Configuring Ethernet Interfaces 6

This chapter describes how to configure dual-port Fast Ethernet (FE-2), eight-port Fast Ethernet (FE-8), and Gigabit Ethernet (GE) interfaces on line modules on your ERX system.

Most of the procedures described here do not apply to the Fast Ethernet port on the SRP module. You can, however, select and display statistics for that port using commands described in this chapter. For information on managing the Fast Ethernet port on the SRP, see *ERX System Basics Configuration Guide, Chapter 3, Managing Line Modules and SRP Modules*.

Overview

For information about installing Ethernet modules in the system, see the *ERX Installation and User Guide*.

Ethernet modules use the Address Resolution Protocol (ARP) to obtain MAC addresses for outgoing Ethernet frames. For more information about ARP, see *ERX Routing Protocols Configuration Guide, Vol. 1, Chapter 2, Configuring IP*.

Note: *We recommend that you read the section Configuration Tasks before you begin to configure an Ethernet interface.*

Features

Ethernet modules support the following features:

- Routing of IP packets
- Quality of Service (QoS) classification
- Virtual LAN (VLAN) configurations (see section *Configuring VLANs*)
- Stacked Virtual LAN (S-VLAN) configurations (see section *Configuring S-VLANs*)
- Configurations with higher-level protocols (see section *Configuring Higher-Level Protocols*), Layer Two Tunneling Protocol (L2TP: see below)
- Multinetting (see below)

L2TP

Ethernet interfaces support the L2TP. To use L2TP, you must first create a PPP interface. See *ERX Broadband Access Configuration Guide, Chapter 4, Configuring L2TP* for information on configuring L2TP.

Multinetting

Ethernet interfaces, except for bridged Ethernet interfaces, support multinetting; that is, adding more than one IP address to an IP interface. If you want to add multiple IP addresses to a single IP interface, use the **ip address** *ipAddress ipMask* [*secondaryAddress*] command, which is described in *ERX Routing Protocols Configuration Guide, Vol. 1, Chapter 2, Configuring IP*.

References

See the following specifications for Ethernet implementations.

- IEEE 802.1q (FE and GE)
- IEEE 802.3 (FE and GE)
- IEEE 802.3u (FE only)
- IEEE 802.3z (GE only)
- RFC 826 – An Ethernet Address Resolution Protocol (November 1982)
- RFC 894 – A Standard for the Transmission of IP Datagrams over Ethernet Networks (April 1984)
- RFC 1042 – A Standard for the Transmission of IP Datagrams over IEEE 802 Networks (February 1988)
- RFC 1112 – Host Extensions for IP Multicasting (August 1989)
- RFC 2516 – Method for Transmitting PPP over Ethernet (PPPoE) (February 1998)

Ethernet interfaces support MIBs that comply with the following specifications:

- RFC 2863 – The Interfaces Group MIB (June 2000)
- RFC 2668 – Definitions of Managed Objects for IEEE 802.3 Medium Attachment Units (MAUs) (August 1999)
- RFC 2665 – Definitions of Managed Objects for the Ethernet-like Interface Types (August 1998))

Numbering Scheme

Use the *slot/port.subinterface* format to identify Ethernet interfaces and/or subinterfaces:

- *slot* – system chassis slot number
- *port* – number of the port on the I/O module
- *subinterface* – subinterface number of the protocol or VLAN subinterface

FE I/O Modules

An FE-2 I/O module accepts up to two RJ45 connectors, and an FE-8 I/O module accepts up to eight RJ45 connectors.

Figure 6-1 shows the physical ports for FE-2 I/O modules in the ERX-700 series and the ERX-1410 system.

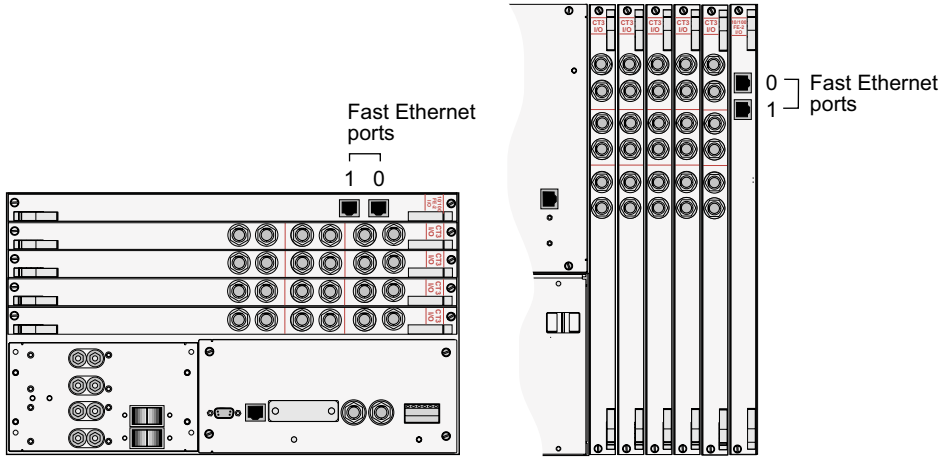

Figure 6-1 FE-2 module in the ERX-700 series and the ERX-1410 system

Figure 6-2 shows the physical ports for FE-8 I/O modules in the ERX-700 series and the ERX-1400 series.

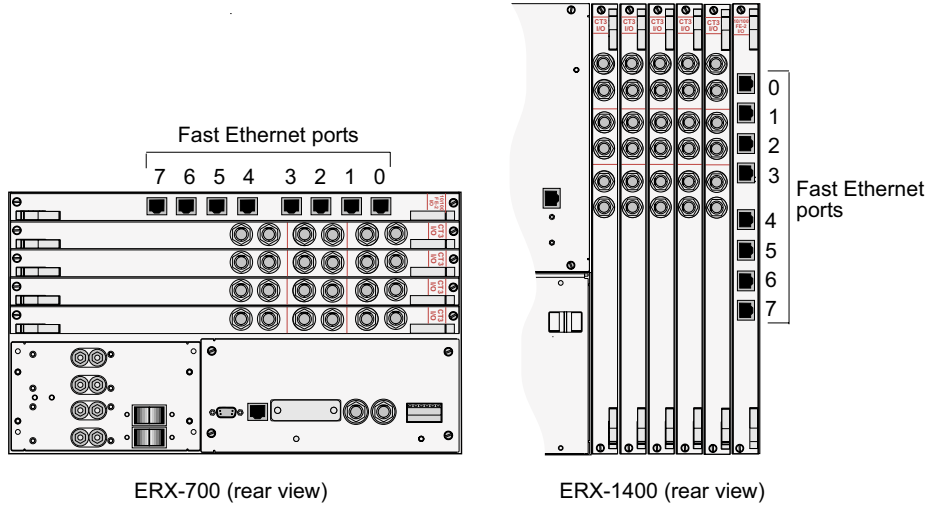

Figure 6-2 FE-8 I/O modules in the ERX-700 series and the ERX-1400 series

GE Modules

A GE module accepts up to two pairs (Tx/Rx) of SC-style fiber-optic connectors.

There are two ports on the GE I/O module: One is active and the other is redundant. If the active port fails, the redundant port automatically becomes active. You can configure only port 0 for a GE interface. Cabling both ports provides a redundant path to the GE interface.

Figure 6-3 shows the physical ports for a GE I/O module in the ERX-700 series and the ERX-1400 series.

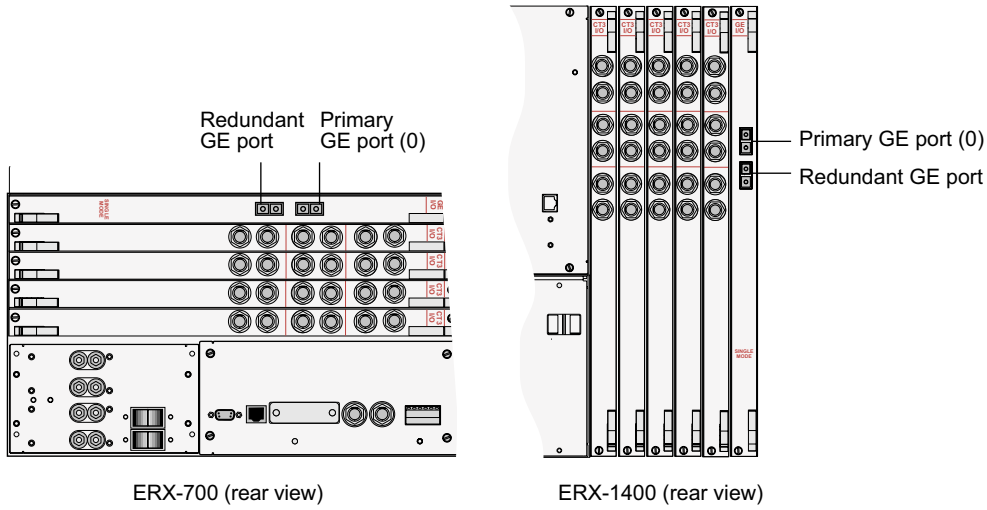

Figure 6-3 GE I/O module in the ERX-700 series and the ERX-1400 series

Configuration Tasks

This section explains your Ethernet configuration options.

