NORTEL
NETWORKS

Nortel Networks Multiservice Switch

7400/15000/20000

IP VPN Configuration Management

NN10600-582

Nortel Networks Multiservice Switch 7400/15000/20000

IP VPN Configuration Management

Publication: NN10600-582
Document status: Standard
Document version: 6.1S2
Document date: November 2004

Printed in Canada

Publication history

November 2004

6.1S2 Standard
General availability. Contains information on Nortel Networks Multiservice Switch 7400, 15000, and 20000 for the PCR6.1 release.

Contents

About this document 15

Who should read this document and why 15

What you need to know 15

What's new in this document 15

Control Plane Protection (CPP) 17

Carrier's Carrier 17

CP support and CP/VPNXc warm standby equipment protection 17

Ethernet access media 17

MD5 Authentication 17

Multi-hop EBGP 18

Route target export policy 18

Router model enhancements 18

Virtual router redundancy protocol (VRRP) on 4-port Gigabit Ethernet, 4-port 10/100 BaseT Ethernet, and 8-port 10/100 BaseT Ethernet FPs 18

Text conventions 19

Procedure conventions 20

Operational mode 20

Provisioning mode 21

Activating configuration changes 22

How to get more help 22

Chapter 1
VPN configuration 23

Chapter 2
BGP/MPLS VPN configuration 25

Configuring the router and primary loopback 29

Configuring IP CPP on the RTR 31

Configuring EBGP and peers for address family ipv4Label 33

Changing an MD5 key on an authenticated BGP peer 36

Configuring BGP and peers for address family ipv4MplsVpn 38

Configuring BGP export policy 41

Configuring BGP import policy 43

Chapter 3
Media configuration 47

Configuring GigE media on the RFC2547 MPLS core 50

Configuring ATM media for single link congestion control 52

Configuring ATM media for multi link congestion control 54

Chapter 4
Routing protocol configuration 57

Configuring OSPF as an IGP 59

Configuring ISIS as an IGP 61

Chapter 5
LDP configuration 63

Configuring LDP 65

Configuring MD5 Authentication on an LDP neighbor 66

Changing an MD5 key on an authenticated LDP session 68

Chapter 6
VRF and access configuration 71

Configuring a VRF 74

Configuring ATM media and feature set 77

Configuring PPP media and feature set for VRF in RFC2547 mode 81

Configuring Ethernet media and feature set 83

Creating a VRRP virtual router 85

Configuring frame relay media and feature set 94
Configuring EBGP 98
Changing an MD5 key on an authenticated EBGP peer 101
Configuring BGP route flood protection 103
Configuring BGP peer route flood protection 105
Configuring OSPF 107
Configuring static routing 109
Configuring RIP 110
Configuring the VRF export policy 112
Configuring IP CPP on the VRF 114
Configuring VRF route flood protection 116

Chapter 7
BGP/MPLS VPN configuration example 119
Example procedure for a backbone configuration for PE node Multiservice Switch 1 (Multiservice Switch 7400 node) 121
Access Configuration for PE node Multiservice Switch 1 (Multiservice Switch 7400 node) 123
Example prcoedure for a backbone configuration for PE node Multiservice Switch 2 (Multiservice Switch 15000 node) 127
Access Configuration for PE node Multiservice Switch 2 (Multiservice Switch 15000 node) 128
Example procedure for configuring a P node on a Multiservice Switch 3 (Multiservice Switch 15000 node) 132
Example procedure for configuring a BGP MED policy 135
Example procedure for configuring a VRF policy 137
Example procedure for configuring a route target export policy 139

Chapter 8
VCG-based IP VPN configuration 141

Chapter 9
VCG-based IP VPN core configuration 143
Configuring core media 146
Configuring VCG 148
Configuring a core protocol port 149
Configuring virtual media for tunnel endpoints 151

Configuring round-trip delay measurements 153
Example of configuring BGP on a VCG with dynamic tunnels 155
Procedures steps 155

Chapter 10 VCG-based IP VPN customer access configuration 159

Configuring a customer virtual router 162
Configuring a dynamic tunnel 164
Configuring a static tunnel 167
Example of configuring a static tunnel 170
Procedure steps 170
Configuring ARP entries 173

List of figures

Figure 1 VPN configuration tasks 24
Figure 2 BGP/MPLS VPN configuration procedures 27
Figure 3 Router and primary loopback component hierarchy 30
Figure 4 RTR IP CPP component hierarchy 32
Figure 5 EBGP and peers for address family ipv4Label component hierarchy 35
Figure 6 BGP-4 component hierarchy for changing an MD5 key on an authenticated BGP peer 37
Figure 7 BGP and peers component hierarchy 40
Figure 8 BGP export policy component hierarchy 42
Figure 9 BGP import policy component hierarchy 46
Figure 10 Media configuration procedures 48
Figure 11 GigE media component hierarchy 51
Figure 12 ATM media for single link congestion control component hierarchy 53
Figure 13 ATM media for multi link congestion control component hierarchy 56
Figure 14 Routing protocol configuration procedures 58
Figure 15 OSPF as an IGP component hierarchy 60
Figure 16 ISIS as an IGP component hierarchy 62
Figure 17 LDP configuration procedures 64
Figure 18 LDP component hierarchy 65
Figure 19 LDP component hierarchy for MD5 authentication on an LDP neighbor 67
Figure 20 LDP component hierarchy for changing an MD5 key on an authenticated LDP session 69
Figure 21 VRF and access configuration procedures 72
Figure 22 VRF and route distinguisher component hierarchy 76
Figure 23 ATM media and feature set component hierarchy 80
Figure 24 Ethernet media and feature set component hierarchy 84
Figure 25 VRRP virtual router component hierarchy 87
Figure 26 Example figure for creating a VRRP virtual router associated with an Ethernet interface in port mode 90
Figure 27 Example figure for creating a VRRP virtual router associated with a VLAN on an Ethernet interface 93
Figure 28 Frame relay and feature set component hierarchy 97
Figure 29 EBGP component hierarchy 100

Figure 30 Component hierarchy for changing an MD5 key on an authenticated BGP peer 102
Figure 31 BGP route flood protection component hierarchy 104
Figure 32 BGP peer route flood protection component hierarchy 106
Figure 33 OSPF component hierarchy 108
Figure 34 Static routing component hierarchy 109
Figure 35 RIP component hierarchy 111
Figure 36 VRF export policy component hierarchy 113
Figure 37 VRF IP CPP component hierarchy 115
Figure 38 VRF route flood protection component hierarchy 117
Figure 39 Example figure for a BGP/MPLS VPN network configuration 120
Figure 40 Example figure for an export policy network configuration 138
Figure 41 VCG-based IP VPN configuration tasks 142
Figure 42 VCG-based IP VPN core configuration procedures 144
Figure 43 Core media component hierarchy 147
Figure 44 VCG component hierarchy 148
Figure 45 Core protocol port component hierarchy 150
Figure 46 Virtual media for tunnel endpoints component hierarchy 152
Figure 47 Round-trip delay measurements component hierarchy 154
Figure 48 Example of configuring BGP for a VCG for dynamic tunnels 157
Figure 49 VCG-based IP VPN customer access configuration procedures 160
Figure 50 Customer virtual router component hierarchy 163
Figure 51 Dynamic tunnel component hierarchy 166
Figure 52 Static tunnel component hierarchy 169
Figure 53 Example of configuring a static tunnel 172
Figure 54 ARP entries component hierarchy 174

List of tables

Table 1 LDP Neighbor operational attributes 70

About this document

This document contains procedural information required to configure virtual private networks (VPNs).

The information in this document is for persons who perform planning, engineering, installing, configuring, operating, maintaining, and troubleshooting tasks for VPNs.

The following topics are discussed in this section:

Who should read this document and why

This guide is for anyone who performs the following VPN tasks in Nortel Networks Multiservice Switch networks:

- planning
- installing and provisioning
- operating and maintaining

What you need to know

This guide assumes that you are familiar with the concepts of VPN services in Nortel Networks Multiservice Switch networks.

See NN10600-581 *Nortel Networks Multiservice Switch 7400/15000/20000 VPN Technology Fundamentals* for supporting information.

What's new in this document

The following features were added to this document:

- "Control Plane Protection (CPP)" (page 17)

- “Carrier’s Carrier” (page 17)
- “CP support and CP/VPNXc warm standby equipment protection” (page 17)
- “Ethernet access media” (page 17)
- “MD5 Authentication” (page 17)
- “Multi-hop EBGP” (page 18)
- “Route target export policy” (page 18)
- “Router model enhancements” (page 18)
- “Virtual router redundancy protocol (VRRP) on 4-port Gigabit Ethernet, 4-port 10/100 BaseT Ethernet, and 8-port 10/100 BaseT Ethernet FPs” (page 18)

Other changes made to this document include the following:

- The terms Passport and PVG have been rebranded in conjunction with the new Nortel Networks’ brand simplified naming format. Passport is now referred to as the Nortel Networks Multiservice Switch, and PVG is now Media Gateway 7480/15000. For more information on the product rebranding, refer to NN10600-000 *Nortel Networks Multiservice Switch 7400/15000/20000 What’s New in PCR6.1*.
- The Differentiated Services chapter has been moved to NN10600-591 *Nortel Networks Multiservice Switch 7400/15000/20000 Layer 3 Traffic Management Configuration*.
- The procedure “Configuring BGP peer route flood protection” (page 105) has been added to the VRF and access configuration chapter.
- CR Q00961343 - The procedures, “Configuring a VRF” (page 74) and “Configuring ATM media and feature set” (page 77) were updated to include information about the VRF loopback interface and address.
- Updated section “Configuring a static tunnel” (page 167) and added “Example of configuring a static tunnel” (page 170) to provide more practical information about configuring static tunnels.
- Added section “Example of configuring BGP on a VCG with dynamic tunnels” (page 155) in order to provide an example of this configuration.

- Updated Figure “VCG-based IP VPN core configuration procedures” (page 144) with the removal of the Virtual media configuration (always-up interface) procedure and added a procedure for Configuring virtual media for loopback address (always-up interface).

Control Plane Protection (CPP)

The following sections were updated for this feature:

- “Configuring IP CPP on the RTR” (page 31)
- “Configuring IP CPP on the VRF” (page 114)

Carrier’s Carrier

The following sections were updated for this feature:

- “Configuring EBGP and peers for address family ipv4Label” (page 33)
- “Configuring BGP and peers for address family ipv4MplsVpn” (page 38)
- “Configuring BGP route flood protection” (page 103)
- “Configuring VRF route flood protection” (page 116)

CP support and CP/VPNXc warm standby equipment protection

The following sections were updated for this feature:

- “Prerequisites to BGP/MPLS VPN configuration” (page 25)
- “Configuring the router and primary loopback” (page 29)

Ethernet access media

The following section was added for this feature:

- “Configuring Ethernet media and feature set” (page 83)

MD5 Authentication

The following sections were added for this feature:

- “Changing an MD5 key on an authenticated BGP peer” (page 36)
- “Changing an MD5 key on an authenticated LDP session” (page 68)
- “Changing an MD5 key on an authenticated EBGP peer” (page 101)

The following sections were updated for this feature:

- “Configuring EBGP and peers for address family ipv4Label” (page 33)
- “Configuring BGP and peers for address family ipv4MplsVpn” (page 38)
- “Configuring EBGP” (page 98)

Multi-hop EBGP

The following section was updated for this feature:

- “Configuring EBGP” (page 98)

Route target export policy

The following section was updated for this feature:

- “Configuring BGP export policy” (page 41)

Router model enhancements

The following sections were updated for this feature:

- “VRF and access configuration” (page 71)

Virtual router redundancy protocol (VRRP) on 4-port Gigabit Ethernet, 4-port 10/100 BaseT Ethernet, and 8-port 10/100 BaseT Ethernet FPs

The following sections were added for this feature:

- “Creating a VRRP virtual router” (page 85)
- “Example procedure for creating a VRRP virtual router associated with an Ethernet interface in port mode” (page 88)
- “Example procedure for creating a VRRP virtual router associated with a VLAN on an Ethernet interface” (page 91)

Text conventions

This document uses the following text conventions:

- `nonproportional spaced plain type`

  Nonproportional spaced plain type represents system generated text or text that appears on your screen.

- **`nonproportional spaced bold type`**

  Nonproportional spaced bold type represents words that you should type or that you should select on the screen.

- *italics*

  Statements that appear in italics in a procedure explain the results of a particular step and appear immediately following the step.

  Words that appear in italics indicate a component or attribute name.

- `[optional_parameter]`

  Words in square brackets represent optional parameters. The command can be entered with or without the words in the square brackets.

- `<general_term>`

  Words in angle brackets represent variables which are to be replaced with specific values.

- UPPERCASE, lowercase

  Nortel Networks Multiservice Switch system commands are not case-sensitive and do not have to match commands and parameters exactly as shown in this document, with the exception of string options values (for example, file and directory names) and string attribute values.

- |

  This symbol separates items from which you may select one; for example, ON|OFF indicates that you may specify ON or OFF. If you do not make a choice, a default ON is assumed.

- ...

  Three dots in a command indicate that the parameter may be repeated more than once in succession.

The term absolute pathname refers to the full specification of a path starting from the root directory. Absolute pathnames always begin with the slash (/) symbol. A relative pathname takes the current directory as its starting point, and starts with any alphanumeric character (other than /).

Procedure conventions

This document uses the following procedure conventions:

- You can enter commands using full component and attribute names, or you can abbreviate them. The commands used in the procedures contain the full component and attribute names in the first instance. In the second instance, the component and attribute names are abbreviated. For more information on abbreviating component and attribute names, see NN10600-060 *Nortel Networks Multiservice Switch 7400/15000/20000 Component Reference*. All component and attribute names are formatted in italics.
- The introduction of every procedure states whether you must perform the procedure in operational mode or provisioning mode. For more information on these modes, see “Operational mode” (page 20) or “Provisioning mode” (page 21).
- When you complete a procedure, you can verify your changes and then activate them as the new node configuration. For more information on completing configuration changes and exiting provisioning mode, see “Activating configuration changes” (page 22).

Operational mode

Procedures contained within this document can either be performed in operational mode or provisioning mode. When you initially log into a Nortel Networks Multiservice Switch node, you are in operational mode. The system uses the following command prompt when you are in operational mode:

```
#>
```

where:
# is the current command number

In operational mode, you work with operational components and attributes. In operational mode, you can

- list operational components and display operational attributes to determine the current operating parameters for the node
- control the state of parts of the node by locking and unlocking components
- set certain operational attributes and enter commands to perform diagnostic tests

Provisioning mode

To change from operational mode to provisioning mode, type the following command at the operator prompt:

```
start Prov
```

Only one user can be in provisioning mode at a time. The system uses the following command prompt whenever you are in provisioning mode:

```
PROV #>
```

where:

# is the current command number

In provisioning mode, you work with the provisionable components and attributes that contain the current and future configurations of the node. You can add and delete components, and display and set provisionable attributes. For information on completing the configuration changes, exiting provisioning mode, and returning to operational mode see “Activating configuration changes” (page 22).

For information on operational and provisionable attributes, see NN10600-060 *Nortel Networks Multiservice Switch 7400/15000/20000 Component Reference*.

Activating configuration changes

Several procedures in this document ask that you complete the configuration changes. When you complete the configuration changes, you are activating the configuration changes, confirming that you want to activate them, and saving the changes. You are instructed to complete the configuration changes only at the end of procedures that you perform in provisioning mode.

> **CAUTION**
> **Activating a provisioning view can affect service**
> Activating a provisioning view can result in a CP reload or restart, causing all services on the node to fail. See NN10600-050 *Nortel Networks Multiservice Switch 7400/15000/20000 Command Reference*, for more information.

1 Verify that the provisioning changes you have made are acceptable.

```
check Prov
```

Correct any errors and then verify the provisioning changes again.

2 If you want to store the provisioning changes in a file, save the provisioning view.

```
save Prov
```

3 If you want these changes as well as other changes made in the edit view to take effect immediately, activate, confirm, and commit the provisioning changes.

```
activate Prov
```

```
confirm Prov
```

```
commit Prov
```

4 End the provisioning session.

```
end Prov
```

How to get more help

For information on training, problem reporting, and technical support, see the “Nortel Networks works support services” section in NN10600-030 *Nortel Networks Multiservice Switch 7400/15000/20000 Overview*.

Chapter 1
VPN configuration

Virtual private network (VPN) configuration allows for multiple customer virtual routers (VRs) in a single virtual network.

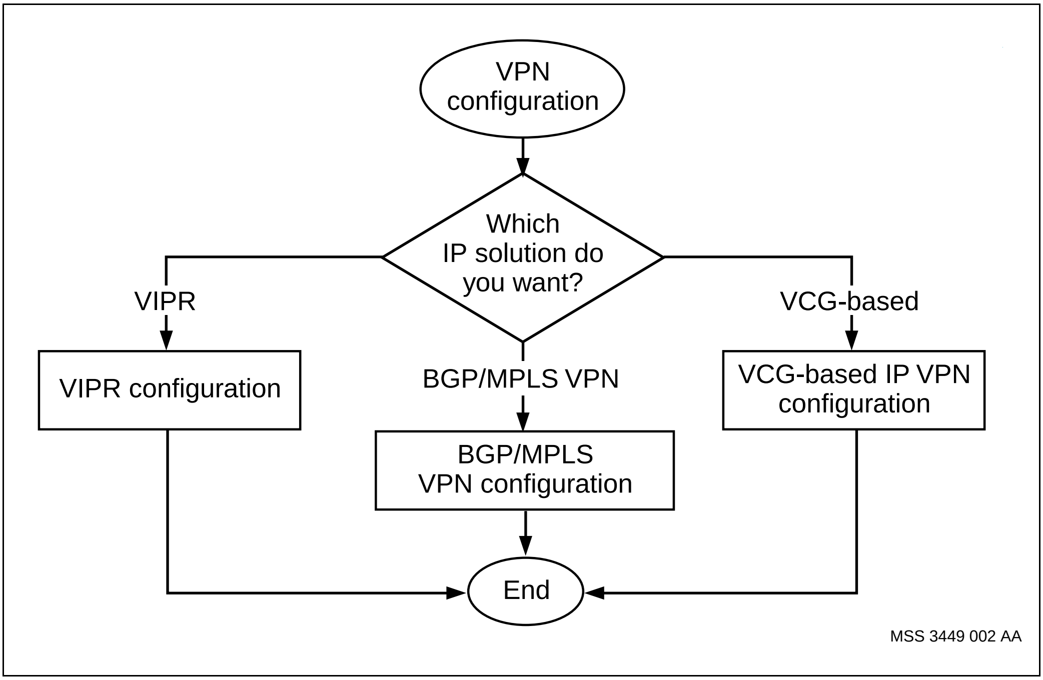
Prerequisites to VPN configuration

- Configure a management VR. For information on configuring a management VR, see NN10600-271 *Nortel Networks Multiservice Switch 7400/15000/20000 Network Management Connectivity*.

VPN configuration tasks

The following work flow shows you the sequence of tasks you perform to configure a VPN. To link to any task, go to “VPN configuration task navigation” (page 24).

Figure 1
VPN configuration tasks


VPN configuration
Which IP solution do you want?
VIPR
VCG-based
VIPR configuration
BGP/MPLS VPN
VCG-based IP VPN configuration
BGP/MPLS VPN configuration
End
MSS 3449 002 AA

VPN configuration task navigation

- VIPR configuration. For information, see NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.
- “BGP/MPLS VPN configuration” (page 25)
- “VCG-based IP VPN configuration” (page 141)

Chapter 2
BGP/MPLS VPN configuration

The border gateway protocol/multiprotocol label switching (BGP/MPLS) virtual private network (VPN) configuration allows you to configure the Provider Edge (PE), Provider (P), and Carrier's Carrier Customer Edge (CE') nodes in Nortel Networks Multiservice Switch networks. For Carrier's Carrier, the CE' node also functions as a PE node. For information about these terms, and for conceptual and reference information about the BGP/MPLS VPN solution, see NN10600-581 *Nortel Networks Multiservice Switch 7400/ 15000/20000 VPN Technology Fundamentals*.

Prerequisites to BGP/MPLS VPN configuration

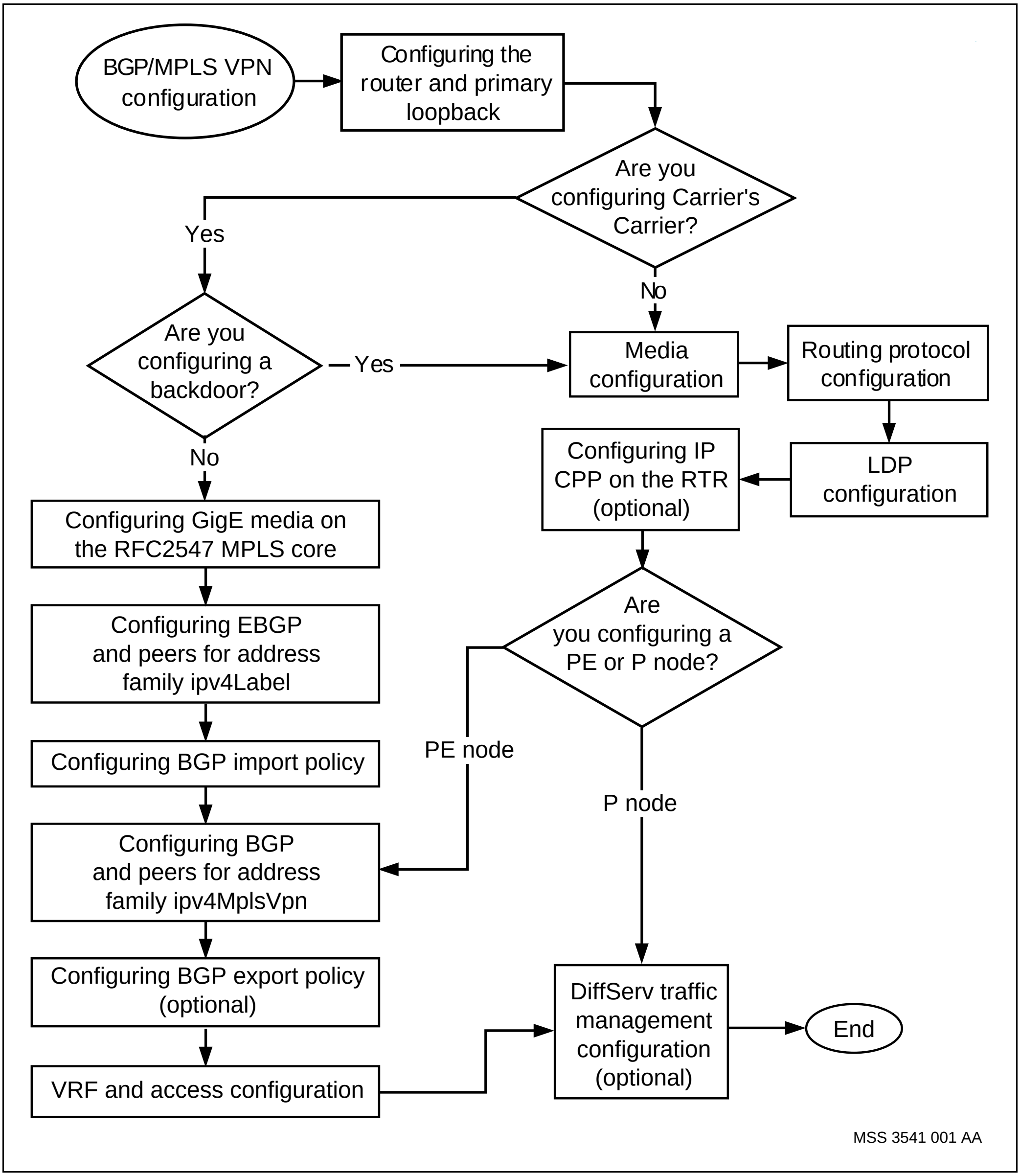
- Determine the media type prior to configuring software features. The decision you make to use ATM, or GigE media, affects the steps you take to configure the BGP/MPLS VPN.
- The task flow and procedures in this section describe configuring BGP/ MPLS services only. Basic configuration at the P node, PE node or CE' node level (in this case, creating an instance of a logical processor type (LPT), and adding the services to the *featureList* component) must be performed first. Use the tasks and procedures in NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures* if you require supporting information or need to provision or reconfigure any node or nodal elements to support BGP/ MPLS features.
    - For the PE node with a configuration that includes the VPN extender card and the CP card: add the *rfc2547bis*, *bgp*, and *ip* features to the VPN extender card feature list and add the *OamEnet, bgp,* and *ip* features to the CP card feature list.

- For the PE node with a configuration that includes the CP card but no VPN extender card: add the *OamEnet*, *rfc2547bis*, *bgp*, and *ip* features to the CP card feature list.
- For the PE node, include the *rfc2547bis* feature to the VRF access cards feature list.
- For the P node, include the *OamEnet* and *ip* features to the CP card feature list.
- For the P node, include the *mplsldp*, *atmmpe*, and *ip* features to the ATM card feature list.
- For the CE’ node, include the *rfc2547bis* feature to the VRF access cards feature list.
- For the CE’ node, include the *rfc2547bis* and *rfc2547bisCscCe* features to the VPN extender card feature list.
- For the CE’ node, include the *rfc2547bisCscCe* feature to the GigE FP card feature list.

BGP/MPLS VPN configuration procedures

This task flow shows you the sequence of procedures you perform to configure a BGP/MPLS VPN. To link to any procedure, go to “BGP/MPLS VPN configuration procedure navigation” (page 28).

Figure 2
BGP/MPLS VPN configuration procedures


BGP/MPLS VPN configuration
Configuring the router and primary loopback
Are you configuring Carrier's Carrier?
Yes
No
Are you configuring a backdoor?
Yes
No
Media configuration
Routing protocol configuration
LDP configuration
Configuring IP CPP on the RTR (optional)
Configuring GigE media on the RFC2547 MPLS core
Configuring EBGP and peers for address family ipv4Label
Configuring BGP import policy
Are you configuring a PE or P node?
PE node
P node
Configuring BGP and peers for address family ipv4MplsVpn
Configuring BGP export policy (optional)
VRF and access configuration
DiffServ traffic management configuration (optional)
End
MSS 3541 001 AA

BGP/MPLS VPN configuration procedure navigation

- “Configuring the router and primary loopback” (page 29)
- “Media configuration” (page 47)
- “Routing protocol configuration” (page 57)
- “LDP configuration” (page 63)
- “Configuring IP CPP on the RTR” (page 31)
- “Configuring GigE media on the RFC2547 MPLS core” (page 50)
- “Configuring EBGP and peers for address family ipv4Label” (page 33)
- “Configuring BGP and peers for address family ipv4MplsVpn” (page 38)
- “Configuring BGP export policy” (page 41)
- “Configuring BGP import policy” (page 43)
- “VRF and access configuration” (page 71)
- DiffServ traffic management configuration. See to NN10600-591 *Nortel Networks Multiservice Switch 7400/15000/20000 Layer 3 Traffic Management Configuration*.
- For examples of BGP/MPLS VPN configuration, refer to “BGP/MPLS VPN configuration example” (page 119).

Configuring the router and primary loopback

Configure a router to operate as a P node, PE node or CE’ node in order to provide IGP connectivity between them. The P node router is a backbone router which is not connected to any CE devices. The PE is connected to one or more CE devices and handles all VPN routing information. Configure a mandatory instance of Loopback interface with mode primaryLoopback. The IP address of this component is called the primaryLoopback address. The primaryLoopback address is used in the RFC 2547 solution by MPLS and BGP control plane as follows:

- LDP-DU as the transport layer next hop when establishing LSP tunnels between PE nodes used to transport VPN traffic
- BGP next hop when distributing VPN routing information between PE nodes

Procedure steps

1 Add the *Router* component.

```
add Rtr/<rtr_name>
```

2 Link the router to the VPN extender card or the CP card.

```
set Rtr/<rtr_name> Vrp lp/<n>
```

3 Add the loopback interface IP address.

```
add -s Rtr/<rtr_name> Interface/<ip_address> Loopback
```

4 Set the netmask. This must be 32 bits.

```
set Rtr/<rtr_name> Interface/<ip_address> netMask
255.255.255.255
```

5 Set the *mode* attribute to primaryLoopback.

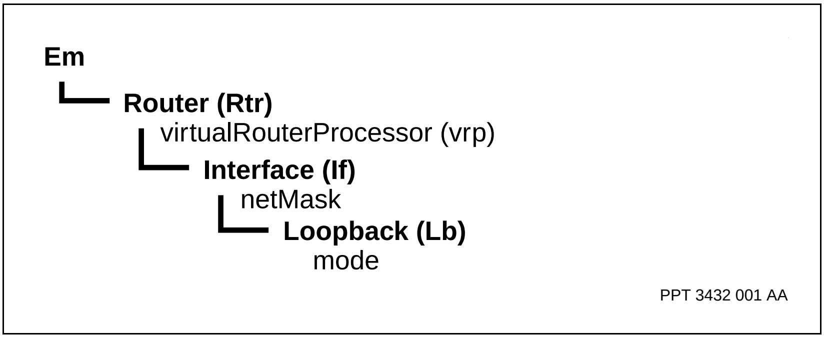
```
set Rtr/<rtr_name> Interface/<ip_address> Loopback
mode primaryLoopback
```

Variable definitions

Variable	Value
<ip_address>	is a 32-bit address assigned to the interface from which the routing information is derived.
<mask>	is the network mask to be used with the IP address. Must be 32 bits in the case of a loopback interface.
<n>	is the number of the LP. The router can be linked to the VpnXc card or CP card. CPs must be Lp/0.
<rtr_name>	is the name of the router.

Procedure job aid

Figure 3
Router and primary loopback component hierarchy



Configuring IP CPP on the RTR

Configure control plane protection (CPP) on the RTR to protect Nortel Networks Multiservice Switch nodes against certain denial of service (DoS) attacks on the control plane by monitoring the flow rate of IP packets destined for local IP destination addresses (DAs).

CPP is supported on PQC-based FPs only.

Prerequisites

- See the section on CPP in the NN10600-800 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Technology Fundamentals*.
- The procedure in this section describes configuring CPP software and services only. Basic configuration at the node level (in this case, creating an instance of a logical processor type (LPT), and adding the *ip* and *ipCpp* services to the *featureList* attribute) must be performed first. Use the tasks and procedures in NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures* if you require supporting information or need to provision or reconfigure any node or nodal elements to support the CPP feature.

