                <interface-name>9/0/53</interface-name>
            </applied-interface>
        </mac-association>
        <has-more>true</has-more>
    </port-profile>
</rpc-reply>
```

Obtaining interface to port-profile mapping

Issue the <get-port-profile-for-intf> custom RPC to return information about port-profiles to which interfaces are applied. Include the <interface-type> and <interface-name> input parameters for the first interface for which you want port profiling information. Check the <has-more> element in the output to determine whether such information exists for more interfaces. If <has-more> returns true, re-issue the RPC and include the <last-received-interface-info> node element. Under <last-received-interface-info>, include the <interface-type> and <interface-name> elements from the previous invocation of the RPC. Repeat until <has-more> returns false.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1732">
    <get-port-profile-for-intf
     xmlns="urn:brocade.com:mgmt:brocade-interface-ext"/>
        <interface-type>tengigabitethernet</interface-type>
        <interface-name>9/0/53</interface-name>
    </get-port-profile-for-intf>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1732">
    <interface xmlns="urn:brocade.com:mgmt:brocade-port-profile-ext">
        <interface-type>tengigabitethernet</interface-type>
        <interface-name>9/0/53</interface-name>
        <port-profile>
            <name>auto-VM_Network</name>
        </port-profile>
    </interface>
```

```
    <has-more>true</has-more>
</rpc-reply>

<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1732">
    <get-port-profile-for-intf
     xmlns="urn:brocade.com:mgmt:brocade-interface-ext"/>
        <last-recevied-interface-info>
            <interface-type>tengigabiethernet</interface-type>
            <interface-name>9/0/53</interface-name>
        </last-recevied-interface-info>
    </get-port-profile-for-intf>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1732">
    <interface xmlns="urn:brocade.com:mgmt:brocade-port-profile-ext">
        <interface-type>tengigabitethernet</interface-type>
        <interface-name>9/0/54</interface-name>
        <port-profile>
            <name>auto-for_iscsi</name>
        </port-profile>
    </interface>
    <has-more>false</has-more>
</rpc-reply>
```

Configure the port-profile-port on the physical interface.

The following example configures the port-profile assignment from interface 1/0/1.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1720" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
                    <port-profile-port/>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1720" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting a port-profile-port

The following example deletes the port-profile assignment from interface 1/0/1.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1721" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
```

```
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
                    <port-profile-port
                     xmlns="urn:brocade.com:mgmt:brocade-port-profile"
                     xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1721" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the basic port-profile-domain

This task creates the port-profile-domain which is the basis for all port-profile-domain options.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-domain
mlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile-domain-name>TENANT_1</port-profile-domain-name>
            </port-profile-domain>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Adding the port-profile to the port-profile-domain

This task adds the AMPP port-profile to the port-profile-domain. Refer to "Configuring AMPP port-profiles" on page 211.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-domain
xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile-domain-name>TENANT_1</port-profile-domain-name>
                <profile>
```

```
                <profile-name>PROFILE_1</profile-name>
            </profile>
        </port-profile-domain>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Deleting a port-profile from the port-profile-domain

This task deletes the AMPP port-profile from the port-profile-domain.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-domain
xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                <port-profile-domain-name>TENANT_1</port-profile-domain-name>
            <profile xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
nc:operation="delete">
                <profile-name>PROFILE_1</profile-name>
            </profile>
            </port-profile-domain>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Associating a port-profile-domain with an interface

This task associates the port-profile-domain with an interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
       <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
           <tengigabitethernet>
               <name>1/1/22</name>
               <port-profile-to-interface-associations
xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                   <port-profile-port>
                       <port-to-port-profile-domain-association>
                           <profile-domain-name>dom100</profile-domain-name>
                       </port-to-port-profile-domain-association>
```

```
                </port-profile-port>
            </port-profile-to-interface-associations>
        </tengigabitethernet>
    </interface>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting a port-profile-domain

This task deletes the port-profile-domain. Any associated AMPP port-profiles are not deleted from the switch.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-profile-domain
xmlns="urn:brocade.com:mgmt:brocade-port-profile" operation="delete">
                <port-profile-domain-name>TENANT_1</port-profile-domain-name>
            </port-profile-domain>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Obtaining the port-profile-domain status

Use the <get-config> RPC to retrieve the current configuration data and operational state data. Refer to "Retrieving configuration data" on page 11 and "Retrieving operational data" on page 15 for detailed instructions.

The following example retrieves the status from the TENANT_1 port-profile-domain.

```
<?xml version="1.0" encoding="UTF-8"?>
    <rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
        <get-config>
            <source>
                <running></running>
            </source>
            <filter type="subtree">
                <port-profile-domain
                   xmlns="urn:brocade.com:mgmt:brocade-port-profile">
                    <port-profile-domain-name>TENANT_1</port-profile-domain-name>
                </port-profile-domain>
            </filter>
        </get-config>
    </rpc>
```

```
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <data>
        <port-profile-domain xmlns="urn:brocade.com:mgmt:brocade-port-profile">
            <port-profile-domain-name>TENANT_1</port-profile-domain-name>
        </port-profile-domain>
    </data>
</rpc-reply>
```

Chapter

Configuring FCoE Interfaces 23

In this chapter

FCoE configuration with NETCONF overview

This chapter provides procedures for assigning a Fibre Channel over Ethernet (FCoE) map to port interfaces or port-channel interfaces, and for retrieving status information about FCoE interfaces and logins.

For a procedure for configuring an FCoE profile, refer to "Configuring FCoE profiles" on page 218.

Refer to the *Network OS Layer 2 Switching Configuration Guide* for conceptual information about FCoE including:

- An overview of what FCoE is and its purpose
- FCoE end-to-end operations
- How Network OS Layer 2 Ethernet supports FCoE through Layer 2 forwarding, VLAN tagging, incoming frame classification, congestion control and queueing, access control, trunking, and flow control
- How the FCoE Initialization Protocol (FIP) works
- How FCoE queueing works
- Configuring FCoE over LAG
- Operational guidelines for applying an FCoE map to a VLAG
- xSTP reconvergence

Through the NETCONF interface, you can perform the following operations on FCoE interfaces:

- Use the <edit-config> RPC to assign an FCoE map to an interface or LAG.
- Use the <get-config> RPC to validate configuration settings.
- Use the <fcoe-get-interface> custom RPC to return information about FCoE port interfaces.
- Use the <fcoe-get-login> custom RPC to return information about logged in FCoE devices.

NOTE
FCoE parameters are defined in the brocade-fcoe YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Configuring FCoE

This section presents the tasks necessary to configure FCoE interfaces and logical SANs.

Configuring logical FCoE ports

The following example shows how to configure additional logical FCoE ports.

```
<rpc message-id="1708" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <fcoe xmlns="urn:brocade.com:mgmt:brocade-fcoe">
                    <fcoe-enodes>384</fcoe-enodes>
                </fcoe>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring fabric maps

The following example shows how to configure fabric maps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <fcoe xmlns="urn:brocade.com:mgmt:brocade-fcoe">
                <fcoe-fabric-map>
                    <fcoe-fabric-map-name>default</fcoe-fabric-map-name>
                    <fcoe-fabric-map-vlan>1002</fcoe-fabric-map-vlan>
                    <fcoe-fabric-map-priority>4</fcoe-fabric-map-priority>
                    <fcoe-fabric-map-fcmap>0E:FC:11</fcoe-fabric-map-fcmap>
                </fcoe-fabric-map>
            </fcoe>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring FCoE logical SANs

This section presents the tasks required to configure FCoE logical SANs and manage those configurations.

Creating fabric maps for logical SANs

The following example shows how to configure fabric map for non default SANs

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <fcoe xmlns="urn:brocade.com:mgmt:brocade-fcoe">
                <fcoe-fabric-map>
                    <fcoe-fabric-map-name>SanA</fcoe-fabric-map-name>
                    <fcoe-fabric-map-vlan>1004</fcoe-fabric-map-vlan>
                    <fcoe-fabric-map-priority>4</fcoe-fabric-map-priority>
                    <fcoe-fabric-map-fcmap>0E:FC:04</fcoe-fabric-map-fcmap>
                    <fcoe-fip-advertisement>

<fcoe-fip-advertisement-interval>10000</fcoe-fip-advertisement-interval>
                    </fcoe-fip-advertisement>
                </fcoe-fabric-map>
            </fcoe>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring FCF Groups for FCoE logical SANs

The following example shows how to configure fcf groups for non default SAN:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <fcoe xmlns="urn:brocade.com:mgmt:brocade-fcoe">
                <fcoe-fabric-map>
                    <fcoe-fabric-map-name>SanA</fcoe-fabric-map-name>
                    <fcoe-fcf-map>
                        <fcf-map-name>rack-1</fcf-map-name>
                        <fcf-map-fcf-rbid>1</fcf-map-fcf-rbid>
                        <fcf-map-fif-rbid>

<fcf-map-fif-rbid-add>5,10-11,13-15,35</fcf-map-fif-rbid-add>
                        </fcf-map-fif-rbid>
                    </fcoe-fcf-map>
```

```
            </fcoe-fabric-map>
        </fcoe>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Assigning an Fabric map onto an interface

The following example applies the default FCoE map to the 1/0/1 interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <edit-config>
          <target>
              <running/>
          </target>
          <config>
              <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                  <tengigabitethernet>
                      <name>1/0/1</name>
                      <fcoeport xmlns="urn:brocade.com:mgmt:brocade-fcoe">
                          <fcoeport-map>default</fcoeport-map>
                      </fcoeport>
                  </tengigabitethernet>
              </interface>
          </config>
      </edit-config>
</rpc>

<rpc-reply message-id="1800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <ok/>
</rpc-reply>
```

Confirm the changes to the interface using the <get-config> RPC with a subtree filter to return only the <fcoeport> node information of the 1/0/1 interface.

The output returns the FCoE mapping association for the interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1801" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <get-config>
          <source>
              <running/>
          </source>
          <filter type="subtree">
              <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                  <tengigabitethernet>
                      <name>1/0/1</name>
                      <fcoeport xmlns="urn:brocade.com:mgmt:brocade-fcoe"/>
                  </tengigabitethernet>
              </interface>
          </filter>
      </get-config>
</rpc>

<rpc-reply message-id="1801" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
            <tengigabitethernet>
                <name>1/0/1</name>
                <fcoe:fcoeport>
                    <fcoe:fcoeport-map>default<fcoe:fcoeport-map>
                </fcoe:fcoeport>
            </tengigabitethernet>
        </interface>
</rpc-reply>
```

Assigning an FCoE Fabric-Map onto a LAG member

The following example applies the default FCoE map to port channel 5.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1802" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <edit-config>
            <target>
                <running/>
            </target>
            <config>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <port-channel>
                        <name>5</name>
                        <fcoeport xmlns="urn:brocade.com:mgmt:brocade-fcoe">
                            <fcoeport-map>default</fcoeport-map>
                        </fcoeport>
                    </port-channel>
                </interface>
            </config>
        </edit-config>
</rpc>

<rpc-reply message-id="1802" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <ok/>
</rpc-reply>
```

Confirm the changes to the interface using the <get-config> RPC with a subtree filter to return only the <fcoeport> node information for a specific port channel.

The output returns the FCoE mapping association for the interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1803" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <get-config>
            <source>
                <running/>
            </source>
            <filter type="subtree">
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <port-channel>
                        <name>5</name>
                        <fcoeport xmlns="urn:brocade.com:mgmt:brocade-fcoe"/>
                    </port-channel>
                </interface>
            </filter>
        </get-config>
</rpc>

<rpc-reply message-id="1803" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
            <port-channel>
                <name>5</name>
                <fcoe:fcoeport>
                    <fcoe:fcoeport-map>default<fcoe:fcoeport-map>
                </fcoe:fcoeport>
            <port-channel>
        </interface>
</rpc-reply>
```

Configuring default fabric map parameters

The following example shows how to configure parameters for default fabric-map:

Configuring VLAN

The following example shows how to configure VLAN for default fabric-map:

```
<rpc message-id="1710" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <fcoe xmlns="urn:brocade.com:mgmt:brocade-fcoe">
                <fcoe-fabric-map>
                    <fcoe-fabric-map-name>default</fcoe-fabric-map-name>
                    <fcoe-fabric-map-vlan>789</fcoe-fabric-map-vlan>
                </fcoe-fabric-map>
            </fcoe>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring fcmap

The following example shows how to configure fcmap for default fabric-map

```
<rpc message-id="1710" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <fcoe xmlns="urn:brocade.com:mgmt:brocade-fcoe">
                <fcoe-fabric-map>
                    <fcoe-fabric-map-name>default</fcoe-fabric-map-name>
                    <fcoe-fabric-map-fcmap>0E:FC:11</fcoe-fabric-map-fcmap>
                </fcoe-fabric-map>
            </fcoe>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring priority

The following example shows how to configure priority for default fabric-map

```
<rpc message-id="1710" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <fcoe xmlns="urn:brocade.com:mgmt:brocade-fcoe">
                <fcoe-fabric-map>
                    <fcoe-fabric-map-name>default</fcoe-fabric-map-name>
                    <fcoe-fabric-map-priority>4</fcoe-fabric-map-priority>
                </fcoe-fabric-map>
            </fcoe>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Adding FIF RBridge-ID

The following example shows how to add multiple RBridge-IDs to the configured fcf group:

```
<rpc message-id="1710" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <fcoe xmlns="urn:brocade.com:mgmt:brocade-fcoe">
                <fcoe-fabric-map>
                    <fcoe-fabric-map-name>SAN_M</fcoe-fabric-map-name>
                    <fcoe-fcf-map>
                        <fcf-map-name>RACK</fcf-map-name>
                        <fcf-map-fif-rbid>
                            <fcf-map-fif-rbid-add>12-15</fcf-map-fif-rbid-add>
                        </fcf-map-fif-rbid>
                    </fcoe-fcf-map>
                </fcoe-fabric-map >
            </fcoe>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Removing FIF RBridge-ID

The following example shows how to remove multiple RBridge-IDs from the configured fcf group:

```
<rpc message-id="1710" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <fcoe xmlns="urn:brocade.com:mgmt:brocade-fcoe">
                <fcoe-fabric-map>
                    <fcoe-fabric-map-name>SAN_M</fcoe-fabric-map-name>
                    <fcoe-fcf-map>
                        <fcf-map-name>RACK</fcf-map-name>
                        <fcf-map-fif-rbid>
                            <fcf-map-fif-rbid-remove>28-30</fcf-map-fif-rbid-remove>
                        </fcf-map-fif-rbid>
                    </fcoe-fcf-map>
                </fcoe-fabric-map >
            </fcoe>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Obtaining FCoE status

The following Custom RPCs exist for returning FCoE information:

- <fcoe-get-interface> returns information about FCoE port interfaces
- <fcoe-get-login> returns information about logged in FCoE devices

Obtaining FCoE port interface information

The following example returns FCoE port interface information for port 1/0/1 and requests statistical information be shown in the output.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1804">
    <fcoe-get-interface xmlns="urn:brocade.com:mgmt:brocade-fcoe-ext">
        <fcoe-intf-name>1/0/1</fcoe-intf-name>
        <fcoe-intf-include-stats>true</fcoe-intf-include-stats>
    </fcoe-get-interface>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1804">
    <fcoe-intf-list xmlns="urn:brocade.com:mgmt:brocade-fcoe-ext">
        <fcoe-intf-fcoe-port-id>1/0/1</fcoe-intf-fcoe-port-id>
        <fcoe-intf-port-type>VF</fcoe-intf-port-type>
    (output truncated)
```

Obtaining FCoE login information

The following example returns information for all FCoE devices logged into routing bridge 13.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1805">
    <fcoe-get-login xmlns="urn:brocade.com:mgmt:brocade-fcoe-ext">
        <fcoe-login-rbridge-id>13</fcoe-login-rbridge-id>
    </fcoe-get-login>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1805">
    <fcoe-login-list xmlns="urn:brocade.com:mgmt:brocade-fcoe-ext">
        <fcoe-login-session-mac>00:50:56:b3:00:01</fcoe-login-session-mac>
        <fcoe-login-fcoe-interface-name>1/0/1</fcoe-login-fcoe-interface-name>
    (output truncated)
```

Enabling Duplicate WWN login

Enabling Duplicate WWN login mode for a particular node/switch in a fabric

The following example enables duplicate WWN login mode for a particular node/switch in a fabric

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <fabric xmlns="urn:brocade.com:mgmt:brocade-fabric-service">
                    <login-policy>
                        <duplicateWWN>
                            <precedence>new-login</precedence>
                        </duplicateWWN>
                    </login-policy>
                </fabric>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Configuring FlexPort 24

In this chapter

FlexPort overview

The FlexPort feature allows some of the ports to transmit data as either 10G Ethernet or Fibre Channel, and to be changed from one type to the other without requiring a reboot.

This chapter provides procedures for configuring Flexport using the NETCONF interface. Refer to the *Network OS Layer 2 Switching Configuration Guide* for more information.

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

- Use the <edit-config> remote procedure call (RPC) to configure Flexport.
- Use the <get-config> RPC to verify all or part of the Flexport configuration.

Flexport parameters are defined in the brocade-hardware YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*.

Configuring FlexPort

You can configure some of the ports to transmit either Ethernet or Fibre Channel.

The FlexPort feature is set to Ethernet by default. You should only need to perform this task to switch to Fibre Channel, or back to Ethernet.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
      <target>
         <running/>
      </target>
      <config>
         <hardware xmlns="urn:brocade.com:mgmt:brocade-hardware">
            <flexport>
               <id>1/0/5</id>
               <flexport_type>
                  <type>fibre-channel</type>
                  <instance>0</instance>
               </flexport_type>
            </flexport>
            <connector>
               <name>1/0/49</name>
```

```
                <sfp>
                    <breakout>true</breakout>
                </sfp>
            </connector>
            <connector-group>
                <id>1/0/1</id>
                <speed>HighMixed</speed>
            </connector-group>
        </hardware>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Retrieving the FlexPort list

Use the <get-flexport> to retrieve the flexPort list.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
   <get-flexports xmlns="urn:brocade.com:mgmt:brocade-hardware"/>
   </get-flexports>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <flexport-list xmlns="urn:brocade.com:mgmt:brocade-hardware">
        <port-id>7/0/1</port-id>
        <port-id>7/0/2</port-id>
        <port-id>7/0/3</port-id>
        <port-id>7/0/4</port-id>
        <port-id>7/0/5</port-id>
        <port-id>7/0/7</port-id>
    </flexport-list>
</rpc-reply>
```

Chapter

Configuring VLANs

25

In this chapter

VLAN configuration with NETCONF overview

This chapter provides procedures for configuring VLANs using the NETCONF interface. Refer to the *Network OS Layer 2 Switching Configuration Guide* for the following related information:

- An overview of VLANs
- An explanation of how ingress VLAN filtering works
- Conceptual information about VLAN classifications
- VLAN configuration guidelines and restrictions
- The default VLAN configuration
- Overview of Private VLANs

Through the NETCONF interface, you can perform the following operations on VLANs:

- Use the <edit-config> RPC to configure VLANs.
- Use the <get-config> RPC to validate configuration settings.
- Use the <get-interface-detail> custom RPC to return information about a VLAN-associated port interface.
- Use the <get-vlan-brief> custom RPC to return information about a specific VLAN.

VLAN parameters are defined in the brocade-interface and brocade-vlan YANG modules. For structural maps of these YANG modules, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all VCS Fabric management parameters, refer to the brocade-interface.yang, and brocade-vlan.yang files.

VLAN configuration and management

NOTE
Use the <bna-config-cmd> RPC to save your configuration changes.

Enabling and disabling an interface port

NOTE
DCB interfaces are enabled by default in Brocade VCS Fabric mode.

NOTE
DCB interfaces do not support auto-negotiation of Ethernet link speeds. The DCB interfaces support 100-Gigabit Ethernet, 40-Gigabit Ethernet, 10-Gigabit Ethernet, and Gigabit Ethernet.

The following example enables port 22/0/1.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1901" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete"/>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1901" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To disable the interface, include the <shutdown> element without the delete operation.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1902" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown/>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1902" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the MTU on an interface port

To configure the maximum transmission unit (MTU) on an interface port.

The following example enables port 22/0/1 and sets its MTU value to 4200.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1903" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                    <mtu>4200</mtu>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1903" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Creating a VLAN interface

The following example creates VLAN 1010.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>1010</name>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling STP on a VLAN

To enable STP for a VLAN, select the type of STP for the VLAN, and then enable spanning tree on that VLAN.

The following example enables MSTP on VLAN 1002 using one RPC.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1905" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns=”urn:brocade.com:mgmt:brocade-xstp”>
                    <mstp>
                        <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    </mstp>
                </spanning-tree>
            </protocol>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>1002</name>
                        <spanning-tree xmlns=”urn:brocade.com:mgmt:brocade-xstp”/>
                            <stp-shutdown
                             xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                             operation="delete"/>
                        </spanning-tree>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>
<rpc-reply message-id="1905" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling STP on a VLAN

The following example disables STP on VLAN 1002.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1906" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>1002</name>
                        <spanning-tree xmlns=”urn:brocade.com:mgmt:brocade-xstp”>
                            <stp-shutdown/>
                        </spanning-tree>
```

```
                </vlan>
            </interface>
        </interface-vlan>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="1906" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring an interface port as a Layer 2 switch port

The following example configures 10-Gigabit Ethernet port 22/0/1 as a Layer 2 switch port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1907" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
                    <name>1/0/1</name>
                    <switchport-basic><basic/></switchport-basic>
                    <switchport>
                        <mode>
                            <vlan-mode>trunk</vlan-mode>
                        </mode>
                    </switchport>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1907" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example returns status details about port 22/0/1.

```
<rpc message-id="1908" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-interface-detail xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type xmlns="urn:brocade.com:mgmt:brocade-interface">
            TenGigabitEthernet</interface-type>
        <interface-name xmlns="urn:brocade.com:mgmt:brocade-interface">
            22/0/1</interface-name>
    </get-interface-detail>
</rpc>

rpc-reply message-id="1908" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type xmlns="urn:brocade.com:mgmt:brocade-interface">
            TenGigabitEthernet</interface-type>
        <interface-name xmlns="urn:brocade.com:mgmt:brocade-interface">1/0/1
            </interface-name>
        <ifindex
```

```
          xmlns="urn:brocade.com:mgmt:brocade-interface">67174401</ifindex>
         <mtu xmlns="urn:brocade.com:mgmt:brocade-interface">2500</mtu>
         <ip-mtu xmlns="urn:brocade.com:mgmt:brocade-interface">2500</ip-mtu>
         <if-name xmlns="urn:brocade.com:mgmt:brocade-interface"></if-name>
         <if-state xmlns="urn:brocade.com:mgmt:brocade-interface">up</if-state>
         <line-protocol-state xmlns="urn:brocade.com:mgmt:brocade-interface">up
            </line-protocol-state>
(output truncated)
```

Configuring an interface port as an access interface

To configure the interface as an access interface requires two RPCs. The first RPC configures the port as a Layer 2 interface; the second RPC configures the interface port as an access interface.

1. The following example configures 10-Gigabit Ethernet port 1/0/1 as an access interface port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1909" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
                    <name>1/0/1</name>
                    <switchport-basic><basic/></switchport-basic>
                </tengigabitethernet>
            </interface>
        </config>
    <edit-config>
</rpc>

<rpc-reply message-id="1909" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. Issue another <edit-config> RPC to configure the <interface> node in the urn:brocade.com:mgmt:brocade-interface namespace.
3. Under the <interface> node, specify the interface type; that is, <gigabitethernet>, <tengigabitethernet>, <fortygigabitethernet>, or <hundredgigabitethernet>.
4. Under the <gigabitethernet>, <tengigabitethernet>, <fortygigabitethernet>, or <hundredgigabitethernet> element, include the following elements:
    a. In the <name> element, specify the port name.
    b. Include the <switchport>/<access> hierarchy of node elements.
5. Under the <access> node element, specify the <access-vlan> leaf element containing the VLAN ID to configure a layer 2 switch port as an access interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1910" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
```

```
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <switchport>
                        <access>
                            <accessvlan>20</accessvlan>
                        </access>
                    </switchport>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1910" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring an interface port as a trunk interface

This procedure performs the following functions:

- Enables the interface.
- Specifies trunk mode.
- Specifies whether one, all, or none of the VLAN interfaces are allowed to transmit and receive through the DCB interface.

To configure the interface as a trunk interface requires two RPCs. The first RPC configures the port as a Layer 2 interface; the second RPC configures the interface port as a trunk interface.

1. Issue an <edit-config> RPC to configure the <interface> node in the urn:brocade.com:mgmt:brocade-interface namespace.
2. Under the <interface> node, specify the interface type element (<gigabitethernet>, <tengigabitethernet>, <fortygigabitethernet>, or <hundredgigabitethernet>).
3. Under the <gigabitethernet>, <tengigabitethernet>, <fortygigabitethernet>, or <hundredgigabitethernet> element, include the following elements:
    a. In the <name> element, specify the port name in [*rbridge-id/*]*slot*/*port* format.
    b. In the <shutdown> element, include the delete operation in the element opening tag to enable the interface port.
    c. Include the <switchport>/<basic> elements to configure the port as a Layer 2 interface.

    The following example configures 10-Gigabit Ethernet port 1/0/1 as a Layer 2 switch port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1911" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
<config>
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
      <tengigabitethernet xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <name>1/0/1</name>
        <switchport-basic><basic/></switchport-basic>
      </tengigabitethernet>
```

```
    </interface>
  </config>
   </edit-config>
</rpc>

<rpc-reply message-id="1911" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

4. Issue another <edit-config> RPC to configure the <interface> node in the urn:brocade.com:mgmt:brocade-interface namespace.
5. Under the <interface> node, specify the same interface type element you specified in step 2 (<gigabitethernet>, <tengigabitethernet>, <fortygigabitethernet>, or <hundredgigabitethernet>).
6. Under the <gigabitethernet>, <tengigabitethernet>, <fortygigabitethernet>, or <hundredgigabitethernet> node element, include the following elements:
    a. In the <name> element, specify the same port name you specified in step 3.
    b. Include the <switchport>/<mode> hierarchy of node elements. Under the <mode> element, specify the <vlan-mode> element as "trunk" to specify trunk mode.
    c. Include the <trunk>/<allowed> node elements containing further XML elements that define which VLAN interfaces are allowed to transmit and receive through the DCB interface.

    The following example XML fragment allows VLAN 30 to transmit and receive through the DCB interface:

```
<trunk>
    <allowed>
        <vlan>
            <add>30</add>
        </vlan>
    </allowed>
</trunk
```

    The following XML fragment allows all VLANs to transmit and receive through the DCB interface.

```
<trunk>
    <allowed>
        <vlan>
            <all/>
        </vlan>
    </allowed>
</trunk
```

    The following fragment allows all VLANs except VLAN 11 to transmit and receive through the DCB interface.

```
<trunk>
    <allowed>
        <vlan>
            <except>11</except>
        </vlan>
    </allowed>
</trunk>
```

    The following fragment allows no VLAN to transmit or receive through the DCB interface.

```
<trunk>
    <allowed>
        <vlan>
            <none/>
        </vlan>
    </allowed>
</trunk>
```

The following complete example <edit-config> RPC enables trunk mode on port 22/0/1 and allows only VLAN 30 to transmit and receive through the DCB interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1912" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <switchport>
                        <mode>
                            <vlan-mode>trunk</vlan-mode>
                        </mode>
                        <trunk>
                            <allowed>
                                <vlan>
                                    <add>30</add>
                                </vlan>
                            </allowed>
                        </trunk>
                    </switchport>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1912" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling a VLAN on a trunk interface

To disable a VLAN on a trunk interface, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete"/>
                    <switchport>
```

```
                        <mode>
                            <vlan-mode>trunk</vlan-mode>
                        </mode>
                    </switchport>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Issue an additional RPC to remove the VLAN ranges from the trunk port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1914" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete"/>
                    <switchport>
                        <trunk>
                            <allowed>
                                <vlan>
                                    <remove>30</remove>
                                </vlan>
                            </allowed>
                        </mode>
                    </switchport>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1914" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring a VLAN classifier rule

To configure an ARP-based VLAN classifier rule

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1915" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <vlan xmlns="urn:brocade.com:mgmt:brocade-vlan">
                <classifier>
```

```
                <rule>
                    <ruleid>5</ruleid>
                    <proto>
                        <proto-val>arp</proto-val>
                        <encap>ethv2</encap>
                    </proto>
                </rule>
            </classifier>
        </vlan>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1915" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

NOTE

Refer to the *Network OS Command Reference* for complete information on all the protocols available for specifying VLAN classifier rules.

Configuring MAC address-based VLAN classifier rules

To configure a MAC address-based VLAN classifier rule,

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1916" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <vlan xmlns="urn:brocade.com:mgmt:brocade-vlan">
                <classifier>
                    <rule>
                        <ruleid>5</ruleid>
                        <mac>
                            <address>0008.744c.7fid</address>
                        </mac>
                    </rule>
                </classifier>
            </vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1916" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Creating a VLAN classifier group and adding rules

VLAN classifier groups (1 through 16) can contain any number of VLAN classifier rules.

The following example creates classifier group 1 and adds rule 1 to it.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1917" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <vlan xmlns="urn:brocade.com:mgmt:brocade-vlan">
                <classifer>
                    <group>
                        <groupid>1</groupid>
                        <oper>add</oper>
                        <rule-name>Rule</rule-name>
                        <ruleid>1</ruleid>
                    </group>
                </classifier>
            </vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1917" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting a VLAN classifier rule

VLAN classifier groups (1 through 16) can contain any number of VLAN classifier rules.

The following example deletes rule 1 from classifier group 1.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1918" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <vlan xmlns="urn:brocade.com:mgmt:brocade-vlan">
                <classifier>
                    <group>
                        <groupid>1</groupid>
                        <oper>delete</oper>
                        <rule-name>Rule</rule-name>
                        <ruleid>1</ruleid>
                    </group>
                </classifier>
            </vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1918" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Activating a VLAN classifier group with an interface port

To associate a VLAN classifier group with an interface port

Group 1 and VLAN 2 are used in the following example. The example assumes that VLAN 2 already exists.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1919" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/10</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete"/>
                    <vlan xmlns="urn:brocade.com:mgmt:brocade-vlan">
                        <classifier>
                            <activate>
                                <group>
                                    <groupid>1</groupid>
                                    <vlan-name>vlan</vlan-name>
                                    <vlan>2</vlan>
                                </group>
                            </activate>
                        </classifier>
                    </vlan>
                <tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1919" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Obtaining VLAN information

The following Custom RPCs return information about VLANs:

- <get-interface-detail> returns information about the associated port interface.
- <get-vlan-brief> returns information about a specific VLAN.

Obtaining port interface information for one port

To return information about an associated port interface, issue the <get-interface-detail> custom RPC from the urn:brocade.com:mgmt:brocade-interface-ext namespace and specify the following input elements:

- In the <interface-type> element in the urn:brocade.com:mgmt:brocade-interface namespace, specify the interface type—For example “tengigabitethernet”.
- In the <interface-name> element from the urn:brocade.com:mgmt:brocade-interface namespace, specify the port name in the [*rbridge-id*/]*slot*/*port* format.

```
<rpc message-id="1920" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-interface-detail xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type xmlns="urn:brocade.com:mgmt:brocade-interface">
```

```
            tengigabitethernet</interface-type>
        <interface-name xmlns="urn:brocade.com:mgmt:brocade-interface">
            1/0/1</interface-name>
    </get-interface-detail>
</rpc>

rpc-reply message-id="1920" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type xmlns="urn:brocade.com:mgmt:brocade-interface">
            TenGigabitEthernet</interface-type>
        <interface-name xmlns="urn:brocade.com:mgmt:brocade-interface">1/0/1
            </interface-name>
        <ifindex xmlns="urn:brocade.com:mgmt:brocade-interface">67174401</ifindex>
        <mtu xmlns="urn:brocade.com:mgmt:brocade-interface">2500</mtu>
        <ip-mtu xmlns="urn:brocade.com:mgmt:brocade-interface">2500</ip-mtu>
        <if-name xmlns="urn:brocade.com:mgmt:brocade-interface"></if-name>
        <if-state xmlns="urn:brocade.com:mgmt:brocade-interface">up</if-state>
        <line-protocol-state xmlns="urn:brocade.com:mgmt:brocade-interface">up
            </line-protocol-state>
(output truncated)
```

Obtaining port information for a sequence of ports

To retrieve information for a sequence ports, issue the <get-interface-detail> RPC multiple times and use the <last-received-interface> node element on input.

Before issuing the <get-interface-detail> RPC, check the output of the previous invocation to determine whether the <has-more> boolean element is set to “true”. If so, information is available for additional interfaces;

```
<rpc message-id="1921" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-interface-detail xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type xmlns="urn:brocade.com:mgmt:brocade-interface">
            tengigabitethernet</interface-type>
        <interface-name xmlns="urn:brocade.com:mgmt:brocade-interface">
            1/0/1</interface-name>
    </get-interface-detail>
</rpc>

rpc-reply message-id="1921" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <interface-type xmlns="urn:brocade.com:mgmt:brocade-interface">
            tengigabitethernet</interface-type>
        <interface-name xmlns="urn:brocade.com:mgmt:brocade-interface">1/0/1
            </interface-name>
        <ifindex xmlns="urn:brocade.com:mgmt:interface">67174401</ifindex>
        <mtu xmlns="urn:brocade.com:mgmt:brocade-interface">2500</mtu>
        <ip-mtu xmlns="urn:brocade.com:mgmt:brocade-interface">2500</ip-mtu>
        <if-name xmlns="urn:brocade.com:mgmt:brocade-interface"></if-name>
        <if-state xmlns="urn:brocade.com:mgmt:brocade-interface">up</if-state>
        <line-protocol-state xmlns="urn:brocade.com:mgmt:brocade-interface">up
            </line-protocol-state>
(output truncated)
    </interface>
    <has-more>true</has-more>
</rpc-reply>
```

Obtaining VLAN information for one VLAN

To return information about a specific VLAN, issue the <get-vlan-brief> custom RPC from the urn:brocade.com:mgmt:brocade-interface-ext namespace and specify the VLAN in the <vlan-id> input parameter.

```
<rpc message-id="1922" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-vlan-brief xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <vlan-id>20</vlan-id>
    </get-vlan-brief>
</rpc>

rpc-reply message-id="1922" 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>tengigabitethernet</interface-type>
            <interfce-name>66/0/10</interface-name>
            <tag>tagged</tag>
        </interface>
    </vlan>
    <last-vlan-id>20</last-vlan-id>
    <has-more>true</has-more>
</rpc>
```

Obtaining VLAN information for multiple VLANs

To retrieve information for a sequence of VLANs, issue the <get-vlan-brief> RPC multiple times and specify the last received vlan ID on input.

Before issuing the <get-vlan-brief> RPC, check the output of the previous invocation to determine whether the <has-more> boolean element is set to “true”. If so, information about additional VLANs is available;

```
<rpc message-id="1923" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-vlan-brief xmlns="urn:brocade.com:mgmt:brocade-interface-ext">
        <last-received-vlan-id>20</last-recevied-vlan-id>
    </get-vlan-brief>
</rpc>

<rpc-reply message-id="1923" 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</vlan-state>
        <interface>
            <interface-type>tengigabitethernet</interface-type>
            <interface-name>66/0/10</interface-name>
            <tag>tagged</tag>
        </interface>
    </vlan>
    <last-vlan-id>30</last-vlan-id>
    <has-more>true</has-more>
    </rpc-reply>
```

Specifying or disabling the aging time for MAC addresses

The following example sets the aging time to 600 seconds.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1924" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac-address-table
            xmlns="urn:brocade.com:mgmt:brocade-mac-address-table">
                <aging-time>600</aging-time>
            </mac-address-table>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1924" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Adding static addresses to the MAC address table

To add a static address to the MAC address table

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1925" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac-address-table
            xmlns="urn:brocade.com:mgmt:brocade-mac-address-table">
                <static>
                    <mac-address>0011.2222.3333</mac-address>
                    <forward>forward</forward>
                    <interface-type>tengigabitethernet</interface-type>
                    <interface-name>66/0/1</interface-name>
                    <vlan>vlan</vlan>
                    <vlan-id>100</vlan-id>
                </static>
            </mac-address-table>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1925" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling dynamic MAC learning globally

The following example is to disable dynamic MAC address learning globally on a switch and enable CML.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac-address-table
xmlns="urn:brocade.com:mgmt:brocade-mac-address-table">
                <learning-mode>conversational</learning-mode>
            </mac-address-table>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling conversational MAC learning on an interface

The following example is to disable MAC learning and enable CML for the entire interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface>
                <fortygigabitethernet>
                    <name>1/2/2</name>
                    <mac-learning>
                        <mac-learn-disable>
                            <vlan>
                                <mac-learning-vlan-add>1000</mac-learning-vlan-add>

<mac-learning-vlan-remove>100</mac-learning-vlan-remove>
                            </vlan>
                        </mac-learn-disable>
                    </mac-learning>
                </fortygigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Retrieving the operational data for a given MAC entry

Use the <get-mac-address-table> custom RPC to the operational data for a given MAC entry.

```
<rpc message-id="303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <get-mac-address-table
xmlns="urn:brocade.com:mgmt:brocade-mac-address-table">
</rpc>

<rpc-reply message-id="303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <mac-address-table xmlns="urn:brocade.com:mgmt:brocade-mac-address-table">
        <vlanid>100</vlanid>
        <mac-address>00:11:22:22:33:33</mac-address>
        <mac-type>static</mac-type>
        <mac-state>inactive</mac-state>
        <forwarding-interface>
            <interface-type>port-channel</interface-type>
            <interface-name>25</interface-name>
        </forwarding-interface>
    </mac-address-table>
</rpc-reply>
```

Configuring a private VLAN

This procedure configures the PVLAN and associates the secondary VLAN with the primary VLAN.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>200</name>
                        <private-vlan>
                            <pvlan-type-leaf>primary</pvlan-type-leaf>
                        </private-vlan>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring a community PVLAN

This procedure configures a community PVLAN.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="55"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
```

```
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>200</name>
                        <private-vlan>
                            <pvlan-type-leaf>community</pvlan-type-leaf>
                        </private-vlan>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>>

<rpc-reply message-id="55" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring an isolated PVLAN

This procedure configures an isolated PVLAN.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="55"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>200</name>
                        <private-vlan>
                            <pvlan-type-leaf>isolated</pvlan-type-leaf>
                        </private-vlan>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="55" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Displaying PVLAN information

Use the <get-config> RPC to retrieve the current configuration data and operational state data. Refer to "Retrieving configuration data" on page 11 and "Retrieving operational data" on page 15 for detailed instructions.

Chapter

Resolving Repeated MAC-Moves 26

In this chapter

Resolving Repeated MAC-Moves with NETCONF overview

This chapter provides procedures for configuring mac-moves using the NETCONF interface. Refer to the *Network OS Layer 2 Switching Configuration Guide* for the following related information:

- Overview of resolving repeated MAC-moves
- MAC-move detection
- MAC consistency-check

Through the NETCONF interface, you can perform the following operations on VLANs:

- Use the <edit-config> RPC to resolve repeated mac-moves.
- Use the <get-config> RPC to validate configuration settings.

MAC-Move parameters are defined in the YANG modules. For structural maps of these YANG modules, refer to the *Network OS YANG Reference Manual*.

Configuring MAC-move detection

The following example shows how to configure MAC-move detection:

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac-address-table
xmlns="urn:brocade.com:mgmt:brocade-mac-address-table">
                <mac-move>
                    <mac-move-detect-enable></mac-move-detect-enable>
                    <mac-move-limit>50</mac-move-limit>
                </mac-move>
            </mac-address-table>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring MAC consistency check

The following example shows how to configure MAC consistency check:

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac-address-table
xmlns="urn:brocade.com:mgmt:brocade-mac-address-table">
                <consistency-check>

<mac-consistency-check-suppress></mac-consistency-check-suppress>

<mac-consistency-check-interval>150</mac-consistency-check-interval>
                </consistency-check>
            </mac-address-table>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Configuring Link-State Tracking (LST) 27

In this chapter

LST overview

Link-state tracking (LST) is a relationship that you can configure and enable for redundant-link networking topology, to prevent traffic loss between upstream and downstream RBridge links.

Configuring LST for independent RBridge

These topics explain how to configure link-state tracking (LST) for non-VCS topologies.

Configuring LST for single-link topologies

The following example shows how to configure LST for single-link topologies:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <track>
                        <track_enable></track_enable>
                        <min-link>1</min-link>
                        <interface>
                            <track-interface-type>ethernet</track-interface-type>
                            <track-interface-name>1/0/7</track-interface-name>
                        </interface>
                    </track>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring LST for multiple-uplink topologies

The following example shows how to configure LST for multiple-link topologies:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <track>
                        <track_enable></track_enable>
                        <min-link>1</min-link>
                        <interface>
                            <track-interface-type>ethernet</track-interface-type>
                            <track-interface-name>1/0/7</track-interface-name>
                        </interface>
                        <interface>
                            <track-interface-type>ethernet</track-interface-type>
                            <track-interface-name>1/0/7</track-interface-name>
                        </interface>
                    </track>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring LST for multiple downlink/uplink topologies

The following example shows how to configure LST for multiple downlink/uplink topologies:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <track>
                        <track_enable></track_enable>
                        <min-link>1</min-link>
```

```
                    <interface>
                        <track-interface-type>ethernet</track-interface-type>
                        <track-interface-name>1/0/7</track-interface-name>
                    </interface>
                    <interface>

<track-interface-type>port-channel</track-interface-type>
                        <track-interface-name>2021</track-interface-name>
                    </interface>
                </track>
            </tengigabitethernet>
        </interface>
    </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring LST for VCS fabrics

These topics explain how to configure link-state tracking (LST) for topologies that include one or more VCS fabrics.

Configuring LST on a VCS cluster

The following example shows how to configure LST for a VCS cluster:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <port-channel>
                    <name>1</name>
                    <track>
                        <track_enable></track_enable>
                        <min-link>1</min-link>
                        <interface>
                            <track-interface-type>ethernet</track-interface-type>
                            <track-interface-name>1/0/5</track-interface-name>
                        </interface>
                        <interface>
                            <track-interface-type>ethernet</track-interface-type>
                            <track-interface-name>2/0/44</track-interface-name>
                        </interface>
                    </track>
                </port-channel>
            </interface>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring LST on a VCS cluster and an independent RBridge

The following example shows how to configure LST on a VCS cluster and an independent RBridge:

1. Log in to the independent RBridge and configure the following.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <track>
                        <track_enable></track_enable>
                        <min-link>1</min-link>
                        <interface>

<track-interface-type>port-channel</track-interface-type>
                            <track-interface-name>2021</track-interface-name>
                        </interface>
                    </track>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. Log in to the VCS principal RBridge and configure the following.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <port-channel>
                    <name>1</name>
                    <track>
                        <track_enable></track_enable>
                        <min-link>1</min-link>
                        <interface>
                            <track-interface-type>ethernet</track-interface-type>
                            <track-interface-name>1/0/5</track-interface-name>
                        </interface>
```

```
                    <interface>
                        <track-interface-type>ethernet</track-interface-type>
                        <track-interface-name>2/0/44</track-interface-name>
                    </interface>
                </track>
            </port-channel>
        </interface>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Disabling LST

The following example shows how to disable LST:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <delete-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <track>
                        <track_enable></track_enable>
                        <interface>

<track-interface-type>port-channel</track-interface-type>
                            <track-interface-name>2021</track-interface-name>
                        </interface>
                    </track>
                </tengigabitethernet>
            </interface>
        </config>
    </delete-config>
</rpc>
```

The following example shows how to delete the complete LST configuration from the interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <delete-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <track>
                        <remove>all</remove>
                    </track>
                </tengigabitethernet>
```

```
            </interface>
        </config>
    </delete-config>
</rpc>
```

Chapter

Configuring VXLAN Overlay Gateways for NSX Controller Deployments 28

In this chapter

Introduction to VXLAN overlay gateways with NSX Controller

Virtual Extensible LAN (VXLAN) is an overlay network that extends Layer 2 domains over Layer 3 networks. The overlay network supports elastic compute architectures, enabling network engineers to scale a cloud computing environment while logically isolating cloud applications and tenants.

VXLAN extends the virtual LAN (VLAN) address space by adding a 24-bit segment ID and increasing the number of available VLAN IDs to 16 million, in a Virtual Fabrics context. The VXLAN segment ID in each frame differentiates individual logical networks, allowing millions of isolated Layer 2 VXLAN networks to coexist on a common Layer 3 infrastructure. As with VLANs, only virtual machines (VMs) within the same logical network can communicate with each other.

VXLAN creates large-scale, isolated virtual Layer 2 networks for virtualized and multi-tenant environments by encapsulating frames in VXLAN packets. Frame encapsulation is performed by means of a VXLAN Network Identifier (VNI) tunnel endpoint (VTEP), which originates or terminates VXLAN tunnels.

Because not all devices and servers are capable of sending or receiving VXLAN traffic, a VXLAN overlay gateway allows communication between the VXLAN-aware world and the non-VXLAN-aware world. In the non-VXLAN-aware world, a broadcast domain represented by a VLAN typically comprises the virtual cluster switch (VCS) and other switches and devices behind the VCS.

In the initial phase, the VXLAN-aware world consists of virtual networks that are managed by a thirdparty system known as the VMware NSX Controller. The NSX Controller is a highly available distributed system that manages, or orchestrates, all network components and connections in a virtual network.

The VXLAN overlay gateway must communicate with the NSX Controller to create tunnels with VXLAN-aware end devices. The NSX Controller function can comprise a cluster of controllers. The orchestrator function resides most commonly at top of rack (ToR); however, it can also be deployed as an aggregator.

NOTE
VXLAN overlay gateways must be in logical chassis cluster mode. This allows the VCS to present itself as a single device to the NSX Controller, in conjunction with a VMware Hypervisor.

Configuring a VXLAN overlay gateway for NSX Controller deployments

Be sure you do the configuration steps in "Configuring VRRP-E for NSX Controller deployments", below, before you configure the VXLAN overlay gateway. The prerequisite steps demonstrate how to configure a RBridge as part of a Virtual Router Redundancy Protocol - Extended (VRRP-E) group, which is a requirement for a VXLAN overlay gateway. Also, you must configure the identical virtual Ethernet (VE) and VRRP-E group on all the RBridges for the VXLAN overlay gateway.

High-level communication in a VXLAN environment

Figure 4 provides a basic view of the interaction of components in a VXLAN environment.

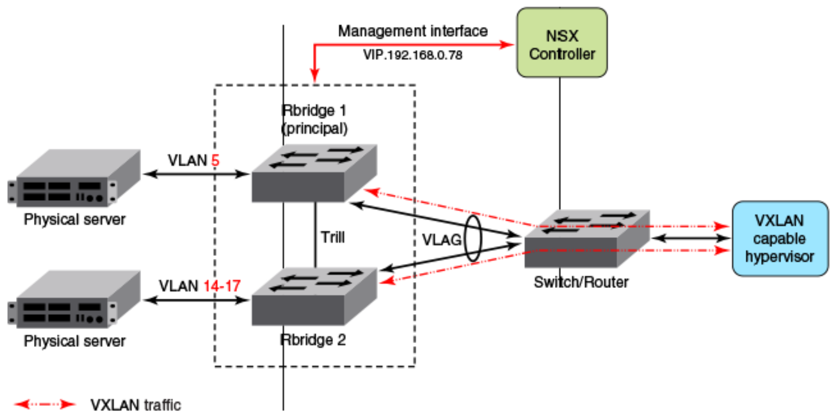


FIGURE 4 **High-level communication for VXLAN overlay gateway**

VXLAN overlay gateways must be part of a two-node virtual switching cluster. In the example shown in Figure 4, RBridge 1 and RBridge 2 make up the two-node cluster. These two RBridges combine to form the VXLAN overlay gateway.

The current principal switch of the VXLAN overlay gateway always communicates with the NSX controller. This communication occurs over what is known as the management interface (depicted by the red line in Figure 4).

NOTE
VXLAN overlay gateways must be in logical chassis cluster mode. This allows the VCS to present itself as a single device to the NSX Controller.

Configuring VRRP-E for NSX Controller deployments

The steps that follow show an example VRRP-Extended (VRRP-E) group configuration for the RBridges shown in Figure 4 on page 304. Perform all the following steps on the principal switch (Rbridge 1). Similarly you can configure on Rbridge 2.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="13">
   <edit-config>
      <target>
         <running></running>
      </target>
      <config>
```

```
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>2</rbridge-id>
                <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <hide-vrrp-holder xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                        <vrrp-extended></vrrp-extended>
                    </hide-vrrp-holder>
                </protocol>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>10</name>
                        <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <ip-config>
                                <address>
                                    <address>101.1.1.1/24</address>
                                </address>
                            </ip-config>
                        </ip>
                        <vrrpe xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                            <vrid>100</vrid>
                            <virtual-ip>
                                <virtual-ipaddr>10.10.10.230</virtual-ipaddr>
                            </virtual-ip>
                            <virtual-mac>
                                <vmac></vmac>
                            </virtual-mac>
                            <short-path-forwarding>
                                <basic></basic>
                            </short-path-forwarding>
                        </vrrpe>
                    </ve>
                </interface>
            <rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="13" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the VXLAN overlay gateway for NSX Controller deployments

Prerequisite steps:

- Before you configure the VXLAN overlay gateway, you need to be sure that the RBridges are configured as part of a virtual-router-extended group.
- Ensure that you configure the identical VE and VRRP-E group on all the RBridges for the VXLAN overlay gateway.
- Create a SPAN <session> element named **1**. This SPAN session must be pre-configured. Refer to Chapter 42, “Configuring Switched Port Analyzer”.

The steps that follow show example VRRP-Extended group configuration for the RBridges shown in Figure 4.

The following steps illustrate how to configure a VXLAN overlay gateway and point it to the NSX controller.

This procedure uses data shown in Figure 4.

1. Establish a NETCONF session with the principal switch (Rbridge 1 in Figure 4).
2. Access the <overlay-gateway> container within the brocade-tunnels RPC to create the name for the VXLAN overlay gateway.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>name1</name>
            </overlay-gateway>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Within the <overlay-gateway> container, include the <name> element and <attach> node to attach existing RBridge IDs to this VXLAN overlay gateway instance. Set the <rb-add> element to **1-2**. This adds RBridge IDs 1 and 2.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>name1</name>
                <attach>
                    <rbridge-id>
                        <rb-add>1,2</rb-add>
                    </rbridge-id>
                </attach>
            </overlay-gateway>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

4. The <vlan> nodes need to be included in the <attach> node of the <overlay-gateway> container. Since multiple VLANs cannot be specified at one time using NETCONF. Multiple such requests needs to be sent for attaching multiple VLANs. For this example, you must repeat this step for VLANs 5, 14, 15, 16, and 17.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
```

```
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>name1</name>
                <attach>
                    <vlan>
                        <vid>5</vid>
                        <mac>0000.0000.0000</mac>
                    </vlan>
                </attach>
            </overlay-gateway>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

All the MAC addresses that the VXLAN overlay gateway learns on these VLANs are shared with the NSX controller. When a MAC address ages out in VCS, the MAC address is removed from the NSX.

5. Within the <overlay-gateway> container, include the <ip>, <interface>, and set the <ve> node name and set the name for the <vrrp-extended-group> element to 100.

   The VRRPE virtual IP configured for this VE and VRRP-E group are used as VXLAN overlay gateway IP address.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="4">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>name1</name>
                <ip>
                    <interface>
                        <ve>
                            <ve-id>10</ve-id>
                            <vrrp-extended-group>100</vrrp-extended-group>
                        </ve>
                    </interface>
                </ip>
            </overlay-gateway>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="4" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>

<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
```

```
                <name>gateway2</name>
                <ip>
                    <interface>
                        <loopback>
                            <loopback-id>121</loopback-id>
                        </loopback>
                    </interface>
                </ip>
            </overlay-gateway>
        </config>
    <edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

6. (optional) Within the <overlay-gateway> container, include the <enable> and <statistics> node so that you can set the <stats-direction> element value to **both** and the <vlan-action> element to **add**. Refer to the Network OS YANG Reference for additional options on these options.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="6">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>name1</name>
                <enable>
                    <statistics>
                        <stats-direction>both</stats-direction>
                        <vlan-action>add</vlan-action>
                        <vlan-list>5,15-17</vlan-list>
                    </statistics>
                </enable>
            </overlay-gateway>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="6" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

7. (optional) Within the <overlay-gateway> container, include the <monitor> node. Set the element values for the <monitor> node as listed below:

   - Set the <session> element to **1**. This SPAN session must be pre-configured. Refer to Chapter 42, "Configuring Switched Port Analyzer".
   - Set the <direction> element to both.Set the <remote-endpoint> element to **any**.
   - Set the <vlan-add-remove> element to **41-43**.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="7">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
```

```
                <name>name1</name>
                <monitor>
                    <session>1</session>
                    <direction>both</direction>
                    <remote-endpoint>any</remote-endpoint>
                    <vlan-add-remove>add</vlan-add-remove>
                    <vlan-range>5,14-17</vlan-range>
                </monitor>
            </overlay-gateway>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="7" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

8. Within the <overlay-gateway> container, include the <activate/> element. The presence of this element activates the VXLAN overlay gateway.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="8">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>name1</name>
                <activate></activate>
            </overlay-gateway>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="8" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

9. The following example shows how to configure the access-list on the overlay-gateway in the ingress direction, so that the access-list is applied on the traffic ingressing on the tunnels which terminate on the overlay-gateway.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>gateway2</name>
                <gw-type>layer2-extension</gw-type>
                <access-lists>
                    <mac>
                        <in>
                            <mac-acl-in-name>stdmacaclin</mac-acl-in-name>
                            <mac-acl-in-dir></mac-acl-in-dir>
                        </in>
                    </mac>
                    <ipv4>
                        <in>
```

```
                        <ipv4-acl-in-name>stdipaclin</ipv4-acl-in-name>
                        <ipv4-acl-in-dir></ipv4-acl-in-dir>
                    </in>
                </ipv4>
                <ipv6>
                    <in>
                        <ipv6-acl-in-name>stdipv6aclin</ipv6-acl-in-name>
                        <ipv6-acl-in-dir></ipv6-acl-in-dir>
                    </in>
                </ipv6>
            </access-lists>
        </overlay-gateway>
    </config>
<edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring the NSX Controller

Prerequisite steps

Before you configure the NSX controller, complete the task in "Configuring the VXLAN overlay gateway for NSX Controller deployments" on page 279. The purpose of this task is to generate the security certificate for the VXLAN overlay gateway. This procedure uses data shown in Figure 4.

1. Issue the <edit-config> RPC to edit the running configuration.
2. Access the <nsx-controller> container within the brocade-tunnels RPC with the name **controller1**.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="9">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <nsx-controller xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>controller1</name>
            </nsx-controller>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="9" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Within the <nsx-controller> container and the <name> node, set the element values for the <connection-addr> container as listed below:
   - Set the <address> element to the IP address for one of the NSX controllers in the control cluster, in this case **10.30.5.74**.
   - Set the <method> element to **ssl**.
   - Set the <port> element to **6632**.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="11">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <nsx-controller xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>controller1</name>
                <connection-addr>
                    <address>10.30.5.74</address>
                    <port>6632</port>
                    <method>ssl</method>
                </connection-addr>
            </nsx-controller>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="11" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

4. (optional) Within the <nsx-controller> container, include the <reconnect-interval> element to change the reconnect interval between the NSX controller and the VCS fabric in case the connection is lost. The default is 10 seconds, meaning that a reconnection is attempted every 10 seconds.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="12">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <nsx-controller xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>controller1</name>
                <reconnect-interval>7</reconnect-interval>
            </nsx-controller>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="12" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

5. Within the <nsx-controller> container, include the <activate/> element. The presence of this element activates the NSX controller profile.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="13">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <nsx-controller xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>controller1</name>
                <activate></activate>
            </nsx-controller>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="13" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Displaying VXLAN information

Use the <get-config> RPC to retrieve the current configuration data and operational state data from the brocade-tunnels.YANG RPC module. Refer to "Retrieving configuration data" on page 11 for detailed instructions.

Chapter

Configuring Distributed VXLAN Gateways 29

In this chapter

Distributed VXLAN gateways overview

The distributed VXLAN gateways feature eliminates the need for an external gateway device.

Prior to Network OS 6.0.1, VXLAN gateways had to be connected to the VCS Fabric (for example, a four-node gateway fabric) as an external device in order to bridge the overlay network and the physical networks that are interconnected by the VCS Fabric. Beginning with the current release, the gateway function can be hosted by the VCS RBridges. This eliminates the need for an external gateway device and optimizes network resources, improving network performance in the data center.

Prior to the current Network OS release, the VXLAN tunnel endpoint (VTEP) address was associated with a VRRP-E virtual Ethernet (VE) interface or a loopback interface. Now the anycast gateway's IP address can also be used for the VTEP IP. The Layer 3 anycast-gateway protocol allows multiple RBridges in the VCS Fabric to form a group of gateway routers and share the same gateway IP address for a given subnet. This gateway IP address is similar to a virtual IP address in VRRP-E. The VXLAN gateway can use an anycast IP address as its VTEP IP address to obtain the same active-active forwarding behavior that is available in VRRP-E.

NOTE
Existing VRRP-E implementations cannot support more than four RBridges in a session. As a result, VRRP-E-based VXLAN gateways are also limited to a maximum of four RBridges.

When the gateway function is enabled on an RBridge, the overlay network classification competes with the Virtual Fabrics configurations in consuming available hardware resources. As a result, a platform-dependent VLAN classification profile is introduced on the Brocade VDX 6940 to enable the user to allocate resources between gateway and Virtual Fabrics features. The user can indicate whether a gateway RBridge is to be deployed at top of rack (ToR) or at the aggregation layer, and whether the Virtual Fabrics or overlay network or both are to be supported, allowing resources to each feature. At ToR, more resources are allocated to Virtual Fabrics edge classification than to the gateway function. Conversely, at the aggregation layer, more resources are allocated to gateway functions than to Virtual Fabrics classification. A vlan-classification keyword in the hardware-profile command allows the user to allocate resources for on a variety of network topologies.

Configuring distributed VXLAN gateways

NOTE
This task uses the Fabric-Virtual-Gateway IP address for the Virtual Ethernet (VE) interface as the VTEP IP address.

ATTENTION
The RBridge membership for the anycast gateway and the overlay gateway VTEP must be the same or this configuration will not work.

---

1. Create and configure a VE interface.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <ve>
                    <ve-id>10</ve-id>
                    <attach>
                        <rbridge-id>
                            <rb-add>1,2</rb-add>
                        </rbridge-id>
                    </attach>
                    <ip xmlns="urn:brocade.com:mgmt:brocade-interface">
                        <fabric-virtual-gateway
xmlns="urn:brocade.com:mgmt:brocade-anycast-gateway">
                            <ip-gw-id>1</ip-gw-id>
                            <gateway-address>1.1.1.1/24</gateway-address>
                            <gratuitous-arp>
                                <timer>40</timer>
                            </gratuitous-arp>
                            <hold-time>25</hold-time>
                            <load-balancing-disable>true</load-balancing-disable>
                            <enable>true</enable>
                        </fabric-virtual-gateway>
                    </ip>
                </ve>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. Create and configure a VXLAN overlay gateway.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
```

```
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>name1</name>
                <attach>
                    <rbridge-id>
                        <rb-add>1,2</rb-add>
                    </rbridge-id>
                    <vlan>
                        <vid>5</vid>
                        <mac>0000.0000.0000</mac>
                    </vlan>
                </attach>
                <ip>
                    <interface>
                        <ve>
                            <ve-id>10</ve-id>
                            <fabric-virtual-gateway
xmlns="urn:brocade.com:mgmt:brocade-anycast-gateway"></fabric-virtual-gateway>
                        </ve>
                    </interface>
                </ip>
                <gw-type>layer2-extension</gw-type>
                <activate></activate>
            </overlay-gateway>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring VLAN classification profiles for distributed VXLAN gateways

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <hardware-profile xmlns="urn:brocade.com:mgmt:brocade-hardware">
                    <vlan-classification>
                        <predefined>
                            <vlan_profiletype>aggregator-basic</vlan_profiletype>
                        </predefined>
                    </vlan-classification>
                </hardware-profile>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <ok/>
</rpc-reply>
```