You configure an Ethernet interface based on the requirements for your system configuration and the protocols you plan to route on the interface. Because there are different ways you can configure an interface, Ethernet configuration tasks are divided into three primary areas. These areas are further described in separate sections in this chapter.

- Configuring the physical interface – These are the basic configuration steps that must be performed for all interfaces. This task begins with selecting an Ethernet interface and setting parameters such as line speed and MTU.

- Configuring VLANs and stacked VLANs (S-VLANs) – Once you configure the physical interface, you must decide whether to configure the Ethernet interface with or without VLANs or S-VLANs. VLANs and S-VLANs allow you to multiplex multiple IP and/or PPPoE interfaces over a single physical Ethernet port. If you are not configuring with VLANs or S-VLANs, go to the section *Configuring Higher-Level Protocols*.
- Configuring higher-level protocols – You must determine which higher-level protocols, such as MPLS, you will configure on the interface. This section focuses on non-VLAN configurations. Some higher-level protocols, such as PPPoE, however, can be configured with or without VLANs.

Configuring the Physical Interface

This section describes how to complete the basic configuration for an FE or a GE interface. CLI examples are provided with the individual command descriptions.

To configure an Ethernet interface:

1. Select an Ethernet interface.
2. (Optional) Specify the line speed and duplex mode.
3. (Optional) Specify the MTU.
4. (Optional) Set the time interval at which the system records bit and packet rates.
5. (Optional) Associate a name with the interface.
6. (Optional) Validate MAC addresses on a per interface basis.

duplex

- Use to specify the duplex mode.
- This command does not work for the Fast Ethernet port on the SRP.
- Specify both line speed and duplex mode to prevent the system from negotiating these parameters. If you set either the speed or duplex mode to automatically negotiate, the system negotiates both parameters.
- Example

```
host1(config-if)#duplex full
```

- Use the **no** version to revert to the default, automatically negotiate.

ethernet description

- Use to associate a text description of up to 15 characters with an Ethernet interface.
- This command does not work for the Fast Ethernet port on the SRP module.
- The description is displayed by the **show config**, **show interfaces fastEthernet**, and **show interfaces gigabitEthernet** commands.
- Example

```
host1(config-if)#ethernet description abcd1234
```

- The **no** version removes the description from the interface.

interface fastEthernet

- Use to select an FE interface on a line module or SRP module.
- Example

```
host1(config)#interface fastEthernet 1/0
```

- Use the **no** version to remove IP from an interface or subinterface. You must issue the **no** version from the highest level down; you cannot remove an interface or subinterface if the one above it still exists.

***Note**: For more details on the use of this command, see the syntax discussion in the ERX Command Reference Guide.*

interface gigabitEthernet

- Use to select a GE interface.

***Note:** You can configure only the primary port, 0, on the Gigabit Ethernet module. The system automatically uses the redundant port if the primary fails.*

- Example

```
host1(config)#interface gigabitEthernet 1/0
```

- Use the **no** version to remove IP from an interface. You must issue the **no** version from the highest level down; you cannot remove an interface or subinterface if the one above it still exists.

***Note**: For more details on the use of this command, see the syntax discussion in the ERX Command Reference Guide.*

ip mac-validate

- Use to enable or disable MAC address validation on a per interface basis.
- Use the **strict** keyword to prevent transmission of IP packets that do not reside in the validation table.
- Use the **loose** keyword to allow IP packets to pass through even though the packets do not have entries in the validation table. Only packets that have matching IP–MAC pair entries in the table are validated.
- There is no default for this command.
- Example

```
host1(config)#interface gigabitEthernet 2/0
host1(config-if)#ip address 4.4.4.2 255.255.255.0
```

```
host1(config-if)#ip mac-validate strict
host1(config-if)#exit
```

- Use the **no** version to disable the command.

Note: *For additional information about MAC address validation, see the* ***arp validate*** *command in ERX Routing Protocols Configuration Guide, Vol. 1, Chapter 2, Configuring IP.*

load-interval

- Use to set the time interval at which the system calculates bit and packet rate counters.
- This command does not work for the Fast Ethernet port on the SRP module.
- Specify a multiple of 30 seconds, in the range 30 to 300 seconds.
- The default value is 300 seconds.
- Example

```
host1(config-if)#load-interval 90
```

- Use the **no** version to restore the default time interval, 300 seconds.

mtu

- Use to specify the MTU for an interface.
- This command does not work for the Fast Ethernet port on the SRP module.
- Example

```
host1(config-if)#mtu 9000
```

- Use the **no** version to specify the default, 1518.

speed

- Use to specify the line speed.
- This command does not work for the Fast Ethernet port on the SRP module.
- Specify both line speed and duplex mode to prevent the system from negotiating these parameters. If you set either the speed or duplex mode to automatically negotiate, the system negotiates both parameters.
- Example

```
host1(config-if)#speed 10
```

- Use the **no** version to revert to the default, automatically negotiate.

Configuring VLANs

A virtual LAN (VLAN) permits multiplexing multiple IP and/or PPPoEPPPoE and/or MPLS interfaces over a single physical Ethernet port. This multiplexing is accomplished through VLAN subinterfaces. Ethernet interfaces support the 802.1q-1998 IEEE Standards for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks, which the system uses as its standardized format for frame tagging.

The Ethernet V2 frame format allows multiplexing of different protocols over a single physical link. IEEE 802.1q compatibility extends the frame format by adding a tag that contains a VLAN ID. This feature allows multiplexing of different channels (VLANs) over the physical link; each channel is able to multiplex different protocols.

This capability works very much like ATM encapsulation as described in RFC 2684 – Multiprotocol Encapsulation over ATM Adaptation Layer 5 (September 1999). This encapsulation type allows multiplexing of multiple protocols over a single ATM virtual circuit (VC).

As shown in Figure 6-4, VLANs are similar to ATM VCs, with the VLAN ID serving the same function as the virtual path identifier (VPI) and virtual channel identifier (VCI) to multiplex the different channels over the physical link. The Ethernet protocol type serves the same function within a VLAN as the logical link control (LLC) subnetwork attachment point (SNAP) within a VC, to multiplex the different protocols over the channel.

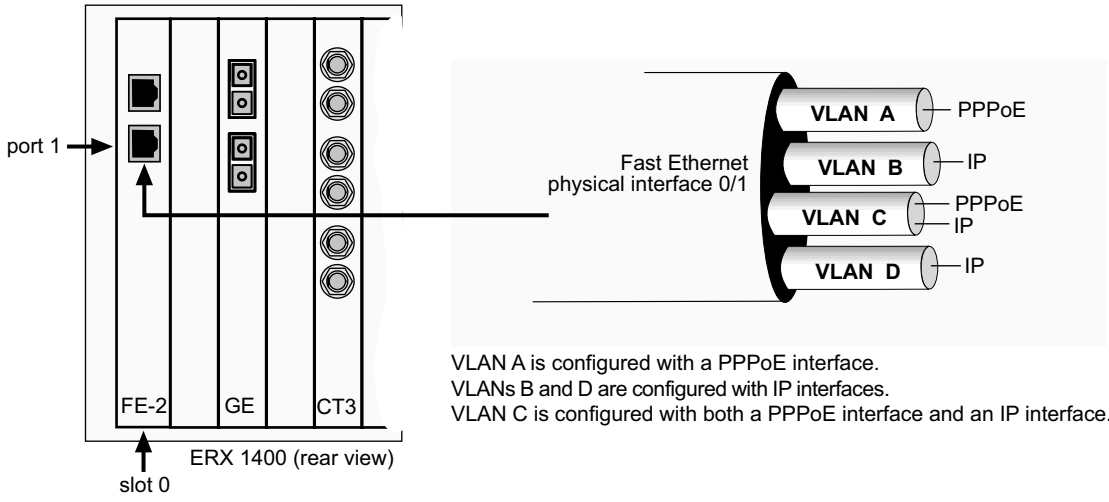

Figure 6-4 Use of VLANs to multiplex different protocols over a single physical link

In a VLAN configuration, the system can send VLAN 0 *tagged* or *untagged frames*.