Procedure steps

1 Add a *Cpp* component as a subcomponent of the *Rtr* component.

```
add Rtr/<rtr_name> cpp
```

2 Configure the number of packets per second before discard occurs.

```
set Rtr/<rtr_name> cpp packetsPerSecond
<packets_per_second>
```

3 Configure the isolation time period.

```
set Rtr/<rtr_name> cpp isolationTime <isolation_time>
```

4 Configure the grace period.

```
set Rtr/<rtr_name> cpp gracePeriod <grace_period>
```

5 Configure the CPP mode.

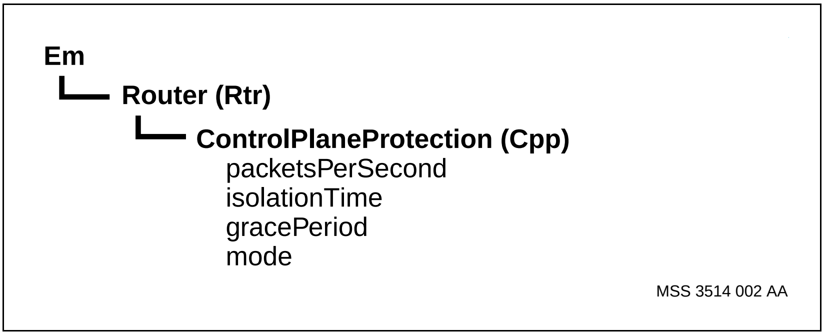
```
set Rtr/<rtr_name> cpp mode <cpp_mode>
```

Variable definitions

Variable	Value
<cpp_mode>	is the CPP operational mode: study, protect or disabled. Use study to determine an acceptable traffic rate. Once you learn the rate created and have the appropriate configuration, use protect. Use disabled to pre-configure the feature without enabling the monitoring process.
<grace_period>	is the period over which the average flow rate is measured to ensure that the exceeded traffic flow rate still exceeds the maximum allowed rate. A value of zero in protect mode means that isolation occurs immediately after an excessive flow is detected.
<isolation_time>	is the amount of time over which traffic will be discarded once isolation has begun. A value of zero indicates to permanently discard traffic until the card is cleared by an operator.
<packets_per_second>	is the flow rate, in packets per second, for the VR's DA, that must be exceeded on a single DA before discard processing occurs.
<rtr_name>	is the name of the router.

Procedure job aid

Figure 4
RTR IP CPP component hierarchy



Configuring EBGP and peers for address family ipv4Label

Configure the exterior border gateway protocol (EBGP) for the CE’ node and add one peer for each remote PE’ node that is directly connected.

EBGP with address family ipv4Label is replacing:

- the interior gateway protocol (IGP) for primary loopback address distribution
- the label distribution protocol (LDP) for transport label distribution

Optionally, configure MD5 authentication on a BGP peer connection to provide protection of BGP neighbor relationships. Change an MD5 key on an authenticated BGP peer using the procedure “Changing an MD5 key on an authenticated BGP peer” (page 36).

Prerequisites

- If configuring MD5 authentication, use different MD5 key values for each BGP peer. A longer MD5 key will be more difficult to break, however, the longer the key used, the greater the impact on performance. A key length of between 12 and 24 characters should be sufficiently secure without having too great an impact on performance.

Procedure steps

1 Add the *Bgp* component.

```
add Rtr/<rtr_name> Bgp
```

2 Set the *localAs* attribute.

```
set Rtr/<rtr_name> Bgp localAs <local_as_value>
```

3 Set the *bgpIdentifier* attribute.

```
set Rtr/<rtr_name> Bgp bgpIdentifier <ip_address>
```

4 Add the BGP peer.

```
add Rtr/<rtr_name> Bgp Peer/<peer_address>
```

The *Descriptor* subcomponent is automatically added.

5 Optionally, configure MD5 authentication on a BGP peer connection. Set the MD5 key value under the *Descriptor*.

```
set Rtr/<rtr_name> Bgp Peer/<ip_addr> Desc md5Key
<ASCII_string>
```

6 Set the *LocalAddressConfigured* (*lac*) attribute.

```
set Rtr/<rtr_name> Bgp Peer/<peer_address> Desc lac
<lac_ip_address>
```

7 Set the autonomous system that is changed for the BGP peer.

```
set Rtr/<rtr_name> Bgp Peer/<peer_address> Desc peerAs
<peer_as_value>
```

8 Set the address family for the node you are configuring.

```
set Rtr/<rtr_name> Bgp Peer/<peer_address> Desc
addressFamily ipv4Label
```

9 Disable AS_PATH loop detection.

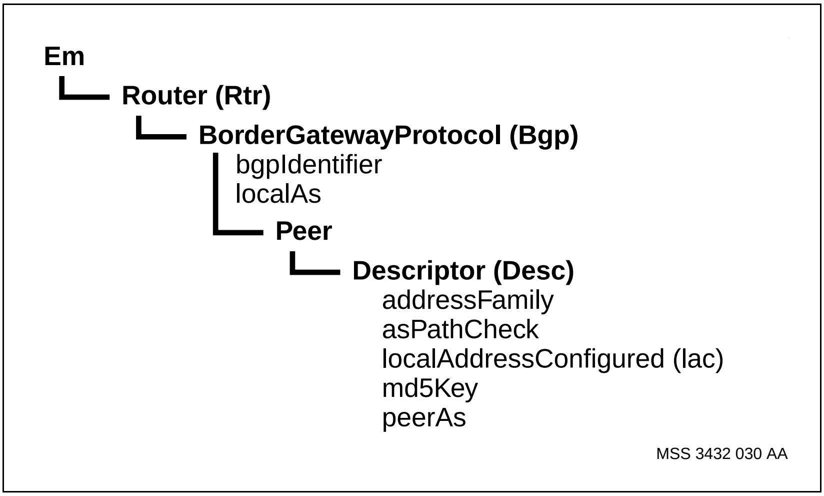
```
set Rtr/<rtr_name> Bgp Peer/<peer_address> Desc
asPathCheck disabled
```

Variable definitions

Variable	Value
<ASCII_string>	is the MD5 key value, 1-255 ASCII characters in length. Any printable ASCII character is allowed. If a space or other non-alphabetic or numeric character is used within the key, then the whole key must be included in quotes. For this reason a “ character is the only printable ascii character not allowed as part of a key. A null string indicates that MD5 authentication is not to be used on the connection to this peer.
<ip_address>	is the IP address that matches the primary loopback address.
<lac_ip_address>	is the IP address that matches the CE’ interface that is directly connected to the VRF PE’ interface.
<local_as_value>	is the value of the autonomous system for the CE’ node.
<peer_address>	is the IP address of the VRF PE’ interface that is directly connected to the CE’ node.
<peer_as_value>	is the value of the autonomous system for the PE’ node.
<rtr_name>	is the name of the router.

Procedure job aid

Figure 5
EBGP and peers for address family ipv4Label component hierarchy



Changing an MD5 key on an authenticated BGP peer

Transition MD5 keys without having to terminate the BGP neighbor session for which the key is being changed, if you have optionally configured MD5 authentication on a BGP peer.

When changing keys on a BGP session between a Multiservice Switch node and a non-Multiservice Switch, change the key on the Multiservice Switch node to minimize the outage time.

A longer MD5 key will be more difficult to break, however, the longer the key used, the greater the impact on performance. A key length of between 12 and 24 characters should be sufficiently secure without having too great an impact on performance.

It is recommended that different MD5 key values be used for each BGP-4 peer.

Procedure steps

1 Set the *keyTransitionDelay* attribute on the local and remote router, for all BGP peers.

```
set Rtr/<rtr_name> Bgp keyTransitionDelay <n>
```

2 Alternatively, set the *keyTransitionDelay* attribute on the local and remote router, for a specific peer's *Descriptor*.

```
set Rtr/<rtr_name> Bgp Peer/<ip_addr> Desc
keyTransitionDelay <n>
```

3 Set the new MD5 key value on the local and remote router.

```
set Rtr/<rtr_name> Bgp Peer/<ip_addr> Desc md5Key
<ASCII_string>
```

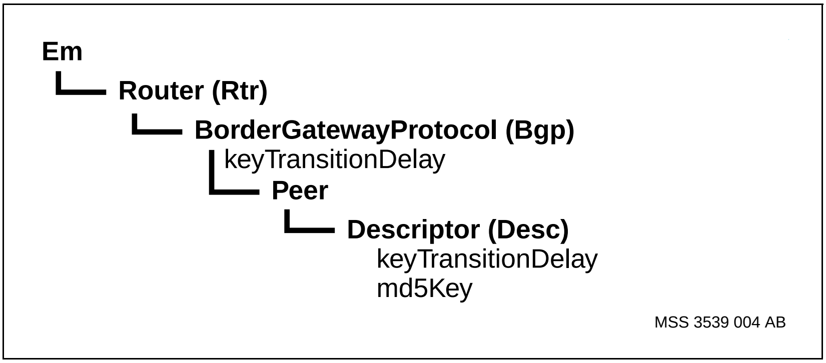
Variable definitions

Variable	Value
<ASCII_string>	is the MD5 key value, 1-255 ASCII characters in length.
<ip_addr>	is a 32-bit address assigned to the interface from which the routing information is derived.
(Sheet 1 of 2)	

Variable	Value
<n>	is the amount of time the operator has to provision both the local and remote ends with the new key, without losing service. for the *Bgp* component, the value is 1 to 20160 (minutes), default is 10 minutes. for the *Descriptor* component, the value is “sameAsBgp, 1 to 20160 (minutes)”, default is “sameAsBgp”.
<rtr_name>	is the name of the router.
(Sheet 2 of 2)	

Procedure job aid

Figure 6
BGP-4 component hierarchy for changing an MD5 key on an authenticated BGP peer



Configuring BGP and peers for address family ipv4MplsVpn

Configure the interior border gateway protocol (IBGP) or the exterior border gateway protocol (EBGP) for address family ipv4MplsVpn.

Configure the IBGP for the PE or CE’ node and add one peer for each remote PE or CE’ node respectively in the network. For inter-autonomous systems (inter-ASs), configure the EBGP peers for the PE node and add one peer for each remote PE node in the network. Using the BGP protocol, VPN IPv4 routes are distributed between the BGP peers.

Optionally, configure MD5 authentication on a BGP peer connection to provide protection of BGP neighbor relationships. You can change an MD5 key on an authenticated BGP peer using the procedure “Changing an MD5 key on an authenticated BGP peer” (page 36).

Prerequisites

- If configuring MD5 authentication, use different MD5 key values for each BGP peer. A longer MD5 key will be more difficult to break, however, the longer the key used, the greater the impact on performance. A key length of between 12 and 24 characters should be sufficiently secure without having too great an impact on performance.

  Try to use a key length of between 12 and 24 characters, which should be sufficiently secure without having too great an impact on performance. A longer MD5 key will be more difficult to break, however, the longer the key used, the greater the impact on performance.

Procedure steps

1 Add the *Bgp* component.

```
add Rtr/<rtr_name> Bgp
```

2 Set the *localAs* attribute.

```
set Rtr/<rtr_name> Bgp localAs <local_as_value>
```

3 Set the *bgpIdentifier* attribute.

```
set Rtr/<rtr_name> Bgp bgpIdentifier <ip_address>
```

4 Add the BGP peer.

```
add Rtr/<rtr_name> Bgp Peer/<peer_address>
```

The *Descriptor* subcomponent is automatically added.

5 Optionally, configure MD5 authentication on a BGP peer connection. Set the MD5 key value under the *Descriptor*.

```
set Rtr/<rtr_name> Bgp Peer/<ip_addr> Desc md5Key
<ASCII_string>
```

6 Set the *LocalAddressConfigured* (*lac*) attribute.

```
set Rtr/<rtr_name> Bgp Peer/<peer_address> Desc lac
<lac_ip_address>
```

7 Set the autonomous system identifier for the BGP peer.

```
set Rtr/<rtr_name> Bgp Peer/<peer_address> Desc peerAs
<peer_as_value>
```

8 Set the address family for the node you are configuring.

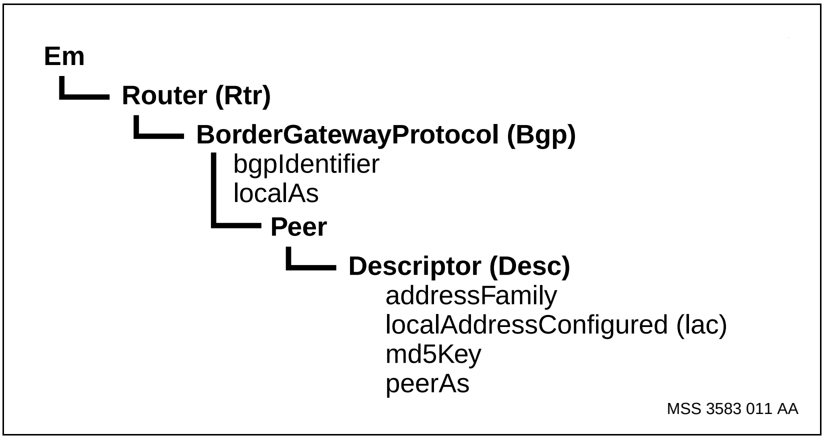
```
set Rtr/<rtr_name> Bgp Peer/<peer_address> Desc
addressFamily ipv4MplsVpn
```

Variable definitions

Variable	Value
<ASCII_string>	is the MD5 key value, 1-255 ASCII characters in length.
<ip_address>	is the IP address that matches the local primary loopback address.
<lac_ip_address>	is the IP address that matches the local primary loopback address.
<local_as_value>	is the value of the autonomous system for the PE or CE' node.
<peer_address>	is the primary loopback address of the peer of the remote PE or CE' node.
<peer_as_value>	is the value of the autonomous system for the remote PE or CE' node. Note that for IBGP, the localAS value and the peerAS value must be the same.
<rtr_name>	is the name of the router.

Procedure job aid

Figure 7
BGP and peers component hierarchy



Configuring BGP export policy

Optionally, you can configure a border gateway protocol (BGP) export policy on the PE node to either block or send certain routes to its peers. You can decide to filter VPN IPv4 routes based on the values of the network address, the remote peer IP address, and the route target.

Procedure steps

1 By default, BGP advertises all routes in the *MplsVpnRibEntry* component to its peers. Optionally, you can configure an export policy on the router to either block or send certain routes to its peers. To start configuring the export policy, add the *ExportPolicy* component.

```
add -s Rtr/<rtr_name> Bgp Export/<export>
```

2 Set the *RouteTarget* attribute to filter routes based on route targets.

```
set Rtr/<rtr_name> Bgp Export/<export> Rt
<route_target>
```

To apply a BGP export policy to a set of IP addresses associated with a VRF, you must set the *RouteTarget* attribute to include all the route target values that have the *mode* attribute set to export for that VRF.

3 Set the *addressFamily* attribute to apply this export policy to VPN IPV4 routes.

```
set Rtr/<rtr_name> Bgp Export/<export> af ipv4MplsVpn
```

4 Add the *Network* component and set the *prefix* and *length* attributes to filter routes based on specific network addresses.

```
add Rtr/<rtr_name> Bgp Export/<export> Net/<network>
prefix <prefix>, length <length>
```

5 Set the *adverstiseStatus* attribute to specify the action to take on a matched policy.

```
set Rtr/<rtr_name> Bgp Export/<export> advertise
<advertise>
```

6 Set the *peerIpAddress* attribute to specify the peer that this policy applies to.

```
set Rtr/<rtr_name> Bgp Export/<export> peerIpAddress
<peer_ip_address>
```

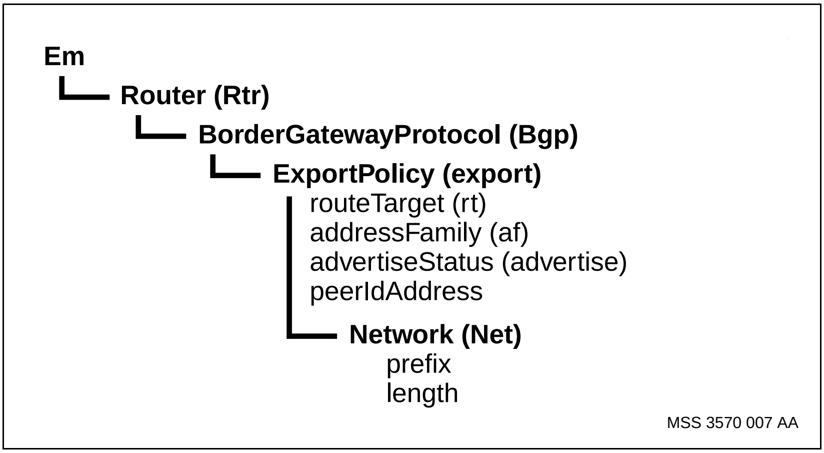
7 Repeat this procedure for all the PE nodes and Route Reflectors with which this PE node will peer.

Variable definitions

Variable	Value
<advertise>	is the advertise status for exporting routes. The status is block or send. The default is send.
<export>	is an instance that describes a BGP export policy.
<length>	is specifying the length of the network *prefix* attribute.
<network>	is an instance that describes a network address.
<peer_ip_address>	is specifying the IP address of the peer that the routes are advertised to.
<prefix>	is specifying the network prefix. The length of the network prefix is determined by the *length* attribute under the same *Network* component.
<route_target>	is specifying a list of routes for which the export policy applies to.
<rtr_name>	is the name of the router.

Procedure job aid

Figure 8
BGP export policy component hierarchy



Configuring BGP import policy

Configure the border gateway protocol (BGP) import policy to decide if certain labeled routes learned from the EBGP peer are to be ignored or used on the CE' node. If no import policy is provisioned, BGP does not use any routes received from its EBGP peer. If multiple import policies are provisioned, the one with the most specific match applies.

Procedure steps

1 Add BGP import policy.

```
add Rtr/<rtr_name> Bgp Import/<import>
```

2 Optionally, set the *addressFamily* (*af*) attribute to match the labeled EBGP addresses only. By default, the addressFamily is set to all.

```
set Rtr/<rtr_name> Bgp Import/<import> af ipv4Label
```

3 Set the action to take upon a policy match.

```
set Rtr/<rtr_name> Bgp Import/<import> usage
<usage_flag>
```

4 Add the network component and set the prefix and length attributes to filter routers based on specific network addresses.

```
add Rtr/<rtr_name> Bgp Import/<import> Net/<network>
prefix <prefix>, length <length>
```

5 Specify the AS number of the EBGP peer from which routes are learned.

```
set Rtr/<rtr_name> Bgp Import/<import> peerAs <asNo>
```

6 Specify the IP address of the EBGP peer from which routes are learned.

```
set Rtr/<rtr_name> Bgp Import/<import> peerIpAddress
<addr>
```

7 Specify the AS that originated the routes learned over the EBGP peer.

```
set Rtr/<rtr_name> Bgp Import/<import> originAs
<or_asNo>
```

8 Specify a regular expression that identifies AS paths from which EBGP accepts route updates if you do not want to use the default value.

```
set Rtr/<rtr_name> Bgp Import/<import> asExpr
<path_expr>
```

9 Specify a regular expression that identifies community paths from which EBGP peer accepts route updates if you do not want to use a default value.

```
set Rtr/<rtr_name> Bgp Import/<import> comExpr
<com_expr>
```

10 If you have configured an AS path and community path expression for the import policy, specify a preference for the policy.

```
set Rtr/<rtr_name> Bgp Import/<import> exprPref <pref>
```

When the expression attributes of two import policies match the same AS or community, EBGP uses the preference metric to select a preferred policy. A higher value indicates a higher preference.

11 Specify the protocol that originated the routes learned over the EBGP peer.

```
set Rtr/<rtr_name> Bgp Import/<import> originProtocol
<protocol>
```

12 If required, override the route preference by changing attribute *ebgpRtePref* for BGP external routes.

```
set Rtr/<rtr_name> Bgp Import/<import>
ebgpRtePreference <ebgp_override>
```

13 Specify a preference for routes that match the import policy:

```
set Rtr/<rtr_name> Bgp Import/<import> locPrf
<loc_pref>
```

If you do not set this value, EBGP applies the local preference configured under the EBGP instance in the *defaultLocalPreference* attribute to routes that meet the import policy criteria.

14 Specify the community number that EBGP inserts in the community path attribute for routes that match the criteria of this import policy.

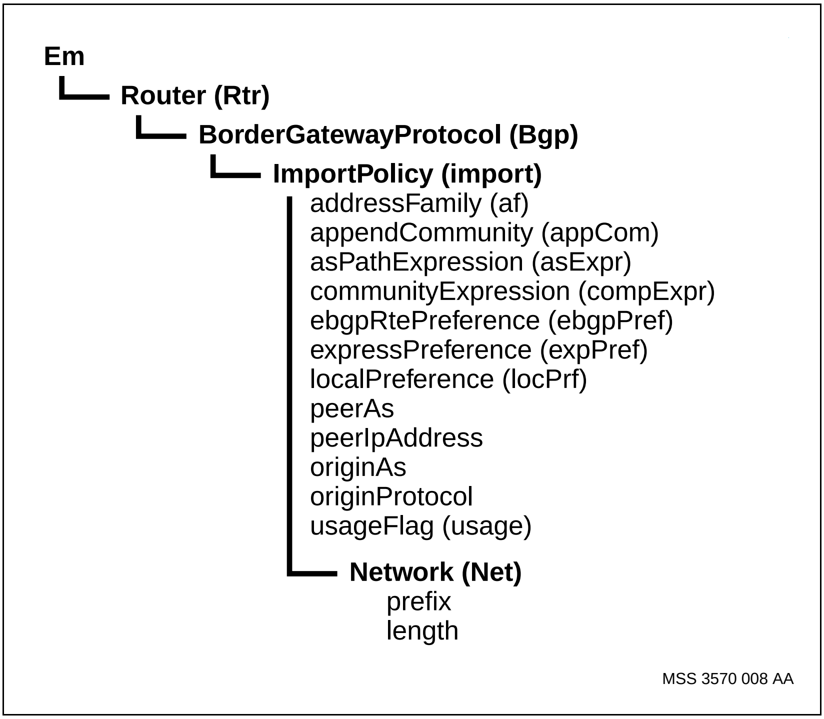
```
set Rtr/<rtr_name> Bgp Import/<import> appCom <com_no>
```

Variable definitions

Variable	Value
<addr>	is the IP address of the EBGP peer from which the routes are learned. If you set this value to 0.0.0.0, the policy matches any IP address.
<asNo>	specifies the autonomous system number to which the EBGP peer belongs. If you set this value to 0, the policy matches any AS number.
<com_expr>	is a regular expression identifying community paths to match.
<com_no>	is the community number added to the community path attribute.
<ebgp_override>	is the override route preference. Attribute default is sameAsBgp, which means use the value of attribute *defaultEbgpRtePref* for the route preference. To prefer BGP external routes over OSPF internal routes, the recommended setting for *ebgpRtePreference* is 6.
<import>	is an instance that describes a BGP import policy.
<length>	is specifying the length of the network *prefix* attribute.
<loc_pref>	is the relative preference for routes that match the import policy's criteria
<network>	is an instance that describes a network address.
<or_asNo>	is the number of the AS that originated the learned routes. If you set this value to 0, the policy matches any AS number.
<path_expr>	a regular expression identifying AS paths to match.
<pref>	is the relative preference of a path-based policy.
<prefix>	is specifying the network prefix.
<protocol>	identifies the protocol that originated the routes.
<rtr_name>	is the name of the router.
<usage_flag>	is the action to take on importing routes. To accept labeled routes, set the value of the *usageFlag* attribute to use. To prevent the use of labeled routes, set the value of the *usageFlag* attribute to ignore. The default value is use.

Procedure job aid

Figure 9
BGP import policy component hierarchy



Chapter 3
Media configuration

The media configuration is used to encapsulate forwarded IP packets into the underlying layer 2 specific media interface (for example, an Ethernet frame).

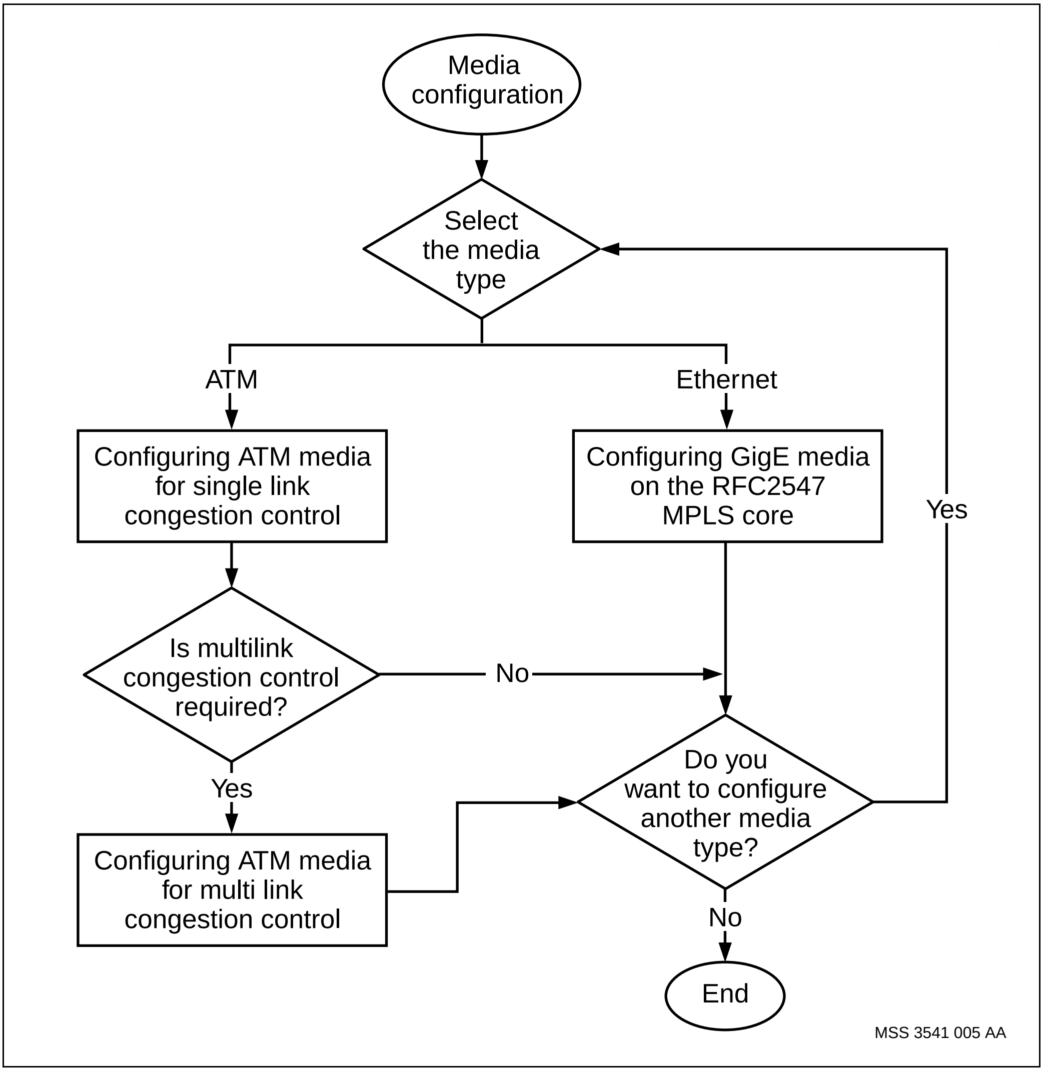
Prerequisites to media configuration

- Determine the media type prior to configuring software features. The decision you make to use ATM, or GigE media, affects the steps you take to configure the BGS/MPLS VPN.

Media configuration procedures

This task flow shows you the sequence of procedures you perform to configure media. To link to any procedure, go to “Media configuration procedure navigation” (page 48).

Figure 10
Media configuration procedures


Media configuration
Select the media type
ATM
Ethernet
Configuring ATM media for single link congestion control
Configuring GigE media on the RFC2547 MPLS core
Yes
Is multilink congestion control required?
No
Yes
Do you want to configure another media type?
Configuring ATM media for multi link congestion control
No
End
MSS 3541 005 AA

Media configuration procedure navigation

- “Configuring GigE media on the RFC2547 MPLS core” (page 50)
- “Configuring ATM media for single link congestion control” (page 52)

- “Configuring ATM media for multi link congestion control” (page 54)

Configuring GigE media on the RFC2547 MPLS core

Configure GigE media to define physical connectivity for BGP/MPLS VPN tunnels. MPLS LSPs will be signalled over these links using LDP-DU. Packets transmitted on the LSPs will be differentiated by multiple levels of drop precedence and scheduling class.

Procedure steps

1 Add the *EnetApplication* component.

```
add Rtr/<rtr_name> Interface/<ip_address> Enet
```

2 Add a Logical Processor.

```
add Lp/<lp_id>
```

3 Add an Ethernet port.

```
add Lp/<lp_id> Enet
```

4 Add the *LanApplication* component.

```
add La/<la_name>
```

The *Framer* subcomponent is automatically added.

5 Link the LanApplication to the GigE port.

```
set La/<la_name> Framer interfaceName Lp/<lp_id> Enet/
<enet_port>
```

6 Link to the LanApplication.

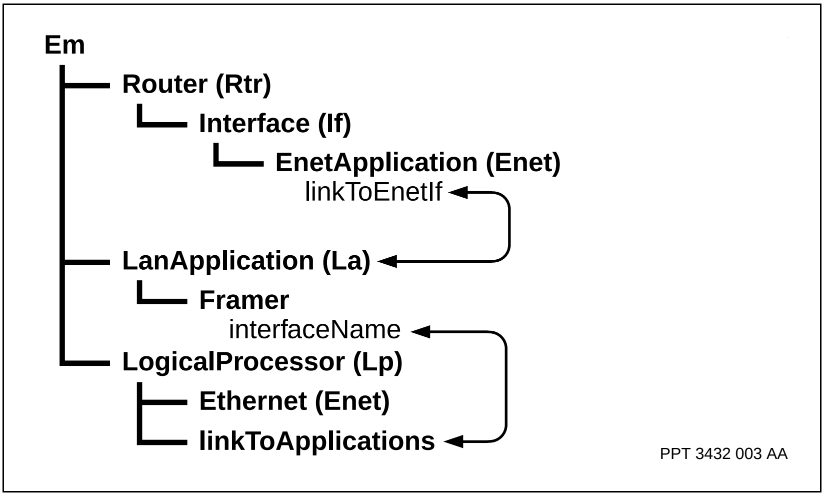
```
set Rtr/<rtr_name> Interface/<ip_address> Enet
linkToEnetIf La/<la_name>
```

Variable definitions

Variable	Value
<enet_port>	is the name assigned to the ethernet port.
<ip_address>	is a 32-bit address assigned to the interface from which the routing information is derived.
<la_name>	is the identifier assigned to the LanApplication.
<lp_id>	is the identifier assigned to the GigE LP.
<rtr_name>	is the name of the router.