Chapter

Configuring Virtual Fabrics

30

In this chapter

Virtual Fabric configuration with NETCONF overview

This chapter provides procedures for configuring a Virtual Fabric using the NETCONF interface. Refer to the *Network OS Layer 2 Switching Configuration Guide* for the following related information:

- An overview of Virtual Fabrics
- An explanation of how Virtual Fabrics work
- Virtual Fabric configuration guidelines and restrictions
- Virtual Fabric operations
- Virtual Fabric instance configurations
- Enabling Virtual Fabrics
- Configuring Layer 3 Virtual Fabric features
- Troubleshooting configuration failures
- Upgrading and downgrading firmware

Through the NETCONF interface, you can perform the following operations on Virtual Fabrics:

- Use the <edit-config> RPC to configure Virtual Fabrics.
- Use the <get-config> RPC to validate configuration settings.

Virtual Fabric parameters are defined in the brocade-interface and brocade-mac-address-table YANG modules. For structural maps of these YANG modules, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all VCS Fabric management parameters, refer to the brocade-interface.yang and brocade-vlan.yang files.

Configuring a Virtual Fabric instance

Configuring a Virtual Fabric instance consists of enabling Virtual Fabric configuration in the fabric, and then configuring a Virtual Fabric VLAN ID instance that is equal to or greater than 4096. Virtual Fabric supports up through 8192 VLANs, with 8191 being the largest VLAN ID that can be assigned.

Enabling Virtual Fabric configuration

Virtual Fabric is enabled by activating the <vfab-enable> node from the brocade-vcs.yang module.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <vcs xmlns="urn:brocade.com:mgmt:brocade-vcs">
        <virtual-fabric><vfab-enable></vfab-enable></virtual-fabric>
      </vcs>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>

</rpc-reply>
```

Creating a Virtual Fabric instance

Under the <vlan> node, specify the <name> element containing the new VLAN ID, where vlan_id <name> element is a number equal to or greater than 4096 through 8191, and is not a reserved VLAN.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
            <target>
                <running/>
            </target>
        <config>
        <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
            <interface>
                <vlan>
                    <name>5000</name>
                </vlan>
            </interface>
        </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring additional Layer 2 Virtual Fabric features

This section addresses additional features that are available on trunk ports once a Virtual Fabric is established.

Configuring Virtual Fabrics and defining and associating PVLANs

The private VLANs (PVLANs) for a Virtual Fabric can be configured in the following three types:

- Primary VLAN
- Isolated VLAN
- Community VLAN

At least two of these three types of VLANS must be configured to create a Virtual Fabric.

1. Under the <vlan> node, specify the <name> element containing the new VLAN ID to create VLAN instances that are equal to or greater than 4096, through 8191. Repeat this command for three Virtual Fabrics: 5000, 6000, and 7000.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1110" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
            <target>
                <running/>
            </target>
        <config>
        <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
            <interface>
                <vlan>
                    <name>5000</name>
                </vlan>
            </interface>
        </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1110" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>

<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1111" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>6000</name>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1111" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>

<?xml version="1.0" encoding="UTF-8"?>
```

```
<rpc message-id="1112" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>7000</name>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1112" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. Use the <pvlan-type-leaf> node to create the three types of PVLAN: primary, isolated, and community.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>5000</name>
                        <private-vlan>
                            <pvlan-type-leaf>primary</pvlan-type-leaf>
                        </private-vlan>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>

<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="55"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
```

```
                    <vlan>
                        <name>6000</name>
                        <private-vlan>
                            <pvlan-type-leaf>isolated</pvlan-type-leaf>
                        </private-vlan>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="55" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>

<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="55"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>7000</name>
                        <private-vlan>
                            <pvlan-type-leaf>isolated</pvlan-type-leaf>
                        </private-vlan>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="55" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Using the <private-vlan>/<association> node, associate the secondary PVLANs (isolated and community) with the primary PVLAN using the <sec-assoc-add> node.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1104" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
       <edit-config>
              <target>
                  <running/>
              </target>
           <config>
           <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
               <interface>
                   <vlan>
                       <name>5000</name>
                           <private-vlan>
                               <association>
                                   <sec-assoc-add>6000</sec-assoc-add>
                               </association>
```

```
                    </private-vlan>
              </vlan>
       </interface>
          </interface-vlan>
          </config>
       </edit-config>
</rpc>

<rpc-reply message-id="1104" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>

<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1105" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
       <edit-config>
              <target>
                 <running/>
              </target>
          <config>
          <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
              <interface>
                 <vlan>
                     <name>5000</name>
                        <private-vlan>
                           <association>
                               <sec-assoc-add>7000</sec-assoc-add>
                           </association>
                        </private-vlan>
              </vlan>
              </interface>
          </interface-vlan>
          </config>
       </edit-config>
</rpc>

<rpc-reply message-id="1105" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring physical interfaces

The physical interfaces can be configured in the following five types:

- Switchport-private-vlan-host mapped to isolated or community VLANs associated on the access port.
- Switchport-private-vlan-trunk-host mapped to isolated or community VLANs associated on the trunk port.
- Switchport-private-vlan-promiscuous mapped to primary VLANs associated on the access port.
- Switchport-private-vlan-trunk-promiscuous mapped to primary VLANs associated on the trunk port.
- Switchport-private-vlan-trunk mapped to primary, isolated, community, or standard VLANs associated on the trunk port.

The following task is one possible example of configuring the physical interface. For additional examples, refer to "Configuring a trunk transport Virtual Fabric on an interface" on page 305, "Configuring native Virtual Fabric on interfaces" on page 300, and "Configuring access Virtual Fabric on interfaces" on page 302.

1. Create classification rules for the primary and isolated or community VLANs at the respective primary and host ports.

    a. Use the <private-vlan-trunk> and <trunk-promiscuous/> nodes to configure interface tengigabitethernet 11/0/1 as the primary promiscuous trunk port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <tengigabitethernet>
                        <name>11/0/1</name>
                        <switchport-basic>
                        <basic/>
                        </switchport-basic>
                        <switchport>
                            <mode>
                                <private-vlan>
                                    <private-vlan-trunk>
                                        <trunk-promiscuous/>
                                    </private-vlan-trunk>
                                </private-vlan>
                            </mode>
                        </switchport>
                    </tengigabitethernet>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

    b. Use the <private-vlan-trunk> and <trunk-host/> nodes to configure interface tengigabitethernet 11/0/2 as the isolated trunk port.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="5"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <tengigabitethernet>
                        <name>11/0/2</name>
```

```
                        <switchport-basic>
                               <basic/>
                        </switchport-basic>
                        <switchport>
                            <mode>
                                <private-vlan>
                                    <private-vlan-trunk>
                                        <trunk-host/>
                                    </private-vlan-trunk>
                                </private-vlan>
                            </mode>
                        </switchport>
                    </tengigabitethernet>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="5" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

c. Use the <private-vlan-trunk> and <trunk-host/> nodes to configure interface tengigabitethernet 11/0/3 as the community trunk port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="7"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
    <running/>
    </target>
    <config>
      <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
           <tengigabitethernet>
               <name>11/0/3</name>
                   <switchport-basic>
                        <basic/>
                   </switchport-basic>
                   <switchport>
                        <mode>
                            <private-vlan>
                                <private-vlan-trunk>
                                    <trunk-host/>
                                </private-vlan-trunk>
                            </private-vlan>
                        </mode>
                   </switchport>
           </tengigabitethernet>
           </interface>
       </interface-vlan>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="7" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

2. Use the <vfabric-trunk-vlan-id> and <vfabric-trunk-ctag-range> nodes to configure the PVLAN association on the promiscuous trunk port with C-TAG 10. You must also add 6000 and 7000 to the mapping using the <oper> node. The <promis-pri-pvlan> node sets 5000 to be the promiscuous trunk port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<edit-config>
 <target>
  <running/>
 </target>
 <config>
 <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
  <tengigabitethernet>
   <name>1/0/1</name>
   <switchport>
    <trunk>
     <trunk-vlan-classification>
      <allowed>
       <vlan>
        <add>
         <trunk-vlan-id>5000</trunk-vlan-id>
         <trunk-ctag-range>10</trunk-ctag-range>
        </add>
       </vlan>
      </allowed>
     </trunk-vlan-classification>
    </trunk>
    <private-vlan>
     <mapping>
      <promis-pri-pvlan>5000</promis-pri-pvlan>
      <oper>add</oper>
      <promis-sec-pvlan-range>6000,7000</promis-sec-pvlan-range>
     </mapping>
    </private-vlan>
   </switchport>
  </tengigabitethernet>
 </interface>
 </config>
</edit-config>
</rpc>

<rpc-reply message-id="2" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

Configuring a trunk Virtual Fabric with a single C-TAG.

When configuring a trunk Virtual Fabric with a single C-TAG on the interface, it is important to remember that the C-TAG is applicable only for a trunk VLAN.

The following example configures 10-Gigabit Ethernet port 1/0/1 as a trunk VLAN with a C-TAG range.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="2"xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
```

```
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
                    <name>1/0/1</name>
                    <switchport-basic> <basic/></switchport-basic>
                    <switchport>
                    <mode>
                        <vlan-mode>trunk</vlan-mode>
                    </mode>
                    <trunk>
                        <trunk-vlan-classification>
                            <allowed>
                                <vlan>
                                    <add>
                                        <trunk-vlan-id>5001</trunk-vlan-id>
                                        <trunk-ctag-range>10</trunk-ctag-range>
                                    </add>
                                </vlan>
                            </allowed>
                        </trunk-vlan-classification>
                    </trunk>
                    </switchport>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring native Virtual Fabric on interfaces

To configure the native Virtual Fabric classifications requires two RPCs. The first RPC configures the port as a Layer 2 interface; the second RPC configures the native VLAN classification.

The following example configures 10-Gigabit Ethernet port 1/0/1 as a native VLAN.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
                    <name>1/0/1</name>
                    <switchport>
                        <trunk>
                            <native-vlan-classification>
                                <native-vlan-id>300</native-vlan-id>
                            </native-vlan-classification>
                        </trunk>
```

```
                    </switchport>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. The following NETCONF xml request configures the native-vlan-xtagged on an Layer 2 interface in trunk-no-default-native mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
                    <name>1/0/1</name>
                    <switchport>
                        <trunk>
                            <native-vlan-xtagged-config>

<native-vlan-id-xtagged>5000</native-vlan-id-xtagged>

<native-vlan-ctag-id-xtagged>50</native-vlan-ctag-id-xtagged>

<native-vlan-egress-type-xtagged>tagged</native-vlan-egress-type-xtagged>
                            </native-vlan-xtagged-config>
                        </trunk>
                    </switchport>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

4. The following NETCONF xml request configured the native-vlan-untagged on an L2 interface in trunk-no-default-native mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
```

```
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
            <tengigabitethernet
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
                <name>1/0/1</name>
                <switchport>
                    <trunk>
                        <native-vlan-untagged-config>

<native-vlan-id-untagged>5001</native-vlan-id-untagged>
                        </native-vlan-untagged-config>
                    </trunk>
                </switchport>
            </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring access Virtual Fabric on interfaces

To configure the interface as an access Virtual Fabric interface requires two RPCs. The first RPC configures the port as a Layer 2 interface; the second RPC configures the interface port as an access Virtual Fabric with a MAC address.

The following example configures 10-Gigabit Ethernet port 1/0/1 as an access VLAN.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
                    <name>1/0/1</name>
                    <switchport-basic>
                        <basic/>
                    </switchport-basic>
                    <switchport>
                        <access-mac-vlan-classification>
                            <access>
                                <vlan>
                                    <access-vlan-id>5002</access-vlan-id>

<access-mac-address>000a.000b.0002</access-mac-address>
                                </vlan>
                            </access>
                        </access-mac-vlan-classification>
                    </switchport>
                </tengigabitethernet>
            </interface>
        </config>
```

```
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring an access Virtual Fabric with a MAC group

To configure the interface as an access Virtual Fabric interface requires two RPCs. The first RPC configures the port as a Layer 2 interface; the second RPC configures the interface port as an access Virtual Fabric with a MAC group.

The following example configures 10-Gigabit Ethernet port 1/0/1 as an access VLAN with a MAC group.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
                    <name>1/0/1</name>
                    <switchport-basic>
                        <basic/>
                    </switchport-basic>
                    <switchport>
                        <access-mac-group-vlan-classification>
                            <access>
                                <vlan>
                                    <access-vlan-id>5001</access-vlan-id>
                                    <access-mac-group>1</access-mac-group>
                                </vlan>
                            </access>
                        </access-mac-group-vlan-classification>
                    </switchport>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Creating a MAC group instance and assigning MAC addresses

Create a MAC group instance to define the MAC addresses of end stations by using the <mac-group> node. The value of <mac-group-id> ranges from 1 through 500. Use the <entry_address> node to add one or more addresses in hexadecimal notation.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac-group xmlns="urn:brocade.com:mgmt:brocade-mac-address-table">
                 <mac-group-id>1</mac-group-id>
                     <mac-group-entry>
                         <entry-address>000a.0001.0001</entry-address>
                     </mac-group-entry>
                     <mac-group-entry>
                         <entry-address>000a.0001.0002</entry-address>
                     </mac-group-entry>
                </mac-group>
            <mac-group xmlns="urn:brocade.com:mgmt:brocade-mac-address-table">
                 <mac-group-id>2</mac-group-id>
                     <mac-group-entry>
                         <entry-address>000a.0002.0001</entry-address>
                     </mac-group-entry>
                     <mac-group-entry>
                         <entry-address>000a.0002.0002</entry-address>
                     </mac-group-entry>
                </mac-group>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting a MAC group

Use the standard NETCONF process to delete a MAC group using the <mac-group-id> node.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac-group xmlns="urn:brocade.com:mgmt:brocade-mac-address-table">
                 <mac-group-id xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
            </mac-group>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Transport Virtual Fabric

This section describes tasks for configuring the transport service for Virtual Fabrics.

Configuring transport service ID on a VLAN

NOTE
Spanning tree must be shutdown on this VLAN before executing this procedure. Refer to "Disabling STP on a VLAN" on page 252.

On Brocade VDX hardware, VLANs are treated as interfaces from a configuration viewpoint.

By default, all the DCB ports are assigned to VLAN 1 (VLAN ID equals 1). The VLAN ID can be 1 through 8192, but VLAN IDs 3584 through 4094 are internally-reserved VLAN IDs. VLAN 8191 is the largest VLAN ID that can be assigned.

To create a VLAN interface.

The following example creates VLAN 6011 with transport service 21. Multiple VLANs can be created in one NETCONF call.

```
<?xml version="1.0" encoding="UTF-8"?> - not throeing error
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>6011</name>
                        <transport-service>21</transport-service>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring a trunk transport Virtual Fabric on an interface

When configuring a trunk transport virtual fabric on the interface, it is important to remember that the range of C-TAGs is applicable only for a trunk transport Virtual Fabric.

To configure the trunk Virtual Fabric requires two RPCs. The first RPC configures the port as a Layer 2 interface; the second RPC configures the Virtual Fabric with a trunk classification.

The following example configures 10-Gigabit Ethernet port 1/0/1 as a trunk VLAN with a C-TAG range.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
                    <name>1/0/1</name>
                    <switchport-basic>
                        <basic/>
                    </switchport-basic>
                    <switchport>
                        <mode>
                            <vlan-mode>trunk</vlan-mode>
                        </mode>
                        <trunk>
                            <trunk-vlan-classification>
                                <allowed>
                                    <vlan>
                                        <add>
                                            <trunk-vlan-id>6011</trunk-vlan-id>
                                            <trunk-ctag-range>51-68</trunk-ctag-range>
                                        </add>
                                    </vlan>
                                </allowed>
                            </trunk-vlan-classification>
                        </trunk>
                    </switchport>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring a VXLAN overlay gateway in a Virtual Fabrics context

In the current release, the VLAN-based broadcast domain is extended over a Layer 3 network without the need for an orchestrator. The extension function commonly resides at the aggregation layer, either embodied in the aggregation node itself, or hanging as a service off the aggregation node in a onearm topology. This feature resides in the network spine, with up to four RBridges participating. Tunnels are provided through router ports and switch ports.

In addition, the user can collect statistics on tunnel and gateway traffic, through the range of attached VLANs. Both transmit and receive traffic can be monitored for a single tunnel and range of VLANs, or all tunnels and a range of VLANs. Support is also provided for similar sFlow monitoring.

For more information refer to *Network OS Layer 2 Switching Configuration Guide*.

Configuring VRRP-E

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="4">
  <edit-config>
    <target>
      <running></running>
    </target>
    <config>
      <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
        <name>name1</name>
        <ip>
          <interface>
            <ve>
              <ve-id>10</ve-id>
              <vrrp-extended-group>100</vrrp-extended-group>
            </ve>
          </interface>
        </ip>
      </overlay-gateway>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="4" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
       <ok/>
</rpc-reply>
```

Creating a Layer 2 extension overlay gateway

The following example shows how to configure a layer2 extension overlay-gateway, to extend the Layer 2 domains (VLANs) from one VCS (Site) to another VCS (Site).

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>gateway2</name>
                <gw-type>layer2-extension</gw-type>
            </overlay-gateway>
        </config>
    <edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Attaching existing RBridge IDs to the VXLAN overlay gateway instance

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2">
  <edit-config>
    <target>
      <running></running>
    </target>
    <config>
      <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
        <name>name1</name>
        <gw-type>layer2-extension</gw-type>
        <attach>
          <rbridge-id>
            <rb-add>1,2</rb-add>
          </rbridge-id>
        </attach>
      </overlay-gateway>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring IP address of the overlay-gateway

The following example shows how to configure the IP address of the overlay-gateway through loopback interface.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>gateway2</name>
                <gw-type>layer2-extension</gw-type>
                <ip>
                    <interface>
                        <loopback>
                            <loopback-id>121</loopback-id>
                        </loopback>
                    </interface>
                </ip>
            </overlay-gateway>
        </config>
    <edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring auto VLAN to VNI mapping

The following example shows how to configure auto VLAN to VNI mapping for the Layer 3 extension overlay-gateway, so that the system automatically maps the VLAN to VNI.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>gateway2</name>
                <gw-type>layer2-extension</gw-type>
                <map>
                    <vlan>
                        <vni>
                            <auto/>
                        </vni>
                    </vlan>
                </map>
            </overlay-gateway>
        </config>
    <edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring explicit VLAN to VNI mapping

The following example shows how to configure explicit VLAN to VNI mapping. The explicit and auto VLAN to VNI mapping configurations are mutually exclusive.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>gateway2</name>
                <gw-type>layer2-extension</gw-type>
                <map>
                    <vlan-vni-mapping>
                        <vid>1</vid>
                        <vni>100</vni>
                    </vlan-vni-mapping>
                    <vlan-vni-mapping>
                        <vid>2</vid>
                        <vni>120</vni>
                    </vlan-vni-mapping>
                    <vlan-vni-mapping>
                        <vid>3</vid>
                        <vni>130</vni>
                    </vlan-vni-mapping>
```

```
                </map>
            </overlay-gateway>
        </config>
    <edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring ingress access-list on the overlay-gateway

The following example shows how to configure the access-list on the overlay-gateway in the ingress direction, so that the access-list is applied on the traffic ingressing on the tunnels which terminate on the overlay-gateway.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>gateway2</name>
                <gw-type>layer2-extension</gw-type>
                <access-lists>
                    <mac>
                        <in>
                            <mac-acl-in-name>stdmacaclin</mac-acl-in-name>
                            <mac-acl-in-dir></mac-acl-in-dir>
                        </in>
                    </mac>
                    <ipv4>
                        <in>
                            <ipv4-acl-in-name>stdipaclin</ipv4-acl-in-name>
                            <ipv4-acl-in-dir></ipv4-acl-in-dir>
                        </in>
                    </ipv4>
                    <ipv6>
                        <in>
                            <ipv6-acl-in-name>stdipv6aclin</ipv6-acl-in-name>
                            <ipv6-acl-in-dir></ipv6-acl-in-dir>
                        </in>
                    </ipv6>
                </access-lists>
            </overlay-gateway>
        </config>
    <edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring a remote site

The following examples shows how to configure a remote site, to which the Layer 2 domain needs to be extended to.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>gateway2</name>
                <gw-type>layer2-extension</gw-type>
                <site>
                    <name>sanjose</name>
                    <tunnel-dst>
                        <address>10.10.10.1</address>
                    </tunnel-dst>
                </site>
            </overlay-gateway>
        </config>
    <edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring extend VLANs for the remote site

The following example shows how to configure the Layer 2 domains (VLANs) which need to be extended to the remote site.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>gateway2</name>
                <gw-type>layer2-extension</gw-type>
                <site>
                    <name>sanjose</name>
                    <tunnel-dst>
                        <address>10.10.10.1</address>
                    </tunnel-dst>
                    <extend>
                        <vlan>
                            <add>1-10</add>
                        </vlan>
                    </extend>
                </site>
            </overlay-gateway>
        </config>
    <edit-config>
```

```
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Administratively shutting down a remote site

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>gateway2</name>
                <gw-type>layer2-extension</gw-type>
                <site>
                    <name>sanjose</name>
                    <shutdown/>
                </site>
            </overlay-gateway>
        </config>
    <edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Chapter

Configuring Spanning Tree Protocols 31

In this chapter

Spanning tree configuration with NETCONF overview

This chapter provides procedures for configuring STP, RSTP, MSTP, PVST, and Rapid PVST using the NETCONF interface. Refer to the *Network OS Layer 2 Switching Configuration Guide* for the following related information:

- Overviews of STP, RSTP, MSTP, PVST, and Rapid PVST
- Configuration guidelines and restrictions
- A summary of spanning tree configuration default values
- How to perform spanning tree operations using the Network OS command line interface
- Spanning Tree Protocol and VCS mode
- Spanning Tree Protocol and DiST

Through the NETCONF interface, you can perform the following operations on STP, RSTP, MSTP, PVST, and Rapid PVST:

- Use the <edit-config> RPC to configure spanning tree globally and on each interface.
- Use the <get-stp-brief-info> custom RPC to obtain operational information about the spanning tree protocols.
- Use the <get-config> RPC to validate configuration settings.
- STP, RSTP, MSTP, PVST, and Rapid PVST parameters are defined in the brocade-xstp YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Configuring STP

To configure STP, perform the following steps.

1. Issue the <edit-config> RPC to configure the <protocol> node in the urn:brocade.com:mgmt:brocade-interface namespace.
2. Under the <protocol> node, include the <spanning-tree> node from the urn:brocade.com:mgmt:brocade-xstp namespace.
3. Under the <spanning-tree> node, include the <stp> node element to configure global STP parameters.

   Refer to "Enabling STP, RSTP, MSTP, PVST, or Rapid PVST" on page 321 for details.
4. Under the STP node, designate the root switch using the <bridge-priority> leaf node.

   For details, refer to "Specifying the bridge priority for all xSTP" on page 322. The range is 0 through 61440 and the priority values can be set only in increments of 4096. The default value is 32768.

```
<protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
    <spanning-tree>
        <stp>
            <bridge-priority>28672</bridge-priority>
        </stp>
    </spanning-tree>
</protocol>
```

5. *Optional*: Enable the port fast feature on switch ports that connect directly to workstations or PCs.

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
    <tengigabitethernet>
        <name>22/0/10</name>
        <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
             operation="delete"/>
        <spanning-tree>
            <portfast>
                <portfastbasic/>
            </portfast>
        </spanning-tree>
    </tengigabitethernet>
</interface>
```

   For details, refer to "Enabling port fast (STP and PVST)" on page 341.

6. Repeat step 5 for every port connected to a workstation or PC.

NOTE
Do not enable port fast on ports that connect to other switches.

Enabling port fast on ports can cause temporary bridging loops, in both trunking and non-trunking mode.

7. To influence selection of the root port, set the port priority.

   The range is 0 through 240 in increments of 16. The default is 128. A lower number designates a higher priority.

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
```

```
    <tengigabitethernet>
        <name>22/0/13</name>
        <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
             operation="delete"/>
        <spanning-tree>
            <priority>32</priority>
        </spanning-tree>
    </tengigabitethernet>
</interface>
```

For details, refer to “Specifying the port priority” on page 342.

8. *Optional:* Enable the guard root feature on a port.

   All other switch ports connected to other switches and bridges are automatically placed in blocking mode.

   Do not apply the guard root to ports connected to workstations or PCs: these ports remain in the forwarding state.

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
    <tengigabitethernet>
        <name>22/0/1</name>
        <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
             operation="delete"/>
        <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
            <guard>
                <root/>
            </guard>
        </spanning-tree>
    </tengigabitethernet>
</interface>
```

   For detailed information, refer to “Enabling the guard root (STP and RSTP)” on page 338.

9. *Optional*: Enable peer-switch feature only on port-channel when interoperating with Cisco and also when peer-switch feature has been enabled on Cisco vPC domain.

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
    <port-channel>
        <name>3</name>
        <spanning-tree>
            <peer-switch/>
        </spanning-tree>
    </port-channel>
</interface>
```

10. Issue the <bna-config-cmd> RPC to save the *running-config* file to the *startup-config* file.

The following example enables STP, designates a root switch, enables port fast on two ports, and establishes the root port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2000" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree>
                    <stp>
                        <bridge-priority>28672</bridge-priority>
```

```
                </stp>
            </spanning-tree>
        </protocol>
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
            <tengigabitethernet>
                <name>22/0/10</name>
                <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                <spanning-tree>
                    <portfast>
                        <portfastbasic/>
                    </portfast>
                </spanning-tree>
            </tengigabitethernet>
            <tengigabitethernet>
                <name>22/0/11</name>
                <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                <spanning-tree>
                    <portfast>
                        <portfastbasic/>
                    </portfast>
                </spanning-tree>
            </tengigabitethernet>
            <tengigabitethernet>
                <name>22/0/13</name>
                <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                <spanning-tree>
                    <priority>32</priority>
                </spanning-tree>
            </tengigabitethernet>
        </interface>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2000" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

When the spanning tree topology is completed, the network switches send and receive data only on the ports that are part of the spanning tree. Data received on ports that are not part of the spanning tree is blocked.

NOTE
Brocade recommends leaving other STP variables at their default values.

For more information on STP, refer to "Spanning tree configuration and management" on page 321.

Configuring RSTP

The basic process for configuring RSTP is as follows.

1. Issue the <edit-config> RPC to configure the <protocol> node in the urn:brocade.com:mgmt:brocade-interface namespace.
2. Under the <protocol> node, include the <spanning-tree> node from the urn:brocade.com:mgmt:brocade-xstp namespace.
3. Under the <spanning-tree> node, include the <rstp> node element.

   Refer to "Enabling STP, RSTP, MSTP, PVST, or Rapid PVST" on page 321 for details.
4. Under the <rstp> node, designate the root switch using the <bridge-priority> leaf node.

   For details, refer to "Specifying the bridge priority for all xSTP" on page 322. The range is 0 through 61440 and the priority values can be set only in increments of 4096. The default value is 32768.

```
<protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
    <spanning-tree>
        <stp>
            <bridge-priority>28672</bridge-priority>
        </stp>
    </spanning-tree>
</protocol>
```

5. Under the <rstp> node, set other RSTP parameters including the bridge forward delay, maximum aging time, error disable timeout period, port channel path cost, and hello time.

```
<protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
    <spanning-tree>
        <rstp>
            <forward-delay>20</forward-delay>
            <max-age>25<max-age>
            <error-disabled-timeout>
                <enable/>
                <interval>60</interval>
            </error-disabled-timeout>
            <port-channel>
                <path-cost>custom</path-cost>
            </port-channel>
            <hello-time>5</hello-time>
        </rstp>
    </spanning-tree>
</protocol>
```

6. *Optional*: Enable the edge port feature on switch ports that connect directly to a workstations or PC.

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
    <tengigabitethernet>
        <name>22/0/10</name>
        <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
             operation="delete"/>
        <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
            <edgeport>
                <edgeportbasic/>
            </edgeport>
        </spanning-tree>
    </tengigabitethernet>
</interface>
```

   For details, refer to "Enabling a port (interface) as an edge port" on page 337.
7. Repeat step 6 for every port connected to workstations or PCs.

NOTE
Do not enable port fast on ports that connect to other switches.

Enabling port fast on ports can cause temporary bridging loops, in both trunking and non-trunking mode.

8. To influence selection of the root port, set the port priority.

   The port priority range is 0 through 240 in increments of 16. The default is 128. A lower number designates a higher priority.

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
    <tengigabitethernet>
        <name>22/0/13</name>
        <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
             operation="delete"/>
        <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
            <priority>32</priority>
        </spanning-tree>
    </tengigabitethernet>
</interface>
```

   For details, refer to "Specifying the port priority" on page 342.

9. *Optional:* Enable the guard root feature on a port.

   All other switch ports connected to other switches and bridges are automatically placed in blocking mode.

   Do not apply the guard root to ports connected to workstations or PCs: these ports remain in the forwarding state.

```
<interface xmlns="urn:brocade.com:mgmt:brocade-interface">
    <tengigabitethernet>
        <name>22/0/1</name>
        <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
             operation="delete"/>
        <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
            <guard>
                <root/>
            </guard>
        </spanning-tree>
    </tengigabitethernet>
</interface>
```

   For detailed information, refer to "Enabling the guard root (STP and RSTP)" on page 338.

10. Issue the <bna-config-cmd> RPC to save the *running-config* file to the *startup-config* file.

The following example enables RSTP, configures RSTP parameters, configures two interfaces as edge ports, and designates one interface the root port.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree>
```

```
                <rstp>
                    <bridge-priority>28582</bridge-priority>
                    <forward-delay>20</forward-delay>
                    <max-age>25<max-age>
                    <error-disabled-timeout>
                        <enable/>
                        <interval>60</interval>
                    </error-disabled-timeout>
                    <port-channel>
                        <path-cost>custom</path-cost>
                    </port-channel>
                    <hello-time>5</hello-time>
                </rstp>
            </spanning-tree>
        </protocol>
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
            <tengigabitethernet>
                <name>22/0/10</name>
                <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <edgeport>
                        <edgeportbasic/>
                    </edgeport>
                </spanning-tree>
            </tengigabitethernet>
            <tengigabitethernet>
                <name>22/0/11</name>
                <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <edgeport>
                        <edgeportbasic/>
                    </edgeport>
                </spanning-tree>
            </tengigabitethernet>
            <tengigabitethernet>
                <name>22/0/13</name>
                <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete"/>
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <priority>32</priority>
                </spanning-tree>
            </tengigabitethernet>
        </interface>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring MSTP

The basic process for configuring MSTP is as follows.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<rpc message-id="2002" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree>
                    <mstp>
                        <region>brocade1</region>
                        <revision>1</revision>
                        <instance>
                            <id>1</id>
                            <vlan>2,3</vlan>
                            <priority>4096</priority>
                        </instance>
                        <instance>
                            <id>2</id>
                            <vlan>4-6</vlan>
                        </instance>
                        <max-hops>25</max-hops>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2002" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring PVST and Rapid PVST

The basic process for configuring PVST or Rapid PVST is as follows.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2003" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree>
                    <pvst>
                        <bridge-priority>4096</bridge-priority>
                        <forward-delay>4</forward-delay>
                        <hello-time>2</hello-time>
                        <max-age>7</max-age>
                    </pvst>
                </spanning-tree xmlns=”urn:brocade.com:mgmt:brocade-xstp”>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2003" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <ok/>
</rpc-reply>
```

Spanning tree configuration and management

This section provides procedures for setting global spanning tree parameters.

NOTE
Issue the <bna-config-cmd> RPC to save your configuration changes.

Enabling STP, RSTP, MSTP, PVST, or Rapid PVST

To enable STP, RSTP, MSTP, PVST, or Rapid PVST, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2004" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp”>
                    <rstp/>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2004" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling STP, RSTP, MSTP, PVST, or Rapid PVST

NOTE
This procedure deletes the context and all the configurations defined within the context or protocol for the interface.

By default, STP, RSTP, MSTP, PVST, and Rapid PVST are not enabled.

To disable STP, RSTP, MSTP, PVST, or Rapid PVST

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2005" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp”
                 xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
```

```
                operation="delete"/>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2005" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Stopping STP, RSTP, MSTP, PVST, or Rapid PVST globally

To shut down STP, RSTP, MSTP, PVST, or Rapid PVST, globally, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2006" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <rstp>
                        <shutdown/>
                    </rstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2006" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the bridge priority for all xSTP

To specify the bridge priority, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2007" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <bridge-priority>20480</bridge-priority>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="2007" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the bridge priority on a per-VLAN basis

To specify a bridge priority for a specific VLAN, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2008" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <pvst>
                        <vlan>
                            <id>100</id>
                            <priority>20480</priority>
                    </pvst>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2008" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the bridge forward delay for all xSTP

To specify the bridge forward delay, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2009" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <forward-delay>20</forward-delay>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2009" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying bridge forward delay on a per-VLAN basis

To specify the bridge forward delay for a specific VLAN, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <rpvst>
                        <vlan>
                            <id>200</id>
                            <forward-delay>20</forward-delay>
                        </vlan>
                    </rpvst>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the bridge maximum aging time for all xSTP

To specify the bridge maximum aging time, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2011" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <stp>
                        <max-age>25</max-age>
                    </stp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2011" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the bridge maximum aging time

To specify the bridge maximum aging time for a VLAN, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2012" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <pvst>
                        <vlan>
                            <id>200</id>
                            <max-age>25</max-age>
                        </vlan>
                    </pvst>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2012" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling the error disable timeout timer for all xSTP

By default, the timeout feature is disabled.

To enable the error disable timeout timer, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2013" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <error-disable-timeout>
                            <enable/>
                        </error-disable-timeout>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2013" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the error disable timeout interval for all xSTP

To specify the time in seconds it takes for an interface to timeout, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2014" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <error-disable-timeout>
                            <interval>60</interval>
                        </error-disable-timeout>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2014" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the port-channel path cost for all xSTP

To specify the port-channel path cost, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2015" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <port-channel>
                            <path-cost>custom</path-cost>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2015" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the bridge hello time for all xSTP

To specify the bridge hello time, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2016" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
```

```
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <pvst>
                        <hello-time>5</hello-time>
                    </pvst>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2016" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the bridge hello time per VLAN (PVST or RPVST)

To specify the bridge hello time for a specific VLAN, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2017" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <pvst>
                        <vlan>
                            <id>200</vlan>
                            <hello-time>5</hello-time>
                        </vlan>
                    </pvst>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2017" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the transmit hold count

To specify the transmit hold count, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2018" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
```

```
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <transmit-holdcount>10</transmit-holdcount>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2018" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling Cisco interoperability (MSTP)

To enable interoperability with certain legacy Cisco switches, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2019" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <cisco-interoperability>enable</cisco-interoperability>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2019" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling Cisco interoperability (MSTP)

If you no longer require the ability to interoperate with certain Cisco legacy switches, use this procedure to deactivate this feature. By default, this ability is deactivated.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2020" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <cisco-interoperability>disable</cisco-interoperability>
```

```
                </mstp>
            </spanning-tree>
        </protocol>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="2020" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Mapping a VLAN to an MSTP instance

Use the this procedure to map a VLAN to an MTSP instance. You can group a set of VLANs to an instance. This <mstp> element can be mapped only after the VLAN is created. Refer to "Creating a VLAN interface" on page 251. VLAN instance mapping is removed from the configuration if the underlying VLANs are deleted.

To map a VLAN to an MSTP instance

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2021" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <instance>
                            <id>5</id>
                            <vlan>300</vlan>
                        </instance>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>


<rpc-reply message-id="2021" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the maximum number of hops for a BPDU (MSTP)

To configure the maximum number of hops for a BPDU in an MSTP region.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2022" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
```

```
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <max-hops>30</max-hops>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2022" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying a name for an MSTP region

Use this procedure to assign a name to an MSTP region. The region name has a maximum length of 32 characters and is case-sensitive.

To assign a name to an MSTP region

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2023" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <region>sydney</region>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2023" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying a revision number for MSTP configuration

Use this procedure to specify a revision number for an MSTP configuration. The range is 0 through 255. The default is 0.

To specify a revision number for an MSTP configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2024" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
```

```
                <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                    <mstp>
                        <revision>17</revision>
                    </mstp>
                </spanning-tree>
            </protocol>
        </config>

    </edit-config>
</rpc>

<rpc-reply message-id="2024" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Retrieving spanning tree-related information

Use the <get-stp-brief-info> custom RPC to display STP, RSTP, MSTP, PVST, or Rapid-PVST-related information.

Issue the <get-stp-brief-info> custom RPC located in the urn:brocade.com:mgmt:brocade-xstp-ext namespace without any input parameters to retrieve the first spanning tree instance. If multiple spanning tree instances exist, the <has-more> element in the output returns “true”. If the <has-more> element returns true, you can retrieve the next spanning tree instance by specifying the value returned in the <instance-id> element of the output as an input parameter in the <,last-rcvd-instance>/<instance-d> field of the next invocation of the <get-stp-brief-info> RPC.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc 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"/>
</rpc>

<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-bidge>
                    <bridge>
                        <priority>32768</priority>
                        <bridge-id>22</bridge-id>
                        <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>Tengigabitethernet</interface-type>
                        <interface-name>22/0/1</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>
```

Reissue the RPC, using the value returned in the <instance-id> element as an input parameter to return the next spanning tree instance. You can continue to repeat the RPC until <has-more> returns false.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2026">
    <get-stp-brief-info xmlns="urn:brocade.com:mgmt:brocade-xstp-ext">
        <last-rcvd-instance>
            <instance-id>91</instance-id>
        </last-rcvd-instance>
    </get-stp-brief-info>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2026">
    <get-stp-brief-info xmlns="urn:brocade.com:mgmt:brocade-xstp-ext">
        <spanning-tree-info>

(output truncated)

        </spanning-tree-info>
        <has-more>false</has-more>
        <last-instance>
            <instnace-id>92</instance-id>
        </last-instance>
    </get-stp-brief-info>
</rpc-reply>
```

Retrieving spanning tree MSTP related information

Use the <get-stp-mst-detail> custom RPC to display MSTP related information.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:message-id="2">
   <get-stp-mst-detail
xmlns="urn:brocade.com:mgmt:brocade-xstp-ext"></get-stp-mst-detail>
</nc:rpc>

<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">
   <cist xmlns="urn:brocade.com:mgmt:brocade-xstp-ext">
      <cist-root-id>8000.01e0.5200.3289</cist-root-id>
      <cist-bridge-id>8000.01e0.5200.3289</cist-bridge-id>
      <cist-reg-root-id>8000.01e0.5200.3289</cist-reg-root-id>
      <root-forward-delay>15</root-forward-delay>
      <hello-time>2</hello-time>
      <max-age>20</max-age>
      <max-hops>20</max-hops>
      <migrate-time>3</migrate-time>
```

```
<port>
   <interface-type>tengigabitethernet</interface-type>
   <interface-name>14/1/11</interface-name>
   <spanningtree-enabled>false</spanningtree-enabled>
   <if-index>203513856</if-index>
   <interface-id>32768</interface-id>
   <if-role>disabled</if-role>
   <if-state>forwarding</if-state>
   <internal-path-cost>0</internal-path-cost>
   <external-path-cost>0</external-path-cost>
   <configured-path-cost>2000</configured-path-cost>
   <designated-port-id>0</designated-port-id>
   <port-priority>128</port-priority>
   <designated-bridge-id>0000.0000.0000.0000</designated-bridge-id>
   <forward-transitions-count>0</forward-transitions-count>
   <port-hello-time>2</port-hello-time>
   <received-stp-type>none</received-stp-type>
   <transmitted-stp-type>mstp</transmitted-stp-type>
   <edge-port>off</edge-port>
   <auto-edge>no</auto-edge>
   <edge-delay>3</edge-delay>
   <admin-edge>no</admin-edge>
   <boundary-port>yes</boundary-port>
   <configured-root-guard>off</configured-root-guard>
   <oper-root-guard>off</oper-root-guard>
   <oper-bpdu-guard>off</oper-bpdu-guard>
   <oper-bpdu-filter>off</oper-bpdu-filter>
   <link-type>point-to-point</link-type>
   <rx-bpdu-count>0</rx-bpdu-count>
   <tx-bpdu-count>0</tx-bpdu-count>
</port>
<port>
   <interface-type>tengigabitethernet</interface-type>
   <interface-name>14/1/21</interface-name>
   <spanningtree-enabled>true</spanningtree-enabled>
   <if-index>203595776</if-index>
   <interface-id>32770</interface-id>
   <if-role>designated</if-role>
   <if-state>forwarding</if-state>
   <internal-path-cost>0</internal-path-cost>
   <external-path-cost>0</external-path-cost>
   <configured-path-cost>2000</configured-path-cost>
   <designated-port-id>32770</designated-port-id>
   <port-priority>128</port-priority>
   <designated-bridge-id>8000.01e0.5200.3289</designated-bridge-id>
   <forward-transitions-count>1</forward-transitions-count>
   <port-hello-time>2</port-hello-time>
   <received-stp-type>mstp</received-stp-type>
   <transmitted-stp-type>mstp</transmitted-stp-type>
   <edge-port>off</edge-port>
   <auto-edge>no</auto-edge>
   <edge-delay>3</edge-delay>
   <admin-edge>no</admin-edge>
   <boundary-port>yes</boundary-port>
   <configured-root-guard>off</configured-root-guard>
   <oper-root-guard>off</oper-root-guard>
   <oper-bpdu-guard>off</oper-bpdu-guard>
   <oper-bpdu-filter>off</oper-bpdu-filter>
   <link-type>point-to-point</link-type>
   <rx-bpdu-count>3</rx-bpdu-count>
```

```
            <tx-bpdu-count>263</tx-bpdu-count>
        </port>
        <port>
            <interface-type>tengigabitethernet</interface-type>
            <interface-name>14/1/22</interface-name>
            <spanningtree-enabled>true</spanningtree-enabled>
            <if-index>203603968</if-index>
            <interface-id>32771</interface-id>
            <if-role>designated</if-role>
            <if-state>forwarding</if-state>
            <internal-path-cost>0</internal-path-cost>
            <external-path-cost>0</external-path-cost>
            <configured-path-cost>2000</configured-path-cost>
            <designated-port-id>32771</designated-port-id>
            <port-priority>128</port-priority>
            <designated-bridge-id>8000.01e0.5200.3289</designated-bridge-id>
            <forward-transitions-count>1</forward-transitions-count>
            <port-hello-time>2</port-hello-time>
            <received-stp-type>mstp</received-stp-type>
            <transmitted-stp-type>mstp</transmitted-stp-type>
            <edge-port>off</edge-port>
            <auto-edge>no</auto-edge>
            <edge-delay>3</edge-delay>
            <admin-edge>no</admin-edge>
            <boundary-port>yes</boundary-port>
            <configured-root-guard>off</configured-root-guard>
            <oper-root-guard>off</oper-root-guard>
            <oper-bpdu-guard>off</oper-bpdu-guard>
            <oper-bpdu-filter>off</oper-bpdu-filter>
            <link-type>point-to-point</link-type>
            <rx-bpdu-count>1</rx-bpdu-count>
            <tx-bpdu-count>252</tx-bpdu-count>
        </port>
    </cist>
    <msti xmlns="urn:brocade.com:mgmt:brocade-xstp-ext">
        <instance-id>2</instance-id>
        <msti-root-id>8002.01e0.5200.3289</msti-root-id>
        <msti-bridge-id>8002.01e0.5200.3289</msti-bridge-id>
        <msti-bridge-priority>32770</msti-bridge-priority>
        <port>
            <interface-type>tengigabitethernet</interface-type>
            <interface-name>14/1/21</interface-name>
            <spanningtree-enabled>true</spanningtree-enabled>
            <if-index>203595776</if-index>
            <interface-id>32770</interface-id>
            <if-role>designated</if-role>
            <if-state>forwarding</if-state>
            <internal-path-cost>0</internal-path-cost>
            <configured-path-cost>2000</configured-path-cost>
            <designated-port-id>32770</designated-port-id>
            <port-priority>128</port-priority>
            <designated-bridge-id>8002.01e0.5200.3289</designated-bridge-id>
            <forward-transitions-count>1</forward-transitions-count>
            <received-stp-type>mstp</received-stp-type>
            <transmitted-stp-type>mstp</transmitted-stp-type>
            <edge-port>off</edge-port>
            <auto-edge>no</auto-edge>
            <edge-delay>3</edge-delay>
            <admin-edge>no</admin-edge>
            <boundary-port>yes</boundary-port>
```

