BCN

Building Core Networks with OSPF, IS-IS, BGP, and MPLS Bootcamp v6.1a

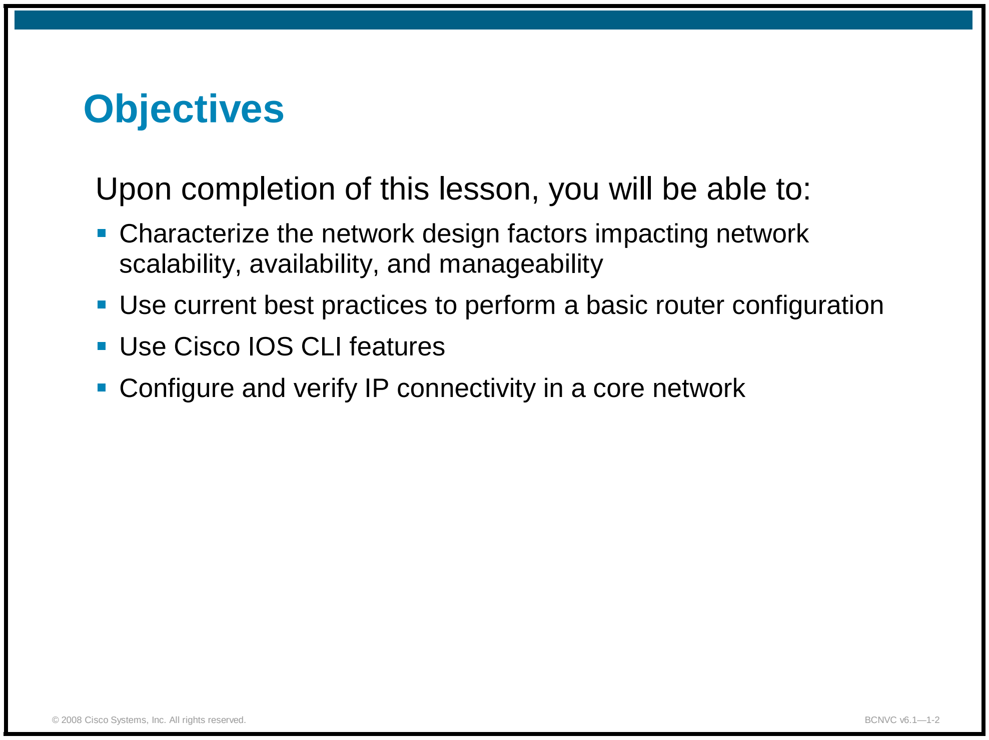
Overview

This is a sample of course material from *Advanced Services' Building Core Networks with OSPF, IS-IS, BGP, and MPLS Bootcamp*, version 6.1a. To register for this course or to learn more about Advanced Services offerings, visit our website at: www.cisco.com/go/ase

Lesson 1

Get Started with Core Networking

Objectives

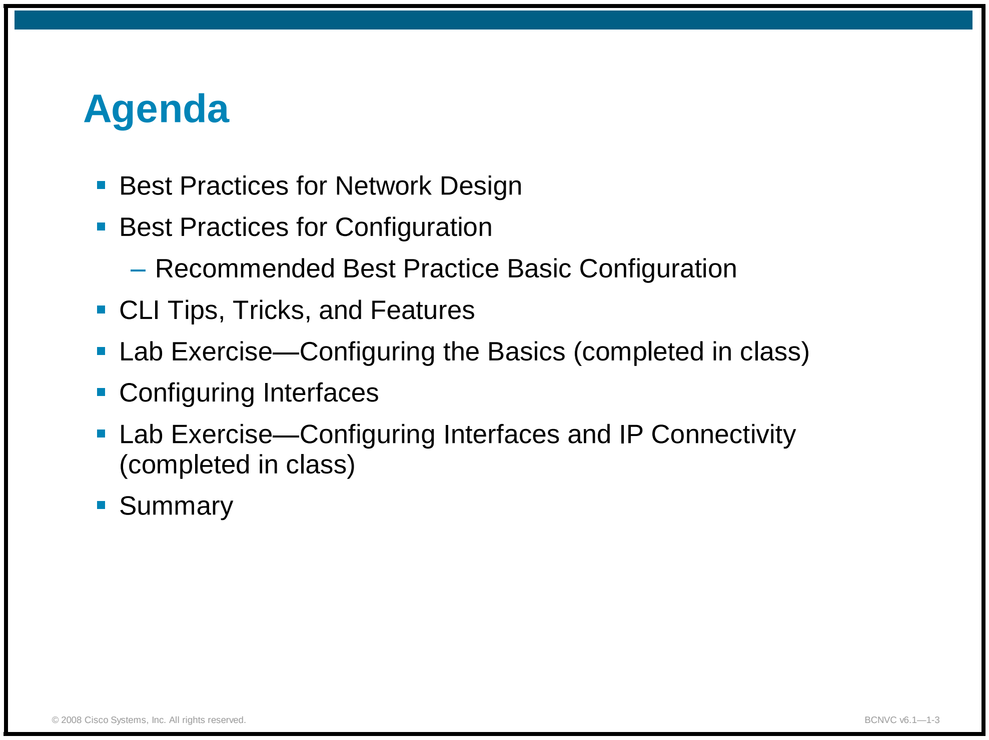


Objectives

Upon completion of this lesson, you will be able to:

- Characterize the network design factors impacting network scalability, availability, and manageability
- Use current best practices to perform a basic router configuration
- Use Cisco IOS CLI features
- Configure and verify IP connectivity in a core network

Agenda



Agenda

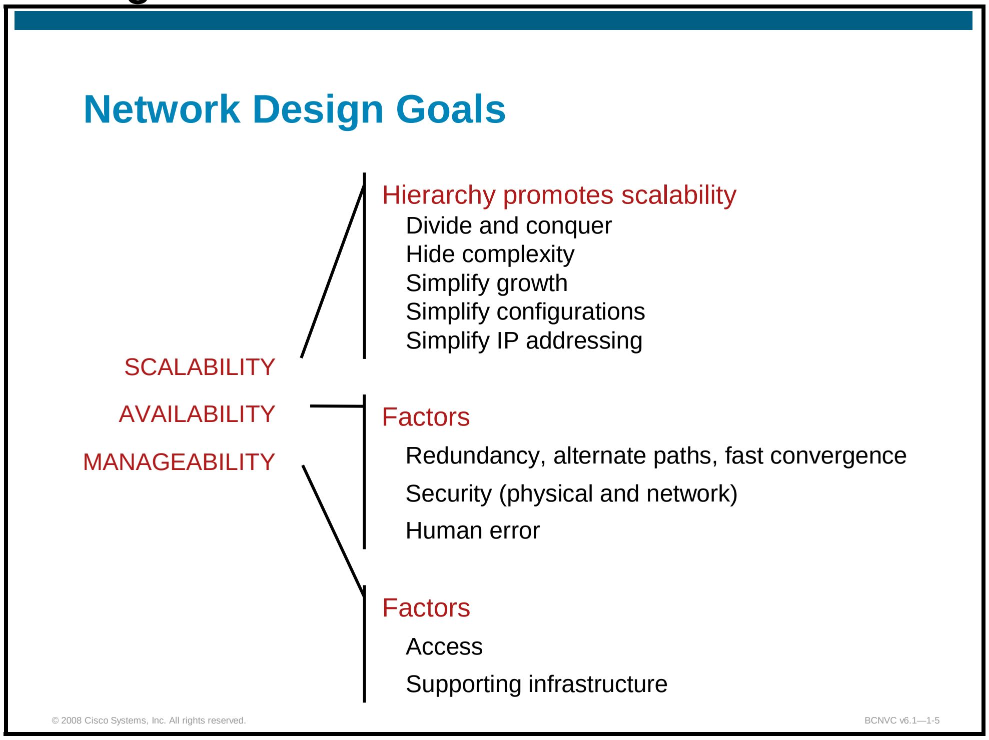
- Best Practices for Network Design
- Best Practices for Configuration
  - Recommended Best Practice Basic Configuration
- CLI Tips, Tricks, and Features
- Lab Exercise—Configuring the Basics (completed in class)
- Configuring Interfaces
- Lab Exercise—Configuring Interfaces and IP Connectivity (completed in class)
- Summary

BCNVC v6.1—1-3

Best Practices for Network Design

This topic covers the best practices for network design. It covers network design goals and hierarchical network design.

Network Design Goals



Network Design Goals

SCALABILITY

AVAILABILITY

MANAGEABILITY

Hierarchy promotes scalability
- Divide and conquer
- Hide complexity
- Simplify growth
- Simplify configurations
- Simplify IP addressing

Factors
- Redundancy, alternate paths, fast convergence
- Security (physical and network)
- Human error

Factors
- Access
- Supporting infrastructure

BCNVC v6.1—1-5

The three goals of a network are high scalability, availability, and manageability.

Scalability is how well a network can grow to meet increasing demand. Hierarchy allows you to hide complexity in one part of a network from other parts and simplifies operating and managing a network.

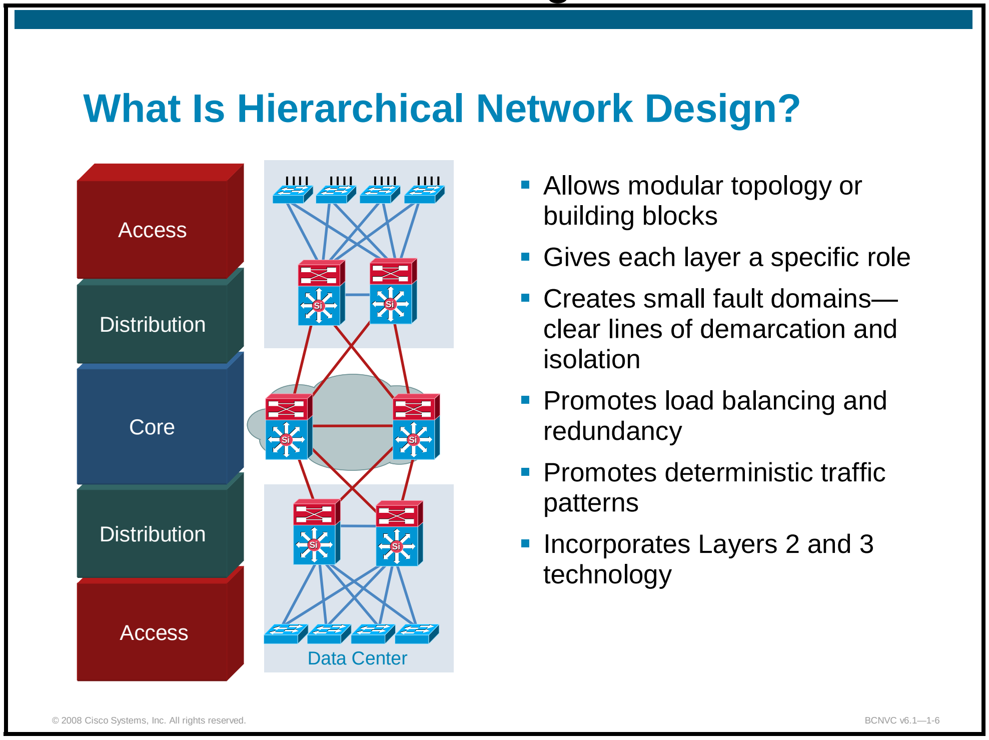
Availability is the percentage of time a network is available to perform its mission. While hardware and network redundancy is the most common way to achieve availability, other factors are of equal or even greater importance. Network security plays a major role in availability. If a hostile entity invades your building or your network, it may cause major damage to your network. However, one of the single largest contributors to network downtime is human error. A hierarchical design helps guard against error because simple configurations are easier to understand and troubleshoot.

Manageability is how easy it is to access, operate, control, support, and maintain a network. Major manageability factors are methods of access, types of supporting infrastructure, tools, and supporting processes.

A hierarchical and modular network design facilitates all of these network design goals or allows them to be met. Modular design makes a network easy to scale, understand, and troubleshoot by promoting deterministic traffic patterns.

This course reiterates these design goals as each topic is addressed.

What Is Hierarchical Network Design?



What Is Hierarchical Network Design?

- Allows modular topology or building blocks
- Gives each layer a specific role
- Creates small fault domains—clear lines of demarcation and isolation
- Promotes load balancing and redundancy
- Promotes deterministic traffic patterns
- Incorporates Layers 2 and 3 technology

BCNVC v6.1—1-6

Cisco introduced the hierarchical design model, which uses a layered approach to network design, in 1999. The building-block components are the access layer, distribution layer, and core (backbone) layer. The principal advantages of this model are its hierarchical structure and modularity.

Hierarchical network design allows you to build a modular, deterministic, and scalable foundation.

The design model is easy to scale, understand, and troubleshoot. The model takes a layered approach to building a network. The layers are access, distribution, and core, and each layer serves its own purpose.

Access layer, server farm, WAN, Internet, and public-switched telephone network (PSTN) are all “modules” that “plug in” as building blocks in this model.

The access layer provides entry into the network for end stations (PCs, phones, and printers) or attached networks (routers and switches). It is connected to two separate distribution-layer devices for redundancy. If the connection between the distribution-layer switches is a Layer 3 connection, then there are no loops and all uplinks actively forward traffic.

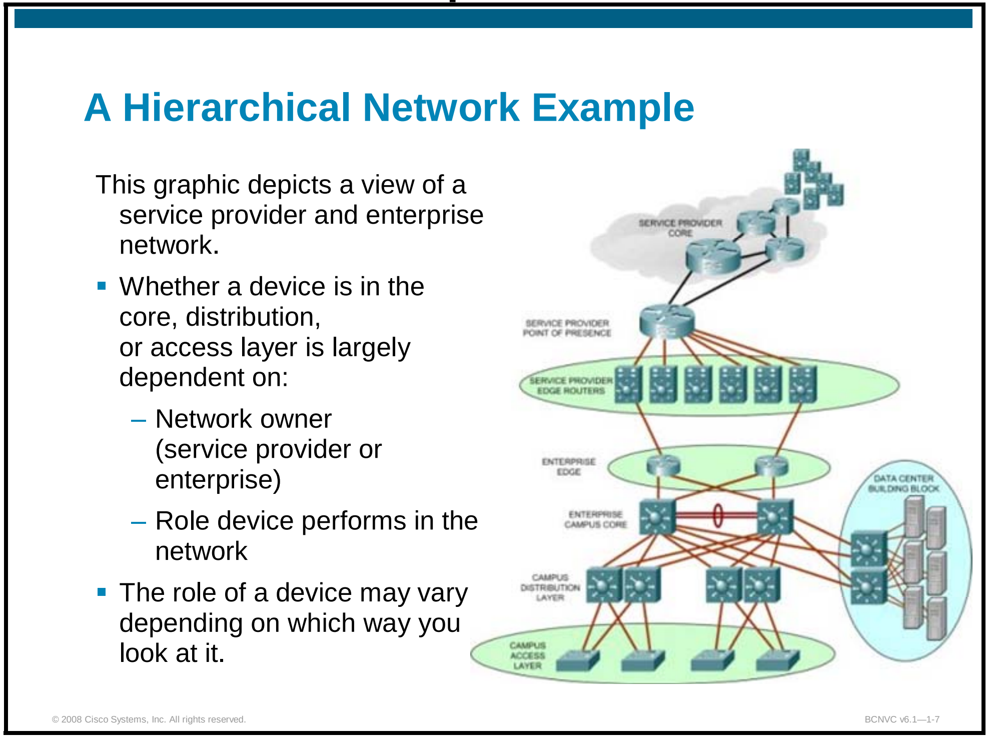
The distribution layer aggregates nodes from the access layer, protecting the core from high-density peering. The distribution layer creates a fault boundary providing a logical isolation point in the event of a failure originating in the access layer. Route summarization, load balancing, quality of service (QoS), and ease of provisioning are key considerations for the distribution layer.