Procedure job aid

Figure 11
GigE media component hierarchy



Configuring ATM media for single link congestion control

Configure ATM media for single link congestion control to define physical connectivity for BGP/MPLS VPN tunnels. MPLS LSPs will be signalled over these links using LDP-DU. Packets transmitted on the LSPs will be differentiated by multiple levels of drop precedence.

Prerequisites

- For more information on *AtmIf* component configuration, see NN10600-710 *Nortel Networks Multiservice Switch 7400/15000/20000 ATM Configuration Management*

Procedure steps

1 Set the LPT feature list to include the ATM card feature set.

```
set Sw Lpt/<lpt_name> featureList atmmpe
```

2 Activate changes. See “Activating configuration changes” (page 22).

3 Add *ATMMPE* media set to use LLC encapsulation. This is the default setting.

```
add -s Rtr/<rtr_name> Interface/<ip_address> AtmMpe
```

4 Set the *netmask* attribute.

```
set Rtr/<rtr_name> Interface/<ip_address> netmask
<ip_mask>
```

5 Add the Nailed Up Endpoint

```
add -s AtmIf/<atmif_id> Vcc/<vpi.vci> nep
```

6 Link the *AtmConnection* subcomponent, which is added automatically with DiffServ connection class 0 when the *atmMpe* component is created, to an ATM VCC that is configured with the ATM UBR service category.

```
set Rtr/<rtr_name> Interface/<ip_address> AtmMpe Ac/
<ac_id> AtmConnection AtmIf/<atmif_id> Vcc/<vpi.vci>
Nep
```

```
set AtmIf/<atmif_id> Vcc/<vpi.vci> Vcd Tm
atmServiceCategory ubr
```

7 Enable packet-wise discard.

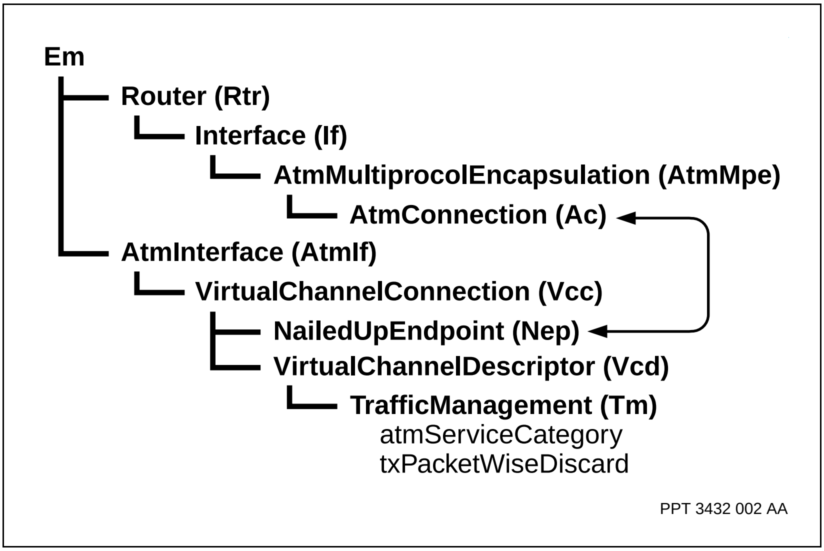
```
set AtmIf/<atmif_id> Vcc/<vpi.vci> Vcd Tm
txPacketWiseDiscard enabled
```

Variable definitions

Variable	Value
<ac_id>	is the instance value of the atm connections. Any mnemonic (for example, AC/1).
<atmif_id>	is the instance value of the *AtmIf* component.
<ip_address>	is a 32-bit address assigned to the interface from which the routing information is derived.
<ip_mask>	is the netmask of the IP address. This value cannot be 32 bits for an interface.
<rtr_name>	is the name of the router.
<vpi.vci>	is the identifier assigned to the VP and VC of the virtual channel.

Procedure job aid

Figure 12
ATM media for single link congestion control component hierarchy



Configuring ATM media for multi link congestion control

Configuring ATM media for multi link congestion control allows packets transmitted on the LSPs to be differentiated by multiple levels of drop precedence and by the multi link congestion control configuration of the ATM links.

At least one ATM connection must be created. Nortel Networks Multiservice Switch systems support up to four ATM connections between hops in IP and MPLS networks. This provides four unique emission priorities when transmitting from the Multiservice Switch to the next node, and allows for individual service differentiation on separate links in the ATM network. To enable this capability, up to three more ATM connections may be added, one for each of the four DiffServ connection class values. For more information on differentiated service, see NN10600-591 *Nortel Networks Multiservice Switch 7400/15000/20000 Layer 3 Traffic Management Configuration*.

Prerequisites

- For more information on *AtmIf* component configuration, see NN10600-710 *Nortel Networks Multiservice Switch 7400/15000/20000 ATM Configuration Management*

Procedure steps

1 Configure the COS value.

```
set Rtr/<rtr_name> Interface/<ip_address> AtmMpe Ac/
<ac_id> IpCos/<ipcos_value>
```

2 Add an ATM connection with DiffServ connection class 1 and link it to an ATM VCC that has an ATM nrtVBR service category.

```
set Rtr/<rtr_name> Interface/<ip_address> AtmMpe Ac/
<ac_id> AtmConnection AtmIf/<atmif_id> Vcc/<vpi.vci>
Nep
```

```
set AtmIf/<atmif_id> Vcc/<vpi.vci> Vcd tm
atmServiceCategory nrtvbr
```

3 Set the traffic management parameter of the ATM connection. This parameter is used to derive the OSPF metric value for this interface.

```
set AtmIf/<atmif_id> Vcc/<vpi.vci> Vcd Tm
txTrafficDescType <desc_type>
```

```
set AtmIf/<atmif_id> Vcc/<vpi.vci> Vcd Tm
txTrafficDescParm 1 <desc_parm>
```

4 Enable packet-wise discard.

```
set AtmIf/<atmif_id> Vcc/<vpi.vci> Vcd Tm
txPacketWiseDiscard enabled
```

5 Add an ATM connection with DiffServ connection class 2 and link it to an ATM VCC that has an ATM rtVBR service category.

```
set Rtr/<rtr_name> Interface/<ip_address> AtmMpe Ac/
<ac_id> AtmConnection AtmIf/<atmif_id> Vcc/<vpi.vci>
Nep
```

```
set AtmIf/<atmif_id> Vcc/<vpi.vci> Vcd Tm
atmServiceCategory rtvbr
```

6 Enable packet-wise discard.

```
set AtmIf/<atmif_id> Vcc/<vpi.vci> Vcd Tm
txPacketWiseDiscard enabled
```

7 Add an ATM connection with DiffServ connection class 3 and link it to an ATM VCC that has an ATM CBR service category.

```
set Rtr/<rtr_name> Interface/<ip_address> AtmMpe Ac/
<ac_id> AtmConnection AtmIf/<atmif_id> Vcc/<vpi.vci>
Nep
```

```
set AtmIf/<atmif_id> Vcc/<vpi.vci> Vcd Tm
atmServiceCategory cbr
```

8 Enable packet-wise discard.

```
set AtmIf/<atmif_id> Vcc/<vpi.vci> Vcd Tm
txPacketWiseDiscard enabled
```

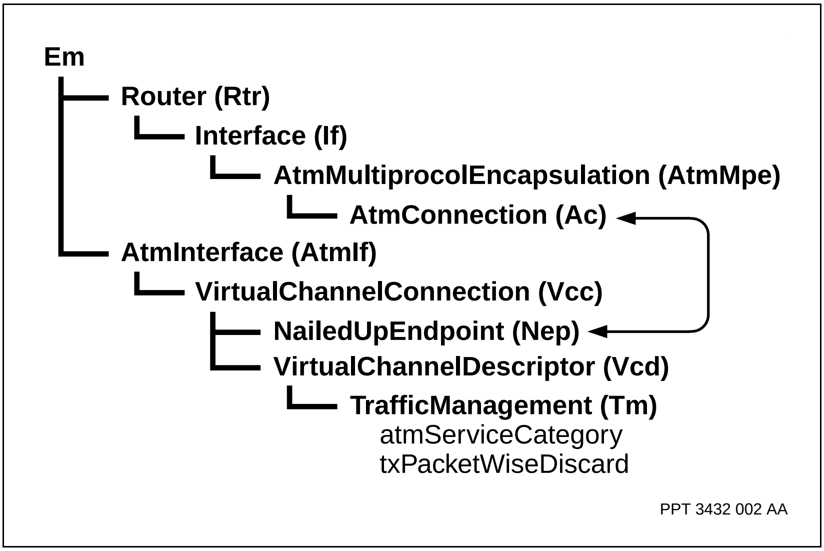
Variable definitions

Variable	Value
<ac_id>	is the instance value of the atm connection. Any mnemonic (for example, AC/1).
<atmif_id>	is the instance value of the *AtmIf* component.
(Sheet 1 of 2)	

Variable	Value
<desc_parm>	is the vector of five traffic parameters whose meanings are defined by the *txTrafficDescType* attribute. The instance value is the peak cell rate (PCR) setting in cell/second.
<desc_type>	is the instance value specifies the type of traffic management which is applied to the transmit direction of this connection. Values: 3 to 8.
<ipcos_value>	is the value assigned to the COS. It is recommended that the IP COS values 0, 1, 2, 3 correlate directly with ATM service categories UBR, nrtVBR, rtVBR, CBR.
<ip_address>	is a 32-bit address assigned to the interface from which the routing information is derived.
<ip_mask>	is the IP address of the netmask.
<rtr_name>	is the name of the router.
<vpi.vci>	is the identifier assigned to the VP and VC of the virtual channel.
(Sheet 2 of 2)	

Procedure job aid

Figure 13
ATM media for multi link congestion control component hierarchy



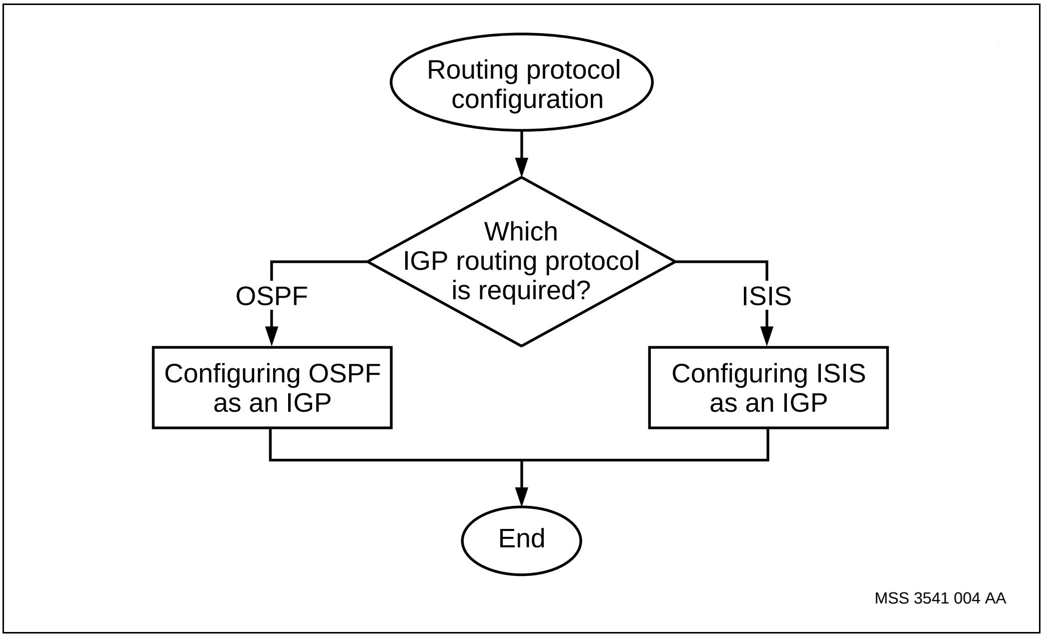
Chapter 4
Routing protocol configuration

The routing protocol configuration is used to set up IP address reachability information to allow for a source IP host to direct traffic to a destination IP host.

Routing protocol configuration procedures

This task flow shows you the sequence of procedures you perform to configure a routing protocol. To link to any procedure, go to “Routing protocol configuration procedure navigation” (page 58).

Figure 14
Routing protocol configuration procedures


Routing protocol configuration
Which IGP routing protocol is required?
OSPF
ISIS
Configuring OSPF as an IGP
Configuring ISIS as an IGP
End
MSS 3541 004 AA

Routing protocol configuration procedure navigation

- "Configuring OSPF as an IGP" (page 59)
- "Configuring ISIS as an IGP" (page 61)

Configuring OSPF as an IGP

Configure Open Shortest Path First (OSPF) as the Interior Gateway Protocol (IGP) and configure a passive OSPF interface on the primary loopback interface so that the loopback address is visible to the IGP running in the backbone.

Prerequisites

- Identify the primaryLoopback IP address. For more information, see “Configuring the router and primary loopback” (page 29).

Procedure steps

1 Add the *OSPF* component and set the *AreaEntry* component.

```
add -s Rtr/<rtr_name> Ospf AreaEntry/<area_id>
```

2 Set the OSPF routerId to match the primaryLoopback address.

```
set Rtr/<rtr_name> Ospf routerId <ip_address>
```

3 Add the OSPF interface under the primaryLoopback interface and set the type to passive.

```
add -s Rtr/<rtr_name> Interface/<ip_address> Ospfif
ifType passive
```

4 Add OSPF on other interfaces, for other media types (ATM or GigE). The OSPF is added on other interfaces to be able to reach the loopback interface.

```
add -s Rtr/<rtr_name> Interface/<ip_address> Ospfif
ifType <interface_type>
```

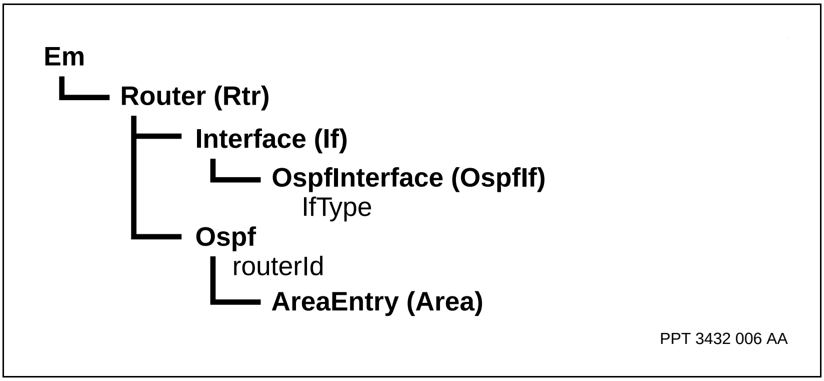
Variable definitions

Variable	Value
<area_id>	is a 32-bit address that contains the information about various areas that the router is connected to. This address uses the dotted decimal notation of an IP address, however it is not an IP address. The backbone area is always set to 0.0.0.0.
<interface_type>	is the type of this OSPF interface (pointToPoint, broadcast, etc.).
(Sheet 1 of 2)	

Variable	Value
<ip_address>	is a 32-bit address assigned to the interface from which the routing information is derived.
<rtr_name>	is the name of the router.
(Sheet 2 of 2)	

Procedure job aid

Figure 15
OSPF as an IGP component hierarchy



Configuring ISIS as an IGP

Configure the Intermediate System to Intermediate System (ISIS) as the Interior Gateway Protocol (IGP) running in the backbone and configure a passive ISIS interface on the primary loopback interface so that the loopback address is visible to the IGP running in the backbone.

Procedure steps

1 Add the *ISIS* protocol under the Router.

```
add Rtr/<rtr_name> Isis
```

2 Add the *Network Entity Title* (*Net*) component.

```
add Rtr/<rtr_name> Isis Net/<net_instance>
```

3 Add the *IsisLevel* component.

```
add Rtr/<rtr_name> Isis IsisLevel/<level>
```

Once ISIS is added at the router level, an ISIS interface must be added for each interface on which you want to use ISIS as an IGP.

4 Add the *IsisInterface* component.

```
add -s Rtr/<rtr_name> Interface/<ip_address> IsisIf
```

Once an instance of the *IsisInterface* component is added, at least one *Level* subcomponent must be added.

5 Add the *IsisIf Level* component.

```
add -s Rtr/<rtr_name> Interface/<ip_address> IsisIf
Level/<if_level>
```

6 Set the *IsisIf* component to passive mode for the primaryLoopback interface to make it visible to the IGP.

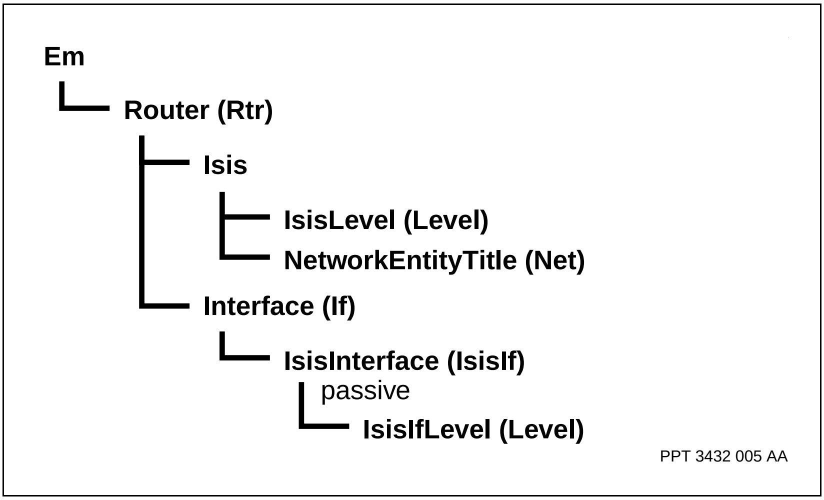
```
set Rtr/<rtr_name> If/<ip_address> IsisIf passive yes
```

Variable definitions

Variable	Value
<if_level>	represents the ISIS Level functionality that is associated with a single logical port. Level/1 only is supported.
<ip_address>	is a 32-bit address assigned to the interface from which the routing information is derived.
<level>	represents the ISIS protocol level specific capability under the router ISIS component. Level/1 only is supported
<net_instance>	is the value assigned to the *NetworkEntityTitle* component that identifies the ISIS router. The *NetworkEntityTitle* component consists of three parts: Area Address, System Identifier, and Network Selector (NSEL). Multiservice Switch systems support up to 3 NETs. On a router, all provisioned NETs must have the same System Identifier.
<rtr_name>	is the name of the router.

Procedure job aid

Figure 16
ISIS as an IGP component hierarchy



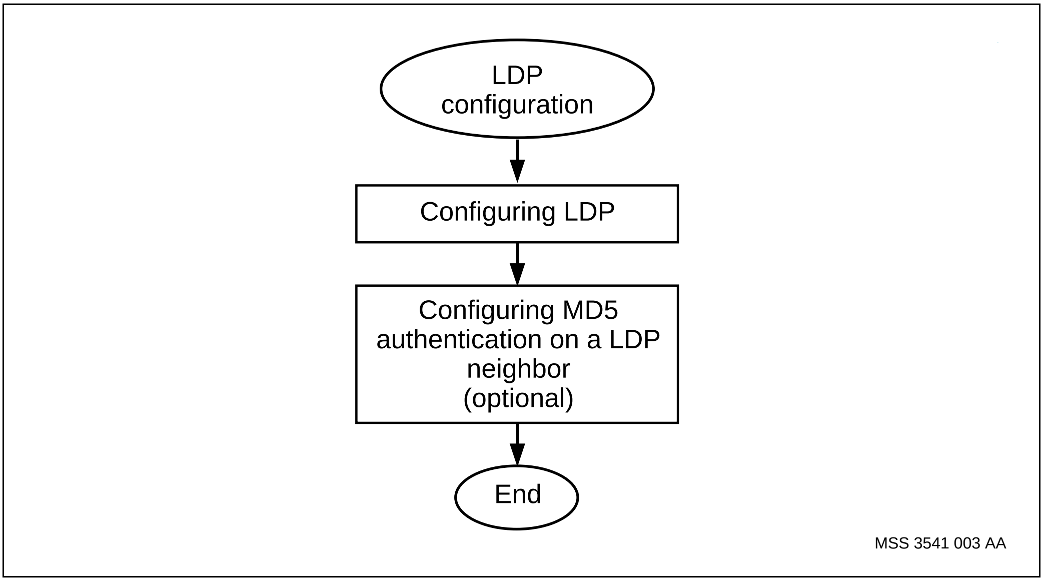
Chapter 5
LDP configuration

The label distribution protocol (LDP) configuration is used to trigger the exchange of MPLS labels for all the neighbor nodes.

LDP configuration procedures

This task flow shows you the sequence of procedures you perform to configure an LDP. To link to any procedure, go to “LDP configuration procedure navigation” (page 64).

Figure 17
LDP configuration procedures



LDP configuration procedure navigation

- "Configuring LDP" (page 65)
- "Configuring MD5 Authentication on an LDP neighbor" (page 66). This is an optional procedure that is done on an installed system, and is not part of initial system configuration.

Configuring LDP

Configure the Label Distribution Protocol (LDP) on the node to trigger the exchange of MPLS labels for all the neighbor nodes. This enables label switching of VPN traffic through tandem nodes in the backbone network.

Procedure steps

1 Add MPLS LDP.

```
add Rtr/<rtr_name> Ldp
```

2 Add an LDP interface on each interface that defines the adjacency. The interface can be ATM or Gigabit Ethernet, and is not required under any loopback interface.

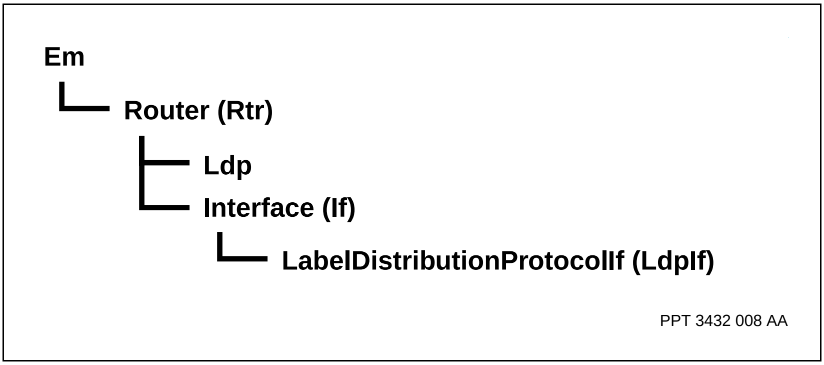
```
add Rtr/<rtr_name> If/<ip_address> LdpIf
```

Variable definitions

Variable	Value
<ip_address>	is a 32-bit address assigned to the interface from which the routing information is derived. Each LDP interface is created on the same interface as the corresponding OSPF or ISIS interface.
<rtr_name>	is the name of the router.

Procedure job aid

Figure 18
LDP component hierarchy



Configuring MD5 Authentication on an LDP neighbor

Configure MD5 authentication on an LDP neighbor to provide protection of neighbor relationships.

To change an MD5 key on an authenticated LDP neighbor, use the procedure "Changing an MD5 key on an authenticated LDP session" (page 68).

If an LDP MD5 Authentication alarm is raised, the operator can isolate the source of the issue using the Neighbor operational attributes described in Table 1, "LDP Neighbor operational attributes," (page 70).

Prerequisites

- A different MD5 key value is used for each LDP neighbor. A longer MD5 key is more difficult to break, however, the longer the key used, the greater the impact on performance. A key length of between 12 and 24 characters should be sufficiently secure without having too great an impact on performance

Procedure steps

1 Add the LDP neighbor.

```
add Rtr/<rtr_name> Ldp Nbr/<ip_addr>,<LabelSpaceId>
```

For provisioned LDP neighbors, the *LabelSpaceId* must be 0 (zero).

The *Descriptor* subcomponent is automatically created.

2 Set the MD5 key under the *Descriptor* component.

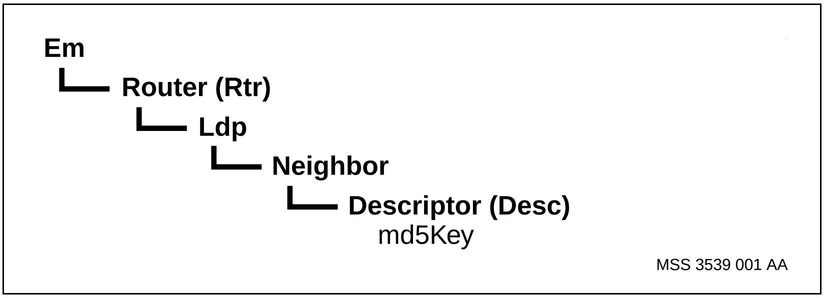
```
set Rtr/<rtr_name> Ldp Nbr/<ip_addr>,<LabelSpaceId>
Desc md5Key <ASCII_string>
```

Variable definitions

Variable	Value
<ASCII_string>	is the MD5 key value, 0-69 ASCII characters. A null string indicates that MD5 authentication is not to be used on the connection to this neighbor. Any printable ascii character is allowed. If a space is used within the key, then the whole key must be included in quotes. For this reason a " character is the only printable ascii character not allowed as part of a key.
<ip_addr>	is the IP address.
<LabelSpaceId>	is the value for the LabelSpaceId element, value 0-65535. For provisioned LDP neighbors, the LabelSpaceId must be 0 (zero).
<rtr_name>	is the name of the router.

Procedure job aid

Figure 19
LDP component hierarchy for MD5 authentication on an LDP neighbor



Changing an MD5 key on an authenticated LDP session

Transition MD5 keys without having to terminate the LDP neighbor session for which the key is being changed.

Use a different MD5 key value be used for each LDP neighbor.

When changing keys on a LDP session between a Multiservice Switch node and a non-Multiservice Switch, change the key on the Multiservice Switch node, to minimize the outage time.

A longer MD5 key will be more difficult to break, however, the longer the key used, the greater the impact on performance. A key length of between 12 and 24 characters should be sufficiently secure without having too great an impact on performance.

Procedure steps

1 Set the *keyTransitionDelay* attribute on the local router to a value (in minutes) sufficient to complete the changes at both ends of the connection.

```
set Rtr/<rtr_name> Ldp Nbr/<ip_addr>,<LabelSpaceId>
Desc keyTransitionDelay <n>
```

2 Set the *keyTransitionDelay* attribute on the remote end.

```
set Rtr/<rtr_name> Ldp Nbr/<ip_addr>,<LabelSpaceId>
Desc keyTransitionDelay <n>
```

3 Set the new MD5 key value on the local router.

```
set Rtr/<rtr_name> Ldp Nbr/<ip_addr>,<LabelSpaceId>
Desc md5Key <ASCII_string>
```

4 Set the new MD5 key value on the remote router, if the remote router is also a Multiservice Switch.

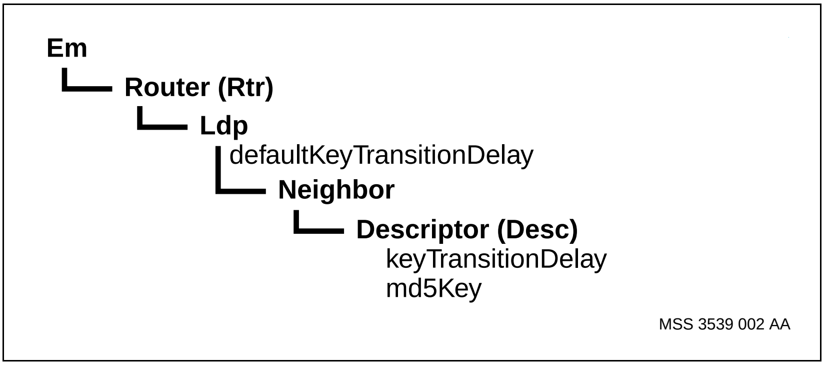
```
set Rtr/<rtr_name> Ldp Nbr/<ip_addr>,<LabelSpaceId>
Desc md5Key <ASCII_string>
```

Variable definitions

Variable	Value
<ASCII_string>	is the MD5 key value, 0-69 ASCII characters.
<ip_addr>	is the IP address.
<LabelSpaceId>	iis the value for the LabelSpaceId element, value 0-65535. For provisioned LDP neighbors, the LabelSpaceId must be 0 (zero).
<n>	is the value for the *keyTransitionDelay*. This is the time that can elapse before the local router transitions to using the new key for transmission (the new key value is immediately accepted for reception). This delay equates to the amount of time the operator has to provision both the local and remote ends with the new key, without losing service. The value can be inherited from the *Ldp* component's *keyTransitionDelay* attribute by setting the value to sameAsLdp. Range: sameAsLdp, 1 to 65535 (minutes), the default is sameAsLdp.
<rtr_name>	is the name of the router.

Procedure job aid

Figure 20
LDP component hierarchy for changing an MD5 key on an authenticated LDP session



Procedure job aid

Table 1
LDP Neighbor operational attributes

Attribute	Description
authenticationType	is the attribute that indicates which type of authentication is used on the TCP connection to the specified Neighbor. Values are none and md5.
noAuthenticationPkts	is the attribute that indicates the number of TCP packets have been received from the specified Neighbor which were unauthenticated when authentication was expected. When this counter is increasing it could mean that the Neighbor is not configured to use authentication or a hacker is attempting to disrupt the LDP connection and is injecting unauthenticated packets.
badAuthenticationPkts	is the attribute that indicates the number of TCP packets have been received from the specified Neighbor which were authenticated incorrectly. When this counter is increasing could mean that the Neighbor is configured with a different key or a hacker is attempting disrupt the LDP connection and is injecting packets authenticated using a different key.
unexpectedAuthenticationPkts	is the attribute that indicates the number of TCP packets have been received from the specified Neighbor which were authenticated when authentication was not expected. When this counter is increasing it could mean that the Neighbor is configured to use authentication or a hacker is attempting to disrupt the LDP connection and is injecting authenticated packets.