```
            <rx-bpdu-count>3</rx-bpdu-count>
            <tx-bpdu-count>263</tx-bpdu-count>
         </port>
         <port>
            <interface-type>tengigabitethernet</interface-type>
            <interface-name>14/1/22</interface-name>
            <spanningtree-enabled>true</spanningtree-enabled>
            <if-index>203603968</if-index>
            <interface-id>32771</interface-id>
            <if-role>designated</if-role>
            <if-state>forwarding</if-state>
            <internal-path-cost>0</internal-path-cost>
            <configured-path-cost>2000</configured-path-cost>
            <designated-port-id>32771</designated-port-id>
            <port-priority>128</port-priority>
            <designated-bridge-id>8002.01e0.5200.3289</designated-bridge-id>
            <forward-transitions-count>1</forward-transitions-count>
            <received-stp-type>mstp</received-stp-type>
            <transmitted-stp-type>mstp</transmitted-stp-type>
            <edge-port>off</edge-port>
            <auto-edge>no</auto-edge>
            <edge-delay>3</edge-delay>
            <admin-edge>no</admin-edge>
            <boundary-port>yes</boundary-port>
            <rx-bpdu-count>1</rx-bpdu-count>
            <tx-bpdu-count>252</tx-bpdu-count>
         </port>
      </msti>
      <has-more xmlns="urn:brocade.com:mgmt:brocade-xstp-ext">false</has-more>
</rpc-reply>
```

Configuring all xSTP on DCB interface ports

This section details the procedures for enabling and configuring STP, RSTP, MSTP, PVST, or Rapid PVST on individual 10-Gigabit, 1-Gigabit, and 40-Gigabit Ethernet DCB interface ports and port channels.

NOTE
In Brocade VCS Fabric mode, all STP options are disabled.

NOTE
Issue the <bna-config-cmd> RPC to save your configuration changes.

Enabling automatic edge detection (RSTP, MSTP, or RPVST)

Use this procedure to automatically identify the edge port. The port can become an edge port if no BPDU is received. By default, automatic edge detection is disabled.

To enable automatic edge detection on the DCB interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2027" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
      <target>
         <running/>
```

```
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <autoedge/>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2027" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the path cost for all xSTP

Use this procedure to configure the path cost for spanning tree calculations. The lower the path cost means there is a greater chance of the interface becoming the root port. The range is 1 through 200000000. The default path cost is 2000 for a 10 Gbps interface.

To configure the path cost for spanning tree calculations on the DCB interface

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2028" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <cost>10000</cost>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2028" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the path cost per VLAN (PVST or Rapid PVST)

Use this procedure to configure the path cost for spanning tree calculations on a per VLAN basis. The lower the path cost means there is a greater chance of the interface becoming the root port. The range is 1 through 200000000. The default path cost is 2000 for a 10 Gbps interface. For the VLANs which have been configured explicitly, the per-VLAN configuration takes precedence over the global configuration.

The VLAN ID value can be 1 through 3583. VLAN IDs 3584 through 4094 are internally-reserved VLAN IDs.

To configure the path cost for spanning tree calculations on the DCB interface for a specific VLAN,

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2029" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <vlan>
                            <id>200</id>
                            <cost>10000</cost>
                        </vlan>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2029" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling a port (interface) as an edge port

Use this procedure to enable the port as an edge port to allow the port to quickly transition to the forwarding state. To configure a port as an edge port, follow these guidelines:

- A port can become an edge port if no BPDU is received.
- When an edge port receives a BPDU, it becomes a normal spanning tree port and is no longer an edge port.
- Because ports that are directly connected to end stations cannot create bridging loops in the network, edge ports transition directly to the forwarding state and skip the listening and learning states.
- This procedure is only for RSTP, MSTP, and Rapid PVST. Use the <spanning-tree>/<portfast> element for STP and PVST (refer to "Enabling port fast (STP and PVST)" on page 341).

To enable the DCB interface as an edge port, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2030" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <edgeport>
                            <edgeportbasic/>
                        </edgeport>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2030" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling the guard root (STP and RSTP)

To enable the guard root on a DCB interface, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2031" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <guard>
                            <root/>
                        </guard>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2031" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling the guard root per LAN (PVST and Rapid PVST)

Use this procedure to enable the guard root on the switch for a specific VLAN. For the VLANs which have been configured explicitly, the per-VLAN configuration takes precedence over the global configuration.

The guard root feature provides a way to enforce the root bridge placement in the network. With the guard root enabled on an interface, the switch is able to restrict which interface is allowed to be the spanning tree root port or the path to the root for the switch. The root port provides the best path from the switch to the root switch. By default, guard root is disabled.

Guard root protects the root bridge from malicious attacks and unintentional misconfigurations in which a bridge device that is not intended to be the root bridge becomes the root bridge. Such attacks can cause severe bottlenecks in the data path. Guard root ensures that the port on which it is enabled is a designated port. If the guard root-enabled port receives a superior BPDU, it goes to a discarding state.

The VLAN ID value can be 1 through 3583. VLAN IDs 3584 through 4094 are internally-reserved VLAN IDs.

To enable the guard root on a DCB interface for a specific VLAN

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2032" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <vlan>
                            <id>100</id>
                            <guard>
                                <root/>
                            </guard>
                        </vlan>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2032" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the MSTP hello time

Use this procedure to set the time interval between BPDUs sent by the root switch. Changing the hello time affects all spanning tree instances.

The <max-age> setting must be greater than the <hello-time> setting (refer to “Specifying the bridge maximum aging time for all xSTP” on page 324). The range is 1 through 10 seconds. The default is 2 seconds.

To specify the MSTP hello time on a DCB interface

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2033" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <hello-time>5</hello-time>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2033" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying restrictions for an MSTP instance

Use this procedure to specify restrictions on the interface for an MSTP instance.

To specify restrictions for an MSTP instance on a DCB interface

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2034" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <instance>
                            <id>5</id>
                            <retricted-tcn/>
                        </instance>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="2034" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying a link type

To specify a link type on a DCB interface, perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2035" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <link-type>shared</link-type>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2035" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling port fast (STP and PVST)

To enable port fast on the DCB interface for STP

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2036" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <portfast>
                            <portfastbasic/>
                        </portfast>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
```

```
    </edit-config>
</rpc>

<rpc-reply message-id="2036" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the port priority

Use this procedure to specify the port priority. The range is 0 through 240 in increments of 16. The default is 128. A lower number designates a higher priority.

To specify the port priority on the DCB interface

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2037" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <priority>32</priority>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2037" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the port priority per VLAN (PVST and Rapid PVST)

Use this procedure to specify the port priority on a per VLAN basis. The range is 0 through 240 in increments of 16. The default is 128. A lower number designates a higher priority. For the VLANs which have been configured explicitly, the per-VLAN configuration takes precedence over the global configuration.

The VLAN ID value can be 1 through 3583. VLAN IDs 3584 through 4094 are internally-reserved VLAN IDs.

To specify the port priority on the DCB interface for a specific VLAN

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2038" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
```

```
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <vlan>
                            <id>100</id>
                            <priority>32</priority>
                        </vlan>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2038" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Restricting the port from becoming a root port (MSTP)

Use this procedure to restrict a port from becoming a root port. The default is to allow the DCB interface to become a root port. This procedure affects MSTP only.

To restrict the DCB interface from becoming a root port

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2039" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <restricted-role/>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2039" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Restricting the topology change notification (MSTP)

Use this procedure to restrict the topology change notification BPDUs sent on the interface. By default, the restriction is disabled. This procedure affects MSTP only.

To restrict the topology change notification BPDUs sent on the DCB interface

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2040" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <restricted-tcn/>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2040" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling spanning tree

To enable spanning tree on the DCB interface

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2041" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2041" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling spanning tree

By default, spanning tree is disabled.

To disable spanning tree on the DCB interface

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2042" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    <spanning-tree xmlns="urn:brocade.com:mgmt:brocade-xstp">
                        <shutdown/>
                    </spanning-tree>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2042" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Configuring UDLD 32

In this chapter

Overview of UDLD and NETCONF

This chapter provides procedures for configuring Unidirectional Link Detection (UDLD) using the NETCONF interface. Refer to the *Network OS Layer 2 Switching Configuration Guide* for information on UDLD and how it works.

Through the NETCONF interface, you can perform the following operations that affect the functioning of UDLD:

- Use the <edit-config> RPC to configure, enable, or disable the UDLD protocol globally.
- Use the <edit-config> RPC to configure, enable, or disable UDLD on specific 100-Gigabit, 10-Gigabit, 40-Gigabit, or Gigabit Ethernet interfaces.
- Use the <get-config> RPC to verify all or part of the global or per-interface UDLD configuration.

UDLD must be enabled globally before it can be enabled on a specific interface.

UDLD parameters are defined in the brocade-udld YANG module. For information about the brocade-udld YANG module, refer to the *Network OS YANG Reference Manual*.

Configuring UDLD

To enable UDLD globally, perform the following steps.

1. Under the <protocol> node, include the <UDLD> node element from the urn:brocade.com:mgmt:brocade-udld namespace to enable UDLD. The <UDLD> node element contains elements that allow you to configure the global UDLD parameters. However, the "presence=true" statement that qualifies the <udld> container definition in the brocade-udld.yang file allows the <udld> node element to also function as a leaf element. The following example enables UDLD globally.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
            <target>
                <running/>
            </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <udld xmlns="urn:brocade.com:mgmt:brocade-udld">
```

```
                        <hello>20</hello>
                        <multiplier>8</multiplier>
                    </udld>
                </protocol>
            </config>
        </edit-config>
    </rpc>

    <rpc-reply message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <ok/>
    </rpc-reply>
```

2. Under the <interface> node, specify the <fortygigabitethernet>, <tengigabitethernet>, or <gigabitethernet> node, include the <udld> node element from the urn:brocade.com:mgmt:brocade-udld namespace to an enabled UDLD on an interface.
3. Under the <udld> node, enable UDLD for the port interface with the <udld-enable/> leaf element to an enabled UDLD on an interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2407" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
            <target>
                <running/>
            </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <udld xmlns="urn:brocade.com:mgmt:brocade-udld">
                        <udld-enable/>
                    </udld>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2407" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

NOTE
When the UDLD protocol is enabled on one end of a link, the timeout period might elapse before the UDLD protocol is enabled on the other end of the link. In this case, the link becomes temporarily blocked. When the UDLD protocol is enabled at the other end of the link and a UDLD PDU is received, UDLD automatically unblocks the link.

Disabling UDLD

To disable UDLD

```
<?xml version="1.0" encoding="UTF-8"?>
    <rpc message-id="1103" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <edit-config>
            <target>
                <running/>
            </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <udld xmlns="urn:brocade.com:mgmt:brocade-udld">
                    <shutdown/>
                </udld>
            </protocol>
        </config>
        </edit-config>
    </rpc>

<rpc-reply message-id="1103" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Retrieving UDLD statistics

Use the <get-config> RPC to retrieve the current configuration data and operational state data. Refer to “Retrieving configuration data” on page 11 and “Retrieving operational data” on page 15 for detailed instructions.

Chapter

Configuring Link Aggregation 33

In this chapter

Link aggregation with NETCONF overview

This chapter provides procedures for configuring Link Aggregation Group (LAG) and Virtual Link Aggregation Group (vLAG) using NETCONF interfaces. Refer to the *Network OS Layer 2 Switching Configuration Guide* for the following related information:

- An overview of what link aggregation is and how it works
- Guidelines for configuring link aggregation groups
- How the Link Aggregation Control Protocol (LACP) works
- An explanation of the supported types of link aggregation: static, dynamic, Brocade proprietary
- An overview of virtual link aggregation
- Configuration guidelines and restrictions
- Troubleshooting tips

Through the NETCONF interface, you can perform the following operations on LAGs and vLAGs:

- Use the <edit-config> RPC to configure a vLAG and LACP.
- Use the <get-port-channel-detail> and <get-portchannel-info-by-intf> Custom RPCs to obtain operational state information about port channels.
- Use the <get-config> RPC to validate configuration settings.

  LAG parameters are defined in the brocade-lag YANG module. LACP parameters are defined in the brocade-lacp YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Configuring a vLAG

To configure a vLAG, you must configure a port-channel interface on each of the member nodes of the vLAG. Perform the following steps on each node.

1. Create a LAG that uses two switches within the Brocade VCS Fabric.

   When the Brocade VCS Fabric detects that the LAG configuration spans multiple switches, the LAG automatically becomes a vLAG.

2. Configure each vLAG to treat FCoE MAC addresses as being multi-homed hosts, similar to LAN traffic.

   The default configuration is to treat FCoE traffic as non-vLAG traffic. This operation must be performed on every switch in the vLAG.

   The following example configures port channel interface 10.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <port-channel>
                    <name>10</name>
                </port-channel>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Use the <get-port-channel-detail> custom RPC defined in the urn:brocade.com:mgmt:brocade-lag namespace to verify the port channel details for the local switch.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1002">
    <get-port-channel-detail xmlns="urn:brocade.com:mgmt:brocade-lag">
    </get-port-channel-detail>
</rpc>

<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</individidial-agg>
            <ready-agg>1</ready-agg>
            <partner-oper-key>0027</partner-oper-key>
            <aggr-member>
                <rbridge-id>231</rbridge-id>
                <interface-type>tengigabitethernet</interface-type>
                <interface-name>231/0/22</interface-name>
                <actor-port>0xE718160201</actor-port>
                <sync>1</sync>
```

```
            </aggr-member>
            <aggr-member>
                <rbridge-id>231</rbridge-id>
                <interface-type>tengigabitethernet</interface-type>
                <interface-name>231/0/23</interface-name>
                <actor-port>0xE718170202</actor-port>
                <sync>1</sync>
            </aggr-member>
            <aggr-member>
                <rbridge-id>231</rbridge-id>
                <interface-type>tengigabitethernet</interface-type>
                <interface-name>231/0/36</interface-name>
                <actor-port>0xE718240305</actor-port>
                <sync>1</sync>
            </aggr-member>
            <aggr-member>
                <rbridge-id>231</rbridge-id>
                <interface-type>tengigabitethernet</interface-type>
                <interface-name>231/0/37</interface-name>
                <actor-port>0xE718250306</actor-port>
                <sync>1</sync>
            </aggr-member>
        </lacp>
        <has-more>true</has-more>
    </get-port-channel-detail>
</rpc-reply>
```

4. To obtain details of additional port channels configured on this switch, reissue the <get-port-channel-detail> RPC, include the <last-aggregator-id> input parameter, and set its value to the value returned in the <aggregator-id> element of the previous call.

   You should reissue the <get-port-channel-detail> RPC only if the <has-more> element returned "true" in the previous call.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1003">
    <get-port-channel-detail xmlns="urn:brocade.com:mgmt:brocade-lag">
        <last-aggregator-id>27</last-aggregator-id>
    </get-port-channel-detail>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1003">
    <get-port-channel-detail xmlns="urn:brocade.com:mgmt:brocade-lag">
        <lacp>
            <aggregator-id>28</aggregator-id>
            <aggregator-type>standard</aggregator-type>
(output truncated)

        <has-more>false</has-more>
    </get-port-channel-detail>
</rpc-reply>
```

5. Use the <get-portchannel-info-by-intf> custom RPC defined in the urn:brocade.com:mgmt:brocade-lag namespace to verify the port-channel interface details.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1004">
    <get-port-channel-info-by-intf xmlns="urn:brocade.com:mgmt:brocade-lag">
        <interface-type>tengigabitethernet></interface-type>
        <interface-name>1/0/21</interface-name>
    </get-port-channel-info-by-intf>
```

```
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1004">
    <get-port-channel-info-by-intf xmlns="urn:brocade.com:mgmt:brocade-lag">
        <lacp>
            <interface-type>tengigabitethernet</interface-type>
            <interface-name>1/0/21</interface-name>
            <actor-port>0x18150014</actor-port>
            <admin-key>10</admin-key>
            <oper-key>0</oper-key>
            <actor-system-id>01-e0-52-00-01-00</actor-system-id>
            <partner-system-id>01-80-c2-00-00-01</partner-system-id>
            <system-priority>32768</system-priority>
            <partner-oper-priority>32768</partner-oper-priority>
            <actor-priority>32768</actor-priority>
            <receive-machine-state>current</recieve-machine-state>
            <periodic-transmission-machine-state>slow-periodic
                </periodic-transmission-machine-state>
            <mux-machine-state>collecting-distributing</mux-machine-state>
            <admin-state>activity aggregation defaulted</admin-state>
            <oper-state>activity aggregation synchronization collecting
                 distributing</oper-state>
            <partner-oper-state>activity aggregation synchronization collecting
                 distributing</partner-oper-state>
            <partner-oper-port>100</partner-oper-port>
        </lacp>
    </get-port-channel-info-by-intf>
</rpc-reply>
```

Configuring the vLAG ignore split option

This procedure is for LACP-based vLAGs. The scope of this configuration is per port-channel. In scenarios where the vLAG spans more than one node, it minimizes the extent of packet loss in the event of one of the nodes in the vLAG going down.

In a case where connectivity between nodes is lost due to a fabric split (as opposed to one of its members going down), duplication of multicast/broadcast packets will occur.

Brocade recommends that you build redundancy in the fabric so that individual links are not single points of failure.

Figure 1 displays a dual vLAG configuration with three legs of RB2, RB3, and RB4. If RB2, RB3, or RB4 reboots while Host-1 is communicating to Host-2 or Host3, a momentary traffic disruption may occur.

NOTE
With ignore-split active, a vLAG node reboot can result in a more than one second loss while interoperating with a Linux server/nic-team/CNA, due to premature egress of traffic from the server.

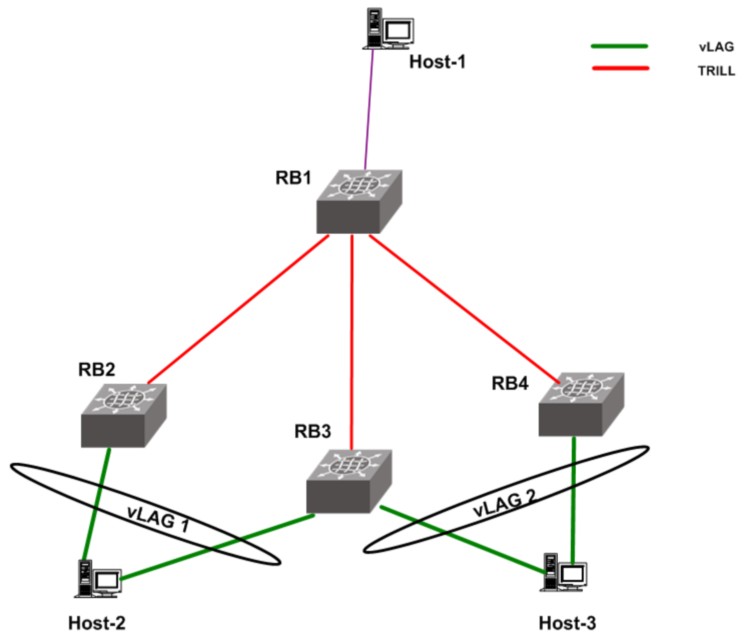


FIGURE 1 **vLAG configuration of the ignore split**

To reduce vLAG failover downtime, you must set the ignore split option on all of the legs in the vLAG (RB2, RB3, and RB4, in this case).

To configure the vLAG ignore split, perform the following steps.

1. Start a NETCONF session with RB2.
2. Activate vLAG ignore split for the first leg.

   The <name> element in the following example identifies the port-channel number.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1005" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <port-channel>
                    <name>1</name>
                    <vlag>
                        <ignore-split/>
                    </vlag>
                </port-channel>
            </interface>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="1005" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Start a NETCONF session with RB3.
4. Activate vLAG ignore split for the second leg.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1006" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <port-channel>
                    <name>2</name>
                    <vlag>
                        <ignore-split/>
                    </vlag>
                </port-channel>
                <port-channel>
                    <name>3</name>
                    <vlag>
                        <ignore-split/>
                    </vlag>
                </port-channel>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1006" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

5. Start a NETCONF session with RB4.
6. Activate vLAG ignore split for the third leg.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1007" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <port-channel>
                    <name>4</name>
                    <vlag>
                        <ignore-split/>
                    </vlag>
                </port-channel>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1007" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
```

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

Configuring protected vLAGs groups

A protected vLAG group is used as an alternative to the spanning-tree-protocol for faster convergence and failover in the critical path of the redundant network.

Protected vLAG groups minimize disruption to the network by protecting critical links from data loss.

A protected vLAG group is a pair vLAGs (or LAGs) configured to act as one active and one backup vLAG connection. The vLAGs and the corresponding port-channels must be configured before the protected vLAG groups can be configured. At any point in time only one vLAG is actively forwarding the traffic and the other vLAG is in a discarding state. If the active vLAG goes down, the backup vLAG take over, by changing the vLAG port state from discarding to forwarding. The failover occurs within milliseconds. The protected vLAG group controls the port state of the member vLAGs to avoid looping errors.

Two port-channels are placed into a protected group. In this protected group, one port-channel acts as the active link, and the other port-channel acts as the standby link.

This task assumes that the port channels have already been created.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <port-channel-redundancy-group
xmlns="urn:brocade.com:mgmt:brocade-lag">
                <group-id>27</group-id>
                <port-channel>
                    <name>3</name>
                    <port-channel-active></port-channel-active>
                </port-channel>
                <port-channel>
                    <name>5</name>
                </port-channel>
                <activate></activate>
            </port-channel-redundancy-group>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3″ xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the load balancing feature

The following example sets the flavor to “destination MAC address and VID-based load balancing.”

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1008" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
```

```
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>2</rbridge-id>
                <fabric xmlns=”urn:brocade.com:mgmt:brocade-fabric-service”>
                    <port-channel>
                        <name>20</name>
                        <vlag-load-balance>dst-mac-vid</vlan-load-balance>
                    </port-channel>
                </fabric>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>
<rpc-reply message-id="1008" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example shows use of the <get-config> RPC to retrieve the configuration by displaying the contents of the <rbridge-id> node.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1009" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge-id"/>
                <rbridge-id>2</rbridge-id>
            </rbridge-id>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="1009" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge-id">
        <rbridge-id>2</rbridge-id>
        <interface-nodespecific>
            <ns-vlan>10</ns-vlan>
            <ns-ethernet>100</ns-ethernet>
        </interface-nodespecific>
        <fabric xmlns=”urn:brocade.com:mgmt:brocade-fabric-service”>
            <port-channel>
                <name>10</name>
                <vlag-load-balance>src-dst-mac-vid</vlan-load-balance>
            </port-channel>
        </fabric>
        <fabric xmlns=”urn:brocade.com:mgmt:brocade-fabric-service”>
            <port-channel>
                <name>20</name>
                <vlag-load-balance>dst-mac-vid</vlan-load-balance>
            </port-channel>
        </fabric>
    </rbridge-id>
</rpc-reply>
```

LACP configuration and management

NOTE
To save the configuration, use the <bna-config-cmd> RPC to copy the running configuration to the startup configuration.

Enabling LACP on a DCB interface

To add interfaces to an existing LAG, repeat this procedure using the same LAG group number for the new interfaces.

To enable LACP on a DCB interface

The following example adds port 8/0/1 to channel group 4 in active mode, standard type.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>8/0/1</name>
                    <shutdown xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete"/>
                    <channel-group>
                        <port-int>4</port-int>
                        <mode>active</mode>
                        <type>standard</type>
                    </channel-group>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1010" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the LACP system priority

To configure the global LACP system priority, issue the <edit-config> RPC to configure the <lacp> node in the urn:brocade.com:mgmt:brocade-lacp namespace and provide a value in the <system-priority> leaf element.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1011" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <lacp xmlns="urn:brocade.com:mgmt:brocade-lacp">
```

```
                <system-priority>25000</system-priority>
            </lacp>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1011" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the LACP timeout period on a DCB interface

The LACP timeout period indicates how long LACP waits before timing out the neighboring device. The short timeout period is 3 seconds and the long timeout period is 90 seconds. The default is long.

To configure the LACP timeout period on a DCB interface

The following example sets the LACP timeout period to three seconds for the 1/0/1 interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1012" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
                    <lacp xmlns="urn:brocade.com:mgmt:brocade-lacp">
                        <timeout>short</timeout>
                    </lacp>
                </tengigabitethernet>
            </interface
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1012" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Configuring LLDP

34

In this chapter

LLDP configuration with NETCONF overview

This chapter provides procedures for configuring the Link Layer Discovery Protocol (LLDP) using NETCONF interfaces. Refer to the *Network OS Layer 2 Switching Configuration Guide* for the following related information:

- An overview of LLDP explaining what it is and how it works
- An explanation of Layer 2 topology mapping
- An overview of the Data Center Bridging Capability Exchange Protocol (DCBX), including Enhanced Transmission Selection (ETS) and Process Flow Control (PFC)
- How DCBX interacts with devices from other vendors
- A summary of configuration guidelines and restrictions
- A summary of configuration default values

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

- Use the <edit-config> remote procedure call (RPC) to configure LLDP.
- Use the <get-config> RPC to verify all or part of the LLDP configuration.

LLDP parameters are defined in the brocade-lldp YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all LLDP parameters, refer to the brocade-lldp.yang file.

Enabling and disabling LLDP

NOTE
Use the <bna-config-cmd> RPC to save your configuration changes.

Enabling LLDP globally

This procedure enables LLDP globally on all interfaces unless it has been specifically disabled on an interface. LLDP is enabled globally by default.

To enable LLDP globally

The following example enables LLDP globally.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp"/>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling and resetting LLDP globally

Resetting LLDP globally returns all configuration settings made under the <protocol>/<lldp> node to their default settings. LLDP is enabled globally by default.

Disabling LLDP disables LLDP globally.

To reset LLDP globally

The following example resets LLDP globally.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1102" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp"
                    xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete"/>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1102" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To disable LLDP globally

The following example disables LLDP globally.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1103" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <disable/>
                </lldp>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1103" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring LLDP global options

The global LLDP configuration options are located under the <protocol>/<lldp> node in the urn:brocade.com:mgmt:brocade-lldp namespace.

Specifying a system name and LLDP description

The global system name for LLDP is useful for differentiating among switches. By default, the “host-name” from the chassis/entity MIB is used. By specifying a descriptive system name, you will find it easier to configure the switch for LLDP.

The system description is seen by neighboring switches.

NOTE
Brocade recommends you use the operating system version for the description or use the description from the chassis/entity MIB. Do not use special characters, such as #, $, !, @, as part of the system name and description.

To specify a global system name and system description for the Brocade VDX hardware

The following example specifies a global system name and global system description.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1104" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <system-name>Brocade_Alpha</system-name>
                    <system-description>IT_1.6.2_LLDP_01</system-description>
                </lldp>
            </protocol>
        </config>
```

```
    </edit-config>
</rpc>

<rpc-reply message-id="1104" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying a user description for LLDP

The user description is for network administrative purposes and is not seen by neighboring switches.

To specify a user description for LLDP

The following example specifies a user description for LLDP.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1105" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <description>Brocade-LLDP-installed-july-25</description>
                </lldp>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1105" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the transmission of LLDP frames

By default both transmission and reception of LLDP frames are enabled.

To enable only reception (rx) of LLDP frames

The following example sets the value of the <mode> element to “rx” to enable only reception of LLDP frames.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1106" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <mode>rx</mode>
                </lldp>
            </protocol>
        </config>
    </edit-config>
```

```
</rpc>

<rpc-reply message-id="1106" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following example sets the value of the <mode> element to “tx” to enable only transmission of LLDP frames.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1107" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <mode>tx</mode>
                </lldp>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1107" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To enable both transmit and receive modes, delete the <mode> element by including the delete operation in its tag.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1108" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <mode xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                    operation="delete"/>
                </lldp>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1108" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the transmit frequency of LLDP frames

The default transmit frequency is 30 seconds. The valid range is 4 through 180 seconds.

To configure the transmit frequency of LLDP frames

The following example sets the transmit frequency to 45 seconds.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1109" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <hello>45</hello>
                </lldp>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1109" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the hold time for receiving devices

This procedure configures the number of consecutive LLDP hello packets that can be missed before declaring the neighbor information as invalid. The default value is 4. The valid range is 2 through 10.

To configure the hold time for receiving devices

The following example configures the number of consecutive LLDP packets that can be missed before declaring the neighbor information as invalid to 6 packets.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1110" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <multiplier>6</multiplier>
                </lldp>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1110" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Advertising the optional LLDP TLVs

To configure the optional LLDP type-length-value (TLV) fields

The following example advertises all the optional TLVs.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1111" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <advertise>
                        <optional-tlv>
                            <management-address/>
                            <port-description/>
                            <system-capabilities/>
                            <adv-tlv-system-description/>
                            <adv-tlv-system-name/>
                        </optional-tlv>
                    </advertise>
                </lldp>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1111" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the advertisement of LLDP DCBX-related TLVs

To configure the LLDP DCBX-related TLVs to be advertised

The following example advertises all the DCBX-related TLVs.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1112" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <advertise>
                        <dcbx-fcoe-app-tlv/>
                        <dcbx-fcoe-logical-link-tlv/>
                        <dcbx-tlv/>
                        <dot1-tlv/>
                        <dot3-tlv/>
                    </advertise>
                </lldp>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1112" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring iSCSI priority

To configure the iSCSI priority

The following example sets the iSCSI priority to 4 and advertises this value in the DCBX iSCSI TLV.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1113" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <iscsi-priority>4</iscsi-priority>
                    <advertise>
                        <dcbx-iscsi-app-tlv/>
                    </advertise>
                </lldp>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1113" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring LLDP profiles

You can configure up to 384 profiles on a switch.

To configure an LLDP profile

NOTE
Brocade recommends against advertising dot1.tlv and dot3.tlv LLDPs if your network contains CNAs from non-Brocade vendors. This configuration may cause functionality problems.

The following example configures an LLDP profile named UK_LLDP_IT. It allows transmission and reception of LLDP frames, provides a transmit frequency of 10 seconds, specifies a hold time of 20 seconds, and advertises optional and DCBX-related TLVs.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1114" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <profile>
                        <profile-name>UK_LLDP_IT</profile-name>
                        <description>Standard profile by Jane</description>
                        <mode xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                        operation="delete"/>
                        <hello>10</hello>
```

```
                        <multiplier>2</multiplier>
                        <advertise>
                            <optional-tlv>
                                <management-address/>
                                <port-description/>
                                <system-capabilities/>
                                <adv-tlv-system-description/>
                                <adv-tlv-system-name/>
                            <dot1-tlv/>
                            <dot3-tlv/>
                            <dcbx-tlv/>
                            <dcbx-fcoe-logical-link-tlv/>
                            <dcbx-fcoe-app-tlv/>
                            <dcbx-iscsi-app-tlv/>
                        </advertise>
                    </profile>
                </lldp>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1114" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the iSCSI profile

You can configure an iSCSI profile to be applied to individual interfaces. However, the priority bit must be set explicitly for each interface.

To configure iSCSI profiles, you must first configure the CEE map, if one has not already been created, configure the iSCSI profile, and then apply that profile to the interfaces.

To configure the CEE map

The following example configures a CEE map, configures an iSCSI profile using that map, and applies the profile to an interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1115" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <cee-map xmlns="urn:brocade.com:mgmt:brocade-cee-map">
                <name>default<name/>
                <priority-group-table>
                    <PGID>1</PGID>
                    <weight>50</weight>
                    <pfc>on</pfc>
                </priority-group-table>
                <priority-group-table>
                    <PGID>2</PGID>
                    <weight>30</weight>
                    <pfc>on</pfc
                </priority-group-table>
                <priority-group-table>
```

```
                    <PGID>3</PGID>
                    <weight>20</weight>
                    <pfc>on</pfc
                </priority-group-table>
                <priority-table>
                    <map-cos0-pgid>1</map-cos0-pgid>
                    <map-cos1-pgid>1</map-cos1-pgid>
                    <map-cos2-pgid>1</map-cos2-pgid>
                    <map-cos3-pgid>1</map-cos3-pgid>
                    <map-cos4-pgid>2</map-cos4-pgid>
                    <map-cos5-pgid>3</map-cos5-pgid>
                    <map-cos6-pgid>1</map-cos6-pgid>
                    <map-cos7-pgid>1</map-cos7-pgid>
                </priority-table>
            </cee-map>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <profile>
                        <profile-name>iscsi_config</profile-name>
                        <advertise>
                            <dcbx-iscsi-app-tlv/>
                        </advertise>
                    </profile>
                </lldp>
            </protocol>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
                    <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                        <cee>
                            <lldp-cee-on-off>on</lldp-cee-on-off>
                        </cee>
                        <profile>iscsi_config</profile>
                        <iscsi-priority>4</iscsi-priority>
                    </lldp>
                </tengigabitethernet>
```

(more interfaces go here)

```
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1115" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting an LLDP profile

To delete an LLDP profile

The following example deletes an LLDP profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1116" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
```

```
            <running/>
        </target>
        <config>
            <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                    <profile xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                        operation="delete">
                        <profile-name>UK_LLDP_IT</profile-name>
                    </profile>
                </lldp>
            </protocol>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1116" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring LLDP interface-level options

To configure LLDP interface-level options

The following example applies an LLDP profile named network_standard to interface 22/0/1.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1117" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <lldp xmlns="urn:brocade.com:mgmt:brocade-lldp">
                        <profile>network_standard</profile>
                        <dcbx-version>cee</dcbx-version>
                    </lldp>
                </tengigabitethernet>

                (more interfaces go here)

            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1117" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Retrieving neighbor related information

Use the <get-lldp-neighbor-detail> custom RPC to display the Neighbor details of all the interfaces of the managed entity.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:message-id="4">
   <get-lldp-neighbor-detail
xmlns="urn:brocade.com:mgmt:brocade-lldp-ext"></get-lldp-neighbor-detail>
</nc:rpc>

<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>Te 14/1/3</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>
   <lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
      <local-interface-name>Te 14/1/11</local-interface-name>
      <local-interface-ifindex>203513856</local-interface-ifindex>
      <local-interface-mac>0005.3379.6de8</local-interface-mac>
      <remote-interface-name>TenGigabitEthernet 6/0/11</remote-interface-name>
      <remote-interface-mac>0005.1ecd.7229</remote-interface-mac>
      <dead-interval>120</dead-interval>
      <remaining-life>100</remaining-life>
      <remote-chassis-id>0005.1ecd.71fa</remote-chassis-id>
      <lldp-pdu-transmitted>16160</lldp-pdu-transmitted>
      <lldp-pdu-received>16159</lldp-pdu-received>
   </lldp-neighbor-detail>
   <lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
      <local-interface-name>Te 14/1/12</local-interface-name>
      <local-interface-ifindex>203522048</local-interface-ifindex>
      <local-interface-mac>0005.3379.6de9</local-interface-mac>
      <remote-interface-name>TenGigabitEthernet 6/0/12</remote-interface-name>
      <remote-interface-mac>0005.1ecd.722a</remote-interface-mac>
      <dead-interval>120</dead-interval>
      <remaining-life>100</remaining-life>
      <remote-chassis-id>0005.1ecd.71fa</remote-chassis-id>
      <lldp-pdu-transmitted>16159</lldp-pdu-transmitted>
      <lldp-pdu-received>16159</lldp-pdu-received>
   </lldp-neighbor-detail>
   <lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
      <local-interface-name>Te 14/1/15</local-interface-name>
      <local-interface-ifindex>203546624</local-interface-ifindex>
      <local-interface-mac>0005.3379.6dec</local-interface-mac>
      <remote-interface-name>port0</remote-interface-name>
      <remote-interface-mac>0005.3348.4b0a</remote-interface-mac>
      <dead-interval>120</dead-interval>
      <remaining-life>120</remaining-life>
      <remote-chassis-id>0005.3348.4b0a</remote-chassis-id>
      <lldp-pdu-transmitted>16159</lldp-pdu-transmitted>
      <lldp-pdu-received>15846</lldp-pdu-received>
```

```
</lldp-neighbor-detail>
<lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
   <local-interface-name>Te 14/1/17</local-interface-name>
   <local-interface-ifindex>203563008</local-interface-ifindex>
   <local-interface-mac>0005.3379.6dee</local-interface-mac>
   <remote-interface-name>TenGigabitEthernet 8/0/49</remote-interface-name>
   <remote-interface-mac>0005.3314.1c55</remote-interface-mac>
   <dead-interval>120</dead-interval>
   <remaining-life>115</remaining-life>
   <remote-chassis-id>0005.3314.1c00</remote-chassis-id>
   <lldp-pdu-transmitted>16159</lldp-pdu-transmitted>
   <lldp-pdu-received>16159</lldp-pdu-received>
</lldp-neighbor-detail>
<lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
   <local-interface-name>Te 14/1/21</local-interface-name>
   <local-interface-ifindex>203595776</local-interface-ifindex>
   <local-interface-mac>0005.3379.6df2</local-interface-mac>
   <remote-interface-name>TenGigabitEthernet 9/1/21</remote-interface-name>
   <remote-interface-mac>0005.3379.2b6e</remote-interface-mac>
   <dead-interval>120</dead-interval>
   <remaining-life>112</remaining-life>
   <remote-chassis-id>0005.3379.2a54</remote-chassis-id>
   <lldp-pdu-transmitted>14039</lldp-pdu-transmitted>
   <lldp-pdu-received>63</lldp-pdu-received>
   <remote-system-name>sw0</remote-system-name>
</lldp-neighbor-detail>
<lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
   <local-interface-name>Te 14/1/22</local-interface-name>
   <local-interface-ifindex>203603968</local-interface-ifindex>
   <local-interface-mac>0005.3379.6df3</local-interface-mac>
   <remote-interface-name>TenGigabitEthernet 9/1/22</remote-interface-name>
   <remote-interface-mac>0005.3379.2b6f</remote-interface-mac>
   <dead-interval>120</dead-interval>
   <remaining-life>110</remaining-life>
   <remote-chassis-id>0005.3379.2a54</remote-chassis-id>
   <lldp-pdu-transmitted>14040</lldp-pdu-transmitted>
   <lldp-pdu-received>63</lldp-pdu-received>
   <remote-system-name>sw0</remote-system-name>
</lldp-neighbor-detail>
<lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
   <local-interface-name>Te 14/1/23</local-interface-name>
   <local-interface-ifindex>203612160</local-interface-ifindex>
   <local-interface-mac>0005.3379.6df4</local-interface-mac>
   <remote-interface-name>TenGigabitEthernet 9/1/23</remote-interface-name>
   <remote-interface-mac>0005.3379.2b70</remote-interface-mac>
   <dead-interval>120</dead-interval>
   <remaining-life>106</remaining-life>
   <remote-chassis-id>0005.3379.2a54</remote-chassis-id>
   <lldp-pdu-transmitted>14038</lldp-pdu-transmitted>
   <lldp-pdu-received>61</lldp-pdu-received>
   <remote-system-name>sw0</remote-system-name>
</lldp-neighbor-detail>
<lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
   <local-interface-name>Te 14/1/33</local-interface-name>
   <local-interface-ifindex>203694080</local-interface-ifindex>
   <local-interface-mac>0005.3379.6dfe</local-interface-mac>
   <remote-interface-name>TenGigabitEthernet 18/0/33</remote-interface-name>
   <remote-interface-mac>0005.336f.2d25</remote-interface-mac>
   <dead-interval>120</dead-interval>
   <remaining-life>115</remaining-life>
```

```
      <remote-chassis-id>0005.336f.2ce0</remote-chassis-id>
      <lldp-pdu-transmitted>16159</lldp-pdu-transmitted>
      <lldp-pdu-received>16159</lldp-pdu-received>
   </lldp-neighbor-detail>
   <lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
      <local-interface-name>Te 14/1/34</local-interface-name>
      <local-interface-ifindex>203702272</local-interface-ifindex>
      <local-interface-mac>0005.3379.6dff</local-interface-mac>
      <remote-interface-name>TenGigabitEthernet 18/0/34</remote-interface-name>
      <remote-interface-mac>0005.336f.2d26</remote-interface-mac>
      <dead-interval>120</dead-interval>
      <remaining-life>115</remaining-life>
      <remote-chassis-id>0005.336f.2ce0</remote-chassis-id>
      <lldp-pdu-transmitted>16159</lldp-pdu-transmitted>
      <lldp-pdu-received>16159</lldp-pdu-received>
   </lldp-neighbor-detail>
   <lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
      <local-interface-name>Te 14/4/1</local-interface-name>
      <local-interface-ifindex>209723392</local-interface-ifindex>
      <local-interface-mac>0005.3379.6efe</local-interface-mac>
      <remote-interface-name>TenGigabitEthernet 9/8/1</remote-interface-name>
      <remote-interface-mac>0005.3379.2dfa</remote-interface-mac>
      <dead-interval>120</dead-interval>
      <remaining-life>108</remaining-life>
      <remote-chassis-id>0005.3379.2a54</remote-chassis-id>
      <lldp-pdu-transmitted>14103</lldp-pdu-transmitted>
      <lldp-pdu-received>62</lldp-pdu-received>
      <remote-system-name>sw0</remote-system-name>
   </lldp-neighbor-detail>
   <lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
      <local-interface-name>Te 14/4/2</local-interface-name>
      <local-interface-ifindex>209731584</local-interface-ifindex>
      <local-interface-mac>0005.3379.6eff</local-interface-mac>
      <remote-interface-name>TenGigabitEthernet 9/8/2</remote-interface-name>
      <remote-interface-mac>0005.3379.2dfb</remote-interface-mac>
      <dead-interval>120</dead-interval>
      <remaining-life>108</remaining-life>
      <remote-chassis-id>0005.3379.2a54</remote-chassis-id>
      <lldp-pdu-transmitted>14105</lldp-pdu-transmitted>
      <lldp-pdu-received>61</lldp-pdu-received>
      <remote-system-name>sw0</remote-system-name>
   </lldp-neighbor-detail>
   <lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
      <local-interface-name>Te 14/4/3</local-interface-name>
      <local-interface-ifindex>209739776</local-interface-ifindex>
      <local-interface-mac>0005.3379.6f00</local-interface-mac>
      <remote-interface-name>TenGigabitEthernet 9/8/3</remote-interface-name>
      <remote-interface-mac>0005.3379.2dfc</remote-interface-mac>
      <dead-interval>120</dead-interval>
      <remaining-life>109</remaining-life>
      <remote-chassis-id>0005.3379.2a54</remote-chassis-id>
      <lldp-pdu-transmitted>14104</lldp-pdu-transmitted>
      <lldp-pdu-received>61</lldp-pdu-received>
      <remote-system-name>sw0</remote-system-name>
   </lldp-neighbor-detail>
   <lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
      <local-interface-name>Te 14/4/4</local-interface-name>
      <local-interface-ifindex>209747968</local-interface-ifindex>
      <local-interface-mac>0005.3379.6f01</local-interface-mac>
      <remote-interface-name>TenGigabitEthernet 9/8/4</remote-interface-name>
```

```
    <remote-interface-mac>0005.3379.2dfd</remote-interface-mac>
    <dead-interval>120</dead-interval>
    <remaining-life>115</remaining-life>
    <remote-chassis-id>0005.3379.2a54</remote-chassis-id>
    <lldp-pdu-transmitted>14102</lldp-pdu-transmitted>
    <lldp-pdu-received>62</lldp-pdu-received>
    <remote-system-name>sw0</remote-system-name>
</lldp-neighbor-detail>
<lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
    <local-interface-name>Te 14/4/9</local-interface-name>
    <local-interface-ifindex>209788928</local-interface-ifindex>
    <local-interface-mac>0005.3379.6f06</local-interface-mac>
    <remote-interface-name>TenGigabitEthernet 9/8/8</remote-interface-name>
    <remote-interface-mac>0005.3379.2e01</remote-interface-mac>
    <dead-interval>120</dead-interval>
    <remaining-life>113</remaining-life>
    <remote-chassis-id>0005.3379.2a54</remote-chassis-id>
    <lldp-pdu-transmitted>14098</lldp-pdu-transmitted>
    <lldp-pdu-received>61</lldp-pdu-received>
    <remote-system-name>sw0</remote-system-name>
</lldp-neighbor-detail>
<lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
    <local-interface-name>Te 14/4/19</local-interface-name>
    <local-interface-ifindex>209870848</local-interface-ifindex>
    <local-interface-mac>0005.3379.6f10</local-interface-mac>
    <remote-interface-name>TenGigabitEthernet 9/8/9</remote-interface-name>
    <remote-interface-mac>0005.3379.2e02</remote-interface-mac>
    <dead-interval>120</dead-interval>
    <remaining-life>115</remaining-life>
    <remote-chassis-id>0005.3379.2a54</remote-chassis-id>
    <lldp-pdu-transmitted>14103</lldp-pdu-transmitted>
    <lldp-pdu-received>62</lldp-pdu-received>
    <remote-system-name>sw0</remote-system-name>
</lldp-neighbor-detail>
<lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
    <local-interface-name>Te 14/4/22</local-interface-name>
    <local-interface-ifindex>209895424</local-interface-ifindex>
    <local-interface-mac>0005.3379.6f13</local-interface-mac>
    <remote-interface-name>TenGigabitEthernet 9/8/22</remote-interface-name>
    <remote-interface-mac>0005.3379.2e0f</remote-interface-mac>
    <dead-interval>120</dead-interval>
    <remaining-life>108</remaining-life>
    <remote-chassis-id>0005.3379.2a54</remote-chassis-id>
    <lldp-pdu-transmitted>14080</lldp-pdu-transmitted>
    <lldp-pdu-received>61</lldp-pdu-received>
    <remote-system-name>sw0</remote-system-name>
</lldp-neighbor-detail>
<lldp-neighbor-detail xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">
    <local-interface-name>Te 14/4/23</local-interface-name>
    <local-interface-ifindex>209903616</local-interface-ifindex>
    <local-interface-mac>0005.3379.6f14</local-interface-mac>
    <remote-interface-name>TenGigabitEthernet 9/8/23</remote-interface-name>
    <remote-interface-mac>0005.3379.2e10</remote-interface-mac>
    <dead-interval>120</dead-interval>
    <remaining-life>108</remaining-life>
    <remote-chassis-id>0005.3379.2a54</remote-chassis-id>
    <lldp-pdu-transmitted>14078</lldp-pdu-transmitted>
    <lldp-pdu-received>61</lldp-pdu-received>
    <remote-system-name>sw0</remote-system-name>
</lldp-neighbor-detail>
```

```
   <has-more xmlns="urn:brocade.com:mgmt:brocade-lldp-ext">false</has-more>
</rpc-reply>
```

Chapter

HTTPS Crypto Certificates 35

In this chapter

Configuring HTTPS crypto certificates

This chapter provides procedures for configuring HTTPS crypto certificates. Refer to the *Network OS Security Guide* for the following related information:

- An overview of HTTPS crypto certificates
- Enabling HTTPS crypto certificates
- Disabling HTTPS crypto certificates

Configuring crypto key

The following examples shows how to configure crypto key.

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <crypto xmlns="urn:brocade.com:mgmt:brocade-crypto">
                    <key>
                        <label>key_label</label>
                        <type>rsa</type>
                        <modulus>2048</modulus>
                    </key>
                </crypto>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring crypto ca authenticate

The following examples shows how to configure crypto ca authenticate.

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <crypto xmlns="urn:brocade.com:mgmt:brocade-crypto">
                    <ca>
                        <trustpoint>trust1</trustpoint>
                        <keypair>key_label</keypair>
                    </ca>
                </crypto>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Configuring ACLs 36

In this chapter

ACL configuration with NETCONF overview

This chapter provides procedures for configuring MAC access control lists (ACLs) and IP ACLs using the NETCONF interface. Refer to the *Network OS Security Configuration Guide* for the following related information:

- An overview of ACLs
- Configuration guidelines and restrictions

Through the NETCONF interface, you can perform the following operations on ACLs:

- Use the <edit-config> remote procedure call (RPC) to configure an ACL.
- Use the <get-mac-acl-for-intf> custom RPC to obtain MAC ACLs applied to an interface.
- Use the <get-config> RPC to validate configuration settings.

MAC ACL parameters are defined in the brocade-mac-access-list YANG module. IP ACL parameters are defined in the brocade-ip-access-list YANG module. IPv6 ACL parameters are defined in the brocade-ipv6-access-list YANG module. For a structural overview of these YANG modules, refer to the *Network OS YANG Reference Manual*.

Default ACL configuration

When none of the policies is enforced on the switch, these default ACL rules are effective in Network OS:

- seq 0 permit tcp any any eq 22
- seq 1 permit tcp any any eq 23
- seq 2 permit tcp any any eq 897
- seq 3 permit tcp any any eq 898
- seq 4 permit tcp any any eq 111
- seq 5 permit tcp any any eq 80
- seq 6 permit tcp any any eq 443

- seq 7 permit udp any any eq 161
- seq 8 permit udp any any eq 111
- seq 9 permit tcp any any eq 123
- seq 10 permit tcp any any range 600 65535
- seq 11 permit udp any any range 600 65535

Refer to the *Network OS Security Configuration Guide* for an explanation of ACL rules.

ACL configuration and management

NOTE
Issue the <bna-config-cmd> RPC to save your configuration changes.

Two types of MAC ACL exist:

- Standard—Permit and deny traffic according to the source MAC address in the incoming frame. Use standard MAC ACLs if you only need to filter traffic based on source addresses.
- Extended—Permit and deny traffic according to the source and destination MAC addresses in the incoming frame, as well as EtherType.

Creating a standard MAC ACL and adding rules

A MAC ACL does not take effect until it is applied to a Layer 2 interface. Refer to "Applying a MAC ACL to a DCB interface" on page 382 and "Applying a MAC ACL to a VLAN interface" on page 384.

To create a standard MAC ACL and add rules

The following example creates a standard MAC ACL named test_01 and adds two rules to it:

- Rule 100 drops traffic from source MAC address 0011.2222.3333 and maintains a count of packets dropped.
- Rule 1000 allows traffic from source MAC address 0022.1111.2222 and maintains a count of packets allowed.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac xmlns="urn:brocade.com:mgmt:brocade-mac-access-list">
                <access-list>
                    <standard>
                        <name>MAC_ACL_STD_EXAMPLE</name>
                        <hide-mac-acl-std>
                            <seq>
                                <seq-id>10</seq-id>
                                <action>permit</action>
                                <source>host</source>
                                <srchost>1111.2222.3333</srchost>
                                <count></count>
                            </seq>
                            <seq>
```

```
                                <seq-id>20</seq-id>
                                <action>deny</action>
                                <source>1111.2222.3333</source>

<src-mac-addr-mask>ffff.ffff.0000</src-mac-addr-mask>
                                <count></count>
                                <log></log>
                            </seq>
                            <seq>
                                <seq-id>30</seq-id>
                                <action>hard-drop</action>
                                <source>host</source>
                                <srchost>0005.334b.1929</srchost>
                            </seq>
                        </hide-mac-acl-std>
                    </standard>
                </access-list>
            </mac>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Creating an extended MAC ACL and adding rules

The MAC ACL name length is limited to 64 characters. A MAC ACL does not take effect until it is applied to a Layer 2 interface. Refer to "Applying a MAC ACL to a DCB interface" on page 382 and "Applying a MAC ACL to a VLAN interface" on page 384.

To create an extended MAC ACL and add rules.

The following example creates an extended MAC access list named test_02 with the following rules:

- Rule 5 allows traffic from MAC address 0022.3333.4444 destined for MAC address 0022.333.555 and maintains a count of accepted packets.
- Rule 1000 allows traffic from MAC address 0022.1111.2222 and maintains a count of accepted packets.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac xmlns="urn:brocade.com:mgmt:brocade-mac-access-list">
                <access-list>
                    <extended>
                        <name>MAC_ACL_EXT_EXAMPLE</name>
                        <hide-mac-acl-ext>
                            <seq>
                                <seq-id>10</seq-id>
                                <action>permit</action>
                                <source>any</source>
                                <dst>host</dst>
                                <dsthost>1111.2222.3333</dsthost>
```

```
                                <count></count>
                            </seq>
                            <seq>
                                <seq-id>20</seq-id>
                                <action>deny</action>
                                <source>1111.2222.3333</source>

<src-mac-addr-mask>ffff.ffff.0000</src-mac-addr-mask>
                                <dst>any</dst>
                                <count></count>
                                <log></log>
                            </seq>
                            <seq>
                                <seq-id>30</seq-id>
                                <action>hard-drop</action>
                                <source>host</source>
                                <srchost>0005.334b.1934</srchost>
                                <dst>host</dst>
                                <dsthost>0005.334b.1929</dsthost>
                            </seq>
                            <seq>
                                <seq-id>40</seq-id>
                                <action>deny</action>
                                <source>1111.2222.3333</source>

<src-mac-addr-mask>ffff.ffff.0000</src-mac-addr-mask>
                                <dst>any</dst>
                                <vlan>10</vlan>
                                <count></count>
                                <log></log>
                            </seq>
                            <seq>
                                <seq-id>50</seq-id>
                                <action>hard-drop</action>
                                <source>host</source>
                                <srchost>0005.334b.1934</srchost>
                                <dst>host</dst>
                                <dsthost>0005.334b.1929</dsthost>
                                <ethertype>ipv4</ethertype>
                                <vlan>20</vlan>
                            </seq>
                        </hide-mac-acl-ext>
                    </extended>
                </access-list>
            </mac>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Applying a MAC ACL to a DCB interface

Ensure that the ACL that you want to apply exists and is configured to filter traffic in the manner that you need for a specific DCB interface. An ACL does not take effect until it is expressly applied to an interface. Frames can be filtered as they enter an interface (ingress direction).

NOTE
The DCB interface must be configured as a Layer 2 switchport before an ACL can be applied as an access-group to the interface.

To apply a MAC ACL to a DCB interface

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <switchport>
                        <basic/>
                    </switchport>
                    <mac xmlns="urn:brocade.com:mmgmt:brocade-mac-access-list">
                        <access-group>
                            <mac-access-list>test_02</mac-access-list>
                            <mac-direction>in</mac-direction>
                        </access-group>
                    </mac>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Applying a MAC ACL to a LAG (logical) interface

The following example shows how to apply a MAC ACL to a LAG (logical) interface:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <port-channel>
                    <name>22/0/1</name>
                    <mac xmlns="urn:brocade.com:mmgmt:brocade-mac-access-list">
                        <access-group>
                            <mac-access-list>test_05</mac-access-list>
                            <mac-direction>in</mac-direction>
                        </access-group>
                    </mac>
                </port-channel>
            </interface>
        </config>
```

```
    </edit-config>
</rpc>

<rpc-reply message-id="2402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Applying a MAC ACL to an overlay gateway

The following example shows how to apply a MAC ACL to an overlay gateway:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>gateway1</name>
                <gw-type>nsx</gw-type>
                <mac xmlns="urn:brocade.com:mmgmt:brocade-mac-access-list">
                    <access-group>
                        <mac-access-list>test_05</mac-access-list>
                        <mac-direction>in</mac-direction>
                    </access-group>
                </mac>
            </overlay-gateway>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Applying a MAC ACL to a VLAN interface

Ensure that the ACL that you want to apply exists and is configured to filter traffic in the manner that you need for a specific VLAN interface. An ACL does not take effect until it is expressly applied to an interface. Frames can be filtered as they enter an interface (ingress direction).

To apply a MAC ACL to a VLAN interface

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2403" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>50</name>
                        <mac xmlns="urn:brocade.com:mmgmt:brocade-mac-access-
                         list">
                           <access-group>
```

```
                        <mac-access-list>test_02</mac-access-list>
                        <mac-direction>in</mac-direction>
                    </access-group>
                </mac>
            </vlan>
        </interface>
    </interface-vlan>
  </config>
 </edit-config>
</rpc>

<rpc-reply message-id="2403" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Modifying MAC ACL rules

You cannot modify the existing rules of a MAC ACL. However, you can remove the rule and then recreate it with the desired changes.

Use a sequence number to specify the rule you wish to modify. Without a sequence number, a new rule is added to the end of the list, and existing rules are unchanged.

Using the **permit** and **deny** keywords, you can create many different rules. The examples in this section provide the basic knowledge needed to modify MAC ACLs.

To modify a MAC ACL, perform the following steps.

1. Issue the <edit-config> RPC to configure the <mac> node in the urn:brocade.com:mgmt:brocade-mac-access-list workspace.
2. Under the <mac> node, include the <access-list>/<extended> or <access-list>/<standard> hierarchy of node elements.
3. Under the <extended> or <standard> node, include the <name> element and specify the name of the ACL you want to modify.
4. Under the <extended> or <standard> node, include the <seq> node and include the delete operation in the element tag.
5. Under the <seq> node, include the <seq-id> leaf element, and specify the sequence ID of the rule you want to change.

   This action deletes the rule. The following example deletes rule 100. It assumes that test_02 contains an existing rule number 100 with the “deny any any” options.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2404" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac xmlns="urn:brocade.com:mgmt:brocade-mac-access-list">
                <access-list>
                    <extended>
                        <name>test_02</name>
                        <seq xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                             operation="delete">
                            <seq-id>100</seq-id>
```

```
                    </seq>
                </extended>
            </access-list>
        </mac>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2404" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

6. Issue another <edit-config> RPC to replace rule 100.

   Refer to "Creating an extended MAC ACL and adding rules" on page 381 for details.

   The following example creates a new rule 100.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2405" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac xmlns="urn:brocade.com:mgmt:brocade-mac-access-list">
                <access-list>
                    <extended>
                        <name>test_02</name>
                        <seq >
                            <seq-id>100</seq-id>
                            <action>permit</action>
                            <source>any</source>
                            <dst>any</dst>
                        </seq>
                    </extended>
                </access-list>
            </mac>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2405" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Creating a MAC ACL rule enabled for counter statistics

The following example shows how to enable counter statistics when you create ACL rules.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac xmlns="urn:brocade.com:mgmt:brocade-mac-access-list">
                <access-list>
                    <standard>
```

```
                      <name>mac_acl_1</name>
                      <hide-mac-acl-std>
                          <seq>
                              <seq-id>100</seq-id>
                              <action>deny</action>
                              <source>0022.3333.4444</source>

<src-mac-addr-mask>0011.2222.3333</src-mac-addr-mask>
                              <count></count>
                          </seq>
                      </hide-mac-acl-std>
                  </standard>
              </access-list>
          </mac>
      </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Removing a MAC ACL

A MAC ACL cannot be removed from the system unless the access-group applying the MAC ACL to a DCB or a VLAN interface is first removed.

To remove a MAC ACL.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2406" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mac xmlns="urn:brocade.com:mgmt:brocade-mac-access-list">
                <access-list>
                    <extended xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete">
                        <name>test_02</name>
                    </extended>
                </access-list>
            </mac>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2406" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Obtaining the MAC ACL applied to an interface

You can query the MAC ACL applied to an interface using the <get-mac-acl-for-intf> custom RPC. By omitting all input parameters, you can obtain the results for all interfaces, but only in the ingress direction. If you specify an interface, you can request results for the ingress direction, the egress direction, or both the ingress and egress directions.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc 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-type>l2vlan</interface-type>
        <interface-name>50</interface-name>
        <direction>all</direction>
    </get-mac-acl-for-intf>
</rpc>

<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>
```

IP ACL

The IP ACLs control access to the switch. The policies do not control the egress and outbound management traffic initiated from the switch. The IP ACLs support both IPv4 and IPv6 simultaneously.