For more information, see:

- *ERX Routing Protocols Configuration Guide, Vol. 1, Chapter 2, Configuring IP*
- *Chapter 16, Configuring Point-to-Point Protocol over Ethernet*

Creating a VLAN Major Interface

To use VLANs, you must first configure the Ethernet interface for VLAN encapsulation. This creates the VLAN major interface. For example:

1 Specify an FE or a GE port.

```
host1(config)#interface fastethernet 4/0
```

2 Specify VLAN as the encapsulation method.

```
host1(config-if)#encapsulation vlan
```

The VLAN major interface is now created. You may then create multiple VLAN subinterfaces to carry higher-level protocols. Examples are provided in the next section.

Common VLAN Configurations

Ethernet interfaces support either IP, PPPoE, MPLS, or both IP and PPPoE on each VLAN. In addition to a VLAN major interface level, there is a VLAN subinterface level to distinguish the VLAN. Four common VLAN configurations, which can be configured on both FE or GE interfaces, are described in this section.

- IP over VLAN
- PPPoE over VLAN
- MPLS over VLAN
- IP over VLAN and PPPoE over VLAN

Note: *You cannot configure VLANs on the Fast Ethernet port of the SRP module.*

Configuring IP over VLAN

To configure IP over VLAN over an Ethernet interface:

1 Specify an FE or a GE port.

```
host1(config)#interface fastethernet 4/0
```

2 Specify VLAN as the encapsulation method.

```
host1(config-if)#encapsulation vlan
```

The VLAN major interface is added.

3 Create a VLAN subinterface by adding a subinterface number to the interface identification command.

```
host1(config-if)#interface fastethernet 4/0.3
```

4 Assign a VLAN ID for the subinterface.

```
host1(config-if)#vlan id 201
```

5 Assign an IP address and mask.

```
host1(config-if)#ip address 192.6.129.5 255.255.255.0
```

6 (Optional) Configure additional VLAN subinterfaces by completing steps 3 through 5.

Figure 6-5 illustrates the IP/VLAN/FE stacking, showing two separate VLAN subinterfaces. Configure one VLAN subinterface entirely; then configure the next VLAN subinterface.

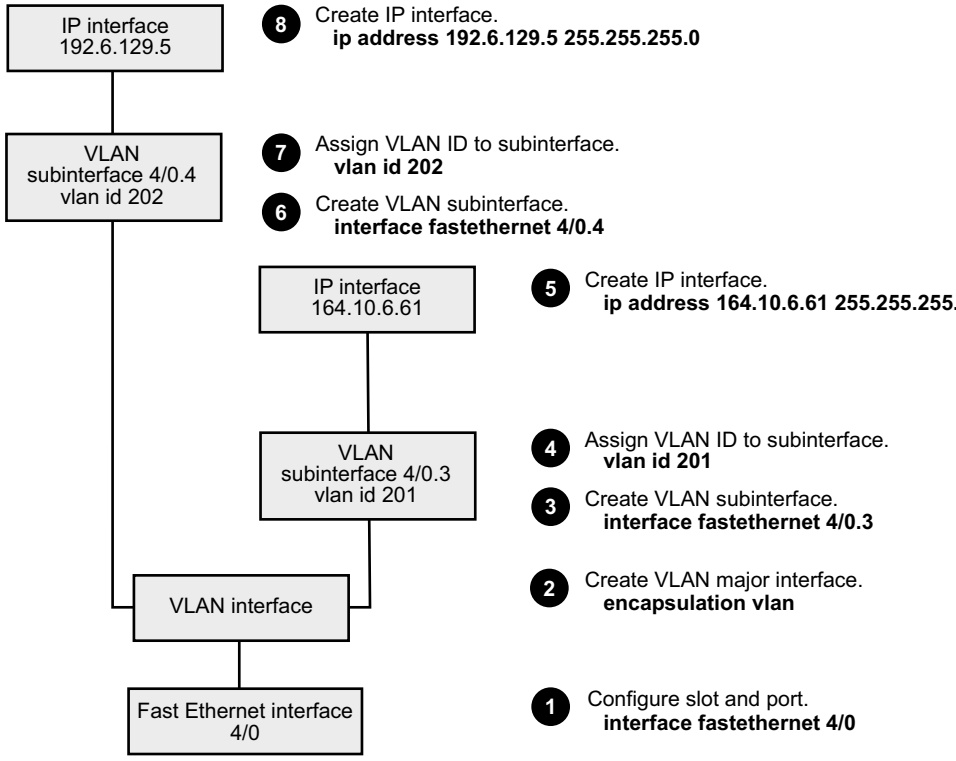

Figure 6-5 Example of IP / VLAN / FE stacking configuration steps

Configuring PPPoE over VLAN

To configure PPPoE over VLAN over an Ethernet interface:

1 Specify an FE or a GE port.

```
host1(config)#interface fastethernet 4/1
```

2 Specify VLAN as the encapsulation method.

```
host1(config-if)#encapsulation vlan
```

The VLAN major interface is added.

3 Create a VLAN subinterface by adding a subinterface number to the interface identification command.

```
host1(config-if)#interface fastethernet 4/1.1
```

4 Assign a VLAN ID for the subinterface.

```
host1(config-if)#vlan id 400
```

5 Specify PPPoE as the encapsulation method on the interface.

```
host1(config-if)#pppoe
```

6 Create a PPPoE subinterface.

```
host1(config-if)#pppoe subinterface fastethernet 4/1.1.1
```

7 Specify PPP as the encapsulation method on the interface.

```
host1(config-if)#encapsulation ppp
```

8 Assign an IP address and mask.

```
host1(config-if)#ip address 192.6.129.5 255.255.255.0
```

9 (Optional) Configure additional VLAN subinterfaces by completing steps 3 through 8.

Figure 6-6 illustrates the PPPoE/VLAN/FE stacking, showing two separate VLAN subinterfaces. One VLAN subinterface has two PPPoE subinterfaces, and one VLAN subinterface has one PPPoE subinterface.

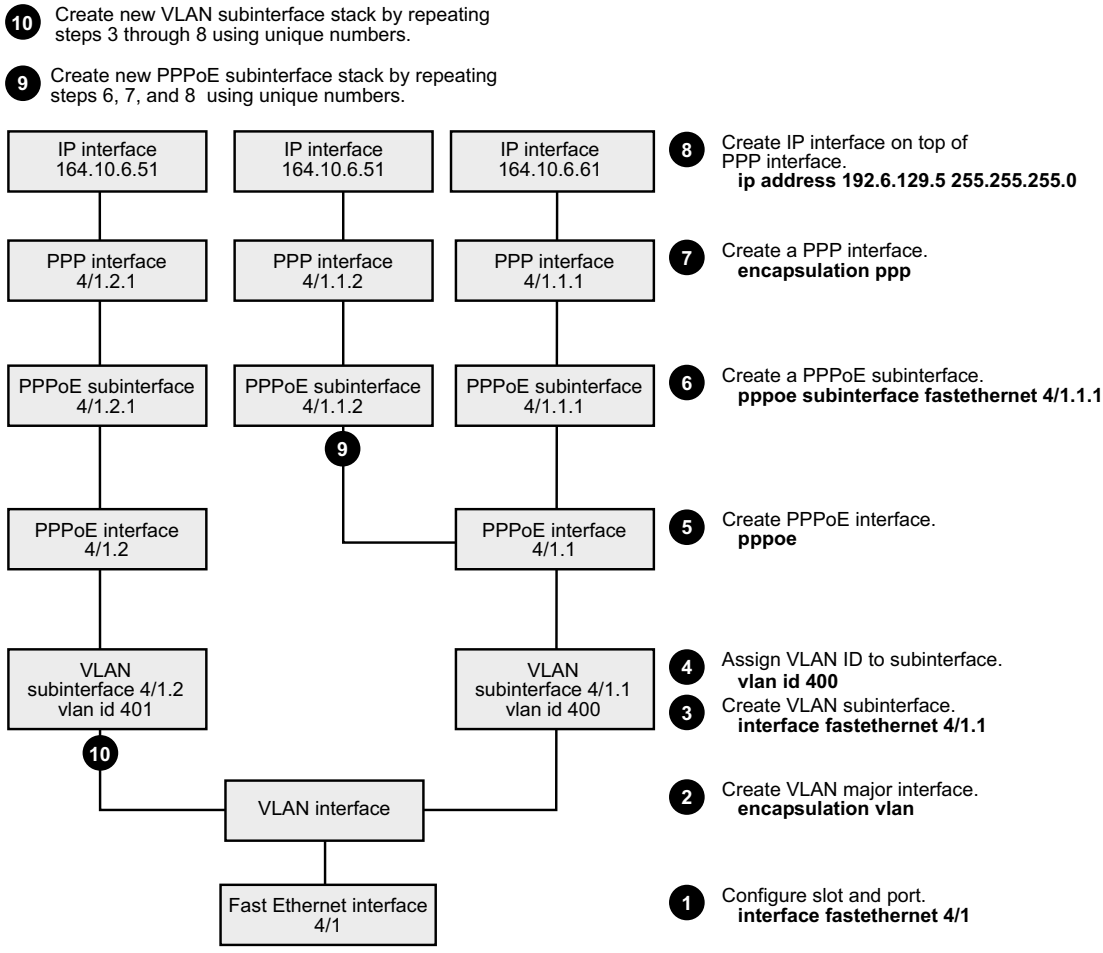

Figure 6-6 Example of PPPoE / VLAN / FE stacking configuration steps

Configuring MPLS over VLAN

To configure MPLS over VLAN over an Ethernet interface:

1 Specify an FE or a GE port.

```
host1(config)#interface fastethernet 4/1
```

2 Specify VLAN as the encapsulation method.

```
host1(config)#encapsulation vlan
```

The VLAN major interface is added.