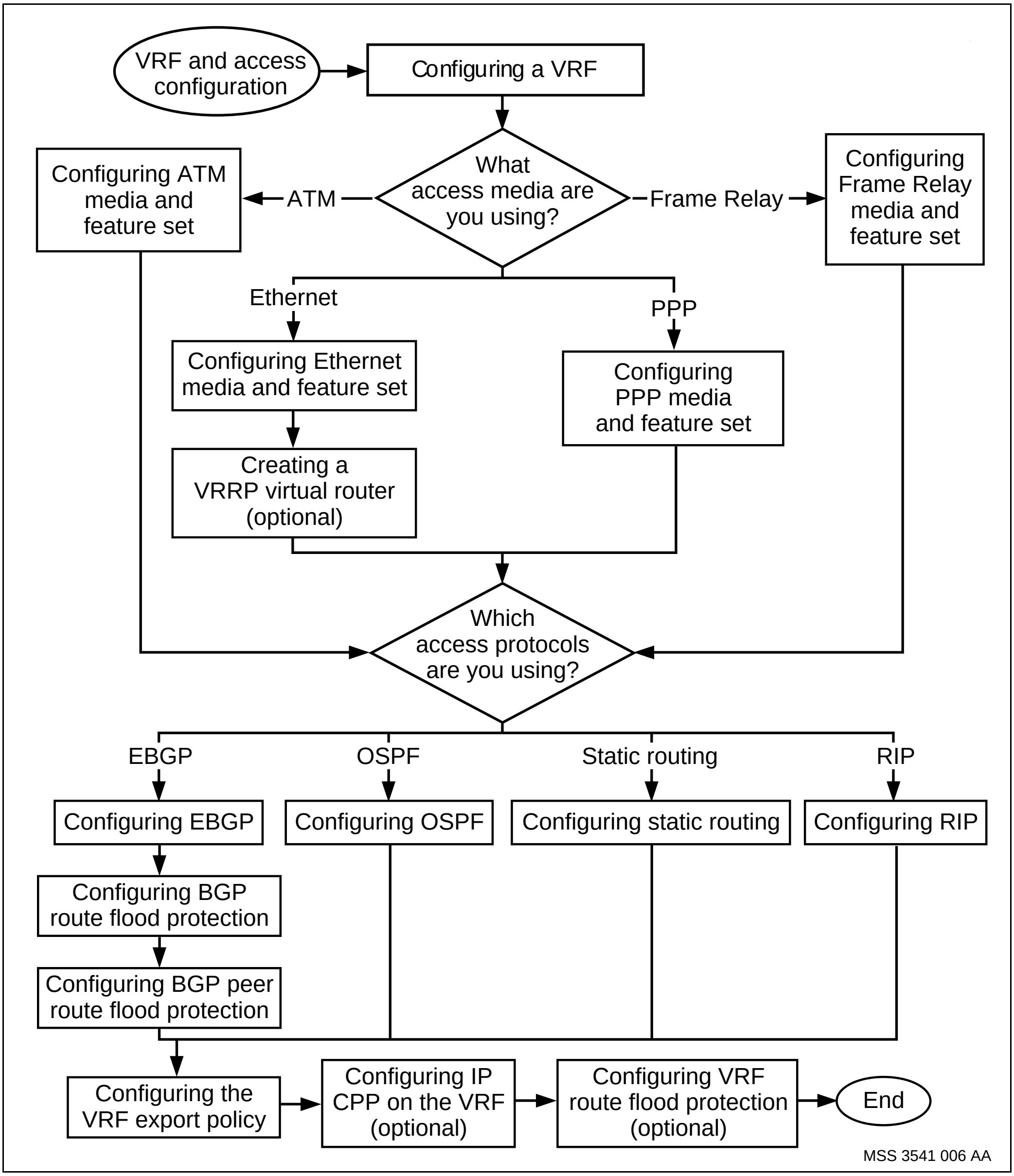
Chapter 6
VRF and access configuration

Configure VPN route forwarders (VRFs) and access interfaces on the node. The VRF is the entity that groups together the routing information learned from several connected sites. A local site can belong to one or more VPNs.

VRF and access configuration procedures

This task flow shows you the sequence of procedures you perform to configure the VRF and the access on the node. To link to any procedure, go to “VRF and access configuration procedure navigation” (page 73).

Figure 21
VRF and access configuration procedures


VRF and access configuration
Configuring a VRF
What access media are you using?
ATM
Frame Relay
Configuring ATM media and feature set
Configuring Frame Relay media and feature set
Ethernet
PPP
Configuring Ethernet media and feature set
Creating a VRRP virtual router (optional)
Configuring PPP media and feature set
Which access protocols are you using?
EBGP
OSPF
Static routing
RIP
Configuring EBGP
Configuring OSPF
Configuring static routing
Configuring RIP
Configuring BGP route flood protection
Configuring BGP peer route flood protection
Configuring the VRF export policy
Configuring IP CPP on the VRF (optional)
Configuring VRF route flood protection (optional)
End
MSS 3541 006 AA

VRF and access configuration procedure navigation

- “Configuring a VRF” (page 74)
- “Configuring ATM media and feature set” (page 77)
- “Configuring PPP media and feature set for VRF in RFC2547 mode” (page 81)
- “Configuring Ethernet media and feature set” (page 83)
- “Creating a VRRP virtual router” (page 85)
- “Configuring frame relay media and feature set” (page 94)
- “Configuring EBGP” (page 98)
- “Configuring BGP route flood protection” (page 103)
- “Configuring BGP peer route flood protection” (page 105)
- “Configuring OSPF” (page 107)
- “Configuring static routing” (page 109)
- “Configuring RIP” (page 110)
- “Configuring the VRF export policy” (page 112)
- “Configuring IP CPP on the VRF” (page 114)
- “Configuring VRF route flood protection” (page 116)

Configuring a VRF

Configuring a VPN Route Forwarder (VRF), with its own unique route distinguisher, allows two different VPNs, with no site in common, to use IPv4 address spaces that overlap with each other. Configuring route targets under the *Vrf* component defines VPN memberships for local sites connected into the PE node, so that two or more customer sites can communicate. In Nortel Networks Multiservice Switch systems, the VRF associates all routes that lead to a particular site with one or more target VPNs.

Procedure steps

1 Add a *VpnRouteForwarder* component.

```
add Rtr/<rtr_name> Vrf/<vrf_name>
```

2 Add the VRF’s applicable Route Distinguisher.

```
add Rtr/<rtr_name> Vrf/<vrf_name> Rd/<rd_value>
```

3 Add the *RouteTarget* component.

```
add Rtr/<rtr_name> Vrf/<vrf_name> Rt/<rt_value>
```

4 Set the *mode* attribute for the *RouteTarget* component.

```
set Rtr/<rtr_name> Vrf/<vrf_name> Rt/<rt_value> mode
<mode>
```

5 Add the *Interface* component.

```
add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
```

6 Set the *netMask* attribute.

```
set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
netMask <mask>
```

7 Add the media to the VRF interfaces.

If you are adding an ATM media to the VRF for each access link to a CE device, complete this command:

```
add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe
```

If you are adding the loopback interface to verify the PE to PE node connections without any CE device connected, complete this command:

```
add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address> Lb
```

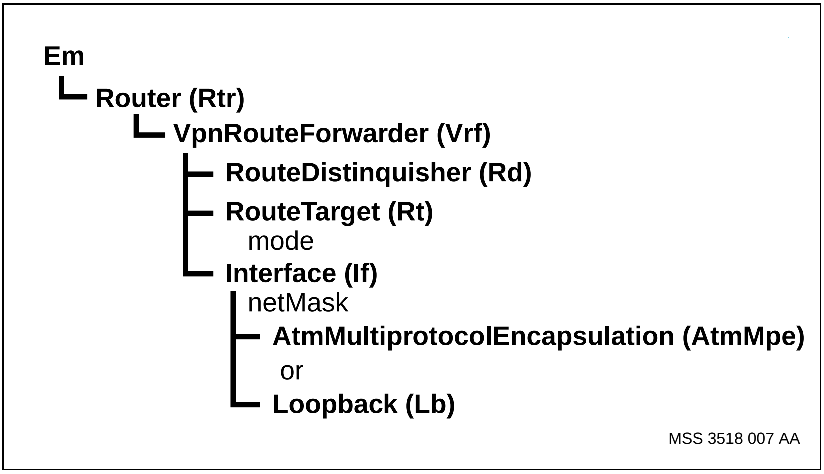
```
set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
netmask 255.255.255.255
```

Variable definitions

Variable	Value
<ip_address>	is the 32-bit address assigned to this interface.
<mask>	is the network mask to be used with the IP address.
<mode>	identifies the mode. Values are both, import, and export. Default = both.
<rd_value>	specifies the route distinguisher for the VRF.The syntax can be in the form of Type 0 or Type 1. Type 0 = a 2 byte AS number and a 4 byte number assigned by the SP (AS:xxxx). Type 1 = 4 byte ipv4 address and a 2 byte number assigned by the SP (ipv4:xx). The RD must be unique on the PE node.
<rtr_name>	is the name of the router.
<rt_value>	identifies the VPN. For two or more customer sites to talk, they must have at least one route target in common. The syntax can be in the form of Type 0 or Type 1. Type 0 = a 2 byte AS number and a 4 byte number assigned by the SP (AS:xxxx). Type 1 = 4 byte ipv4 address and a 2 byte number assigned by the SP (ipv4:xx).
<vrf_name>	is the name of the VPN route forwarder.

Procedure job aid

Figure 22
VRF and route distinguisher component hierarchy



Configuring ATM media and feature set

Configure ATM Multiprotocol Encapsulation (AtmMpe) media for a VRF Interface instance to define ATM connectivity between the PE node and the CE route. The AtmMpe media enables IP data traffic to be received and transmitted over AAL5 using RFC 1483 encapsulation methods. IP over ATM is a supported access media for BGP/MPLS VPN. The following mode/ encapsulations are supported:

- direct (single AC subcomponent under AtmMpe linked to IP interface)
- VCC-bundle (multiple AC subcomponents under the AtmMpe) for PVC or SPVC
- IP over ATM SPVC between ATM UNI and ATM MPE
- IP bridge termination over ATM (LLC-SNAP encapsulation for Ethernet frames)

Prerequisites

For NPVC, you must do the following:

- Configure the *AtmIf* component. See NN10600-710 *Nortel Networks Multiservice Switch 7400/15000/20000 ATM Configuration Management*

For SPVC, you must do the following:

- Configure the required ATM routing components. See NN10600-710 *Nortel Networks Multiservice Switch 7400/15000/20000 ATM Configuration Management*.
- Configure the *Pnni* component under the *AtmRouting* and *AtmInterface* components. See NN10600-710 *Nortel Networks Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

Procedure steps

1 Configure the feature set on ATM supported access cards.

```
set Sw Lpt/<lpt_name> featureList rfc2547bis atmMpe ip
```

2 Configure the ip routes pool capacity on a Multiservice Switch 7400 or Multiservice Switch 15000 ATM access LP.

```
add -s Lp/<lp_number> eng fcrc pqc
```

```
set Lp/<lp_number> eng fcrc pqc ov
ipRoutesPoolCapacity 64000
```

3 Set the ATM connection for direct mode (single AC subcomponent).

```
set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id> atmConnection AtmIf/<n> Vcc/
<vpi.vci> Nep
```

4 Set the ATM connection for VCC-bundle mode (multiple AC subcomponents).

```
add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id1> atmConnection AtmIf/<n> Vcc/
<vpi.vci1> Nep

add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id2> atmConnection AtmIf/<n> Vcc/
<vpi.vci2> Nep

add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id3> atmConnection AtmIf/<n> Vcc/
<vpi.vci3> Nep

add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id4> atmConnection AtmIf/<n> Vcc/
<vpi.vci4> Nep

set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id1> ipCos 0

set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id2> ipCos 1

set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id3> ipCos 2

set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id4> ipCos 3
```

5 Set the ATM connection for IP over ATM SPVC mode.

When CPE is setting up the SPVC connection:

```
add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe stp
```

```
add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id> dstpvc
```

When the VRF is setting up the SPVC connection:

```
add -s Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id> SourcePvc
```

```
set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id> SourcePvc remoteAddress
<remote_NSAP>, remoteConnectionIdentifier
<remote_ConnID>
```

6 Set the ATM connection for IP bridge termination over ATM mode.

```
add -s Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe Ac/<ac_id> atmConnection AtmIf/<n> Vcc/
<vpi.vci> Nep

set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
AtmMpe encapType llcBridgeEncap, maxTransmissionUnit
1500
```

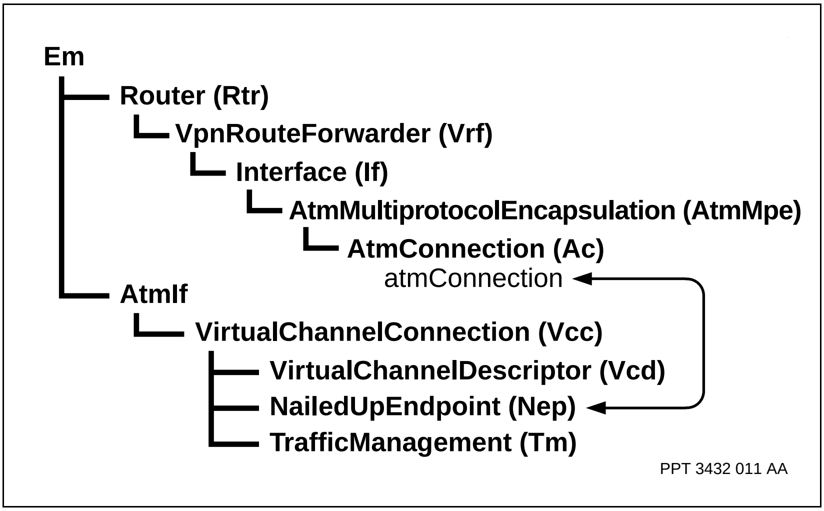
Variable definitions

Variable	Value
<ac_id>	is the instance value of the atm connection component under AtmMpe. Any mnemonic (for example, AC/1).
<atmif_id>	is the instance value of the *AtmIf* component.
<ip_address>	is the 32-bit address assigned to this interface.
<lp_number>	is the number assigned to the LP.
<lpt_name>	is the name assigned to the software Lpt for this Lp.
<remote_ConnID>	is the remote VCI value (assuming VPI 0) if the SPVC terminates on an AtmIf, or the AC instance value if the SPVC terminates into a remote AtmMpe.
<remote_NSAP>	is the ATM NSAP address of the remote (called end) ATM interface of the SPVC connection.
(Sheet 1 of 2)	

Variable	Value
<rtr_name>	is the name of the router.
<vpi.vci>	is the identifier assigned to the VP and VC of the virtual channel. It is a good practice to assign the vci value of the atm connection to the ac_id (example: set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address> AtmMpe Ac/40 atmConnection AtmIf/<n> Vcc/0.40 Nep. It is the ac_id that is displayed in the ARP table for the IP address mapping to the layer 2 attribute (inverse ARP).
(Sheet 2 of 2)	

Procedure job aid

Figure 23
ATM media and feature set component hierarchy



Configuring PPP media and feature set for VRF in RFC2547 mode

Configure PPP media and feature set for VRF in RFC2547 mode in order to carry IP over PPP.

Prerequisites

- Required PPP interfaces are configured. For more information about PPP interface configuration, see NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.
- Hardware connections for the PPP interface are configured.

Procedure steps

1 Configure the feature set.

```
set sw lpt/<lpt_name> featurelist ppp ip rfc2547bis
```

2 Set the Vrf interface netmask.

```
set Rtr/<rtr_name> vrf/<vrf_name> If/<ip_address>
netmask <netmask>
```

3 If the netmask is a 32-bit mask, configure the Vrf interface link destination address.

```
set Rtr/<rtr_name> vrf/<vrf_name> If/<ip_address>
linkDestinationAddress <remote_ip_address>
```

4 Add a PPP media application to the Router Vrf interface.

```
add -s Rtr/<rtr_name> vrf/<vrf_name> If/<ip_address>
PppApplication
```

5 Link the application to the corresponding PPP media interface that is under the root component.

```
set Rtr/<rtr_name> vrf/<vrf_name> If/<ip_address>
PppApplication linkToMedia ppp/<ppp_instance>
```

Variable definitions

Variable	Value
<ip_address>	is the Internet Protocol address for the interface.
<lpt_name>	is the label for the interface.
<netmask>	is the is the interface netmask.
<ppp_instance>	is the instance number of the PPP media interface.
<remote_ip_address>	is the IP address configured at the other end of this PPP interface.
<rtr_name>	is the name of the router.
<vrf_name>	is the name of the VPN route forwarder.

Configuring Ethernet media and feature set

Configure the Ethernet media (port or virtual LAN [VLAN]) to carry IP over Ethernet.

For more information on the Ethernet media, refer to the chapter on Multiservice Switch IP fundamentals in NN10600-800 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Technology Fundamentals*.

Prerequisite

- To configure an Ethernet interface in port mode or VLAN mode, see NN10600-580 *Nortel Networks Multiservice Switch 7400/15000/20000 Ethernet Service Operations*.

Procedure steps

1. Add *EnetApplication* component as the media interface.

   ```
   add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address> Enet
   ```

2. Link the *EnetApplication* component to the Ethernet media in port mode.

   ```
   set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address> Enet
   linkToEnetIf La/<la_number>
   ```

   or

   ```
   set La/<la_number> enetConnection Rtr/<rtr_name> Vrf/
   <vrf_name> If/<ip_address> Enet
   ```

3. Link the *EnetApplication* component to the Ethernet media in VLAN mode.

   ```
   set Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address> Enet
   linkToEnetIf La/<la_number> Vlan/<vlan_number>
   ```

   or

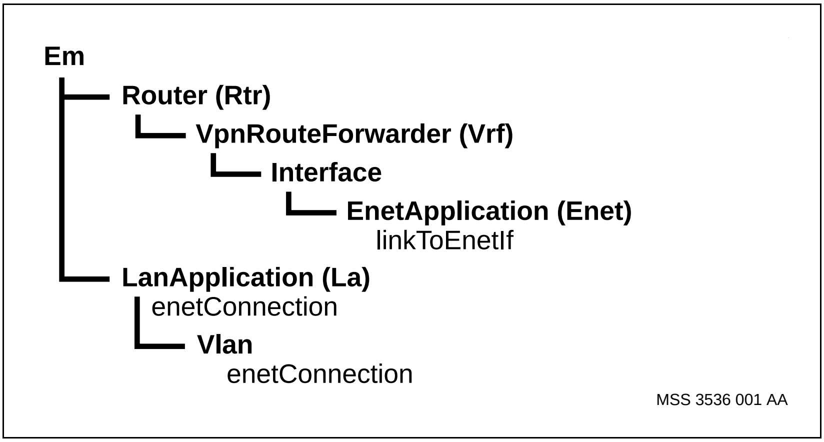
   ```
   set La/<la_number> Vlan/<vlan_number> enetConnection
   Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address> Enet
   ```

Variable definitions

Variable	**Value**
<ip_address>	is the 32-bit address assigned to this interface.
<la_number>	is the number of the *LanApplication* instance.
<rtr_name>	is the name of the router.
<vlan_number>	is the identifier of the *vlan* instance.
<vrf_name>	is the name of the VPN route forwarder.

Procedure job aid

Figure 24
Ethernet media and feature set component hierarchy



Creating a VRRP virtual router

Create a virtual router redundancy protocol (VRRP) virtual router to provide VPN route forwarder (VRF) redundancy.

Optionally, set the VRRP advertisement interval on both the master and backup VRRP virtual routers (VRs) through the *advertisementInterval* attribute. The master and backup VRRP VRs require the same value.

Procedure steps

1 Add the *Vrrp* component to the Ethernet interface. The VRRP virtual router (VR) is the master.

```
add Rtr/<router_A> Vrf/<vrf_name> If/
<interface_address> Vrrp/<vrrp_name>
```

2 Set the *ipAddresses* attribute for the VRRP VR with which it is associated.

```
set Rtr/<router_A> Vrf/<vrf_name> If/
<interface_address> Vrrp/<vrrp_name> ipAddresses
<ipaddress>
```

3 Set the *priority* attribute of the master VRRP VR.

If the master VRRP VR owns the IP address of the VRF, set the priority to 255.

```
set Rtr/<router_A> Vrf/<vrf_name> If/<ip_address>
Vrrp/<vrrp_name> priority <priority_value>
```

4 Add the *Vrrp* component to the Ethernet interface. The VRRP VR is a backup.

```
add Rtr/<router_B> Vrf/<vrf_name> If/
<interface_address> Vrrp/<vrrp_name>
```

5 Set the *ipAddresses* attribute of VRRP VR.

```
set Rtr/<router_B> Vrf/<vrf_name> If/
<interface_address> Vrrp/<vrrp_name> ipAddresses
<ipaddress>
```

6 Set the *priority* attribute of the backup VRRP VR.

```
set Rtr/<router_B> Vrf/<vrf_name> If/
<interface_address> Vrrp/<vrrp_name> priority
<priority_value>
```

When you have multiple backup VRRP VRs, it is very important to set the priority because it defines which backup VRRP VR is active when the master VRRP VR has a failure.

7 Optionally, set the *advertisementInterval* attribute on the master VRRP VR. Ensure that the *advertisementInterval* attribute value is the same for the backup and master VRRP VRs.

```
set Rtr/<router_A> Vrf/<vrf_name> If/
<interface_address> Vrrp/<vrrp_name>
advertisementInterval <ad_inter>
```

8 Optionally, set the *advertisementInterval* attribute on the backup VRRP VR. Ensure that the *advertisementInterval* attribute value is the same for the backup and master VRRP VRs.

```
set Rtr/<router_B> Vrf/<vrf_name> If/
<interface_address> Vrrp/<vrrp_name>
advertisementInterval <ad_inter>
```

9 Activate the changes. See “Activating configuration changes” (page 22).

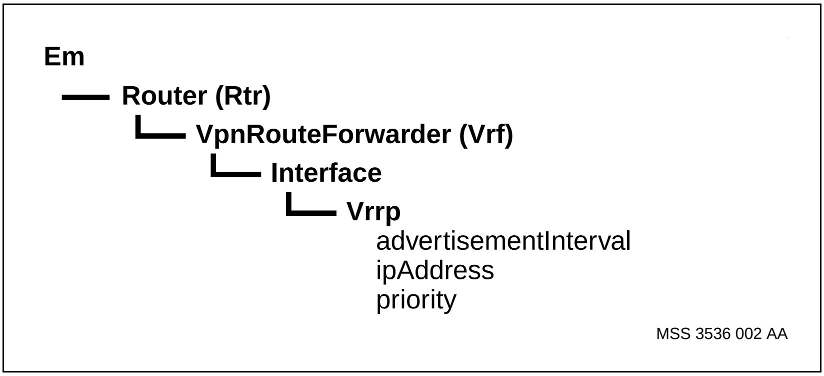
Variable definitions

Variable	Value
<ad_inter>	is the advertisement interval time in seconds.
<interface_address>	is the 32-bit address assigned to this interface.
<ipaddress>	specifies one or more IP addresses associated with the *Vrrp* instance. Typically, this IP address is the same as the IP addresses of the master VRRP VR.
<priority_value>	is a decimal value between 1 and 255. The higher priority is for the master VRRP VR.
<router_A>	is an instance name of a Multiservice Switch RTR on node A.
<router_B>	is an instance name of a Multiservice Switch RTR on node B.
(Sheet 1 of 2)	

Variable	Value
<vrf_name>	is the name of the VPN route forwarder.
<vrrp_name>	is instance name of the VRRP VR.
(Sheet 2 of 2)	

Procedure job aid

Figure 25
VRRP virtual router component hierarchy



Example procedure for creating a VRRP virtual router associated with an Ethernet interface in port mode

Prior to starting this example procedure, ensure that Nodes 1 and 2 have the *LanApplication* component in port mode. For more information about Configuring an Ethernet interface in port mode, see NN10600-580 *Nortel Networks Multiservice Switch 7400/15000/20000 Ethernet Service Operations*.

1 Set the *featureList* attribute to include ipvrrp.

```
set sw Lpt/ethernet featureList ipvrrp
```

2 Activate the changes without ending your provisioning session. See “Activating configuration changes” (page 22).

For Node 1, complete the following steps:

3 Add the *Vrrp 1* component to the Ethernet interface. The VRRP virtual router (VR) is the master.

```
add Rtr/1 Vrf/1 If/<interface_address> Vrrp/1
```

4 Set the *ipAddresses* attribute that the VRRP VR is associated with.

```
set Rtr/1 Vrf/1 If/<interface_address> Vrrp/1
ipAddresses <ipaddress_1B>
```

5 This VRRP VR is the master because it owns the IP address set in step 4. Set the *priority* attribute of the master VRRP VR to 255.

```
set Rtr/1 Vrf/1 If/<interface_address> Vrrp/1 priority
255
```

For Node 2, complete the following steps:

6 Add the *Vrrp* component to Ethernet interface. The VRRP virtual router (VR) is the backup.

```
add Rtr/1 Vrf/1 If/<interface_address> Vrrp/1
```

7 Set the *ipAddresses* attribute that the VRRP VR is associated with.

```
set Rtr/1 Vrf/1 If/<interface_address> Vrrp/1
ipAddresses <ipaddress_1B>
```

8 This VRRP VR does not own the IP address set in step 7 because it is the backup VRRP VR. Set the *priority* attribute of the backup VRRP VR to a value less than 255.

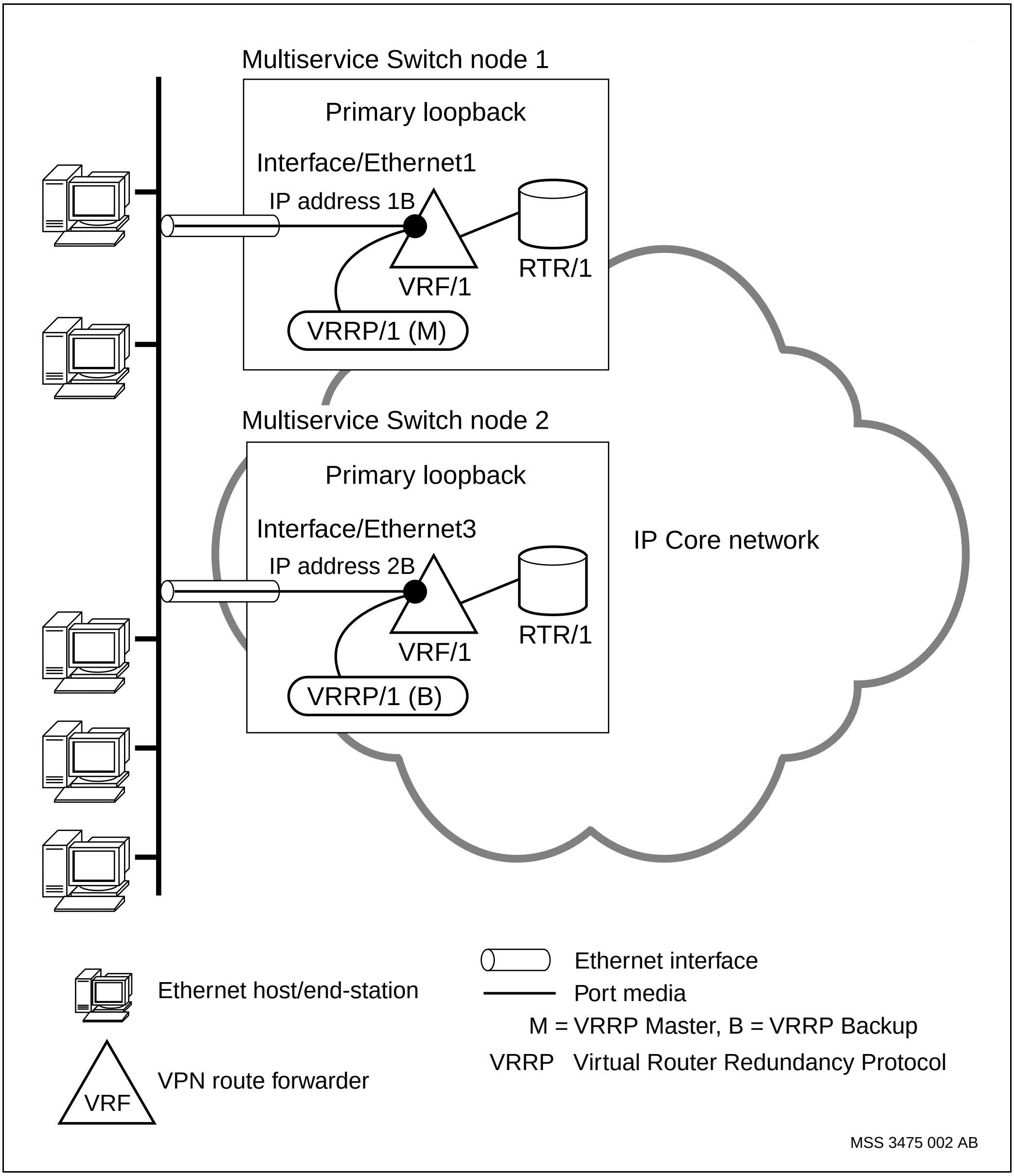
```
set Rtr/1 Vrf/1 If/<interface_address> Vrrp/1 priority
<priority_value>
```

9 Activate configuration changes. See “Activating configuration changes” (page 22).

Variable definitions

Variable	**Value**
<ipaddress_1B>	is the IP address of the master VRRP VR.
<priority_value>	is a decimal value between 1 and 255. The higher priority is for the master VRRP VR.

Figure 26
Example figure for creating a VRRP virtual router associated with an Ethernet interface in port mode



Example procedure for creating a VRRP virtual router associated with a VLAN on an Ethernet interface

Prior to starting this example procedure, ensure that Nodes 3 and 4 have the *LanApplication* component in VLAN mode. For more information about Configuring an Ethernet interface in VLAN mode, see NN10600-580 *Nortel Networks Multiservice Switch 7400/15000/20000 Ethernet Service Operations*

For Node 3, complete the following steps:

1 Add the *Vrrp 4* component to the interface. The VRRP virtual router (VR) is the master.

```
add Rtr/1 Vrf/1 If/<interface_address> Vrrp/4
```

2 Set the *ipAddresses* attribute that the VRRP VR is associated with.

```
set Rtr/1 Vrf/1 If/<interface_address> Vrrp/4
ipAddresses <ipAddress_3B>
```

3 This VRRP VR is the master because it owns the IP address set in step 2. Set the *priority* attribute of the master VRRP VR to 255.

```
set Rtr/1 Vrf/1 If/<interface_address> Vrrp/4 priority
255
```

4 Activate the changes without ending your provisioning session. See “Activating configuration changes” (page 22).

For Node 4, complete the following steps:

5 Add the *Vrrp 4* component to the interface. The VRRP VR is a backup.

```
add Rtr/1 Vrf/1 If/<interface_address> Vrrp/4
```

6 Set the *ipAddresses* attribute that the VRRP VR is associated with.

```
set Rtr/1 Vrf/1 If/<interface_address> Vrrp/4
ipAddresses <ipAddress_3B>
```

7 This VRRP VR does not own the IP address set in step 6 because it is the backup VRRP VR. Set the *priority* attribute of the backup VRRP VR to a value less than 255.