An IP ACL is a set of rules that are applied to the interface as a packet filtering firewall. Each rule defines whether traffic of a combination of source and destination IP address, protocol, or port, is to be denied or permitted.

Each ACL must have a unique name, but there is no limit to the number of ACLs to be defined. An ACL can contain rules for only one version of IP (either IPv4 or IPv6). Only one ACL by the version of IP can be active on the interface at a time. In other words, one ACL for IPv4 addresses and one ACL for IPv6 address on the interface for packet filtering can be active at the same time.

For filtering the traffic, each rule of the ACL applied to the interface is checked in the ascending order of their sequence numbers. A maximum of 2048 rules can be added to an access list. When the ACL is applied to an interface, only the 256 lowest-numbered rules are applied. If an ACL does not contain any rules and is applied to the interface, it becomes “no-op” and all ingress traffic is denied through the interface. For Layer 2 ACL, if there are no rules applied to the interface then the action is permitted through that interface. But in Layer 3 ACL or IP ACL, it is denied.

After an IP ACL rule is created, it is not possible to modify any of its options.

The default configuration of the switch consists of two ACLs: one IPv4 ACL and one IPv6 ACL is applied to the interface.

There are two types of IP access lists:

- Standard: Contains rules for only the source IP address. The rules are applicable to all ports of that source IP address.
- Extended: Contains rules for a combination of IP protocol, source IP address, destination IP address, source port, and destination port.

Creating a standard IP or IPv6 ACL

The following example creates a standard IP ACL.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2408" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ip-acl xmlns="urn:brocade.com:mgmt:brocade-ip-access-list">
                <ip>
                    <access-list>
                        <standard>
                            <name>stdACL3</name>
                            <seq>
                                <seq-id>5</seq-id>
                                <action>permit</action>
                                <src-host-any-sip>host</src-host-any-sip>
                                <src-host-ip>10.20.33.4</src-host-ip>
                            </seq>
                            <seq>
                                <seq-id>15</seq-id>
                                <action>deny</action>
                                <src-host-any-sip>any</src-host-any-sip>
                            </seq>
                            <seq>
                                <seq-id>30</seq-id>
                                <action>hard-drop</action>
                                <src-host-any-sip>host</src-host-any-sip>
                                <src-host-ip>10.2.1.1</src-host-ip>
                                <log></log>
                            </seq>
                        </standard>
                    </access-list>
                </ip>
            </ip-acl>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2408" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Creating an extended IP or IPv6 ACL

To create an extended IP ACL

The following example creates an extended IP ACL named extdACL5 that includes the following rules:

- Rule 5 denies TCP traffic from host 10.24.26.145 or bound for port 23 on any destination host.
- Rule 7 denies TCP traffic from any source host on port 80 of any destination port.
- Rule 10 denies UDP traffic from any source host to ports in the range 10 through 25 on any destination host.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2409" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ip-acl xmlns="urn:brocade.com:mgmt:brocade-ip-access-list">
                <ip>
                    <access-list>
                        <extended>
                            <name>extdACL5</name>
                            <seq>
                                <seq-id>5</seq-id>
                                <action>deny</action>
                                <protocol-type>tcp</protocol-type>
                                <src-host-any-sip>host</src-host-any-sip>
                                <src-host-ip>10.24.26.145</src-host-ip>
                                <dst-host-any-dip>any</dst-host-any-dip>
                                <dport>eq</dport>
                                <dport-number-eq-neq-tcp>23
                                 </dport-number-eq-neq-tcp>
                            </seq>
                            <seq>
                                <seq-id>7</seq-id>
                                <action>deny</action>
                                <protocol-type>tcp</protocol-type>
                                <src-host-any-sip>any</src-host-any-sip>
                                <dst-host-any-dip>any</dst-host-any-dip>
                                <dport>eq</dport>
                                <dport-number-eq-neq-tcp>80
                                 </dport-number-eq-neq-tcp>
                            </seq>
                            <seq>
                                <seq-id>10</seq-id>
                                <action>deny</action>
                                <protocol-type>udp</protocol-type>
                                <src-host-any-sip>any</src-host-any-sip>
                                <dst-host-any-dip>any</dst-host-any-dip>
                                <dport>range</dport>
                                <dport-number-range-lower-upd>10
</dport-number-range-lower-udp>
                                <dport-number-range-higher-udp>25
<dport-number-range-higher-udp>
                            </seq>
                            <seq>
                                <seq-id>30</seq-id>
```

```
                        <action>hard-drop</action>
                        <src-host-any-sip>any</src-host-any-sip>
                    </seq>
                </extended>
            </access-list>
        </ip>
      </ip-acl>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2409" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Applying an IP or IPv6 ACL to a management interface

NOTE
Applying a permit or deny UDP ACL to the management interface enacts an implicit deny for TCP and vice versa.

To apply the IP ACLs to a management interface

The following example applies stdV6ACL1 to the management interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2410" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <management>
          <name>3/1</name>
          <ipv6>
            <access-group
             xmlns="urn:brocade.com:mgmt:brocade-ipv6-access-list">
               <mgmt-ipv6-access-list>stdV6ACL1</mgmt-ipv6-access-list>
            </access-group>
          </ipv6>
        </management>
      </interface>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2410" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Applying an IP ACL to a data interface

To apply the IP ACLs to a data interface

The following example applies stdV6ACL1 to the 101/0/1 interface.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<rpc message-id="2410" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>101/0/1</name>
                    <ip-acl-interface>
                        <access-group
                         xmlns="urn:brocade.com:mgmt:brocade-ipv6-access-list">
                            <ipv6-access-list>stdV6ACL1</ipv6-access-list>
                        </access-group>
                    </ip-acl-interface>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2410" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Applying an IP or IPv6 ACL to an overlay gateway

The following example applies ipv6 ACL to the overlay-gateway.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>gateway1</name>
                <gw-type>nsx</gw-type>
                <ipv6>
                    <access-group
xmlns="urn:brocade.com:mgmt:brocade-ipv6-access-list">
                        <mac-access-list>stdipv6aclin</mac-access-list>
                    </access-group>
                    </ipv6>
            </overlay-gateway>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2410" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Binding an ACL in fabric cluster mode

In fabric cluster mode, an ACL can be applied to any node present in the cluster by specifying its RBridge ID. One ACL per IPv4 and one ACL per IPv6 can be applied to the management interface. Applying a new ACL replaces the ACL that was previously applied. Removing the active ACL results in default behavior of “permit any.”

You can bind an IP ACL in the ingress direction for the management interface, and you are not required to create an ACL before binding it to the management interface.

On a management interface, the default action of “permit any” is inserted at the end of an ACL that has been bound.

NOTE
Before downgrading firmware, you must unbind any ACLs on the management interface, or the downgrade will be blocked.

Obtaining the IP or IPv6 ACL configuration

To obtain the IP or IPv6 ACL configuration, issue the <get-config> RPC to retrieve the access list. Set up the filter to restrict the output to the part of the configuration you want to retrieve.

The following example returns the entire IP ACL configuration.

```
?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2411">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <ip-acl xmlns="urn:brocade.com:mgmt:brocade-ip-access-list>
                <ip>
                    <access-list/>
                </ip>
            </ip-acl>
        </filter>
    </get-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2411">
    <ip-acl xmlns="urn:brocade.com:mgmt:brocade-ip-access-list">
        <ip>
            <access-list>
                <name>stdACL3</name>
                <seq>
                    <seq-id>5</seq-id>
                    <action>deny</action>
                    <protocol-type>tcp</protocol-type>
                    <src-host-any-sip>host</src-host-any-sip>
                    <src-host-ip>10.24.26.145</src-host-ip>
                    <dst-host-any-dip>any</dst-host-any-dip>
                    <dport>eq</dport>
                    <dport-number-eq-neq-tcp>23
                     </dport-number-eq-neq-tcp>
                </seq>
                <seq>
                    <seq-id>7</seq-id>
```

```
                <action>deny</action>
(output truncated)
```

Chapter

Configuring Dynamic ARP Inspection (DAI) 37

In this chapter

Dynamic ARP inspection (DAI) overview

Dynamic ARP inspection (DAI) is a security feature that validates address resolution protocol (ARP) packets in a subnet, and discards packets with invalid IP/MAC address bindings.

Implementing ARP ACLs

Address-resolution protocol (ARP) access-lists (ACLs), applied to untrusted ports, permit ARP packets with specified IP/MAC address-bindings.

Creating an ARP-ACL

The following example shows how to create an ARP-ACL:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <arp xmlns="urn:brocade.com:mgmt:brocade-dai">
                <access-list>
                    <acl-name>acl1</acl-name>
                    <permit>
                        <permit-list>
                            <ip-type>host</ip-type>
                            <host-ip>1.1.1.1</host-ip>
                            <mac-type>host</mac-type>
                            <host-mac>0001.0001.0001</host-mac>
                        </permit-list>
                        <permit-list>
                            <ip-type>host</ip-type>
                            <host-ip>1.1.1.2</host-ip>
                            <mac-type>host</mac-type>
                            <host-mac>0001.0001.0002</host-mac>
                        </permit-list>
```

```
                </permit>
            </access-list>
        </arp>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Applying an ARP ACL to a VLAN

The following example shows how to apply an ARP ACL to a VLAN:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ip>
                            <arp xmlns="urn:brocade.com:mgmt:brocade-dai">
                                <inspection>
                                    <filter>
                                        <acl-name>acl1</acl-name>
                                    </filter>
                                </inspection>
                            </arp>
                        </ip>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Defining trusted and untrusted interfaces under DAI

The following example shows how to define trusted and untrusted interfaces under DAI:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
```

```
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <ip>
                        <arp-node-config xmlns="urn:brocade.com:mgmt:brocade-dai">
                            <arp>
                                <inspection>
                                    <trust></trust>
                                </inspection>
                            </arp>
                        </arp-node-config>
                    </ip>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Defining a port-channel interface as trusted

The following example shows how to define trusted and untrusted interfaces under DAI:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <port-channel>
                    <name>1</name>
                    <ip>
                        <arp-node-config xmlns="urn:brocade.com:mgmt:brocade-dai">
                            <arp>
                                <inspection>
                                    <trust></trust>
                                </inspection>
                            </arp>
                        </arp-node-config>
                    </ip>
                </port-channel>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling dynamic ARP inspection (DAI)

The following example shows how to enable dynamic ARP inspection (DAI).

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ip>
                            <arp xmlns="urn:brocade.com:mgmt:brocade-dai">
                                <inspection>
                                    <trust></trust>
                                </inspection>
                            </arp>
                        </ip>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Router Advertisement (RA) Guard 38

In this chapter

Router Advertisement (RA) Guard

This chapter provides procedures to enable or disable RA Guards. Refer to the *Network OS Security Guide* for the following related information:

- RA Guard overview
- RA Guard configuration guidelines

Enabling RA Guard

The following examples shows how to enable RA guard on an interface.

```
<rpc message-id="2609" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/3</name>
                    <ipv6>
                        <raguard>true</raguard>
                    </ipv6>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2609" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Configuring QoS 39

In this chapter

QoS configuration under NETCONF overview

This chapter provides procedures for configuring QoS and Auto QOS using the NETCONF interface. Refer to the *Network OS Layer 2 Switching Configuration Guide* for an explanation of related concepts and presentation of default values.

QoS

Quality of Service (QoS) provides you with the capability to control how the traffic is moved from switch to switch. In a network that has different types of traffic with different needs (also known as CoS), the goal of QoS is to provide each traffic type with a virtual pipe. FCoE uses traffic class mapping, scheduling, and flow control to provide quality of service.

The "Queueing", "Congestion control", "</rpc-reply>", "</rpc-reply>", and "Data Center Bridging map configuration" sections of this chapter provide procedures for configuring and managing QoS.

Rewriting

Rewriting a frame header field is typically performed by an edge device. Rewriting occurs on frames as they enter or exit a network because the neighboring device is untrusted, unable to mark the frame, or is using a different QoS mapping.

The frame rewriting rules set the Ethernet CoS and VLAN ID fields. Egress Ethernet CoS rewriting is based on the user-priority mapping derived for each frame as described later in the queueing section.

Queueing

Queue selection begins by mapping an incoming frame to a configured user priority, and then each user-priority mapping is assigned to one of the switch's eight unicast traffic class queues or one of the eight multicast traffic class queues.

User-priority mapping

For a description of user priority mapping default values, refer to the *Network OS Layer 2 Switching Configuration Guide*.

Configuring user-priority mappings

To configure user-priority mappings

The following example sets the user priority mapping to 3 on 10-gigabit interface 22/0/2.

```
<rpc message-id="2303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/2</name>
                    <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                        <default-cos>3</default-cos>
                    </qos>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Creating a CoS-to-CoS mutation QoS map

To create a CoS-to-CoS mutation QoS map.

The following example creates a CoS-to-CoS mutation QoS map named "test."

```
<rpc message-id="2304" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
```

```
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                <map>
                    <cos-mutation>
                        <name>test</name>
                        <cos0>0</cos0>
                        <cos1>1</cos1>
                        <cos2>2</cos2>
                        <cos3>3</cos3>
                        <cos4>4</cos4>
                        <cos5>5</cos5>
                        <cos6>6</cos6>
                        <cos7>7</cos7>
                    </cos-mutation>
                </map>
            </qos>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2304" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Applying a CoS-to-CoS mutation QoS map to an interface

To apply a CoS-to-CoS mutation QoS map

The following example activates a map named “test” on 10-gigabit interface 22/0/2, and establishes trust mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2305" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/2</name>
                    <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                        <cos-mutation>test</cos-mutation>
                    </qos>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2305" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Verifying CoS-to-CoS mutation QoS mapping

To verify a CoS-to-CoS mutation mapping, issue the <get-config> RPC to retrieve the CoS-to-CoS mutation QoS map and the interface names to which a map is bound.

1. Verify a CoS-to-CoS mutation QoS map using a subtree filter to view only the contents of the <qos>/<map>/<cos-mutation> node. To limit the returned information to a specific QoS map, refine the content match node with the QoS map name.

   The following example returns the CoS-to-CoS mutation QoS map for a map named “test.”

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2306" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                <map>
                    <cos-mutation>
                        <name>test</name>
                    </cos-mutation>
                </map>
            </qos>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="2306" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
        <map>
            <cos-mutation>
                <name>test</name>
                <cos0>0</cos0>
                <cos1>1</cos1>
                <cos2>2</cos2>
                <cos3>3</cos3>
                <cos4>4</cos4>
                <cos5>5</cos5>
                <cos6>6</cos6>
                <cos7>7</cos7>
            </cos-mutation>
        </map>
    </qos>
</rpc-reply>
```

2. Return a list of interfaces that are bound to a CoS-to-CoS mutation QoS map using an xpath filter.

   You must use an xpath filter and not a subtree filter in this case, because 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).

   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 0/59 and 0/60.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2307">
    <get-config>
        <source>
            <running></running>
        </source>
        <filter type="xpath"
            select="/interface/tengigabitethernet/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="2607">
    <data>
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
            <tengigabitethernet>
                <name>0/59</name>
                <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                    <default-cos>0</default-cos>
                    <cos-mutation>test</cos-mutation>
                </qos>
            </tengigabitethernet>
            <tengigabitethernet>
                <name>0/60</name>
                <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                    <default-cos>0</default-cos>
                    <cos-mutation>test</cos-mutation>
                </qos>
            </tengigabitethernet>
        </interface>
    </data>
</rpc-reply>
```

Creating a DSCP mutation map

NOTE
This feature is only supported on Brocade VDX 8770-4, VDX 8770-8, and later models.

To create a DSCP mutation and remap the incoming DSCP value of the ingress packet to other DSCP values.

The following example creates a DSCP mutation map named "test," which performs the following mapping of DSCP values:

- DSCP values 1, 3, 5, and 7 are set to output as DSCP number 9.
- DSCP values 11, 13, 15, and 17 are set to output as DSCP number 19.
- DSCP values 12, 14, 16, and 18 are set to output as DSCP number 20
- DSCP values 2, 4, 6, and 8 are set to output as DSCP number 10.

```
<rpc message-id="2310" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                <map>
                    <dscp-mutation>
                        <dscp-mutation-map-name>test</dscp-mutation-map-name>
                        <mark>
                            <dscp-in-values>1,3,5,7</dscp-in-values>
                            <to>9</to>
                        </mark>
                        <mark>
                            <dscp-in-values>11,13,15,17</dscp-in-values>
```

```
                        <to>19</to>
                    </mark>
                    <mark>
                        <dscp-in-values>12,14,16,18</dscp-in-values>
                        <to>20</to>
                    </mark>
                    <mark>
                        <dscp-in-values>2,4,6,8</dscp-in-values>
                        <to>10</to>
                    </mark>
                </dscp-mutation>
            </map>
        </qos>
    </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2310" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Applying a DSCP mutation map to an interface

To apply a DSCP mutation QoS map

The following example activates a map named “test” on 10-gigabit interface 22/0/2, and establishes DSCP trust mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2311" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/2</name>
                    <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                        <dscp-mutation>test</dscp-mutation>
                    </qos>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2311" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Verifying DSCP mutation mapping

To verify a DSCP mutation mapping, issue the <get-config> RPC to retrieve the DSCP mutation QoS map and the interface names to which a map is bound.

1. Verify a DSCP mutation QoS map using a subtree filter to view only the contents of the <qos>/<map>/<dscp-mutation> node. To limit the returned information to a specific QoS map, refine the content match node with the QoS map name.

   The following example returns the DSCP mutation QoS map for a map named "test."

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2312" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                <map>
                    <dscp-mutation>
                        <dscp-mutation-map-name>test</dscp-mutation-map-name>
                    </dscp-mutation>
                </map>
            </qos>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="2312" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
        <map>
            <dscp-mutation>
                <name>test</name>
                <dscp-mutation-map-name>test</dscp-mutation-map-name>
                <mark>
                    <dscp-in-values>1,3,5,7</dscp-in-values>
                    <to>9</to>
                </mark>
                <mark>
                    <dscp-in-values>11,13,15,17</dscp-in-values>
                    <to>19</to>
                </mark>
                <mark>
                    <dscp-in-values>12,14,16,18</dscp-in-values>
                    <to>20</to>
                </mark>
                <mark>
                    <dscp-in-values>2,4,6,8</dscp-in-values>
                    <to>10</to>
                </mark>
            </dscp-mutation>
        </map>
    </qos>
</rpc-reply>
```

2. Return a list of interfaces that are bound to a DSCP mutation QoS map using an xpath filter.

   You must use an xpath filter and not a subtree filter in this case because the element to be used for the selection criteria (<dscp-mutation>*name*</dscp-mutation>) resides at a lower level in the hierarchy than the information to be retrieved (the interface name).

   The following example returns the interface names to which the DSCP mutation QoS map named "test" is bound. In this case, the map named "test" is bound to interfaces 0/59 and 0/60.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2313">
    <get-config>
        <source>
            <running></running>
        </source>
        <filter type="xpath"
            select="/interface/tengigabitethernet/qos[dscp-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="2613">
   <data>
      <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
         <tengigabitethernet>
            <name>0/59</name>
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
               <default-cos>0</default-cos>
               <dscp-mutation>test</dscp-mutation>
            </qos>
         </tengigabitethernet>
         <tengigabitethernet>
            <name>0/60</name>
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
               <default-cos>0</default-cos>
               <dscp-mutation>test</dscp-mutation>
            </qos>
         </tengigabitethernet>
      </interface>
   </data>
</rpc-reply>
```

Creating a DSCP-to-CoS mutation map

You can use the incoming DSCP value of ingress packets to remap the outgoing 802.1P CoS priority values by configuring a DSCP-to-CoS mutation map on the ingress interface.

The following example creates a DSCP -to-CoS map named “test,” which performs the following mapping of DSCP values to CoS priorities:

- DSCP values 1, 3, 5, and 7 are set to output as CoS priority 3.
- DSCP values 11, 13, 15, and 17 are set to output as CoS priority 5.
- DSCP values 12, 14, 16, and 18 are set to output as CoS priority 6.
- DSCP values 2, 4, 6, and 8 are set to output as CoS priority 7.

```
<rpc message-id="2314" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                <map>
                    <dscp-cos>
                        <dscp-cos-map-name>test</dscp-cos-map-name>
                        <mark>
```

```
                <dscp-in-values>1,3,5,7</dscp-in-values>
                <to>3</to>
            </mark>
            <mark>
                <dscp-in-values>11,13,15,17</dscp-in-values>
                <to>5</to>
            </mark>
            <mark>
                <dscp-in-values>12,14,16,18</dscp-in-values>
                <to>6</to>
            </mark>
            <mark>
                <dscp-in-values>2,4,6,8</dscp-in-values>
                <to>7</to>
            </mark>
          </dscp-cos>
        </map>
      </qos>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2314" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Applying a DSCP-to-CoS map to an interface

To apply a DSCP-to-CoS map to an interface.

The following example activates a map named “test” on 10-gigabit interface 22/0/2, and establishes DSCP trust mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2315" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <tengigabitethernet>
          <name>22/0/2</name>
          <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
            <dscp-cos>test</dscp-cos>
          </qos>
        </tengigabitethernet>
      </interface>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2315" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Verifying a DSCP-to-CoS mutation map

To verify a DSCP-to-CoS mapping, issue the <get-config> RPC to retrieve the DSCP-to-CoS map and the interface names to which a map is bound.

1. Verify a DSCP-to-CoS map using a subtree filter to view only the contents of the <qos>/<map>/<dscp-cos> node. To limit the returned information to a specific QoS map, refine the content match node with the QoS map name.

   The following example returns the DSCP mutation QoS map for a map named “test.”

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2316" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                <map>
                    <dscp-cos>
                        <dscp-cos-map-name>test</dscp-cos-map-name>
                    </dscp-cos>
                </map>
            </qos>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="2316" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
        <map>
            <dscp-cos>
                <dscp-cos-map-name>test</dscp-cos-map-name>
                <mark>
                    <dscp-in-values>1,3,5,7</dscp-in-values>
                    <to>3</to>
                </mark>
                <mark>
                    <dscp-in-values>11,13,15,17</dscp-in-values>
                    <to>5</to>
                </mark>
                <mark>
                    <dscp-in-values>12,14,16,18</dscp-in-values>
                    <to>6</to>
                </mark>
                <mark>
                    <dscp-in-values>2,4,6,8</dscp-in-values>
                    <to>7</to>
                </mark>
            </dscp-cos>
        </map>
    </qos>
</rpc-reply>
```

2. Return a list of interfaces that are bound to a DSCP-to-CoS map using an xpath filter.

   You must use an xpath filter and not a subtree filter in this case, because the element to be used for the selection criteria (<dscp-cos>*name*</dscp-cos>) resides at a lower level in the hierarchy than the information to be retrieved (the interface name).

The following example returns the interface names to which the DSCP-to-CoS map named "test" is bound. In this case, the map named "test" is bound to interfaces 0/59 and 0/60.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2317">
    <get-config>
        <source>
            <running></running>
        </source>
        <filter type="xpath"
            select="/interface/tengigabitethernet/qos[dscp-cos='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="2317">
   <data>
      <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
         <tengigabitethernet>
            <name>0/59</name>
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
               <default-cos>0</default-cos>
               <dscp-cos>test</dscp-cos>
            </qos>
         </tengigabitethernet>
         <tengigabitethernet>
            <name>0/60</name>
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
               <default-cos>0</default-cos>
               <dscp-cos>test</dscp-cos>
            </qos>
         </tengigabitethernet>
      </interface>
   </data>
</rpc-reply>
```

Traffic class mapping

The Brocade switch supports eight unicast traffic classes for isolation and to control servicing for different priorities of application data. Traffic classes are numbered from 0 through 7, with higher values designating higher priority.

Refer to the *Network OS Layer 2 Switching Configuration Guide* for an explanation of default user-priority to traffic class mappings for unicast traffic and for multicast traffic.

Mapping DSCP-to-Traffic-Class

Ingress DSCP values can be used to classify traffic for the ingress interface into a specific traffic class using a DSCP-to-Traffic Class map. To map a DSCP-to-Traffic-Class.

The following example creates a DSCP-to-Traffic-Class map named "test," which performs the following mapping of DSCP values to traffic classes:

- DSCP values 1, 3, 5, and 7 are mapped to traffic class 3.
- DSCP values 11, 13, 15, and 17 are mapped to traffic class 5.
- DSCP values 12, 14, 16, and 18 are mapped to traffic class 6.

- DSCP values 2, 4, 6, and 8 are mapped to traffic class 7.

```
<rpc message-id="2322" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                <map>
                    <dscp-traffic-class>
                        <dscp-traffic-class-map-name>test
                         </dscp-traffic-class-map-name>
                        <mark>
                            <dscp-in-values>1,3,5,7</dscp-in-values>
                            <to>3</to>
                        </mark>
                        <mark>
                            <dscp-in-values>11,13,15,17</dscp-in-values>
                            <to>5</to>
                        </mark>
                        <mark>
                            <dscp-in-values>12,14,16,18</dscp-in-values>
                            <to>6</to>
                        </mark>
                        <mark>
                            <dscp-in-values>2,4,6,8</dscp-in-values>
                            <to>7</to>
                        </mark>
                    </dscp-traffic-class>
                </map>
            </qos>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2322" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
    </rpc-reply>
```

Applying DSCP-to-Traffic-Class mapping to an interface

To apply a CoS-to-Traffic Class mapping to an interface.

The following example activates a CoS-to-Traffic-Class mapping on 10-gigabit Ethernet interface 22/0/2.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2323" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/2</name>
                    <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                        <dscp-traffic-class>test</dscp-traffic-class>
                    </qos>
```

```
            </tengigabitethernet>
        </interface>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2323" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
  </rpc-reply>
```

Verifying DSCP-to-Traffic-Class mapping

To verify a DSCP-to-Traffic-Class mapping, you can use one or both of the following options from global configuration mode.

1. Verify a DSCP-traffic-class map using a subtree filter to view only the contents of the <qos>/<map>/<dscp-traffic-class> node. To limit the returned information to a specific QoS map, refine the content match node with the QoS map name.

   The following example returns the DSCP-to-traffic class map named "test."

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2324" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                <map>
                    <dscp-traffic-class>
                        <dscp-traffic-class-map-name>test
                         </dscp-traffic-class-map-name>
                    </dscp-traffic-class>
                </map>
            </qos>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="2324" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
        <map>
            <dscp-traffic-class>
                <dscp-traffic-class-map-name>test</dscp-traffic-class-map-name>
                <mark>
                    <dscp-in-values>1,3,5,7</dscp-in-values>
                    <to>3</to>
                </mark>
                <mark>
                    <dscp-in-values>11,13,15,17</dscp-in-values>
                    <to>5</to>
                </mark>
                <mark>
                    <dscp-in-values>12,14,16,18</dscp-in-values>
                    <to>6</to>
                </mark>
                <mark>
                    <dscp-in-values>2,4,6,8</dscp-in-values>
                    <to>7</to>
```

```
                </mark>
            </dscp-traffic-class>
        </map>
    </qos>
</rpc-reply>
```

2. Return a list of interfaces that are bound to a DSCP-to-traffic class map using an xpath filter.

   You must use an xpath filter and not a subtree filter in this case, because the element to be used for the selection criteria (<dscp-traffic-clsss>*name*</dscp-traffic-class>) resides at a lower level in the hierarchy than the information to be retrieved (the interface name).

   The following example returns the interface names to which the DSCP-to-CoS map named "test" is bound. In this case, the map named "test" is bound to interfaces 0/59 and 0/60.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="2325">
    <get-config>
        <source>
            <running></running>
        </source>
        <filter type="xpath"
        select="/interface/tengigabitethernet/qos[dscp-traffic-class='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="2325">
   <data>
      <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
         <tengigabitethernet>
            <name>0/59</name>
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
               <default-cos>0</default-cos>
               <dscp-traffic-class>test</dscp-traffic-class>
            </qos>
         </tengigabitethernet>
         <tengigabitethernet>
            <name>0/60</name>
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
               <default-cos>0</default-cos>
               <dscp-traffic-class>test</dscp-traffic-class>
            </qos>
         </tengigabitethernet>
      </interface>
   </data>
</rpc-reply>
```

Configuring Maximum Burst Size

This section provides procedure to configure Ingress Queue Depth and Egress Queue Depth. This is a Rbridge level configuration.

Configuring ingress queue depth

This section provides procedures for configuring ingress queue depth.

```
<?xml version="1.0" ?>
```

```
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                    <rcv-queue>
                        <rcv-queue-limit>1000</rcv-queue-limit>
                    </rcv-queue>
                </qos>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>

</rpc-reply>
```

Configuring egress queue depth

This section provides procedures for configuring egress queue depth.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                    <tx-queue>
                        <tx-queue-limit>2000</tx-queue-limit>
                    </tx-queue>
                </qos>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Congestion control

For conceptual information about the various congestion control methods supported in Network OS, including IEEE 802.3x Ethernet Pause, Tail Drop, and Ethernet Priority Flow Control (PFC), refer to the *Network OS Layer 2 Switching Configuration Guide*.

Random Early Detection

Procedures for configuring and applying Random Early Detection (RED) profiles follow. For conceptual information about RED profiles and for operational considerations, refer to the *Network OS Layer 2 Switching Configuration Guide*.

Configuring RED profiles

To configure an egress RED profile.

```
<rpc message-id="2328" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                <red-profile>
                    <profile-id>10
                    <min-threshold>10</min-threshold>
                    <max-threshold>80</max-threshold>
                    <drop-probability>80</drop-probability>
                </red-profile>
            </qos>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2328" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling a RED profile to use traffic-class

To map a traffic-class value on a per-port basis to the RED profile created under "Configuring RED profiles" on page 416,

The following example applies traffic-class 3 on port 22/0/2 to RED profile 10.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2329" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/2</name>
                    <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                        <random-detect>
                            <traffic-class>
                                <red-tc-value>3</red-tc-value>
                                <red-profile-id>10</red-profile-id>
                            </traffic-class>
                        </random-detect>
                    </qos>
                </tengigabitethernet>
```

```
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2329" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Ethernet Pause

This section provides procedures for configuring Ethernet Pause congestion control.

Enabling Ethernet Pause

To enable Ethernet Pause

The following example enables Ethernet Pause for transmit and receive directions on 10-gigabit Ethernet interface 22/0/1.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2330" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                        <flowcontrol>
                            <link-level-flowcontrol>
                                <flowcontrol-tx>on</flowcontrol-tx>
                                <flowcontrol-rx>on</flowcontrol-rx>
                            </link-level-flowcontrol>
                        </flowcontrol>
                    </qos>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2330" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling drop monitor

The following example shows how to enable the polling on RX/Tail and RED drops (in VDX67**) on this interface.

```
<rpc message-id="2310" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
```

```
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <port-channel>
                    <name>122</name>
                    <qos xmlns="urn:brocade.com:mgmt:brocade-qos">
                        <drop-monitor>
                            <drop-monitor-enable></drop-monitor-enable>
                        </drop-monitor>
                    </qos>
                </port-channel>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2310" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Broadcast, unknown unicast, and multicast storm control

Broadcast, unknown unicast, and multicast (BUM) storm control can be configured for the following physical interface types:

- gigabitethernet
- tengigabitethernet
- fortygigabitethernet
- hundredgigabitethernet

For conceptual information about BUM storm control and operational considerations, refer to the *Network OS Layer 2 Switching Configuration Guide*.

Configuring BUM storm control

To configure storm control on a physical interface

The following example rate-limits the broadcast traffic type on interface 101/0/1 to 1,000,000 bps, and shuts the port down if the rate limit is violated.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2333" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>101/0/2</name>
                    <storm-control
                     xmlns="urn:brocade.com:mgmt:brocade-bum-storm-control">
                        <ingress>
                            <protocol>broadcast</protocol>
```

```
                        <rate-format>limit-bps</rate-format>
                        <rate-bps>1000000</rate-bps>
                        <action>shutdown</action>
                    </ingress>
                </storm-control>
            </tengigabitethernet>
        </interface>
    <config>
    </edit-config>
</rpc>

<rpc-reply message-id="2333" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Data Center Bridging map configuration

For an overview of Data Center Bridging map configuration and applicable default values, refer to the *Network OS Layer 2 Switching Configuration Guide*.

Creating a CEE map

To create a CEE map

The following example creates a CEE map named "default."

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2336" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <cee-map xmlns="urn:brocade.com:mgmt:brocade-cee-map">
                <name>default</name>
            </cee-map>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2336" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Defining a priority group table

To define a priority group table

The following example defines two priority groups, assigns 50% bandwidth to each, enables PFC for the first priority group, and disables PFC for the second priority group.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2337" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
```

```
        </target>
        <config>
            <cee-map xmlns="urn:brocade.com:mgmt:brocade-cee-map">
                <name>default</name>
                <priority-group-table>
                    <PGID>0</PGID>
                    <weight>50</weight>
                    <pfc>on</pfc>
                </priority-group-table>
                <priority-group-table>
                    <PGID>1</PGID>
                    <weight>50</weight>
                    <pfc>off</pfc>
                </priority-group-table>
            </cee-map>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2337" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Defining a priority-table map

To define a priority-table map

The following example maps CoS 0 through CoS 2 and CoS 4 through CoS 6 to PGID 1, CoS 3 to PGID 0, and CoS 7 to PGID 15.0.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2338" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <cee-map xmlns="urn:brocade.com:mgmt:brocade-cee-map">
                <name>default</name>
                <priority-table>
                    <map-cos0-pgid>1</map-cos0-pgid>
                    <map-cos1-pgid>1</map-cos1-pgid>
                    <map-cos2-pgid>1</map-cos2-pgid>
                    <map-cos3-pgid>0</map-cos3-pgid>
                    <map-cos4-pgid>1</map-cos4-pgid>
                    <map-cos5-pgid>1</map-cos5-pgid>
                    <map-cos6-pgid>1</map-cos6-pgid>
                    <map-cos7-pgid>15.0</map-cos7-pgid>
                </priority-table>
            </cee-map>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2338" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Applying a CEE provisioning map to an interface

To apply a CEE provisioning map to an interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2339" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/2</name>
                    <cee>default</cee>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2339" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Verifying the CEE maps

To verify the CEE map, issue the <get-config> RPC with a subtree filter to return the CEE default map information.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2340" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <cee-map xmlns="urn:brocade.com:mgmt:brocade-cee-map">
                <name>default</name>
            </cee-map>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="2341" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <cee-map xmlns="urn:brocade.com:mgmt:brocade-interface">
        <name>default</name>
        <priority-group-table>
            <PGID>0</PGID>
            <weight>50</weight>
            <pfc>on</pfc>
        </priority-group-table>
        <priority-group-table>
            <PGID>1</PGID>
            <weight>50</weight>
            <pfc>off</pfc>
        </priority-group-table>
        <priority-table>
            <map-cos0-pgid>1</map-cos0-pgid>
```

```
            <map-cos1-pgid>1</map-cos1-pgid>
            <map-cos2-pgid>1</map-cos2-pgid>
            <map-cos3-pgid>0</map-cos3-pgid>
            <map-cos4-pgid>1</map-cos4-pgid>
            <map-cos5-pgid>1</map-cos5-pgid>
            <map-cos6-pgid>1</map-cos6-pgid>
            <map-cos7-pgid>15.0</map-cos7-pgid>
        </priority-table>
        <remap>
            <fabric-priority>
                <fabric-remapped-priority>2</fabric-remapped-priority>
            </fabric-priority>
            <lossless-prioroty>
                <lossless-remapped-priority>2</lossless-remapped-priority>
            </lossless-priority>
        </remap>
    </cee-map>
</rpc-reply>
```

Brocade VCS Fabric QoS

Brocade VCS Fabric QoS requires very little user configuration. The only options to modify are the fabric priority and the lossless priority.

Brocade VCS Fabric reserves a mapping priority and fabric priority of seven (7). Any traffic that enters the Brocade VCS Fabric cluster from upstream that is using the reserved priority value is automatically remapped to a lower priority.

Changing the mapping or fabric priority is not required. By default the values are set to zero (0) for both of the re-mapped priorities.

In Brocade VCS Fabric mode:

- All incoming priority 7 tagged packets are dropped on the edge ports.
- Untagged control frames are counted in queue 7 (TC7).

All switches in the Brocade VCS Fabric cluster must have matching re-mapping priority values and the same priority-group-table values.

Configuring Brocade VCS Fabric QoS

To configure the remapping priorities for the Brocade VCS Fabric

The following example remaps the lossless priority and fabric priority values to 2.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <cee-map xmlns="urn:brocade.com:mgmt:brocade-cee-map">
                <name>default</name>
                <remap>
                    <lossless-priority>
                        <lossless-remapped-priority>2</lossless-remapped-priority>
```

```
                    </lossless-priority>
                    <fabric-priority>
                        <fabric-remapped-priority>2</fabric-remapped-priority>
                    </fabric-priority>
                </remap>
            </cee-map>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <name>22/0/1</name>
                <cee>default</cee>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Restrictions for Layer 3 features in VCS mode

For an overview of Layer 3 restrictions in VCS mode, refer to the *Network OS Layer 2 Switching Configuration Guide*.

Flow-based QoS

1. Configure a class-map to classify traffic according to the traffic properties required for your flowbased QoS needs.
2. Configure a policy-map to associate a policy-map with the class-map and also add the QoS action to be applied on the type of flow determined by the class-map.
3. Bind the policy-map to a specific interface using the service-policy command, or bind the policymap to an Rbridge ID.

Configuring a class-map

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="11">
   <edit-config>
      <target>
         <running></running>
      </target>
      <config>
         <class-map xmlns="urn:brocade.com:mgmt:brocade-policer">
            <name>new_class_map</name>
         </class-map>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="11" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring a policy-map

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
                <po-name>pmap1</po-name>
            </policy-map>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Associating the class-map to a policy-map

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
   <edit-config>
      <target>
         <running></running>
      </target>
      <config>
         <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
            <po-name>new_policy_map</po-name>
            <class>
               <cl-name>new_class_map</cl-name>
            </class>
         </policy-map>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring flow-based QoS actions using policy-map

- "Configuring QoS policier action"
- "Configuring QoS mutation map actions"
- "Configuring QoS shaping action"
- "Configuring the QoS scheduling action"
- "Configuring the sFlow profile action"
- "Configuring the CEE map action"

Configuring QoS policier action

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<rpc message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <police-priority-map xmlns="urn:brocade.com:mgmt:brocade-policer">
                <name>pmap1</name>
                <conform>
                    <map-pri0-conform>1</map-pri0-conform>
                    <map-pri1-conform>2</map-pri1-conform>
                    <map-pri2-conform>3</map-pri2-conform>
                    <map-pri3-conform>4</map-pri3-conform>
                    <map-pri4-conform>5</map-pri4-conform>
                    <map-pri5-conform>6</map-pri5-conform>
                    <map-pri6-conform>7</map-pri6-conform>
                    <map-pri7-conform>0</map-pri7-conform>
                </conform>
                <exceed>
                    <map-pri0-exceed>0</map-pri0-exceed>
                    <map-pri1-exceed>1</map-pri1-exceed>
                    <map-pri2-exceed>2</map-pri2-exceed>
                    <map-pri3-exceed>3</map-pri3-exceed>
                    <map-pri4-exceed>4</map-pri4-exceed>
                    <map-pri5-exceed>5</map-pri5-exceed>
                    <map-pri6-exceed>6</map-pri6-exceed>
                    <map-pri7-exceed>7</map-pri7-exceed>
                </exceed>
            </police-priority-map>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>

<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
                <po-name>plolicymap</po-name>
                <class>
                    <cl-name>clas1</cl-name>
                    <police>
                        <cir>40000</cir>
                        <eir>50000</eir>
                    </police>
                </class>
            </policy-map>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
```

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

Configuring QoS mutation map actions

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2345" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
                <po-name>pmap1</po-name>
                <class>
                    <cl-name>default</cl-name>
                    <map>
                        <cos-mutation>map1</cos-mutation>
                    </map>
                </class>
            </policy-map>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2345" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring QoS shaping action

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2345" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
                <po-name>pmap1</po-name>
                <class>
                    <cl-name>default</cl-name>
                    <port-shape>3000</port-shape>
                </class>
            </policy-map>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2345" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the QoS scheduling action

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2334" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
```

```
            <running/>
        </target>
        <config>
            <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
                <po-name>pmap1</po-name>
                <class>
                    <cl-name>default</cl-name>
                    <scheduler>
                        <strict-priority>
                            <priority-number>1</priority-number>
                            <scheduler-type>dwrr</scheduler-type>
                            <dwrr-traffic-class0>10</dwrr-traffic-class0>
                            <dwrr-traffic-class1>10</dwrr-traffic-class1>
                            <dwrr-traffic-class2>10</dwrr-traffic-class2>
                            <dwrr-traffic-class3>10</dwrr-traffic-class3>
                            <dwrr-traffic-class4>10</dwrr-traffic-class4>
                            <dwrr-traffic-class5>10</dwrr-traffic-class5>
                            <dwrr-traffic-class-last>40</dwrr-traffic-class-last>
                            <TC7>30000</TC7>
                        </strict-priority>
                    </scheduler>
                </class>
            </policy-map>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2334" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the sFlow profile action

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2345" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
                <po-name>pmap1</po-name>
                <class>
                    <cl-name>default</cl-name>
                    <map>
                        <sflow>map1</sflow>
                    </map>
                </class>
            </policy-map>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2345" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the CEE map action

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
                <po-name>p1</po-name>
                <class>
                    <cl-name>cee</cl-name>
                    <priority-mapping-table>
                        <import>
                            <cee>default</cee>
                        </import>
                    </priority-mapping-table>
                </class>
            </policy-map>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Binding the policy-map to an interface

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fortygigabitethernet>
                    <name>1/0/49</name>
                    <service-policy xmlns="urn:brocade.com:mgmt:brocade-policer">
                        <out>pmap1</out>
                    </service-policy>
                </fortygigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Binding flow-based QoS at the system level

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <system-qos xmlns="urn:brocade.com:mgmt:brocade-policer">
                <qos>
                    <service-policy>
                        <direction>in</direction>
                        <policy-map-name>pmap1</policy-map-name>
                        <attach>
                            <rbridge-id>
                                <add>
                                    <rb-add-range>1-2</rb-add-range>
                                </add>
                            </rbridge-id>
                        </attach>
                    </service-policy>
                </qos>
            </system-qos>
        </config>
    </edit-config>
</rpc>


<rpc-reply message-id="2342" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring Auto-QOS

Auto QoS (Quality of Service) for NAS creates a minimum bandwidth guarantee for Network Attached Storage traffic. Auto QoS for NAS is disabled by default; you must enable Auto QoS to allow NAS packets to have the correct service levels.

The **cee-map** priority group and priority-map settings must be their default values.

Enabling Auto QoS for NAS:

- Changes the CoS value of NAS packets to 2
- Reduces the weight of PGID 2 from 60 to 40
- Creates a new PGID 3 with a weight of 20
- Modifies the priority table to include PGID 3 for the user-configured NAS CoS, or the default NAS CoS if the CoS has not been otherwise modified

1. Issue the <edit-config> RPC to configure the <qos> node in the urn:brocade.com:mgmt:brocade-qos namespace.
2. Enable Auto QoS for all NAS traffic with the NAS <auto-qos> node under the <nas> container. The presence of the node activates Auto-QOS.

```
<?xml version="1.0">
<rpc-reply message-id="2351" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<edit-config>
    <target>
        <running/>
    </target>
```

```
    <config>
        <nas>
            <nas xmlns="urn:brocade.com:mgmt:brocade-qos">
            </auto-qos>
        </nas>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="2351" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

3. Set the CoS value for all NAS traffic by entering a value for the <cos> node.