In a typical hierarchical model, individual building blocks are interconnected using a core layer. The core serves as the backbone for the network. The core must be fast and extremely resilient because every building-block depends on it for connectivity. Current hardware-accelerated systems have the potential to deliver complex services at wire speed. However, in the core, a “less is more” approach is taken. A minimal configuration in the core reduces configuration complexity, limiting the possibility of operational error.

Although it is possible to achieve redundancy with a fully-meshed or highly-meshed topology, it does not provide consistent convergence if a link or node fails. Also, peering and adjacency issues exist with a fully meshed design, making routing complex to configure and difficult to scale. The high port-count adds unnecessary cost and increases complexity as a network grows or changes. Following are other key design issues:

- Design the core layer as a high-speed Layer 3 switching environment using only hardware-accelerated services. Layer 3 core designs are superior to Layer 2 and other alternatives because they provide:
  - Faster convergence around a link or node failure
  - Increased scalability because neighbor relationships and meshing are reduced
  - More efficient bandwidth utilization
- Use redundant point-to-point Layer 3 interconnections in the core (triangles, not squares) wherever possible, because this design yields the fastest and most deterministic convergence results.

A Hierarchical Network Example



A Hierarchical Network Example

This graphic depicts a view of a service provider and enterprise network.

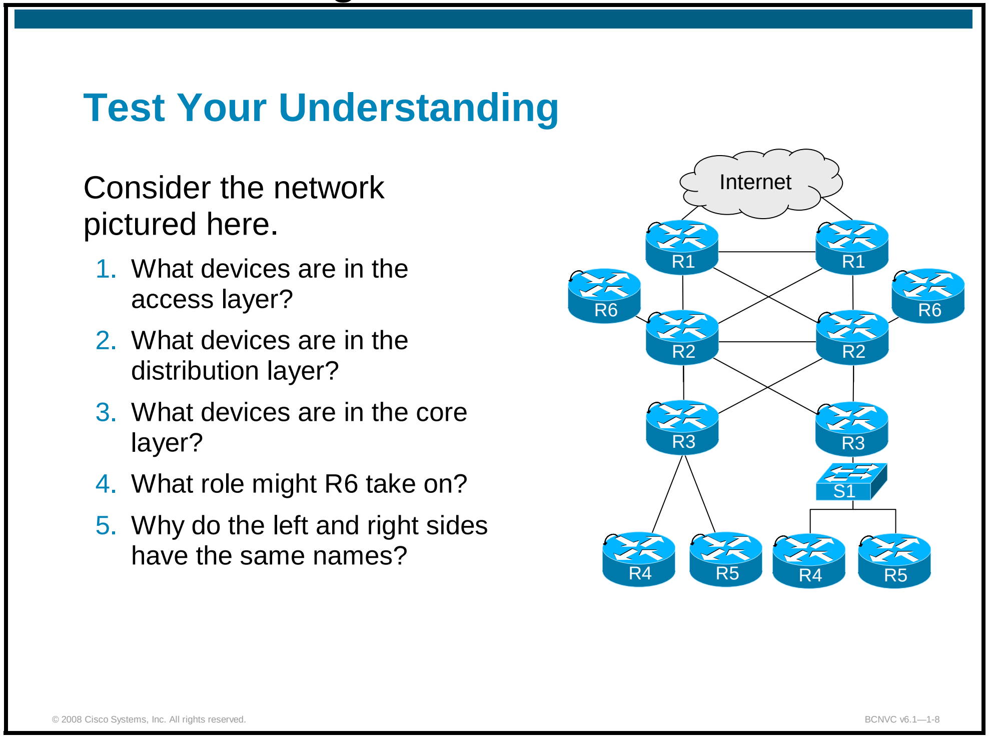
- Whether a device is in the core, distribution, or access layer is largely dependent on:
  - Network owner (service provider or enterprise)
  - Role device performs in the network
- The role of a device may vary depending on which way you look at it.

SERVICE PROVIDER CORE
SERVICE PROVIDER POINT OF PRESENCE
SERVICE PROVIDER EDGE ROUTERS
ENTERPRISE EDGE
ENTERPRISE CAMPUS CORE
CAMPUS DISTRIBUTION LAYER
CAMPUS ACCESS LAYER
DATA CENTER BUILDING BLOCK

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—1-7

This figure shows a service provider and enterprise network. Whether a device in this network is core, distribution, or access depends on two things: the network owner (service provider or enterprise), and the role of the device within the network. The role of a device may vary depending on your perspective.

Test Your Understanding



Test Your Understanding

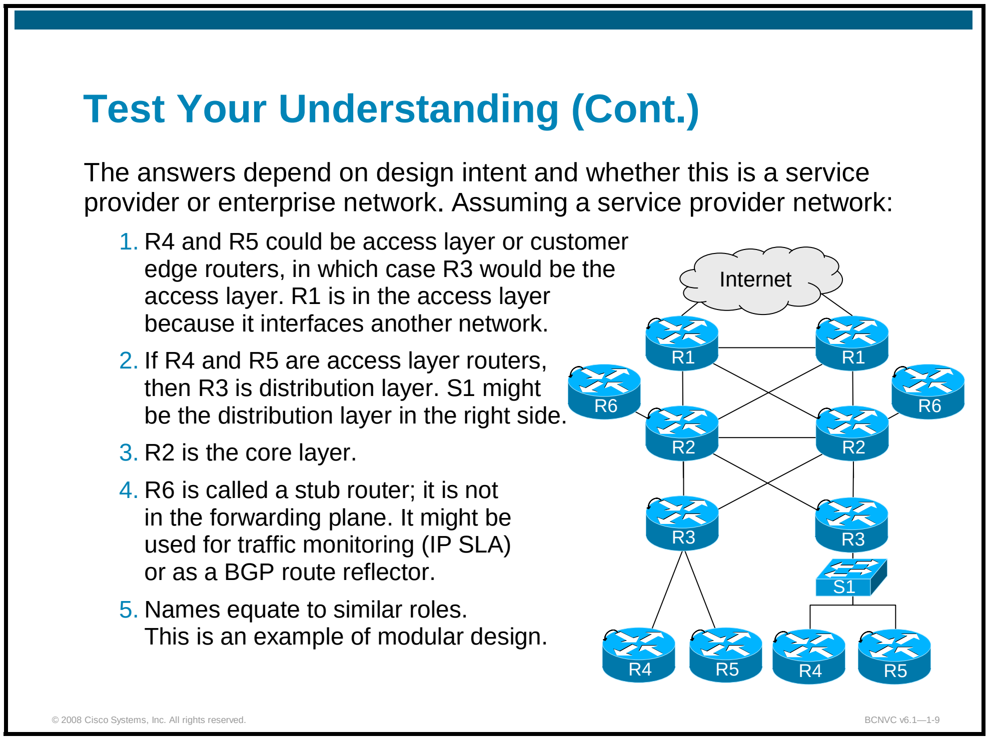
Consider the network pictured here.

1. What devices are in the access layer?
2. What devices are in the distribution layer?
3. What devices are in the core layer?
4. What role might R6 take on?
5. Why do the left and right sides have the same names?

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—1-8

Take some time now to do a quick self-test. Refer to the network in the figure above and answer the questions.

Depending on the perspective, there may be more than one correct answer to the questions.



Test Your Understanding (Cont.)

The answers depend on design intent and whether this is a service provider or enterprise network. Assuming a service provider network:

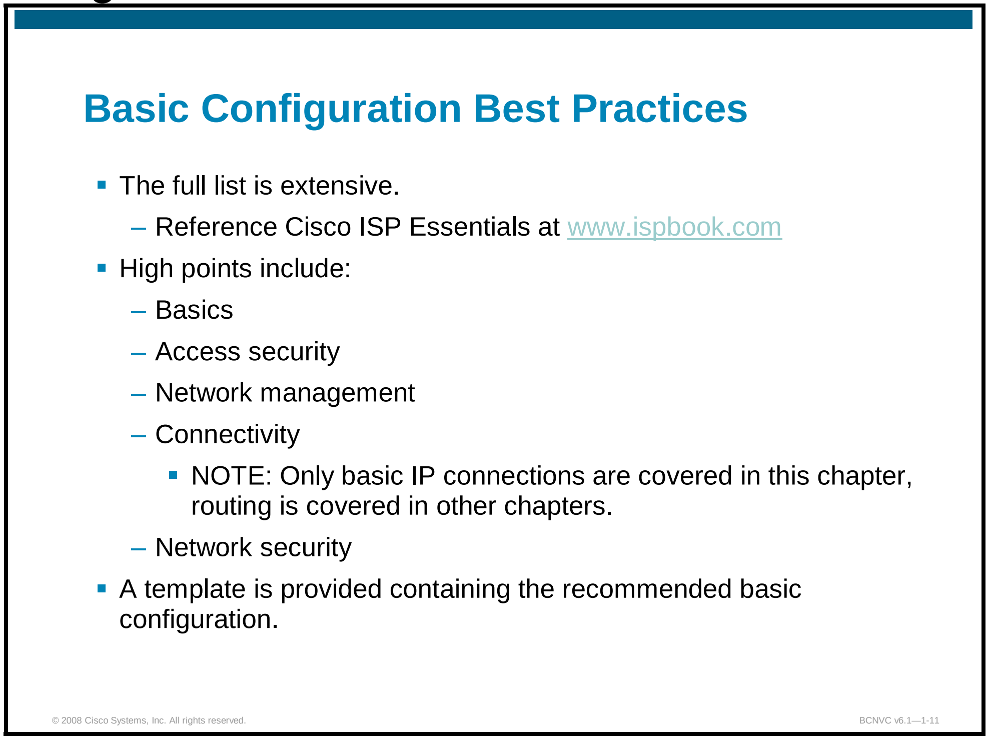
1. R4 and R5 could be access layer or customer edge routers, in which case R3 would be the access layer. R1 is in the access layer because it interfaces another network.
2. If R4 and R5 are access layer routers, then R3 is distribution layer. S1 might be the distribution layer in the right side.
3. R2 is the core layer.
4. R6 is called a stub router; it is not in the forwarding plane. It might be used for traffic monitoring (IP SLA) or as a BGP route reflector.
5. Names equate to similar roles. This is an example of modular design.

BCNVC v6.1—1-9

Best Practices for Configuration

This topic covers best practices to follow when configuring networking devices in a core network. Typically in an Internet service provider (ISP) setting, you configure large numbers of devices. A set of best practices makes this job easier, faster, and less error-prone, while complying with company policy. This topic covers the steps in configuring routers, user authentication and security, use of banners, using an SMTP server, logging, Cisco Express Forwarding (CEF), configuration management, and using general system templates.

Basic Configuration Best Practices



Basic Configuration Best Practices

- The full list is extensive.
  - Reference Cisco ISP Essentials at www.ispbook.com
- High points include:
  - Basics
  - Access security
  - Network management
  - Connectivity
    - NOTE: Only basic IP connections are covered in this chapter, routing is covered in other chapters.
  - Network security
- A template is provided containing the recommended basic configuration.

BCNVC v6.1—1-11

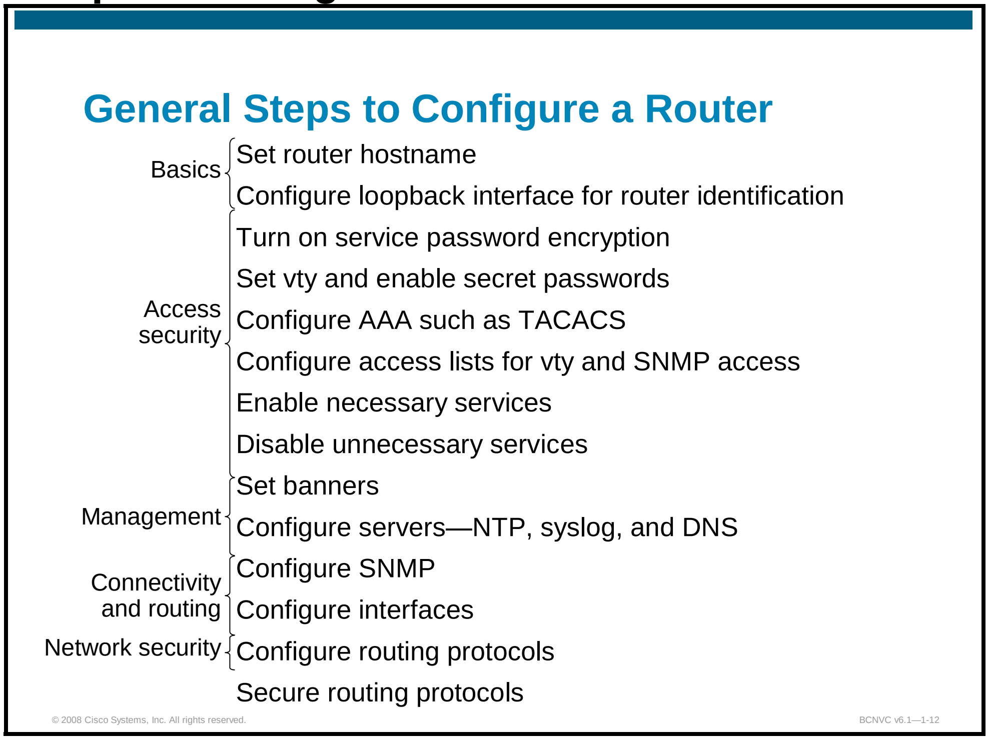
Throughout this course you will be provided examples and tips from the book Cisco ISP Essentials available from Cisco Press:

(http://www.ciscopress.com/bookstore/product.asp?isbn=1587050412)

Cisco ISP Essentials highlights many of the key Cisco IOS features in everyday use in the major ISP backbones of the world to help new network engineers gain understanding of the power of Cisco IOS software and the richness of features available specifically for them. Cisco ISP Essentials also provides a detailed technical reference for the expert ISP engineer, with descriptions of the various controls and special features that have been specifically designed for ISPs. The configuration examples and diagrams describe many scenarios, ranging from good operational practices to network security. Finally, an appendix explains the best principles to use when configuring a router in a small ISP point of presence (POP).

Note While previous versions of this book can be found as PDF files on the web (search for "ISP Essentials"), the most recent version is from 2002. The latest version from Cisco Press contains up-to-date information.