3 Create a VLAN subinterface by adding a subinterface number to the interface identification command.

```
host1(config-if)#interface fastethernet 4/1.1
```

4 Assign a VLAN ID for the subinterface.

```
host1(config-if)#vland id 400
```

5 Enable MPLS on the interface.

```
host1(config-if)#mpls
```

Figure 6-7 illustrates the MPLS/VLAN/FE stacking showing one VLAN subinterface.

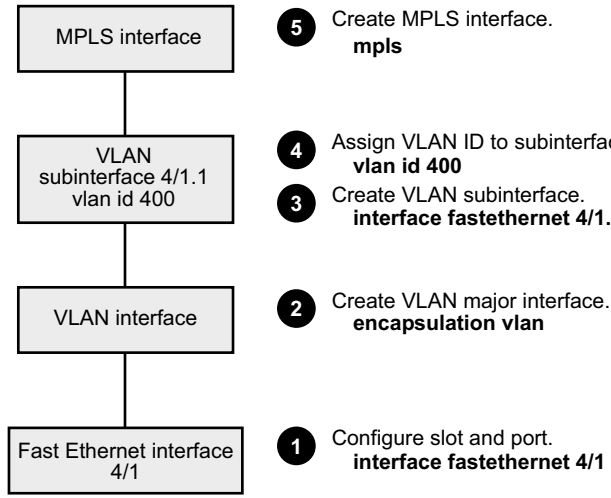

Figure 6-7 Example of MPLS / VLAN / FE stacking configuration steps

Configuring IP over VLAN and PPPoE over VLAN

To configure IP over VLAN with PPPoE over the same VLAN over an Ethernet interface:

1 Specify an FE or a GE port.

```
host1(config)#interface fastethernet 4/1
```

2 Specify VLAN as the encapsulation method.

```
host1(config-if)#encapsulation vlan
```

The VLAN major interface is added.

3 Create a VLAN subinterface by adding a subinterface number to the interface identification command.

```
host1(config-if)#interface fastethernet 4/1.1
```

4 Assign a VLAN ID for the subinterface.

```
host1(config-if)#vlan id 400
```

5 Create an IP interface on the same VLAN as the PPPoE interface.

```
host1(config-if)#ip address 164.10.6.71 255.255.255.0
```

6 Specify PPPoE as the encapsulation method on the interface.

```
host1(config-if)#pppoe
```

7 Create a PPPoE subinterface.

```
host1(config-if)#pppoe subinterface fastethernet 4/1.1.1
```

8 Specify PPP as the encapsulation method on the interface.

```
host1(config-if)#encapsulation ppp
```

9 Assign an IP address and mask.

```
host1(config-if)#ip address 192.6.129.5 255.255.255.0
```

10 (Optional) Configure additional PPPoE subinterfaces by completing steps 7 through 9 using unique numbering.

To configure additional IP interfaces over the VLAN major interface:

1 Create a new VLAN subinterface by adding a unique subinterface number to the interface identification command.

```
host1(config-if)#interface fastethernet 4/1.2
```

2 Assign a VLAN ID for the subinterface.

```
host1(config-if)#vlan id 401
```

3 Assign an IP address and mask.

```
host1(config-if)#ip address 164.10.6.51 255.255.255.0
```

Figure 6-8 shows two VLAN subinterfaces, one with only an IP interface, the other with an IP interface, a PPPoE interface, and multiple PPPoE subinterface stacks.

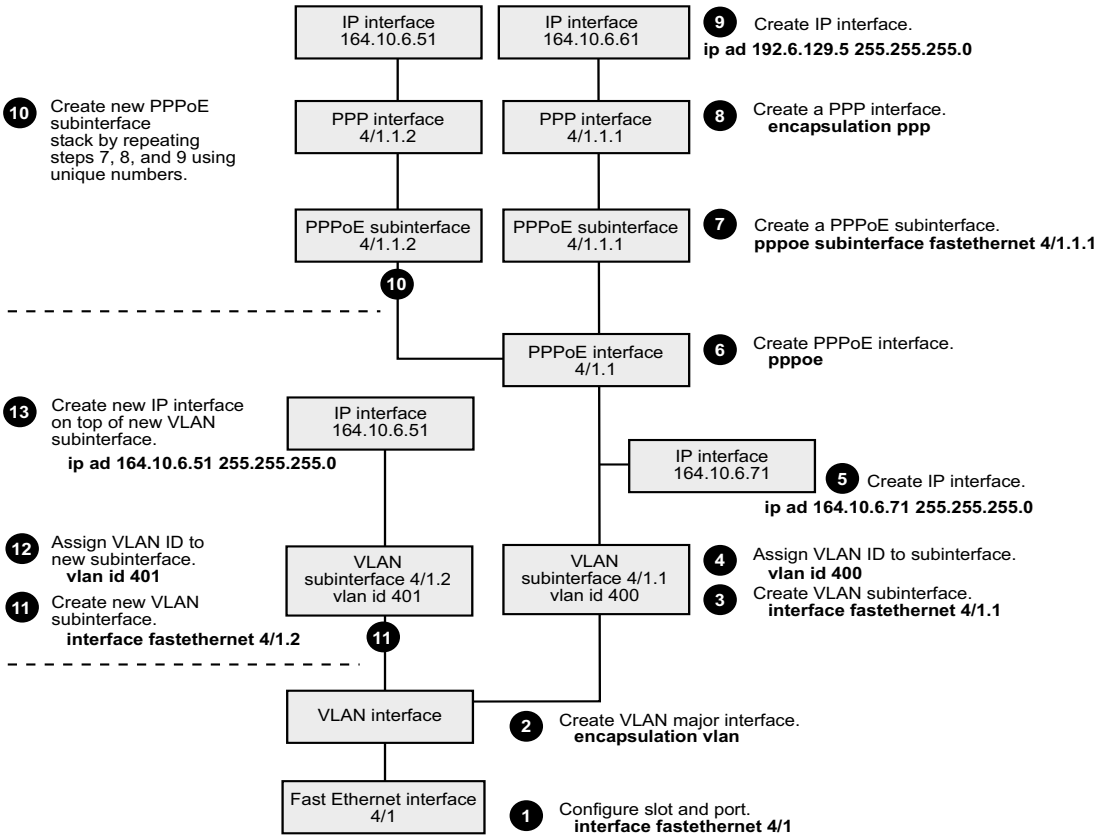

Figure 6-8 Example of PPPoE over VLAN with IP over VLAN stacking configuration steps

Note: *Before you can remove a VLAN subinterface, you must remove the upper-layer interface stack.*

encapsulation ppp

- Use to configure PPP as the encapsulation method for the interface.
- Example

```
host1(config-if)#encapsulation ppp
```

- Use the **no** version to disable PPP on the interface.

encapsulation vlan

- Use to configure VLAN as the encapsulation method for the interface.
- Example

```
host1(config-if)#encapsulation vlan
```

- Use the **no** version to disable VLAN on an interface.

ip address

- Use to set a primary or secondary IP address for an interface or subinterface.

```
host1(config-if)#ip address 192.6.129.5 255.255.255.0
```

- Specify the layer 2 encapsulation before you set the IP address.
- Use the **no** version to remove an IP address or disable IP processing.

pppoe

- Use to configure PPPoE as the encapsulation method on the interface.
- Example

```
host1(config-if)#pppoe
```

- Use the **no** version to disable PPPoE on the interface.

pppoe subinterface fastEthernet

- Use to create a PPPoE subinterface on an FE interface.
- Example

```
host1(config-if)#pppoe subinterface fastethernet 4/1.1.1
```

- Use the **no** version to remove a PPPoE subinterface on an FE interface.

pppoe subinterface GigabitEthernet

- Use to create a PPPoE subinterface on a GE interface.
- Example

```
host1(config-if)#pppoe subinterface gigabitethernet 4/2.1.1
```

- Use the **no** version to remove a PPPoE subinterface on a GE interface.

vlan description

- Use to assign an alias or description to a VLAN subinterface.
- You can use a maximum of 64 characters for the description or to name the alias.
- Example

```
host1(config-if)#vlan description randolph56a
```

- Use the **no** version to remove the VLAN description.

vlan id

- Use to specify the VLAN ID.
- Use a VLAN ID that is in the range of 0–4095 and is unique within the Ethernet interface.
- Example

```
host1(config-if)#vlan id 400
```

- Issue the **vlan id** command before any upper bindings are made, such as IP or PPPoE.