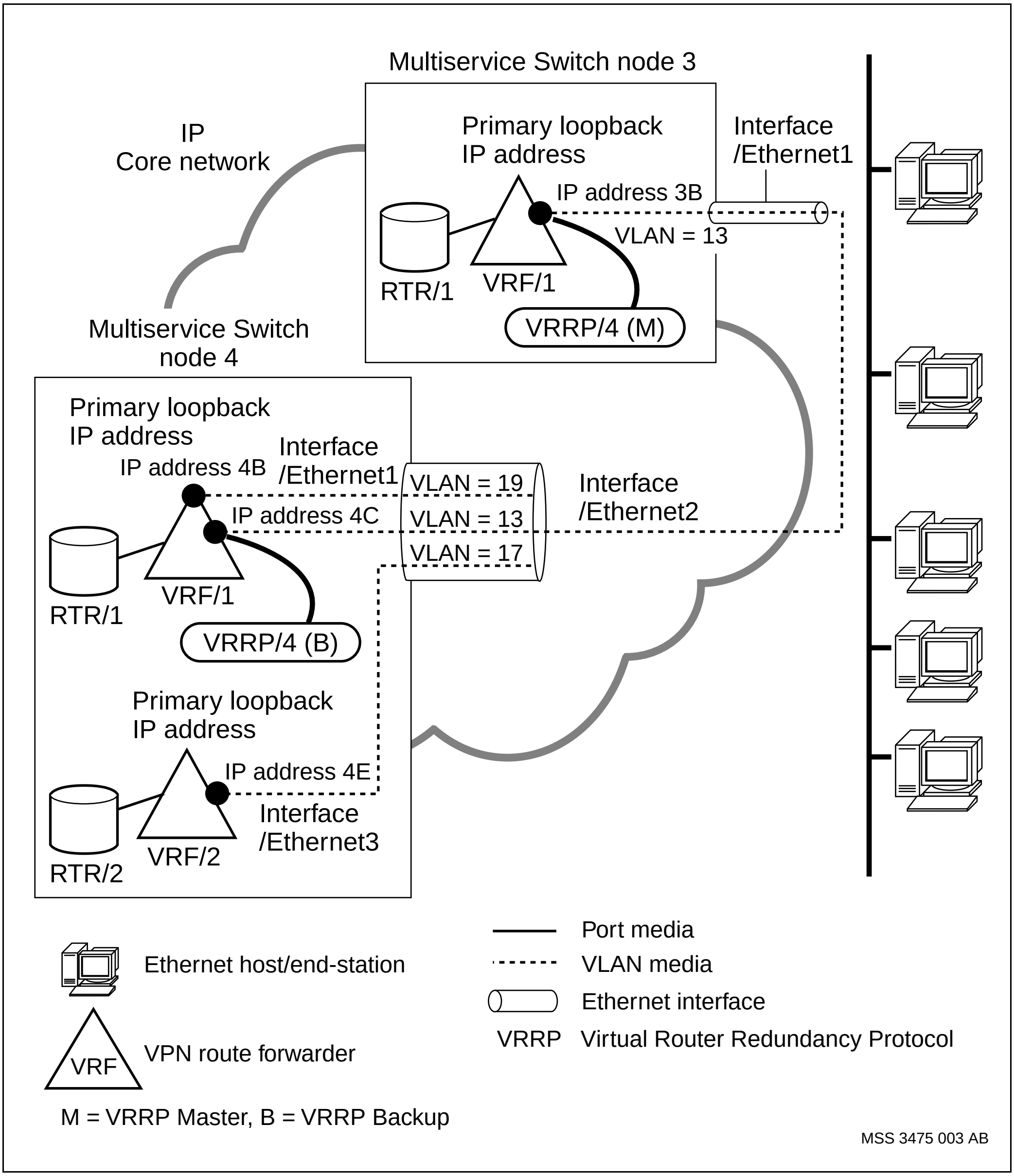
```
set Rtr/1 Vrf/1 If/<interface_address> Vrrp/4 priority
<priority_value>
```

8 Activate configuration changes. See “Activating configuration changes” (page 22).

Variable definitions

Variable	Value
<ipaddress_3B>	is the IP address of the master VRRP VR.
<priority_value>	is a decimal value between 1 and 255. The higher priority is for the master VRRP VR.

Figure 27
Example figure for creating a VRRP virtual router associated with a VLAN on an Ethernet interface



Configuring frame relay media and feature set

Configure frame relay media for a VRF Interface instance to define Frame relay physical connectivity between the CE and the PE node. Frame relay access to a VRF can be achieved through a direct connection over an existing channel (Ip Optimized direct) or through a backhaul scenario (DPRS trunk). You can use a software configuration with a vframer component or for enhanced quality of service, use a hairpin configuration.

Prerequisites

- A *FrUni Dlci IpConn* component must be provisioned and linked to the IpoDlci media. This configuration allows for direct Frame relay access to an IP VPN network. For more information about configuring Frame relay, see NN10600-901 *Nortel Networks Multiservice Switch 7400/ 15000/20000 Frame Relay Configuration Management*.

Procedure steps

1 Configure the feature set (frUniAllDprs, is for backhaul access only).

```
set Sw Lpt/<lpt_name> featureList rfc2547bis
frUniIpOptimized, ip, frUniAllDprs
```

2 Add the Ip Optimized DLCI media to the VRF interface.

```
add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
IpODlci Frc/<frc>
```

3 Set the numberOfEmissionsQs attribute to 4.

```
set Fruni/<fruni_inst> numberOfEmissionsQs 4
```

Regardless of the number of DLCIs being used, it is recommended that the numberOfEmissionsQs attribute be set to 4, so that 4 emission queues are available instead of the default 2.

4 Link the Ip Optimized DLCI media to the Ip Connection under the appropriate Fruni and Dlci (ipOptimized direct).

```
set Rtr/<rtr_name> vrf/<vrf_name> If/<ip_address>
IpODlci Frc/<frc> linkToFrDlci Fruni/<fruni_inst>
Dlci/<dlci> IpConnection
```

5 Add the DPRS Fruni component for CE connectivity.

```
add FrUni/<fruni_inst>
```

6 Remove the physical framer and add the virtual framer for the software connection.

```
del FrUni/<fruni_inst> framer
```

```
add FrUni/<fruni_inst> Vframer
```

7 Link the IP optimized FrUni to the DPRS FrUni (direct call) and associate it with a frame relay based LP.

```
set FrUni/<fruni_inst> Vframer otherVirtualFramer
FrUni/<fruni_inst> VFramer
```

```
set FrUni/<fruni_inst> Vframer logicalProcessor Lp/
<lpt_name>
```

8 Set the DPRS FrUni for remote signalling onto a remote Multiservice Switch node for CE connectivity. This DLCI will signal the remote X121 FrUni DLCI through the DPRS backbone

```
add FrUni/<fruni_inst> Dlci/<dlci> dc
```

9 If rate enforcement and LMI is required, it should be applied on this FrUni, facing the DPRS core.

```
set FrUni/<fruni_inst> Dlci/<dlci> Sp rateEnforcement
on
```

```
set FrUni/<fruni_inst> lmi proc itu (or ansi)
```

10 Set the DP to DE mapping.

```
set FrUni/<fruni_inst> Dlci/<dlci> IpConn
dpToDeMapLevel <level>
```

11 For the hairpin scenario, link the framer to an existing physical port.

```
add FrUni/<fruni_inst>
```

```
set FrUni/<fruni_inst> Vframer logicalProcessor Lp/
<lpt_name> port/<port>
```

```
add FrUni/<fruni_inst> Dlci/<dlci> dc
```

```
set FrUni/<fruni_inst> Dlci/<dlci> Sp rateEnforcement
on
```

```
set FrUni/<fruni_inst> lmi proc itu (or ansi)
```

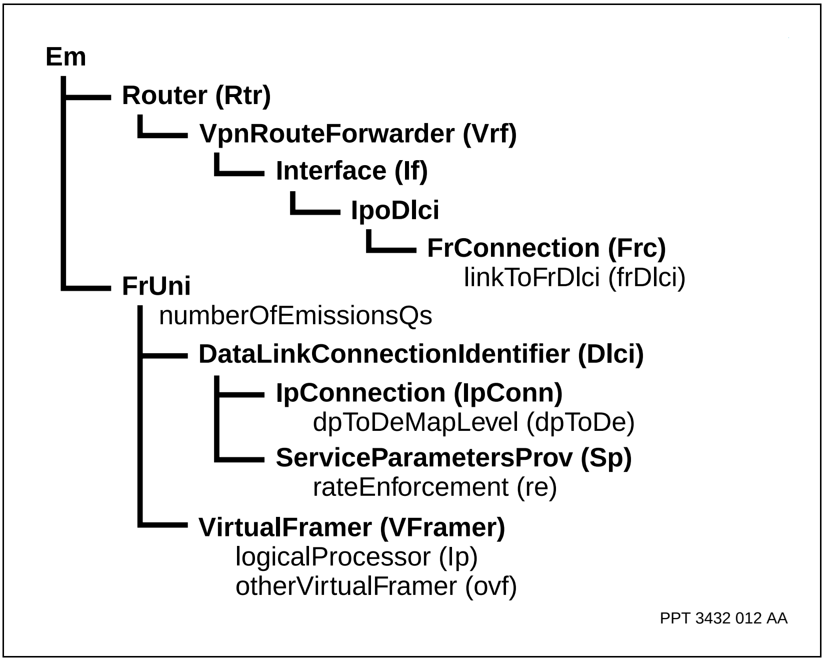
```
set FrUni/<fruni_inst> Dlci/<dlci> IpConn
dpToDeMapLevel <level>
```

Variable definitions

Variable	Value
<dlci>	is the highest DLCI value permitted for a permanent connection.
<frc>	is the name assigned to the FR connection.
<fruni_inst>	is the instance number of the FR UNI. The instance value must be unique.
<ip_address>	is the 32-bit address assigned to this interface.
<level>	is NotApplicable (default), high, medium or low.
<lpt_name>	is the name assigned to the software Lpt for this Lp.
<port>	is the port number.
<rtr_name>	is the name of the router.

Procedure job aid

Figure 28
Frame relay and feature set component hierarchy



Configuring EBGP

Configure External Border Gateway Protocol (EBGP) as the access protocol to exchange IP address prefixes between the CE router and the PE router. EBGP provides the benefit of having low routing overhead to support it and gives access to powerful policy control mechanisms.

Optionally, configure MD5 authentication on an EBGP peer connection to provide protection of EBGP neighbor relationships. Change an MD5 key on an authenticated EBGP peer using the procedure "Changing an MD5 key on an authenticated EBGP peer" (page 101).

Procedure steps

1 Add BGP on the VRF, and set the *localAs* and *BGPIdentifier* attributes.

```
add -s Rtr/<rtr_name> Vrf/<vrf_name> Bgp localAs
<as_value> bgpIdentifier <ip_address>
```

2 Add the EBGP peer over a VRF access interface.

```
add -s Rtr/<rtr_name> Vrf/<vrf_name> Bgp Peer/<Peer>
Desc peerAs <value>
```

3 Configure the *removePrivateAs* attribute to learn the IP address from the remote VPN site.

```
set Rtr/<router> Vrf/<vrf_name> Bgp Peer/<Peer> Desc
remPrivate enabled
```

4 Optionally, configure MD5 authentication on a BGP peer connection. Set the MD5 key value under the *Descriptor*.

```
Set Rtr/<rtr_name> Vrf/<vrf_name> Bgp Peer/<ip_addr>
Desc md5Key <ASCII_string>
```

5 Optionally, add the BGP export policy, setting the *protocol* attribute value to bgpMplsInternal.

```
add -s Rtr/<rtr_name> Vrf/<vrf_name> Bgp Export/
<export_policy_number> protocol bgpMplsInternal
```

6 Enable the multi-hop EBGP route distribution on a peer if you want to allow the distribution of routes across multiple hops between BGP peers that belong to different autonomous systems (AS).

```
set Rtr/<rtr_name> Vrf/<vrf_name> Bgp Peer/<Peer> Desc
multiHopEbgp <multihop_ebgp>
```

7 Set the *peerIpAddress* attribute.

```
set Rtr/<rtr_name> Vrf/<vrf_name> Bgp Export/
<export_policy_name> peerIpAddress <peer_ip_address>
```

8 Optionally, set the Multi Exit Discriminator (MED) value.

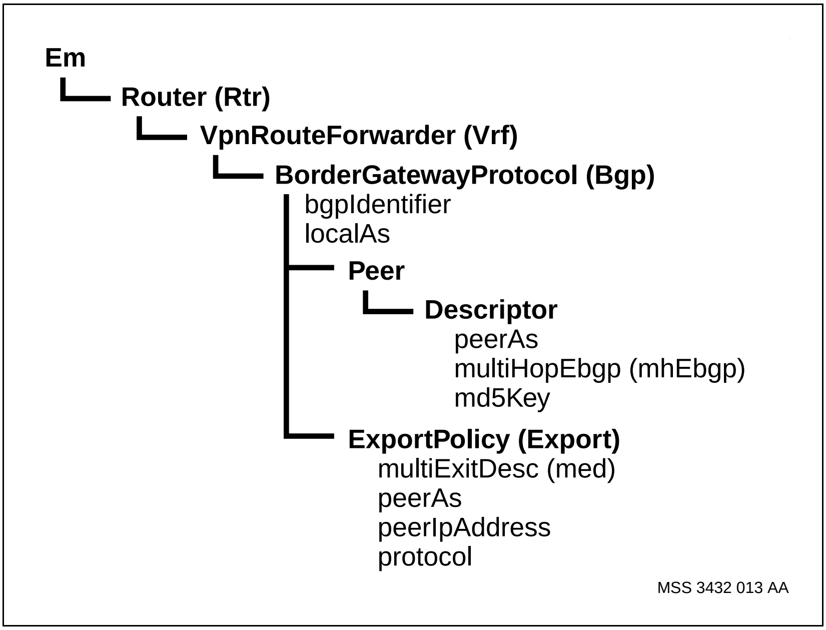
```
set Rtr/<rtr_name> Vrf/<vrf_name> Bgp Export/
<export_policy_name> med <med_value>
```

Variable definitions

Variable	Value
<ASCII_string>	is the MD5 key value, 1-255 ASCII characters in length.
<as_value>	is a value assigned to the Autonomous System. Note that the localAS value and the peerAS value must be different. The VRF AS and the router AS must be the same.
<export_policy_number>	is the numeric designation assigned to the export policy.
<ip_address>	is the IP address of the BGP Identifier.
<med_value>	is the value assigned to the multi exit discriminator.
<multihop_ebgp>	is identifying whether the multihop EBGP route distribution capability is disabled or enabled on a BGP peer.
<peer>	is the IP address of the BGP peer.
<peer_ip_address>	is the IP address of the peer.
<rtr_name>	is the name of the router.
<vrf_name>	is the name of the VPN route forwarder.

Procedure job aid

Figure 29
EBGP component hierarchy



Changing an MD5 key on an authenticated EBGP peer

Transition MD5 keys without having to terminate the BGP session for which the key is being changed, if you have optionally configured MD5 authentication on a BGP peer.

When changing keys on an BGP session between a Multiservice Switch node and a non-Multiservice Switch, change the key on the Multiservice Switch node to minimize the outage time.

A longer MD5 key will be more difficult to break, however, the longer the key used, the greater the impact on performance. A key length of between 12 and 24 characters should be sufficiently secure without having too great an impact on performance.

It is recommended that different MD5 key values be used for each BGP peer.

Procedure steps

1 Set the *keyTransitionDelay* attribute on the local and remote router, for all BGP peers.

```
Set Rtr/<rtr_name> Vrf/<vrf_name> Bgp
keyTransitionDelay <n>
```

2 Alternatively, set the *keyTransitionDelay* attribute on the local and remote router, for a specific peer's *Descriptor*.

```
set Rtr/<rtr_name> Vrf/<vrf_name> Bgp peer/<ip_addr>
Desc keyTransitionDelay <n>
```

3 Set the new MD5 key value on the local and router.

```
set Rtr/<rtr_name> Vrf/<vrf_name> Bgp peer/<ip_addr>
Desc md5Key <ASCII_string>
```

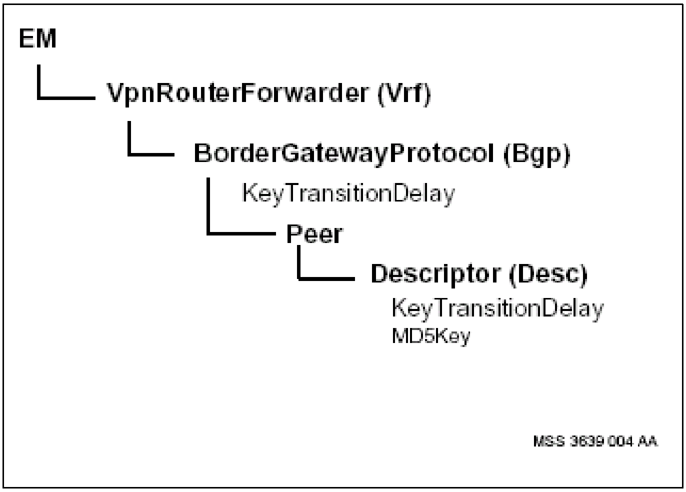
Variable definitions

Variable	Value
<ASCII_string>	is the MD5 key value, 1-255 ASCII characters in length.
<ip_addr>	is a 32-bit address assigned to the interface from which the routing information is derived.
(Sheet 1 of 2)	

Variable	Value
<n>	is the amount of time the operator has to provision both the local and remote ends with the new key, without losing service. for the *Bgp* component, the value is 1 to 20160 (minutes), default is 10 minutes. for the *Descriptor* component, the value is “sameAsBgp, 1 to 20160 (minutes)”, default is “sameAsBgp”.
<rtr_name>	is the name of the router.
(Sheet 2 of 2)	

Procedure job aid

Figure 30
Component hierarchy for changing an MD5 key on an authenticated BGP peer



Configuring BGP route flood protection

Optionally, configure the border gateway protocol (BGP) route flood protection to control the maximum number of IPv4 learned routes imported by all BGP sessions connected to a specific RFC2547 VPN route forwarder (VRF).

> ***Note:*** The variables *bgpImportRouteLimit* and *bgpImportRouteDiscardLimit* have default values of zero, meaning that there is no user provisioned value and BGP route flood protection is not enforced.

Procedure steps

1 Set the Bgp import alarm limit.

```
set Rtr/<rtr_name> Vrf/<vrf_name> Bgp

bgpImportRouteLimit <import_alarm_limit>
```

2 Set the Bgp import discard limit.

```
set Rtr/<rtr_name> Vrf/<vrf_name> Bgp

bgpImportRouteDiscardLimit <import_discard_limit>
```

Variable definitions

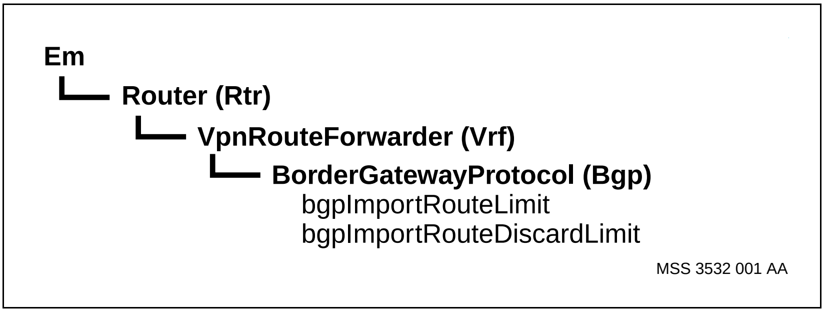
Variable	Value
<import_alarm_limit>	specifies a limit to control the maximum number of imported BGP Ipv4 Unicast routes that will be processed until an alarm is generated. This limit is chosen based on the requirements of various network configurations. The default value of zero implies that there is no limit set. In such a configuration, no alarm is generated and default behavior is expected. This attribute is critical under BGP.
<import_discard_limit>	specifies a limit to control the maximum number of imported BGP Ipv4 Unicast routes that will be processed until an alarm is generated and subsequent routes are discarded. This limit is chosen based on the requirements of various network configurations. The default value of zero implies that there is no limit set. In such a configuration, no alarm is generated and default behavior is expected. This attribute is critical under BGP.
(Sheet 1 of 2)	

Variable	Value
<rtr_name>	is the name of the router.
<vrf_name>	is the name of the VPN route forwarder.

(Sheet 2 of 2)

Procedure job aid

Figure 31
BGP route flood protection component hierarchy



Configuring BGP peer route flood protection

Optionally, configure the border gateway protocol (BGP) peer route flood protection to control the maximum number of IPv4 learned routes imported by a specific customer edge (CE) BGP session connected to an RFC2547 VPN route forwarder (VRF)..

> ***Note:*** The variables *bgpPeerRouteLimit* and *bgpPeerRouteDiscard* have default values of zero, meaning that there is no user provisioned value and BGP per peer route flood protection is not enforced.

Procedure steps

1 Set the Bgp peer import alarm limit.

```
set Rtr/<rtr_name> Vrf/<vrf_name> Bgp peer/
<peer_address> desc bgpPeerRouteLimit
<import_alarm_limit>
```

2 Set the Bgp peer import discard limit.

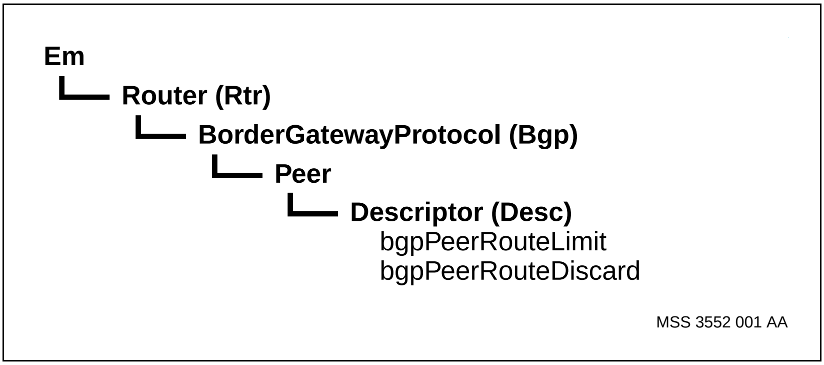
```
set Rtr/<rtr_name> Vrf/<vrf_name> Bgp peer/
<peer_address> desc bgpPeerRouteDiscard
<import_discard_limit>
```

Variable definitions

Variable	Value
<import_alarm_limit>	specifies a limit to control the maximum number of imported BGP Ipv4 Unicast routes that will be processed by one peer until an alarm is generated. This limit is chosen based on the requirements of various network configurations. This attribute is optional but critical under BGP peer.
<import_discard_limit>	specifies a limit to control the maximum number of imported BGP Ipv4 Unicast routes that will be processed by one peer until an alarm is generated and subsequent routes are discarded. This limit is chosen based on the requirements of various network configurations. This attribute is optional but critical under BGP peer.
<peer_address>	is the IP address of the peer.
<rtr_name>	is the name of the router.
<vrf_name>	is the name of the VPN route forwarder.

Procedure job aid

Figure 32
BGP peer route flood protection component hierarchy



Configuring OSPF

Configure Open Shortest Path First (OSPF) as the access protocol to exchange IP address prefixes between the CE router and the PE router. Although EBGP is the recommended access routing protocol, OSPF is also supported.

> ***Note:*** This procedure assumes a point-to-point link.

Procedure steps

1 Add the *Ospf* component, and set the *routerId*, *redistributeIbgp*, and *asBr* attribute values.

```
add -s Rtr/<rtr_name> Vrf/<vrf_name> Ospf routerId
<router_id>, redistributeIbgp true, asbr true
```

2 Add the OSPF area.

```
add -s Rtr/<rtr_name> Vrf/<vrf_name> Ospf Area/
<area_id>
```

3 Add the OSPF export policy and set the *protocol* attribute.

```
add -s Rtr/<rtr_name> Vrf/<vrf_name> Ospf Export/
<export_policy_number> protocol bgpMplsInternal
```

4 Add point-to-point OSPF on an existing interface.

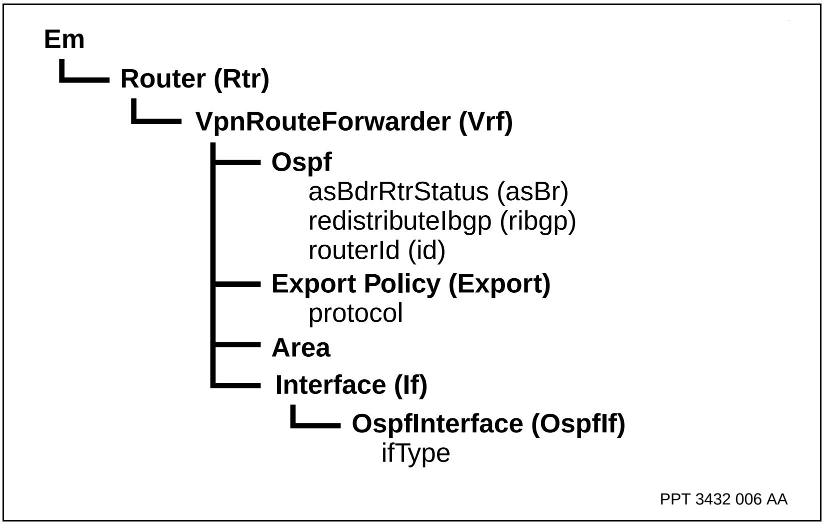
```
add Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
Ospfif ifType pointToPoint
```

Variable definitions

Variable	Value
<area_id>	is the 32-bit IP address for the OSPF area in which the router is located.
<export_policy_number>	is the numeric designation assigned to the export policy.
<router_id>	is the IP address that uniquely identifies the router in an autonomous system.
<rtr_name>	is the name of the router.
<vrf_name>	is the name of the VPN route forwarder.

Procedure job aid

Figure 33
OSPF component hierarchy



Configuring static routing

Configure static routing to allow the PE router to forward IP traffic to the CE router in cases where dynamic routing protocols are not being used between the CE and PE routers. This might also be done to address specific cases even when a dynamic routing protocol is in use.

Procedure steps

1 Add a static route.

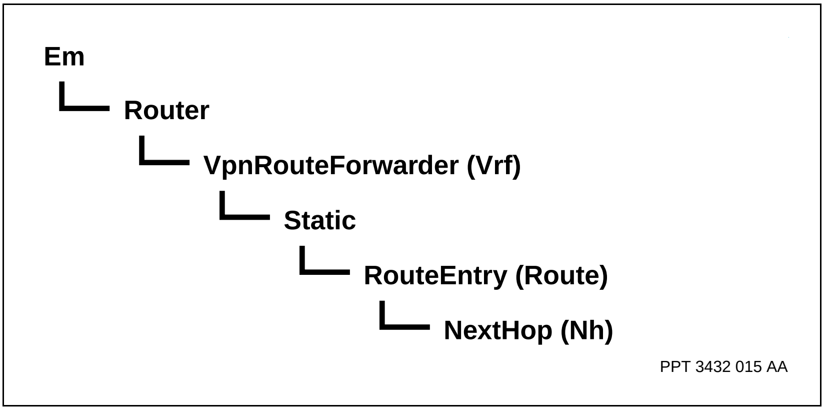
```
add -s Rtr/<rtr_name> Vrf/<vrf_name> Static
RouteEntry/<ip_address> Nexthop/<ip_address>
```

Variable definitions

Variable	Value
<ip_address>	is the next hop IP address of the static route. The next hop IP address must belong to a local subnet that exists on the VRF.
<rtr_name>	is the name of the router.
<vrf_name>	is the name of the VPN route forwarder.

Procedure job aid

Figure 34
Static routing component hierarchy



Configuring RIP

Configure Routing Information Protocol (RIP) to exchange the set of address prefixes between the CE router and the PE router. Although EBGP is the recommended access routing protocol, RIP is also supported.

Procedure steps

1 Add the *Rip* component.

```
add -s Rtr/<rtr_name> Vrf/<vrf_name> Rip
redistributeIbgp true
```

2 Add *RipIf* component under the router interface.

```
add -s Rtr/<rtr_name> Vrf/<vrf_name> If/<ip_address>
Ripif
```

3 Add the *Export* component and set the *protocol* attribute.

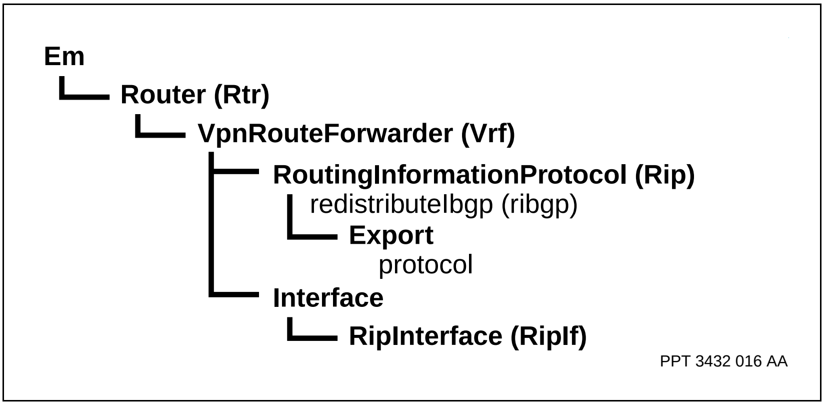
```
add -s Rtr/<rtr_name> vrf/<vrf_name> Rip Export/
<export_policy_number> protocol bgpMplsInternal
```

Variable definitions

Variable	Value
<export_policy_number>	is the numeric designation assigned to the export policy.
<ip_address>	is the ip address of the interface.
<rtr_name>	is the name of the router.
<vrf_name>	is the name of the VPN route forwarder.

Procedure job aid

Figure 35
RIP component hierarchy



PPT 3432 016 AA

Configuring the VRF export policy

Configure the VPN Route Forwarder (VRF) export policy to optionally control the redistribution of locally learned routes to remote PEs.