General Steps to Configure a Router



General Steps to Configure a Router

Basics
- Set router hostname
- Configure loopback interface for router identification

Access security
- Turn on service password encryption
- Set vty and enable secret passwords
- Configure AAA such as TACACS
- Configure access lists for vty and SNMP access
- Enable necessary services
- Disable unnecessary services

Management
- Set banners
- Configure servers—NTP, syslog, and DNS
- Configure SNMP

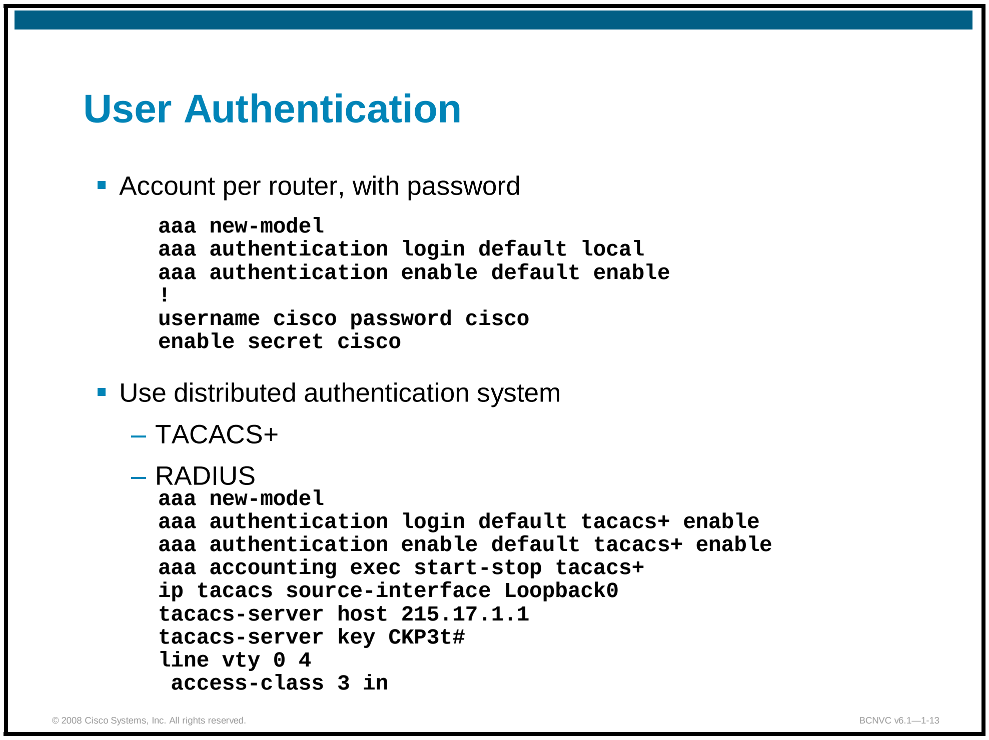
Connectivity and routing
- Configure interfaces
- Configure routing protocols

Network security
- Secure routing protocols

BCNVC v6.1—1-12

The steps listed above cover the basic configuration of a router. While the precise order does not need to be followed, this sequence prevents you from referencing something that may not have been previously defined. The information in this lesson is presented in this sequence. Rather than cover all of this material here, this material is presented in more detail throughout the rest of this lesson.

User Authentication



User Authentication

- Account per router, with password

```
aaa new-model
aaa authentication login default local
aaa authentication enable default enable
!
username cisco password cisco
enable secret cisco
```

- Use distributed authentication system
  - TACACS+
  - RADIUS

```
aaa new-model
aaa authentication login default tacacs+ enable
aaa authentication enable default tacacs+ enable
aaa accounting exec start-stop tacacs+
ip tacacs source-interface Loopback0
tacacs-server host 215.17.1.1
tacacs-server key CKP3t#
line vty 0 4
 access-class 3 in
```

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—1-13

The series of commands shown above tells a router to look locally for a standard user login and to locally configure **enable secret** for the enable login. By default, the login is enabled on all virtual terminals (vtys) so other teams can gain access.

Caution Use extreme care when entering the **username** *value* **password** *value* command. If you include the "7," it is expected that the password value you type in is an encryption code. It is unchanged by the encryption process. However, when you log in afterwards, the password you enter is encrypted, the two values do not match, and you are not able to log in. A password recovery procedure is needed and all existing configurations are lost.

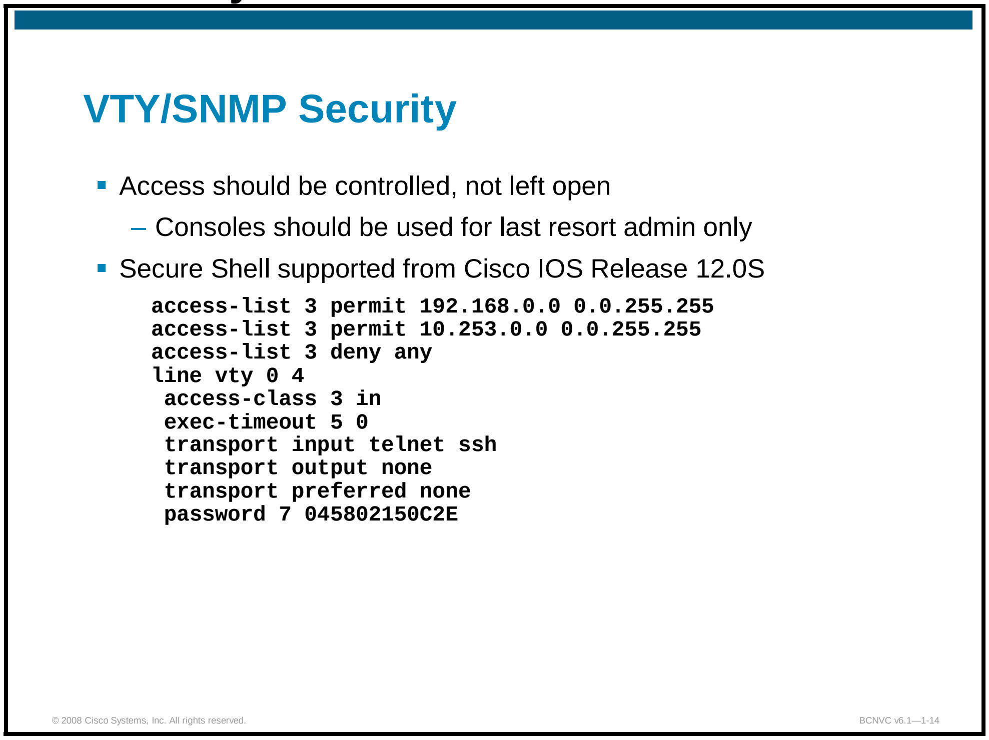
Most ISPs use TACACS+ or Radius for user authentication. Very few define accounts on the router itself as this offers more opportunity for the system to be compromised. A well-protected TACACS+ server accessed only from the router’s loopback interface address block offers more security of user and enable accounts. A sample configuration for standard and enable passwords are:

```
aaa new-model
aaa authentication login default tacacs+ enable
aaa authentication enable default tacacs+ enable
aaa accounting exec start-stop tacacs+
!
ip tacacs source-interface Loopback0
tacacs-server host 215.17.1.2
tacacs-server host 215.17.34.10
tacacs-server key CKr3t#
!
```

When using Remote Access Dial-In User Service (RADIUS), either for administrative access to the router, or for dial-in user authentication and accounting, the router configuration to support loopback interfaces as the source address for RADIUS packets originating from the router looks like this:

```
radius-server host 215.17.1.2 auth-port 1645 acct-port 1646
radius-server host 215.17.34.10 auth-port 1645 acct-port 1646
ip radius source-interface Loopback0
!
```

VTY/SNMP Security



VTY/SNMP Security

- Access should be controlled, not left open
  - Consoles should be used for last resort admin only
- Secure Shell supported from Cisco IOS Release 12.0S

```
access-list 3 permit 192.168.0.0 0.0.255.255
access-list 3 permit 10.253.0.0 0.0.255.255
access-list 3 deny any
line vty 0 4
 access-class 3 in
 exec-timeout 5 0
 transport input telnet ssh
 transport output none
 transport preferred none
 password 7 045802150C2E
```

You can do several things to improve VTY security. Use both a username and password rather than the traditional method of only using a VTY password. Use access control lists to restrict Telnet connections. Use shorter timeouts. Or, use Secure Shell instead of Telnet.

Before 12.0S software, the only method really used to access the VTY ports was Telnet. Rlogin has been used by some ISPs, especially for executing one-off commands but the protocol is insecure and cannot be recommended. SSH support has since been added, giving ISPs greater flexibility and some security when accessing their equipment across the Internet.

Before you can configure SSH, the router needs to be running a cryptographic image that supports SSH. Images are freely available for download.

Once an appropriate cryptographic image is running, SSH needs to be setup on the router. The following sequence of configuration commands gives an example of how this may be achieved:

```
P1R1(config)#crypto key generate rsa
```

and select a key size of at least 1024 bits. After this, add **ssh** as the input transport on the VTYs:

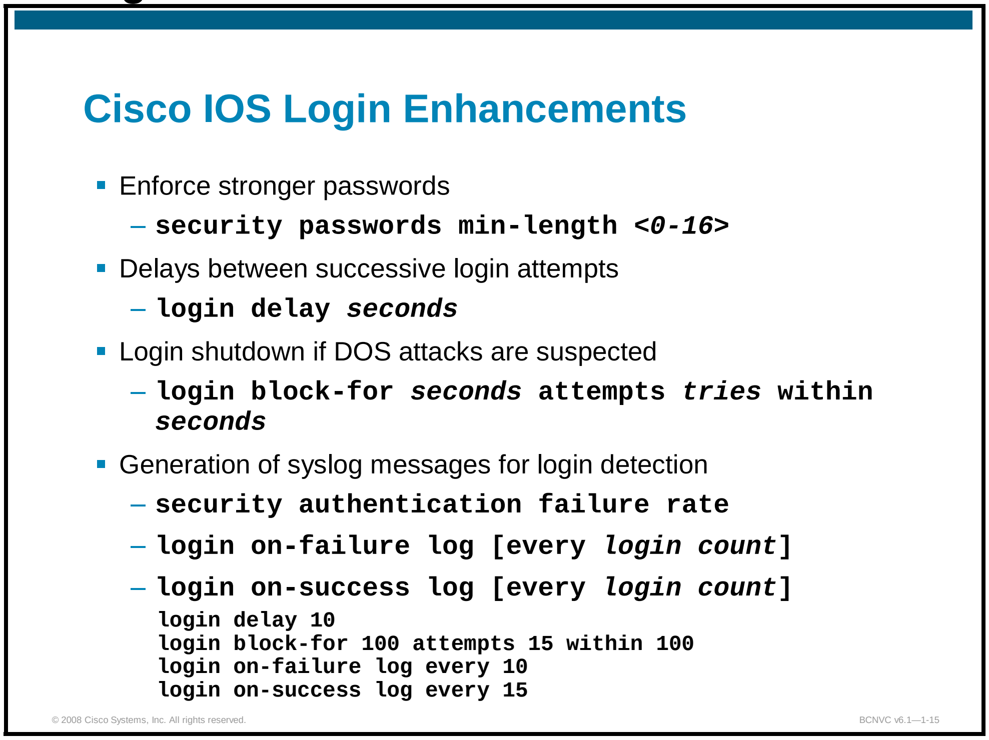
```
line vty 0 4
transport input ssh
```

It is now possible to use SSH to access the router.

Note A username and password pair must be configured on the router before SSH access works. However, it is strongly recommended that AAA is used to authenticate users as this is the preferred way of securing the router.

You can do several things to improve SNMP security. Change the default SNMP communities, and use difficult or hard-to-guess passwords. Protect the SNMP community with an access list. Use an SNMP failure trap to alert administrators of a possible SNMP intrusion.

Cisco IOS Login Enhancements



Cisco IOS Login Enhancements

- Enforce stronger passwords
  - **security passwords min-length <*0-16*>**
- Delays between successive login attempts
  - **login delay *seconds***
- Login shutdown if DOS attacks are suspected
  - **login block-for *seconds* attempts *tries* within *seconds***
- Generation of syslog messages for login detection
  - **security authentication failure rate**
  - **login on-failure log [every *login count*]**
  - **login on-success log [every *login count*]**

```
login delay 10
login block-for 100 attempts 15 within 100
login on-failure log every 10
login on-success log every 15
```

BCNVC v6.1—1-15

The Cisco IOS login enhancements feature allows you to better secure Cisco IOS devices when creating a virtual connection, such as Telnet, Secure Shell (SSH), or HTTP. Thus, users can slow down dictionary attacks and help protect their router from a possible denial-of-service (DoS) attack.

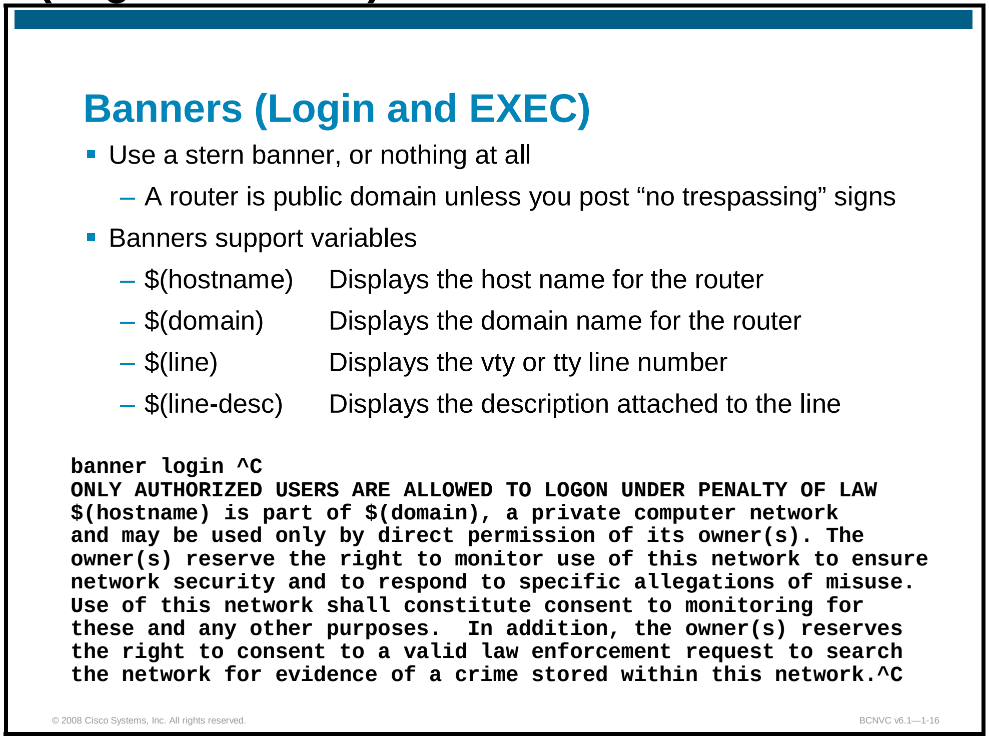
The **security passwords min-length** command provides enhanced security access to a router by allowing you to specify a minimum password length, eliminating common passwords that are prevalent on most networks, such as "lab" and "cisco." This command affects user passwords, enable passwords and secrets, and line passwords. After this command is enabled, any password that is less than the specified length fails.