- Use the optional keyword **untagged** to specify that frames be sent untagged. The keyword is valid only for VLAN ID 0. It allows tagged frames to be received, but sends out untagged frames.
- There is no **no** version.

Configuring S-VLANs

As previously mentioned, VLANs permit multiplexing multiple IP and/or PPPoE interfaces over a single physical Ethernet port accomplished through the creation of VLAN subinterfaces. As specified in IEEE Standard 802.1q, the twelve-bit VLAN identifier's tagged frames enables the construction of a maximum of 4,096 distinct VLANs. In an Ethernet B-RAS application environment, however, this VLAN limit is inadequate. A stacked VLAN (S-VLAN) provides a two-level VLAN tag structure, which extends the VLAN ID space to over 16 million VLANs.

Creating an S-VLAN requires the use of a second encapsulation tag. The system performs decapsulation twice, once to get the S-VLAN tag and once to get the VLAN tag. Using this "double tagging" approach allows for over 16 million address possibilities, which more than satisfies the scaling requirement for Ethernet B-RAS applications.

VLAN and S-VLAN subinterfaces can co-exist over the same VLAN major interface. You configure S-VLANs in the same manner as you configure VLANs, with the addition of a few new commands. Because of the similarities, only one S-VLAN example is provided.

Note: *See Release Notes, Appendix A, System Maximums for S-VLAN limitations.*

Configuring PPPoE over S-VLAN

To configure PPPoE over an S-VLAN over an Ethernet interface:

1 Specify an FE or a GE port.

```
host1(config)#interface fastethernet 4/1
```

2 Specify VLAN as the encapsulation method.

```
host1(config-if)#encapsulation vlan
```

The VLAN major interface is added.

3 Create a VLAN subinterface by adding a subinterface number to the interface identification command.

```
host1(config-if)#interface fastethernet 4/1.1
```

4 Assign S-VLAN ID and VLAN ID for the subinterface.

```
host1(config-if)#svlan id 4 255
```

5 Assign an S-VLAN Ethertype.

```
host1(config-if)#svlan ethertype 9200
```

6 Specify PPPoE as the encapsulation method on the interface.

```
host1(config-if)#pppoe
```

7 Create a PPPoE subinterface.

```
host1(config-if)#pppoe subinterface fastethernet 4/1.1.1
```

8 Specify PPP as the encapsulation method on the interface.

```
host1(config-if)#encapsulation ppp
```

9 Assign an IP address and mask.

```
host1(config-if)#ip address 192.6.129.5 255.255.255.0
```

10 (Optional) Configure additional PPPoE subinterfaces by completing steps 7 through 9 using unique numbering.

Figure 6-8 shows one S-VLAN subinterface with multiple PPPoE subinterface stacks.

Note: *Before you can remove an S-VLAN/VLAN subinterface, you must remove the upper-layer interface stack.*

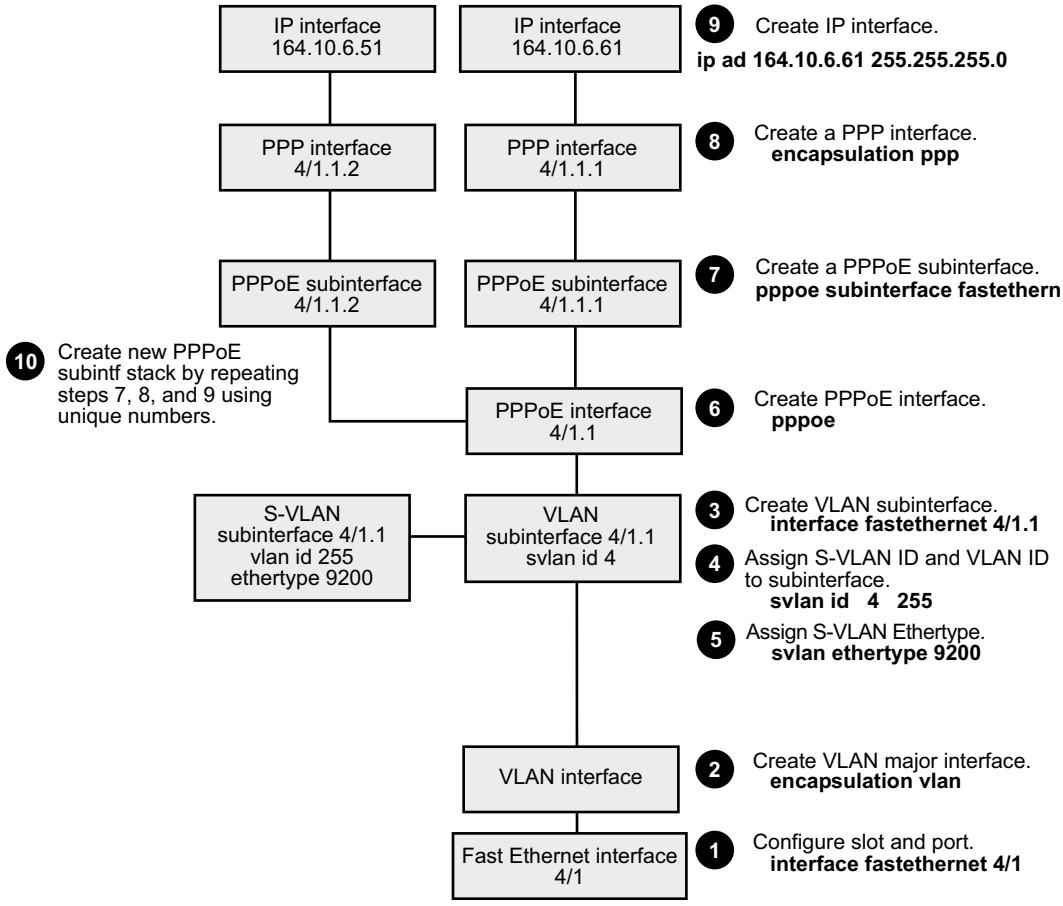

Figure 6-9 Example of PPPoE over S-VLAN stacking configuration steps

encapsulation ppp

- Use to configure PPP as the encapsulation method for the interface.
- Use the **no** version to disable PPP on the interface.

encapsulation vlan

- Use to configure VLAN as the encapsulation method for the interface.
- Use the **no** version to disable VLAN on an interface.

pppoe

- Use to configure PPPoE as the encapsulation method on the interface.
- Use the **no** version to disable PPPoE on the interface.

ip address

- Use to set a primary or secondary IP address for an interface or subinterface.
- Specify the layer 2 encapsulation before you set the IP address.
- Use the **no** version to remove an IP address or disable IP processing.

pppoe subinterface fastEthernet

- Use to create a PPPoE subinterface on an FE interface.
- Use the **no** version to remove a PPPoE subinterface on an FE interface.

pppoe subinterface GigabitEthernet

- Use to create a PPPoE subinterface on a GE interface.
- Use the **no** version to remove a PPPoE subinterface on a GE interface.

svlan ethertype

- Use to assign an Ethertype for the S-VLAN subinterface.
- Example

```
host1(config-if)#svlan ethertype 9200
```

- The Ethertype must be either 9100 or 9200. The default is 9100.
- Use the **no** version to remove the Ethertype assignment.

svlan id

- Use to assign S-VLAN IDs and VLAN IDs to VLAN subinterfaces.
- Use S-VLAN ID and VLAN ID numbers that are in the range of 0–4095 and that are unique within the Ethernet interface.
- Example

```
host1(config-if)#svlan id 4 255
```

- Issue the **svlan id** command before any upper bindings are made, such as IP or PPPoE.
- There is no **no** version.

Configuring Higher-Level Protocols

You can configure one or more protocols over Ethernet with or without VLANs. This section focuses on non-VLAN configurations only. You can configure the following higher-level protocols on FE and GE interfaces:

- IP
- Point-to-Point Protocol over Ethernet (PPPoE)
- Multiprotocol Label Switching (MPLS)

Ethernet configuration examples using combinations of these protocols are shown in this section. Figure 6-10 illustrates how different protocols can be multiplexed over a single physical link without the use of VLANs.