```
<?xml version="1.0">
<rpc-reply message-id="2351" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<edit-config>
    <target>
        <running/>
    </target>
    <config>
        <nas xmlns="urn:brocade.com:mgmt:brocade-qos">
            <auto-qos>
                <set>
                    <cos>4</cos>
                </set>
            </auto-qos>
        </nas>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="2351" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

4. Set the DSCP value for all NAS traffic by entering a value for the <dscp> node.

   The Differentiated Services Code Point (DSCP) value affects how Auto-QoS operates by specifying the priority value for Network Attached Storage traffic on IP networks. Higher numbers provide a higher level of priority.

```
<?xml version="1.0">
<rpc-reply message-id="2351" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<edit-config>
    <target>
        <running/>
    </target>
    <config>
        <nas xmlns="urn:brocade.com:mgmt:brocade-qos">
            <auto-qos>
                <set>
                    <dscp>56</dscp>
                </set>
            </auto-qos>
        </nas>
    </config>
</edit-config>
</rpc>
```

```
<rpc-reply message-id="2351" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

5. Identify the IPv4/IPv6 network addresses (either origination or destination) used by the NAS devices by adding the <server-ip> node, followed by either the <vlan-number> node or the <vrf-name> node.

IPv4:

```
<?xml version="1.0">
<rpc-reply message-id="2351" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<edit-config>
    <target>
        <running/>
    </target>
    <config>
        <nas xmlns="urn:brocade.com:mgmt:brocade-qos">
            <server-ip>
                <server-ip>1.1.1.1/32</server-ip>
                    <vrf>
                        <vrf-name>bruce</vrf-name>
                    </vrf>
            </server-ip>
        </nas>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="2351" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

IPv6:

```
<?xml version="1.0">
<rpc-reply message-id="2351" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<edit-config>
    <target>
        <running/>
    </target>
    <config>
        <nas xmlns="urn:brocade.com:mgmt:brocade-qos">
            <server-ip>
                <server-ip>2000::1/16</server-ip>
                    <vrf>
                        <vrf-name>bruce</vrf-name>
                    </vrf>
            </server-ip>
        </nas>
    </config>
</edit-config>
</rpc>
<rpc-reply message-id="2351" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Chapter

Configuring 802.1x Port Authentication 40

In this chapter

802.1x port authentication with NETCONF overview

This chapter provides procedures for configuring 802.1x authentication using NETCONF interfaces. Refer to the *Network OS Layer 2 Switching Configuration Guide* for the following related information:

- Conceptual and overview information about the 802.1x port authentication and the 802.1x protocol
- Configuring 802.1x port authentication using the Network OS command line interface (CLI)

Through the NETCONF interface, you can perform the following operations related to 802.1x port authentication:

- Use the <edit-config> RPC to configure 802.1x port authentication globally and on a per-interface basis.
- Use the <get-config> RPC to verify all or part of the global or per-port 802.1x port authentication configuration.

802.1x port authentication parameters are defined in the brocade-dot1x YANG module. For information about the brocade-dot1x YANG module, refer to the *Network OS YANG Reference Manual*.

802.1x authentication configuration tasks

The tasks in this section describe the common 802.1x operations that you may need to perform. For complete configuration options using the NETCONF interface, refer to the *Network OS YANG Reference Manual* and the brocade-dot1x.yang source file.

Configuring authentication between the switch and CNA or NIC

To configure authentication, you must add a RADIUS server to perform the authentication, and then enable 802.1x authentication globally. The authentication process attempts to connect to the first RADIUS server. If the RADIUS server is not reachable, the next RADIUS server is contacted. However, if the RADIUS server is contacted and the authentication fails, the authentication process does not check for the next server in the sequence.

To add a RADIUS server and enable 802.1x authentication globally

The following example specifies 10.0.0.5 as a RADIUS server and enables 802.1x authentication globally.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2400" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <radius-server xmlns="urn:brocade.com:mgmt:brocade-aaa">
                <host>
                    <hostname>10.0.0.5</hostname>
                </host>
            </radius-server>
            <dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x>
                <enable/>
            </dot1x>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2400" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Setting a global timeout value for performing readiness checks

The 802.1x readiness check monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable.

Before running the readiness check, you can set a timeout value in seconds. The default timeout value is 10 seconds. To configure a readiness check timeout value

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x">
                <test>
                    <timeout>40</timeout>
```

```
            </test>
        </dot1x>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Disabling 802.1x globally

To disable 802.1x authentication globally,

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x">
        <enable xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
         operation="delete"/>
      </dot1x>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2402" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Configuring 802.1x on specific interface ports

To configure 802.1x port authentication on a specific interface port,

The following example configures 802.1x authentication on 10-gigabit Ethernet port 22/0/1.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2403" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <tengigabitethernet>
          <name>22/0/1</name>
          <dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x">
            <authentication/>
          </dot1x>
        </tengigabitethernet>
      </interface>
    </config>
  </edit-config>
</rpc>
```

```
<rpc-reply message-id="2403" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring 802.1x timeouts on specific interface ports

NOTE
While you are free to modify the timeouts, Brocade recommends that you leave timeouts set to their default values.

You can configure the following timeout values:

- <re-authperiod>—Configures the time in seconds between reauthentication attempts. The value must be in the range 1 through 4,294,967,295. The default is 3600.
- <server-timeout>—Configures a timeout interval in seconds for the 802.1x server. This period is the amount of time the switch waits for a reply before retransmitting the response to the server, when relaying the response from the client to the authentication server. The value must be in the range 1 through 65535. The default value is 30.
- <supp-timeout>—Configures a timeout interval in seconds for the 802.1x supplicant. This period is the amount of time the switch waits for a response before retransmitting the request to the client when relaying a request from the authentication server to client. The value must be in the range 1 through 65535. The default value is 30.
- <tx-period>—Configures the transmission timeout. This value specifies the number of seconds the switch waits for a response to an EAP request/identity from the client before retransmitting the request. The value must be in the range 1 through 65535. The default value is 30.

To configure 802.1x timeout attributes on a specific interface port,

The following example changes the supplicant timeout value to 40 seconds.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2404" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x">
                        <timeout>
                            <supp-timeout>40</supp-timeout>
                        </timeout>
                    </dot1x>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2404" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring 802.1x re-authentication on interface ports

To configure 802.1x port re-authentication on a specific interface port, perform the following steps. Repeat this task for each interface port you wish to modify.

The default re-authentication period is 3600 seconds.

The following example sets reauthorization for the 10 gigabit Ethernet 22/0/1 interface and sets the reauthorization timer to 4000 seconds.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2405" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x">
                        <authentication/>
                        <reauthentication/>
                        <timeout>
                            <re-authperiod>4000</re-authperiod>
                        </timeout>
                    </dot1x>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2405" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring 802.1x port-control on specific interface ports

You can set 802.1x port-control on an interface to one of the following states:

- auto—802.1x authentication is enabled. The port moves to the authorized state only after successful authentication. “auto” is the default value.
- force-authorized—802.1x authentication is disabled and the port moves to the authorized state.
- force-unauthorized—802.1x authentication is disabled and the port moves to the unauthorized state.

NOTE
If you globally disable 802.1x, all interface ports with 802.1x authentication enabled automatically switch to force-authorized port-control mode.

To configure 802.1x port-control on a specific interface port, perform the following steps. Repeat this task for each interface port you wish to modify.

The following example sets port-control on 10 gigabit Ethernet interface 22/0/1 to “force-authorized”.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2406" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x">
                        <authentication/>
                        <port-control>force-authorized</port-control>
                    </dot1x>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2406" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling 802.1x on specific interface ports

To disable 802.1x authentication on a specific interface port

The following example disables 802.1x authentication on interface 22/0/1.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2407" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x">
                        <authentication
                         xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete">
                    </dot1x>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2407" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Checking 802.1x configurations

You cannot obtain 802.1x authentication operational data using the NETCONF interface. To obtain information about dot1x statistical and diagnostic information requires the CLI. Refer to the *Network OS Layer 2 Switching Configuration Guide* for details.

To retrieve running configuration information for global 802.1x authentication, issue the <get-config> RPC with a subtree filter to return the <dot1x> portion of the configuration, as shown in the following example. The <enable/> node is returned in the reply if dot1x is configured globally. The readiness timeout value is also returned.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2408" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x"/>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="2408" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x">
        <enable/>
        <test>
            <timeout>20</timeout>
        </test>
    </dot1x>
</rpc-reply>
```

To obtain dot1x configuration information for a specific interface, issue the <get-config> RPC with a subtree filter to restrict the output to the <dot1x> node under a specific interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2409" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/1</name>
                    <dot1x xmlns="urn:brocade.com:mgmt:brocade-dot1x"/>
                </tengigabitethernet>
            </interface>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="2409" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <tengigabitethernet>
            <name>22/0/1</name>
            <dot1x>
                <authentication/>
                <port-control>auto</port-control>
```

```
                <protocol-version>2<protocol-version>
                <quiet-period>120</quiet-period>
                <reauthMax>5</reauthMax>
                <reauthentication/>
                <timeout>
                    <re-authperiod>25</re-authperiod>
                    <server-timeout>40</server-timeout>
                    <supp-timeout>40</supp-timeout>
                    <tx-period>34</tx-period>
                </timeout>
            </dot1x>
        </tengigabitethernet>
    </interface>
</rpc-reply>
```

Chapter

Configuring sFlow 41

In this chapter

sFlow configuration with NETCONF overview

This chapter provides procedures for configuring sFlow using the NETCONF interface. Refer to the *Network OS Layer 2 Switching Configuration Guide* for related conceptual and overview information about the sFlow protocol.

Through the NETCONF interface, you can perform the following operations that affect the functioning of sFlow:

- Use the <edit-config> RPC to activate, configure, or deactivate the sFlow protocol globally.
- Use the <edit-config> RPC to activate, configure, or deactivate sFlow on specific 10-Gigabit, 40-Gigabit, or Gigabit Ethernet interfaces.
- Use the <get-config> RPC to verify all or part of the global or per-port sFlow configuration.

sFlow must be enabled globally before it can be enabled on a specific interface.

sFlow parameters are defined in the brocade-sflow YANG module. For information about the brocade-sflow YANG module, refer to the *Network OS YANG Reference Manual*.

Configuring the sFlow protocol globally

To configure sFlow globally.

The following example enables sFlow globally, designates 102.10.128.176 as the sFlow collector, sets the polling interval to 35, and the sample rate to 4096.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <sflow xmlns="urn:brocade.com:mgmt:brocade-sflow">
                <enable/>
                <collector>
                    <collector-ip-address>1.1.1.1</collector-ip-address>
```

```
                <collector-port-number>50</collector-port-number>
                <use-vrf></use-vrf>
            </collector>
            <polling-interval>35</polling-interval>
            <sample-rate>4096</sample-rate>
        </sflow>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Interface-specific administrative tasks for sFlow

After the global sFlow configuration, sFlow must be explicitly enabled on all the required interface ports.

NOTE
When sFlow is enabled on an interface port, it inherits the sampling rate and polling interval from the global sFlow configuration.

Enabling and customizing sFlow on specific interfaces

NOTE
On the Brocade VDX 8770, SPAN and sFlow can be enabled at the same time. However, on the SPAN and sFlow cannot be enabled at the same time.

To enable and customize sFlow on an interface.

The following example enables sFlow on 10-gigabit Ethernet port 1/0/16 and sets port-specific values for the polling interval and sample rate.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2501" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/16</name>
                    <sflow xmlns="urn:brocade.com:mgmt:brocade-sflow">
                        <enable/>
                        <polling-interval>35</polling-interval>
                        <sample-rate>8192</sample-rate>
                    </sflow>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="2501" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
       <ok/>
</rpc-reply>
```

Disabling sFlow on specific interfaces

NOTE
Disabling sFlow on the interface port does not completely shut down the network communication on the interface port.

To disable sFlow on a specific interface.

The following example disables sFlow on 10-gigabit Ethernet port 1/0/16.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2502" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/16</name>
                    <sflow xmlns="urn:brocade.com:mgmt:brocade-sflow">
                        <enable xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                        operation="delete"/>
                    </sflow>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2502" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
       <ok/>
</rpc-reply>
```

Flow-based sFlow

Refer to the *Network OS Layer 2 Switching Configuration Guide* for related conceptual and overview information about flow-based sFlow.

Configuring flow-based sFlow

Flow-based sFlow is used to analyze a specific type of traffic (flow based on access control lists, or ACLs). This involves configuring an sFlow policy map and binding it to an interface.

NOTE
The "deny ACL" rule is not supported for flow-based sflow. Only the permit action is supported.

Perform the following steps, beginning in global configuration mode.

1. Create an sFlow profile.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="9">
   <edit-config>
      <target>
         <running></running>
      </target>
      <config>
         <sflow-profile xmlns="urn:brocade.com:mgmt:brocade-sflow">
            <profile-name>new_sflow_profile</profile-name>
            <profile-sampling-rate>512</profile-sampling-rate>
         </sflow-profile>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="9" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

2. Create a standard MAC ACL.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
   <edit-config>
      <target>
         <running></running>
      </target>
      <config>
         <mac xmlns="urn:brocade.com:mgmt:brocade-mac-access-list">
            <access-list>
               <standard>
                  <name>new_acl</name>
               </standard>
            </access-list>
         </mac>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

3. Create a class map and attach the ACL to the class map.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="11">
   <edit-config>
      <target>
         <running></running>
      </target>
      <config>
         <class-map xmlns="urn:brocade.com:mgmt:brocade-policer">
            <name>new_class_map</name>
         </class-map>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="11" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="12">
   <edit-config>
      <target>
         <running></running>
      </target>
      <config>
         <class-map xmlns="urn:brocade.com:mgmt:brocade-policer">
            <name>new_class_map</name>
            <match>
               <access-group>
                  <access-group-name>new_acl</access-group-name>
               </access-group>
            </match>
         </class-map>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="12" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

4. Create a policy map and attach the class map to the policy map.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="13">
   <edit-config>
      <target>
        <running></running>
      </target>
      <config>
         <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
            <po-name>new_policy_map</po-name>
         </policy-map>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="13" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>

<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
   <edit-config>
      <target>
         <running></running>
      </target>
      <config>
         <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
            <po-name>new_policy_map</po-name>
            <class>
               <cl-name>new_class_map</cl-name>
            </class>
         </policy-map>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
```

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

5. Add an sFlow profile name by using the **map** command.

   This example assigns the profile name “new_sflow_profile.”

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="15">
   <edit-config>
      <target>
         <running></running>
      </target>
      <config>
         <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
            <po-name>new_policy_map</po-name>
            <class>
               <cl-name>new_class_map</cl-name>
               <map>
                  <sflow>new_sflow_profile</sflow>
               </map>
            </class>
         </policy-map>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="15" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

6. Bind the policy map to an interface.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="16">
   <edit-config>
      <target>
         <running></running>
      </target>
      <config>
         <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
            <tengigabitethernet>
               <name>1/0/1</name>
               <service-policy xmlns="urn:brocade.com:mgmt:brocade-policer">
                  <in>new_policy-map</in>
               </service-policy>
            </tengigabitethernet>
         </interface>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="16" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

Disabling flow-based sFlow on specific interfaces

NOTE
Disabling sFlow on the interface port does not completely shut down the network communication on the interface port.

To disable sFlow on a specific interface, perform the following steps in interface configuration mode.

```
switch# show sflow interface tengigabitethernet 0/12
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="13">
   <edit-config>
      <target>
         <running></running>
      </target>
        <config>
              <sflow xmlns="urn:brocade.com:mgmt:brocade-sflow">
                 <enable xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete"/>
              </sflow>
       </config>
   </edit-config>
</rpc>

<rpc-reply message-id="2502" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

Configuring sFlow for VXLAN overlay gateway tunnels

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <sflow xmlns="urn:brocade.com:mgmt:brocade-sflow">
                <sflow-profile-name>profile1</sflow-profile-name>
                <remote-endpoint>1.2.3.4</remote-endpoint>
                <sflow-vlan-action>add</sflow-vlan-action>
                <sflow-vlan-range>2000</sflow-vlan-range>
            </sflow>
            <sflow xmlns="urn:brocade.com:mgmt:brocade-sflow">
                <sflow-profile-name>profile1</sflow-profile-name>
                <remote-endpoint>any</remote-endpoint>
                <sflow-vlan-action>add</sflow-vlan-action>
                <sflow-vlan-range>2000-3000</sflow-vlan-range>
            </sflow>
            <sflow xmlns="urn:brocade.com:mgmt:brocade-sflow">
                <sflow-profile-name>profile1</sflow-profile-name>
                <remote-endpoint>1.2.3.4</remote-endpoint>
                <sflow-vlan-action>remove</sflow-vlan-action>
                <sflow-vlan-range>2000</sflow-vlan-range>
            </sflow>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Retrieving flow-based sFlow statistics

Use the <get-config> RPC to retrieve the current configuration data and operational state data. Refer to "Retrieving configuration data" on page 11 and "Retrieving operational data" on page 15 for detailed instructions.

Chapter

Configuring Switched Port Analyzer 42

In this chapter

SPAN configuration with NETCONF overview

This chapter provides procedures for configuring Switched Port Analyzer (SPAN) monitoring sessions using the NETCONF interface. Refer to the *Network OS Layer 2 Switching Configuration Guide* for the following related information:

- A conceptual overview of SPAN
- General guidelines for using SPAN
- Procedures for configuring SPAN using the Network OS command line interface (CLI)
- Conceptual overview of RSPAN
- General guidelines for using RSPAN

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

- Use the <edit-config> remote procedure call (RPC) to configure SPAN.
- Use the <get-config> RPC to verify all or part of the SPAN configuration.

SPAN parameters are defined in the brocade-span YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all SPAN parameters, refer to the brocade-span.yang file.

Configuring ingress SPAN, egress SPAN, or bidirectional SPAN

Repeat the following procedure for each source port you want to monitor. A monitor session can have only one source port. For additional ports you must create additional monitor sessions.

To configure SPAN.

The following example configures an ingress SPAN session. It designates 1/0/15 as the source port and 1/0/18 as the destination port.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<rpc message-id="2800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <monitor xmlns="urn:brocade.com:mgmt:brocade-span">
                <session>
                    <session-number>1</session-number>
                    <description>Hello World</description>
                    <span-command>
                        <source>source</source>
                        <src-tengigabitethernet>tengigabitethernet
                         </src-tengigabitethernet>
                        <src-tengigabitethernet-val>1/0/15
                         </src-tengigabitethernet-val>
                        <destination>destination</destination>
                        <dest-tengigabitethernet>tengigabitethernet
                         </dest-tengigabitethernet>
                        <dest-tengigabitethernet-val>1/0/18
                         </dest-tengigabitethernet-val>
                        <direction>rx</direction>
                    </span-command>
                </session>
            </monitor>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2800" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting a SPAN connection from a session

To remove a single connection from a SPAN session.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2801" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <monitor xmlns="urn:brocade.com:mgmt:brocade-span">
                <session>
                    <session-number>1</session-number>
                    <description>Hello Wrold</description>
                    <span-command
                     xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                     operation="delete">
                        <source>source</source>
                        <src-tengigabitethernet>tengigabitethernet
                         </src-tengigabitethernet>
                        <src-tengigabitethernet-val>1/0/15
                         </src-tengigabitethernet-val>
                        <destination>destination</destination>
                        <dest-tengigabitethernet>tengigabitethernet
```

```
                        </dest-tengigabitethernet>
                       <dest-tengigabitethernet-val>1/0/18
                        </dest-tengigabitethernet-val>
                       <direction>both</direction>
                   </span-command>
               </session>
           </monitor>
       </config>
   </edit-config>
</rpc>

<rpc-reply message-id="2801" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Deleting a SPAN session

To remove a SPAN session, perform the following steps.

1. Issue the <get-config> RPC with a subtree filter to restrict the output to the <monitor> node in the urn:brocade.com:mgmt:brocade-span namespace.

   This step returns configuration information about existing monitoring sessions.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2802" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <monitor xmlns="urn:brocade.com:mgmt:brocade-span"/>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="2802" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <monitor xmlns="urn:brocade.com:mgmt:brocade-span">
        <session>
            <session-number>1</session-number>
            <description>Hello World</description>
            <span-command>
                <source>source</source>
                <src-tengigabitethernet>tengigabitethernet
                 </src-tengigabitethernet>
                <src-tengigabitethernet-val>1/0/15
                 </src-tengigabitethernet-val>
                <destination>destination</destination>
                <dest-tengigabitethernet>tengigabitethernet
                 </dest-tengigabitethernet>
                <dest-tengigabitethernet-val>1/0/18
                 </dest-tengigabitethernet-val>
                <direction>both</direction>
            </span-command>
        </session>
        <session>
            <session-number>2</session-number>
(output truncated)
```

2. Issue the <edit-config> RPC to configure the <monitor> node in the urn:brocade.com:mgmt:brocade-span namespace.
3. Under the <monitor> node, include the <session> node element and include the delete operation in the element tag.
4. Under the <session> node, include the <session-number> leaf element and specify the session number you want to delete.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2803" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <monitor xmlns="urn:brocade.com:mgmt:brocade-span">
                <session xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                 operation="delete">
                    <session-number>1</session-number>
                </session>
            </monitor>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2803" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

5. Reissue the <get-config> RPC used in step 1 to check that the session configuration information has been removed.

SPAN in management cluster

SPAN in management cluster supports mirroring of a source port to a destination port lying on a different switch in the management cluster. SPAN in management cluster is configured in the same manner, with the exception of the <source> leaf.

The <source> leaf controls the source and destination switches in the management cluster by the interface designation. The source and destination port can be anywhere in the management cluster. In this example, the <source> leaf is set as the third switch in the management cluster by the 3/0/15 leaf. However the destination is set to the fifth switch in the management cluster by the 5/0/18 leaf.

```
<rpc-reply message-id="2802" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <monitor xmlns="urn:brocade.com:mgmt:brocade-span">
        <session>
            <session-number>1</session-number>
            <description>Hello World</description>
            <span-command>
                <source>source</source>
                <src-tengigabitethernet>tengigabitethernet
                 </src-tengigabitethernet>
                <src-tengigabitethernet-val>3/0/15
                 </src-tengigabitethernet-val>
                <destination>destination</destination>
                <dest-tengigabitethernet>tengigabitethernet
```

```
                    </dest-tengigabitethernet>
                   <dest-tengigabitethernet-val>5/0/18
                    </dest-tengigabitethernet-val>
                   <direction>both</direction>
               </span-command>
           </session>
```

This configuration rule applies to ingress, egress, and both directions of SPAN. Refer to "SPAN configuration with NETCONF overview" on page 449.

Configuring RSPAN

The principal difference between configuring SPAN and RSPAN is that RSPAN requires a remote SPAN VLAN to be created first. This example demonstrates the configuration of a bidirectional RSPAN.

1. To create a VLAN interface

   The following example creates VLAN 1010.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>1010</name>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. Under the <vlan> node, set the value to <remote-span> to make the VLAN remote.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="11">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>1010</name>
                        <remote-span
xmlns="urn:brocade.com:mgmt:brocade-span"></remote-span>
                    </vlan>
                </interface>
            </interface-vlan>
```

```
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="11" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Issue the <edit-config> RPC to configure the <monitor> node in the urn:brocade.com:mgmt:brocade-span namespace.
4. Under the <monitor> node, include the <session> node element.
5. Under the <session> node, include the following leaf elements.
    a. In the <session-number> field, identify the session with a unique session number.
    b. *Optional:* In the <description> field, provide a descriptive text for the session.
6. Under the <session> node, include the <span-command> node element.
7. Under the <span-command> node, include the following leaf elements.
    a. In the <source> element, specify "source" to designate subsequent parameters as pertaining to the source port.
    b. In the <src-tengigabitethernet> element, specify "tengigabitethernet", "fortygigabitethernet", or "gigabitethernet", depending on the source port type.
    c. In the <src-tengigabitethernet-val> element, specify the source port in [*rbridge-id*/]*slot*/*port* format.
    d. In the <destination> element, specify "destination" to designate subsequent parameters as pertaining to the destination port.
    e. In the <dest-tengigabitethernet> element, specify "tengigabitethernet", "fortygigabitethernet", or "gigabitethernet", depending on the destination port type.
    f. In the <dest-vlan-val> element, specify the destination port in VLAN format.
    g. In the <direction> element, specify "both" to configure bidirectional SPAN.

The following example configures an RSPAN session. It designates 1/0/11 as the source port and VLAN 1010 as the destination.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
    <edit-config>
        <target>
            <running></running>
        </target>
        <config>
            <monitor xmlns="urn:brocade.com:mgmt:brocade-span">
                <session>
                    <session-number>1</session-number>
                    <span-command>
                        <source>source</source>

<src-tengigabitethernet>tengigabitethernet</src-tengigabitethernet>

<src-tengigabitethernet-val>1/0/11</src-tengigabitethernet-val>
                        <destination>destination</destination>

<dest-tengigabitethernet>rspan-vlan</dest-tengigabitethernet>
```

```
                        <dest-vlan-val>1010</dest-vlan-val>
                        <direction>both</direction>
                    </span-command>
                </session>
            </monitor>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Flow-based SPAN and RSPAN

You can replicate traffic from a defined source and direct it to snooping software on a designated port.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <monitor xmlns="urn:brocade.com:mgmt:brocade-span">
                <session>
                    <session-number>2</session-number>
                    <span-command>
                        <destination>destination</destination>

<dest-tengigabitethernet>rspan-vlan</dest-tengigabitethernet>
                        <dest-vlan-val>10</dest-vlan-val>
                    </span-command>
                </session>
            </monitor>
            <policy-map xmlns="urn:brocade.com:mgmt:brocade-policer">
                <po-name>pl1</po-name>
                <class>
                    <cl-name>c1</cl-name>
                    <span>
                        <session>1</session>
                    </span>
                </class>
            </policy-map>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>22/0/2</name>
                    <shutdown/>
                    <service-policy xmlns=”urn:brocade.com:mgmt:brocade-policer”>
                        <in>policymap1</in>
                    </service-policy>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="11" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <ok/>
</rpc-reply>
```

Chapter

Configuring SFP Breakout Mode

43

In this chapter

SFP breakout overview

Breakout interfaces are those interfaces created on the breakout SFP.Refer to the *Network OS Layer 2 Switching Configuration Guide* for the following related information:

- Breakout mode properties
- Breakout mode support
- Breakout mode interfaces
- Breakout mode limitations
- Configuring breakout mode for a chassis system

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

- Use the <edit-config> remote procedure call (RPC) to configure SFP breakout
- Use the <get-config> RPC to verify all or part of the SFP breakout configuration.

SFP breakout parameters are defined in the brocade-span YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all SPAN parameters, refer to the brocade-span.yang file.

Configuring breakout mode for a chassis system

The following example shows how to configure breakout mode on a blade in a chassis.

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <hardware xmlns="urn:brocade.com:mgmt:brocade-hardware">
                <connector>
                    <name>1/0/49</name>
                    <sfp>
                        <breakout>true</breakout>
                    </sfp>
                </connector>
```

```
            </hardware>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Section

I

Network OS Layer 3 Routing Features

This section describes Layer 3 routing features of Network OS, and includes the following chapters:

Chapter

IP Route Policy 44

In this chapter

IP route policy configuration with NETCONF overview

IP route policy controls how routes or IP subnets are transported from one subsystem to another subsystem. The IP route policy may perform “permit” or “deny” actions so that matched routes may be allowed or denied to the target subsystem accordingly. Additionally, an IP route policy may be used to modify the characteristics of a matched route and IP subnet pair.

Two types of IP route policies are supported; prefix-list and route-map.

This chapter provides procedures for configuring IP prefix lists and route maps. For conceptual details about IP route policies, refer to the *Network OS Layer 3 Routing Configuration Guide*.

Through the NETCONF interface, you can perform the following operations on route policies:

- Use the <edit-config> RPC to configure and activate route policies.
- Use the <get-config> RPC to view all or part of the route policy configuration.

Route policy parameters are defined in the brocade-ip-policy YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Configuring an IP prefix list

The following example configures an IP prefix list named test with two instances. A route is considered a match for instance 1 if this route is inside subnet 1.2.0.0/16 and has a mask length between 17 and 30. That is, route 1.2.1.0/24 matches, but route 1.2.1.1/32 does not, due to mask length.

```
<rpc message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>30</rbridge-id>
                <ip xmlns="urn:brocade.com:mgmt:brocade-common-def">
                    <prefix-list>
```

```
                    <name>test</name>
                    <instance>1</instance>
                    <action-ipp>deny</action-ipp>
                    <prefix-ipp>1.2.0.0/16</prefix-ipp>
                    <ge-ipp>17</ge-ipp>
                    <le-ipp>30</le-ipp>
                </prefix-list>
                <prefix-list>
                    <name>test</name>
                    <instance>2</instance>
                    <action-ipp>permit</action-ipp>
                    <prefix-ipp>1.1.0.0/16</prefix-ipp>
                </prefix-list>
            </ip>
        </rbridge-id>
    </config>
    </edit-config>
</rpc>

<rpc-reply message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring a route map

The following example configures a route-map named *test* that comprises two instances; instance 1 denies entry for any routes whose next-hop interface is te 0/1, and instance 2 allows entry for routes whose next-hop matches the IP subnets specified in the prefix-list pre-test (not shown). Additionally, each matched route has its tag set to 5000.

```
<rpc message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>30</rbridge-id>
                <route-map>
                    <name>test</name>
                    <action-rm>deny</action-rm>
                    <instance>1</instance>
                    <content>
                        <match>
                            <interface>
                                <tengigabitethernet-rmm>0/1</tengigabitethernet-rmm>
                            </interface>
                        </match>
                    </content>
                </route-map>
                <route-map>
                    <name>test</name>
                    <action-rm>permit</action-rm>
                    <instance>2</instance>
                    <content>
                        <match>
                            <ip>
```

```
                                <next-hop>
                                    <prefix-list-rmm-n>pre-test</prefix-list-rmm-n>
                                </next-hop>
                            </ip>
                        </match>
                        <set>
                            <tag>
                                <tag-rms>5000</tag-rms>
                            </tag>
                        </set>
                    </content>
                </route-map>
            </rbridge-id>>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <ok/>
</rpc-reply>
```

Configuring and activating an IP route policy

To set an IP route policy, configure the route policy, define static routes, and then apply the policy to the protocol.

In the following example, when route map test is applied, only static route 1.1.1.0/24 is exported into the OSPF domain because no matching rule exists in the IP prefix-list named pretest for route 11.11.11.0/24. The default action of prefix list is deny (no match), thus route 11.11.11.0/24 is not exported into the OSPF domain.

1. Define the route policy.
    a. Configure the IP prefix instance.

       This example configures instance 2 an IP prefix named pretest that permits routes that match 1.1.1.0/24.

```
<rpc message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>30</rbridge-id>
                <ip xmlns="urn:brocade.com:mgmt:brocade-common-def">
                    <prefix-list>
                        <name>pretest</name>
                        <instance>2</instance>
                        <action-ipp>permit</action-ipp>
                        <prefix-ipp>1.1.0.0/24</prefix-ipp>
                    </prefix-list>
                </ip>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="913"
 xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

b. Create the route map instance.

The following example provides a route-map that permits routes allowed by the prefix list named pretest.

```
<rpc message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>30</rbridge-id>
                <route-map>
                    <name>test</name>
                    <action-rm>permit</action-rm>
                    <instance>1</instance>
                    <content>
                        <match>
                            <ip>
                                <next-hop>
                                    <prefix-list-rmm-n>pre-test
                                     </prefix-list-rmm-n>
                                </next-hop>
                            </ip>
                        </match>
                    </content>
                </route-map>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="913"
 xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

2. Create the prefix and next hop for each static route.

```
<rpc message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
                <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                    <rbridge-id>30</rbridge-id>
                    <ip xmlns="urn:brocade.com:mgmt:brocade-common-def">
                        <route xmlns="urn:brocade.com:mgmt:brocade-rtm>
                            <static-route-nh>

<static-route-dest>11.11.11.0/24</static-route-dest>

<static-route-next-hop>2.2.2.1</static-route-next-hop>
```

```
                            </static-route-nh>
                            <static-route-nh>

<static-route-dest>11.11.11.0/24</static-route-dest>

<static-route-next-hop>2.2.2.2</static-route-next-hop>
                            </static-route-nh>
                        </route>
                    </ip>
                </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Apply the route map to the OSPF protocol to redistribute the static routes.

```
<rpc message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>30</rbridge-id>
                <router xmlns="urn:brocade.com:mgmt:brocade-common-def">
                    <ospf>
                        <redistribute>
                            <static>
                                <static-route-map>test<static-route-map>
                            </static>
                        </redistribute>
                        <area>
                            <area-id>0</area-id>
                        </area>
                    </ospf>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="913" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

IP Route Management 45

In this chapter

IP route management with NETCONF overview

IP route management is the term used in this chapter to refer to software that manages routes and next hops from different sources in a routing table, from which your Brocade device selects the best routes for forwarding IP packets. This route management software automatically gets activated at system bootup and does not require pre-configuration.

This chapter provides procedures and examples for configuring static routes using the NETCONF interface. Refer to the *Network OS Layer 3 Routing Configuration Guide* for the following related information:

- A conceptual overview of route management, including a discussion about how route management determines the best route among dynamic, static, and directly connected routes
- Procedures and examples for configuring static routes using the Network OS command line interface

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

- Use the <edit-config> remote procedure call (RPC) to configure static routes and perform other IP route management operations.
- Use the <get-config> RPC to verify all or part of the route management configuration.

Static route parameters are defined in the brocade-rtm YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*. The brocade-rtm.yang file provides definitions and explanations of all route management parameters.

NOTE
IP route management supports both IPv4 and IPv6 routes.

Configuring static routes

You can add a static route to IP route management using NETCONF operations. You can specify either the next-hop gateway or egress interface to add the route.

Specifying the next hop gateway

To specify the next hop gateway.

The following example configures a static route to 207.95.7.0, using 207.95.6.157 as the next-hop gateway.

```
<rpc message-id="3300" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>30</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <route>
                            <static-route-nh>
                                <static-route-dest>207.95.7.0/24</static-route-dest>
                                <static-route-next-hop>207.95.6.157
                                 </static-route-next-hop>
                            </static-route-nh/>
                        </route>
                    </rtm-config>
                </ip>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3300" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the egress interface

To configure a static IP route with a physical interface port or port-channel

The following example configures a static IP route for destination network 192.128.2.0/24. Because an Ethernet port is specified instead of a gateway IP address as the next hop, the Brocade device forwards traffic for the 192.128.2.0/24 network to the tengigabitethernet port 101/4/1.

```
<rpc message-id="3301" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>30</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <route>
                            <static-route-oif>
                                <static-route-dest>192.128.2.0/24
                                 </static-route-dest>
                                <static-route-oif-type>tengigabitethernet
                                 </tengigabitethernet>
```

```
                                <static-route-oif-name>101/4/1
                                 </static-route-oif-name>
                            </static-route-oif/>
                        </route>
                    </rtm-config>
                </ip>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3301" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring the default route

The following example configures a default route with a next hop of 207.95.6.157.

```
<rpc message-id="3302" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>30</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <route>
                            <static-route-nh>
                                <static-route-dest>0.0.0.0/0</static-route-dest>
                                <static-route-next-hop>207.95.6.157
                                 </static-route-next-hop>
                            </static-route-nh/>
                        </route>
                    </rtm-config>
                </ip>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3302" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying route attributes

When specifying a route, include leaf elements under the <route-attributes> node that let you:

- Specify a tag value of a route to use for route filtering with a route map (<tag> element).
- Specify a cost metric of a route (<metric> element).

The following example sets the cost of a route to 10 and specifies a tag value of 5 for route filtering.

```
rpc message-id="3303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
```

```
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>30</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <route>
                            <static-route-nh>
                                <static-route-dest>207.95.7.0/24
                                 </static-route-dest>
                                <static-route-next-hop>207.95.6.157
                                 </static-route-next-hop>
                                <route-attributes>
                                    <tag>5</tag>
                                    <metric>10</metric>
                                </route-attributes>
                            </static-route-nh/>
                        </route>
                    </rtm-config>
                </ip>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3303" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling IP load sharing

Use the <load-sharing> element to balance IP traffic across up to eight equal paths.

```
<rpc message-id="3304" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>30</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <load-sharing/>
                    </rtm-config>
                </ip>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3304" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

NOTE
This command is deprecated.

Chapter

Configuring OSPF 46

In this chapter

OSPF configuration with NETCONF overview

Open Shortest Path First (OSPF) is a link-state routing protocol that uses link-state advertisements (LSAs) to update neighboring routers about its interfaces. Each router maintains an identical area-topology database to determine the shortest path to any neighboring router.

This chapter provides procedures and examples for configuring OSPF using the NETCONF interface. Refer to the *Network OS Layer 3 Routing Configuration Guide* for the following related information:

- A conceptual overview of OSPF
- An overview of Designated routers
- Conceptual details about key configurable entities, such as stubs, stubby areas, not so stubby areas, totally stubby areas, and virtual links
- Procedures for configuring the Ethernet management interface

  You need an Ethernet management interface before you can configure a Secure Shell (SSH) connection. As a result, you cannot begin a NETCONF session until this interface is configured.
- Procedures and examples for configuring OSPF using the Network OS command line interface

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

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

OSPF parameters are defined in the brocade-ospf YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all OSPF parameters, refer to the brocade-ospf.yang file.

OSPF over VRF

With Network OS 4.0 and later, OSPF can run over multiple Virtual Forwarding and Routing (VRF) mechanisms. OSPF maintains multiple instances of the routing protocol to exchange route information among various VRFs. A multi-VRF-capable router maps an input interface to a unique VRF, based on user configuration. These input interfaces can be physical or SVIs. By default, all input interfaces are attached to the default VRF. All OSPF commands supported in Network OS 4.0 and later are available over default and non-default OSPF instances.

NOTE
For more information about OSPF over VRF, refer to Chapter 52, “Configuring VRF”.

OSPF in a VCS environment

Figure 2 shows one way in which OSPF can be used in a VCS Fabric cluster environment. Routers RB1 and RB2, as well as the MLX switches, are configured with OSPF. Switches RB3, RB4, and RB5 are Layer 2 switches.

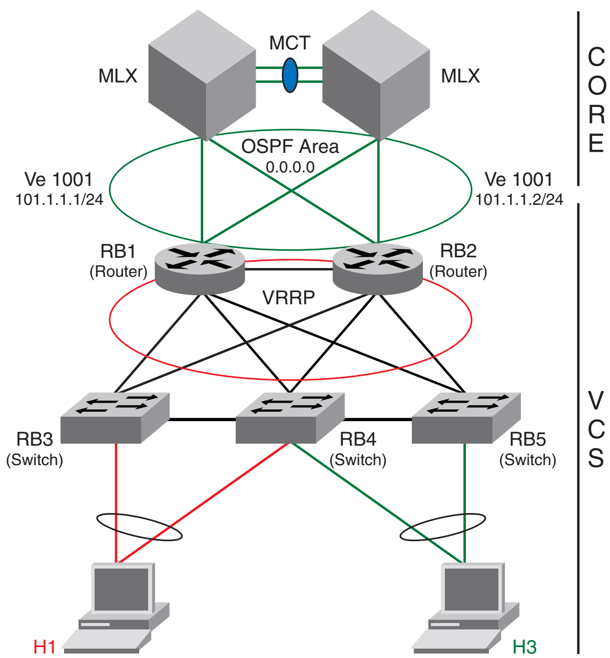


FIGURE 2 **OSPF example in a VCS environment**

1. On Router RB1, issue an <edit-config> RPC to perform the following tasks:
    a. Configure a VLAN for the router.
    b. Enable OSPF for the RB1.
    c. Create an OSPF area on RB1.
    d. Configure a virtual Ethernet (VE) interface using the VLAN number created in step a.
    e. Configure an IP address for the VE interface.

f. Assign the VE interface to the area created in step c.

g. Enable the VE interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2600" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface-vlan>
                    <interface>
                        <vlan>
                            <name>1001</name>
                        </vlan>
                    </interface>
                </interface-vlan>
            </interface>
            <rbridge-id xmlns xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <ospf/>
                    <area>0.0.0.0</area>
                </router>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>1001</name>
                        <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <ip-config>
                                <address>
                                    <address>101.1.1.1/24</address>
                                </address>
                            </ip-config>
                            <interface-vlan-ospf-conf
                             xmlns="urn:brocade.com:mgmt:brocade-ospf">
                                <ospf1>
                                    <area>0.0.0.0</area>
                                </ospf1>
                            </interface-vlan-ospf-conf>
                        </ip>
                        <shutdown
                         xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    </ve>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2600" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. On Router RB2, issue an <edit-config> RPC to perform the following tasks.

   a. Configure a VLAN for Router RB2.

   b. Enable OSPF for Router RB2.

c. Create an OSPF area on Router RB2.

d. Configure a virtual Ethernet (VE) interface using the VLAN number created in step a.

e. Configure an IP address for the VE.

f. Assign the interface to the area created in step c.

g. Enable the VE interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2601" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface-vlan>
                    <interface>
                        <vlan>
                            <name>1001</name>
                        </vlan>
                    </interface>
                </interface-vlan>
            </interface>
            <rbridge-id xmlns xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>2</rbridge-id>
                <router>
                    <ospf/>
                    <area>0.0.0.0</area>
                </router>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>1001</name>
                        <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <ip-config>
                                <address>
                                    <address>101.1.1.2/24</address>
                                </address>
                            </ip-config>
                            <interface-vlan-ospf-conf
                             xmlns="urn:brocade.com:mgmt:brocade-ospf">
                                <ospf1>
                                    <area>0.0.0.0</area>
                                </ospf1>
                            </interface-vlan-ospf-conf>
                        </ip>
                        <shutdown
                         xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/>
                    </ve>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2601" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Assign VLAN 1001 to a VLAG.

Performing basic OSPF configuration

To begin using OSPF on the router, perform these steps.

1. Follow the rules in the "OSPF configuration rules" on page 477.
2. Enable OSPF on the router.
3. Assign the areas to which the router will be attached. Refer to "Assigning OSPF areas" on page 478.
4. Assign individual interfaces to the OSPF areas. Refer to "Assigning interfaces to an area" on page 482.
5. Assign a virtual link to any Area Border Router (ABR) that does not have a direct link to the OSPF backbone area. Refer to "Assigning virtual links" on page 483.
6. Refer to "Changing other settings" on page 484.

OSPF configuration rules

- If a router is to operate as an Autonomous System Boundary Router (ASBR), you must enable the ASBR capability at the system level.
- Redistribution must be enabled on routers configured to operate as ASBRs.
- All router ports must be assigned to one of the defined areas on an OSPF router. When a port is assigned to an area, all corresponding subnets on that port are automatically included in the assignment.

Enabling OSPF on the router

OSPF can be activated only in the RBridge ID context. To enable OSPF on the router.

The following example enables OSPF on routing bridge 101.

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
       <target>
          <running/>
       </target>
       <config>
          <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
              <rbridge-id>101</rbridge-id>
              <router>
                  <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf”>
                      <vrf>default-vrf</vrf>
                  </ospf>
              </router>
           </rbridge-id>
       </config>
   </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
```

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

Disabling OSPF on the router

To disable OSPF, include the delete operation in the <ospf> header tag, as shown in the following example RPC.

```
<rpc message-id="2603" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>101</rbridge-id>
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf”
operation="delete">
                        <vrf>default-vrf</vrf>
                    </ospf>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2603" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Assigning OSPF areas

Once OSPF is enabled on the system, you can assign areas. Assign an IP address or number as the area ID for each area. The area ID is representative of all IP addresses (subnets) on a router port. Each port on a router can support one area.

An area can be normal, a stub, a totally stubby area, or a Not-So-Stubby Area (NSSA). For a detailed explanation of these terms, refer to the *Network OS Layer 3 Routing Configuration Guide*.

The following example RPC sets up the backbone area (0.0.0.0).

```
<rpc message-id="2604" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>1001</name>
                    </vlan>
                </interface>
            </interface-vlan>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>10</rbridge-id>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>1001</name>
```

```
                        <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <ip-config>
                                <address>
                                    <address>101.1.1.1/24</address>
                                </address>
                            </ip-config>
                            <interface-vlan-ospf-conf
                             xmlns="urn:brocade.com:mgmt:brocade-ospf">
                                <ospf1>
                                    <area>0.0.0.0</area>
                                </ospf1>
                            </interface-vlan-ospf-conf>
                        </ip>
                    </ve>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2604" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Assigning a totally stubby area

The following example RPC configures OSPF area 1.1.1.1 as a totally stubby area. That is, it disables summary LSAs for the stub area. The <stub> node element designates a stub area. The <stub-value> element determines the cost of entering or leaving the area. The empty <no-summary> element disables summary LSAs and renders the stub area a totally stubby area.

```
<rpc message-id="2605" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>101</rbridge-id>
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf>
                        <area>
                            <area-id>1.1.1.1</area-id>
                            <stub>
                                <metric>
                                    <stub-value>99</stub-value>
                                    <no-summary/>
                                </metric>
                            </stub>
                        </area>
                    </ospf>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2605" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
```

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

Assigning a Not-So-Stubby Area

The following example RPC configures OSPF area 1.1.1.1 as an NSSA. The <nssa> node designates a not-so-stubby area. The <nssa-value> element determines the cost of entering or leaving the area.

```
<rpc message-id="2606" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>101</rbridge-id>
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf>
                        <area>
                            <area-id>1.1.1.1</area-id>
                            <nssa>
                                <metric>
                                    <nssa-value>1</nssa-value>
                                </metric>
                            </nssa>
                        </area>
                    </ospf>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2606" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring a summary-address for the NSSA

To configure a summary-address in NSSA 1.1.1.1, issue the following RPC. (This example assumes that you have already configured NSSA 1.1.1.1.)

```
<rpc message-id="2607" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>101</rbridge-id>
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf>
                        <area>
                            <area-id>1.1.1.1</area-id>
                            <nssa>
                                <metric>
                                    <nssa-value>10</nssa-value>
                                </metric>
                            </nssa>
```

```
                    </area>
                    <summary-address>
                        <sum-address>209.157.1.0</sum-address>
                        <sum-address-mask>255.255.255.0</sum-address-mask>
                    </summary-address>
                </ospf>
            </router>
        </rbridge-id>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2607" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Assigning an area range (optional)

If you do assign a range for an area, you must also specify a range effect, which can be “advertise” or “not-advertise”.

The following example RPC defines an area range for subnets on 0.0.0.10 and 0.0.0.20.

```
<rpc message-id="2608" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>101</rbridge-id>
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf>
                        <area>
                            <area-id>0.0.0.10</area-id>
                            <normal/>
                            <range>
                                <range-address>193.45.0.0</range-address>
                                <range-mask>255.255.0.0</range-mask>
                                <range-effect>not-advertise</range-effect>
                            </range>
                        </area>
                        <area>
                            <area-id>0.0.0.20</area-id>
                            <normal/>
                            <range>
                                <range-address>193.45.0.0</range-address>
                                <range-mask>255.255.0.0</range-mask>
                                <range-effect>not-advertise</range-effect>
                            </range>
                        </area>
                    </ospf>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2608" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
```

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

Assigning interfaces to an area

For example, to assign interface 7/1/8 of a router area whose area ID is 192.5.0.0, and then save the changes, issue the following RPC.

```
<rpc message-id="2609" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>7/1/8</name>
                    <ip>
                        <interface-te-ospf-conf
                         xmlns="urn:brocade.com:mgmt:brocade-ospf">
                            <ospf1>
                                <area>192.5.0.0</area>
                            </ospf1>
                        </interface-te-ospf-conf>
                    </ip>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2609" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

If you want to set an interface to passive mode, use the empty <passive/> leaf element instead of the <area> leaf element in the previous example.

If you want to block flooding of outbound LSAs on specific OSPF interfaces, instead of the <area> element in the previous example, using the following elements:

```
<database-filter>
    <all-out/>
</database-filter>
```

Assigning virtual links

Figure shows an OSPF area border router, Device A, that is cut off from the backbone area (area 0). To provide backbone access to Device A, you can add a virtual link between Device A and Device C using area 1 as a transit area. To configure the virtual link, you define the link on the router that is at each end of the link. No configuration for the virtual link is required on the routers in the transit area.

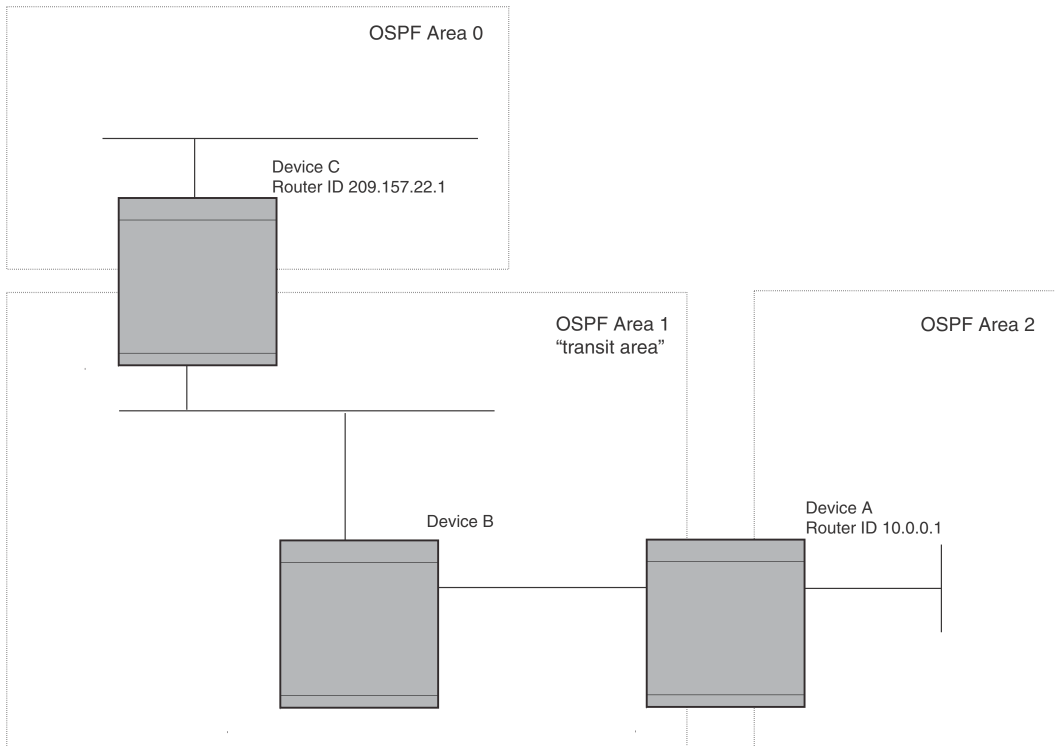


FIGURE 3 **Defining OSPF virtual links within a network**

To define the virtual link on Device A, establish a NETCONF session with Device A, and issue the following <edit-config> RPC. The RPC configures both areas in which the router participates (area 2 and area 1). For the transition area (area 1), the <area> node element also includes the <virt-link-neighbor> element, which specifies the router address of the ABR that connects the transition area to the backbone area.

```
<rpc message-id="2610" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>101<rbridge-id>
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf>
                        <area>
                            <area-id>2</area-id>
                        </area>
                        <area>
                            <area-id>1</area-id>
```

```
                        <virtual-link>
                            <virt-link-neighbor>209.157.22.1
                             </virt-link-neighbor>
                        </virtual-link>
                    </area>
                </ospf>
            </router>
        </rbridge-id>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="2610" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

To configure the virtual link on Device C, establish a NETCONF session with Device C and issue the following <edit-config> RPC.

```
<rpc message-id="2611" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>101</rbridge-id>
                <router>
                    <ospf>
                        <area>
                            <area-id>0</area-id>
                        </area>
                        <area>
                            <area-name>1</area-name>
                            <virtual-link>
                                <virt-link-neighbor>10.0.0.1</virt-link-neighbor>
                            </virtual-link>
                        </area>
                    </ospf>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2611" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Changing other settings

Refer to the *Network OS YANG Reference Manual* for other global and interface-level parameters you can use to change default OSPF settings. Refer to the brocade-ospf.yang file for descriptions of each parameter. Some commonly configured items include:

- Changing reference bandwidth to change interface costs by using the <auto-cost> node.
- Defining redistribution filters for the Autonomous System Boundary Router (ASBR) by editing the <redistribute> node.

Chapter

Configuring OSPFv3 47

In this chapter

OSPFv3 configuration with NETCONF overview

IPv6 supports OSPF Version 3 (OSPFv3). OSPFv3 functions similarly to OSPF Version 2 (OSPFv2), with several enhancements.

Open Shortest Path First (OSPF) is a link-state routing protocol. OSPF uses link-state advertisements (LSAs) to update neighboring routers about its interfaces and information on those interfaces. A device floods LSAs to all neighboring routers to update them about the interfaces. Each router maintains an identical database that describes its area topology to help a router determine the shortest path between it and any neighboring router.

To configure OSPFv3, you must perform the following steps.

- Configure the router ID.
- Enable OSPFv3 globally.
- Configure OSPFv3 areas.

- Assign OSPFv3 areas to interfaces.

Configuring the router ID, OSPFv3, and OSPFv3 areas

When configuring OSPFv3 the router ID for a device must be specified. When OSPFv3 is enabled on a device, the device enters IPv6 OSPF configuration level. Several commands can then be accessed that allow the configuration of OSPFv3.

Areas can be assigned as OSPFv3 areas. Enable IPv6 on each interface over which you plan to enable OSPFv3. You enable IPv6 on an interface by configuring an IPv6 address.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>195</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <router-id>1.1.1.1</router-id>
                    </rtm-config>
                </ip>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <area>
                                <area-id>0</area-id>
                            </area>
                            <area>
                                <area-id>3.3.3.3</area-id>
                            </area>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Assigning OSPFv3 areas to interfaces

Defined OSPFv3 areas can be assigned to device interfaces.

Ensure that OSPFv3 areas are assigned.

The following example configures and enables OSPFv3 on two specified interfaces, and assigns a loopback interface 1 to two router areas.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <loopback xmlns="urn:brocade.com:mgmt:brocade-intf-loopback">
                    <id>1</id>
                    <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                        <ipv6-config>
                            <address>
                                <ipv6-address>
                                    <address>2001:54:54:54::54/128</address>
                                </ipv6-address>
                            </address>
                        </ipv6-config>
                        <interface-ospfv3-conf
xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <interface-area>0.0.0.0</interface-area>
                        </interface-ospfv3-conf>
                    </ipv6>
                </loopback>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Configuring an NSSA

OSPFv3 areas can be defined as NSSA areas with modifiable parameters.

The following example sets an additional cost of 33 on an NSSA defined as 3

```
<?xml version="1.0" ?>
<rpc nc:message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>122</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <router-id>122.122.122.122</router-id>
                </ip>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <area>
                                <area-id>53</area-id>
```

```
                                <nssa>
                                    <nssa-area-no-summary></nssa-area-no-summary>
                                    <area-default-information-originate>

<area-default-information-originate-metric>33</area-default-information-originate
-metric>
                                    </area-default-information-originate>
                                </nssa>
                            </area>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Configuring a stub area

OSPFv3 areas can be defined as stub areas with modifiable parameters.

The following example sets an additional cost of 100 on a stub area defined as 4.

```
<?xml version="1.0" ?>
<rpc nc:message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>122</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <router-id>122.122.122.122</router-id>
                </ip>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <area>
                                <area-id>0.0.0.55</area-id>
                                <stub>
                                    <stub-area-metric>55</stub-area-metric>
                                </stub>
                            </area>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Configuring virtual links

A virtual link between two devices can be configured.

In this optional task, you configure a virtual link and define a virtual link endpoint on two devices, ABR1 and ABR 2.

On ABR1:

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <router-id>1.1.1.1</router-id>
                    </rtm-config>
                </ip>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <area>
                                <area-id>1</area-id>
                                <virtual-link>

<virtual-link-neighbor>2.2.2.2</virtual-link-neighbor>
                                </virtual-link>
                            </area>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

On ABR2:

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
```

```
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>10</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <router-id>2.2.2.2</router-id>
                    </rtm-config>
                </ip>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <area>
                                <area-id>1</area-id>
                                <virtual-link>

<virtual-link-neighbor>1.1.1.1</virtual-link-neighbor>
                                </virtual-link>
                            </area>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Redistributing routes into OSPFv3

OSPFv3 routes can be redistributed and the routes to be redistributed can be specified.

In this optional task, you configure the redistribution of static routes on a device (device1) into OSPFv3 and the redistribution of connected routes on a second device (device2) into OSPFv3, specifying which connected routes are redistributed.

The following example redistributes static routes into OSPFv3 on a device.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>122</rbridge-id>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <redistribute>
                                <redistribute-static></redistribute-static>
                            </redistribute>
                        </ospf>
                    </router>
```

```
            </ipv6>
        </rbridge-id>
    </config>
  </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
  <ok/>
</rpc-reply>
```

The following example redistributes connected routes into OSPFv3 on a device and specifies a route map.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>122</rbridge-id>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <redistribute>
                                <redistribute-connected>
                                    <connected-route-map>rmap1</connected-route-map>
                                </redistribute-connected>
                            </redistribute>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Modifying Shortest Path First timers

The Shortest Path First (SPF) delay and hold time can be modified.