A Cisco IOS device can accept virtual connections as fast as it can process them. Introducing a delay between login attempts helps to protect your router from a possible dictionary attack, which attempts to gain access to your username and password information. Delays can be enabled in one of the following ways:

- With the new global configuration mode command **login delay**, which allows you to specify a specific number of seconds.
- With the **login block-for** command. You must enter this command before issuing the **login delay** command; however, if you enter only **login block-for**, a login delay of 1 second is automatically enforced.

The **security authentication failure rate** command provides enhanced security access to a router by generating syslog messages after the number of unsuccessful login attempts exceeds the configured threshold rate. This command ensures that there are no continuous failures to access the router.

Banners (Login and Exec)



Banners (Login and EXEC)

- Use a stern banner, or nothing at all
  - A router is public domain unless you post “no trespassing” signs
- Banners support variables

– $(hostname)	Displays the host name for the router
– $(domain)	Displays the domain name for the router
– $(line)	Displays the vty or tty line number
– $(line-desc)	Displays the description attached to the line

```
banner login ^C
ONLY AUTHORIZED USERS ARE ALLOWED TO LOGON UNDER PENALTY OF LAW
$(hostname) is part of $(domain), a private computer network
and may be used only by direct permission of its owner(s). The
owner(s) reserve the right to monitor use of this network to ensure
network security and to respond to specific allegations of misuse.
Use of this network shall constitute consent to monitoring for
these and any other purposes.  In addition, the owner(s) reserves
the right to consent to a valid law enforcement request to search
the network for evidence of a crime stored within this network.^C
```

BCNVC v6.1—1-16

As a general practice, banners should contain the following information and warnings:

- "Only authorized personnel may gain access."
- "System logs are being maintained and could be used as evidence in criminal and/or civil court."
- "Unauthorized access is unlawful and subject to civil and/or criminal penalties."

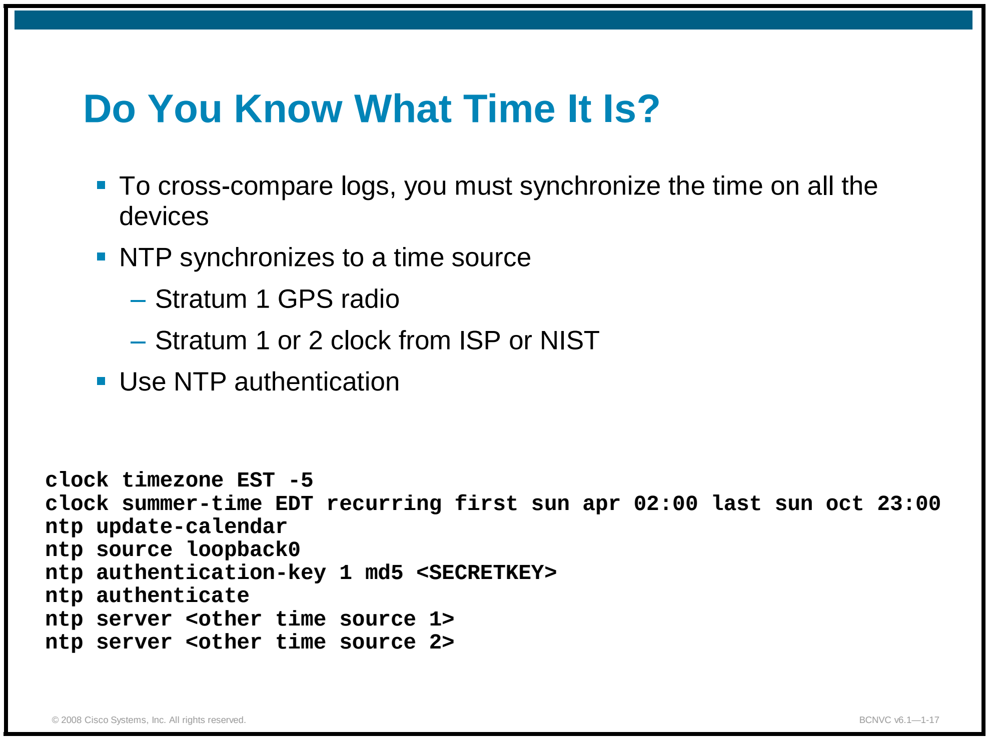
Be sure banners comply with corporate policies. Consider having banners reviewed by corporate legal counsel.

Do not put the following information in a banner:

- The company name or physical device location
- The word "welcome"

Different banner messages may be used in different network locations. Border routers may use a message such as the message shown in the figure. Internal routers may include warnings regarding disciplinary actions in addition to, or instead of, criminal or civil actions.

Do You Know What Time It Is?



Do You Know What Time It Is?

- To cross-compare logs, you must synchronize the time on all the devices
- NTP synchronizes to a time source
  - Stratum 1 GPS radio
  - Stratum 1 or 2 clock from ISP or NIST
- Use NTP authentication

```
clock timezone EST -5
clock summer-time EDT recurring first sun apr 02:00 last sun oct 23:00
ntp update-calendar
ntp source loopback0
ntp authentication-key 1 md5 <SECRETKEY>
ntp authenticate
ntp server <other time source 1>
ntp server <other time source 2>
```

BCNVC v6.1—1-17

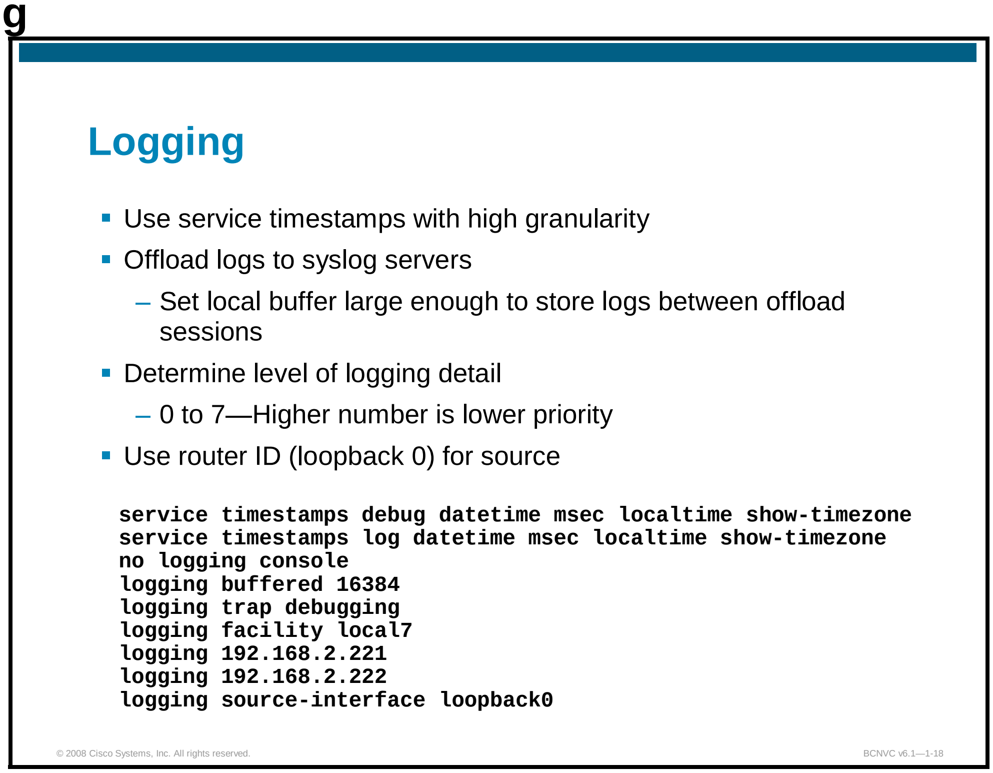
If an interface does not need to receive Network Time Protocol (NTP) packets, disable the function with the **ntp disable** command.

Time synchronization across a network is critical. Without a mechanism to ensure that all devices in a network are synchronized to exactly the same time source, functions such as accounting, event logging, fault analysis, security incident response, and network management would not be possible on more than one network device. Whenever a system or network engineer needs to compare two logs from two different systems, each system needs a frame of reference to match the logs. That frame of reference is synchronized time.

An NTP network usually gets its time from an authoritative time source, such as a radio clock, global positioning system (GPS) device, or atomic clock attached to a time server. NTP then distributes this time across the network. NTP is hierarchical, with different time servers maintaining authority levels. The highest authority is Stratum 1. Levels of authority descend from 2 to a maximum of 16. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two machines to within a millisecond of one another.

The time kept on a machine is a critical resource, so we strongly recommend that you use the security features of NTP to avoid the accidental or malicious setting of an incorrect time. Two mechanisms are available: an access-list–based restriction scheme and an encrypted authentication mechanism. The above example highlights NTP security using encrypted authentication.

Logging



Logging

- Use service timestamps with high granularity
- Offload logs to syslog servers
  - Set local buffer large enough to store logs between offload sessions
- Determine level of logging detail
  - 0 to 7—Higher number is lower priority
- Use router ID (loopback 0) for source

```
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
no logging console
logging buffered 16384
logging trap debugging
logging facility local7
logging 192.168.2.221
logging 192.168.2.222
logging source-interface loopback0
```

Keeping logs is a common and accepted operational practice. Interface status, security alerts, environmental conditions, CPU process hog, and many other events on a router can be captured and analyzed with UNIX syslog. NTP synchronization is vital to logging and determining when security incidents occur. It is desirable to have logging to the informational level for good granularity of events. By default, log messages are not time-stamped. If routers are configured for UNIX logging, you should want detailed time stamps for each log entry:

service timestamps message-type datetime [msec] [localtime] [show-timezone]

The command-line options in the **timestamps** command are as follows:

debug: All debug information is time-stamped.

log: All log information is time-stamped.

datetime: The date and time are included in the syslog message.

localtime: The local time (instead of UTC) is used in the log message.

show-timezone: The time zone defined on the router is included. This is useful if a network crosses multiple time zones.

msec: Time accuracy is expressed as milliseconds, which is useful if NTP is configured.

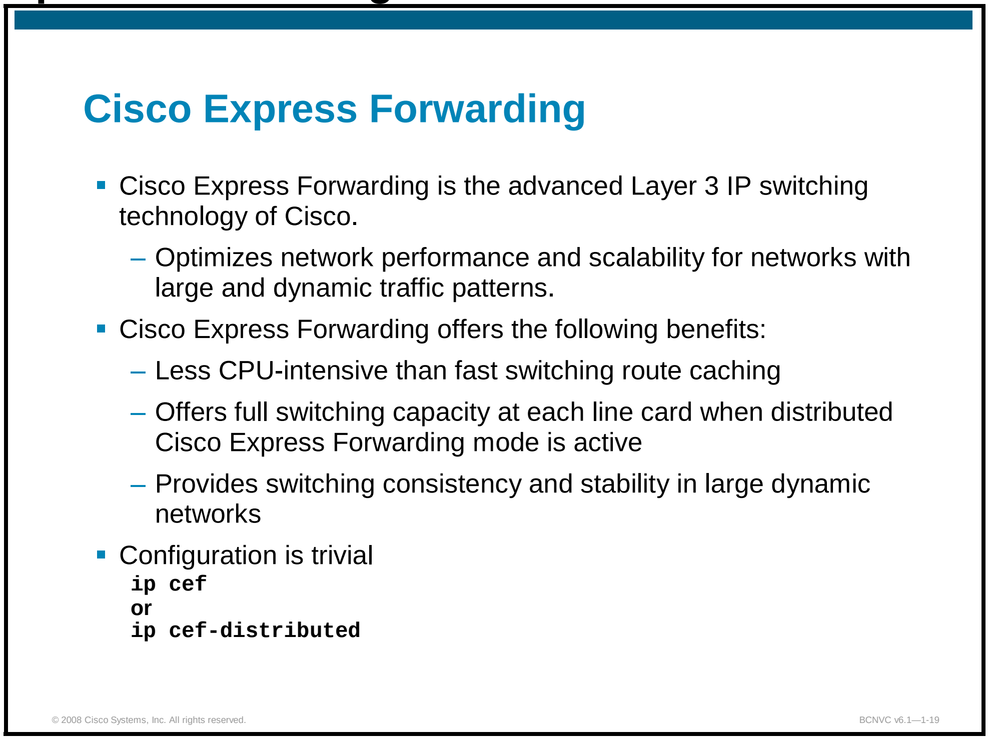
logging source-interface loopback0

By default, a syslog message contains the IP address of the interface it uses to leave the router. You can require all syslog messages to contain the same IP address, regardless of which interface they use.

no logging console

Sometimes logging generates a tremendous amount of traffic on the console port. It is good practice to turn off console logging to keep the console port free for maintenance.

Cisco Express Forwarding



Cisco Express Forwarding

- Cisco Express Forwarding is the advanced Layer 3 IP switching technology of Cisco.
  - Optimizes network performance and scalability for networks with large and dynamic traffic patterns.
- Cisco Express Forwarding offers the following benefits:
  - Less CPU-intensive than fast switching route caching
  - Offers full switching capacity at each line card when distributed Cisco Express Forwarding mode is active
  - Provides switching consistency and stability in large dynamic networks
- Configuration is trivial

```
ip cef
or
ip cef-distributed
```

Cisco Express Forwarding (CEF) is the recommended forwarding and switching path for Cisco routers. CEF increases performance, scalability, and resilience, and enables new functionality over the older optimum switching.

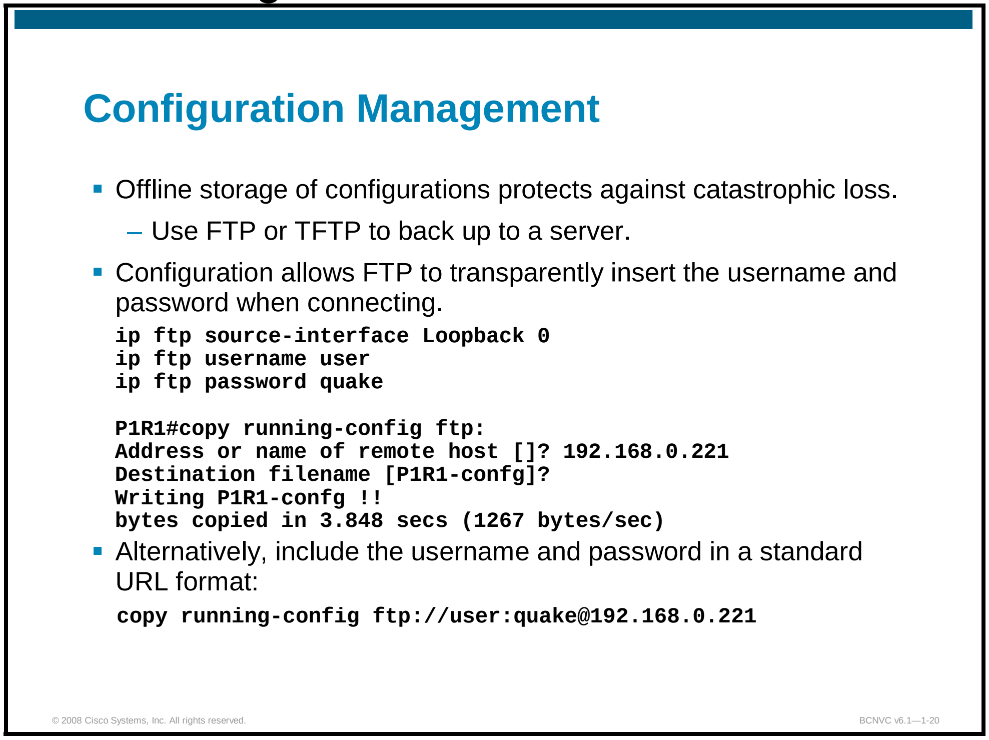
Implementation is simple with either of the following commands (depending on the platform):

ip cef

ip cef-distributed

The key issue is ensuring that Cisco Express Forwarding is turned on. .On most Cisco platforms running newer versions of Cisco IOS Software CEF is enabled by default. Even so, it only takes a second to type in the command and it ensures that CEF is on in your platform. It is a good idea to include these commands in any configuration templates you use as well.