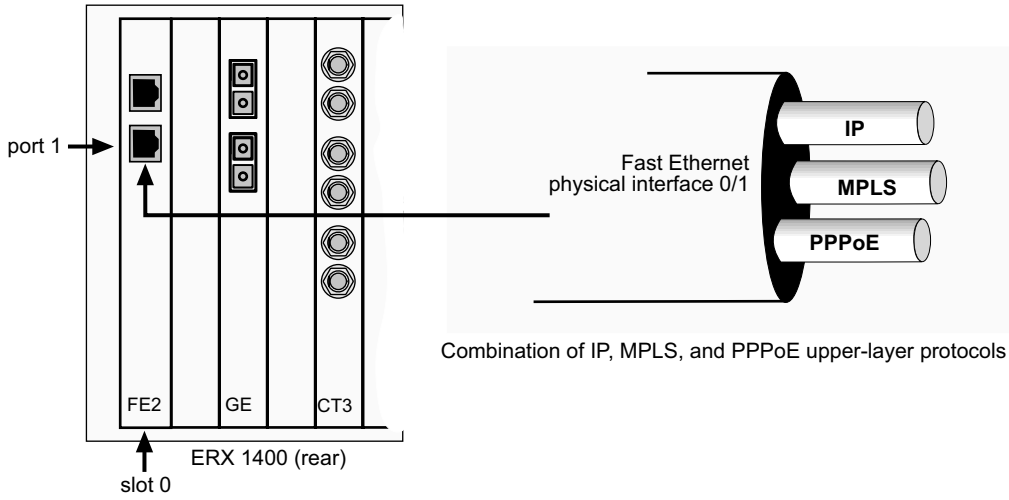

Figure 6-10 Multiplexing multiple protocols over a single physical link

There are two methods to configure multiple protocols over Ethernet without VLANs. The older method, described in *Alternative Method* on page 6-26, requires the use of Ethernet subinterfaces. This method is still supported for backward compatibility, but we do not recommend it for new configurations.

Note: *There are two variations of the procedure to create a multiprotocol configuration without VLANs. Follow the current methods if you are running the current system software version. Follow the alternative method if you are using a software version earlier than Release 3.0.0, or if you are running scripts or macros that were created based on a software version earlier than Release 3.0.0.*

Common Non-VLAN Configurations

Four common non-VLAN configurations are described in this section, which can be configured on both FE or GE interfaces:

- IP over Ethernet
- PPPoE over Ethernet
- IP over Ethernet and MPLS over Ethernet
- IP over Ethernet, MPLS over Ethernet, and PPPoE over Ethernet

Configuring IP over Ethernet

To configure IP over an Ethernet interface:

1 Specify an FE or a GE port.

```
host1(config)#interface fastethernet 4/1
```

2 Create an IP interface.

```
host1(config-if)#ip address 192.5.127.8 255.255.255.0
```

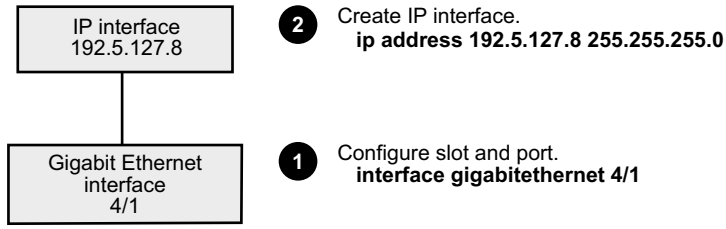

Figure 6-11 Example of IP over Ethernet stacking configuration steps

Configuring PPPoE over Ethernet

To configure PPPoE over an Ethernet interface:

1 Specify an FE or a GE port.

```
host1(config)#interface fastethernet 4/1
```

2 Specify PPPoE as the encapsulation method on the interface.

```
host1(config-if)#pppoe
```

3 Create a PPPoE subinterface.

```
host1(config-if)#pppoe subinterface fastethernet 4/1.1
```

4 Specify PPP as the encapsulation method on the interface.

```
host1(config-if)#encapsulation ppp
```

5 Assign an IP address and mask.

```
host1(config-if)#ip address 192.6.129.5 255.255.255.0
```

6 (Optional) Configure additional PPPoE subinterfaces by completing steps 3 through 5 using unique numbering.

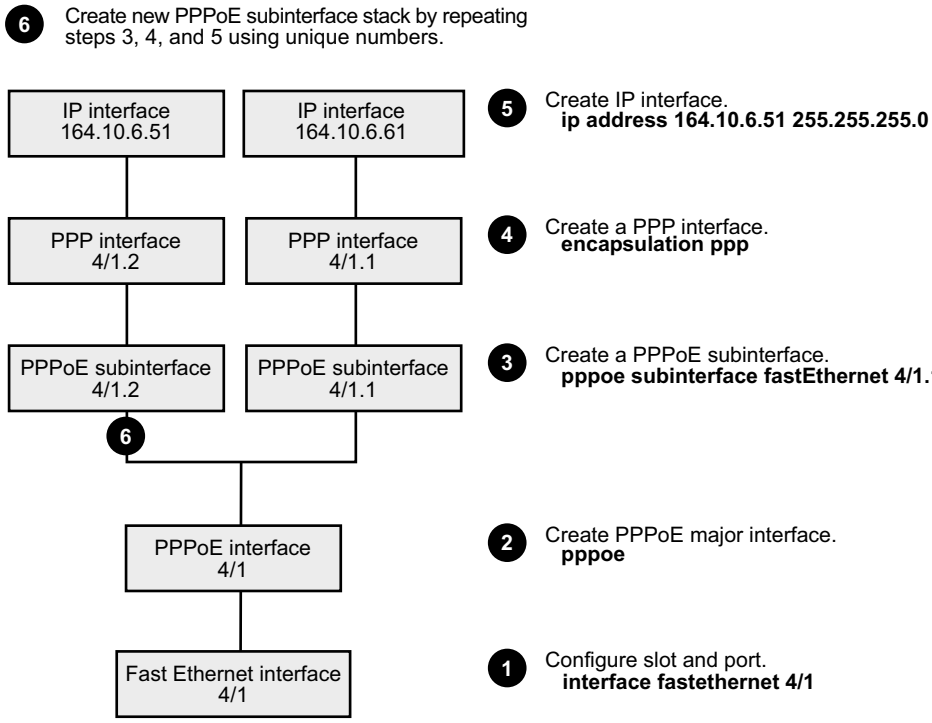

Figure 6-12 Example of PPPoE stacking configuration steps

Configuring IP and MPLS over Ethernet

To configure both IP and MPLS over an Ethernet interface:

1 Specify an FE or a GE port.

```
host1(config)#interface fastethernet 4/1
```

2 Create an IP interface.

```
host1(config-if)#ip address 192.5.127.8 255.255.255.0
```

3 Create an MPLS interface.

```
host1(config-if)#mpls
```

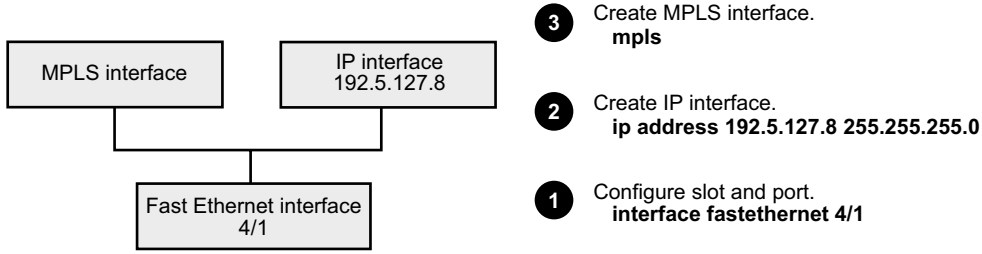

Figure 6-13 Example of IP and MPLS stacking configuration steps

Configuring IP, MPLS, and PPPoE over Ethernet

To configure IP, MPLS, and PPPoE over an Ethernet interface:

1 Specify an FE or a GE port.

```
host1(config)#interface fastethernet 4/1
```

2 Create an IP interface.

```
host1(config-if)#ip address 192.5.127.8 255.255.255.0
```

3 Create an MPLS interface.

```
host1(config-if)#mpls
```

4 Create a PPPoE interface by specifying PPPoE as the encapsulation method on the interface.

```
host1(config-if)#pppoe
```

5 Create a PPPoE subinterface.

```
host1(config-if)#pppoe subinterface fastethernet 4/1.1
```

6 Specify PPP as the encapsulation method on the interface.

```
host1(config-if)#encapsulation ppp
```

7 Assign an IP address and mask.

```
host1(config-if)#ip address 192.6.129.5 255.255.255.0
```

8 (Optional) Configure additional PPPoE subinterfaces by completing steps 5 through 7 using unique numbering.