The following example changes the SPF delay and hold time.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>122</rbridge-id>
                <ipv6>
```

```
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                        <vrf>default-vrf</vrf>
                        <timers>
                            <spf>
                                <spf-delay>10</spf-delay>
                                <spf-hold-time>20</spf-hold-time>
                            </spf>
                        </timers>
                    </ospf>
                </router>
            </ipv6>
        </rbridge-id>
    </config>
  </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Configuring the OSPFv3 LSA pacing interval

The interval between OSPFv3 LSA refreshes can be modified.

The following example changes the interval between OSPFv3 LSA refreshes and then restores the pacing interval to its default value.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>122</rbridge-id>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <timers>
                                <lsa-group-pacing>120</lsa-group-pacing>
                            </timers>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Configuring default route origin

OSPFv3 default routes can be created and advertised.

The following example creates and advertises a default route with a metric of 2 and a type 1 external route.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>122</rbridge-id>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <global-default-information-originate>

<global-default-information-originate-always></global-default-information-origina
te-always>

<global-default-information-originate-metric>2</global-default-information-origin
ate-metric>

<global-default-information-originate-metric-type>type1</global-default-informati
on-originate-metric-type>
                            </global-default-information-originate>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Enabling event logging

OSPFv3 event logging, such as neighbor state changes and database overflow conditions, can be disabled and re-enabled.

The following example enables the logging of OSPFv3 events.

```
<?xml version="1.0" ?>
<rpc nc:message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
```

```
                <rbridge-id>122</rbridge-id>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <log status-change></log status-change>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Configuring administrative distance based on route type

The default administrative distances for intra-area routes, inter-area routes, and external routes can be altered.

In the following example, the default administrative distances for intra-area routes, inter-area routes, and external routes are changed.

```
<?xml version="1.0" ?>
<rpc nc:message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>122</rbridge-id>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <distance>
                                <route-type>external</route-type>
                                <distance-value>100</distance-value>
                            </distance>
                            <distance>
                                <route-type>inter-area</route-type>
                                <distance-value>90</distance-value>
                            </distance>
                            <distance>
                                <route-type>intra-area</route-type>
                                <distance-value>80</distance-value>
                            </distance>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Changing the reference bandwidth for the cost on OSPFv3

The reference bandwidth for OSPFv3 can be altered, resulting in various costs.

In the following example, the auto-cost reference-bandwidth is changed to 500.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>122</rbridge-id>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <auto-cost>
                                <reference-bandwidth>500</reference-bandwidth>
                            </auto-cost>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Setting all OSPFv3 interfaces to the passive state

All OSPFv3 interfaces for a specified RBridge can be set as passive. This causes them to drop all OSPFv3 control packets.

In the following example, all OSPFv3 interfaces for a specified RBridge are set as passive, causing them to drop all the OSPFv3 control packets.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>122</rbridge-id>
```

```
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <default-passive-interface></default-passive-interface>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Configuring IPSec on an OSPFv3 area

The following example shows how to configure IPSec on an OSPFv3 area.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>2</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <router-id>10.10.10.10</router-id>
                    </rtm-config>
                </ip>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <area>
                                <area-id>10</area-id>
                                <normal></normal>
                                <virtual-link>

<virtual-link-neighbor>10.1.1.1</virtual-link-neighbor>
                                    <link-properties>
                                        <authentication>
                                            <ipsec-auth-key-config>
                                                <spi>520</spi>
                                                <ah>hmac-sha1</ah>
                                                <ah-no-encrypt></ah-no-encrypt>

<ah-key>0000000000000000000000000000000000000000</ah-key>
                                            </ipsec-auth-key-config>
                                        </authentication>
                                    </link-properties>
                                </virtual-link>
                            </area>
```

```
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring IPsec on an OSPFv3 interface

The following example shows how to configure IPsec on an OSPFv3 interface.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fortygigabitethernet>
                    <name>1/0/49</name>
                    <ipv6>
                        <interface-ospfv3-conf
xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <interface-area>11</interface-area>
                            <authentication>
                                <ipsec-auth-key-config>
                                    <spi>512</spi>
                                    <esp>NULL</esp>
                                    <esp-auth>hmac-sha1</esp-auth>

<esp-auth-key>$WlpaWlpaWlpaWlpaWlpaWlpaWlpaWlpaWlpaWlpaWlpaWlpaWg==</esp-auth
-key>
                                </ipsec-auth-key-config>
                            </authentication>
                        </interface-ospfv3-conf>
                    </ipv6>
                </fortygigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring IPsec on OSPFv3 virtual links

The following example shows how to configure IPsec on OSPFv3 virtual links.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>2</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <router-id>10.10.10.10</router-id>
                    </rtm-config>
                </ip>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <area>
                                <area-id>10</area-id>
                                <normal></normal>
                                <virtual-link>

<virtual-link-neighbor>10.1.1.1</virtual-link-neighbor>
                                    <link-properties>
                                        <authentication>
                                            <ipsec-auth-key-config>
                                                <spi>520</spi>
                                                <ah>hmac-sha1</ah>
                                                <ah-no-encrypt></ah-no-encrypt>

<ah-key>0000000000000000000000000000000000000000</ah-key>
                                            </ipsec-auth-key-config>
                                        </authentication>
                                    </link-properties>
                                </virtual-link>
                            </area>
                            <nonstop-routing></nonstop-routing>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring max-metric on OSPFv3 virtual links

```
<rpc message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
```

```
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <ospf>>
                        <vrf>default-vrf</vrf>
                        <max-metric>
                            <router-lsa>
                                <external-lsa>
                                    <external-lsa-val>1234343</external-lsa-val>
                                </external-lsa>
                                <summary-lsa>
                                    <summary-lsa-val>1223324</summary-lsa-val>
                                </summary-lsa>
                                <link>
                                    <ptp>true</ptp>
                                    <stub>true</stub>
                                    <transit>true</transit>
                                </link>
                                <on-startup>
                                    <time>10</time>
                                    <external-lsa>

<external-lsa-val-onstartup>100</external-lsa-val-onstartup>
                                    </external-lsa>
                                    <summary-lsa>

<summary-lsa-val-onstartup>199</summary-lsa-val-onstartup>
                                    </summary-lsa>
                                    <link>
                                        <ptp>true</ptp>
                                        <stub>true</stub>
                                        <transit>true</transit>
                                    </link>
                                </on-startup>
                            </router-lsa>
                        </max-metric>
                    <ospf>
                <router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2602" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Specifying the key rollover timer

The following example shows how to specify the key rollover timer.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
```

```
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <router-id>10.11.12.13</router-id>
                    </rtm-config>
                </ip>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <key-rollover-interval>240</key-rollover-interval>
                            <nonstop-routing></nonstop-routing>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Chapter

Configuring VRRP 48

In this chapter

VRRP and VRRP-E configuration with NETCONF overview

This chapter provides procedures for configuring the Virtual Router Redundancy Protocol (VRRP) using the NETCONF interface. Refer to the *Network OS Layer 3 Routing Configuration Guide* for the following related information:

- A conceptual overview of VRRP
- General guidelines
- An overview of VRRP and VRRP-E packet behavior
- Procedures for configuring VRRP using the command line interface

Through the NETCONF interface, you can perform the following operations on VRRP and VRRP-E:

- Use the <edit-config> RPC to configure VRRP and VRRP-E.
- Use the <get-config> RPC to validate configuration settings.

VRRP parameters are defined in the brocade-vrrp YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Figure 4 shows an example of a basic VRRP setup to illustrate some basic VRRP concepts. Router 1 and Router 2 are two physical routers that can be configured to compose one virtual router. This virtual router would provide redundant network access for Host 1. If Router 1 were to fail, Router 2 could provide the default gateway out of the subnet.

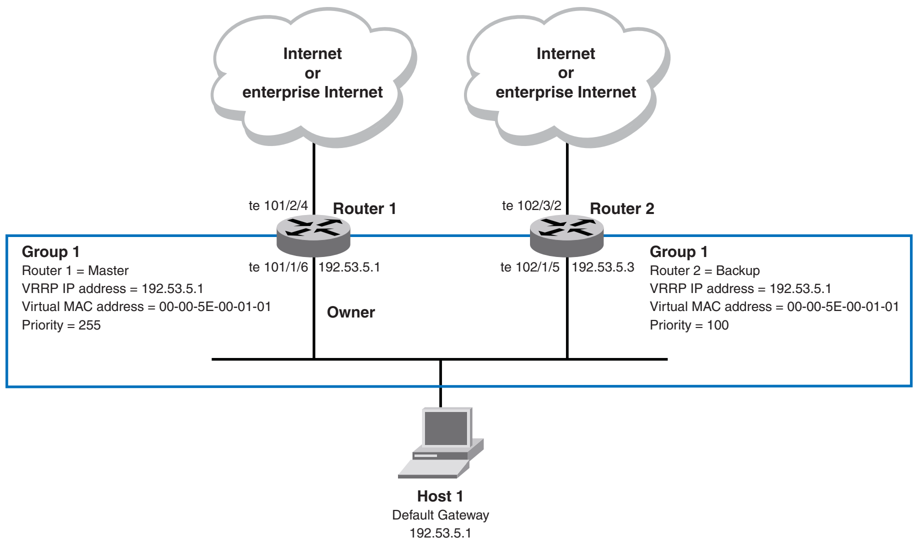


FIGURE 4 Basic VRRP configuration example

The procedures that follow show how to implement this basic configuration using NETCONF operations. The procedure is for VRRP. Refer to "VRRP-E differences for basic configuration" on page 505 for variations for VRRP-E.

Before configuring VRRP or VRRP-E, consider the following terms:

- Virtual Router—A collection of physical routers that can use either VRRP or VRRP-E to provide redundancy to routers within a LAN.
- Virtual Router Group—A group of physical routers that are assigned to the same virtual router.
- Virtual Router Address—The address you are backing up:
  - For VRRP: The virtual router IP address must belong to the same subnet as a real IP address configured on the VRRP interface, and can be the same as a real IP addresses configured on the VRRP interface. The virtual router whose virtual IP address is the same as a real IP address is the IP address *owner* and the default *master*.
  - For VRRP-E: The virtual router IP address must belong to the same subnet as a real IP address configured on the VRRP-E interface, but cannot be the same as a real IP address configured on the VRRP-E interface.
- Owner—This term applies only to VRRP, not to VRRP-E. The owner is the physical router whose real interface IP address is the IP address that you assign to the virtual router. The owner responds to packets addressed to any of the IP addresses in the corresponding virtual router. The owner, by default, is the master (refer to "Master") and has the highest priority (255).
- Master—The physical router that responds to packets addressed to any of the IP addresses in the corresponding virtual router. For VRRP, if the physical router whose real interface IP address is the IP address of the virtual router, then this physical router is always the master. For VRRP-E, the router with the highest priority becomes the master. If two routers have the same priority, the router with the highest IP address becomes the master.

- Backup—Routers that belong to a virtual router but are not the master. Then, if the master becomes unavailable, the backup router with the highest priority (a configurable value) becomes the new master. By default, backup routers are given a priority of 100. You can assign a backup a priority value of 3 through 254.

VRRP basic configuration example

The following procedures configure the basic configuration shown in Figure 4 on page 502 for VRRP.

NOTE
The interface links used in this example are all 10 Gigabit Ethernet. For VRRP, these links could also be Gigabit Ethernet, 40 Gigabit Ethernet, 100 Gigabit Ethernet, port-channel, or VE interface.

Configuring the master router

To create a basic master router configuration for Router 1 in Figure 4 on page 502,

The following example configures the basic master router shown in Figure 4 on page 502.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
            <config>
            <vcsmode xmlns="urn:brocade.com:mgmt:brocade-vcs">
                <vcs-mode>true</vcs-mode>
                <vcs-cluster-mode>false</vcs-cluster-mode>
            </vcsmode>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>101</rbridge-id>
                <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <hide-vrrp-holder xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                        <vrrp/>
                    </hide-vrrp-holder>
                </protocol>
            </rbridge-id>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>101/1/6</name>
                    <ip>
                        <ip-config
                         xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <address>
                                <address>192.53.5.1</address>
                            </address>
                        </ip-config>
                    </ip>
                    <vrrp xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                        <vrid>1</vrid>
                        <virtual-ip>
                            <virtual-ipaddr>192.53.5.1</virtual-ipaddr>
                        </virtual-ip>
```

```
                </vrrp>
            </tengigabitethernet>
        </interface>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Configuring the backup router

To create a basic backup router configuration for Router 2 in Figure 4 on page 502,

The following example configures the basic backup router shown in Figure 4 on page 502

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <vcsmode xmlns="urn:brocade.com:mgmt:brocade-vcs">
        <vcs-mode>true</vcs-mode>
        <vcs-cluster-mode>false</vcs-cluster-mode>
      </vcsmode>
      <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
        <rbridge-id>102</rbridge-id>
        <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
          <hide-vrrp-holder xmlns="urn:brocade.com:mgmt:brocade-vrrp">
            <vrrp/>
          </hide-vrrp-holder>
        </protocol>
      </rbridge-id>
      <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <tengigabitethernet>
          <name>102/1/5</name>
          <ip>
            <ip-config
             xmlns="urn:brocade.com:mgmt:brocade-ip-config">
              <address>
                <address>192.53.5.3</address>
              </address>
            </ip-config>
          </ip>
          <vrrp xmlns="urn:brocade.com:mgmt:brocade-vrrp">
            <vrid>1</vrid>
            <virtual-ip>
              <virtual-ipaddr>192.53.5.1</virtual-ipaddr>
            </virtual-ip>
          </vrrp>
        </tengigabitethernet>
      </interface>
    </config>
  </edit-config>
</rpc>
```

```
<rpc-reply message-id="3401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

VRRP-E differences for basic configuration

If you were to configure the two routers shown in Figure 4 on page 502, you must consider the following items specific to VRRP-E:

- Specifying the <vrrp> element in the urn:brocade.com:mgmt:brocade-vrrp namespace enables VRRP-E as well as VRRP.
- VRRP-E virtual routers ca be configured on VE interfaces only.
- VRRP and VRRP-E cannot be simultaneously enabled on the VDX 6740 or 6740T.
- The group ID and the virtual router IP address are specified under the <vrrpe> node instead of the <vrrp> node.

```
<vrrpe xmlns="urn:brocade.com:mgmt:brocade-vrrp">
    <vrid>1</vrid>
    <virtual-ip>
        <virtual-ipaddr>192.56.7.25</virtual-ipaddr>
    </virtual-ip>
</vrrpe>
```

- Specification of the master router is done by giving the master a higher priority than the backup router. The priority is also specified under the <vrrpe> node.

```
<vrrpe xmlns="urn:brocade.com:mgmt:brocade-vrrp">
    <priority>110</priority>
    <vrid>1</vrid>
    <virtual-ip>
        <virtual-ipaddr>192.56.7.25</virtual-ipaddr>
    </virtual-ip>
</vrrpe>
```

- For VRRP-E, you cannot assign the same IP address to the physical interface and to the virtual router.

Enabling preemption

You can allow a backup router that is acting as the master to be preempted by another backup router with a higher priority value.

By default, preemption is enabled for VRRP, or disabled for VRRP-E.

NOTE
If preemption is disabled for VRRP, the owner router is not affected because the owner router always preempts the active master.

The procedure for enabling pre-emption differs depending on the Ethernet link interface type, which for VRRP can be a physical Ethernet link (10 Gigabit Ethernet, Gigabit Ethernet, 40 Gigabit Ethernet), port-channel, or VE. For VRRP-E, the Ethernet link interface type must be VE.

Enabling preemption for physical Ethernet or port-channel

To enable preemption for a physical or port-channel router interface.

The following example enables preemption.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3502" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>101/1/6</name>
                    <vrrp xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                        <vrid>1</vrid>
                        <preempt-mode/>
                    </vrrp>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3502" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling preemption for a VE interface

To enable preemption for a VE interface.

The following example enables preemption.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3502" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <ve>
                        <name>5</name>
                        <vrrpe xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                            <vrid>1</vrid>
                            <preempt-mode/>
                        </vrrpe>
                    </ve>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3502" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <ok/>
</rpc-reply>
```

Configuring the track priority

The track priority is a priority that adjusts an Ethernet link interface when the physical uplink port that it is tracking fails. In this way, a lower track priority can force a different router to take over as master should the tracked uplink port fail.

For additional conceptual information about track ports and track priorities, refer to the *Network OS Layer 3 Routing Configuration Guide*.

The procedure for configuring track priority differs depending on the Ethernet link interface type, which for VRRP can be a physical Ethernet link (10 Gigabit Ethernet, Gigabit Ethernet, 40 Gigabit Ethernet), port-channel, or VE. For VRRP-E, the Ethernet link interface type must be VE.

Configuring track priority for physical Ethernet or port-channel

To configure the track priority for a physical Ethernet link or port-channel.

The following example reduces the priority of interface 101/1/6 by 60 if interface 2/4 goes down.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3503" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>101/1/6</name>
                    <vrrp xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                        <vrid>1</vrid>
                        <track>
                            <interface>
                                <interface-type>tengigabitethernet</interface-type>
                                <interface-name>2/4</interface-name>
                                <track-priority>60</track-priority>
                            </interface>
                        </track>
                    </vrrp>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3503" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring track priority for a VE link interface

To configure the track priority for a VE interface.

The following example reduces the priority of interface 101/1/6 by 60 if physical interface 2/4 goes down.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3503" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
      <target>
         <running/>
      </target>
      <config>
         <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
            <interface>
               <ve>
                  <name>6</name>
                  <vrrpe xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                     <vrid>1</vrid>
                     <track>
                        <interface>
                           <interface-type>tengigabitethernet
                            </interface-type>
                           <interface-name>2/4</interface-name>
                           <track-priority>60</track-priority>
                        </interface>
                     </track>
                  </vrrpe>
               </ve>
            </interface>
         </interface-vlan>
      </config>
   </edit-config>
</rpc>

<rpc-reply message-id="3503" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Enabling short-path forwarding (VRRP-E only)

For conceptual information about short-path forwarding, refer to the *Network OS Layer 3 Routing Configuration Guide*.

To enable short-path forwarding.

The following example enables short-path forwarding on VE interface 5.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3504" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
      <target>
         <running/>
      </target>
      <config>
         <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
            <interface>
               <ve>
                  <name>5</name>
                  <vrrpe xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                     <vrid>100</vrid>
```

```
                        <short-path-forwarding/>
                    </vrrpe>
                </ve>
            </interface>
        </interface-vlan>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="3504" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Configuring a multigroup virtual router cluster

Figure 5 shows a commonly employed virtual router setup. This setup introduces redundancy by configuring two virtual router groups. The first group has Router 1 as the master and Router 2 as the backup. The second group has Router 2 as the master and Router 1 as the backup. This type of configuration is sometimes called Multigroup VRRP.

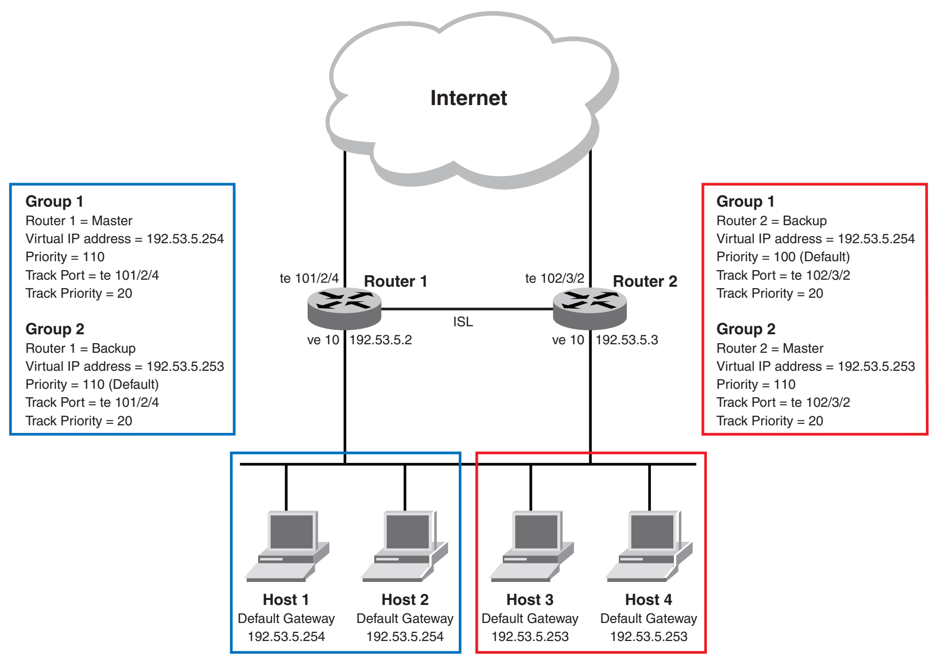


FIGURE 5 Dual redundant network access

In this example, Router 1 and Router 2 use VRRP-E to load share as well as provide redundancy to the hosts. The load sharing is accomplished by creating two VRRP-E groups. Each group has its own virtual IP address. Half of the clients point to Group 1's virtual IP address as their default gateway and the other half point to Group 2's virtual IP address as their default gateway. This arrangement will enable some of the outbound Internet traffic to go through Router 1 and the rest to go through Router 2.

NOTE
Load sharing is supported by VRRP as well as VRRP-E.

Router 1 is the master for Group 1 (master priority = 110) and Router 2 is the backup for Group 1 (backup priority = 100). Router 1 and Router 2 both track the uplinks to the Internet. If an uplink failure occurs on Router 1, its backup priority is decremented by 20 (track priority = 90), so that all traffic destined to the Internet is sent through Router 2 instead.

Similarly, Router 2 is the master for Group 2 (master priority = 110) and Router 1 is the backup for Group 2 (backup priority = 100). Router 1 and Router 2 are both tracking the uplinks to the Internet. If an uplink failure occurs on Router 2, its backup priority is decremented by 20 (track priority = 90), so that all traffic destined to the internet is sent through Router 1 instead.

To implement the configuration shown in Figure 5 on page 509, configure one VRRP-E router to act as a master in the first virtual router group and a backup in the second virtual group, and then configure the second VRRP-E router to act as a backup in the first virtual group and master in the second virtual group.

NOTE
The procedures assume VRRP-E.

Configuring Router 1 as master for first virtual router group

The following example <edit-config> RPC configures Router 1 as the master router for the first router group. Make sure VCS Fabric mode is enabled, and then perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3505" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
            <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>101</rbridge-id>
                <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <hide-vrrp-holder xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                        <vrrp/>
                    </hide-vrrp-holder>
                </protocol>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>10</name>
                        <ip>
                            <ip-config
                             xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                                <address>
                                    <address>192.53.5.2/24</address>
                                </address>
                            </ip-config>
                        </ip>
                        <vrrpe xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                            <vrid>1</vrid>
                            <virtual-ip>
                                <virtual-ipaddr>192.53.5.1</virtual-ipaddr>
```

```
                        </virtual-ip>
                        <track>
                            <interface>
                                <interface-type>tengigabitethernet
                                 </interface-type>
                                <interface-name>2/4</interface-name>
                                <track-priority>20</track-priority>
                            </interface>
                        </track>
                        <priority>110</priority>
                    </vrrpe>
                </ve>
            </interface>
        </rbridge-id>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="3505" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring Router 1 as backup for second virtual router group

The following example <edit-config> RPC configures Router 1 as a backup for the second router group. Make sure VCS Fabric mode is enabled, and then perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3506" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
            <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>101</rbridge-id>
                <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <hide-vrrp-holder xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                        <vrrp/>
                    </hide-vrrp-holder>
                </protocol>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>5</name>
                        <vrrpe xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                            <vrid>2</vrid>
                            <virtual-ip>
                                <virtual-ipaddr>192.53.5.253</virtual-ipaddr>
                            </virtual-ip>
                            <track>
                                <interface>
                                    <interface-type>tengigabitethernet
                                     </interface-type>
                                    <interface-name>2/4</interface-name>
                                    <track-priority>20</track-priority>
                                </interface>
                            </track>
```

```
                    </vrrpe>
                </ve>
            </interface>
        </rbridge-id>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="3506" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring Router 2 as backup for first virtual router group

The following example <edit-config> RPC configures Router 2 as the backup router for the first router group. Ensure that VCS Fabric mode is enabled, and then perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3507" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
            <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>102</rbridge-id>
                <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <hide-vrrp-holder xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                        <vrrp/>
                    </hide-vrrp-holder>
                </protocol>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>15</name>
                        <ip>
                            <ip-config
                             xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                                <address>
                                    <address>192.53.5.3/24</address>
                                </address>
                            </ip-config>
                        </ip>
                        <vrrpe xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                            <vrid>1</vrid>
                            <virtual-ip>
                                <virtual-ipaddr>192.53.5.254</virtual-ipaddr>
                            </virtual-ip>
                            <track>
                                <interface>
                                    <interface-type>tengigabitethernet
                                     </interface-type>
                                    <interface-name>3/2</interface-name>
                                    <track-priority>20</track-priority>
                                </interface>
                            </track>
                        </vrrpe>
                    </ve>
```

```
            </interface>
        </rbridge-id>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="3507" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Configuring Router 2 as master for second virtual router group

The following example <edit-config> RPC configures Router2 as the master for the second router group. Ensure that VCS Fabric mode is enabled, and then perform the following steps.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3508" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
      <config>
      <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
        <rbridge-id>102</rbridge-id>
        <protocol xmlns="urn:brocade.com:mgmt:brocade-interface">
          <hide-vrrp-holder xmlns="urn:brocade.com:mgmt:brocade-vrrp">
            <vrrp/>
          </hide-vrrp-holder>
        </protocol>
        <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
          <ve>
            <name>15</name>
            <vrrpe xmlns="urn:brocade.com:mgmt:brocade-vrrp">
              <vrid>2</vrid>
              <virtual-ip>
                <virtual-ipaddr>192.53.5.253</virtual-ipaddr>
              </virtual-ip>
              <track>
                <interface>
                  <interface-type>tengigabitethernet
                   </interface-type>
                  <interface-name>3/2</interface-name>
                  <track-priority>20</track-priority>
                </interface>
              </track>
              <priority>110</priority>
            </vrrpe>
          </ve>
        </interface>
      </rbridge-id>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="2508" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Verifying VRRP and VRRP-E configuration

To obtain configuration information about VRRP or VRRP-E for a specific interface, issue the <get-config> RPC with a subtree filter to limit the output to VRRP information, VRRP-E information, or information about a specific VRRP or VRRP-E group.

The following example uses a subtree filter to return information about VRRP group 1. To return information about all VRRP groups configured on this interface, remove the <vrid> element. To request configuration information about VRRP-E, replace the <vrrp> node with the <vrrpe> node.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3509" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <get-config>
        <source>
            <running/>
        </source>
        <filter type="subtree">
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>101/1/6</name>
                    <vrrp xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                        <vrid>2</vrid>
                    </vrrp>
                </tengigabitethernet>
            </interface>
        </filter>
    </get-config>
</rpc>

<rpc-reply message-id="3509" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
        <tengigabitethernet>
            <name>102/5/1</name>
            <vrrp xmlns="urn:brocade.com:mgmt:brocade-vrrp">
                <priority>110</priority>
                <vrid>2</vrid>
                <virtual-ip>
                    <virtual-ipaddr>192.53.5.253</virtual-ipaddr>
                </virtual-ip>
                <track>
                    <interface>
                        <interface-type>tengigabitethernet</interface-type>
                        <interface-name>3/2</interface-name>
                        <track-priority>20</track-priority>
                    </interface>
                </track>
            </vrrp>
        </tengigabitethernet>
    </interface>
</rpc-reply>
```

Chapter

Configuring VRRPv3

49

In this chapter

VRRPv3 configuration with NETCONF overview

This chapter provides procedures for configuring the Virtual Router Redundancy Protocol version 3 (VRRPv3) using the NETCONF interface. Refer to the *Network OS Layer 3 Routing Configuration Guide* for the following related information:

- A conceptual overview of VRRP
- General guidelines
- An overview of VRRPv3 and VRRP-E-v3 packet behavior
- Procedures for configuring VRRPv3 using the command line interface

Through the NETCONF interface, you can perform the following operations on VRRPv3 and VRRP-E-v3:

- Use the <edit-config> RPC to configure VRRPv3 and VRRP-E-v3.
- Use the <get-config> RPC to validate configuration settings.

VRRPv3 parameters are defined in the brocade-vrrpv3 YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Enabling IPv6 VRRP3

Configures a Layer3 device as a member of a VRRPv3 group and enables IPv6 VRRPv3.

In this task, you can configure IPv6 VRRP version 3 on a virtual ethernet interface, assign a VRRPv3 group to the device, and enable the VRRPv3 session using a virtual IP address. The device must be a router or another device that supports Layer 3 routing.

```
<?xml version="1.0" ?>
```

```
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>195</rbridge-id>
                <ipv6>
                    <proto-vrrpv3 xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                        <vrrp></vrrp>
                    </proto-vrrpv3>
                </ipv6>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>4</name>
                        <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <ipv6-config>
                                <address>
                                    <ipv6-address>
                                        <address>2001:2018:8192::125/64</address>
                                    </ipv6-address>
                                </address>
                            </ipv6-config>
                            <vrrpv3-group
xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                                <vrid>18</vrid>
                                <virtual-ip>

<virtual-ipaddr>2001:2018:8192::1</virtual-ipaddr>
                                </virtual-ip>
                                <virtual-ip>
                                    <virtual-ipaddr>fe80::2018:1</virtual-ipaddr>
                                </virtual-ip>
                            </vrrpv3-group>
                        </ipv6>
                    </ve>
                </interface>
            </rbridge>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Enabling IPv6 VRRP-E-v3

Configures a Layer3 device as a member of a VRRPv3 extended group and enables IPv6 VRRP-E-v3.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
```

```
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>195</rbridge-id>
                <ipv6>
                    <proto-vrrpv3 xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                        <vrrp-extended></vrrp-extended>
                    </proto-vrrpv3>
                </ipv6>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>4</name>
                        <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <ipv6-config>
                                <address>
                                    <ipv6-address>
                                        <address>2001:2018:8192::122/64</address>
                                    </ipv6-address>
                                </address>
                            </ipv6-config>
                            <vrrpv3-group
xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                                <vrid>19</vrid>
                                <virtual-ip>

<virtual-ipaddr>2001:2018:8192::2</virtual-ipaddr>
                                </virtual-ip>
                                <virtual-ip>
                                    <virtual-ipaddr>fe80::2018:1</virtual-ipaddr>
                                </virtual-ip>
                            </vrrpv3-group>
                        </ipv6>
                    </ve>
                </interface>
            </rbridge>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Port tracking using IPv6 VRRPv3

Enables the tracking of the link status of an interface not configured for VRRP or VRRP-E and configures a priority that can result in dynamic changes to the VRRP device priority.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>195</rbridge-id>
                <ipv6>
```

```
                    <proto-vrrpv3 xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                        <vrrp-extended></vrrp-extended>
                    </proto-vrrpv3>
                </ipv6>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>4</name>
                        <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <ipv6-config>
                                <address>
                                    <ipv6-address>
                                        <address>2001:2018:8192::120/64</address>
                                    </ipv6-address>
                                </address>
                            </ipv6-config>
                            <vrrpv3e-group xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                                <vrid>20</vrid>
                                <virtual-ip>

<virtual-ipaddr>2001:2018:8192::3</virtual-ipaddr>
                                </virtual-ip>
                                <virtual-ip>
                                    <virtual-ipaddr>fe80::2018:5</virtual-ipaddr>
                                </virtual-ip>
                                <track>
                                    <interface>

<interface-type>tengigabitethernet</interface-type>
                                        <interface-name>195/1/1</interface-name>
                                        <track-priority>15</track-priority>
                                    </interface>
                                </track>
                                <enable></enable>
                                <priority>120</priority>
                            </vrrpv3e-group>
                        </ipv6>
                    </ve>
                </interface>
            </rbridge>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring VRRP hold timer support

A hold timer can be configured on a VRRP-enabled interface to set an interval, in seconds, before a backup device becomes the master VRRP device.

To configure a hold timer, VRRP must be enabled on the device.

This task is supported in both versions of VRRP and VRRP-E, but the configuration below is for VRRPv3.

```
<?xml version="1.0" ?>
```

```
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>20</rbridge-id>
                    <ipv6>
                        <proto-vrrpv3 xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                            <vrrp></vrrp>
                        </proto-vrrpv3>
                    </ipv6>
                    <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                        <ve>
                            <name>50</name>
                            <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                                <ipv6-config>
                                    <address>
                                        <ipv6-address>
                                            <address>2001:2018:8192::122/64</address>
                                </ipv6-address>
                                    </address>
                                </ipv6-config>
                                <vrrpv3-group
xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                                    <vrid>30</vrid>
                                    <virtual-ip>

<virtual-ipaddr>2001:2018:8192::1</virtual-ipaddr>
                                    </virtual-ip>
                                    <virtual-ip>
                                        <virtual-ipaddr>fe80::2018:1</virtual-ipaddr>
                                    </virtual-ip>
                                    <enable></enable>
                                    <hold-time>5</hold-time>
                                    <preempt-mode></preempt-mode>
                                    <description>Product Marketing
group</description>
                                </vrrpv3-group>
                            </ipv6>
                        </ve>
                    </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring VRRP-Ev3 load-balancing in VCS mode

VRRP-Ev3 traffic can be load-balanced using short-path forwarding on the backup devices. Short-path forwarding is only supported in VCS mode.

Before configuring VRRP-Ev3 load-balancing, VRRP-Ev3 must be configured on all devices in the VRRP-Ev3 session.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>10</rbridge-id>
                <ipv6>
                    <proto-vrrpv3 xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                        <vrrp-extended></vrrp-extended>
                    </proto-vrrpv3>
                </ipv6>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>20</name>
                        <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <ipv6-config>
                                <address>
                                    <ipv6-address>
                                        <address>2001:2019:8192::122/64</address>
                                    </ipv6-address>
                                </address>
                            </ipv6-config>
                            <vrrpv3e-group
xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                                <vrid>80</vrid>
                                <short-path-forwarding>
                                    <revert-priority>50</revert-priority>
                                </short-path-forwarding>
                            </vrrpv3e-group>
                        </ipv6>
                    </ve>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring sub-second failover using VRRP-Ev3

Configuring a scale factor making the interval between VRRP advertisements to be set in milliseconds allows a sub-second convergence time if a master VRRP device fails.

The configuring sub-second failover using VRRP-Ev3 task is only supported by VRRP-Ev3.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
```

```
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>10</rbridge-id>
                <ipv6>
                    <proto-vrrpv3 xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                        <vrrp-extended></vrrp-extended>
                    </proto-vrrpv3>
                </ipv6>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>20</name>
                        <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <ipv6-config>
                                <address>
                                    <ipv6-address>
                                        <address>2001:2019:8192::122/64</address>
                                    </ipv6-address>
                                </address>
                            </ipv6-config>
                            <vrrpv3e-group
xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                                <vrid>80</vrid>

<advertisement-interval-scale>10</advertisement-interval-scale>

<vrrpe-advertisement-interval>1</vrrpe-advertisement-interval>
                            </vrrpv3e-group>
                        </ipv6>
                    </ve>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Disabling VRRP-Ev3 router advertisements

The ability to suppress VRRP-Ev3 master device interface router advertisements is introduced.

```
<?xml version="1.0" ?>
<rpc nc:message-id="10" xmlns=”urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>10</rbridge-id>
                <ipv6>
                    <proto-vrrpv3 xmlns="urn:brocade.com:mgmt:brocade-vrrpv3">
                        <vrrp-extended></vrrp-extended>
                    </proto-vrrpv3>
```

```
                </ipv6>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>20</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>

<vrrp-suppress-interface-ra></vrrp-suppress-interface-ra>
                                </ipv6-intf-cmds>
                            </ipv6-nd-ra>
                        </ipv6>
                    </ve>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Enabling the v2 checksum computation method

Enabling the v2 checksum computation method in a VRRPv3 IPv4 session

The following example shows how to enable v2 checksum computation method in a VRRPv3 IPv4 session.

```
<?xml version="1.0" ?>
<rpc nc:message-id="10" xmlns=”urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>10</rbridge-id>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>20</name>
                        <vrrp-group xmlns="urn:brocade.com:mgmt:brocade-vrrp"
y:self="/rest/config/running/rbridge-id/1/interface/Ve/1/vrrp-group/10%2C3">
                            <vrid>10</vrid>
                            <version>3</version>
                            <use-v2-checksum>true</use-v2-checksum>
                            <track
y:self="/rest/config/running/rbridge-id/1/interface/Ve/1/vrrp-group/10%2C3/track"
/>
                            <advertisement-interval>1000</advertisement-interval>
                            <preempt-mode>true</preempt-mode>
                        </vrrp-group>
                    </ve>
                </interface>
            </rbridge-id>
```

```
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Chapter

Configuring Bidirectional Forwarding Detection (BFD) 50

In this chapter

BFD configuration with NETCONF overview

This chapter provides procedures for configuring the Bidirectional Forwarding Detection (BFD) using the NETCONF interface. Refer to the *Network OS Layer 3 Routing Configuration Guide* for the following related information:

- A conceptual overview of BFD
- General guidelines
- Procedures for configuring BFD using the command line interface

Through the NETCONF interface, you can perform the following operations on BFD:

- Use the <edit-config> RPC to configure BFD.
- Use the <get-config> RPC to validate configuration settings.

BFD parameters are defined in the brocade-bfd YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Configuring BFD on an interface

The following example shows how to configure BFD on an interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
```

```
                <bfd>
                    <interval>
                        <min-tx>110</min-tx>
                        <min-rx>120</min-rx>
                        <multiplier>15</multiplier>
                    </interval>
                </bfd>
            </tengigabitethernet>
        </interface>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Disabling BFD on an interface

The following example shows how to disable BFD on an interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <bfd>
                        <bfd-shutdown></bfd-shutdown>
                    </bfd>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

BFD for BGP

BFD is supported for BGP and is disabled by default.

Configuring BFD session parameters for BGP

The following example shows hot to configure BFD session parameters for BGP enabled interfaces.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
```

```
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <bfd>
                                <holdover-interval>15</holdover-interval>
                                <interval>
                                    <min-tx>110</min-tx>
                                    <min-rx>120</min-rx>
                                    <multiplier>15</multiplier>
                                </interval>
                            </bfd>
                        </router-bgp-attributes>
                    </router-bgp>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling BFD sessions for a specified BGP neighbor

The following example shows how to enable BFD sessions for a specified BGP neighbor.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <neighbor>
                                <neighbor-ips>
                                    <neighbor-addr>

<router-bgp-neighbor-address>10.10.1.1</router-bgp-neighbor-address>
                                        <remote-as>3500</remote-as>
                                        <bfd>
                                            <holdover-interval>12</holdover-interval>
                                            <interval>
                                                <min-tx>120</min-tx>
                                                <min-rx>140</min-rx>
                                                <multiplier>10</multiplier>
                                            </interval>
                                        </bfd>
```

```
                                </neighbor-addr>
                            </neighbor-ips>
                        </neighbor>
                    </router-bgp-attributes>
                </router-bgp>
            </router>
        </rbridge-id>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling BFD sessions for a specified BGP neighbor in a nondefault VRF

The following example shows how to enable BFD sessions for a specified BGP neighbor in a nondefault VRF.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <address-family>
                            <ipv4>
                                <ipv4-unicast>
                                    <af-vrf>
                                        <af-vrf-name>red</af-vrf-name>
                                        <neighbor>
                                            <af-ipv4-vrf-neighbor-address-holder>
                                                <af-ipv4-neighbor-addr>

<af-ipv4-neighbor-address>10.10.1.1</af-ipv4-neighbor-address>
                                                    <remote-as>3000</remote-as>
                                                    <bfd>
                                                        <bfd-enable></bfd-enable>

<holdover-interval>12</holdover-interval>
                                                        <interval>
                                                            <min-tx>120</min-tx>
                                                            <min-rx>140</min-rx>
                                                            <multiplier>10</multiplier>
                                                        </interval>
                                                    </bfd>
                                                </af-ipv4-neighbor-addr>
                                            </af-ipv4-vrf-neighbor-address-holder>
                                        </neighbor>
                                    </af-vrf>
                                </ipv4-unicast>
```

```
                    </ipv4>
                </address-family>
            </router-bgp>
        </router>
    </rbridge-id>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Enabling BFD sessions for a specified BGP peer group

The following example shows how to enable BFD sessions for a specified BGP peer group.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <neighbor>
                                <peer-grps>
                                    <neighbor-peer-grp>

<router-bgp-neighbor-peer-grp>pg1</router-bgp-neighbor-peer-grp>
                                        <bfd>
                                            <bfd-enable></bfd-enable>
                                            <holdover-interval>17</holdover-interval>
                                            <interval>
                                                <min-tx>200</min-tx>
                                                <min-rx>220</min-rx>
                                                <multiplier>25</multiplier>
                                            </interval>
                                        </bfd>
                                    </neighbor-peer-grp>
                                </peer-grps>
                            </neighbor>
                        </router-bgp-attributes>
                    </router-bgp>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling BFD sessions for a specified BGP peer group in a nondefault VRF

The following example shows how to enable BFD sessions for a specified BGP peer group in a non default VRF.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <af-ipv6-vrf>
                                        <af-ipv6-vrf-name>green</af-ipv6-vrf-name>
                                        <neighbor>
                                            <af-ipv6-vrf-neighbor-address-holder>
                                                <af-ipv6-neighbor-addr>

<af-ipv6-neighbor-address>2001:284d::21:22</af-ipv6-neighbor-address>
                                                    <remote-as>2500</remote-as>
                                                    <bfd>

<holdover-interval>18</holdover-interval>
                                                        <interval>
                                                            <min-tx>145</min-tx>
                                                            <min-rx>155</min-rx>
                                                            <multiplier>15</multiplier>
                                                        </interval>
                                                    </bfd>
                                                </af-ipv6-neighbor-addr>
                                            </af-ipv6-vrf-neighbor-address-holder>
                                        </neighbor>
                                    </af-ipv6-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

BFD for OSPF

BFD is supported for OSPF and is disabled by default.

Enabling BFD on a specified OSPFv2

The following example shows how to enable BFD on a specified OSPFv2.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <ip>
                        <ospf>
                            <bfd></bfd>
                        </ospf>
                    </ip>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring BFD globally on OSPFv2

The following example shows how to configure BFD globally on OSPFv2.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf">
                        <vrf>default-vrf</vrf>
                        <global-bfd>
                            <bfd-enable></bfd-enable>
                            <holdover-interval>12</holdover-interval>
                        </global-bfd>
                    </ospf>
                </router>
            </rbridge-id>
        </config>
```

```
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring BFD globally on OSPFv2 in a nondefault VRF instance

The following example shows how to configure BFD globally on OSPFv2 in a nondefault VRF.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf">
                        <vrf>green</vrf>
                        <global-bfd>
                            <bfd-enable></bfd-enable>
                            <holdover-interval>12</holdover-interval>
                        </global-bfd>
                    </ospf>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling BFD on a specified OSPFv3

The following example shows how to enable BFD on a specified OSPFv3.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <ipv6>
                        <ospf>
                            <bfd></bfd>
                        </ospf>
```

```
                </ipv6>
            </tengigabitethernet>
        </interface>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring BFD globally on OSPFv3

The following example shows how to configure BFD globally on ODPFv3.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ipv6>
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>default-vrf</vrf>
                            <bfd>
                                <bfd-enable></bfd-enable>
                                <bfd-holdover-interval>20</bfd-holdover-interval>
                            </bfd>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring BFD globally on OSPFv3 in a nondefault VRF

The following example shows how to configure BFD globally on OSPFv3 in a nondefault VRF.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ipv6>
```

```
                    <router>
                        <ospf xmlns="urn:brocade.com:mgmt:brocade-ospfv3">
                            <vrf>green</vrf>
                            <nonstop-routing></nonstop-routing>
                            <bfd>
                                <bfd-enable></bfd-enable>
                                <bfd-holdover-interval>18</bfd-holdover-interval>
                            </bfd>
                        </ospf>
                    </router>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

BFD for VXLAN extension tunnels

VXLAN extension tunnels facilitate a Layer 3 overlay to extend Layer 2 VLANs across VCS clusters. Two different VCS clusters can be connected by a VXLAN tunnel. A VXLAN tunnel can have its end points in either VCS, with the possibility of multiple Layer 3 ECMP paths between the end points. BFD has the ability to monitor the tunnel for reachability and quick fault detection.

Configuring BFD on a VXLAN extension tunnel

The following example shows how to configure BFD on a VXLAN extension tunnel.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <overlay-gateway xmlns="urn:brocade.com:mgmt:brocade-tunnels">
                <name>og1</name>
                <gw-type>layer2-extension</gw-type>
                <ip>
                    <interface>
                        <loopback>22</loopback>
                    </interface>
                </ip>
                <site>
                    <name>site1</name>
                    <tunnel-dst>
                        <address>1.1.1.1</address>
                    </tunnel-dst>
                    <bfd-enable></bfd-enable>
                    <bfd>
                        <params>
                            <interval>
                                <min-tx>2000</min-tx>
```

```
                                <min-rx>3000</min-rx>
                                <multiplier>26</multiplier>
                            </interval>
                        </params>
                    </bfd>
                </site>
            </overlay-gateway>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

BFD for Static Routes

BFD for Static Routes allows you to detect failures that impact the forwarding path of a static route. This feature supports both single hop and multihop BFD Static Routes for both IPv4 and IPv6. Unless the BFD session is up, the gateway for the static route is considered unreachable, and the affected routes are not installed in the routing table (RIBMGR). BFD can remove the associated static route from the RIBMGR if the next-hop becomes unreachable indicating that the BFD session has gone down.

Configuring BFD on an IP static route

The following example shows how to configure BFD on an IP static route.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ip>
                    <rtm-config xmlns="urn:brocade.com:mgmt:brocade-rtm">
                        <route>
                            <static>
                                <bfd>
                                    <bfd-static-route>

<bfd-static-route-dest>10.0.2.1</bfd-static-route-dest>

<bfd-static-route-src>10.1.1.1</bfd-static-route-src>
                                        <bfd-interval-attributes>
                                            <interval>500</interval>
                                            <min-rx>500</min-rx>
                                            <multiplier>5</multiplier>
                                        </bfd-interval-attributes>
                                    </bfd-static-route>
                                    <holdover-interval>15</holdover-interval>
                                </bfd>
                            </static>
```

```
                    </route>
                </rtm-config>
            </ip>
        </rbridge-id>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring BFD on an IP static route in a nondefault VRF

The following example shows how to configure BFD on an IP static route in a nondefault VRF.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>green</vrf-name>
                    <address-family>
                        <ip>
                            <unicast>
                                <ip xmlns="urn:brocade.com:mgmt:brocade-rtm">
                                    <route>
                                        <static>
                                            <bfd>
                                                <bfd-static-route>

<bfd-static-route-dest>10.0.0.1</bfd-static-route-dest>

<bfd-static-route-src>10.1.1.2</bfd-static-route-src>
                                                    <bfd-interval-attributes>
                                                        <interval>500</interval>
                                                        <min-rx>500</min-rx>
                                                        <multiplier>5</multiplier>
                                                    </bfd-interval-attributes>
                                                </bfd-static-route>
                                            </bfd>
                                        </static>
                                    </route>
                                </ip>
                            </unicast>
                        </ip>
                    </address-family>
                </vrf>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <ok/>
</rpc-reply>
```

Configuring BFD on an IPv6 static route

The following example shows how to configure BFD on an IPv6 static route.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ipv6>
                    <route xmlns="urn:brocade.com:mgmt:brocade-ipv6-rtm">
                        <static>
                            <bfd>
                                <bfd-ipv6-link-local-static-route>

<bfd-ipv6-link-local-dest>fe80::a</bfd-ipv6-link-local-dest>

<bfd-ipv6-link-local-src>fe80::b</bfd-ipv6-link-local-src>
                                    <bfd-interface-type>ve</bfd-interface-type>
                                    <bfd-interface-name>100</bfd-interface-name>
                                    <bfd-ipv6-interval-attributes>
                                        <interval>100</interval>
                                        <min-rx>100</min-rx>
                                        <multiplier>10</multiplier>
                                    </bfd-ipv6-interval-attributes>
                                </bfd-ipv6-link-local-static-route>
                                <ipv6-holdover-interval>25</ipv6-holdover-interval>
                            </bfd>
                        </static>
                    </route>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring BFD on an IPv6 static route in a nondefault VRF

The following example shows how to configure BFD on an IPv6 static route in a nondefault VRF.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
```

```
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>green</vrf-name>
                    <address-family>
                        <ipv6>
                            <unicast>
                                <ipv6 xmlns="urn:brocade.com:mgmt:brocade-rtm">
                                    <route>
                                        <static>
                                            <bfd>
                                                <bfd-static-route>

<bfd-static-route-dest>fe70::a</bfd-static-route-dest>

<bfd-static-route-src>fe70::b</bfd-static-route-src>
                                                    <ve>10</ve>
                                                    <bfd-interval-attributes>
                                                        <interval>1000</interval>
                                                        <min-rx>2000</min-rx>
                                                        <multiplier>20</multiplier>
                                                    </bfd-interval-attributes>
                                                </bfd-static-route>
                                            </bfd>
                                        </static>
                                    </route>
                                </ipv6>
                            </unicast>
                        </ipv6>
                    </address-family>
                </vrf>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Configuring Fabric-Virtual-Gateway

51

In this chapter

Fabric-Virtual-Gateway overview

This chapter provides procedures for configuring the Fabric-Virtual-Gateway. Refer to the *Network OS Layer 3 Configuration Guide* for the following related information:

- A conceptual overview of Fabric-Virtual-Gateway
- General guidelines
- Procedures for configuring Fabric-Virtual-Gateway using the command line interface

Through the NETCONF interface, you can perform the following operations on Fabric-Virtual-Gateway:

- Use the <edit-config> RPC to configure Fabric-Virtual-Gateway.
- Use the <get-config> RPC to validate configuration settings.

Fabric-Virtual-Gateway parameters are defined in the brocade-anycast-gateway YANG module. For details, refer to the *Network OS YANG Reference Manual*

Configuring Fabric-Virtual-Gateway globally

The following examples shows how to enable Fabric-Virtual-Gateway globally.