Configuration Management



Configuration Management

- Offline storage of configurations protects against catastrophic loss.
  - Use FTP or TFTP to back up to a server.
- Configuration allows FTP to transparently insert the username and password when connecting.

```
ip ftp source-interface Loopback 0
ip ftp username user
ip ftp password quake

P1R1#copy running-config ftp:
Address or name of remote host []? 192.168.0.221
Destination filename [P1R1-confg]?
Writing P1R1-confg !!
bytes copied in 3.848 secs (1267 bytes/sec)
```

- Alternatively, include the username and password in a standard URL format:

```
copy running-config ftp://user:quake@192.168.0.221
```

BCNVC v6.1—1-20

Trivial File Transfer Protocol (TFTP) is the most common tool for uploading and downloading configurations. The TFTP server security is critical, which means you should always use security tools with IP source addresses. Cisco IOS Software allows TFTP to be configured to use specific IP interface addresses. This allows a fixed ACL on the TFTP server, based on a fixed address on the router (for example, the loopback interface).

ip tftp source-interface Loopback0

Since Cisco IOS Software Release 12.0, File Transfer Protocol (FTP) also can be used to copy configurations to an FTP server. This provides more security because an FTP server requires a username and password. Cisco IOS Software has two ways to provide the username and password to the FTP client.

The first puts the username and password as part of the Cisco IOS Software configuration. With service password-encryption turned on, the FTP password is stored with encryption type 7:

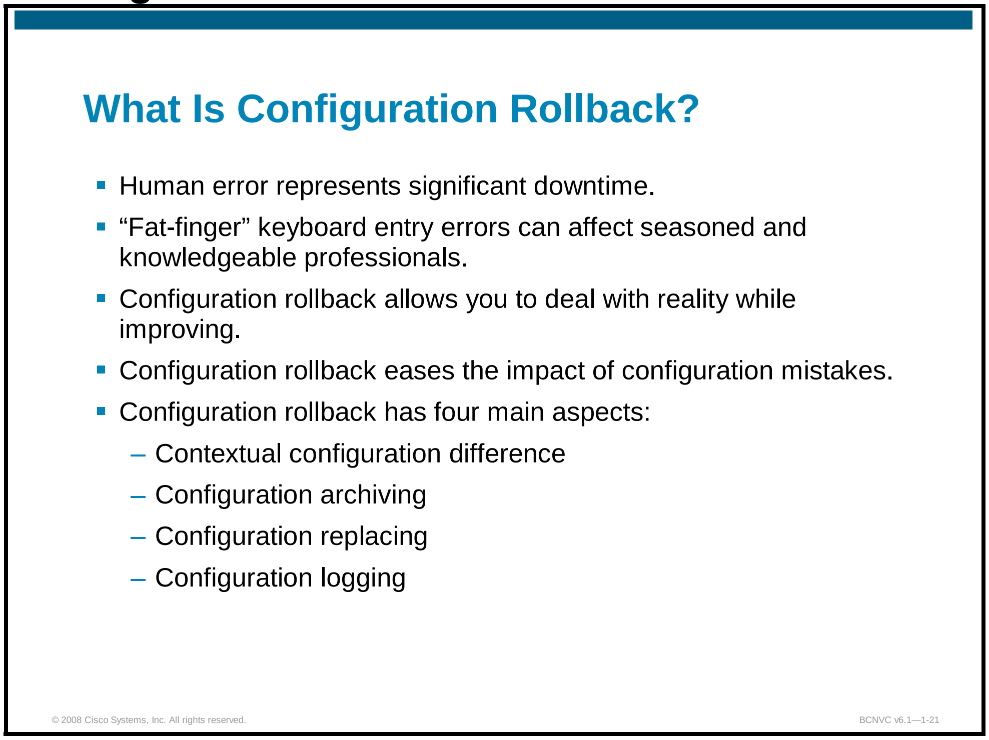
ip ftp source-interface Loopback 0

ip ftp username user

ip ftp password quake

This allows the FTP command to transparently insert the username and password when connecting to an FTP server.

What Is Configuration Rollback?



What Is Configuration Rollback?

- Human error represents significant downtime.
- “Fat-finger” keyboard entry errors can affect seasoned and knowledgeable professionals.
- Configuration rollback allows you to deal with reality while improving.
- Configuration rollback eases the impact of configuration mistakes.
- Configuration rollback has four main aspects:
  - Contextual configuration difference
  - Configuration archiving
  - Configuration replacing
  - Configuration logging

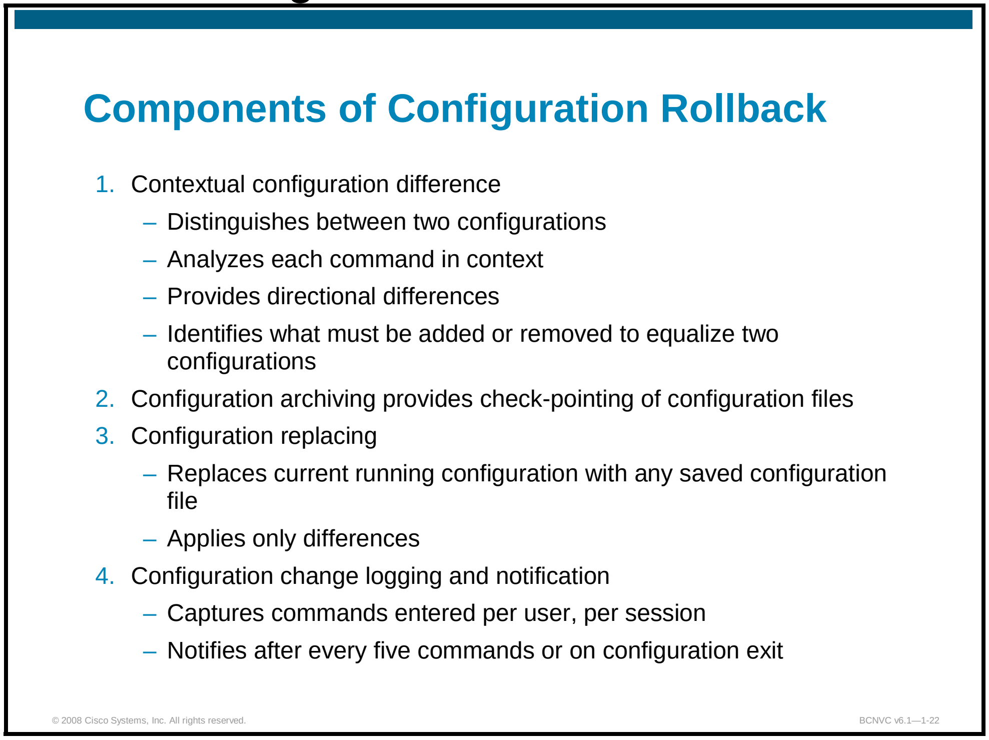
BCNVC v6.1—1-21

The concept of rollback comes from the transactional processing model common to database operations. In a database transaction, you might make a set of changes to a given database table. You then must choose whether to commit the changes (apply the changes permanently) or to roll back the changes (discard the changes and revert to the previous state of the table). In this context, rollback means that a journal file containing a log of the changes is discarded, and no changes are applied. The result of the rollback operation is to revert to the previous state, before any changes were applied.

The configure replace command allows you to revert to a previous configuration state, effectively rolling back changes that were made since the previous configuration state was saved. Instead of basing the rollback operation on a specific set of changes that were applied, the Cisco IOS configuration rollback capability uses the concept of reverting to a specific configuration state based on a saved Cisco IOS configuration file. This concept is similar to the database idea of saving a checkpoint (a saved version of the database) to preserve a specific state.

If the configuration rollback capability is desired, you must save the Cisco IOS running configuration before making any configuration changes. Then, after entering configuration changes, you can use that saved configuration file to roll back the changes (using the **configure replace target-url** command). Furthermore, since you can specify any saved Cisco IOS configuration file as the replacement configuration, you are not limited to a fixed number of rollbacks, as is the case in some rollback models based on a journal file.

Components of Configuration Rollback



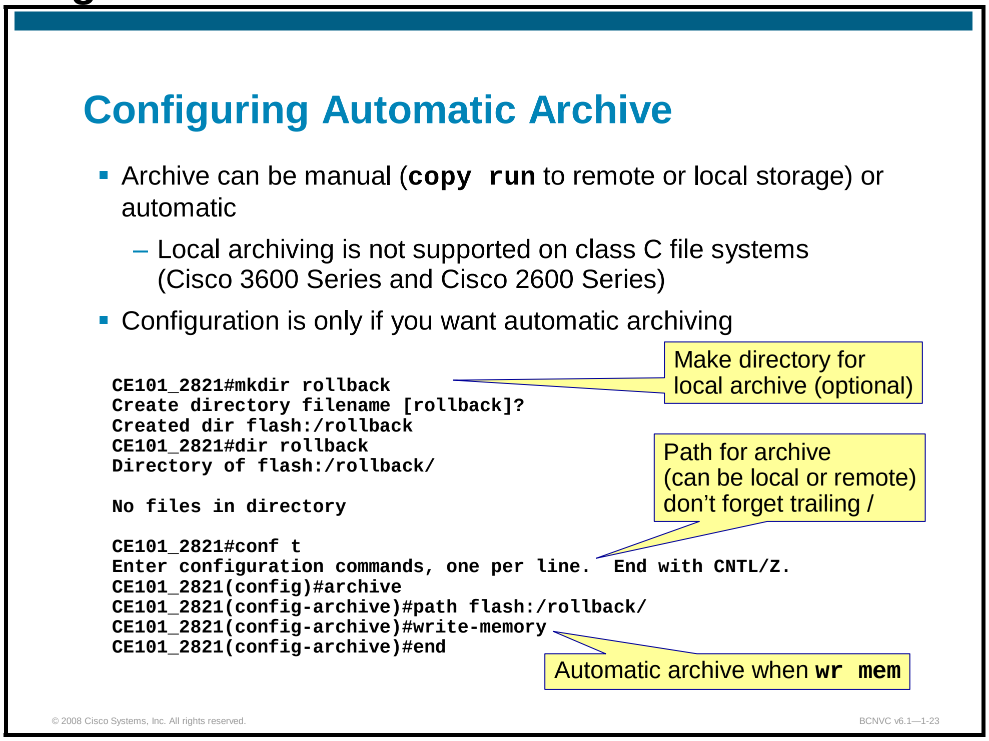
Components of Configuration Rollback

1. Contextual configuration difference
   - Distinguishes between two configurations
   - Analyzes each command in context
   - Provides directional differences
   - Identifies what must be added or removed to equalize two configurations
2. Configuration archiving provides check-pointing of configuration files
3. Configuration replacing
   - Replaces current running configuration with any saved configuration file
   - Applies only differences
4. Configuration change logging and notification
   - Captures commands entered per user, per session
   - Notifies after every five commands or on configuration exit

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—1-22

Configuration rollback has four components. First is the contextual configuration difference. There must be a difference between the current configuration and the configuration you want to roll back to. The second component is configuration archiving. You should be able to save different configurations from different points in time to roll back to. The third is configuration replacing. When you decide to roll back to a different configuration, only the differences between that configuration and the current configuration should be applied. The last component is configuration change logging and notification. Records need to be kept of configuration management changes, and notifications made of configuration activity.

Configuring Automatic Archive



Configuring Automatic Archive

- Archive can be manual (**`copy run`** to remote or local storage) or automatic
  - Local archiving is not supported on class C file systems (Cisco 3600 Series and Cisco 2600 Series)
- Configuration is only if you want automatic archiving

```
CE101_2821#mkdir rollback
Create directory filename [rollback]?
Created dir flash:/rollback
CE101_2821#dir rollback
Directory of flash:/rollback/

No files in directory

CE101_2821#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
CE101_2821(config)#archive
CE101_2821(config-archive)#path flash:/rollback/
CE101_2821(config-archive)#write-memory
CE101_2821(config-archive)#end
```

Make directory for local archive (optional)

Path for archive (can be local or remote) don’t forget trailing /

Automatic archive when **`wr mem`**

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—1-23

The Cisco IOS configuration archive is intended to provide a mechanism to store, organize, and manage an archive of Cisco IOS configuration files to enhance the configuration rollback capability provided by the configure replace command. Before this feature was introduced, you could save copies of the running configuration using the copy running-config destination-url command, storing the replacement file either locally or remotely. However, this method lacked any automated file management. On the other hand, the Configuration Replace and Configuration Rollback feature provides the capability to automatically save copies of the running configuration to the Cisco IOS configuration archive. These archived files serve as checkpoint configuration references and can be used by the configure replace command to revert to previous configuration states.

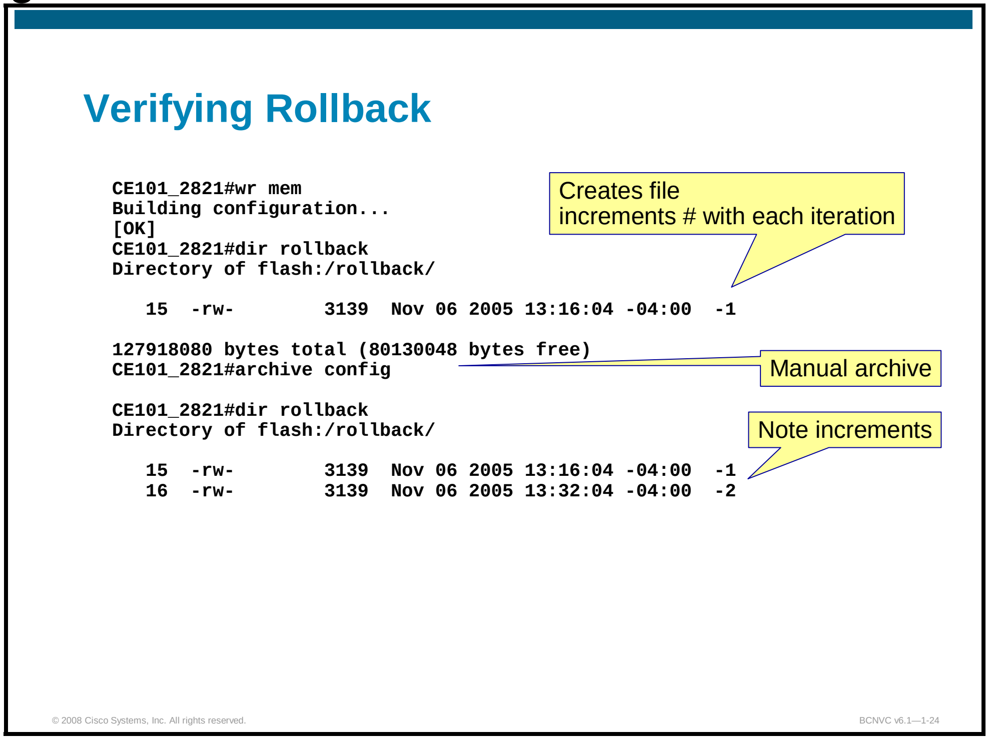
The archive config command allows you to save Cisco IOS configurations in the configuration archive using a standard location and filename prefix that is automatically appended with an incremental version number (and optional timestamp) as each consecutive file is saved. This functionality provides a means for consistent identification of saved Cisco IOS configuration files. You can specify how many versions of the running configuration are kept in the archive. After the maximum number of files are saved in the archive, the oldest file is automatically deleted when the next, most recent file is saved. The show archive command displays information for all configuration files saved in the Cisco IOS configuration archive.