Procedure steps

1 Add the VRF export policy.

```
add Rtr/<rtr_name> Vrf/<vrf_name> Export/
<export_policy_number>
```

2 Set the protocol.

```
set Rtr/<rtr_name> Vrf/<vrf_name> Export/
<export_policy_number> protocol <protocol>
```

3 Set the local preference value.

```
set Rtr/<rtr_name> Vrf/<vrf_name> Export/
<export_policy_number> localPreference <pref_value>
```

4 Add the *Network* component, and set its prefix and length.

```
add -s Rtr/<rtr_name> Vrf/<vrf_name> Export/
<export_policy_number> Network/<net> prefix
<net_prefix>, length <net_length>
```

5 Set the export advertiseStatus mode.

```
set Rtr/<rtr_name> Vrf/<vrf_name> Export/
<export_policy_number> advertiseStatus <mode>
```

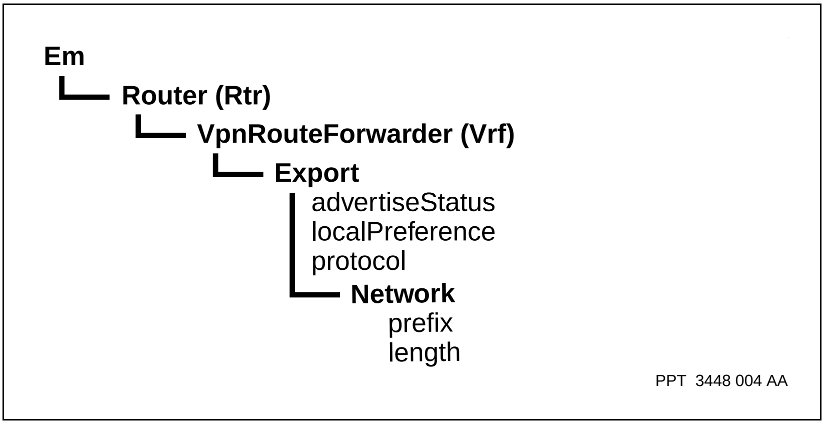
Variable definitions

Variable	Value
<export_policy_number>	is the numeric designation assigned to the export policy. Value = 0 - 65535
<mode>	is send or block. The default = send.
<net>	is the identifier assigned to network. The network prefix, under the VRF export policy, can be used to further refine a filtering policy or to leak a specific subnet from the local VRF to remote PEs when all the remaining local prefixes are being blocked by a more generic policy.
(Sheet 1 of 2)	

Variable	Value
<net_length>	is the value assigned to the network length. Values 1 - 32. Default = <system supplied>.
<net_prefix>	is the value assigned to the network IP prefix.
<pref_value>	is the value of localPreference. Values 0 - 4294967295. Default = 0.
<protocol>	is the policy applies to routes learned from the specified protocol. Values all, rip, ospfInternal, ospfExternal, staticLocal, staticRemote, bgpExternal. Default = all.
<rtr_name>	is the name of the router.
<vrf_name>	is the name of the VPN route forwarder.
(Sheet 2 of 2)	

Procedure job aid

Figure 36
VRF export policy component hierarchy



Configuring IP CPP on the VRF

Configure control plane protection (CPP) on the VRF to protect Nortel Networks Multiservice Switch nodes against certain denial of service (DoS) attacks on the control plane by monitoring the flow rate of IP packets destined for local IP destination addresses (DAs).

> ***Note:*** CPP is supported on PQC-based FPs only.

Prerequisites

- See the section on CPP in NN10600-800 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Technology Fundamentals*.
- The procedure in this section describes configuring CPP software and services only. Basic configuration at the node level (in this case, creating an instance of a logical processor type (LPT), and adding the *ip* and *ipCpp* services to the *featureList* attribute) must be performed first. Use the tasks and procedures in NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures* if you require supporting information or need to provision or reconfigure any node or nodal elements to support the CPP feature.

Procedure steps

1. Add a *Cpp* component as a subcomponent of the *Rtr* component.

   ```
   add Rtr/<rtr_name> vrf/<vrf_name> cpp
   ```

2. Configure the number of packets per second before discard occurs.

   ```
   set Rtr/<rtr_name> vrf/<vrf_name> cpp packetsPerSecond
   <packets_per_second>
   ```

3. Configure the isolation time period.

   ```
   set Rtr/<rtr_name> vrf/<vrf_name> cpp isolationTime
   <isolation_time>
   ```

4. Configure the grace period.

   ```
   set Rtr/<rtr_name> vrf/<vrf_name> cpp gracePeriod
   <grace_period>
   ```

5. Configure the CPP mode.

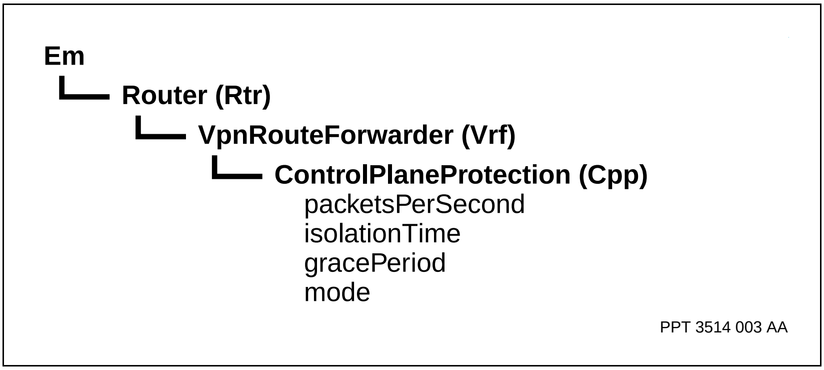
   ```
   set Rtr/<rtr_name> vrf/<vrf_name> cpp mode <cpp_mode>
   ```

Variable definitions

Variable	Value
<cpp_mode>	is the CPP operational mode: study, protect or disabled. Use study to determine an acceptable traffic rate. Once you learn the rate created and have the appropriate configuration, use protect. Use disabled to pre-configure the feature without enabling the monitoring process.
<grace_period>	is the period over which the average flow rate is measured to ensure that the exceeded traffic flow rate still exceeds the maximum allowed rate. A value of zero in protect mode means that isolation occurs immediately after an excessive flow is detected.
<isolation_time>	is the amount of time over which traffic will be discarded once isolation has begun. A value of zero indicates to permanently discard traffic until the card is cleared by an operator.
<packets_per_second>	is the flow rate, in packets per second, for the VR's DA, that must be exceeded on a single DA before discard processing occurs.
<rtr_name>	is the name of the router.
<vrf_name>	is the name of the VPN route forwarder.

Procedure job aid

Figure 37
VRF IP CPP component hierarchy



Configuring VRF route flood protection

Optionally, configure the VPN route forwarder (VRF) route flood protection to control the maximum number of IPv4 learned routes to be:

- exported from the VRF routing table to the router VPN routing information database
- imported into the VRF routing table from the router VPN routing information database

Procedure steps

1 Set the VRF import alarm limit.

```
set Rtr/<rtr_name> Vrf/<vrf_name>
importRoutesAlarmLimit <import_alarm_limit>
```

2 Set the VRF import discard limit.

```
set Rtr/<rtr_name> Vrf/<vrf_name>
importRoutesDiscardLimit <import_discard_limit>
```

3 Set the VRF export alarm limit.

```
set Rtr/<rtr_name> Vrf/<vrf_name>
exportRoutesAlarmLimit <export_alarm_limit>
```

4 Set the VRF export discard limit.

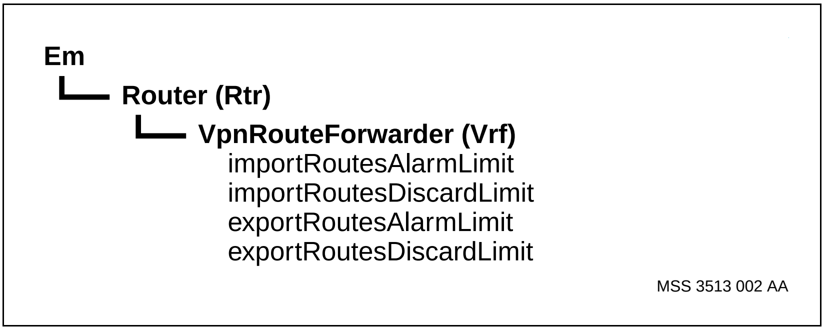
```
set Rtr/<rtr_name> Vrf/<vrf_name>
exportRoutesDiscardLimit <export_discard_limit>
```

Variable definitions

Variable	Value
<export_alarm_limit>	specifies the alarm threshold of number of routes to be exported from this VRF's routing table to the VPN routing database of the *Router* component. When the number of routes exported from this VRF reaches this limit, an alarm is generated and the VRF will continue exporting routes.
<export_discard_limit>	specifies the maximum number of routes to be exported from this VRF's routing table to the VPN routing database of the *Router* component. When the number of routes exported from this VRF reaches this limit, an alarm is generated and the VRF will stop exporting routes.
<import_alarm_limit>	specifies the alarm threshold of number of VPN routes to be imported to this VRF's routing table from the *Router* component. When the number of routes imported to this VRF reaches this limit, an alarm is generated and the VRF will continue importing routes.
<import_discard_limit>	specifies the maximum number of VPN routes to be imported to this VRF's routing table from the *Router* component. When the number of routes imported to this VRF reaches this limit, an alarm is generated and the VRF will stop importing routes.
<rtr_name>	is the name of the router.
<vrf_name>	is the name of the VPN route forwarder.

Procedure job aid

Figure 38
VRF route flood protection component hierarchy

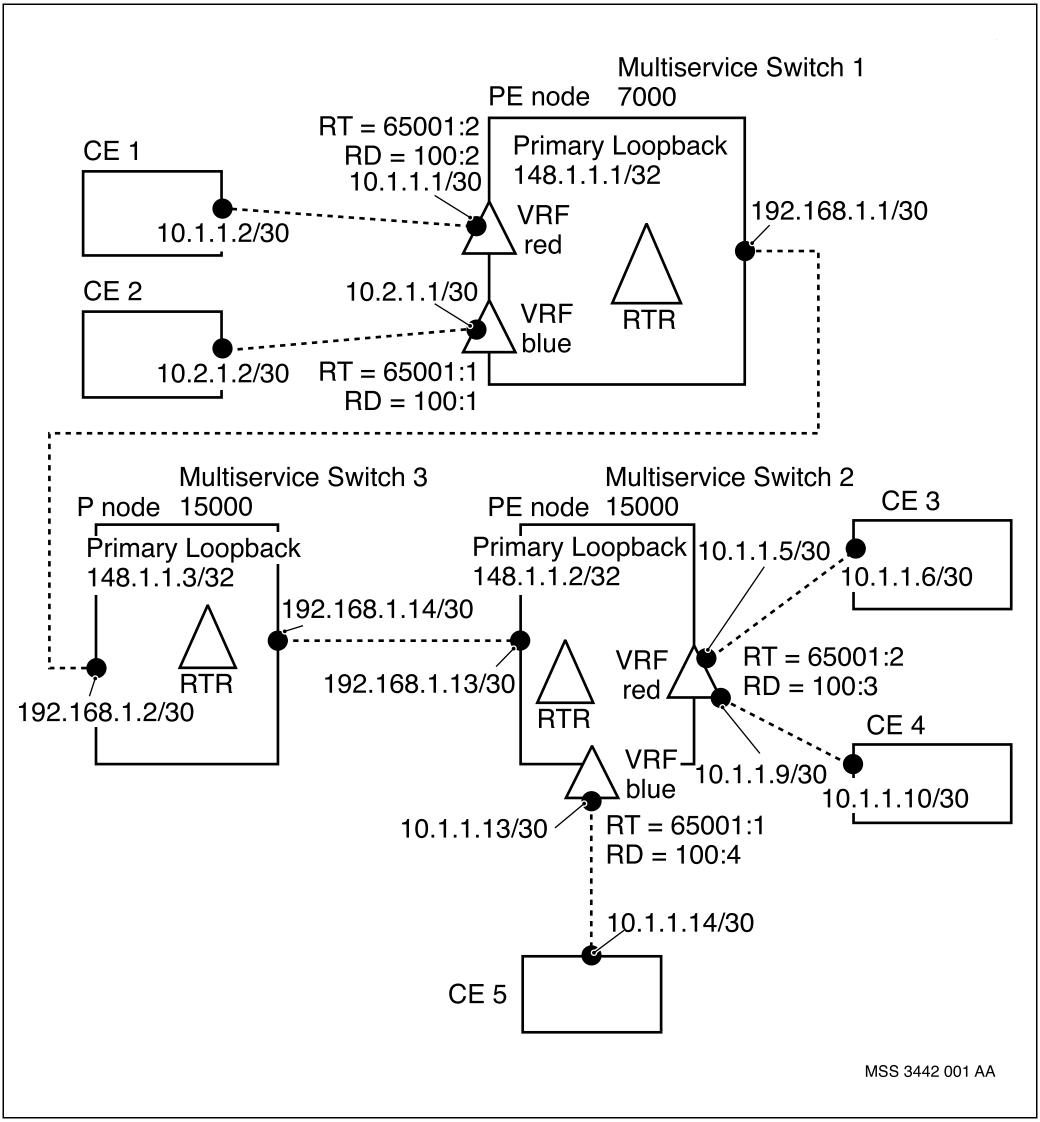


Chapter 7
BGP/MPLS VPN configuration example

This section provides examples of configuring BGP/MPLS VPN. The diagram “Example figure for a BGP/MPLS VPN network configuration” (page 120) represents the network being used in these examples. In these example procedures, the Layer 1 and Layer 2 provisioning steps (for example, AtmIf, La, FrUni, etc.) are not described.

> ***Note:*** The following are example procedures. The values you use in your configuration can differ from the values shown here. Consult your network engineer to ensure the values you are using are accurate for your configuration.

Figure 39
Example figure for a BGP/MPLS VPN network configuration



Example procedure for a backbone configuration for PE node Multiservice Switch 1 (Multiservice Switch 7400 node)

1 Configure RFC 2547 features.

```
set sw lpt/CP featureList ! OamEnet bgp ip

set sw lpt/VPNXC featureList ! rfc2547bis bgp ip

set sw lpt/atm_trunk featureList ! mplsLdp atmmpe ip
atmtrunks

check prov

activate prov

confirm prov
```

a. For ATM PQC cards:

```
add -s lp/<atm_lp> eng arc ov
MulticastBranchesCapacity <existing prov number +
4000>

add -s lp/<atm_lp> eng fcrc pqc

set lp/<atm_lp> eng fcrc pqc ov ipRoutesPoolCapacity
64000

add -s lp/<atm_lp> eng fcrc ov
subConnectionPoolCapacity 4096
```

2 Configure the Router and the Primary Loopback Interface.

```
add router/RTR1 VRP lp/<VpnXc_LP>

add router/RTR1 If/148.1.1.1 netmask 255.255.255.255

add router/RTR1 If/148.1.1.1 loopback mode
primaryLoopback
```

3 Configure AtmMpe Media for four emission priorities.

```
add router/RTR1 If/192.168.1.1 netmask 255.255.255.252

add router/RTR1 If/192.168.1.1 atmMpe

add atmif/<x> vcc/<n.m1> nep

set router/RTR1 If/192.168.1.1 atmMpe Ac/1
atmConnection AtmIf/<x> Vcc/<n.m1> nep
```

```
set AtmIf/<x> Vcc/<n.m1> Vcd Tm txPacketWiseDiscard
enabled
```

a. Add an ATM connection with DiffServ connection class 1 and link it to an ATM VCC that is configured with the ATM nrtVBR service category.

```
add router/RTR1 If/192.168.1.1 atmMpe Ac/2 ipCos 1

set router/RTR1 If/192.168.1.1 atmMpe Ac/2
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep

set AtmIf/<x> Vcc/<vpi.vci> vcd tm atmServiceCategory
nrtvbr

set AtmIf/<x> Vcc/<n.m> Vcd Tm txPacketWiseDiscard
enabled
```

b. Add an ATM connection with DiffServ connection class 2 and link it to an ATM VCC that is configured with the ATM rtVBR service category.

```
add router/RTR1 If/192.168.1.1 atmMpe Ac/3 ipCos 2

set router/RTR1 If/192.168.1.1 atmMpe Ac/3
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep

set AtmIf/<x> Vcc/<vpi.vci> vcd tm atmServiceCategory
rtvbr

set AtmIf/<x> Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

c. Add an ATM connection with DiffServ connection class 3 and link it to an ATM VCC that is configured with the ATM CBR service category.

```
add router/RTR1 If/192.168.1.1 atmMpe Ac/4 ipCos 3

set router/RTR1 If/192.168.1.1 atmMpe Ac/4
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep

set AtmIf/<x> Vcc/<vpi.vci> vcd tm atmServiceCategory
rtvbr

set AtmIf/<x> Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

4 Configure OSPF as IGP.

```
add -s router/RTR1 ospf area/0.0.0.0

set router/RTR1 ospf routerId 148.1.1.1
```

```
add router/RTR1 If/148.1.1.1 ospfif ifType passive
```

5 Configure OSPF Interfaces.

```
add router/RTR1 If/192.168.1.1 ospfif ifType
pointToPoint
```

6 If you want ISIS as IGP, complete the following commands and step 7:

```
add -s router/RTR1 isis Level/1

add router/RTR1 isis Net/47.0001.1480.0100.1001.00

add -s router/RTR1 If/148.1.1.1 isisif Level/1

set router/RTR1 If/148.1.1.1 isisif passive yes
```

7 Configure ISIS Interfaces.

```
add -s router/RTR1 If/192.168.1.1 isisif Level/1
```

8 Configure BGP.

```
add -s router/RTR1 Bgp localAs 1

set router/RTR1 Bgp bgpIdentifier 148.1.1.1

add -s router/RTR1 Bgp peer/148.1.1.2 desc peerAs 1

set router/RTR1 bgp peer/148.1.1.2 desc addressFamily
ipv4mplsvpn, lac 148.1.1.1
```

9 Configure LDP.

```
add router/RTR1 ldp

add router/RTR1 If/192.168.1.1 ldpif
```

10 Confirm configuration.

```
check prov

activate prov

confirm prov
```

Access Configuration for PE node Multiservice Switch 1 (Multiservice Switch 7400 node)

1 Configure RFC 2547 feature set on access.

```
set sw lpt/atm_access featureList ! rfc2547bis atmmpe
ip
```

```
check prov
```

```
activate prov
```

```
confirm prov
```

a. For ATM PQC cards:

```
add -s lp/<atm_lp> eng fcrc pqc
```

```
set lp/<atm_lp> eng fcrc pqc ov ipRoutesPoolCapacity
64000
```

2 Configure VRF.

```
add -s router/RTR1 vrf/RED rd/100:2
```

```
add router/RTR1 vrf/RED rt/65001:2
```

```
add -s router/RTR1 vrf/BLUE rd/100:1
```

```
add router/RTR1 vrf/BLUE rt/65001:1
```

3 Configure Interfaces on VRF.

```
add router/RTR1 vrf/RED If/10.1.1.1 netmask
255.255.255.252
```

```
add router/RTR1 vrf/BLUE If/10.2.1.1 netmask
255.255.255.252
```

4 Configure ATM access media for VPN RED.

```
add router/RTR1 vrf/RED If/10.1.1.1 atmMpe
```

a. Link the Ac/1 subcomponent to an ATM VCC that is configured with the ATM UBR service category.

```
set router/RTR1 vrf/RED if/10.1.1.1 atmMpe Ac/1
atmConnection AtmIf/x Vcc/<vpi.vci> nep
```

```
set AtmIf/x Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

b. Link the Ac/2 subcomponent to an ATM VCC that is configured with the ATM nrtVBR service category.

```
add router/RTR1 vrf/RED If/10.1.1.1 atmMpe Ac/2 ipCos 1
```

```
set router/RTR1 vrf/RED If/10.1.1.1 atmMpe Ac/2
atmConnection AtmIf/x Vcc/<vpi.vci> nep
```

```
set AtmIf/x Vcc/<vpi.vci> vcd tm atmServiceCategory
nrtvbr
```

```
set AtmIf/x Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

c. Link the Ac/3 subcomponent to an ATM VCC that is configured with the ATM rtVBR service category.

```
add router/RTR1 vrf/RED If/10.1.1.1 atmMpe Ac/3 ipCos 2
```

```
set router/RTR1 vrf/RED If/10.1.1.1 atmMpe Ac/3
atmConnection AtmIf/x Vcc/<vpi.vci> nep
```

```
set AtmIf/x Vcc/<vpi.vci> vcd tm atmServiceCategory
rtvbr
```

```
set AtmIf/x Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

d. Link the Ac/4 subcomponent to an ATM VCC that is configured with the ATM CBR service category.

```
add router/RTR1 vrf/RED If/10.1.1.1 atmMpe Ac/4 ipCos 3
```

```
set router/RTR1 vrf/RED If/10.1.1.1 atmMpe Ac/4
atmConnection AtmIf/x Vcc/n.m4 nep
```

```
set AtmIf/x Vcc/<vpi.vci> vcd tm atmServiceCategory
cbr
```

```
set AtmIf/x Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

5 Configure ATM access media for VPN BLUE. The same set of commands shown in step 4 is required to configure ATM media on interface/10.2.1.1 for VPN BLUE. Exchange each instance of vrf/RED with vrf/BLUE.

6 Configure routing protocols on VRF.

a. Add EBGP on the VRF RED media.

```
add -s router/RTR1 vrf/RED bgp localAs 1, bgpIdentifier
10.1.1.1
```

```
add -s router/RTR1 vrf/RED bgp peer/10.1.1.2 desc
peerAs 65101
```

```
add router/RTR1 vrf/RED bgp export/0 protocol
bgpMplsInternal
```

b. Add OSPF on the VRF BLUE media. This example assumes a point-to-point link.

```
add router/RTR1 vrf/BLUE ospf routerId 10.2.1.1, asbr
true, redistributeIbgp true
```

```
add router/RTR1 vrf/BLUE ospf area/0.0.0.0
```

```
add router/RTR1 vrf/BLUE ospf export/0 protocol
bgpMplsInternal
```

```
add router/RTR1 vrf/BLUE If/10.2.1.1 ospfif ifType
pointToPoint
```

7 Confirm configuration.

```
check prov
```

```
activate prov
```

```
confirm prov
```

Example prcoedure for a backbone configuration for PE node Multiservice Switch 2 (Multiservice Switch 15000 node)

1 Configure RFC 2547 features.

```
set sw lpt/CP featureList ! OamEnet bgp ip
```

```
set sw lpt/VPNXC featureList ! rfc2547bis bgp ip atmmpe
```

```
set sw lpt/gige_trunk featureList ! mplsLdp ip
```

```
check prov
```

```
activate prov
```

```
confirm prov
```

2 Configure the Router and the Primary Loopback Interface.

```
add router/RTR2 vrp lp/<VpnXc>
```

```
add router/RTR2 If/148.1.1.2 netmask 255.255.255.255
```

```
add router/RTR2 If/148.1.1.2 loopback mode
primaryLoopback
```

3 Configure GigE media.

```
add router/RTR2 If/192.168.1.13 netmask
255.255.255.252
```

```
add router/RTR2 If/192.168.1.13 enet linkToEnetIf La/n
```

4 Configure OSPF as an IGP.

```
add -s router/RTR2 ospf area/0.0.0.0
```

```
set router/RTR2 ospf routerId 148.1.1.2
```

```
add router/RTR2 If/148.1.1.2 ospfif ifType passive
```

5 Configure OSPF Interfaces.

```
add router/RTR2 If/192.168.1.13 ospfif ifType
pointToPoint
```

6 If you want ISIS as IGP, complete the following commands and step 7:

```
add -s router/RTR2 isis Level/1
```

```
add router/RTR2 isis Net/47.0001.1480.0100.1002.00
add -s router/RTR2 If/148.1.1.2 isisIf Level/1
```

```
set router/RTR2 If/148.1.1.2 isisIf passive yes
```

7 Configure ISIS Interfaces.

```
add -s router/RTR2 If/192.168.1.13 isisif Level/1
```

8 Configure BGP.

```
add -s router/RTR2 Bgp localAs 1
```

```
set router/RTR2 bgp bgpIdentifier 148.1.1.2
```

```
add -s router/RTR2 bgp peer/148.1.1.1 desc peerAs 1
```

```
set router/RTR2 bgp peer/148.1.1.1 desc addressFamily
ipv4mplsvpn, lac 148.1.1.2
```

9 Configure LDP.

```
add router/RTR2 ldp
```

```
add router/RTR2 If/192.168.1.13 ldpif
```

10 Confirm configuration.

```
check prov
```

```
activate prov
```

```
confirm prov
```

Access Configuration for PE node Multiservice Switch 2 (Multiservice Switch 15000 node)

1 Configure RFC 2547 feature set on access.

```
set sw lpt/atm_access featureList ! rfc2547bis atmmpe
ip
```

```
set sw lpt/frameRelay_access featureList ! rfc2547bis
frameRelayUni frUniIpOptimized ip
```

```
check prov
```

```
activate prov
```

```
confirm prov
```

a. For ATM PQC cards:

```
add -s lp/<atm_lp> eng fcrc pqc
```

```
set lp/<atm_lp> eng fcrc pqc ov ipRoutesPoolCapacity
64000
```

2 Configure VRF.

```
add -s router/RTR2 vrf/RED rd/100:3
```

```
add router/RTR2 vrf/RED rt/65001:2
```

```
add -s router/RTR2 vrf/BLUE rd/100:4
```

```
add router/RTR2 vrf/BLUE rt/65001:1
```

3 Configure Interfaces.

```
add router/RTR2 vrf/RED If/10.1.1.5 netmask
255.255.255.252
```

```
add router/RTR2 vrf/RED If/10.1.1.9 netmask
255.255.255.252
```

```
add router/RTR2 vrf/BLUE If/10.2.1.13 netmask
255.255.255.252
```

4 Configure ATM access media for VPN RED.

```
add router/RTR2 vrf/RED If/10.1.1.5 atmMpe
```

a. Link the Ac/1 subcomponent to an ATM VCC that is configured with the ATM UBR service category.

```
set router/RTR2 vrf/RED if/10.1.1.5 atmMpe Ac/1
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep
```

```
set AtmIf/<x> Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

b. Link the Ac/2 subcomponent to an ATM VCC that is configured with the ATM nrtVBR service category.

```
add router/RTR2 vrf/RED If/10.1.1.5 atmMpe Ac/2 ipCos 1
```

```
set router/RTR2 vrf/RED If/10.1.1.5 atmMpe Ac/2
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep
```

```
set AtmIf/<x> Vcc/<vpi.vci> vcd tm atmServiceCategory
nrtvbr
```

```
set AtmIf/<x> Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

c. Link the Ac/3 subcomponent to an ATM VCC that is configured with the ATM rtVBR service category.

```
add router/RTR2 vrf/RED If/10.1.1.5 atmMpe Ac/3 ipCos 2
```

```
set router/RTR2 vrf/RED If/10.1.1.5 atmMpe Ac/3
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep
```

```
set AtmIf/<x> Vcc/<vpi.vci> vcd tm atmServiceCategory
rtvbr
```

```
set AtmIf/<x> Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

d. Link the Ac/4 subcomponent to an ATM VCC that is configured with the ATM CBR service category.

```
add router/RTR2 vrf/RED If/10.1.1.5 atmMpe Ac/4 ipCos 3
```

```
set router/RTR2 vrf/RED If/10.1.1.5 atmMpe Ac/4
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep
```

```
set AtmIf/<x> Vcc/<vpi.vci> vcd tm atmServiceCategory
cbr
```

```
set AtmIf/<x> Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

5 Configure second ATM access media for VPN RED. The same set of commands shown in step 4 is required to configure ATM media on interface/10.1.1.9 for VPN RED.

6 Configure Frame relay access media for VPN BLUE.

```
add router/RTR2 vrf/BLUE If/10.2.1.13 IpoDlci
```

```
set Fruni/x numberOfEmissionsQs 4
```

```
set FrUni/<fruni_inst> Dlci/<dlci> IpConn
dpToDeMapLevel/<level>
```

```
set router/RTR2 vrf/BLUE If/10.2.1.13 IpoDlci Frc/1
linkToFrDlci Fruni/<x> Dlci/<dlci> IpConnection
```

a. Add a frame relay connection for DiffServ connection class 1 and link it to another Dlci subcomponent of the same Fruni component. The Dlci for Frc/2 is configured for better QoS in the FR network than Frc/1.

```
add router/RTR2 vrf/BLUE If/10.2.1.13 IpoDlci Frc/2
ipCos 1
```

```
set router/RTR2 vrf/BLUE If/10.2.1.13 IpoDlci Frc/2
linkToFrDlci Fruni/<x> Dlci/<dlci> IpConnection
```

b. Add a frame relay connection for DiffServ connection class 2 and link it to another Dlci subcomponent of the same Fruni component. The Dlci for Frc/3 is configured for better QoS in the FR network than Frc/2.

```
add router/RTR2 vrf/BLUE If/10.2.1.13 IpoDlci Frc/3
ipCos 2
```

```
add router/RTR2 vrf/BLUE If/10.2.1.13 IpoDlci Frc/3
linkToFrDlci Fruni/<x> Dlci/<dlci> IpConnection
```

c. Add a frame relay connection for DiffServ connection class 3 and link it to another Dlci subcomponent of the same Fruni component. The Dlci for Frc/4 is configured for better QoS in the FR network than Frc/3.

```
add router/RTR2 vrf/BLUE If/10.2.1.13 IpoDlci Frc/4
ipCos 3
```

```
set router/RTR2 vrf/BLUE If/10.2.1.13 IpoDlci Frc/4
linkToFrDlci Fruni/<x> Dlci/<dlci> IpConnection
```

7 Configure access Routing protocols.

a. Add static routes for interfaces 10.1.1.5 and 10.1.1.9.

```
add -s router/RTR2 vrf/RED static route/N1 nexthop/
10.1.1.6
```

```
add -s router/RTR2 vrf/RED static route/N2 nexthop/
10.1.1.10
```

b. Add RIP for interface 10.2.1.13.

```
add -s router/RTR2 vrf/BLUE rip redistributeIbgp true
```

```
add -s router/RTR2 vrf/BLUE If/10.2.1.13 ripif
```

```
add -s router/RTR2 vrf/BLUE rip export/0 protocol
bgpMplsInternal
```

8 Confirm configuration.

```
check prov
```

```
activate prov
```

```
confirm prov
```

Example procedure for configuring a P node on a Multiservice Switch 3 (Multiservice Switch 15000 node)

1 Configure RFC 2547 features.

```
set sw lpt/CP featureList ! OamEnet
```

```
set sw lpt/atm_trunk featureList ! mplsLdp ip atmmpe
atmTrunks
```

```
check prov
```

```
activate prov
```

```
confirm prov
```

```
add -s lp/<atm_lp> eng arc ov
MulticastBranchesCapacity <existing prov number + 4K>
```

```
add -s lp/<atm_lp> eng fcrc pqc ov ipRoutesPoolCapacity
64000
```

```
add -s lp/<atm_lp> eng fcrc ov
subConnectionPoolCapacity 4096
```

```
set sw lpt/GigE_trunk featureList ! mplsLdp ip
```

2 Configure the Router and Primary Loopback Interface.

```
add router/RTR3 VRP lp/<VpnXc_LP>
```

```
add router/RTR3 Interface/148.1.1.3 netmask
255.255.255.255
```

```
add router/RTR3 If/148.1.1.3 loopback mode
primaryLoopback
```

3 Configure Interfaces.

```
add router/RTR3 If/192.168.1.2 netmask 255.255.255.252
```

```
add router/RTR3 If/192.168.1.14 netmask
255.255.255.252
```

4 Configure OSPF as IGP.

```
add -s router/RTR3 ospf area/0.0.0.0
```

```
set router/RTR3 ospf routerId 148.1.1.3
```

```
add router/RTR3 If/148.1.1.3 ospfif ifType passive
```

5 Configure OSPF Interfaces.

```
add router/RTR3 If/192.168.1.2 ospfif ifType
pointToPoint
```

```
add router/RTR3 If/192.168.1.14 ospfif ifType
pointToPoint
```

6 If you want ISIS as IGP, complete the following commands and step 7:

```
add -s router/RTR3 isis Level/1
```

```
add router/RTR3 isis Net/47.0001.1480.0100.1003.00
```

```
add -s router/RTR3 If/148.1.1.3 isisif Level/1
```

```
set router/RTR3 If/148.1.1.3 isisif passive yes
```

7 Configure ISIS Interfaces.

```
add -s router/RTR3 If/192.168.1.2 isisif Level/1
```

```
add -s router/RTR3 If/192.168.1.14 isisif Level/1
```

8 Configure LDP.

```
add router/RTR3 ldp
```

```
add router/RTR3 If/192.168.1.2 ldpif
```

```
add router/RTR3 If/192.168.1.14 ldpif
```

9 Configure ATM media for four emission priorities.

```
add router/RTR3 If/192.168.1.2 atmMpe
```

```
set router/RTR3 If/192.168.1.2 atmMpe Ac/1
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep
```

```
set AtmIf/x Vcc/n.m vcd tm atmServiceCategory ubr
```

```
set AtmIf/<x> Vcc/<n.m> Vcd Tm txPacketWiseDiscard
enabled
```

a. Add an ATM connection with DiffServ connection class 1 and link it to an ATM VCC that is configured with the ATM nrtVBR service category.

```
add router/RTR3 If/192.168.1.2 atmMpe Ac/2 ipCos 1
```

```
set router/RTR3 If/192.168.1.2 atmMpe Ac/2
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep
```

```
set AtmIf/<x> Vcc/<vpi.vci> vcd tm atmServiceCategory
nrtvbr
```

```
set AtmIf/<x> Vcc/<n.m> Vcd Tm txPacketWiseDiscard
enabled
```

b. Add an ATM connection with DiffServ connection class 2 and link it to an ATM VCC that is configured with the ATM rtVBR service category.

```
add router/RTR3 If/192.168.1.2 atmMpe Ac/3 ipCos 2
```

```
set router/RTR3 If/192.168.1.2 atmMpe Ac/3
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep
```

```
set AtmIf/<x> Vcc/<vpi.vci> vcd tm atmServiceCategory
rtvbr
```

```
set AtmIf/<x> Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

c. Add an ATM connection with DiffServ connection class 3 and link it to an ATM VCC that is configured with the ATM CBR service category.