Example 1 - Enabling IP Fabric-Virtual-Gateway configuration.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <router xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <fabric-virtual-gateway
xmlns="urn:brocade.com:mgmt:brocade-anycast-gateway">
                    <address-family>
                        <ipv4>
                            <enable_global></enable_global>
```

```
                            <gratuitous-arp>
                                <timer>60</timer>
                            </gratuitous-arp>

<gateway-mac-address>0011.1122.2233</gateway-mac-address>

<accept-unicast-arp-request></accept-unicast-arp-request>
                        </ipv4>
                    </address-family>
                </fabric-virtual-gateway>
            </router>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Example 2 - Enabling IPv6 Fabric-Virtual-Gateway configuration.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <router xmlns="urn:brocade.com:mgmt:brocade-common-def">
                <fabric-virtual-gateway
xmlns="urn:brocade.com:mgmt:brocade-anycast-gateway">
                    <address-family>
                        <ipv6>
                            <enable_global></enable_global>
                            <gratuitous-arp>
                                <timer>55</timer>
                            </gratuitous-arp>

<gateway-mac-address>02e2.5e00.0012</gateway-mac-address>
                        </ipv6>
                    </address-family>
                </fabric-virtual-gateway>
            </router>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>

</rpc-reply>
```

Configuring Fabric-Virtual-Gateway under global interface VE

The following examples shows how to configure Fabric-Virtual-Gateway under global interface VE.

Example 1 - IP Fabric-Virtual-Gateway configuration in VE interface.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <ve>
                        <gve-name>1</gve-name>
                        <ip>
                            <ip-anycast-gateway
xmlns="urn:brocade.com:mgmt:brocade-anycast-gateway">
                                <ip-gw-id>1</ip-gw-id>

<ipv4-gateway-address>1.1.1.1/24</ipv4-gateway-address>
                                <gratuitous-arp>
                                    <gve-timer>40</gve-timer>
                                </gratuitous-arp>
                                <hold-time>25</hold-time>
                                <load-balancing-disable></load-balancing-disable>
                                <enable></enable>
                                <description>anycastip</description>
                            </ip-anycast-gateway>
                        </ip>
                    </ve>
                </interface>
            </interface-vlan>
         </config>
     </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Example 2 - IPv6 Fabric-Virtual-Gateway configuration in VE interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <ve>
                        <gve-name>1</gve-name>
                        <ipv6>
                            <ipv6-anycast-gateway
xmlns="urn:brocade.com:mgmt:brocade-anycast-gateway">
                                <ipv6-gw-id>1</ipv6-gw-id>
                                <ipv6-gateway-address>
                                    <ipv6-gw-addr>2001:384d::25:24/1</ipv6-gw-addr>
                                </ipv6-gateway-address>
                                <gratuitous-arp>
                                    <gve-timer>55</gve-timer>
                                </gratuitous-arp>
```

```
                                <hold-time>25</hold-time>
                                <load-balancing-disable></load-balancing-disable>
                                <enable></enable>
                                <description>anycastipv6</description>
                            </ipv6-anycast-gateway>
                        </ipv6>
                    </ve>
                </interface>
            </interface-vlan>
         </config>
     </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring Fabric-Virtual-Gateway under RBridge-level VE interface

The following example shows how to configure Fabric-Virtual-Gateway under RBridge-level VE interface.

Example 1 - IP Fabric-Virtual-Gateway configuration under RBridge VE interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>1</name>
                        <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <ip-local-anycast-gateway
xmlns="urn:brocade.com:mgmt:brocade-anycast-gateway">
                                <local-ip-gw-id>1</local-ip-gw-id>
                                <enable_local></enable_local>
                                <load-balancing>
                                    <threshold-priority>100</threshold-priority>
                                </load-balancing>
                                <track>
                                    <interface>

<interface-type>tengigabitethernet</interface-type>
                                        <interface-name>1/0/5</interface-name>
                                        <interface-priority>25</interface-priority>
                                    </interface>
                                    <network>
                                        <network-address>1.1.1.1/24</network-address>
                                        <network-priority>26</network-priority>
                                    </network>
```

```
                                <next-hop>
                                    <next-hop-address>1.1.1.1</next-hop-address>
                                    <next-hop-priority>28</next-hop-priority>
                                </next-hop>
                            </track>
                        </ip-local-anycast-gateway>
                    </ip>
                </ve>
            </interface>
        </rbridge-id>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Example 2- IPv6 Fabric-Virtual-Gateway configuration under RBridge VE interface.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>1</name>
                        <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <ipv6-local-anycast-gateway
xmlns="urn:brocade.com:mgmt:brocade-anycast-gateway">
                                <local-ipv6-gw-id>1</local-ipv6-gw-id>
                                <enable_local></enable_local>
                                <load-balancing>
                                    <threshold-priority>30</threshold-priority>
                                </load-balancing>
                                <ipv6-track>
                                    <ipv6-interface>

<ipv6-interface-type>fortygigabitethernet</ipv6-interface-type>

<ipv6-interface-name>1/0/50</ipv6-interface-name>

<ipv6-interface-priority>25</ipv6-interface-priority>
                                    </ipv6-interface>
                                    <ipv6-network>

<ipv6-network-address>1::/64</ipv6-network-address>

<ipv6-network-priority>10</ipv6-network-priority>
                                    </ipv6-network>
                                    <ipv6-next-hop>

<ipv6-next-hop-address>2001::2</ipv6-next-hop-address>
```

```
<ipv6-next-hop-priority>28</ipv6-next-hop-priority>
                                </ipv6-next-hop>
                            </ipv6-track>
                        </ipv6-local-anycast-gateway>
                    </ipv6>
                </ve>
            </interface>
        </rbridge-id>
     </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Chapter

Configuring VRF

52

In this chapter

VRF configuration with NETCONF overview

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. Refer to the *Network OS Layer 3 Routing Configuration Guide* for the following related information:

- A conceptual overview of VRF
- An example of a VRF configuration
- A description of VRF-lite for customer edge routers
- Procedures and examples for configuring VRF using the Network OS command line interface
- Procedures and examples for configuring Inter-VRF Route Leaking using the Network OS command line interface
- A procedure for enabling VRF for VRRP

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 brocade-vrf YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all VRF parameters, refer to the brocade-vrf.yang file.

Configuring VRF

Typical full-blown implementations of VRFs are designed to support BGP/MPLS VPNs, whereas VRF-lite implementations typically are much simpler with moderate scalability (as compared to BGP/MPLS VPNs). These two flavors share a lot in common but differ in the interconnect schemes, routing protocols used over the interconnect, and also in the VRF classification mechanisms. All references to VRF in this document implicitly indicate VRF-lite. Figure 1 shows a typical single VCS comprising Customer Edge 1, Provider Edge, and Customer Edge 2 routers.

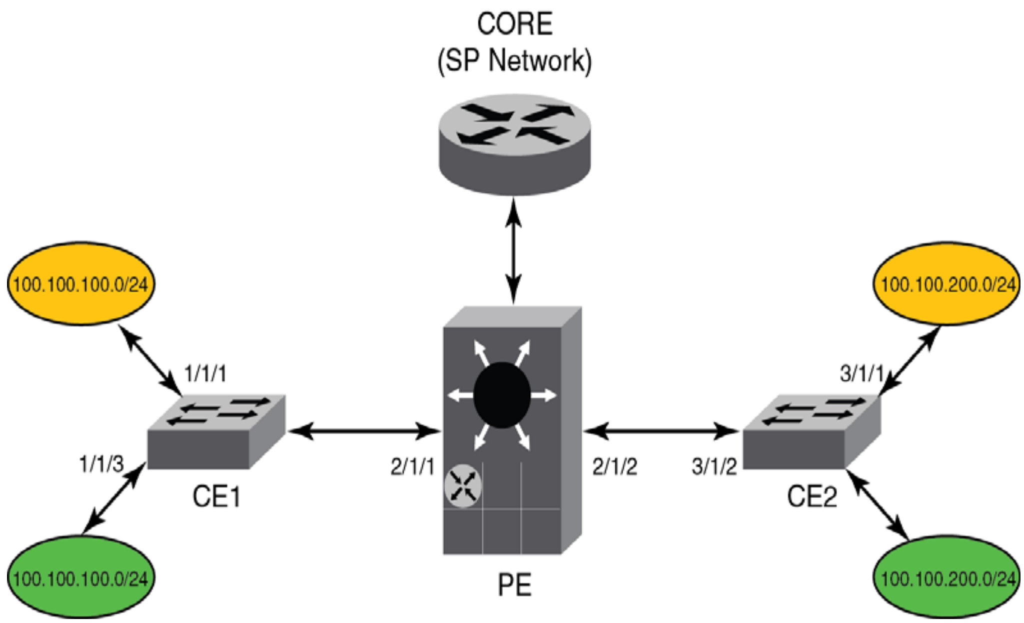


FIGURE 1 **VRF configuration diagram**

ORANGE (v11) and GREEN (v12) are the two VPNs supporting two different customer sites. Both of them have overlapping IP subnets; 100.100.100.0/24 and 100.100.200.0/24.

VRF is supported on the Brocade VDX 8770 and VDX 6740, supporting up to 32 VRFs.

This configuration gives an example of a typical VRF-lite use case and is not meant to give an ideal configuration.

The examples in this section are based on the network diagram in Figure 1.

1. Configure the edge routers by enabling OSPF protocol in VRF configuration mode.

   Example of enabling routing and configuring VRF on the ORANGE edge router. Repeat this example for the GREEN edge router.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2600" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>orange</vrf-name>
```

```
                    <route-distiniguisher>19:1</route-distiniguisher>
                    <address-family>
                        <ipv4>
                            <max-route>399</max-route>
                        </ipv4>
                    </address-family>
                    <ip>
                        <vrf-router-id>11.1.1.1</vrf-router-id>
                    </ip>
                </vrf>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2600" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

2. Configure VRF on the interface.

NOTE
Once VRF is configured on an interface, all Layer 3 configurations on the interface are removed, and you must configure them again.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="2607" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>128</name>
                            <vrf xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                                <forwarding>red</forwarding>
                            </vrf>
                            </ve>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="2607" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

3. Configure the static routes. The static route and ARP must be configured under address family mode.

```
<route>
    <static-route-nh-vrf>
        <static-route-next-vrf-dest>44.4.4.4/32</static-route-next-vrf-dest>
            <next-hop-vrf>default-vrf</next-hop-vrf>
        <static-route-next-hop>2.2.2.2</static-route-next-hop>
    </static-route-nh-vrf>
```

```
</route>
```

4. Configure the static ARP for the interface. The static route and ARP must be configured under address family mode.

```
<arp-entry xmlns="urn:brocade.com:mgmt:brocade-arp">
    <arp-ip-address>3.3.3.3</arp-ip-address>
    <mac-address-value>4.4.4</mac-address-value>
    <interfacename>interface</interfacename>
    <TenGigabitEthernet>2/0/9</TenGigabitEthernet>
</arp-entry>
```

The following example configures all commands under the VRF submode. This configuration is non-default VRF.

```
<vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
    <vrf-name>red</vrf-name>
        <route-distiniguisher>10:1</route-distiniguisher>
        <address-family>
            <ipv4>
                <max-route>1200</max-route>
        <ip xmlns="urn:brocade.com:mgmt:brocade-rtm">
            <route>
                <static-route-nh-vrf>
                    <static-route-next-vrf-dest>44.4.4.4/32
                    </static-route-next-vrf-dest>
                    <next-hop-vrf>default-vrf</next-hop-vrf>
                    <static-route-next-hop>2.2.2.2
                    </static-route-next-hop>
                </static-route-nh-vrf>
            </route>
        </ip>
            <arp-entry xmlns="urn:brocade.com:mgmt:brocade-arp">
                <arp-ip-address>3.3.3.3</arp-ip-address>
                <mac-address-value>4.4.4</mac-address-value>
                <interfacename>interface</interfacename>
                <TenGigabitEthernet>2/0/9</TenGigabitEthernet>
            </arp-entry>
            </ipv4>
        </address-family>
    <ip>
        <vrf-router-id>6.7.8.9</vrf-router-id>
    </ip>
</vrf>
```

Enabling VRRP for VRF

To enable the Virtual Router Redundancy Protocol (VRRP) or VRRP-Extended (VRRP-E) for a Virtual Routing and Forwarding (VRF) region, an interface should be assigned to a VRF region before enabling VRRP or VRRP-E. VRRP is enabled or disabled globally on the switch under RBridge ID configuration mode; it cannot be enabled or disabled on a specific VRF instance. For more information on VRRP on Brocade switches, refer to Chapter 48, “Configuring VRRP”.

Configuring Static Inter-VRF route leaking

Static Inter-VRF route leaking is a feature which should only be deployed by an advanced user.

In this example, one of the static routes in the "Red" VRF (10.50.2.0/24) is being allowed to communicate with one in the "Green" VRF (10.55.2.0/24).

The center icon represents a Brocade VDX router. The red, green and orange ovals represent virtual partitions (VRFs) in that same router. The Destination VRF is where the route is being leaked to ("Green"), and the Source VRF is where the route is being leaked from ("Red").

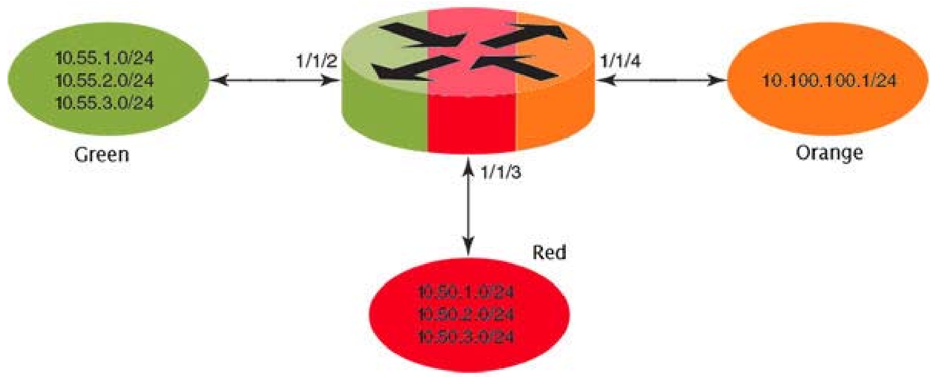


FIGURE 2 **Static Inter-VRF leaking**

1. Configure VRF "Green".

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>Green</vrf-name>
                    <route-distiniguisher>1:2</route-distiniguisher>
                    <address-family>
                        <ip>
                            <unicast></unicast>
                        </ip>
                    </address-family>
                </vrf>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

2. Configure VRF "Red".

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
```

```
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>Red</vrf-name>
                    <route-distiniguisher>2:1</route-distiniguisher>
                    <address-family>
                        <ip>
                            <unicast></unicast>
                        </ip>
                    </address-family>
                </vrf>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

3. Configure interface in the destination VRF "Green" (using the IP address and subnet mask).

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns"urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
                    <vrf>
                        <forwarding>Green</forwarding>
                    </vrf>
                    <ip>
                        <ip-config xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <address>
                                <address>10.55.1.2/24</address>
                            </address>
                        </ip-config>
                    </ip>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

4. Configure interface in the source VRF "Red" (using the IP address and subnet mask).

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
```

```
        </target>
        <config>
            <interface>
                <tengigabitethernet>
                    <name>1/0/3</name>
                    <vrf>
                        <forwarding>Red</forwarding>
                    </vrf>
                    <ip>
                        <ip-config xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <address>
                                <address>10.50.1.2/24</address>
                            </address>
                        </ip-config>
                    </ip>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

5. Navigate to the source VRF address family context for configuring static route leak.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>Red</vrf-name>
                    <address-family>
                        <ip>
                            <unicast>
                                <max-route>300</max-route>
                            </unicast>
                        </ip>
                    </address-family>
                </vrf>
            </rbridge-id>
        </config>
    </edit-config>
</nc:rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

6. Configure the route leak for a network (using the IP address and subnet mask), by mentioning the destination next-hop VRF name and the next hop in the destination VRF.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
```

```
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>Red</vrf-name>
                    <address-family>
                        <ip>
                            <unicast>
                                <max-route>300</max-route>
                                <ip xmlns="urn:brocade.com:mgmt:brocade-rtm">
                                    <route>
                                        <static-route-nh-vrf>

<static-route-next-vrf-dest>10.55.2.0/24</static-route-next-vrf-dest>
                                            <next-hop-vrf>Green</next-hop-vrf>

<static-route-next-hop>10.55.1.1</static-route-next-hop>
                                        </static-route-nh-vrf>
                                    </route>
                                </ip>
                            </unicast>
                        </ip>
                    </address-family>
                </vrf>
            </rbridge-id>
        </config>
    </edit-config>
</nc:rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

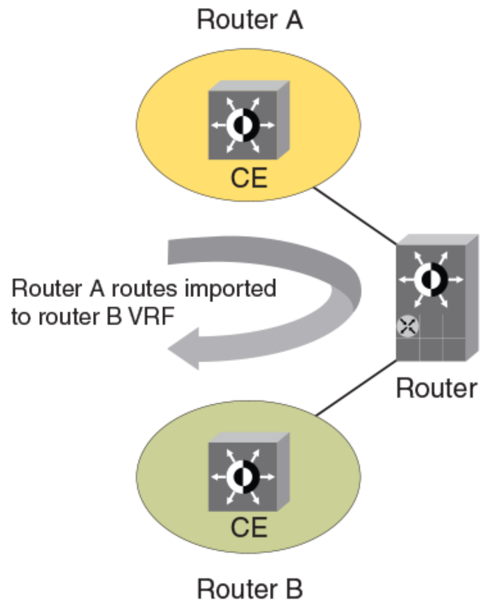
7. For bidirectional route leak traffic, configure a route leak from VRF "Green" to VRF "Red".

Configuring Dynamic Inter-VRF route leaking

Dynamic Inter-VRF route leaking is a feature which should only be deployed by an advanced user.

Use the following procedure to set up Inter-VRF route leaking.

In this example, a route map in the "Red" VRF (10.50.2.0/24) is being allowed to communicate with the "Green" VRF (10.55.2.0/24).



The center icon represents a Brocade VDX router. The red and green ovals represent virtual partitions (VRFs) in that same router. The Destination VRF is where the route is being leaked to ("Green"), and the Source VRF is where the route is being leaked from ("Red"). "Red" has a routing map called importmap that learns routes from an OSPF configuration that is running on "Red”. In this example, IPv4 is used.

1. Configure VRF "Green".

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>Green</vrf-name>
                    <route-distiniguisher>1:2</route-distiniguisher>
                    <address-family>
                        <ip>
                            <unicast></unicast>
                        </ip>
                    </address-family>
                </vrf>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
```

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

2. Configure VRF "Yellow".

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>Yellow</vrf-name>
                    <route-distiniguisher>2:1</route-distiniguisher>
                    <address-family>
                        <ip>
                            <unicast></unicast>
                        </ip>
                    </address-family>
                </vrf>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

3. Configure import-prefix.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ip>
                    <prefix-list xmlns="urn:brocade.com:mgmt:brocade-ip-policy">
                        <name>import-prefix</name>
                        <prefix-ipp>10.2.3.0/24</prefix-ipp>
                    </prefix-list>
                    <prefix-list xmlns="urn:brocade.com:mgmt:brocade-ip-policy">
                        <name>import-prefix</name>
                        <prefix-ipp>10.1.2.0/24</prefix-ipp>
                    </prefix-list>
                </ip>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

4. Configure route-map.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <route-map xmlns="urn:brocade.com:mgmt:brocade-ip-policy">
                    <name>import-map</name>
                    <action-rm>permit</action-rm>
                    <instance>10</instance>
                    <content>
                        <match>
                            <ip>
                                <address>
                                    <prefix-list-rmm>prefix</prefix-list-rmm>
                                </address>
                            </ip>
                        </match>
                    </content>
                </route-map>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

5. Import the desired route map after first navigating to destination VRF.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>Green</vrf-name>
                    <address-family>
                        <ip>
                            <unicast>
                                <ip xmlns="urn:brocade.com:mgmt:brocade-rtm">
                                    <import>
                                        <routes>
                                            <src-vrf>Yellow</src-vrf>
                                            <map>import-map</map>
                                        </routes>
                                    </import>
                                </ip>
                            </unicast>
                        </ip>
                    </address-family>
```

```
            </vrf>
         </rbridge-id>
      </config>
   </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
   <ok/>
</rpc-reply>
```

Understanding and using the management VRF

ATTENTION
Beginning with release 6.0.1, support is provided for the management VRF and default VRF.

Enabling a management VRF on an Ethernet interface

The following enables the management VRF on an Ethernet interface and assigns the interface to a subnet.

```
<?xml version="1.0" ?>
<rpc nc:message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
      <target>
         <running/>
      </target>
      <config>
         <interface xmlns="urn:brocade.com:mgmt:brocade-interface”>
            <fortygigabitethernet>
               <name>2/0/51</name>
               <vrf>
                  <forwarding>mgmt-vrf</forwarding>
               </vrf>
               <ip>
                  <ip-config xmlns="urn:brocade.com:mgmt:brocade-ip-config”>
                     <address>
                        <address>10.1.1.1/24</address>
                     </address>
                  </ip-config>
               </ip>
            </fortygigabitethernet>
         </interface>
      </config>
   </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
   <ok/>
</rpc-reply>
```

NOTE
Disabling a management VRF on a VE interface can be done by using the No command of the management vrf command. It disables a management VRF previously configured on a VE interface.

Configuring IPv4 routing for a management VRF on an RBridge interface.

The following configures an IPv4 route subnet for the management VRF, enters address family IPv4 configuration mode, and assigns the management VRF to an Ethernet interface.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id<2</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>mgmt-vrf</vrf-name>
                    <address-family>
                        <ip>
                            <unicast>
                                <ip xmlns="urn:brocade.com:mgmt:brocade-rtm">
                                    <route>
                                        <static-route-oif>

<static-route-dest>10.1.1.0/32</static-route-dest>

<static-route-oif-type>fortygigabitethernet</static-route-oif-type>

<static-route-oif-name>2/0/52</static-route-oif-name>
                                        </static-route-oif>
                                    </route>
                                </ip>
                            </unicast>
                        </ip>
                    </address-family>
                </vrf>
            </rbridge-id>
        </config>
    </edit-config>
</nc:rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Chapter

Configuring Multi-VRF 53

In this chapter

Multi-VRF overview

Virtual Routing and Forwarding (VRF) allows routers to maintain multiple routing tables and forwarding tables on the same router. A Multi-VRF router can run multiple instances of routing protocols with a neighboring router with overlapping address spaces configured on different VRF instances.

Some vendors also use the terms Multi-VRF CE or VRF-Lite for this technology. VRF-Lite provides a reliable mechanism for a network administrator to maintain multiple virtual routers on the same device. The goal of providing isolation among different VPN instances is accomplished without the overhead of heavyweight protocols (such as MPLS) used in secure VPN technologies. Overlapping address spaces can be maintained among the different VPN instances.

Central to VRF-Lite is the ability to maintain multiple VRF tables on the same Provider Edge (PE) Router. VRF-Lite uses multiple instances of a routing protocol such as OSPF or BGP to exchange route information for a VPN among peer PE routers. The VRF-Lite capable PE router maps an input customer interface to a unique VPN instance. The router maintains a different VRF table for each VPN instance on that PE router. Multiple input interfaces may also be associated with the same VRF on the router, if they connect to sites belonging to the same VPN. This input interface can be a physical interface or a virtual Ethernet interface on a port.

In Multi-VRF deployments:

- Two VRF-capable routers must be directly connected at Layer 3, deploying BGP, OSPF, RIP, or static routes.
- Each VRF maintains unique routing and forwarding tables.
- Each VRF can be assigned one or more Layer 3 interfaces on a router to be part of the VRF.
- Each VRF can be configured with IPv4 address family, IPv6 address family, or both.
- A packet’s VRF instance is determined based on the VRF index of the interface on which the packet is received.
- Separate routing protocol instances are required for each VRF instance.
- Overlapping address spaces can be configured on different VRF instances.

Multi-VRF deployments provide the flexibility to maintain multiple virtual routers, which are segregated for each VRF instance.

Multi-VRF with eBGP and OSPF

Multi-VRF with eBGP and OSPF: Configuring PE1

1. Create VLANs 10 and 20.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>10</name>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>


<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>20</name>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

2. Create VRF "green" and “red” and assign a Route Distinguisher.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
```

```
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>green</vrf-name>
                    <route-distiniguisher>10:10</route-distiniguisher>
                </vrf>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>red</vrf-name>
                    <route-distiniguisher>20:20</route-distiniguisher>
                </vrf>
             </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

3. Enable BGP routing and configure as following.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <local-as>1</local-as>
                        </router-bgp-attributes>
                        <address-family>
                            <ipv4>
                                <ipv4-unicast>
                                    <af-vrf>
                                        <af-vrf-name>green</af-vrf-name>
                                        <af-ipv4-uc-and-vrf-cmds-call-point-holder>
                                            <redistribute>
                                                <ospf>

<redistribute-ospf></redistribute-ospf>
                                                    <match>
                                                        <ospf-internal></ospf-internal>

<ospf-external1></ospf-external1>

<ospf-external2></ospf-external2>
                                                    </match>
                                                </ospf>
                                            </redistribute>
                                        </af-ipv4-uc-and-vrf-cmds-call-point-holder>
                                        <network>

<network-ipv4-address>10.3.1.0/24</network-ipv4-address>
```

```
                                    </network>
                                    <neighbor>
                                        <af-ipv4-vrf-neighbor-address-holder>
                                            <af-ipv4-neighbor-addr>

<af-ipv4-neighbor-address>10.3.1.2</af-ipv4-neighbor-address>
                                                <remote-as>2</remote-as>
                                            </af-ipv4-neighbor-addr>
                                        </af-ipv4-vrf-neighbor-address-holder>
                                    </neighbor>

<client-to-client-reflection></client-to-client-reflection>
                                </af-vrf>
                                <af-vrf>
                                    <af-vrf-name>red</af-vrf-name>
                                    <af-ipv4-uc-and-vrf-cmds-call-point-holder>
                                        <redistribute>
                                            <ospf>

<redistribute-ospf></redistribute-ospf>
                                                <match>
                                                    <ospf-internal></ospf-internal>

<ospf-external1></ospf-external1>

<ospf-external2></ospf-external2>
                                                </match>
                                            </ospf>
                                        </redistribute>
                                    </af-ipv4-uc-and-vrf-cmds-call-point-holder>
                                    <network>

<network-ipv4-address>10.3.1.0/24</network-ipv4-address>
                                    </network>
                                    <neighbor>
                                        <af-ipv4-vrf-neighbor-address-holder>
                                            <af-ipv4-neighbor-addr>

<af-ipv4-neighbor-address>10.3.1.2</af-ipv4-neighbor-address>
                                                <remote-as>2</remote-as>
                                            </af-ipv4-neighbor-addr>
                                        </af-ipv4-vrf-neighbor-address-holder>
                                    </neighbor>

<client-to-client-reflection></client-to-client-reflection>
                                </af-vrf>
                            </ipv4-unicast>
                        </ipv4>
                    </address-family>
                </router-bgp>
            </router>
        </rbridge-id>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply
```

4. Enable OSPF routing and configure the following.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf">
                        <vrf>green</vrf>
                        <redistribute>
                            <bgp></bgp>
                        </redistribute>
                        <area>
                            <area-id>0</area-id>
                            <normal></normal>
                        </area>
                    </ospf>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf">
                        <vrf>red</vrf>
                        <redistribute>
                            <bgp></bgp>
                        </redistribute>
                        <area>
                            <area-id>0</area-id>
                            <normal></normal>
                        </area>
                    </ospf>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply
```

5. Configure the VE interfaces for the appropriate VRF and network.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>10</name>
                        <vrf xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <forwarding>green</forwarding>
                        </vrf>
                        <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <ip-config>
```

```
                                <address>
                                    <address>10.3.1.1/24</address>
                                </address>
                            </ip-config>
                        </ip>
                    </ve>
                    <ve>
                        <name>20</name>
                        <vrf xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <forwarding>red</forwarding>
                        </vrf>
                        <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <ip-config>
                                <address>
                                    <address>10.3.1.1/24</address>
                                </address>
                            </ip-config>
                        </ip>
                    </ve>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply
```

Multi-VRF with eBGP and OSPF: Configuring PE2

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <interface-nodespecific>
                    <ns-vlan>10</ns-vlan>
                    <ns-ethernet>100</ns-ethernet>
                </interface-nodespecific>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>green</vrf-name>
                    <route-distiniguisher>10:10</route-distiniguisher>
                    <address-family>
                        <ip>
                            <unicast></unicast>
                        </ip>
                    </address-family>
                </vrf>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>red</vrf-name>
                    <route-distiniguisher>20:20</route-distiniguisher>
                    <address-family>
                        <ip>
                            <unicast></unicast>
```

```
                    </ip>
                </address-family>
            </vrf>
            <router>
                <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                    <router-bgp-attributes>
                        <local-as>1</local-as>
                    </router-bgp-attributes>
                    <address-family>
                        <ipv4>
                            <ipv4-unicast>
                                <af-vrf>
                                    <af-vrf-name>green</af-vrf-name>

<af-ipv4-uc-and-vrf-cmds-call-point-holder>
                                        <redistribute>
                                            <ospf>

<redistribute-ospf></redistribute-ospf>
                                                <match>
                                                    <ospf-internal></ospf-internal>

<ospf-external1></ospf-external1>

<ospf-external2></ospf-external2>
                                                </match>
                                            </ospf>
                                        </redistribute>
                                    </af-ipv4-uc-and-vrf-cmds-call-point-holder>
                                    <network>

<network-ipv4-address>10.3.1.0/24</network-ipv4-address>
                                    </network>
                                    <neighbor>
                                        <af-ipv4-vrf-neighbor-address-holder>
                                            <af-ipv4-neighbor-addr>

<af-ipv4-neighbor-address>10.3.1.2</af-ipv4-neighbor-address>
                                                <remote-as>2</remote-as>
                                            </af-ipv4-neighbor-addr>
                                        </af-ipv4-vrf-neighbor-address-holder>
                                    </neighbor>
                                </af-vrf>
                                <af-vrf>
                                    <af-vrf-name>red</af-vrf-name>
                                    <af-ipv4-uc-and-vrf-cmds-call-point-holder>
                                        <redistribute>
                                            <ospf>

<redistribute-ospf></redistribute-ospf>
                                                <match>
                                                    <ospf-internal></ospf-internal>

<ospf-external1></ospf-external1>

<ospf-external2></ospf-external2>
                                                </match>
                                            </ospf>
                                        </redistribute>
                                    </af-ipv4-uc-and-vrf-cmds-call-point-holder>
```

```
                                    <network>

<network-ipv4-address>10.3.1.0/24</network-ipv4-address>
                                    </network>
                                    <neighbor>
                                        <af-ipv4-vrf-neighbor-address-holder>
                                            <af-ipv4-neighbor-addr>

<af-ipv4-neighbor-address>10.3.1.2</af-ipv4-neighbor-address>
                                                <remote-as>2</remote-as>
                                            </af-ipv4-neighbor-addr>
                                        </af-ipv4-vrf-neighbor-address-holder>
                                    </neighbor>
                                </af-vrf>
                            </ipv4-unicast>
                        </ipv4>
                    </address-family>
                </router-bgp>
                <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf">
                    <vrf>green</vrf>
                    <redistribute>
                        <bgp></bgp>
                    </redistribute>
                    <area>
                        <area-id>0</area-id>
                        <normal></normal>
                    </area>
                </ospf>
                <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf">
                    <vrf>red</vrf>
                    <redistribute>
                        <bgp></bgp>
                    </redistribute>
                    <area>
                        <area-id>0</area-id>
                        <normal></normal>
                    </area>
                </ospf>
            </router>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <ve>
                    <name>10</name>
                    <vrf xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                        <forwarding>green</forwarding>
                    </vrf>
                    <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                        <ip-config>
                            <address>
                                <address>10.3.1.1/24</address>
                            </address>
                        </ip-config>
                    </ip>
                </ve>
                <ve>
                    <name>20</name>
                    <vrf xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                        <forwarding>red</forwarding>
                    </vrf>
                    <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                        <ip-config>
```

```
                                <address>
                                    <address>10.3.1.1/24</address>
                                </address>
                            </ip-config>
                        </ip>
                    </ve>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply
```

Multi-VRF with eBGP and OSPF: Configuring CE1 and CE2

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf">
                        <vrf>default-vrf</vrf>
                        <redistribute>
                            <connected></connected>
                        </redistribute>
                        <area>
                            <area-id>0</area-id>
                            <normal></normal>
                        </area>
                    </ospf>
                </router>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <loopback xmlns="urn:brocade.com:mgmt:brocade-intf-loopback">
                        <id>1</id>
                        <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <ip-config>
                                <address>
                                    <address>10.1.3.1/32</address>
                                </address>
                            </ip-config>
                        </ip>
                    </loopback>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply
```

Multi-VRF with eBGP and OSPF: Configuring CE3 and CE4

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <ospf xmlns="urn:brocade.com:mgmt:brocade-ospf">
                        <vrf>default-vrf</vrf>
                        <redistribute>
                            <connected></connected>
                        </redistribute>
                        <area>
                            <area-id>0</area-id>
                            <normal></normal>
                        </area>
                    </ospf>
                </router>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <loopback xmlns="urn:brocade.com:mgmt:brocade-intf-loopback">
                        <id>1</id>
                        <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <ip-config>
                                <address>
                                    <address>10.1.3.1/32</address>
                                </address>
                            </ip-config>
                        </ip>
                    </loopback>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply
```

Chapter

Configuring BGP 54

In this chapter

BGP configuration with NETCONF overview

Border Gateway Protocol (BGP) is an exterior gateway protocol that can do inter-domain and intra-domain routing. It peers to other BGP-speaking systems over TCP to exchange network reachability and routing information.

Refer to the *Network OS Layer 3 Routing Configuration Guide* for information on BGP and for the following related information:

- BGP Peering
- BGP Attributes
- Best-Path Algorithm
- Device ID
- Neighbor configuration
- Configuration fundamentals and optimization.

Through the NETCONF interface, you can perform the following operations that affect the functioning of BGP:

- Use the <edit-config> RPC to activate, configure, or deactivate BGP for an RBridge.
- Use the <edit-config> RPC to add, delete, or edit IPV4 address-family specific configurations.
- Use the <get-config> RPC to verify all or part of the BGP configuration.

BGP parameters are defined in the brocade-bgp YANG module. For information about the brocade-bgp YANG module, refer to the *Network OS YANG Reference Manual*.

Configuring BGP

Configuring BGP can be divided into three separate phases:

- "Enabling BGP on an RBridge" on page 570
- "Configuring BGP global mode" on page 571
- "Configuring IPv4/IPv6 unicast address family" on page 572

Enabling BGP on an RBridge

To enable BGP on an RBridge, configure BGP with the default vrf-name for that RBridge.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
     </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Disabling BGP on an RBridge

To disable BGP on an RBridge, enter RBridge ID configuration mode and delete the <bgp> node. The following examples disables BGP on the RBridge by deleting the node.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <delete-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
     </delete-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring BGP global mode

Configurations that are not specific to address-family configuration are available in the BGP global configuration mode. The nodes supporting BGP global configuration mode are:

- always-compare-med—Allow comparing MED from different neighbors
- as-path-ignore—Ignore AS_PATH length for best route selection
- capability—Set capability
- cluster-id—Configure Route-Reflector Cluster-ID
- compare-med-empty-aspath—Allow comparing MED from different neighbors even with empty as-path attribute.
- compare-routerid—Compare router-id for identical BGP paths
- default-local-preference—Configure default local preference value
- distance—Define an administrative distance
- enforce-first-as—Enforce the first AS for EBGP routes
- fast-external-fallover—Reset session if link to EBGP peer goes down
- install-igp-cost—Install IGP cost to next hop instead of MED value as BGP route cost
- local-as—Configure local AS number
- log-dampening-debug—Log dampening debug messages
- maxas-limit—Impose limit on number of ASes in AS-PATH attribute
- med-missing-as-worst—Consider routes missing MED attribute as least desirable
- neighbor—Specify a neighbor router
- timers—Adjust routing timers

For complete information on all of these nodes, refer to the BGP parameters defined in the brocade-bgp YANG module. For information about the brocade-bgp YANG module, refer to the *Network OS YANG Reference Manual*.

The following illustrates the configuration of remote AS number for neighbor 1.1.1.1.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <neighbor>
                                <neighbor-ips>
                                    <neighbor-addr>

<router-bgp-neighbor-address>1.1.1.1</router-bgp-neighbor-address>
                                        <remote-as>20</remote-as>
                                    </neighbor-addr>
                                </neighbor-ips>
                            </neighbor>
                        </router-bgp-attributes>
```

```
                  </router-bgp>
              </router>
          </rbridge-id>
       </config>
   </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring IPv4/IPv6 unicast address family

Currently only the IPv4 unicast address family is supported.

The following configurations are allowed under BGP IPv4/IPv6 address-family mode:

- Network (including static networks)
- Route aggregation
- Route redistribution
- Route reflection
- Dampening
- Default route origination
- Multipathing (including maximum paths)
- Address-family-specific neighbor configuration
- Explicit specification of networks to advertise

Nodes that are specific to address-family configuration are:

- aggregate-address—Configure BGP aggregate entries
- always-propagate—Allow readvertisement of best BGP routes not in IP Forwarding table
- bgp-redistribute-internal—Allow redistribution of iBGP routes into IGPs
- client-to-client-reflection—Configure client to client route reflection
- dampening—Enable route-flap dampening
- default-information-originate—Originate Default Information
- default-metric—Set metric of redistributed routes
- maximum-paths—Forward packets over multiple paths
- multipath—Enable multipath for iBGP or EBGP neighbors only
- neighbor—Specify a neighbor router
- network—Specify a network to announce via BGP
- next-hop-enable-default—Enable default route for BGP next-hop lookup
- next-hop-recursion—Perform next-hop recursive lookup for BGP route
- redistribute—Redistribute information from another routing protocol
- rib-route-limit—Limit BGP rib count in routing table
- static-network—Special network that do not depends on IGP and always treat as best route in BGP

- table-map—Map external entry attributes into routing table
- update-time—Configure IGP route update interval

For complete information on all of these nodes, refer to the BGP parameters are defined in the brocade-bgp YANG module. For information about the brocade-bgp YANG module, refer to the *Network OS YANG Reference Manual*.

This configuration example configures the neighbor with a weight of 20.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <address-family>
                            <ipv4>
                                <ipv4-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <neighbor>
                                            <af-ipv4-neighbor-address-holder>
                                                <af-ipv4-neighbor-address>

<af-ipv4-neighbor-address>1.1.1.1</af-ipv4-neighbor-address>
                                                    <af-nei-weight>20</af-nei-weight>
                                                </af-ipv4-neighbor-address>
                                            </af-ipv4-neighbor-address-holder>
                                        </neighbor>
                                    </default-vrf>
                                </ipv4-unicast>
                            </ipv4>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
     </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Chapter

Configuring BGP4+ 55

In this chapter

Configure BGP4+

Before enabling BGP4+ on a device, you must enable the forwarding of IPv6 traffic on the device using the **ipv6 unicast-routing** command and enable IPv6 on at least one interface by configuring an IPv6 address or explicitly enabling IPv6 on that interface.

For more information on performing these configuration tasks, refer to *FastIron Ethernet Switch* Administration Guide.

To configure BGP4+, you must do the following:

- Enable BGP4+.
- Configure BGP4+ neighbors using one of the following methods:
  - Add one neighbor at a time (neighbor uses global or site-local IPv6 address).
  - Add one neighbor at a time (neighbor uses global or unique local IPv6 address).
  - Add one neighbor at a time (neighbor uses a link-local IPv6 address).
  - Create a peer group and add neighbors individually.

The following configuration tasks are optional:

- Advertise the default route.
- Import specified routes into BGP4+.
- Redistribute prefixes into BGP4+.
- Aggregate routes advertised to BGP4 neighbors.
- Use route maps.

Enable BGP4+ and adding neighbors

The following example is to enable BGP4+ and adding neighbors.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <local-as>1000</local-as>
                        </router-bgp-attributes>
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <neighbor>
                                            <af-ipv6-neighbor-address-holder>
                                                <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>2001:db8:e0ff:783a::4</af-ipv6-neighbor-address>
                                                    <activate></activate>
                                                </af-ipv6-neighbor-address>
                                            </af-ipv6-neighbor-address-holder>
                                        </neighbor>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring BGP4+ neighbors using global IPv6 addresses

To configure BGP4+ neighbors using global or link-local IPv6 addresses, you must add the IPv6 address of a neighbor in a remote autonomous system to the BGP4+ neighbor table of the local device.

You must repeat this procedure for each neighbor that you want to add to a local device.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
```

```
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <local-as>1000</local-as>
                            <neighbor>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>2001:db8:e0ff:783a::4</router-bgp-neighbor-ipv6
-address>
                                        <remote-as>12</remote-as>
                                    </neighbor-ipv6-addr>
                                </neighbor-ipv6s>
                            </neighbor>
                        </router-bgp-attributes>
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <neighbor>
                                            <af-ipv6-neighbor-address-holder>
                                                <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>2001:db8:e0ff:783a::4</af-ipv6-neighbor-address>
                                                    <activate></activate>
                                                </af-ipv6-neighbor-address>
                                            </af-ipv6-neighbor-address-holder>
                                        </neighbor>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                /router>
            </rbridge-id>
         </config>
     </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Adding BGP4+ neighbors using link-local addresses

To configure BGP4+ neighbors that use link-local addresses, you must do the following:

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
```

```
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <route-map xmlns="urn:brocade.com:mgmt:brocade-ip-policy">
                    <name>map1</name>
                    <action-rm>permit</action-rm>
                    <instance>10</instance>
                    <content>
                        <set>
                            <ipv6>
                                <next-ip>
                                    <next-hop>
                                        <next-hop>2001::10</next-hop>
                                    </next-hop>
                                </next-ip>
                            </ipv6>
                        </set>
                    </content>
                </route-map>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <local-as>1000</local-as>
                            <neighbor>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>fe80:4398:ab30:45de::1</router-bgp-neighbor-ipv
6-address>
                                        <remote-as>1001</remote-as>
                                        <update-source>
                                            <ethernet-interface>

<interface-type>tengigabitethernet</interface-type>
                                                <ethernet>1/0/5</ethernet>
                                            </ethernet-interface>
                                        </update-source>
                                    </neighbor-ipv6-addr>
                                </neighbor-ipv6s>
                            </neighbor>
                        </router-bgp-attributes>
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <neighbor>
                                            <af-ipv6-neighbor-address-holder>
                                                <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>fe80:4398:ab30:45de::1</af-ipv6-neighbor-address>
                                                    <activate></activate>
                                                    <neighbor-route-map>

<neighbor-route-map-direction-out>

<neighbor-route-map-name-direction-out>map1</neighbor-route-map-name-direction-ou
t>

</neighbor-route-map-direction-out>
```

```
                                                </neighbor-route-map>
                                            </af-ipv6-neighbor-address>
                                        </af-ipv6-neighbor-address-holder>
                                    </neighbor>
                                </default-vrf>
                            </ipv6-unicast>
                        </ipv6>
                    </address-family>
                </router-bgp>
            </router>
        </rbridge-id>
     </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring a BGP4+ peer group

If a peer group has multiple neighbors with similar attributes, you can configure a peer group, then add neighbors to the group instead of configuring neighbors individually for all parameters.

NOTE
You can add IPv6 neighbors only to an IPv6 peer group. You cannot add an IPv4 neighbor to an IPv6 peer group and vice versa. IPv6 and IPv6 peer groups must remain separate.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <local-as>1000</local-as>
                            <neighbor>
                                <peer-grps>
                                    <neighbor-peer-grp>

<router-bgp-neighbor-peer-grp>mypeergroup1</router-bgp-neighbor-peer-grp>
                                        <peer-group-name></peer-group-name>
                                        <remote-as>11</remote-as>
                                    </neighbor-peer-grp>
                                </peer-grps>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>2001:2018:8192::124</router-bgp-neighbor-ipv6-a
ddress>
```

```
<associate-peer-group>mypeergroup1</associate-peer-group>
                                </neighbor-ipv6-addr>
                                <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>2001:2018:8192::125</router-bgp-neighbor-ipv6-a
ddress>

<associate-peer-group>mypeergroup1</associate-peer-group>
                                </neighbor-ipv6-addr>
                            </neighbor-ipv6s>
                        </neighbor>
                    </router-bgp-attributes>
                    <address-family>
                        <ipv6>
                            <ipv6-unicast>
                                <default-vrf>
                                    <default-vrf-selected></default-vrf-selected>
                                    <neighbor>
                                        <af-ipv6-neighbor-peergroup-holder>
                                            <af-ipv6-neighbor-peergroup>

<af-ipv6-neighbor-peergroup-name>mypeergroup1</af-ipv6-neighbor-peergroup-name>
                                                <activate></activate>
                                            </af-ipv6-neighbor-peergroup>
                                        </af-ipv6-neighbor-peergroup-holder>
                                    </neighbor>
                                </default-vrf>
                            </ipv6-unicast>
                        </ipv6>
                    </address-family>
                </router-bgp>
            </router>
        </rbridge-id>
     </config>
   </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring a peer group with IPv4 and IPv6 peers

To create a peer group with both IPv6 and IPv4 peers and activate the peer group in the IPv6 address family, refer to the "Configuring a BGP4+ peer group" section. To configure an IPv4 peer, `<router-bgp-neighbor-ipv6-address/>` should contain the IPv4 IP address.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
```

```
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <local-as>1000</local-as>
                            <neighbor>
                                <peer-grps>
                                    <neighbor-peer-grp>

<router-bgp-neighbor-peer-grp>p1</router-bgp-neighbor-peer-grp>
                                        <peer-group-name></peer-group-name>
                                        <remote-as>11</remote-as>
                                    </neighbor-peer-grp>
                                </peer-grps>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>2001:2018:8192::124</router-bgp-neighbor-ipv6-a
ddress>

<associate-peer-group>p1</associate-peer-group>
                                    </neighbor-ipv6-addr>
                                </neighbor-ipv6s>
                                <neighbor-ips>
                                    <neighbor-addr>

<router-bgp-neighbor-address>10.0.0.1</router-bgp-neighbor-address>

<associate-peer-group>p1</associate-peer-group>
                                    </neighbor-addr>
                                </neighbor-ips>
                            </neighbor>
                        </router-bgp-attributes>
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <neighbor>
                                            <af-ipv6-neighbor-peergroup-holder>
                                                <af-ipv6-neighbor-peergroup>

<af-ipv6-neighbor-peergroup-name>p1</af-ipv6-neighbor-peergroup-name>
                                                    <activate></activate>
                                                </af-ipv6-neighbor-peergroup>
                                            </af-ipv6-neighbor-peergroup-holder>
                                        </neighbor>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
```

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

Importing routes into BGP4+

You can explicitly specify routes to be advertised by BGP.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <neighbor>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>fe80:4398:ab30:45de::1</router-bgp-neighbor-ipv
6-address>
                                        <remote-as>1001</remote-as>
                                        <update-source>
                                            <ethernet-interface>

<interface-type>tengigabitethernet</interface-type>
                                     <ethernet>1/0/5</ethernet>
                                            </ethernet-interface>
                                        </update-source>
                                    </neighbor-ipv6-addr>
                                </neighbor-ipv6s>
                            </neighbor>
                        </router-bgp-attributes>
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <network>