The Cisco IOS configuration archive, in which the configuration files are stored and available for use with the configure replace command, can be located on the following file systems:

•If your platform has disk0—disk0:, disk1:, ftp:, pram:, rcp:, slavedisk0:, slavedisk1:, or tftp:

•If your platform does not have disk0—ftp:, http:, pram:, rcp:, or tftp:

Verifying Rollback



Verifying Rollback

```
CE101_2821#wr mem
Building configuration...
[OK]
CE101_2821#dir rollback
Directory of flash:/rollback/

   15  -rw-        3139  Nov 06 2005 13:16:04 -04:00  -1

127918080 bytes total (80130048 bytes free)
CE101_2821#archive config

CE101_2821#dir rollback
Directory of flash:/rollback/

   15  -rw-        3139  Nov 06 2005 13:16:04 -04:00  -1
   16  -rw-        3139  Nov 06 2005 13:32:04 -04:00  -2
```

Creates file
increments # with each iteration

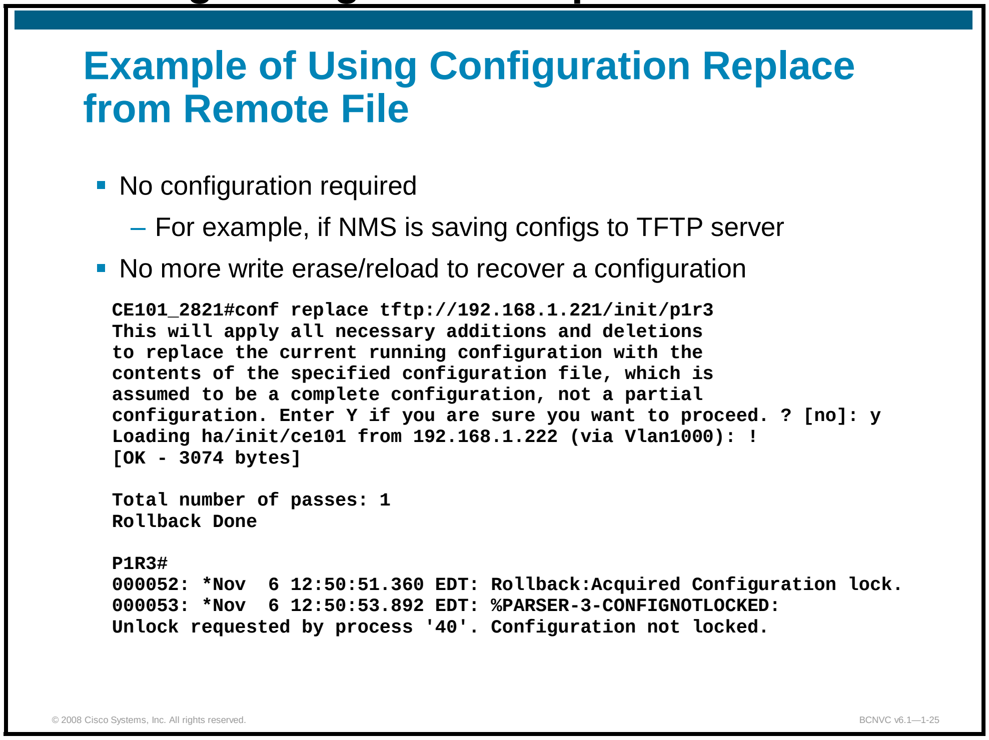
Manual archive

Note increments

BCNVC v6.1—1-24

When you perform a configuration rollback, a marker associated with the file increments by one. This marker is shown in the above illustration as "-1". In this illustration, a manual archive is performed. This adds another instance, and increments the marker by one. You can see on the last line above that the marker of the most recent instance has increased to 2.

Example of Using Configuration Replace from Remote File



Example of Using Configuration Replace from Remote File

- No configuration required
  - For example, if NMS is saving configs to TFTP server
- No more write erase/reload to recover a configuration

```
CE101_2821#conf replace tftp://192.168.1.221/init/p1r3
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: y
Loading ha/init/ce101 from 192.168.1.222 (via Vlan1000): !
[OK - 3074 bytes]

Total number of passes: 1
Rollback Done

P1R3#
000052: *Nov  6 12:50:51.360 EDT: Rollback:Acquired Configuration lock.
000053: *Nov  6 12:50:53.892 EDT: %PARSER-3-CONFIGNOTLOCKED:
Unlock requested by process '40'. Configuration not locked.
```

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—1-25

The configure replace command provides the capability to replace the current running configuration with any saved Cisco IOS configuration file. This functionality can be used to revert to a previous configuration state, effectively rolling back any configuration changes that were made since the previous configuration state was saved.

When using the configure replace command, you must specify a saved Cisco IOS configuration as the replacement configuration file for the current running configuration. The replacement file must be a complete configuration generated by a Cisco IOS device (for example, a configuration generated by the copy running-config destination-url command), or, if generated externally, the replacement file must comply with the format of files generated by Cisco IOS devices. When the configure replace command is entered, the current running configuration is compared with the specified replacement configuration and a set of diffs is generated. The algorithm used to compare the two files is the same as that employed by the show archive config differences command. The resulting diffs are then applied by the Cisco IOS parser to achieve the replacement configuration state. Only the diffs are applied, avoiding potential service disruption from reapplying configuration commands that already exist in the current running configuration. This algorithm effectively handles configuration changes to order-dependent commands (such as access lists) through a multiple pass process. Under normal circumstances, no more than three passes are needed to complete a configuration replace operation, and a limit of five passes is performed to preclude any looping behavior.

The Cisco IOS **copy source-url running-config** command is often used to copy a stored Cisco IOS configuration file to the running configuration. When using the copy source-url running-config command as an alternative to the configure replace target-url command, the following major differences should be noted:

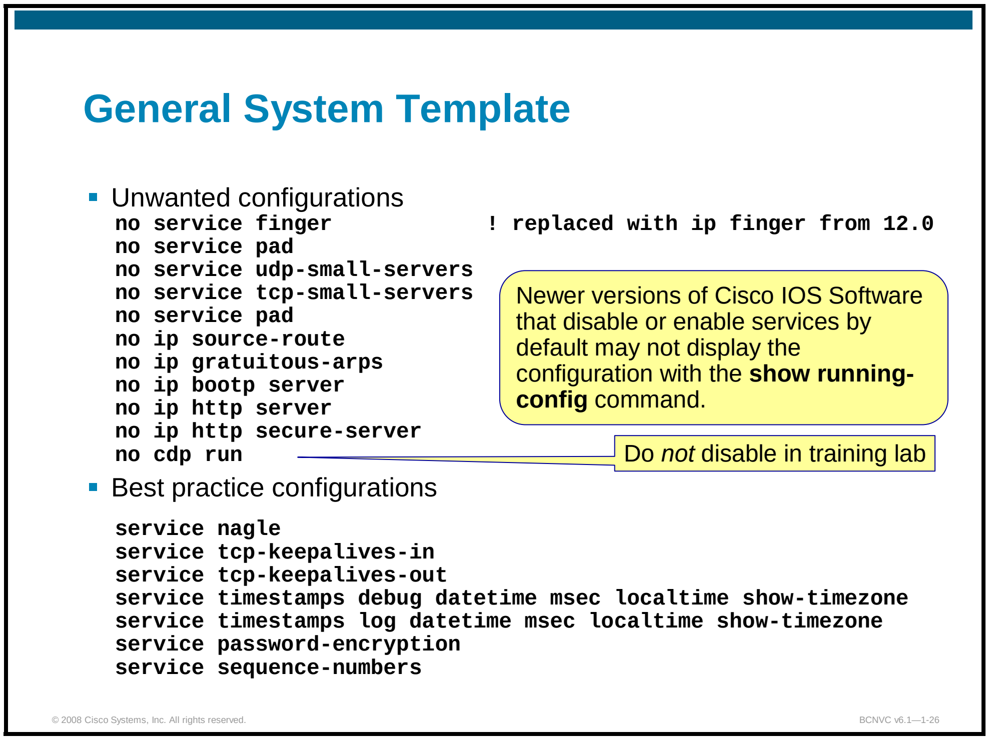
- The **copy source-url running-config** command is a merge operation and preserves all the commands from both the source file and the current running configuration. This command does not remove commands from the current running configuration that are not present in the source file. In contrast, the **configure replace target-url** command removes commands from the current running configuration that are not present in the replacement file and adds commands to the current running configuration that need to be added.

- The **copy source-url running-config** command applies every command in the source file, whether or not the command is already present in the current running configuration. This algorithm is inefficient and, in some cases, can result in service outages. In contrast, the **configure replace target-url** command only applies the commands that need to be applied—no existing commands in the current running configuration are reapplied.
- A partial configuration file may be used as the source file for the copy source-url running-config command, whereas a complete Cisco IOS configuration file must be used as the replacement file for the configure replace target-url command.

Note In Cisco IOS Release 12.2(25)S and 12.3(14)T, a locking feature for the configuration replace operation was introduced. When the configure replace command is used, the running configuration file is locked by default for the duration of the configuration replace operation. This locking mechanism prevents other users from changing the running configuration while the replacement operation is taking place, which might otherwise cause the replacement operation to terminate unsuccessfully. You can disable the locking of the running configuration by using the nolock keyword when issuing the configure replace command.

Note The running configuration lock is automatically cleared at the end of the configuration replace operation. You can display any locks that may be currently applied to the running configuration using the show configuration lock command.

General System Template for Best Practice Basic Configuration



General System Template

- Unwanted configurations

```
no service finger            ! replaced with ip finger from 12.0
no service pad
no service udp-small-servers
no service tcp-small-servers
no service pad
no ip source-route
no ip gratuitous-arps
no ip bootp server
no ip http server
no ip http secure-server
no cdp run
```

Newer versions of Cisco IOS Software that disable or enable services by default may not display the configuration with the **show running-config** command.

Do *not* disable in training lab

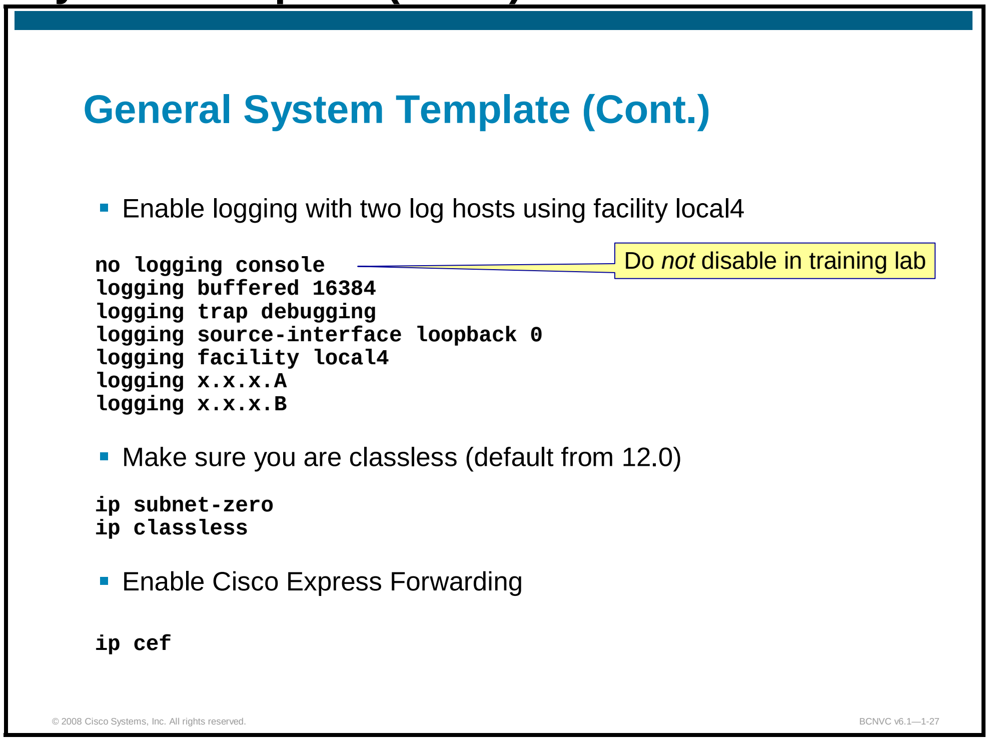
- Best practice configurations

```
service nagle
service tcp-keepalives-in
service tcp-keepalives-out
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
service sequence-numbers
```

BCNVC v6.1—1-26

The general service template is a cut-and-paste template you can modify and use to configure your routers. Be sure to change any IP addresses and autonomous system numbers when you use any template. The commands listed here cover numerous functions. We are not going to cover all of them in this lesson. Refer to the Cisco IOS Software user documentation for command explanation.

General System Template (Cont.)



General System Template (Cont.)