```
add router/RTR3 If/192.168.1.2 atmMpe Ac/4 ipCos 3
```

```
set router/RTR3 If/192.168.1.2 atmMpe Ac/4
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep
```

```
set AtmIf/<x> Vcc/<vpi.vci> vcd tm atmServiceCategory
rtvbr
```

```
set AtmIf/<x> Vcc/<vpi.vci> Vcd Tm txPacketWiseDiscard
enabled
```

10 Configure GigE media.

```
add -s router/RTR3 If/192.168.1.14 enet linkToEnetIf
La/n
```

11 Confirm configuration.

```
check prov
```

```
activate prov
```

```
confirm prov
```

Example procedure for configuring a BGP MED policy

This is an example procedure for configuring a BGP multi exit discriminator (MED) policy on the PE nodes such that the access link between CE4 and Multiservice Switch 2 is preferred over the access link between CE4 and Multiservice Switch 1. For redundancy purposes, CE4 is connected to VRF RED on both Multiservice Switch 1 and Multiservice Switch 2. The diagram “Example figure for an export policy network configuration” (page 138) represents the network being used in the following example.

1 Configure the eBGP peers between CE and PE on Multiservice Switch 2.

```
add -s router/RTR2 vrf/RED bgp localAs 1
```

```
add -s router/RTR2 vrf/RED bgp peer/10.1.1.10 desc
peerAs 65001
```

2 Add the BGP export policy on Multiservice Switch 2 and set the MED value lower than that of Multiservice Switch 1.

```
add -s router/RTR2 vrf/RED bgp export/0 protocol
bgpMplsInternal, med 100
```

```
set router/RTR2 vrf/RED bgp export/0 peerIpAddress
10.1.1.10
```

Note: A lower MED value is preferable to a higher MED value.

3 For redundancy purposes, add another Interface with AtmMpe media on Vrf RED on Multiservice Switch 1.

```
add -s router/RTR1 vrf/RED if/10.1.1.13 netmask
255.255.255.252
```

```
add -s router/RTR1 vrf/RED if/10.1.1.13 atmmpe
atmConnection AtmIf/<x> Vcc/<vpi.vci> nep
```

4 Configure the peers between CE to PE links on Multiservice Switch 1.

```
add -s router/RTR1 vrf/RED bgp localAs 1
```

```
add -s router/RTR1 vrf/RED bgp peer/10.1.1.14 desc
peerAs 65001
```

5 Add the BGP export policy on Multiservice Switch 1 and set the MED value higher than that of Multiservice Switch 2.

```
add -s router/RTR1 vrf/RED bgp export/0 protocol
bgpMplsInternal, med 200
```

```
set router/RTR1 vrf/RED bgp export/0 peerIpAddress
10.1.1.14
```

6 Confirm configuration.

```
check prov
```

```
activate prov
```

```
confirm prov
```

Note: For routes learned across the MPLS backbone, CE4 will prefer the ones learned via Multiservice Switch 2 because the MED attached to these routes is lower. If Multiservice Switch 2 stops advertising these routes, or the eBGP peer fails, routes learned from Multiservice Switch 1 will be used.

Example procedure for configuring a VRF policy

This is an example procedure for configuring a VRF export policy to advertise a local preference for routes received from CE via eBGP. The diagram “Example figure for an export policy network configuration” (page 138) represents the network being used in the following example.

1 Set the protocol and local preference.

```
add -s router/RTR1 vrf/RED export/1 protocol ebgp,
localPreference 140, mode send
```

2 Add the *Network* component and set its prefix and length.

```
add s- router/RTR1 vrf/RED export/1 network/1 prefix
<ip_address X>, length 24
```

3 Confirm the configuration.

```
check prov

activate prov

confirm prov
```

4 Set the protocol and local preference.

```
add -s router/RTR2 vrf/RED export/1 protocol ebgp,
localPreference 200, mode send
```

5 Add the *Network* component and set its prefix and length.

```
add -s router/RTR2 vrf/RED export/1 network/1 prefix
<ip_address X>, length 24
```

6 Confirm the configuration.

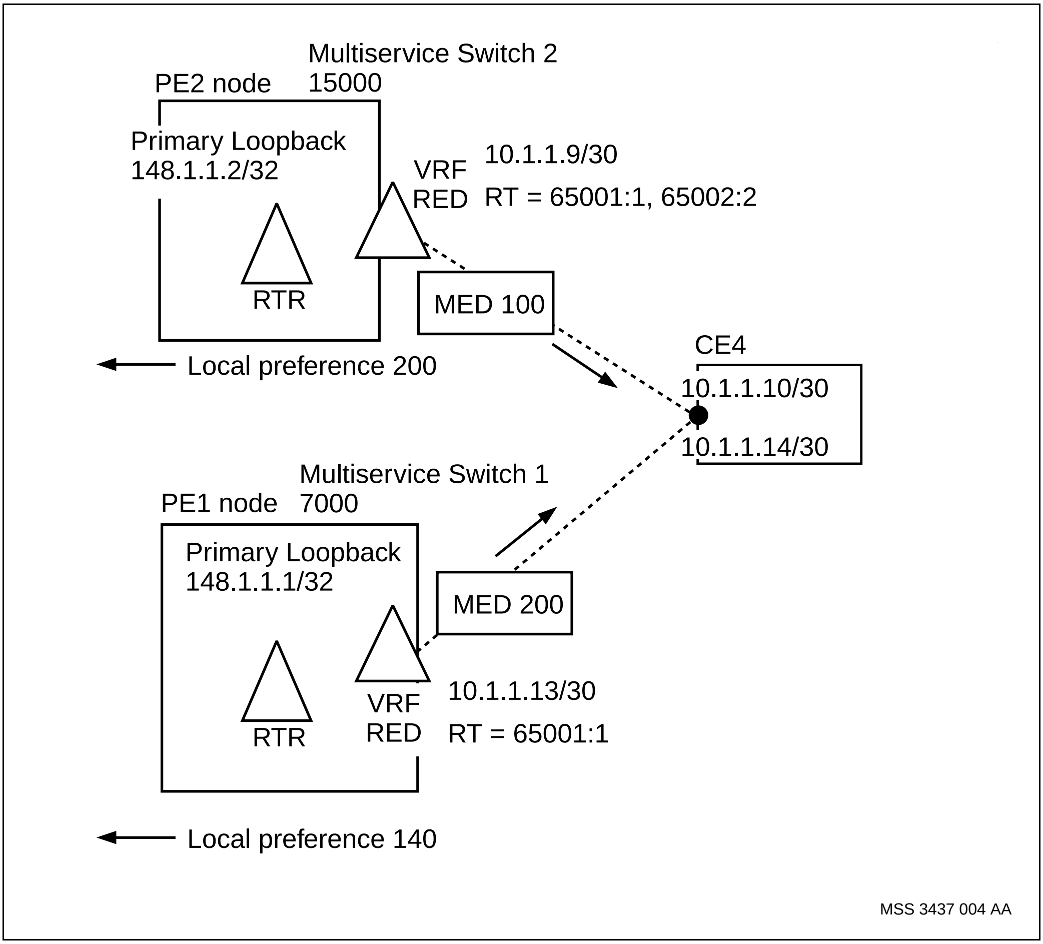
```
check prov

activate prov

confirm prov
```

Note: Remote VRFs of VPN RED receive route X from both PE1 and PE2; they will prefer PE2 (higher local preference = 200). If route X stops being advertised from VRF RED on PE2 but it is still advertised from VRF RED on PE1 (local preference = 140), traffic destined to X is redirected to PE1.

Figure 40
Example figure for an export policy network configuration



Example procedure for configuring a route target export policy

This is an example procedure for configuring a route target export policy on the PE node to block VPN IPv4 routes based on route targets to be advertised to a specific peer. The diagram “Example figure for an export policy network configuration” (page 138) represents the network being used in the following example.

1 Add the BGP export policy on the Multiservice Switch 2 and set the *addressFamily* attribute to match the VPN IPv4 addresses.

```
add Rtr/RTR2 Bgp Export/0 af ipv4MplsVpn
```

2 Set the route targets.

```
set Rtr/RTR2 Bgp Export/0 rt 65001:1, 65002:2
```

3 Specify the *peerIpAddress* attribute to which the policy applies to.

```
set Rtr/RTR2 Bgp Export/0 peerIpAddress x.x.x.x
```

4 Specify the action to take on the policy match by setting the *advertiseStatus* attribute.

```
set Rtr/RTR2 Bgp Export/0 advertiseStatus block
```

Chapter 8
VCG-based IP VPN configuration

Configure a VCG-based IP VPN to:

- aggregate different customer networks over a common backbone connection
- provide a customer site access to a VPN

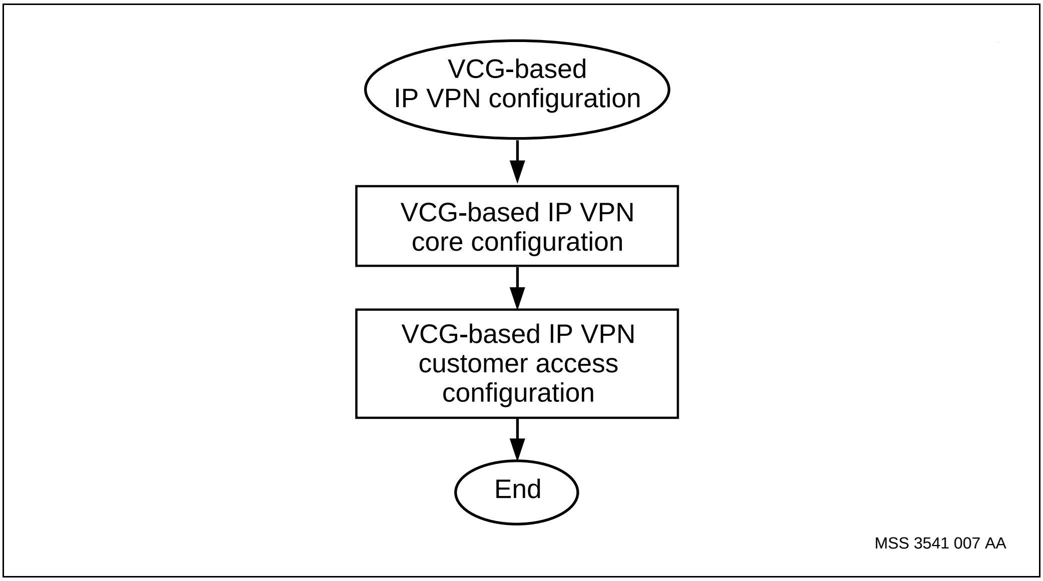
Prerequisites to VCG-based IP VPN configuration

- Nortel Networks Multiservice Switch IP services must be installed as described in NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.

VCG-based IP VPN tasks

This work flow shows you the sequence of tasks you perform to configure a VCG-based IP VPN. To link to any procedure, go to “VCG-based IP VPN configuration task navigation” (page 142).

Figure 41
VCG-based IP VPN configuration tasks



VCG-based IP VPN configuration task navigation

- “VCG-based IP VPN core configuration” (page 143)
- “VCG-based IP VPN customer access configuration” (page 159)

Chapter 9
VCG-based IP VPN core configuration

Configure a VCG-based IP VPN core configuration to aggregate different customer networks over a common backbone connection.

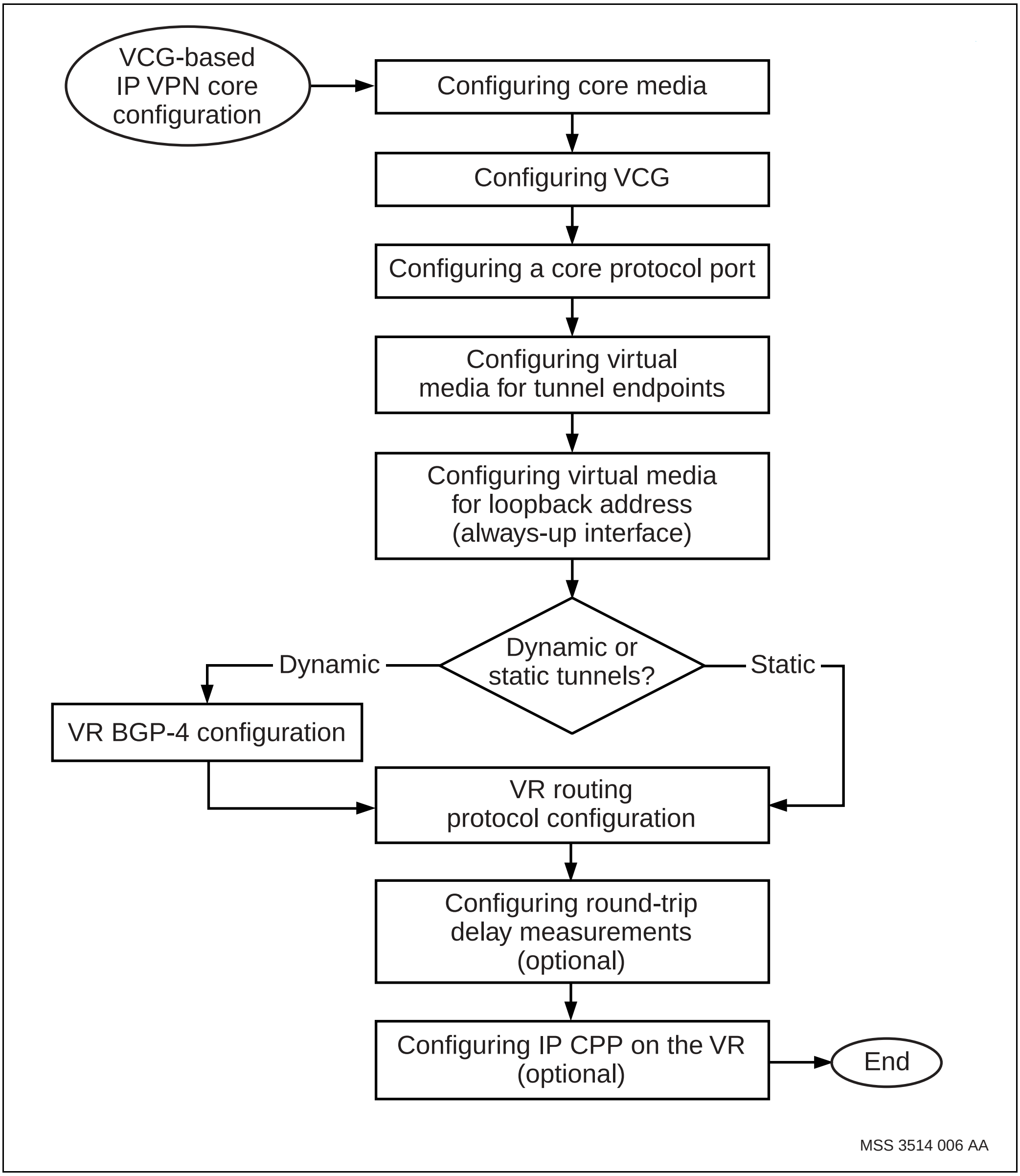
Prerequisites to VCG-based IP VPN core configuration

- Nortel Networks Multiservice Switch IP services must be installed as described in NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.
- The task flow and procedures in this section describe configuring VCG-based IP VPN core configuration services only. Basic configuration (in this case, creating an instance of a logical processor type (LPT), and adding the services to the *featureList* component) must be performed first. Use the tasks and procedures in NN10600-060 *Nortel Networks Multiservice Switch 7400/15000/20000 Component Reference* if you require supporting information or need to provision or reconfigure any node or nodal elements to support VCG-based IP VPN features.
  - For the CP card, include the mvr bgp ip software feature in the featurelist.
  - For the ATM card, include the atmmpe ip software feature in the featurelist.

VCG-based IP VPN core configuration procedures

This task flow shows you the sequence of procedures you perform to configure a VCG-based IP VPN core configuration. To link to any procedure go to “VCG-based IP VPN core configuration procedure navigation” (page 145).

Figure 42
VCG-based IP VPN core configuration procedures


VCG-based IP VPN core configuration
Configuring core media
Configuring VCG
Configuring a core protocol port
Configuring virtual media for tunnel endpoints
Configuring virtual media for loopback address (always-up interface)
Dynamic or static tunnels?
Dynamic
Static
VR BGP-4 configuration
VR routing protocol configuration
Configuring round-trip delay measurements (optional)
Configuring IP CPP on the VR (optional)
End
MSS 3514 006 AA

VCG-based IP VPN core configuration procedure navigation

- “Configuring core media” (page 146)
- “Configuring VCG” (page 148)
- “Configuring a core protocol port” (page 149)
- “Configuring virtual media for tunnel endpoints” (page 151)
- Configuring virtual media for loopback address, always-up interface. See NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.
- VR BGP-4 configuration. See NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.
- VR routing protocol configuration. See NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.
- “Configuring round-trip delay measurements” (page 153)
- Configuring IP CPP on the VR. See NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.

Configuring core media

Configure core media to build the VPN backbone to interconnect the VCGs.

Prerequisites

- You must configure an AtmIf prior to configuring core media. See the Configuring an AtmIf procedure in NN10600-710 *Nortel Networks Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

Procedure steps

1 Add the *VirtualChannelConnection* and *NailedUpEndPoint* subcomponents to the *AtmInterface* component.

```
add AtmIf/<atmif_value> Vcc/<vpi.vci> Nep
```

2 Add the *AtmMultiprotocolEncapsulation* component.

```
add AtmMpe/<atmmpe_value>
```

3 Link the *AtmMpe Ac* components to the *AtmIf Vcc* components.

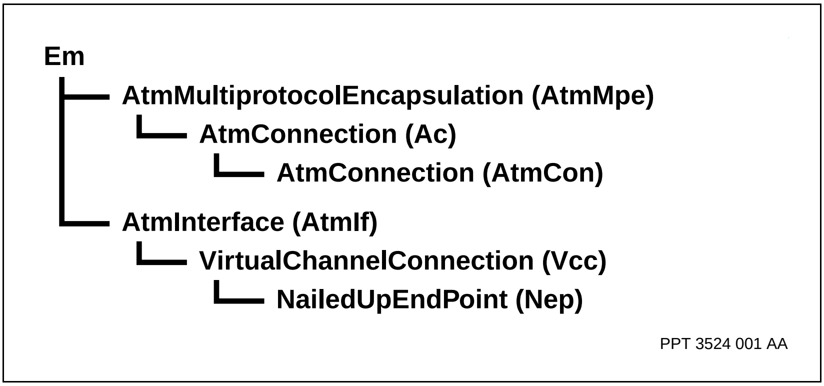
```
set AtmMpe/<atmmpe_value> Ac/<ac_id> AtmCon AtmIf/
<atmif_value> Vcc/<vpi.vci> Nep
```

Variable definitions

Variable	Value
<ac_id>	is the instance value of the atm connection. Any mnemonic (for example, AC/1).
<atmif_value>	is the instance value of the *AtmIf* component.
<atmmpe_value>	is the logical interface between the virtual router and ATM media (*AtmIf* component).
<vpi.vci>	is the identifier assigned to the VP and VC of the virtual channel. It is a good practice to assign the vci value of the atm connection to the ac_id (example: set Router/<rtr_name> Vrf/<vrf_name> If/<ip_address> AtmMpe Ac/40 atmConnection AtmIf/<n> Vcc/0.40 Nep. It is the ac_id that is displayed in the ARP table for the IP address mapping to the layer 2 attribute (inverse ARP).

Procedure job aid

Figure 43
Core media component hierarchy



PPT 3524 001 AA

Configuring VCG

Configure a virtual connection gateway (VCG) to connect customer sites through a VPN.

Procedure steps

1 Add a virtual router (VR).

```
add Vr/<vr_name>
```

2 Set the logical processor (LP) on which the virtual router processor resides.

```
set Vr/vr_name> vrp Lp/<lp_value>
```

3 Set the *vpnMode* attribute to carrier to define this virtual router as a VCG.

```
set Vr/<vr_name> vpnm carrier
```

4 Add the IP protocol to the VR.

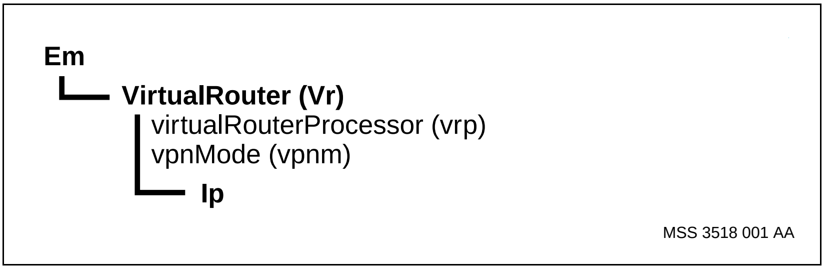
```
add Vr/<vr_name> ip
```

Variable definitions

Variable	Value
<lp_value>	is the value of the logical processor.
<vr_name>	is the name of the virtual router.

Procedure job aid

Figure 44
VCG component hierarchy



Configuring a core protocol port

Configure a core protocol port to link the physical backbone interface to the VCG virtual router.

Procedure steps

1. Add a core protocol port.

   ```
   add Vr/<vr_name> Pp/<pp_value>
   ```

2. Link the protocol port to the media application.

   ```
   set Vr/<vr_name> Pp/<pp_value> media AtmMpe/
   <atmmpe_value>
   ```

3. Add the *IpPort* component which contains IP attributes specific to this port.

   ```
   add Vr/<vr_name> Pp/<pp_value> IpPort
   ```

4. Add the *IpLogicalInterface* component.

   ```
   add Vr/<vr_name> Pp/<pp_value> IpPort LogicalIf/
   <ipport_address>
   ```

5. Set the subnet mask for this logical interface.

   ```
   set Vr/<vr_name> Pp/<pp_value> IpPort LogicalIf/
   <ipport_address> netMask <netmask_address>
   ```

6. Set the broadcast address for this logical interface.

   ```
   set Vr<vr_name> Pp/<pp_value> IpPort LogicalIf/
   <ipport_address> broadcastAddress <broadcast_address>
   ```

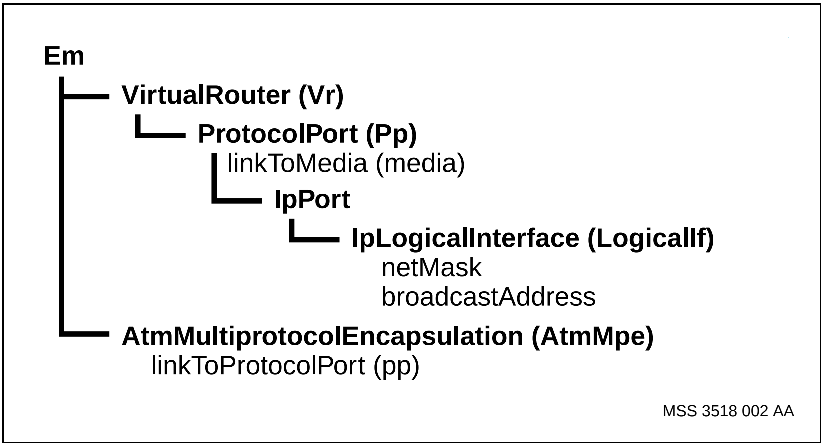
Variable definitions

Variable	Value
<atmmpe_value>	is the logical interface between the virtual router and ATM media (*AtmIf* component).
<broadcast_address>	is the IP address of the broadcast address for this *LogicalIf* component.
(Sheet 1 of 2)	

Variable	Value
<ipport_address>	is the IP address of this IpPort on an IP network.
<netmask_address>	is the IP address of the subnet mask for this *LogicalIf* component.
<pp_value>	is the logical interface to the network.
<vr_name>	is the name of the virtual router.
(Sheet 2 of 2)	

Procedure job aid

Figure 45
Core protocol port component hierarchy



Configuring virtual media for tunnel endpoints

Configure virtual media for tunnel endpoints to define the customer tunnel address to access the VPN.

Procedures steps

1 Add a virtual media (VM).

```
add Vm/<vm_value>
```

2 Set the mode of operation as alwaysUpSummaryInterface for the virtual media interface.

```
set Vm/<vm_value> Interface/0 mode
alwaysUpSummaryInterface
```

3 Add a protocol port.

```
add Vr/<vr_name> Pp/<pp_value>
```

4 Link the protocol port to the media application.

```
set Vr/<vr_name> Pp/<pp_value> media Vm/<vm_value>
Interface/0
```

5 Add the *IpPort* component which contains IP attributes specific to this port.

```
add Vr/<vr_name> Pp/<pp_value> IpPort
```

6 Add the *IpLogicalInterface* component. Add the network address logical interface for the dynamic tunnel. For the static tunnel, add the host address.

```
add Vr/<vr_name> Pp/<pp_value> IpPort LogicalIf/
<ipport_address>
```

7 Set the subnet mask for this logical interface.

```
set Vr/<vr_name> Pp/<pp_value> IpPort LogicalIf/
<ipport_address> netMask/<netmask_address>
```

8 Set the broadcast address for this logical interface.

```
set Vr<vr_name> Pp/<pp_value> IpPort LogicalIf/
<ipport_address> broadcastAddress <broadcast_address>
```

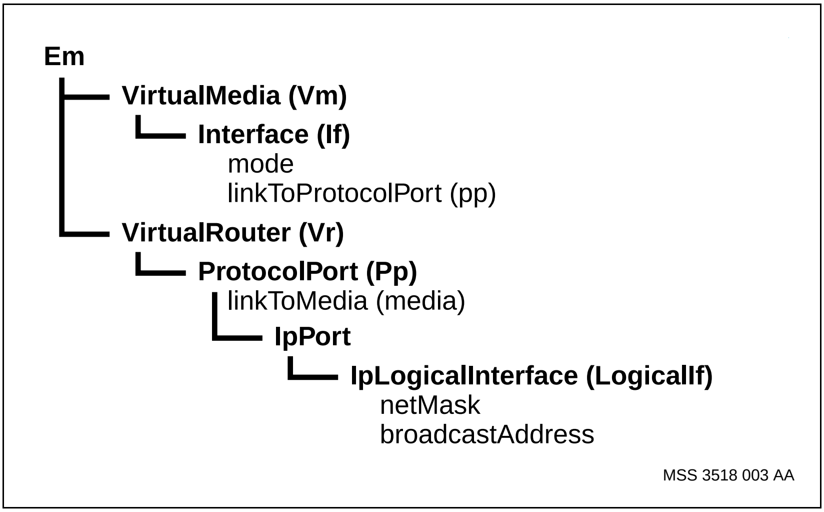
9 Activate changes. See “Activating configuration changes” (page 22).

Variable definitions

Variable	Value
<broadcast_address>	is the IP address of the broadcast address for this *LogicalIf* component.
<ipport_address>	is the IP address of the single IpPort on a logical IP network.
<netmask_address>	is the IP address of the subnet mask for this *LogicalIf* component.
<pp_value>	is the logical interface to the network.
<vm_value>	is an instance of a logical media.
<vr_name>	is the name of the virtual router.

Procedure job aid

Figure 46
Virtual media for tunnel endpoints component hierarchy



Configuring round-trip delay measurements

Configure round-trip delay measurements to measure how long it takes for a packet to travel from a host to a remote end and back again.

Prerequisites

- Ensure that the BGP router ID is the same as the BGP loopback address, which is an always-up IP interface (for example, the IP address of a virtual media protocol port where the mode is set to alwaysUpInterface).

Procedure steps

1 Configure round-trip delay (RTD) measurements.

```
add Vr/<vr_name> Ip Rtd
```

2 Set the *rtdDstAddrList* component.

```
set Vr/<vr_name> Ip Rtd dst <ip_address>
```

3 Enable spooling for the RTD statistics. If you do not enable statistics spooling, you cannot gather ongoing information for historical reporting purposes.