<network-ipv6-address>2001:db8::/32</network-ipv6-address>
                                        </network>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
     </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
```

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

Advertising the default BGP4+ route

You can enable a BGP4+ device to advertise the default BGP4+ route and send the default route to a neighbor.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <af-common-cmds-holder>

<default-information-originate></default-information-originate>

<client-to-client-reflection></client-to-client-reflection>
                                        </af-common-cmds-holder>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
     </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Advertising the default BGP4+ route to a specific neighbor

You can configure a BGP device to advertise the default IPv6 route to a specific neighbor.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
```

```
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <local-as>1000</local-as>
                            <neighbor>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>2001:db8:93e8:cc00::1</router-bgp-neighbor-ipv6
-address>
                                        <remote-as>1001</remote-as>
                                    </neighbor-ipv6-addr>
                                </neighbor-ipv6s>
                            </neighbor>
                        </router-bgp-attributes>
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <neighbor>
                                            <af-ipv6-neighbor-address-holder>
                                                <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>2001:db8:93e8:cc00::1</af-ipv6-neighbor-address>
                                                    <default-originate>

<default-originate-status></default-originate-status>
                                                    </default-originate>
                                                </af-ipv6-neighbor-address>
                                            </af-ipv6-neighbor-address-holder>
                                        </neighbor>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Using the IPv6 default route as a valid next hop for a BGP4+ route

You can configure a BGP device to use the default route as a valid next hop.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
```

```
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <af-common-cmds-holder>

<client-to-client-reflection></client-to-client-reflection>

<next-hop-enable-default></next-hop-enable-default>
                                        </af-common-cmds-holder>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Enabling next-hop recursion

You can enable next-hop recursion so that a device can the IGP route to the next-hop gateway for a BGP4+ route.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <af-common-cmds-holder>

<client-to-client-reflection></client-to-client-reflection>
                                        </af-common-cmds-holder>

<ipv6-ucast-next-hop-recursion></ipv6-ucast-next-hop-recursion>
                                    </default-vrf>
```

```
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring a cluster ID for a route reflector

The cluster ID can be changed if there is more than one route reflector, so that all route reflectors belong to the same cluster.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <cluster-id>
                                <cluster-id-value>321</cluster-id-value>
                            </cluster-id>
                        </router-bgp-attributes>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Aggregating routes advertised to BGP neighbors

You can configure a device to aggregate routes in a range of networks into a single IPv6 prefix.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
```

```
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <aggregate-ipv6-address>

<aggregate-ip-prefix>2001:db8::/32</aggregate-ip-prefix>
                                        </aggregate-ipv6-address>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Enabling load-balancing across different paths

You can enable load-balancing across different paths.

The following example sets the maximum number of BGP4+ shared paths to 8.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <af-common-cmds-holder>
                                            <maximum-paths>

<load-sharing-value>8</load-sharing-value>
                                            </maximum-paths>

<client-to-client-reflection></client-to-client-reflection>
```

```
                                    </af-common-cmds-holder>
                                </default-vrf>
                            </ipv6-unicast>
                        </ipv6>
                    </address-family>
                </router-bgp>
            </router>
        </rbridge-id>
     </config>
 </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

The following example sets the maximum number of BGP4+ shared paths to that of the value already configured using the **ip load-sharing** command.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <af-common-cmds-holder>
                                            <maximum-paths>
                                                <use-load-sharing></use-load-sharing>
                                            </maximum-paths>

<client-to-client-reflection></client-to-client-reflection>
                                    </af-common-cmds-holder>
                                </default-vrf>
                            </ipv6-unicast>
                        </ipv6>
                    </address-family>
                </router-bgp>
            </router>
        </rbridge-id>
     </config>
 </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring a route map for BGP4+ prefixes

Route maps can be applied to IPv6 unicast address prefixes either as the inbound or outbound routing policy for neighbors under the specified address family.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ipv6>
                    <prefix-list xmlns="urn:brocade.com:mgmt:brocade-ip-policy">
                        <name>myprefixlist</name>
                        <seq-keyword>seq</seq-keyword>
                        <instance>10</instance>
                        <action-ipp>permit</action-ipp>
                        <ipv6-prefix-ipp>2001:db8::/32</ipv6-prefix-ipp>
                    </prefix-list>
                </ipv6>
                <route-map xmlns="urn:brocade.com:mgmt:brocade-ip-policy">
                    <name>map1</name>
                    <action-rm>permit</action-rm>
                    <instance>10</instance>
                    <content>
                        <match>
                            <ipv6>
                                <address>

<ipv6-prefix-list-rmm>myprefixlist</ipv6-prefix-list-rmm>
                                </address>
                            </ipv6>
                        </match>
                    </content>
                </route-map>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <local-as>1000</local-as>
                            <neighbor>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>fe80:4398:ab30:45de::1</router-bgp-neighbor-ipv
6-address>
                                        <remote-as>1001</remote-as>
                                        <update-source>
                                            <ethernet-interface>

<interface-type>tengigabitethernet</interface-type>
                                                <ethernet>1/0/5</ethernet>
                                            </ethernet-interface>
                                        </update-source>
                                    </neighbor-ipv6-addr>
                                </neighbor-ipv6s>
                            </neighbor>
                        </router-bgp-attributes>
```

```
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <neighbor>
                                            <af-ipv6-neighbor-address-holder>
                                                <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>fe80:4398:ab30:45de::1</af-ipv6-neighbor-address>
                                                    <activate></activate>
                                                    <neighbor-route-map>

<neighbor-route-map-direction-out>

<neighbor-route-map-name-direction-out>map1</neighbor-route-map-name-direction-ou
t>

</neighbor-route-map-direction-out>
                                                    </neighbor-route-map>
                                                </af-ipv6-neighbor-address>
                                            </af-ipv6-neighbor-address-holder>
                                        </neighbor>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Redistributing prefixes into BGP4+

Static, connected, and OSPF routes can be redistributed into BGP.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
```

```
                                    <af-ipv6-uc-and-vrf-cmds-call-point-holder>
                                        <redistribute>
                                            <ospf>

<redistribute-ospf></redistribute-ospf>
                                                <match>

<ospf-external1></ospf-external1>
                                                </match>
                                            </ospf>
                                        </redistribute>
                                    </af-ipv6-uc-and-vrf-cmds-call-point-holder>
                                </default-vrf>
                            </ipv6-unicast>
                        </ipv6>
                    </address-family>
                </router-bgp>
            </router>
        </rbridge-id>
     </config>
  </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring BGP4+ outbound route filtering

The BGP4+ Outbound Route Filtering (ORF) prefix list capability can be configured in receive mode, send mode, or both send and receive modes, minimizing the number of BGP updates exchanged between BGP peers.

The following example configures ORF in receive mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <neighbor>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>2001:db8:e0ff:783a::4</router-bgp-neighbor-ipv6
-address>
                                        <remote-as>1001</remote-as>
                                    </neighbor-ipv6-addr>
                                </neighbor-ipv6s>
                            </neighbor>
                        </router-bgp-attributes>
```

```
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <neighbor>
                                            <af-ipv6-neighbor-address-holder>
                                                <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>2001:db8:e0ff:783a::4</af-ipv6-neighbor-address>
                                                    <af-neighbor-capability>
                                                        <orf>
                                                            <prefixlist>

<prefixlist-status></prefixlist-status>

<prefixlist-receive></prefixlist-receive>
                                                            </prefixlist>
                                                        </orf>
                                                    </af-neighbor-capability>
                                                    <activate></activate>
                                                </af-ipv6-neighbor-address>
                                            </af-ipv6-neighbor-address-holder>
                                        </neighbor>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

The following example configures ORF in send mode.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
             <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <neighbor>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>2001:db8:e0ff:783a::4</router-bgp-neighbor-ipv6
-address>
```

```
                                    <remote-as>1001</remote-as>
                                </neighbor-ipv6-addr>
                            </neighbor-ipv6s>
                        </neighbor>
                    </router-bgp-attributes>
                    <address-family>
                        <ipv6>
                            <ipv6-unicast>
                                <default-vrf>
                                    <default-vrf-selected></default-vrf-selected>
                                    <neighbor>
                                        <af-ipv6-neighbor-address-holder>
                                            <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>2001:db8:e0ff:783a::4</af-ipv6-neighbor-address>
                                                <af-neighbor-capability>
                                                    <orf>
                                                        <prefixlist>

<prefixlist-status></prefixlist-status>

<prefixlist-send></prefixlist-send>
                                                        </prefixlist>
                                                    </orf>
                                                </af-neighbor-capability>
                                                <activate></activate>
                                                <prefix-list>
                                                    <direction-in>

<prefix-list-direction-in-prefix-name>myprefixlist</prefix-list-direction-in-pref
ix-name>

<prefix-list-direction-in></prefix-list-direction-in>
                                                    </direction-in>
                                                </prefix-list>
                                            </af-ipv6-neighbor-address>
                                        </af-ipv6-neighbor-address-holder>
                                    </neighbor>
                                </default-vrf>
                            </ipv6-unicast>
                        </ipv6>
                    </address-family>
                </router-bgp>
            </router>
        </rbridge-id>
    </config>
  </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

The following example configures ORF in both send and receive modes.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
```

```
            <running/>
        </target>
        <config>
             <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <neighbor>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>2001:db8:e0ff:783a::4</router-bgp-neighbor-ipv6
-address>
                                        <remote-as>1</remote-as>
                                    </neighbor-ipv6-addr>
                                </neighbor-ipv6s>
                            </neighbor>
                        </router-bgp-attributes>
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
                                        <neighbor>
                                            <af-ipv6-neighbor-address-holder>
                                                <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>2001:db8:e0ff:783a::4</af-ipv6-neighbor-address>
                                                    <af-neighbor-capability>
                                                        <orf>
                                                            <prefixlist>

<prefixlist-status></prefixlist-status>

<prefixlist-send></prefixlist-send>

<prefixlist-receive></prefixlist-receive>
                                                            </prefixlist>
                                                        </orf>
                                                    </af-neighbor-capability>
                                                    <activate></activate>
                                                    <prefix-list>
                                                        <direction-in>

<prefix-list-direction-in-prefix-name>myprefixlist</prefix-list-direction-in-pref
ix-name>

<prefix-list-direction-in></prefix-list-direction-in>
                                                        </direction-in>
                                                    </prefix-list>
                                                </af-ipv6-neighbor-address>
                                            </af-ipv6-neighbor-address-holder>
                                        </neighbor>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
```

```
            </rbridge-id>
         </config>
     </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring BGP4+ confederations

You can configure BGP4+ confederations, composed of multiple subautonomous systems

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
             <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <local-as>65520</local-as>
                            <confederation>
                                <identifier>100</identifier>
                            </confederation>
                        </router-bgp-attributes>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
     </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Defining BGP extended communities

In order to apply a BGP4+ extended community filter, a BGP4+ extended community filter must be defined.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
             <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
```

```
                <ip>
                    <extcommunity-list
xmlns="urn:brocade.com:mgmt:brocade-ip-policy">
                        <extcommunity-list-num>1</extcommunity-list-num>
                        <ext-community-action>permit</ext-community-action>
                        <ext-community-expr>soo 123:2</ext-community-expr>
                    </extcommunity-list>
                </ip>
                <route-map xmlns="urn:brocade.com:mgmt:brocade-ip-policy">
                    <name>extComRmap</name>
                    <action-rm>permit</action-rm>
                    <instance>10</instance>
                    <content>
                        <match>
                            <extcommunity>
                                <extcommunity-num>1 </extcommunity-num>
                            </extcommunity>
                        </match>
                    </content>
                </route-map>
                <route-map xmlns="urn:brocade.com:mgmt:brocade-ip-policy">
                    <name>sendExtComRmap</name>
                    <action-rm>permit</action-rm>
                    <instance>10</instance>
                    <content>
                        <set>
                            <extcommunity>
                                <rt>
                                    <ASN-NN-rt>3:3 </ASN-NN-rt>
                                </rt>
                                <soo>
                                    <ASN-NN-soo>2:2</ASN-NN-soo>
                                </soo>
                            </extcommunity>
                        </set>
                    </content>
                </route-map>
            </rbridge-id>
         </config>
     </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Applying a BGP4+ extended community filter

BGP communities must already be defined.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
             <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
```

```
            <rbridge-id>1</rbridge-id>
            <ip>
                <extcommunity-list
xmlns="urn:brocade.com:mgmt:brocade-ip-policy">
                    <extcommunity-list-num>1</extcommunity-list-num>
                    <ext-community-action>permit</ext-community-action>
                    <ext-community-expr>soo 123:2</ext-community-expr>
                </extcommunity-list>
            </ip>
            <router>
                <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                    <router-bgp-attributes>
                        <local-as>1000</local-as>
                        <neighbor>
                            <neighbor-ipv6s>
                                <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>fe80:4398:ab30:45de::1</router-bgp-neighbor-ipv
6-address>
                                    <remote-as>1001</remote-as>
                                    <update-source>
                                        <ethernet-interface>

<interface-type>tengigabitethernet</interface-type>
                                            <ethernet>1/0/5</ethernet>
                                        </ethernet-interface>
                                    </update-source>
                                </neighbor-ipv6-addr>
                            </neighbor-ipv6s>
                        </neighbor>
                    </router-bgp-attributes>
                    <address-family>
                        <ipv6>
                            <ipv6-unicast>
                                <default-vrf>
                                    <default-vrf-selected></default-vrf-selected>
                                    <neighbor>
                                        <af-ipv6-neighbor-address-holder>
                                            <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>fe80:4398:ab30:45de::1</af-ipv6-neighbor-address>
                                                <send-community>

<send-community-status></send-community-status>
                                                    <both></both>
                                                    <extended></extended>
                                                    <standard></standard>
                                                </send-community>
                                                <activate></activate>
                                                <neighbor-route-map>

<neighbor-route-map-direction-in>

<neighbor-route-map-name-direction-in>map1</neighbor-route-map-name-direction-in>

</neighbor-route-map-direction-in>

<neighbor-route-map-direction-out>
```

```
<neighbor-route-map-name-direction-out>map1</neighbor-route-map-name-direction-ou
t>

</neighbor-route-map-direction-out>
                                              </neighbor-route-map>
                                          </af-ipv6-neighbor-address>
                                      </af-ipv6-neighbor-address-holder>
                                  </neighbor>
                              </default-vrf>
                          </ipv6-unicast>
                      </ipv6>
                  </address-family>
              </router-bgp>
          </router>
      </rbridge-id>
   </config>
 </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Configuring BGP4+ graceful restart

You can configure graceful restart on a routing device.

The following example enables the graceful restart feature.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <local-as>1</local-as>
                            <neighbor>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>1000::1</router-bgp-neighbor-ipv6-address>
                                        <remote-as>2</remote-as>
                                    </neighbor-ipv6-addr>
                                </neighbor-ipv6s>
                            </neighbor>
                        </router-bgp-attributes>
                        <address-family>
                            <ipv6>
                                <ipv6-unicast>
                                    <default-vrf>
                                        <default-vrf-selected></default-vrf-selected>
```

```
                                        <neighbor>
                                            <af-ipv6-neighbor-address-holder>
                                                <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>1000::1</af-ipv6-neighbor-address>
                                                    <activate></activate>
                                                </af-ipv6-neighbor-address>
                                            </af-ipv6-neighbor-address-holder>
                                        </neighbor>
                                        <af-common-cmds-holder>

<client-to-client-reflection></client-to-client-reflection>
                                            <graceful-restart>

<graceful-restart-status></graceful-restart-status>
                                            </graceful-restart>
                                        </af-common-cmds-holder>
                                    </default-vrf>
                                </ipv6-unicast>
                            </ipv6>
                        </address-family>
                    </router-bgp>
                </router>
            </rbridge-id>
         </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

The following example enables the graceful restart feature and sets the purge time to 300 seconds, overwriting the default value.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <local-as>1</local-as>
                            <neighbor>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>1000::1</router-bgp-neighbor-ipv6-address>
                                        <remote-as>2</remote-as>
                                    </neighbor-ipv6-addr>
                                </neighbor-ipv6s>
                            </neighbor>
                        </router-bgp-attributes>
                        <address-family>
```

```
                    <ipv6>
                        <ipv6-unicast>
                            <default-vrf>
                                <default-vrf-selected></default-vrf-selected>
                                <neighbor>
                                    <af-ipv6-neighbor-address-holder>
                                        <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>1000::1</af-ipv6-neighbor-address>
                                            <activate></activate>
                                        </af-ipv6-neighbor-address>
                                    </af-ipv6-neighbor-address-holder>
                                </neighbor>
                                <af-common-cmds-holder>

<client-to-client-reflection></client-to-client-reflection>
                                    <graceful-restart>
                                        <purge-time>300</purge-time>
                                    </graceful-restart>
                                </af-common-cmds-holder>
                            </default-vrf>
                        </ipv6-unicast>
                    </ipv6>
                </address-family>
            </router-bgp>
        </router>
    </rbridge-id>
   </config>
  </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Disabling the BGP AS_PATH check function

You can configure the device such that the AS_PATH check function for routes learned from a specific location is disabled.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="3500">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <router>
                    <router-bgp xmlns="urn:brocade.com:mgmt:brocade-bgp">
                        <router-bgp-attributes>
                            <neighbor>
                                <neighbor-ipv6s>
                                    <neighbor-ipv6-addr>

<router-bgp-neighbor-ipv6-address>2001:db8:e0ff:783a::4</router-bgp-neighbor-ipv6
-address>
```

```
                                    <remote-as>1001</remote-as>
                                </neighbor-ipv6-addr>
                            </neighbor-ipv6s>
                        </neighbor>
                    </router-bgp-attributes>
                    <address-family>
                        <ipv6>
                            <ipv6-unicast>
                                <default-vrf>
                                    <default-vrf-selected></default-vrf-selected>
                                    <neighbor>
                                        <af-ipv6-neighbor-address-holder>
                                            <af-ipv6-neighbor-address>

<af-ipv6-neighbor-address>2001:db8:e0ff:783a::4</af-ipv6-neighbor-address>
                                                <allowas-in>3</allowas-in>
                                            </af-ipv6-neighbor-address>
                                        </af-ipv6-neighbor-address-holder>
                                    </neighbor>
                                </default-vrf>
                            </ipv6-unicast>
                        </ipv6>
                    </address-family>
                </router-bgp>
            </router>
        </rbridge-id>
     </config>
   </edit-config>
</rpc>

<rpc-reply message-id="3500" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Chapter

Configuring and Managing IPv6 ACLs

56

In this chapter

Creating an IPv6 extended ACL

This chapter provides procedures for configuring IPv6 access control lists (ACLs) using the NETCONF interface. Refer to the *Network OS Layer 3 Routing configuration guide* for the following related information:

To create an extended ACL, begin from global-configuration mode.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ipv6-acl xmlns="urn:brocade.com:mgmt:brocade-ipv6-access-list">
                <ipv6>
                    <access-list>
                        <extended>
                            <name>ip_acl_1</name>
                            <seq>
                                <seq-id>10</seq-id>
                                <action>deny</action>
                                <protocol-type>ipv6</protocol-type>

<src-host-any-sip>2001:2002:1234:1::/64</src-host-any-sip>

<dst-host-any-dip>2001:1001:1234:1::/64</dst-host-any-dip>
                                <count></count>
                            </seq>
                        </extended>
                    </access-list>
                </ipv6>
            </ipv6-acl>
        </config>
    </edit-config>
</nc:rpc>
```

```
<rpc-reply message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Applying an IPv6 ACL to a router interface

An ACL does not take effect until you apply it to an interface, using the **access-group** command.

Before you begin, ensure that the ACL you want to apply exists and is configured to filter traffic according to your needs.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fortygigabitethernet>
                    <name>1/0/49</name>
                    <ipv6>
                        <access-group 
xmlns="urn:brocade.com:mgmt:brocade-ipv6-access-list">
                            <ipv6-access-list>ip_acl_1</ipv6-access-list>
                            <ip-direction>in</ip-direction>
                        </access-group>
                    </ipv6>
                </fortygigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Filtering packets based on DSCP values

In extended IPv4 and IPv6 ACLs, you can filter packets based on DSCP values.

```
<?xml version="1.0" ?>
<rpc nc:message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ipv6-acl xmlns="urn:brocade.com:mgmt:brocade-ipv6-access-list">
                <ipv6>
                    <access-list>
                        <extended>
                            <name>netw</name>
                            <seq>
                                <action>deny</action>
```

```
                                <protocol-type>ipv6</protocol-type>
                                <src-host-any-sip>any</src-host-any-sip>
                                <dst-host-any-dip>any</dst-host-any-dip>
                                <dscp>3</dscp>
                            </seq>
                        </extended>
                    </access-list>
                </ipv6>
            </ipv6-acl>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

ACL Counter Statistics

If an ACL rule contains the **count** parameter, you can access statistics for that rule, including number of frames permitted or denied by that rule.

Creating an IPv6 ACL rule enabled for counter statistics

When you create ACL rules, including **count** enables you to display counter statistics.

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ipv6-acl xmlns="urn:brocade.com:mgmt:brocade-ipv6-access-list">
                <ipv6>
                    <access-list>
                        <extended>
                            <name>ip_acl_1</name>
                            <seq>
                                <seq-id>20</seq-id>
                                <action>deny</action>
                                <protocol-type>ipv6</protocol-type>

<src-host-any-sip>2002:2003:1234:1::/64</src-host-any-sip>

<dst-host-any-dip>2001:3001:1234:1::/64</dst-host-any-dip>
                                <count></count>
                            </seq>
                        </extended>
                    </access-list>
                </ipv6>
            </ipv6-acl>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

ACL logs

ACL logging can provide insight into permitted and denied network traffic.

Creating an IPv6 ACL rule enabled for logging

When you create ACL rules for which you want to enable logging, you must include the log parameter.

```
<?xml version="1.0" ?>
<nc:rpc nc:message-id="10" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <ipv6-acl xmlns="urn:brocade.com:mgmt:brocade-ipv6-access-list">
                <ipv6>
                    <access-list>
                        <extended>
                            <name>ip_acl_1</name>
                            <seq>
                                <seq-id>20</seq-id>
                                <action>deny</action>
                                <protocol-type>ipv6</protocol-type>

<src-host-any-sip>2002:2003:1234:1::/64</src-host-any-sip>

<dst-host-any-dip>2001:3001:1234:1::/64</dst-host-any-dip>
                                <log></log>
                            </seq>
                        </extended>
                    </access-list>
                </ipv6>
            </ipv6-acl>
        </config>
    </edit-config>
</nc:rpc>

<rpc-reply message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ok/>
</rpc-reply>
```

Chapter

Configuring Protocol-independent multicast (PIM) 57

In this chapter

PIM configuration with NETCONF overview

This chapter provides procedures for configuring Protocol-independent multicast (PIM) using the NETCONF interface. Refer to the *Network OS Layer 3 Routing Configuration Guide* for the following related information:

- A conceptual overview of PIM
- General guidelines
- Procedures for configuring PIM using the command line interface

Through the NETCONF interface, you can perform the following operations on PIM:

- Use the <edit-config> RPC to configure PIM.
- Use the <get-config> RPC to validate configuration settings.

PIM parameters are defined in the brocade-igmp-snooping YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Configuring PIM multicast router detection

The PIM hello-based multicast router presence detection feature scans the network traffic for incoming PIM hellos.

The PIM topology and VLANs must be configured before activating PIM multicast router detection.

```
<?xml version="1.0"?>
<rpc message-id="3401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <interface>
                        <vlan>
```

```
                        <name>100</name>
                        <ip>
                            <igmp
xmlns="urn:brocade.com:mgmt:brocade-igmp-snooping">
                                <snooping>

<restrict-unknown-multicast></restrict-unknown-multicast>
                                </snooping>
                            </igmp>
                        </ip>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling IGMP snooping on access-layer switches

The following example shows how to enable IGMP snooping on access-layer switches.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>100</name>
                        <ip>
                            <igmp
xmlns="urn:brocade.com:mgmt:brocade-igmp-snooping">
                                <snooping>
                                    <enable></enable>
                                </snooping>
                            </igmp>
                        </ip>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Enabling PIM on aggregation-layer switches

The following example shows how to enable PIM on aggregation-layer switches.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ip>
                    <prefix-list xmlns="urn:brocade.com:mgmt:brocade-ip-policy">
                        <name>prefList1</name>
                        <action-ipp>deny</action-ipp>
                        <prefix-ipp>1.2.0.0/16</prefix-ipp>
                        <le-ipp>30</le-ipp>
                    </prefix-list>
                </ip>
                <router>
                    <hide-pim-holder xmlns="urn:brocade.com:mgmt:brocade-pim">
                        <pim>
                            <rp-address>
                                <rp-ip-addr>10.22.22.22</rp-ip-addr>
                                <prefix-list>prefList1</prefix-list>
                            </rp-address>
                        </pim>
                    </hide-pim-holder>
                </router>
                <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                    <ve>
                        <name>100</name>
                        <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                            <ip-config>
                                <address>
                                    <address>10.1.1.11/24</address>
                                </address>
                            </ip-config>
                            <pim-intf-vlan-cont
xmlns="urn:brocade.com:mgmt:brocade-pim">
                                <pim-int-cmd>
                                    <pim-sparse></pim-sparse>
                                    <pim>
                                        <neighbor-filter>prefList1</neighbor-filter>
                                    </pim>

<mcast-bdry-prefix-list>prefList1</mcast-bdry-prefix-list>
                                </pim-int-cmd>
                            </pim-intf-vlan-cont>
                        </ip>
                    </ve>
                </interface>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="3401" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Restricting unknown multicast

The following example shows how to enable the restrict-unknown-multicast feature.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>100</name>
                        <ip>
                            <igmp
xmlns="urn:brocade.com:mgmt:brocade-igmp-snooping">
                                <snooping>

<restrict-unknown-multicast></restrict-unknown-multicast>
                                </snooping>
                            </igmp>
                        </ip>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Chapter

Configuring Dual Stack Support

58

In this chapter

Dual Stack support configuration with NETCONF overview

This chapter provides procedures for configuring the Dual stack support using the NETCONF interface. Refer to the *Network OS Layer 3 Routing Configuration Guide* for more information:

Through the NETCONF interface, you can perform the following operations on Dual stack:

- Use the <edit-config> RPC to configure Dual stack.
- Use the <get-config> RPC to validate configuration settings.

For details, refer to the *Network OS YANG Reference Manual*.

Configuring a global IPv6 address for an interface

The following example is to configure a global IPv6 address for an interface, including a manually configured interface ID.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
      <target>
         <running/>
      </target>
```

```
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fortygigabitethernet>
                    <name>1/0/49</name>
                    <ipv6>
                        <ipv6-config
xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <address>
                                <ipv6-address>

<address>2001:200:12d:1300:240:d0ff:fe48:4672/64</address>
                                </ipv6-address>
                            </address>
                        </ipv6-config>
                    </ipv6>
                </fortygigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

The following example is to configure a global IPv6 address with an automatically computed EUI-64 interface ID in the low-order 64 bits.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fortygigabitethernet>
                    <name>1/0/49</name>
                    <ipv6>
                        <ipv6-config
xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <address>
                                <ipv6-address>
                                    <address>2001:db8:12d:1300::/64</address>
                                    <eui-config>
                                        <eui64></eui64>
                                    </eui-config>
                                </ipv6-address>
                            </address>
                        </ipv6-config>
                    </ipv6>
                </fortygigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Configuring a link-local IPv6 address

You can configure a link-local IPv6 address without configuring a global or site-local address for the interface.

The following example is to configure an automatically computed address:

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fortygigabitethernet>
                    <name>1/0/49</name>
                    <ipv6>
                        <ipv6-config
xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <address>
                                <use-link-local-only></use-link-local-only>
                            </address>
                        </ipv6-config>
                    </ipv6>
                </fortygigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

The following example is to configure an explicit address:

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fortygigabitethernet>
                    <name>1/0/49</name>
                    <ipv6>
                        <ipv6-config
xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <address>
                                <link-local-config>

<link-local-address>fe80::240:d0ff:fe48:4672</link-local-address>
                                    <link-local></link-local>
                                </link-local-config>
                            </address>
                        </ipv6-config>
                    </ipv6>
                </fortygigabitethernet>
```

```
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring an IPv6 anycast address

The following example is to configure an anycast address on an interface.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fortygigabitethernet>
                    <name>1/0/49</name>
                    <ipv6>
                        <ipv6-config
xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <address>
                                <ipv6-address>
                                    <address>2002::6/64</address>
                                    <anycast></anycast>
                                </ipv6-address>
                            </address>
                        </ipv6-config>
                    </ipv6>
                </fortygigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring IPv4 and IPv6 protocol stacks

Do the following to configure an interface to support both IPv4 and IPv6 protocol stacks.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
```

```
                <fortygigabitethernet>
                    <name>1/0/49</name>
                    <ip xmlns="urn:brocade.com:mgmt:brocade-ip-config">
                        <ip-config>
                            <address>
                                <address>101.1.1.1/24</address>
                            </address>
                        </ip-config>
                    </ip>
                    <ipv6>
                        <ipv6-config
xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <address>
                                <ipv6-address>
                                    <address>2001:db8:12d:1300::/64</address>
                                    <eui-config>
                                        <eui64></eui64>
                                    </eui-config>
                                </ipv6-address>
                            </address>
                        </ipv6-config>
                    </ipv6>
                </fortygigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Configuring an IPv6 address family

Do the following to configure an IPv6 address family.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <vrf xmlns="urn:brocade.com:mgmt:brocade-vrf">
                    <vrf-name>red</vrf-name>
                    <address-family>
                        <ipv6>
                            <unicast></unicast>
                        </ipv6>
                    </address-family>
                </vrf>
            </rbbridge-id>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
                    <vrf>
```

```
                        <forwarding>red</forwarding>
                    </vrf>
                    <ipv6>
                        <ipv6-config
xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <address>
                                <ipv6-address>
                                    <address>1111::1111/64</address>
                                </ipv6-address>
                            </address>
                        </ipv6-config>
                    </ipv6>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring static IPv6 routes

The following example is to configure a static IPv6 route with a destination and next-hop gateway address as a global unicast interface.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                    <ipv6>
                        <route xmlns="urn:brocade.com:mgmt:brocade-ipv6-rtm">
                            <static-route-nh>

<static-route-dest>3ffe:abcd::/64</static-route-dest>

<static-route-next-hop>2001::11:1234</static-route-next-hop>
                        </static-route-nh>
                    </route>
                </ipv6>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

The following example is to configure a static IPv6 route with a destination and next-hop gateway address as a global unicast interface with a metric of 10 and an administrative distance of 110.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ipv6>
                    <route xmlns="urn:brocade.com:mgmt:brocade-ipv6-rtm">
                        <static-route-nh>
                            <static-route-dest>3ffe:abcd::/64</static-route-dest>

<static-route-next-hop>2001::11:1234</static-route-next-hop>
                            <route-attributes>
                                <metric>10</metric>
                                <distance>110</distance>
                            </route-attributes>
                        </static-route-nh>
                    </route>
                </ipv6>
            <rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

The following example is to configure a static IPv6 route with a destination and next-hop gateway address as a link-local address on an Ethernet interface.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ipv6>
                    <route xmlns="urn:brocade.com:mgmt:brocade-ipv6-rtm">
                        <link-local-static-route-nh>

<link-local-static-route-dest>3001:1234::/64</link-local-static-route-dest>
                            <link-local-nexthop>fe80::1234</link-local-nexthop>

<link-local-route-oif-type>fortygigabitethernet</link-local-route-oif-type>

<link-local-route-oif-name>2/0/49</link-local-route-oif-name>
                        </link-local-static-route-nh>
                    </route>
                </ipv6>
            <rbridge-id>
        </config>
    </edit-config>
```

```
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

The following example is to configure a static IPv6 route with a destination global unicast address and next-hop gateway address.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ipv6>
                    <route xmlns="urn:brocade.com:mgmt:brocade-ipv6-rtm">
                        <static-route-nh>
                            <static-route-dest>2001:db8::0/32</static-route-dest>

<static-route-next-hop>2001:db8:0:ee44::1</static-route-next-hop>
                        </static-route-nh>
                    </route>
                </ipv6>
            <rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

The following example is to configure a static IPv6 route with a destination and next-hop gateway address and cause packets to those addresses to be dropped by shunting them to the "null0" interface.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ipv6>
                    <route xmlns="urn:brocade.com:mgmt:brocade-ipv6-rtm">
                        <static-route-oif>

<static-route-dest>2fe0:1234:5678::/64</static-route-dest>
                            <static-route-oif-type>null</static-route-oif-type>
                            <static-route-oif-name>0</static-route-oif-name>
                        </static-route-oif>
                    </route>
                </ipv6>
            <rbridge-id>
        </config>
```

```
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

The following example is to configure a static route and next-hop gateway on a virtual Ethernet interface.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ipv6>
                    <route xmlns="urn:brocade.com:mgmt:brocade-ipv6-rtm">
                        <static-route-oif>

<static-route-dest>2fe0:1234:5678::/64</static-route-dest>
                            <static-route-oif-type>ve</static-route-oif-type>
                            <static-route-oif-name>500</static-route-oif-name>
                        </static-route-oif>
                    </route>
                </ipv6>
            <rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

The following example is to configure a static route in the default VRF instance.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <ipv6>
                    <route xmlns="urn:brocade.com:mgmt:brocade-ipv6-rtm">
                        <static-route-nh>
                            <static-route-dest>3ffe:1002::/64</static-route-dest>
                            <static-route-next-hop>2001::100</static-route-next-hop>
                        </static-route-nh>
                    </route>
                </ipv6>
            <rbridge-id>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Changing IPv6 MTU

The following example is to change the IPv6 MTU on an interface.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
                    <ipv6>
                        <ipv6-config
xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                            <address>
                                <ipv6-address>

<address>2001:200:12d:1300:240:d0ff:fe48:4672/64</address>
                                </ipv6-address>
                            </address>
                            <mtu>1280</mtu>
                        </ipv6-config>
                    </ipv6>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Setting Neighbor Solicitation parameters for DAD

The following example shows how to change the number of NS messages sent on Ethernet interface 1/0/1 to 3 and the interval between the transmission of the two messages to 4 seconds,

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
```

```
                        <ipv6>
                            <ipv6-nd-ra xmlns="urn:brocade.com:mgmt:brocade-ipv6-nd-ra">
                                <ipv6-intf-cmds>
                                    <nd>
                                        <dad>
                                            <attempts>3</attempts>
                                            <time>4</time>
                                        </dad>
                                    </nd>
                                </ipv6-intf-cmds>
                            </ipv6-nd-ra>
                        </ipv6>
                    </tengigabitethernet>
                </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="14">
    <ok/>
</rpc-reply>
```

Setting IPv6 Router Advertisement parameters

You can adjust the following parameters for Router Advertisement (RA) messages:

- The interval (in seconds) at which an interface sends RA messages. By default, an interface sends an RA message randomly, every 200 to 600 seconds.
- The "router lifetime" value, which is included in RA messages sent from a particular interface. The value (in seconds) indicates whether the router is advertised as a default router on this interface.

If you set the value of this parameter to 0, the router is not advertised as a default router on an interface. If you set this parameter to a value that is not 0, the router is advertised as a default router on this interface. By default, the router lifetime value included in router advertisement messages sent from an interface is 1800 seconds.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
                    <ipv6>
                        <ipv6-nd-ra xmlns="urn:brocade.com:mgmt:brocade-ipv6-nd-ra">
                            <ipv6-intf-cmds>
                                <nd>
                                    <mtu>2400</mtu>
                                    <hoplimit>32</hoplimit>
                                    <ra-interval>
                                        <max-interval>1200</max-interval>
                                        <min>400</min>
                                    </ra-interval>
                                </nd>
```

```
                        </ipv6-intf-cmds>
                    </ipv6-nd-ra>
                </ipv6>
            </tengigabitethernet>
        </interface>
    </config>
</edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Controlling prefixes advertised in IPv6 Router Advertisement messages

By default, Router Advertisement (RA) messages include prefixes configured as addresses on interfaces. You can set exactly which prefixes are included in RA messages.

The following example is to configure which prefixes should be included in RA messages.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
                    <ipv6>
                        <ipv6-nd-ra xmlns="urn:brocade.com:mgmt:brocade-ipv6-nd-ra">
                            <ipv6-intf-cmds>
                                <nd>
                                    <prefix>

<prefix-ipv6-address>2ffe:1111::/64</prefix-ipv6-address>
                                    </prefix>
                                </nd>
                            </ipv6-intf-cmds>
                        </ipv6-nd-ra>
                    </ipv6>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Setting flags in IPv6 Router Advertisement messages

By default, the Managed Address Configuration and Other Stateful Configuration flags are not set in router advertisement messages. The following example is to set these flags in router advertisement messages sent from an interface.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
                    <ipv6>
                        <ipv6-nd-ra xmlns="urn:brocade.com:mgmt:brocade-ipv6-nd-ra">
                            <ipv6-intf-cmds>
                                <nd>
                                    <managed-config-flag></managed-config-flag>
                                    <other-config-flag></other-config-flag>
                                </nd>
                            </ipv6-intf-cmds>
                        </ipv6-nd-ra>
                    </ipv6>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring IPv6 static neighbor entries

In some cases a neighbor cannot be reached by means of Neighbor Discovery. To resolve this you can add a static entry to the ND cache, causing a neighbor to reachable at all times. (A static IPv6 ND entry is like a static IPv4 ARP entry.)

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/1</name>
                    <ipv6>
                        <ipv6-nd-ra xmlns="urn:brocade.com:mgmt:brocade-ipv6-nd-ra">
                            <ipv6-intf-cmds>
                                <neighbor>
                                    <ipv6-address>2001:db8:2678::2</ipv6-address>
```

```
<hardware-address>0000.002b.8641</hardware-address>
                            </neighbor>
                        </ipv6-intf-cmds>
                    </ipv6-nd-ra>
                </ipv6>
            </tengigabitethernet>
        </interface>
    </config>
</edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Enabling MLD snooping globally

The following example is to enable MLD snooping globally.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <mld-snooping xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                <ipv6>
                    <mld>
                        <snooping>
                            <enable></enable>
                        </snooping>
                    </mld>
                </ipv6>
            </mld-snooping>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Enabling MLD snooping at the interface level

The following example is to enable MLD snooping at interface level.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
```

```
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ipv6>
                            <mldVlan
xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                                <snooping>
                                    <enable></enable>
                                </snooping>
                            </mldVlan>
                        </ipv6>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Enabling MLD querier functionality on a VLAN

The following example is to enable the MLD querier functionality on a VLAN.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ipv6>
                            <mldVlan
xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                                <snooping>
                                    <querier>
                                        <qenable></qenable>
                                    </querier>
                                </snooping>
                            </mldVlan>
                        </ipv6>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring an MLD static group on a VLAN

To enable static group on a VLAN interface, select a multicast address to be joined, as well as a physical interface, as in the following example:

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ipv6>
                            <mldVlan
xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                                <static-group>
                                    <mcast-address>ff1e::1</mcast-address>
                                    <interface>interface</interface>
                                    <if-type>FortyGigabitEthernet</if-type>
                                    <value>1/2/1</value>
                                </static-group>
                            </mldVlan>
                        </ipv6>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Enabling MLD fast-leave on a VLAN

MLD fast-leave allows a receiver to move from one multicast group to another instantly if it is the only receiver on the segment that is subscribed to a group. This minimizes the leave latency of group memberships on an interface, as the device does not send group-specific queries. As a result, the group entry is removed from the multicast forwarding table as soon as a group leave message is received.

The following example is to enable MLD fast-leave on a VLAN.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
```

```
                <vlan>
                    <name>2000</name>
                    <ipv6>
                        <mldVlan
xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                            <snooping>
                                <fast-leave></fast-leave>
                            </snooping>
                        </mldVlan>
                    </ipv6>
                </vlan>
            </interface>
        </interface-vlan>
    </config>
  </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring the MLD query interval

You can configure the frequency at which MLD host query messages are sent. Larger values cause queries to be sent less often. The default is 125 seconds.

The following example is to set the MLD query interval.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ipv6>
                            <mldVlan
xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                                <snooping>
                                    <query-interval>1200</query-interval>
                                </snooping>
                            </mldVlan>
                        </ipv6>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring the MLD last-member query interval

The following example shows how set the MLD last-member query interval.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ipv6>
                            <mldVlan
xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                                <snooping>

<last-member-query-interval>1500</last-member-query-interval>
                                </snooping>
                            </mldVlan>
                        </ipv6>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring the MLD last-member query count

You can set the number of times that an MLD query is sent in response to a host leave message. This is the number of times, separated by the last-member query-response interval (configured by the ipv6 mld last-member-query-interval command), that an MLD query is sent in response to a host leave message from the last known active host on the subnet. The default is 2.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ipv6>
                            <mldVlan
xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                                <snooping>
```

```
<last-member-query-count>5</last-member-query-count>
                                </snooping>
                            </mldVlan>
                        </ipv6>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring the MLD query maximum response time

The following example is to configure the maximum response time for IPv6 MLDv1 snooping MLD queries for a specific VLAN interface.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ipv6>
                            <mldVlan
xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                                <snooping>

<query-max-response-time>15</query-max-response-time>
                                </snooping>
                            </mldVlan>
                        </ipv6>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring the MLD snooping robustness variable

The following example is to change the default robustness variable on a VLAN.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ipv6>
                            <mldVlan
xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                                <snooping>
                                    <robustness-variable>7</robustness-variable>
                                </snooping>
                            </mldVlan>
                        </ipv6>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring the MLD startup query count

The following example is to change the startup-query interval on a VLAN.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ipv6>
                            <mldVlan
xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                                <snooping>
                                    <startup-query-count>2</startup-query-count>
                                </snooping>
                            </mldVlan>
                        </ipv6>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
```

```
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Configuring the MLD startup query interval

The following example is to configure the startup-query interval on a VLAN.

```
<?xml version="1.0" ?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>2000</name>
                        <ipv6>
                            <mldVlan
xmlns="urn:brocade.com:mgmt:brocade-mld-snooping">
                                <snooping>

<startup-query-interval>2</startup-query-interval>
                                </snooping>
                            </mldVlan>
                        </ipv6>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Chapter

Configuring IGMP 59

In this chapter

IGMP configuration with NETCONF overview

This chapter provides procedures for configuring Internet Group Management Protocol (IGMP) using NETCONF RPCs. Refer to the *Network OS Layer 2 Switching Configuration Guide* for the following related information:

- An overview of IGMP, including descriptions of how active IGMP snooping works, how IGMP mitigates the effects of multicast routing, and how IGMP is handled over a LAG or vLAG
- How to monitor IGMP snooping. (You cannot monitor IGMP snooping using the NETCONF interface.)
- Scalability information about IGMP in fabric cluster mode

Using the NETCONF interface, you can perform the following IGMP-related operations:

- Use the <edit-config> remote procedure call (RPC) to configure IGMP snooping and the IGMP snooping querier.
- Use the <get-config> RPC to verify all or part of the IGMP configuration.

IGMP parameters are defined in the brocade-igmp-snooping YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*. For definitions and explanations of all IGMP parameters, refer to the brocade-igmp-snooping.yang file.

Configuring IGMP snooping

By default, IGMP snooping is globally disabled on all VLAN interfaces.

Use the following procedure to configure IGMP on a DCB/FCoE switch.

The following example enables IGMP snooping for VLAN 10.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3000" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
      <target>
         <running/>
      </target>
      <config>
```

```
            <igmp-snooping xmlns="urn:brocade.com:mgmt:brocade-igmp-snooping">
                <ip>
                    <igmp>
                        <snooping>
                            <enable/>
                        </snooping>
                    </igmp>
                </ip>
            </igmp-snooping>
            <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
                <interface>
                    <vlan>
                        <name>10</name>
                            <ip>
                                <igmp
                        xmlns="urn:brocade.com:mgmt:brocade-igmp-snooping">
                                    <snooping>
                                        <querier>
                                            <qenable/>
                                        </querier>
                                    </snooping>
                                </igmp>
                            </ip>
                    </vlan>
                </interface>
            </interface-vlan>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="3000" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring IGMP snooping querier

If your multicast traffic is not routed because Protocol Independent Multicast (PIM) and IGMP are not configured, use the IGMP snooping querier in a VLAN.

IGMP snooping querier sends out IGMP queries to trigger IGMP responses from switches that wish to receive IP multicast traffic. IGMP snooping listens for these responses to map the appropriate forwarding addresses.

Use the following procedure to map forwarding addresses to the appropriate interfaces.

In this example, this basic NETCONF request would be repeated for each of the attributes. This example sets the <query-interval> value.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<edit-config>
    <target>
        <running/>
    </target>
    <config>
        <interface-vlan xmlns="urn:brocade.com:mgmt:brocade-interface">
            <interface>
                <vlan>
```

```
                    <name>25</name>
                    <ip>
                        <igmp
                         xmlns="urn:brocade.com:mgmt:brocade-igmp-snooping">
                           <query-interval>125</query-interval>
                        </igmp>
                    </ip>
                </vlan>
            </interface>
        </interface-vlan>
    </config>
</edit-config>
</rpc>

<rpc-reply message-id="3001" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
```

Monitoring IGMP snooping

Monitoring the performance of your IGMP traffic allows you to diagnose any potential issues on your switch. This helps you utilize bandwidth more efficiently by setting the switch to forward IP multicast traffic only to connected hosts that request multicast traffic.

Use the <get-config> RPC to validate configuration settings. IGMP parameters are defined in the brocade-igmp-snooping YANG module. For details, refer to the *Network OS YANG Reference Manual*.

Chapter

Configuring DHCP Relay

60

In this chapter

DHCP Relay configuration with NETCONF overview

This chapter provides procedures for configuring DHCP Relay using NETCONF RPCs. Refer to the *Network OS Layer 3 Routing Configuration Guide* for the following related information:

- An overview of DHCP Relay, including descriptions of how active DHCP Relay functions
- How to monitor DHCP Relay

Using the NETCONF interface, you can perform the following DHCP Relay-related operations:

- Use the <edit-config> remote procedure call (RPC) to configure DHCP Relay.
- Use the <get-config> RPC to verify all or part of the DHCP Relay configuration.

DHCP Relay parameters are defined in the brocade-dhcp YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*.

Configuring DHCP Relay

The following are considerations and limitations when configuring the IP DHCP Relay agent:

- You can configure the feature in VCS mode.
- You can configure up to four DHCP server IP addresses per interface. When multiple addresses are configured, the relay agent relays the packets to all server addresses.
- The DHCP server and clients it communicates with can be attached to different Virtual Forwarding and Routing (VRF) instances. When clients and the DHCP server are on different VRF instances, use the <server-vrf-name> node with the <relay-ip-addr> node, where <server-vrf-name> is the VRF where the DHCP server is located. For more information on VRF support for the IP DHCP Relay, refer to Chapter 52, “Configuring VRF”.

The following is an example of configuring an IP DHCP Relay address on a 10-gigabit Ethernet interface. The 10-gigabit Ethernet interface is modified with the DHCP Relay container's <relay-ip-addr> node to set the IP address.

```
<rpc message-id="1202" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
      <target>
```

```
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <ip>
                        <interface-te-dhcp-conf 
xmlns="urn:brocade.com:mgmt:brocade-dhcp">
                            <dhcp>
                                <relay>
                                    <servers>
                                        <relay-ip-addr>1.1.1.1</relay-ip-addr>
                                        <server-vrf-name>mgmt-vrf</server-vrf-name>
                                    </servers>
                                    <gateway>1.1.1.1</gateway>
                                </relay>
                            </dhcp>
                        </interface-te-dhcp-conf>
                    </ip>
                </tengigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1202" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

The following is an example of configuring the same IP DHCP Relay address, except this time on VE interface 100.

```
<rpc message-id="1202" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <ve>
                    <name>100</name>
                    <dhcp>
                        <relay>
                            <servers>
                                <relay-ip-addr>1.1.1.1</relay-ip-addr>
                                <server-vrf-name>mgmt-vrf</server-vrf-name>
                            </servers>
                            <gateway>1.1.1.1</gateway>
                        </relay>
                    </dhcp>
                </ve>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1202" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Removing the DHCP Relay address

To remove the IP DHCP Relay address, use the standard delete process for NETCONF.

```
<?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:brocade.com:mgmt:brocade-interface">
                <dhcp xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
                         operation="delete"/></dhcp>
            </protocol>
        </config>
    </edit-config>
</rpc>

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

Verifying configuration information

For detailed information on retrieving configuration information, refer to “Retrieving configuration data” on page 11.

Chapter

Configuring IP DHCPv6 Relay 61

In this chapter

IP DHCPv6 overview

This chapter provides procedures for configuring the IP DHCPv6. Refer to the *Network OS Layer 3 Configuration Guide* for the following related information:

- A conceptual overview of IP DHCPv6
- General guidelines
- Procedures for configuring IP DHCPv6 using the command line interface

Through the NETCONF interface, you can perform the following operations on IP DHCPv6:

- Use the <edit-config> RPC to configure IP DHCPv6.
- Use the <get-config> RPC to validate configuration settings.

IP DHCPv6 parameters are defined in the brocade-IP-DHCPv6 YANG module. For details, refer to the *Network OS YANG Reference Manual*

Configuring IP DHCPv6 Relay

The following example shows how to configure IP DHCPv6 Relay.

Example 1

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <ve>
                    <name>1</name>
                    <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                        <interface-ve-dhcp-conf 
xmlns="urn:brocade.com:mgmt:brocade-dhcpv6">
                            <dhcp>
                                <relay>
                                    <servers>
                                        <relay-ip-addr>2000::1</relay-ip-addr>
```

```
                                    </servers>
                                </relay>
                            </dhcp>
                        </interface-ve-dhcp-conf>
                    </ipv6>
                    <shutdown
xmlns="urn:brocade.com:mgmt:brocade-ip-config"></shutdown>
                </ve>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Example 2

```
<?xml version="1.0" ?>
<rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <fortygigabitethernet>
                    <name>1/0/49</name>
                    <ipv6 xmlns="urn:brocade.com:mgmt:brocade-ipv6-config">
                        <interface-ve-dhcp-conf
xmlns="urn:brocade.com:mgmt:brocade-dhcpv6">
                            <dhcp>
                                <relay>
                                    <servers>
                                        <relay-ip-addr>2000::1</relay-ip-addr>
                                    </servers>
                                </relay>
                            </dhcp>
                        </interface-ve-dhcp-conf>
                    </ipv6>
                    <shutdown
xmlns="urn:brocade.com:mgmt:brocade-ip-config"></shutdown>
                </fortygigabitethernet>
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="10">
    <ok/>
</rpc-reply>
```

Chapter

Configuring Monitoring and Alerting Policy Suite 62

In this chapter

Monitoring and Alerting Policy Suite overview

This chapter provides procedures for configuring Monitoring and Alerting Policy Suite (MAPS) using NETCONF RPCs. Refer to the *Network OS MAPS Administrator's Guide* for the following related information:

- An overview of MAPS

Using the NETCONF interface, you can perform the following DHCP Relay-related operations:

- Use the <edit-config> remote procedure call (RPC) to configure MAPS.
- Use the <get-config> RPC to verify all or part of the MAPS configuration.

MAPS parameters are defined in the brocade-maps YANG module. For a structural map of the YANG module, refer to the *Network OS YANG Reference Manual*.

Enabling MAPS

The following example shows how to enable MAPS.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <maps xmlns="urn:brocade.com:mgmt:brocade-maps">
                    <enable>
                        <policy>dflt_conservative_policy</policy>
                        <actions>RASLOG</actions>
                    </enable>
                </maps>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>
```

```
<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Configuring MAPS alert targets

The following example shows how to configure MAPS alert targets.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <rbridge-id xmlns="urn:brocade.com:mgmt:brocade-rbridge">
                <rbridge-id>1</rbridge-id>
                <maps xmlns="urn:brocade.com:mgmt:brocade-maps">
                    <email>
                        <email-list>
                            <email>admin@abc123.com</email>
                        </email-list>
                    </email>
                    <relay>
                        <hostip>10.25.248.25</hostip>
                        <domainname>abc123.com</domainname>
                    </relay>
                </maps>
            </rbridge-id>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Monitoring iSCSI storage

The following example shows how to designate a tengigabitEthernet port as iSCSI storage for MAPS.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
        <target>
            <running/>
        </target>
        <config>
            <interface xmlns="urn:brocade.com:mgmt:brocade-interface">
                <tengigabitethernet>
                    <name>1/0/5</name>
                    <connectivity xmlns="urn:brocade.com:mgmt:brocade-maps">
                        <deviceconnectivity>iSCSI</deviceconnectivity>
                    </connectivity>
                </tengigabitethernet>
```

```
            </interface>
        </config>
    </edit-config>
</rpc>

<rpc-reply message-id="1904" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <ok/>
</rpc-reply>
```

Obtaining MAPS details

The following Custom RPCs exist for returning MAPS information:

- maps-get-all-policy returns information about the MAPS policy.
- maps-get-rules returns information about the MAPS rules.

The following Custom RPC retrieves the existing MAPS Policies.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:message-id="3">
    <maps-get-all-policy xmlns="urn:brocade.com:mgmt:brocade-maps-ext">
        <rbridge-id>5</rbridge-id>
    </maps-get-all-policy>
</nc:rpc>

<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="3">
    <policy xmlns="urn:brocade.com:mgmt:brocade-maps-ext">
        <policyname>dflt_conservative_policy</policyname>
        <policyname>dflt_aggressive_policy</policyname>
        <policyname>dflt_moderate_policy</policyname>
    </policy>
</rpc-reply>
```

The following Custom RPC retrieves the existing MAPS rules details.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:message-id="3">
    <maps-get-rules xmlns="urn:brocade.com:mgmt:brocade-maps-ext">
        <rbridge-id>7</rbridge-id>
    </maps-get-rules>
</nc:rpc>

[2015-05-22 16:27:36] RECV admin@10.20.237.7
<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="3">
    <rules xmlns="urn:brocade.com:mgmt:brocade-maps-ext">
        <rulename>defALL_ETH_PORTS_CRCALN_6</rulename>
        <groupname>ALL_ETH_PORTS</groupname>
        <monitor>CRCALN</monitor>
        <op>></op>
        <value>6</value>
        <action>RASLOG,SNMP,EMAIL</action>
        <timebase>MIN</timebase>
        <policyname>dflt_aggressive_policy</policyname>
    </rules>
    <rules xmlns="urn:brocade.com:mgmt:brocade-maps-ext">
        <rulename>defALL_ETH_PORTS_RX_SYM_ERR_3</rulename>
        <groupname>ALL_ETH_PORTS</groupname>
        <monitor>RX_SYM_ERR</monitor>
        <op>></op>
```

```
            <value>3</value>
            <action>RASLOG,SNMP,EMAIL</action>
            <timebase>MIN</timebase>
            <policyname>dflt_aggressive_policy</policyname>
        </rules>
        <rules xmlns="urn:brocade.com:mgmt:brocade-maps-ext">
            <rulename>defALL_ETH_PORTS_RX_ABN_FRAME_6</rulename>
            <groupname>ALL_ETH_PORTS</groupname>
            <monitor>RX_ABN_FRAME</monitor>
            <op>></op>
            <value>6</value>
            <action>RASLOG,SNMP,EMAIL</action>
            <timebase>MIN</timebase>
            <policyname>dflt_aggressive_policy</policyname>
        </rules>
        <rules xmlns="urn:brocade.com:mgmt:brocade-maps-ext">
            <rulename>defCHASSISETH_MGMT_PORT_STATE_DOWN</rulename>
            <groupname>CHASSIS</groupname>
            <monitor>ETH_MGMT_PORT_STATE</monitor>
            <op>==</op>
            <value>DOWN</value>
            <action>RASLOG,SNMP,EMAIL</action>
            <timebase>NONE</timebase>
            <policyname>dflt_moderate_policy</policyname>
        </rules>
        <rules xmlns="urn:brocade.com:mgmt:brocade-maps-ext">
            <rulename>defCHASSISETH_MGMT_PORT_STATE_UP</rulename>
            <groupname>CHASSIS</groupname>
            <monitor>ETH_MGMT_PORT_STATE</monitor>
            <op>==</op>
            <value>UP</value>
            <action>RASLOG,SNMP,EMAIL</action>
            <timebase>NONE</timebase>
            <policyname>dflt_moderate_policy</policyname>
        </rules>
    </rpc-reply>
```

Section

I

Appendixes

This section contains the following appendix:

Appendix

Managing NETCONF

A

In this appendix

Viewing NETCONF client capabilities

You can view the NETCONF client capabilities for all active sessions through the NETCONF interface or using the Network OS CLI. The session-ID, logon name of the user of the NETCONF client session, host IP address, and login time are always returned. The application vendor name, application product name, version number, and identity of the client are also returned if these values were advertized in the client capabilities at the start of the session.

To view NETCONF client capabilities using the CLI, in privilege-exec mode, enter the **show netconf client-capabilities** command.

The following example shows two client sessions. The first client session has provided client capabilities in its <hello> message to the server at the start of the session. The second client has not provided this information.

```
switch# show netconf client-capabilities
Session Id     : 10
User name      : root
Vendor         : Brocade
Product        : Brocade Network Advisor
Version        : 9.1.0 Build 123
Client user    : admin-user
Host IP        : 10.24.65.8
Login time     : 2011-08-18T08:54:24Z

Session Id     : 11
User name      : root
Vendor         : Not Available
Product        : Not Available
Version        : Not Available
Client user    : Not Available
Host IP        : 10.24.65.8
Login time     : 2011-08-18T08:54:24Z
```

To obtain NETCONF client capabilities using the NETCONF interface, issue the <get-netconf-client-capabilities> custom RPC located in the urn:brocade.com:mgmt:brocade-netconf-ext namespace.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="206">
    <get-netconf-client-capabilities
        xmlns="urn:brocade.com:mgmt:brocade-netconf-ext"/>
</rpc>
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="206">
    <session xmlns="urn:brocade.com:mgmt:brocade-netconf-ext">
        <session-id>10</session-id>
        <user-name>root</user-name>
        <vendor>Brocade</vendor>
        <product>Brocade Network Advisor</product>
        <version>9.1.0 Build 123</version>
        <identity>admin-user</identity>
        <host-ip>10.24.65.8</host-ip>
        <time>2011-08-18T08:54:24Z</time>
    </session>
    <session xmlns="urn:brocade.com:mgmt:brocade-netconf-ext">
        <session-id>11</session-id>
        <user-name>root</user-name>
        <host-ip>10.24.65.8</host-ip>
        <time>2011-08-18T08:54:24Z</time>
    </session>
</rpc-reply>
```

Viewing NETCONF statistics and session information

To view NETCONF statistics and session information, use the **show netconf-state** command. Using this command, you can view the following kinds of information:

- Capabilities
- Datastores
- Schemas
- Sessions
- Statistics

NOTE
You cannot view NETCONF statistics and session information using the NETCONF interface.

To view NETCONF capabilities provided by the server, in privileged-EXEC mode, enter the **show netconf-state capabilities** command.

```
switch# show netconf-state capabilities
netconf-state capabilities capability urn:ietf:params:netconf:base:1.0
netconf-state capabilities capability
urn:ietf:params:netconf:capability:writable-running:1.0
netconf-state capabilities capability
urn:ietf:params:netconf:capability:startup:1.0
netconf-state capabilities capability
urn:ietf:params:netconf:capability:xpath:1.0
netconf-state capabilities capability
urn:ietf:params:netconf:capability:validate:1.0
netconf-state capabilities capability http://tail-f.com/ns/netconf/actions/1.0
netconf-state capabilities capability
http://tail-f.com/ns/aaa/1.1?revision=2010-06-17&module=tailf-aaa
netconf-state capabilities capability
urn:brocade.com:mgmt:brocade-aaa?revision=2010-10-21&module=brocade-aaa
(output truncated)
```

To view the NETCONF datastores on the NETCONF server and related locking information, enter the **show netconf-state datastores** command.

```
switch# show netconf-state datastores
         LOCKED                  LOCKED
         BY       LOCKED  LOCK   BY       LOCKED           LOCKED
NAME     SESSION  TIME    ID     SESSION  TIME    SELECT   NODE
-----------------------------------------------------------------
running  -        -
startup  -        -
```

To view the data models supported by the NETCONF server, enter the **show netconf-state schemas** command.

```
switch# show netconf-state schemas
```

To view the currently active NETCONF sessions, enter the **show netconf-state sessions** command.

```
switch# show netconf-state sessions
netconf-state sessions session 6
transport cli-console
username admin
source-host 127.0.0.1
login-time 2011-09-05T11:29:31Z
netconf-state sessions session 9
transport netconf-ssh
username root
source-host 172.21.132.67
login-time 2011-09-05T11:50:33Z
in-rpcs 0
in-bad-rpcs 0
out-rpc-errors 0
out-notifications 0
```

To view NETCONF server statistics, enter the **show netconf-state statistics** command.

```
switch# show netconf-state statistics
netconf-state statistics netconf-start-time 2012-04-27T09:12:09Z
netconf-state statistics in-bad-hellos 0
netconf-state statistics in-sessions 94
netconf-state statistics dropped-sessions 78
netconf-state statistics in-rpcs 800
netconf-state statistics in-bad-rpcs 59
netconf-state statistics out-rpc-errors 59
netconf-state statistics out-notifications 0
```