- Enable logging with two log hosts using facility local4

```
no logging console
logging buffered 16384
logging trap debugging
logging source-interface loopback 0
logging facility local4
logging x.x.x.A
logging x.x.x.B
```

Do *not* disable in training lab

- Make sure you are classless (default from 12.0)

```
ip subnet-zero
ip classless
```

- Enable Cisco Express Forwarding

```
ip cef
```

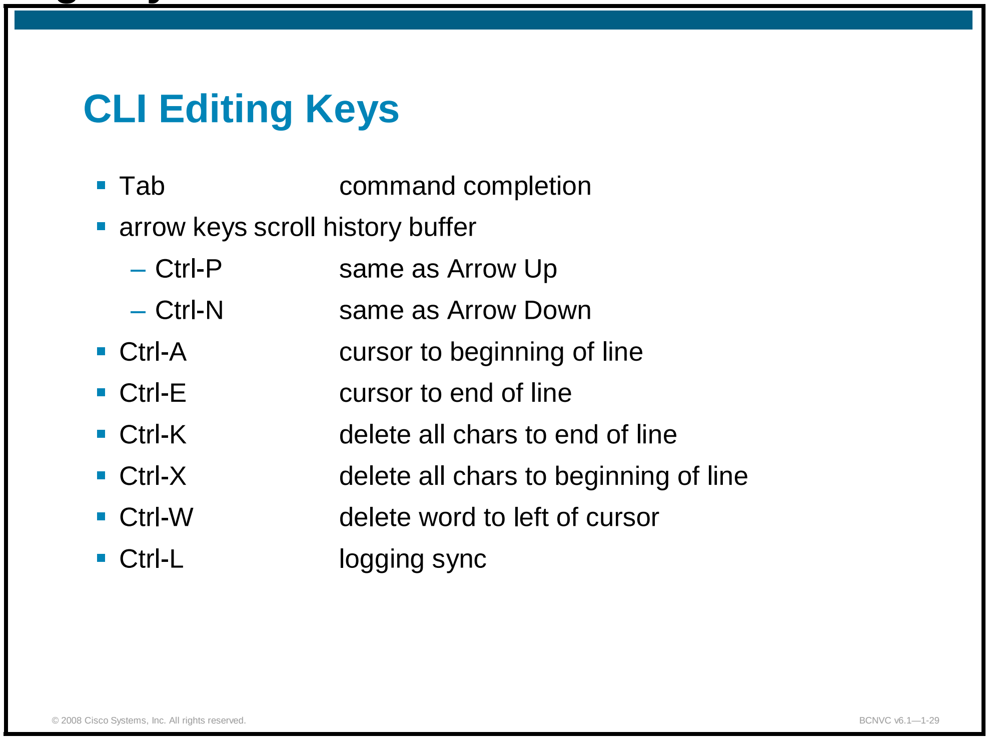
The command **no logging console** prevents logging messages from appearing while you are typing in the IOS console.

CLI Tips, Tricks, and Features

The Cisco IOS Software command line interface (CLI) is the traditional (and favored) way of interacting with a router to enter and change a configuration and to monitor router operation. This section describes how an ISP operator uses the CLI.

The CLI is well-documented in the Cisco UniverCD documentation set. However, a few tips and tricks that are regularly used are worth mentioning here.

CLI Editing Keys



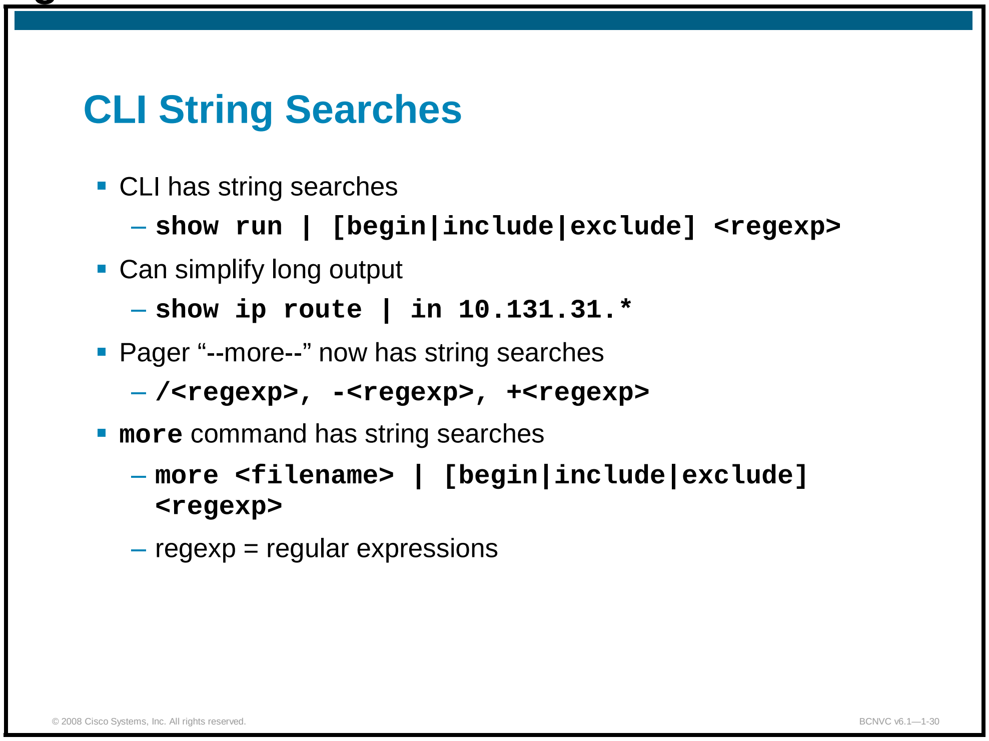
CLI Editing Keys

Tab	command completion
arrow keys scroll history buffer	
– Ctrl-P	same as Arrow Up
– Ctrl-N	same as Arrow Down
Ctrl-A	cursor to beginning of line
Ctrl-E	cursor to end of line
Ctrl-K	delete all chars to end of line
Ctrl-X	delete all chars to beginning of line
Ctrl-W	delete word to left of cursor
Ctrl-L	logging sync

BCNVC v6.1—1-29

Several keys are useful as shortcuts for editing the Cisco IOS Software configuration. Although these are covered in detail in the Cisco IOS Software Release 12.0 documentation, it is useful to point out above some of those that are most commonly used.

CLI String Searches



CLI String Searches

- CLI has string searches
  - **show run | [begin|include|exclude] <regexp>**
- Can simplify long output
  - **show ip route | in 10.131.31.***
- Pager “--more--” now has string searches
  - **/<regexp>, -<regexp>, +<regexp>**
- **more** command has string searches
  - **more <filename> | [begin|include|exclude] <regexp>**
  - regexp = regular expressions

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—1-30

The Cisco IOS CLI provides ways of searching through large amounts of command output and filtering output to exclude information you do not need. This UNIX grep-like function (pattern search) allows operators to search for common expressions in configuration and other terminal output. Only salient points are discussed here.

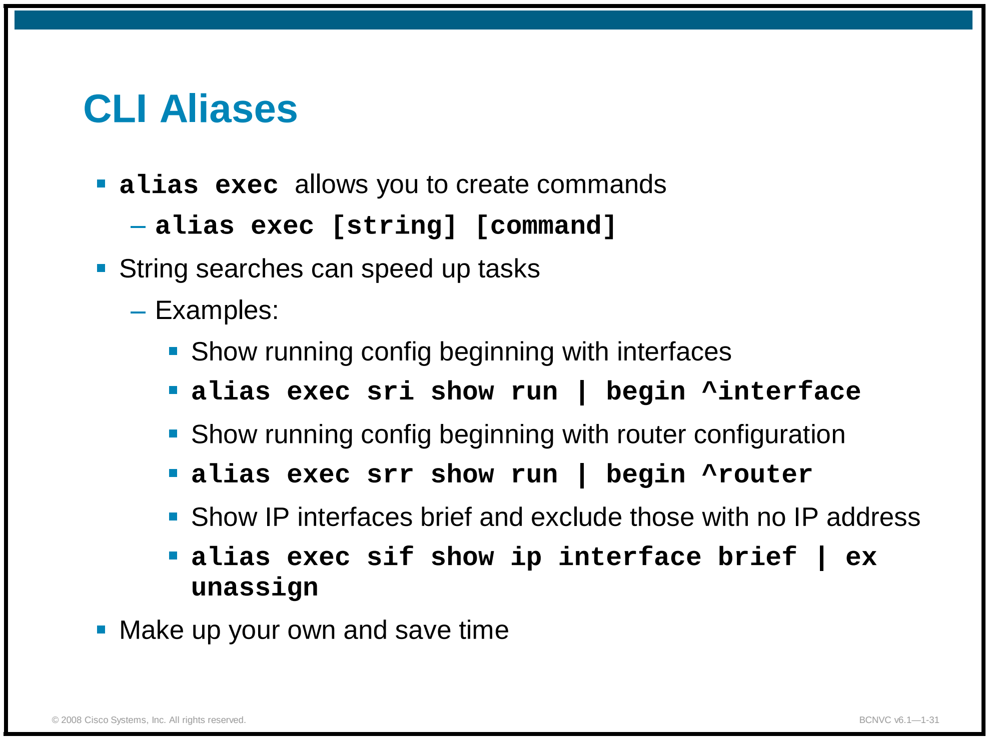
In addition to making information more manageable, using these features also provides the benefit of reducing router CPU usage by removing output before incurring transmission costs.

The function is invoked by using a vertical bar "|" like the UNIX pipe command.

During the display of configuration or file contents, the screen pager "—More—" appears if the output is longer than the current terminal length setting. It is possible to do a regular expression search at this prompt, too. The "/ key" matches the begin keyword; the "**-** key" means to exclude, and the "+ key" means to include.

Finally, in Enable mode it is possible to use the **more** command to display file contents. Regular expressions can be used with **more**. By using the "|" after the **more** command and its options, it is possible to search within the file for the strings of interest in the same way as discussed previously.

CLI Aliases



CLI Aliases

- **alias exec** allows you to create commands
  - **alias exec [string] [command]**
- String searches can speed up tasks
  - Examples:
    - Show running config beginning with interfaces
    - **alias exec sri show run | begin ^interface**
    - Show running config beginning with router configuration
    - **alias exec srr show run | begin ^router**
    - Show IP interfaces brief and exclude those with no IP address
    - **alias exec sif show ip interface brief | ex unassign**
- Make up your own and save time

BCNVC v6.1—1-31

To save time and the repetition of entering the same command multiple times, you can use a command alias. An alias can be configured to do anything that can be done at the command line, but an alias cannot move between modes, type in passwords, or perform any interactive functions.

The table below shows the default command aliases.

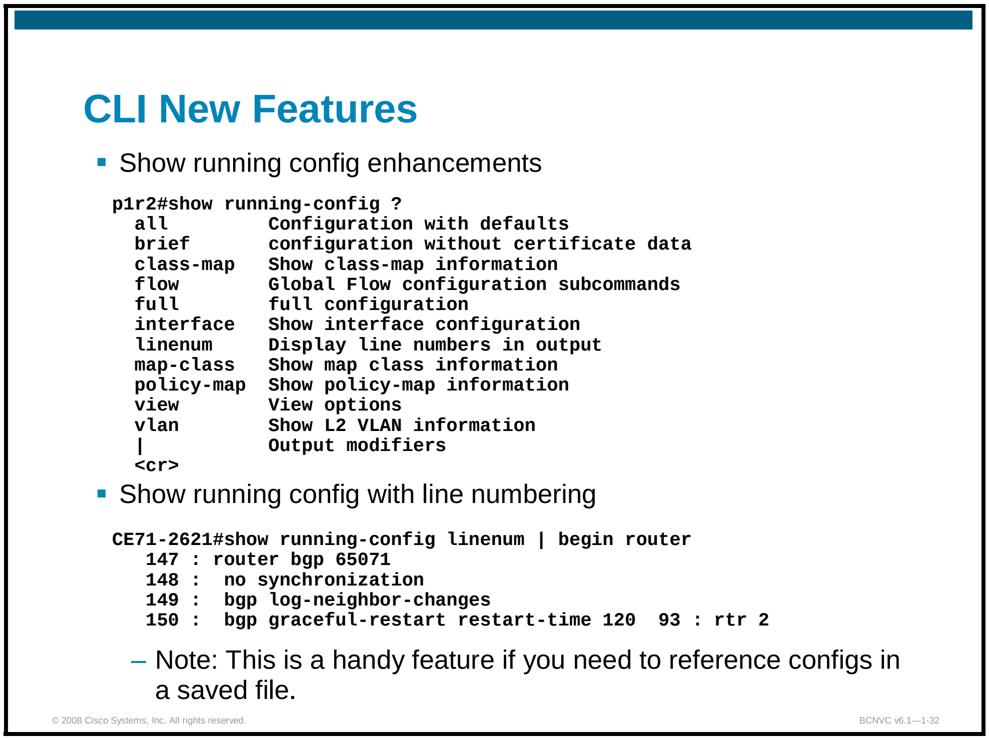
Command Alias	**Original Command**
h	help
lo	logout
p	ping
s	show
u or un	undebug
w	where

To create a command alias, issue the **alias** command in global configuration mode. The syntax of the command is **alias** *mode command-alias original-command*. Following are some examples:

```
P1R1(config)# alias exec prt partition—privileged EXEC mode
P1R1(config)# alias configure sb source-bridge—global configuration
mode
P1R1(config)# alias interface rl rate-limit—interface configuration
mode
```

To view both default and user-created aliases, issue the **show alias** command.

CLI New Features



CLI New Features

- Show running config enhancements

```
p1r2#show running-config ?
  all         Configuration with defaults
  brief       configuration without certificate data
  class-map   Show class-map information
  flow        Global Flow configuration subcommands
  full        full configuration
  interface   Show interface configuration
  linenum     Display line numbers in output
  map-class   Show map class information
  policy-map  Show policy-map information
  view        View options
  vlan        Show L2 VLAN information
  |           Output modifiers
  <cr>
```

- Show running config with line numbering

```
CE71-2621#show running-config linenum | begin router
   147 : router bgp 65071
   148 :  no synchronization
   149 :  bgp log-neighbor-changes
   150 :  bgp graceful-restart restart-time 120  93 : rtr 2
```

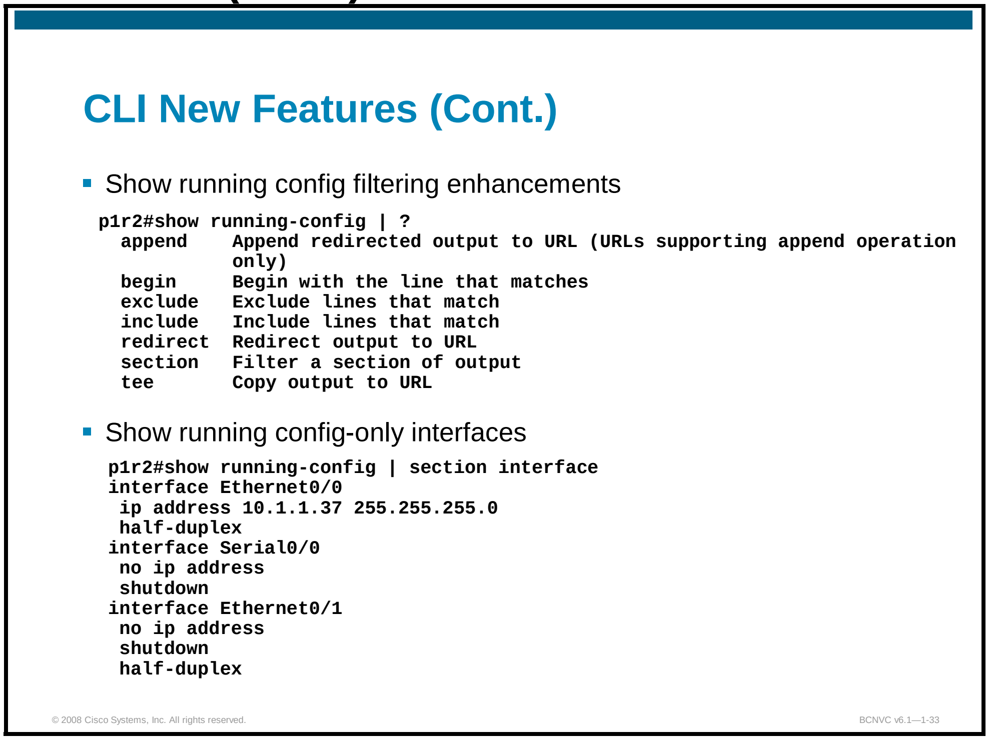
  – Note: This is a handy feature if you need to reference configs in a saved file.