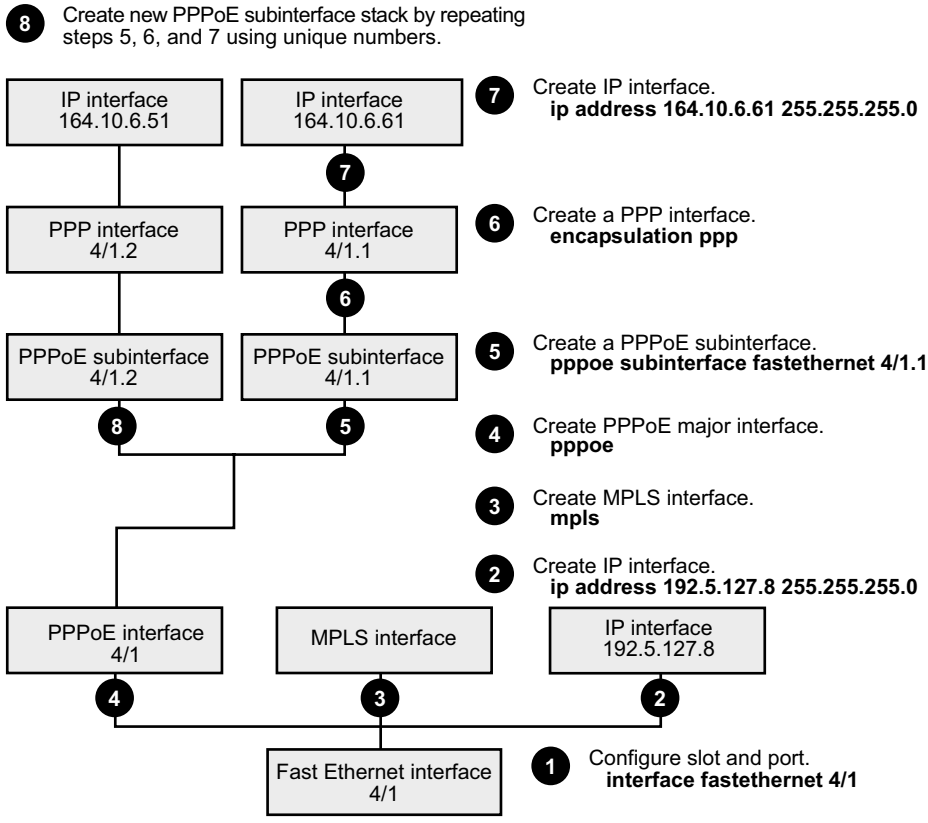

Figure 6-14 Example of IP, MPLS, and PPPoE stacking configuration steps

mpls

- Use to enable, disable, or delete MPLS on an interface. MPLS is disabled by default.
- Example

```
host1(config)#mpls
```

- Use the **no** version to halt MPLS on the interface and delete the MPLS interface configuration.

Alternative Method

Use the following configuration method if you are using a software version earlier than Release 3.0.0, or if you are running scripts or macros that were created based on a version of software earlier than Release 3.0.0. MPLS interfaces were not supported in earlier software versions.

The protocol subinterface is distinguished by adding a subinterface number to the interface identification string **fastEthernet** or **gigabitEthernet**.

Note: *Remove the upper-layer interface stack before removing a protocol subinterface.*

Configuring IP and PPPoE over Ethernet

To configure IP and PPPoE over an Ethernet interface:

1 Specify an FE or a GE port.

```
host1(config)#interface fastethernet 2/0
```

2 Create a Fast Ethernet subinterface.

```
host1(config-if)#interface fastethernet 2/0.1
```

3 Assign an IP address and subnet mask to the interface.

```
host1(config-if)#ip address 192.1.1.1 255.255.255.0
```

4 Create a new Fast Ethernet subinterface.

```
host(config-if)#interface fastethernet 2/0.2
```

5 Specify PPPoE as the encapsulation method on the interface.

```
host1(config-if)#encapsulation pppoe
```

6 Create a PPPoE subinterface.

```
host1(config-if)#interface fastethernet 2/0.2.1
```

7 Specify PPP as the encapsulation method on the interface.

```
host1(config-if)#encapsulation ppp
```

8 Assign an IP address and mask.

```
host1(config-if)#ip address 192.6.129.5 255.255.255.0
```

9 (Optional) Configure additional PPPoE subinterfaces by completing steps 5 through 7 using unique numbering.

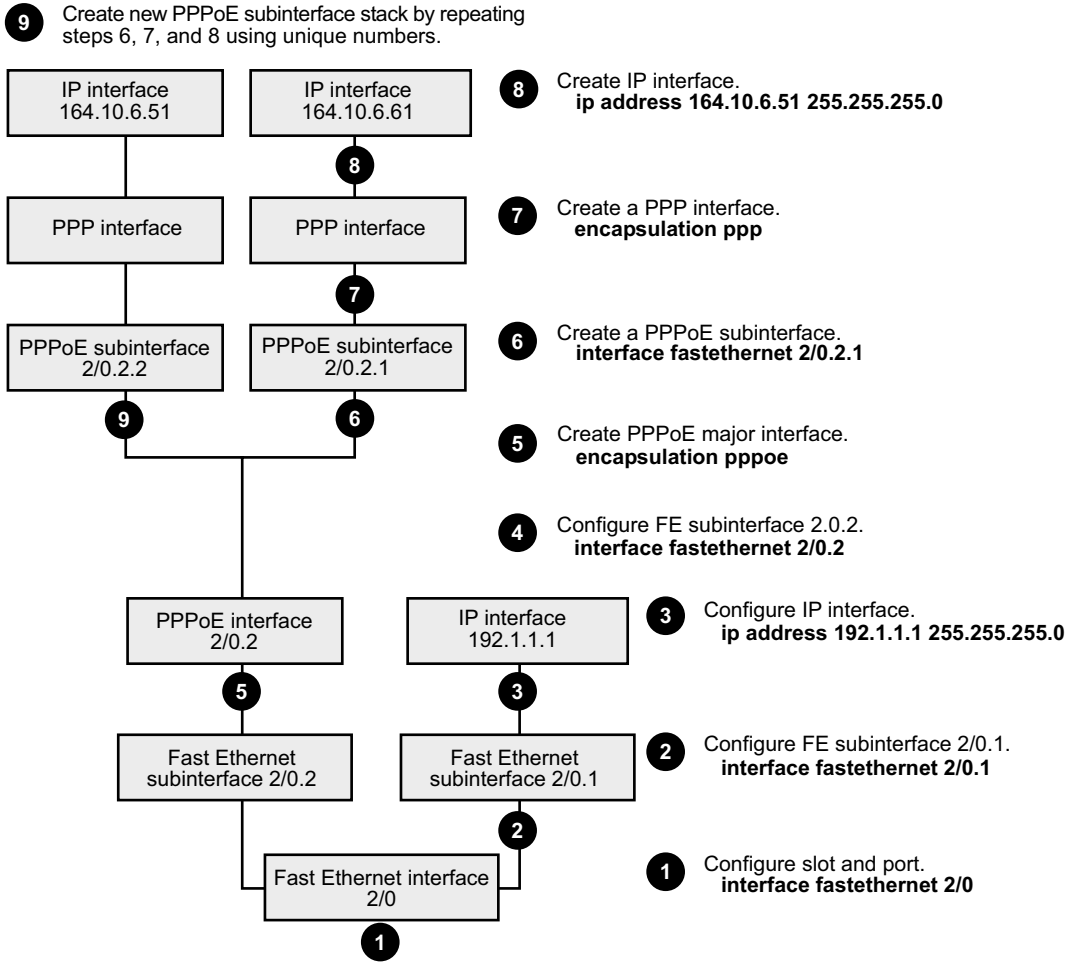

Figure 6-15 Example of alternative method stacking configuration steps

Disabling Ethernet Interfaces

Use the following command and guidelines to disable an Ethernet interface.

shutdown

- Use to disable an Ethernet interface.
- Example

```
host1(config-if)#shutdown
```

- Use the **no** version to restart a disabled Ethernet interface.

Monitoring Ethernet Interfaces

You can display the physical characteristics and the configured settings for Ethernet interfaces.