```
set Col/<collector_value> Sp spool on

set Lp/<lp_value> Eng Ds/<datastream_type> agentQ
<agentQ_value>
```

4 Activate changes. See “Activating configuration changes” (page 22).

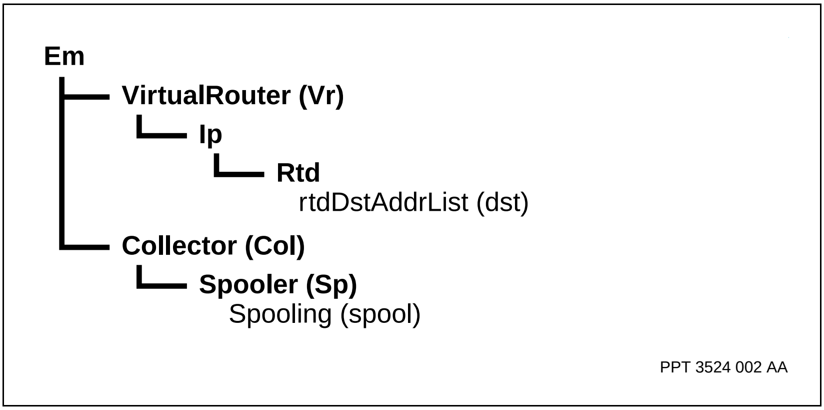
Variable definitions

Variable	Value
<agentQ_value>	is the maximum size of the agent queue size for the data stream of the data collection system (DCS) on the LP.
<collector_value>	is an instance of a data type, for example stats. The main function of the collector is to coordinate the collection of data from all its Agent subcomponents and distribute that data to the downstream applications which have requested it.
(Sheet 1 of 2)	

Variable	Value
<datastream_type>	is engineering parameters used by DCS on this LP for a particular datastream.
<ip_address>	is a list of IP destination addresses for which the round trip delays are calculated for the current round trip delay measurement session.
<lp_value>	is the number of the LP.
<vr_name>	is the name of the virtual router.
(Sheet 2 of 2)	

Procedure job aid

Figure 47
Round-trip delay measurements component hierarchy



Example of configuring BGP on a VCG with dynamic tunnels

This is an example procedure. The values you use in your configuration can differ from the values shown here. Consult your network engineer to ensure the values you are using are accurate for your configuration. The configuration used in this example is shown in “Example of configuring BGP for a VCG for dynamic tunnels” (page 157).

Prerequisites

- The atmMpe/<mpe_value> must be present. See the procedure “Configuring an ATM MPE interface for IP traffic” in NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.

Procedures steps

1 Add a virtual media (Vm).

```
add Vm/<vm_value>
```

2 Add the IP BGP attributes.

```
add -s Vr/VCG1 Ip Bgp
```

3 Set the virtualRouterProcessor.

```
set Vr/VCG1 Vrp Lp/0
```

4 Set the local autonomous system for BGP.

```
set Vr/VCG1 Ip Bgp localAs 100
```

5 Set the bgpIdentifier.

```
set Vr/VCG1 Ip Bgp bgpIdentifier <IP_address_VCG1>
```

6 Add the BPG peer.

```
add Vr/VCG1 Ip Bgp Peer/<IP_address_VCG2>
```

7 Set the peer autonomous system of the BGP peer.

```
set Vr/VCG1 Ip Bgp Peer/<IP_address_VCG2> Desc peerAs
100
```

8 Set the IP address of the BGP peer.

```
set Vr/VCG1 Ip Bgp Peer/<IP_address_VCG2> Desc lac
<IP_address_VCG1>
```

9 Set the addressFamily of the BGP peer.

```
set Vr/VCG1 Ip Bgp Peer/<IP_address_VCG2> Desc
addressFamily ipv4Vpn
```

10 Add the AtmMpe protocol port, IP address and set the network mask.

```
add -s Vr/VCG1 Pp/MPE1A IpPort log/<IP_add_1A> netMask
255.255.255.252
```

11 Add the OSPF.

```
add -s Vr/VCG1 Ip Ospf area/0.0.0.0
```

12 Enable the OSPF spareInstance.

```
set Vr/VCG1 Ip Ospf spareInstance enable
```

13 Set the asBdrRtrStatus.

```
set Vr/VCG1 Ip Ospf asBdrRtrStatus true
```

14 Add the OSPF interface.

```
add Vr/VCG1 Pp/MPE1A Ip Log/<IP_add_1A> OspfIf area
0.0.0.0
```

15 Link the protocol port to the atmMpe.

```
set Vr/VCG1 Pp/MPE1A linkToMedia atmMpe/<mpe_value>
```

16 Add a protocol port for the VM interface.

```
add -s Vr/VCG1 Pp/<vm> Ip Log/<IP_address_VCG1> OspfIf
area 0.0.0.0,ifType passive
```

17 Set the netmask for the protocol port.

```
set Vr/VCG1 Pp/<vm> Log/<IP_address_VCG1> netMask
255.255.255.255
```

18 Set the linkModel to pointToPoint.

```
set Vr/VCG1 Pp/<vm> IpPort linkModel pointToPoint
```

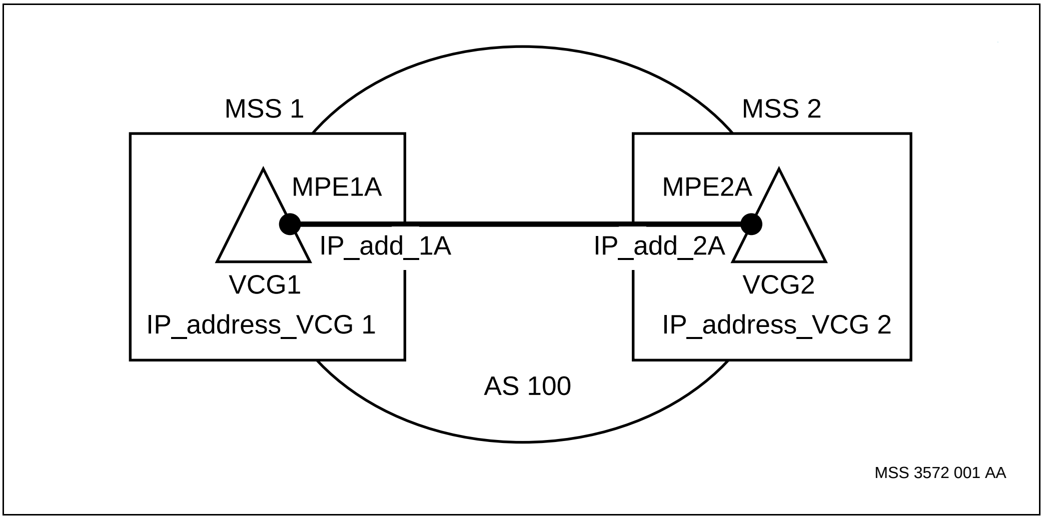
19 Repeat steps 1 through 14 for the second node (MSS2) using the corresponding values (for example, VCG2 would be used to specify the name of the Vr component) indicated in the figure “Example of configuring BGP for a VCG for dynamic tunnels” (page 157).

Variable definitions

Variable	Value
<IP_add_1A>	is the IP address of the AtmMpe protocol port.
<IP_address_VCG1>	is the loopback address of VCG1.
<IP_address_VCG2>	is the loopback address of peer VCG2.
<vm_value>	is the name of the virtual media.

Procedure job aid

Figure 48
Example of configuring BGP for a VCG for dynamic tunnels



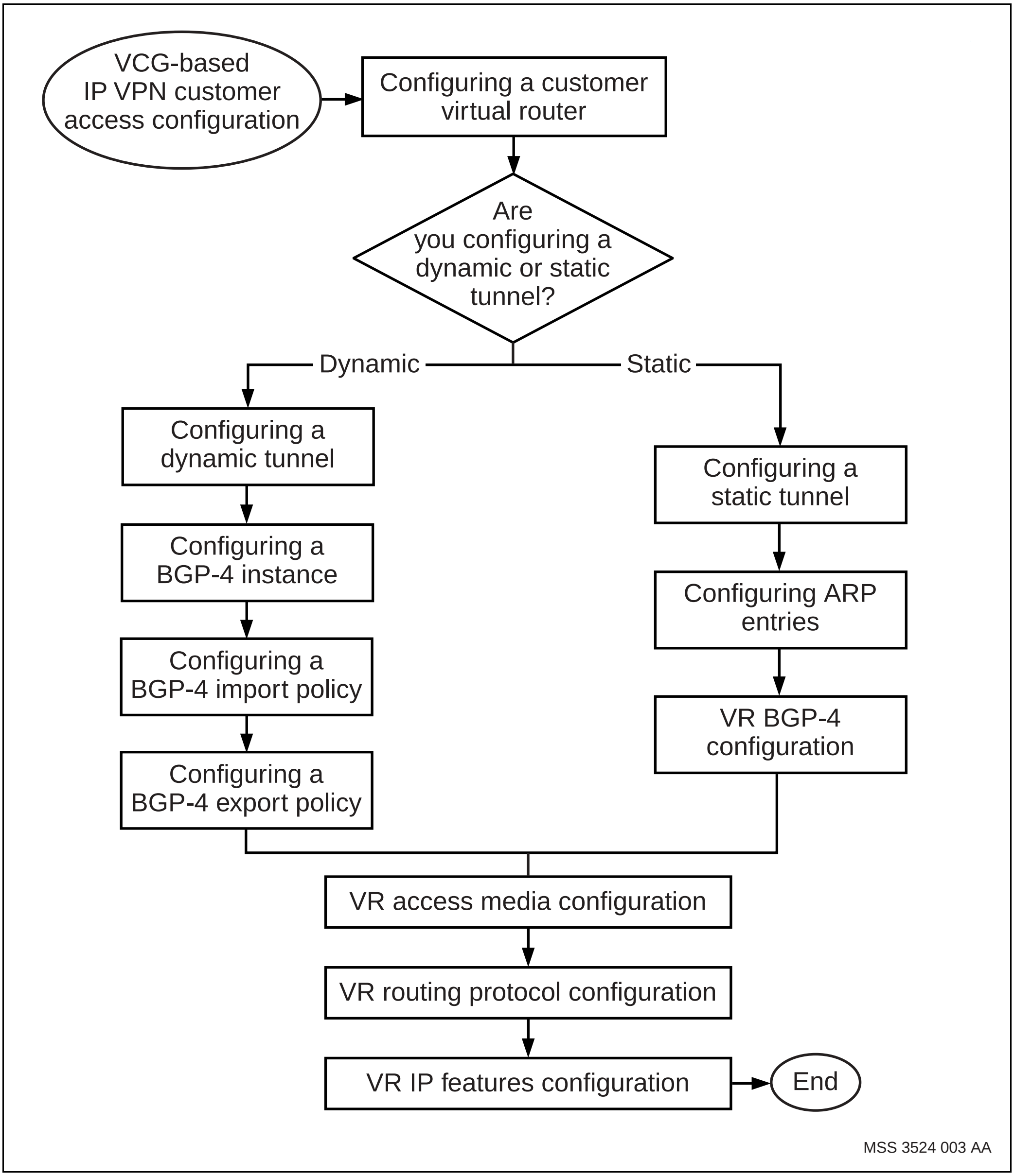
Chapter 10
VCG-based IP VPN customer access configuration

Configure a VCG-based IP VPN customer access configuration to provide a customer site access to a VPN.

VCG-based IP VPN customer access configuration procedures

This task flow shows you the sequence of procedures you perform to configure a VCG-based IP VPN access configuration. To link to any procedure, go to “VCG-based IP VPN customer access configuration procedure navigation” (page 161).

Figure 49
VCG-based IP VPN customer access configuration procedures


VCG-based IP VPN customer access configuration
Configuring a customer virtual router
Are you configuring a dynamic or static tunnel?
Dynamic
Static
Configuring a dynamic tunnel
Configuring a BGP-4 instance
Configuring a BGP-4 import policy
Configuring a BGP-4 export policy
Configuring a static tunnel
Configuring ARP entries
VR BGP-4 configuration
VR access media configuration
VR routing protocol configuration
VR IP features configuration
End
MSS 3524 003 AA

VCG-based IP VPN customer access configuration procedure navigation

- "Configuring a customer virtual router" (page 162)
- "Configuring a dynamic tunnel" (page 164)
- Configuring a BGP-4 instance. See the VR BGP-4 configuration chapter in NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/ 20000 IP Configuration Management*.
- Configuring a BGP-4 import policy. See VR BGP-4 configuration chapter in NN10600-801 *Nortel Networks Multiservice Switch 7400/ 15000/20000 IP Configuration Management*.
- Configuring a BGP-4 export policy. See VR BGP-4 configuration chapter in NN10600-801 *Nortel Networks Multiservice Switch 7400/ 15000/20000 IP Configuration Management*.
- "Configuring a static tunnel" (page 167)
- "Configuring ARP entries" (page 173)
- VR BGP-4 configuration. See NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.
- VR access media configuration. See NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.
- VR routing protocol configuration. See NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.
- VR IP features configuration. See NN10600-801 *Nortel Networks Multiservice Switch 7400/15000/20000 IP Configuration Management*.

Configuring a customer virtual router

Configure a customer virtual router to provide the customer with an access point to the VPN.

Procedure steps

1. Add the virtual router.

   ```
   add Vr/<vr_name>
   ```

2. Set the logical processor on which the virtual router processor resides.

   ```
   set Vr/vr_name> vrp lp/<lp_value>
   ```

3. Set the *virtualPrivateNetworkIdentifier* attribute.

   ```
   set Vr/vr_name> vpnId <vpnid_value>
   ```

4. Add the IP protocol to the VR.

   ```
   add Vr/<vr_name> Ip
   ```

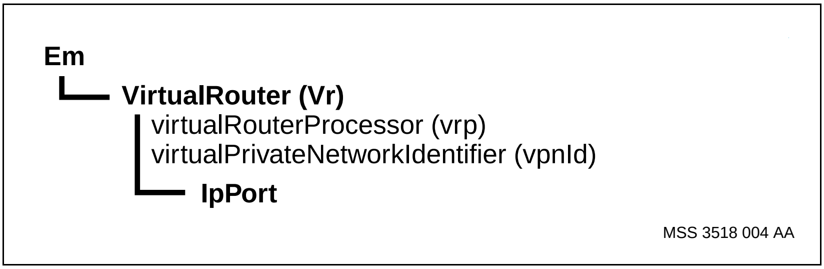
5. Activate changes. See “Activating configuration changes” (page 22)

Variable definitions

Variable	Value
<lp_value>	is the instance value of the logical processor that is linked to the shelf card on which the virtual router resides.
<vpnid_value>	is the instance value of the virtual private network (VPN) associated with this virtual router.
<vr_name>	is the name of the virtual router.

Procedure job aid

Figure 50
Customer virtual router component hierarchy


Em
VirtualRouter (Vr)
virtualRouterProcessor (vrp)
virtualPrivateNetworkIdentifier (vpnId)
IpPort
MSS 3518 004 AA

Configuring a dynamic tunnel

Configure a dynamic tunnel to allow VCGs to automatically exchange customer topology information.

Procedure steps

1 Add the *Tunnel* component to the *VirtualRouter* component. The *MultipointStaticEndPoint* component is added automatically.

```
add Vr/<vr_name> Ip Tunnel
```

2 Set the *autoDiscovery* attribute to enabled.

```
set Vr/<vr_name> Ip Tunnel Msep/<msep_value> disc
enabled
```

3 Set the *sharedDomainVirtualRouter* attribute to be the VCG at the source endpoint of the IP tunnel.

```
set Vr/<vr_name> Ip Tunnel Msep/<msep_value> sdvr
<sdvr_value>
```

4 Set the *sourceAddress* attribute to be one host IP address of local end of the IP tunnel on the shared domain virtual router.

```
set Vr/<vr_name> Ip Tunnel Msep/<msep_value> src
<src_address>
```

5 Add *ProtocolPort* component.

```
add Vr/<vr_name> Pp/<pp_value>
```

6 Link the *ProtocolPort* component to the *MultipointStaticEndPoint* component.

```
set Vr/<vr_name> Pp/<pp_value> media Vr/<vr_name> Ip
Tunnel Msep/<msep_value>
```

7 Add the *IpPort* component which contains IP attributes specific to this port.

```
add Vr/<vr_name> Pp/<pp_value> IpPort
```

8 Add the *IpLogicalInterface* component.

```
add Vr/<vr_name> Pp/<pp_value> IpPort LogicalIf/
<ipport_address>
```

9 Set subnet mask for this logical interface.

```
set Vr/<vr_name> Pp/<pp_value> IpPort LogicalIf/
<ipport_address> netMask/<netmask_address>,
broadcastAddress <broadcast_address>
```

10 Set the broadcast address for this logical interface.

```
set Vr/<vr_name> Pp/<pp_value> IpPort LogicalIf/
<ipport_address> broadcastAddress <broadcast_address>
```

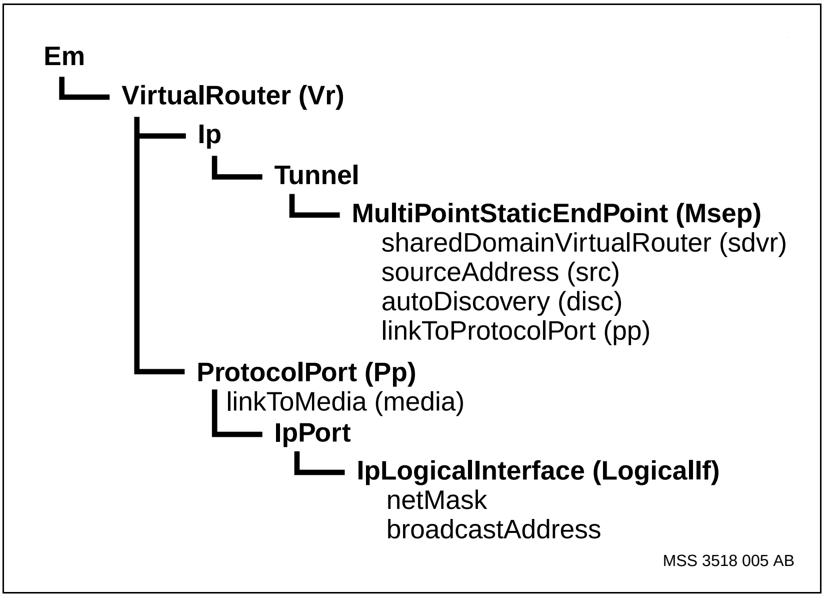
11 Activate changes. See “Activating configuration changes” (page 22).

Variable definitions

Variable	Value
<broadcast_address>	is the IP address of the broadcast address for this *LogicalIf* component.
<ipport_address>	is the IP address of the single IpPort on a logical IP network.
<msep_value>	is an instance that represents a static IP tunnel end point of a point-to-multipoint IP tunneling configuration.
<netmask_address>	is the IP address of the subnet mask for this *LogicalIf* component.
<pp_value>	is the logical interface to the network.
<sdvr_value>	is the *VirtualRouter* component. This virtual router is at the source endpoint of the IP tunnel. The *vpnMode* attribute of that VR must be set to carrier.
<src_address>	is one host IP address of the local end of the IP tunnel on the shared domain virtual router.
<vr_name>	is the name of the virtual router.

Procedure job aid

Figure 51
Dynamic tunnel component hierarchy



Configuring a static tunnel

Configure a static tunnel to manually configure the exchange of topology information between customer virtual routers.

Procedure steps

1 Add the *Tunnel* component to the *VirtualRouter* component. The *MultipointStaticEndPoint* component is added automatically.

```
add Vr/<vr_name> Ip Tunnel
```

2 Set the *sharedDomainVirtualRouter (sdvr)* attribute to be the virtual router at the source endpoint of the IP tunnel.

```
set Vr/<vr_name> Ip Tunnel Msep/<msep_value> sdvr vr/
<vcg_value>
```

3 Set the *sourceAddress* (src) attribute to be the IP address of local end of the IP tunnel in the shared domain.

```
set Vr/<vr_name> Ip Tunnel Msep/<msep_value> src
<src_value>
```

4 Set the *destinationAddresses* attribute.

```
set Vr/<vr_name> Ip Tunnel Msep/<msep_value> dst
<dst_address>
```

5 Optionally, enable tunnel optimization.

```
set Vr/<vr_name> Ip Tunnel optimization enabled
```

6 Create a static address resolution protocol (ARP) entry for the private address of each PTMP IP tunnel end point and specify the public address of the tunnel endpoint on the associated VCGs.

```
add -s Vr/<vr_name> Ip Arp Host/<host>,na tda <tda>
```

7 Add *ProtocolPort* component.

```
add Vr/<vr_name> Pp/<pp_value>
```

8 Link the *ProtocolPort* component to the *MultipointStaticEndPoint* component.

```
set Vr/<vr_name> Pp/<pp_value> media Vr/<vr_name> Ip
Tunnel Msep/<msep_value>
```

9 Add the *IpPort* component which contains IP attributes specific to this port.

```
add Vr/<vr_name> Pp/<pp_value> IpPort
```

10 Add the *IpLogicalInterface* component.

```
add Vr/<vr_name> Pp/<pp_value> IpPort LogicalIf/
<ipport_address>
```

11 Set subnet mask for this logical interface.

```
set Vr/<vr_name> Pp/<pp_value> IpPort LogicalIf/
<ipport_address> netMask/<netmask_address>,
broadcastAddress <broadcast_address>
```

12 Set the broadcast address for this logical interface.

```
set Vr/<vr_name> Pp/<pp_value> IpPort LogicalIf/
<ipport_address> broadcastAddress <broadcast_address>
```

13 Activate changes. See “Activating configuration changes” (page 22).

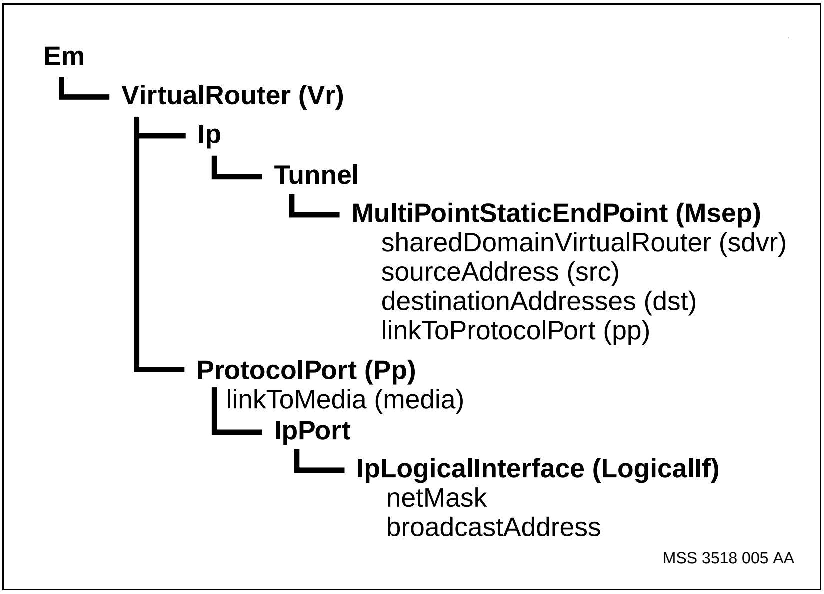
Variable definitions

Variable	Value
<broadcast_address>	is the IP address of the broadcast address for this *LogicalIf* component.
<dst_address>	is a list of IP addresses mapping to each of the far end IP tunnel endpoints in the private domain.
<ipport_address>	is the IP address of the single IpPort on a logical IP network.
<msep_value>	is an instance that represents a static IP tunnel end point of a point-to-multipoint IP tunneling configuration.
<netmask_address>	is the IP address of the subnet mask for this *LogicalIf* component.
<pp_value>	is the logical interface to the network.
(Sheet 1 of 2)	

Variable	Value
<sdvr_value>	is the *VirtualRouter* component. This virtual router is at the source endpoint of the IP tunnel. The *vpnMode* attribute of that VR must be set to carrier.
<src_address>	is the IP address of the local end of the IP tunnel in the shared domain.
<src_address>	is the IP address of the local end of the IP tunnel in the shared domain.
<vr_name>	is the name of the virtual router.
(Sheet 2 of 2)	

Procedure job aid

Figure 52
Static tunnel component hierarchy



Example of configuring a static tunnel

This is an example procedure. The values you use in your configuration can differ from the values shown here. Consult your network engineer to ensure the values you are using are accurate for your configuration. The configuration used in this example is shown in “Example of configuring a static tunnel” (page 172).

Procedure steps

1 Create a PTMP IP tunnel end point on the customer VR.

```
add -s Vr/<CVR1> Ip Tunnel Msep/<msep>
```

2 Specify the VCG as the shared domain VR at the source end of the PTMP IP tunnel.

```
set Vr/<CVR1> Ip Tunnel Msep/<msep_value> sdvr Vr/VCG1
```

3 Specify the source and destination address of the tunnel.

```
set Vr/<CVR1> Ip Tunnel Msep/<msep_value> src
<IP_public_1B>
```

```
set Vr/<CVR1> Ip Tunnel Msep/<msep_value> dst
<IP_public_2B>
```

The source address represents the public address of the tunnel end point on the local VCG. The destination addresses represent the public addresses of the tunnel end points on neighboring VCGs.

4 Create a protocol port and logical interface for the tunnel source address on the customer VR.

```
add -s Vr/<CVR1> Pp/TNLIA IpPort LogicalIf/
<IP_private_1A> netmask 255.255.255.0
```

5 Link the protocol port defined for the PTMP IP tunnel on the customer VR to the IP tunnel end point.

```
set Vr/<CVR1> Pp/TNLIA linkToMedia Vr/<IP_private_1A>
Ip Tunnel Msep/<msep>
```

6 Optionally, enable tunnel optimization.

```
set Vr/<CVR1> Ip Tunnel optimization enabled
```

7 Create a static Address Resolution Protocol (ARP) entry for the private address of each PTMP IP tunnel end point and specify the public address of the tunnel end point on the associated VCGs.

```
add -s Vr/<CVR1> Ip Arp Host/<IP_private_2A>,na tda
<IP_public_1B>
```

8 Optionally, set the maximum transmission unit (MTU) size for the IP packets in the IP tunnel. Set the MTU size for the PTMP IP tunnel to the lowest MTU value supported in the Carrier routing domain. The sample command shows the default MTU value for Ethernet.

```
set Vr/<CVR1> Ip Tunnel Msep/<msep_value> mtu 1500
```

9 Repeat steps 1 through 8 for every PTMP IP tunnel on the Multiservice Switch node.

10 Repeat steps 1 through 9 on Multiservice Switch node 2 (MSS2) replacing the corresponding values (for example, VCG1 becomes VCG2, and so on) as indicated in figure “Example of configuring a static tunnel” (page 172).

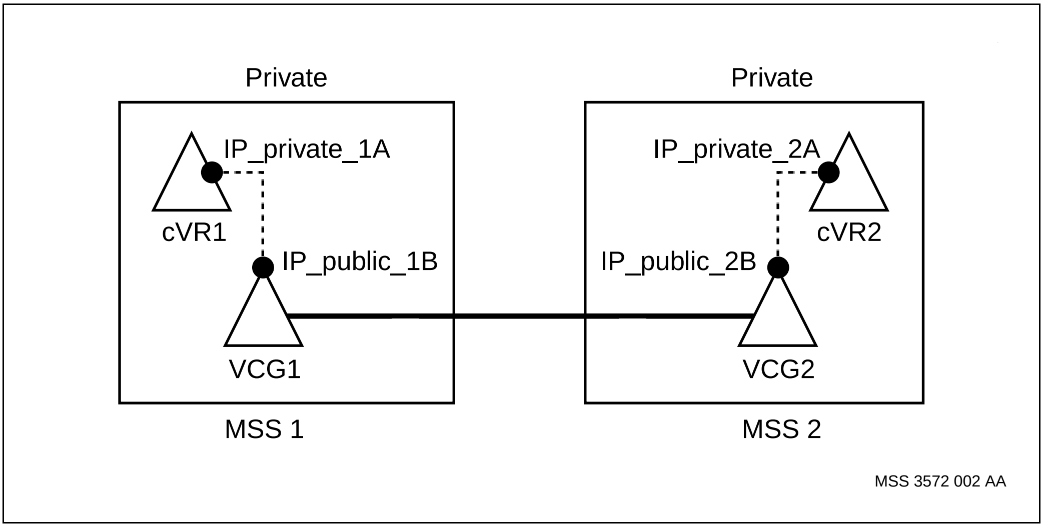
11 Activate changes. See “Activating configuration changes” (page 22).

Variable definitions

Variable	Value
<msep_value>	is an instance that represents a static IP tunnel end point of a point-to-multipoint IP tunneling configuration.

Procedure job aid

Figure 53
Example of configuring a static tunnel



Configuring ARP entries

Configure address resolution protocol (ARP) entries to associate a tunnel destination address to the IP address of the remote customer virtual router.

Procedure steps

1 Add the *HostEntry* component.

```
add Vr/<vr_name> Ip Arp Host/<host_ip_address>, <Cos>
```

2 Set the *tunnelDestinationAddress* attribute.

```
set Vr/<vr_name> Ip Arp Host/<host_ip_address>, <Cos>
tda <tunnel_ip_address>
```

3 Repeat step 1 and step 2 for every tunnel destination in this VPN.

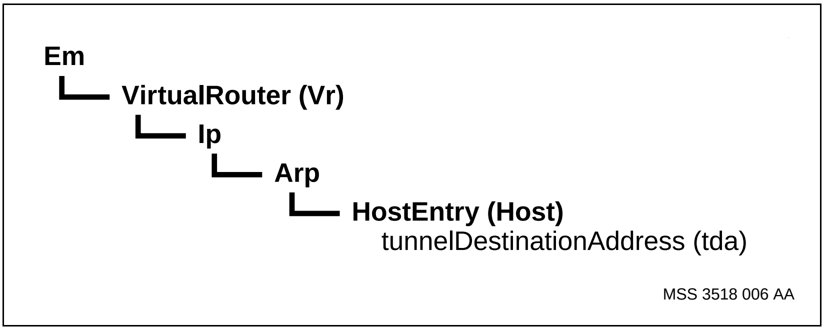
4 Activate changes. See “Activating configuration changes” (page 22).

Variable definitions

Variable	Value
<Cos>	is the one value of the class of service (Cos).
<host_ip_address>	is defining a static host entry in the ARP table. This instance contains an IP address.
<tunnel_ip_address>	is the IP address of a remote IP tunnel endpoint in the shared domain.
<vr_name>	is the name of the virtual router.

Procedure job aid

Figure 54
ARP entries component hierarchy



Nortel Networks Multiservice Switch 7400/15000/20000

IP VPN Configuration Management

Release 6.1

Publication: NN10600-582
Document status: Standard
Document version: 6.1S2
Document date: November 2004
Printed in Canada

NORTEL
NETWORKS