The system stores statistics in system counters that reset only when you reboot. However, you can establish a baseline for Ethernet statistics by setting a group of reference counters to zero.

baseline interface fastEthernet | gigabitEthernet

- Use to establish a baseline for Fast Ethernet or Gigabit Ethernet statistics on a line module or an SRP module.
- Use the **delta** keyword with the **show interfaces fastEthernet** or the **show interfaces gigabitEthernet** command to display baselined statistics.

show interfaces fastEthernet

- Use to display the status of FE interfaces on line modules or SRP modules.
- You can specify the following keywords:
  - **delta** – specifies that baselined statistics are to be shown
  - **brief** – displays the operational status of all configured interfaces
- Field descriptions
  - *FastEthernet interfaceSpecifier* – status of the hardware on this interface
    - up – hardware is operational
    - down – hardware is not operational
  - Administrative status – operational state that you configured for this interface
    - up – interface is enabled
    - down – interface is disabled
  - Hardware – type of MAC device on this interface
  - Address – MAC address of the processor on this interface
  - MTU – size of the MTU for this interface
    - Operational – size of the largest packet processed
    - Administrative – setting for MTU size that you specified
  - Duplex – duplex option for this interface
    - Operational – duplex option currently used
    - Administrative – setting for duplex that you specified
  - Speed – line speed for this interface
    - Operational – current rate at which packets are processed
    - Administrative – setting for line speed
    - 5 minute input rate – data rates based on traffic received in the last 5 minutes
    - 5 minute output rate – data rates based on traffic sent in the last 5 minutes
  - In – analysis of inbound traffic on this interface

- Bytes – number of bytes received in error-free packets
- Unicast – number of unicast packets received
- Multicast – number of multicast packets received
- Broadcast – number of broadcast packets received
- Errors – total number of errors in all received packets; some packets may contain more than one error
- Discards – total number of discarded incoming packets
- MAC Errors – number of incoming packets discarded because of MAC sublayer failures
- Alignment – number of incomplete octets received
- CRC – number of packets discarded because the checksum the system computed from the data does not match the checksum generated by the originating devices
- Too Longs – number of packets discarded because the size exceeded the MTU
- Symbol Errors – number of symbols received that the system could not correctly decode

› Out – analysis of outbound traffic on this interface

- Bytes – number of bytes sent
- Unicast – number of unicast packets sent
- Multicast – number of multicast packets sent
- Broadcast – number of broadcast packets sent
- Errors – total number of errors in all transmitted packets; some packets may contain more than one error
- Discards – total number of discarded outgoing packets
- MAC Errors – number of outgoing packets discarded because of MAC sublayer failures.
- Deferred – number of packets that the system delayed sending because the line was busy. In half duplex mode, a high number of deferrals means the link is very busy with traffic from other stations. In full duplex mode, the link is always available for transmission, and this number should be zero.
- No Carrier – number of packets sent when carrier sense was unavailable

› Collisions – analysis of the collisions that occurred

- Single – packets sent after one collision
- Multiple – packets sent after multiple collisions
- Late – packets aborted during sending because of collisions after 64 bytes
- Excessive – packets not sent because of too many collisions

- Example

```
host1:vr2#show interfaces fastEthernet2/0
FastEthernet2/0 is Up, Administrative status is Up
  Hardware is Intel 21440, address is 0090.1a10.0552
  MTU: Operational 1518, Administrative 1518
  Duplex Mode: Operational Full Duplex, Administrative Auto Negotiate
  Speed: Operational 100 Mbps, Administrative Auto Negotiate

  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec

  In: Bytes 448, Unicast 0
   Multicast 0, Broadcast 7
   Errors 0, Discards 0, Mac Errors 0, Alignment 0
   CRC 0, Too Longs 0, Symbol Errors 0
  Out: Bytes 384, Unicast 0
   Multicast 0, Broadcast 6
   Errors 0, Discards 0, Mac Errors 0, Deferred 0, No Carrier 0
   Collisions: Single 0, Multiple 0, Late 0, Excessive 0
```

show interfaces gigabitEthernet

- Use to display the status of GE interfaces.
- You can specify the following keywords:
  - **delta** – specifies that baselined statistics are to be shown
  - **brief** – displays the operational status of all configured interfaces
- Field descriptions
  - gigabitEthernet *interfaceSpecifier* – status of the hardware on this interface
    - up – hardware is operational
    - down – hardware is not operational
  - Administrative status – operational state that you configured for this interface
    - up – interface is enabled
    - down – interface is disabled
  - Hardware – type of MAC device on this interface
  - Address – MAC address of the processor on this interface
  - MTU – size of the MTU for this interface
    - Operational – size of the largest packet processed
    - Administrative – setting for MTU size that you specified
  - Duplex – duplex option for this interface
    - Operational – duplex option currently used
    - Administrative – setting for duplex that you specified
  - Speed – line speed for this interface
    - Operational – current rate at which packets are processed
    - Administrative – setting for line speed that you specified

- 5 minute input rate – data rates based on the traffic received in the last 5 minutes
- 5 minute output rate – data rates based on the traffic sent in the last 5 minutes
- In – analysis of inbound traffic on this interface
  - Bytes – number of bytes received in error-free packets
  - Unicast – number of unicast packets received
  - Multicast – number of multicast packets received
  - Broadcast – number of broadcast packets received
  - Errors – total number of errors in all received packets; some packets may contain more than one error
  - Discards – total number of discarded incoming packets
  - Mac Errors – number of incoming packets discarded because of MAC sublayer failures
  - Alignment – number of incomplete octets received
  - CRC – number of packets discarded because the checksum that the system computed from the data does not match the checksum generated by the originating devices
  - Too Longs – number of packets discarded because the size exceeded the MTU
  - Symbol Errors – number of symbols received that the system could not correctly decode
- Out – analysis of outbound traffic on this interface
  - Bytes – number of bytes sent
  - Unicast – number of unicast packets sent
  - Multicast – number of multicast packets sent
  - Broadcast – number of broadcast packets sent
  - Errors – total number of errors in all transmitted packets; note that some packets may contain more than one error
  - Discards – total number of discarded outgoing packets
  - Mac Errors – number of outgoing packets discarded because of MAC sublayer failures
  - Deferred – number of packets that the system delayed sending because the line was busy. In half duplex mode, a high number of deferrals means the link is very busy with traffic from other stations. In full duplex mode, the link is always available for transmission, and this number should be zero.
  - No Carrier – number of packets sent when carrier sense was unavailable
- Collisions – analysis of the collisions that occurred
  - Single – packets sent after one collision
  - Multiple – packets sent after multiple collisions
  - Late – packets aborted during sending because of collisions after 64 bytes
  - Excessive – packets not sent because of too many collisions

- Example

```
host1:vr2#show interfaces gigabitEthernet2/0
gigabitEthernet2/0 is Up, Administrative status is Up
  Hardware is SEEQ 8101, address is 0090.1a00.0038
  MTU: Operational 1518, Administrative 1518
  Duplex Mode: Operational Full Duplex, Administrative Auto Negotiate
  Speed: Operational 1000 Mbps, Administrative Auto Negotiate

  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec

  In: Bytes 448, Unicast 0
   Multicast 0, Broadcast 7
   Errors 0, Discards 0, Mac Errors 0, Alignment 0
   CRC 0, Too Longs 0, Symbol Errors 0
  Out: Bytes 384, Unicast 0
   Multicast 0, Broadcast 6
   Errors 0, Discards 0, Mac Errors 0, Deferred 0, No Carrier 0
   Collisions: Single 0, Multiple 0, Late 0, Excessive 0
```

show ip mac-validate interface

- Use to display the status of the MAC address validation on the physical interface.
- Field descriptions
  - › *interfaceSpecifier* – Fast Ethernet or Gigabit Ethernet interface slot/port
  - › Keyword assigned to interface – options: Strict or Loose
  - › Address – IP address of the entry
  - › Hardware Addr – physical (MAC) address of the entry
- Example

```
host1:boston#show ip mac-validate interface fastEthernet 11/0
FastEthernet11/0:  Strict

          Address     Hardware Addr
          3.3.3.3     0090.1a30.3365
          4.4.4.4     0090.1a30.3368
```

show vlan subinterfaces

- Use to display the status of VLAN subinterfaces on the FE or GE module you specify.
- Field descriptions
  - Subinterface number – location of the subinterface that carries the VLAN traffic
  - S-VLAN ID – domain number of the stacked VLAN
  - VLAN ID – domain number of the VLAN
- Examples

```
host1#show vlan sub
  gigabitEthernet2/1.1, S-VLAN ID is 10, VLAN ID is 201
  gigabitEthernet2/2.1, S-VLAN ID is 10, VLAN ID is 202
  gigabitEthernet3/0.1, VLAN ID is 16

host1#show vlan sub fast 2/0.1
  S-VLAN ID is 35, VLAN ID is 301
```