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—1-32

There are new keywords you can add to the **show running-configuration** command that modify the results of the command output. These keywords are shown in the upper portion of the example above. The purpose of this lesson is not to cover the functions of all of these keywords, but merely to introduce you to their availability. The functions of these keywords are all covered in the Cisco IOS Software user documentation.

As an example, you can add line numbers to output generated from the **show running-configuration** command. Do this by adding the **linenum** keyword after the **show running-configuration** command as shown in the lower portion of the figure above.

CLI New Features (Cont.)



CLI New Features (Cont.)

- Show running config filtering enhancements

```
p1r2#show running-config | ?
  append    Append redirected output to URL (URLs supporting append operation
            only)
  begin     Begin with the line that matches
  exclude   Exclude lines that match
  include   Include lines that match
  redirect  Redirect output to URL
  section   Filter a section of output
  tee       Copy output to URL
```

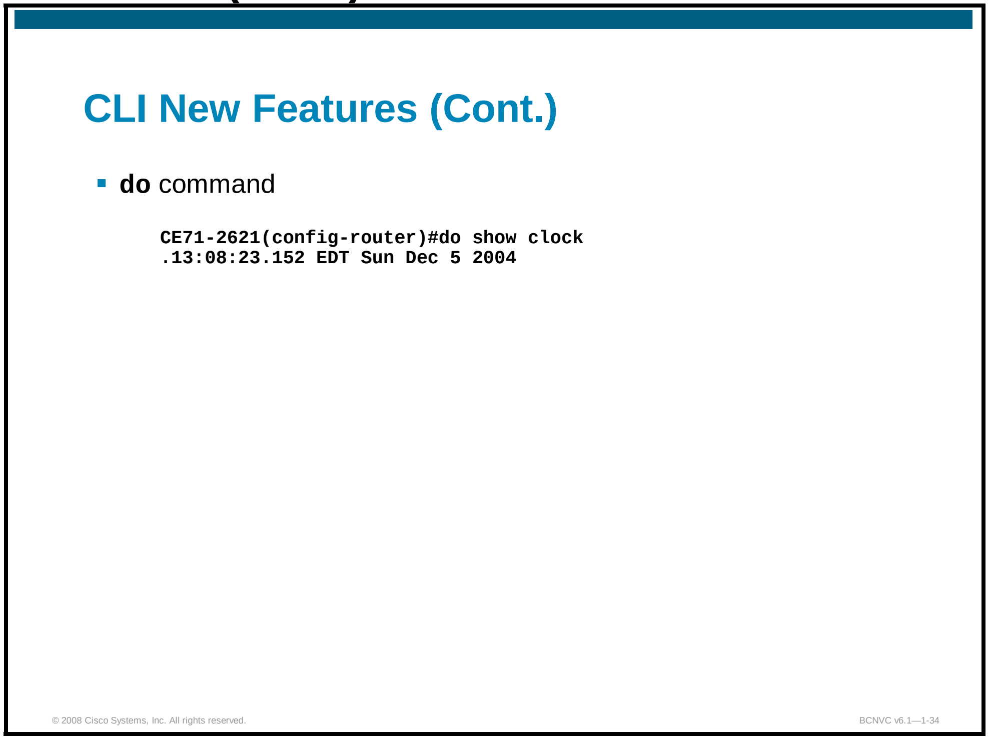
- Show running config-only interfaces

```
p1r2#show running-config | section interface
interface Ethernet0/0
 ip address 10.1.1.37 255.255.255.0
 half-duplex
interface Serial0/0
 no ip address
 shutdown
interface Ethernet0/1
 no ip address
 shutdown
 half-duplex
```

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—1-33

You can filter the output of the **show running-configuration** command by using "|" (similar to the Unix pipe character) followed by keywords. This way you do not have to scroll through a large amount of output; you can only select the output that you want. The upper half of the example above shows the keywords that are available for filtering. The lower half shows an example of filtering using the "section interface" keywords followed by the | character.

CLI New Features (Cont.)



CLI New Features (Cont.)

- **do** command

```
CE71-2621(config-router)#do show clock
.13:08:23.152 EDT Sun Dec 5 2004
```

BCNVC v6.1—1-34

To execute an EXEC-level command from global configuration mode or any configuration submode, use the **do** command in any configuration mode. This saves you from having to switch between command modes.

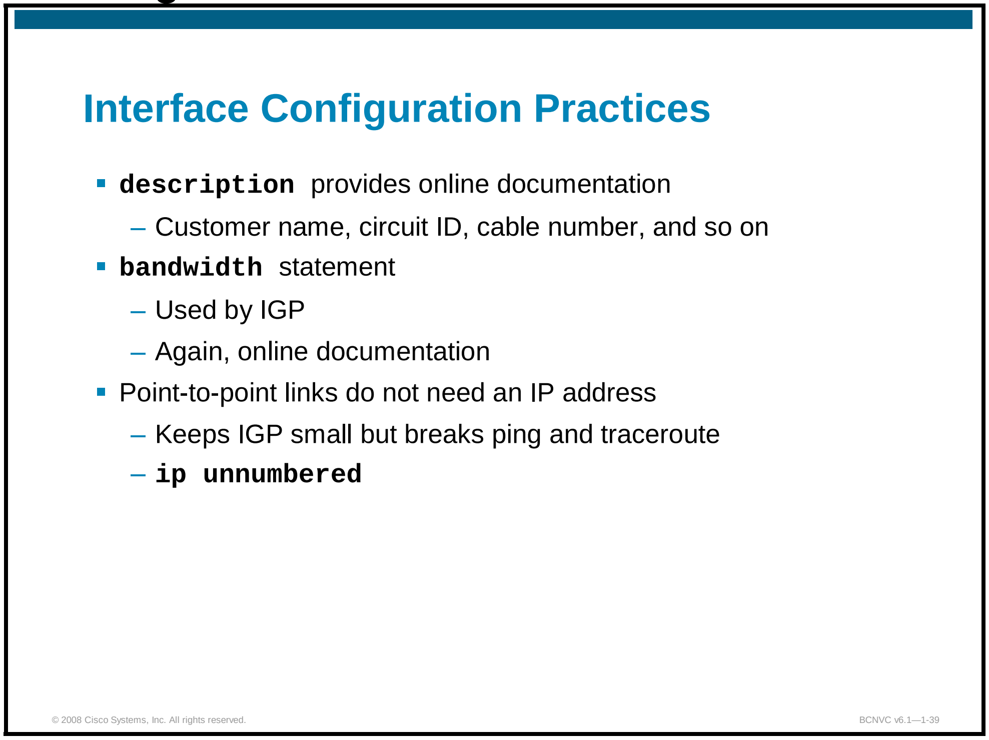
do *command*

Note	These features were first included in the 12.2(4)T/12.3(8)T timeframe. Not all options are available on all platforms, releases, or feature sets.

Configuring Interfaces

This topic covers best practices for configuring interfaces on networking devices in a core network. It covers interface configuration practices and provides examples, discusses the differences in configuring layer 2 and layer 3 interfaces, and covers how to check the status of interfaces.

Interface Configuration Practices



Interface Configuration Practices

- **`description`** provides online documentation
  - Customer name, circuit ID, cable number, and so on
- **`bandwidth`** statement
  - Used by IGP
  - Again, online documentation
- Point-to-point links do not need an IP address
  - Keeps IGP small but breaks ping and traceroute
  - **`ip unnumbered`**

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—1-39

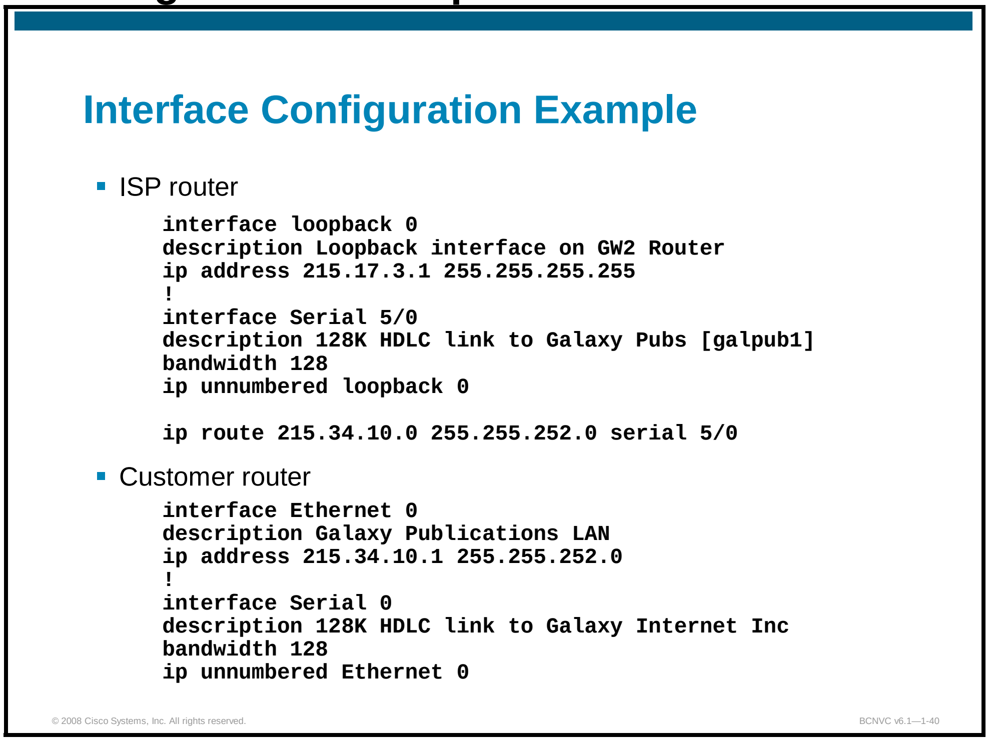
Configuring interfaces involves more than simply plugging in the cable and activating the interface with the **no shutdown** command. Also consider whether it is a WAN or a LAN, whether a routing protocol is running across the interface, addressing and masks to be used, and operator information.

description: Use the **description** interface command to document details such as the circuit bandwidth, customer name, database entry mnemonic, circuit number that the circuit supplier gave you, and cable number. This sounds like overkill, but it makes it easy to learn relevant details from the **show interface** command without having to reference offline documents. This ensures that reconstructing configurations and diagnosing problems are made considerably easier.

bandwidth: The **bandwidth interface** command is used by interior routing protocols to decide optimum routing, and it is especially important to set this command properly in the case of backbone links using only a portion of the available bandwidth support by the interface. For example, a serial interface on a router supports speeds up to 4 Mbps but has a default bandwidth setting of 1.5 Mbps. If the backbone has different size links from 64 kbps to 4 Mbps and the bandwidth command is not used, the interior routing protocol assumes that all the links have the same cost and calculates optimum paths accordingly — which could be less than ideal. It also provides very useful online documentation for the circuit bandwidth.

ip unnumbered: To avoid problems of having many /30 networks in the internal routing protocol, and to avoid the problems of keeping internal documentation consistent with network deployment, you could use unnumbered point-to-point links. In doing so, the loopback interface is referenced.

Interface Configuration Example



Interface Configuration Example

- ISP router

```
interface loopback 0
description Loopback interface on GW2 Router
ip address 215.17.3.1 255.255.255.255
!
interface Serial 5/0
description 128K HDLC link to Galaxy Pubs [galpub1]
bandwidth 128
ip unnumbered loopback 0

ip route 215.34.10.0 255.255.252.0 serial 5/0
```

- Customer router

```
interface Ethernet 0
description Galaxy Publications LAN
ip address 215.34.10.1 255.255.252.0
!
interface Serial 0
description 128K HDLC link to Galaxy Internet Inc
bandwidth 128
ip unnumbered Ethernet 0
```

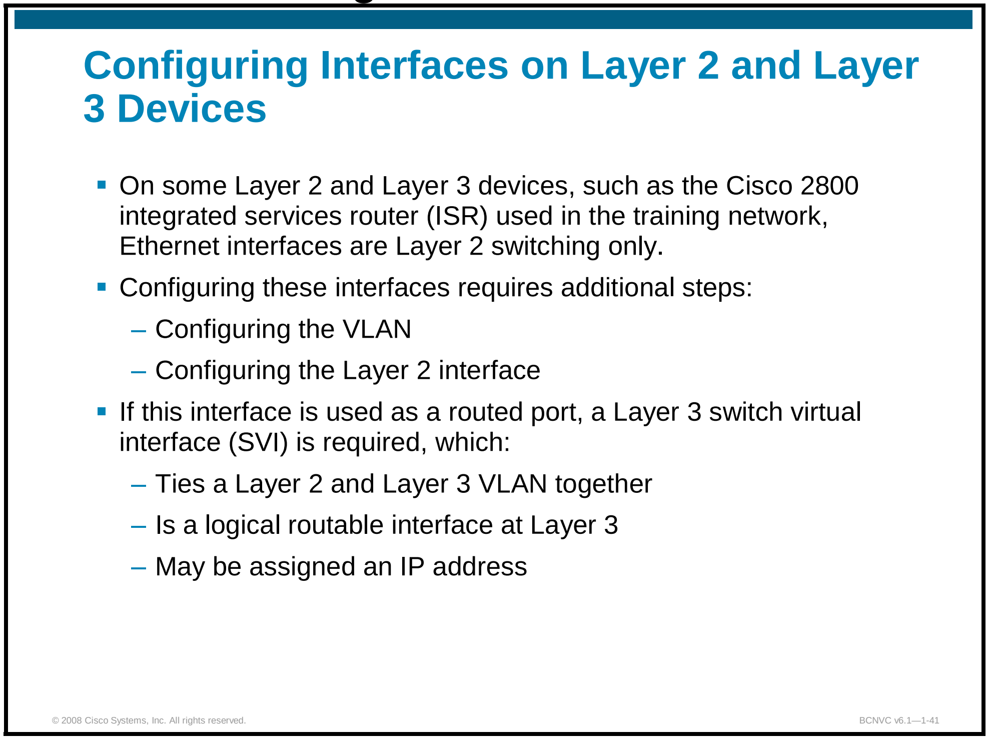
BCNVC v6.1—1-40

In this example, the regional or local registry has allocated the customer the network block 215.34.10.0/22. This is routed to the customer site with the static route pointing to Serial 5/0. The customer router simply needs a default route pointing to its serial interface to ensure a connection.

With this configuration, there are no /30s from point-to-point links present in the IGP, and the ISP does not need to document the link address or keep a table or database up-to-date. This makes for easier configuration as well as easier operation of the ISP business.

Note the contents of the description fields. All of this data is online documentation, seemingly superfluous, but very necessary to ensure smooth and efficient operations. All the information pertinent to the customer connection from the cabling to the IP values is contained in the interface configuration. If the ISP database is down or unavailable, any debug information required by operators or engineers can be found on the router itself.

Interface Status Checking



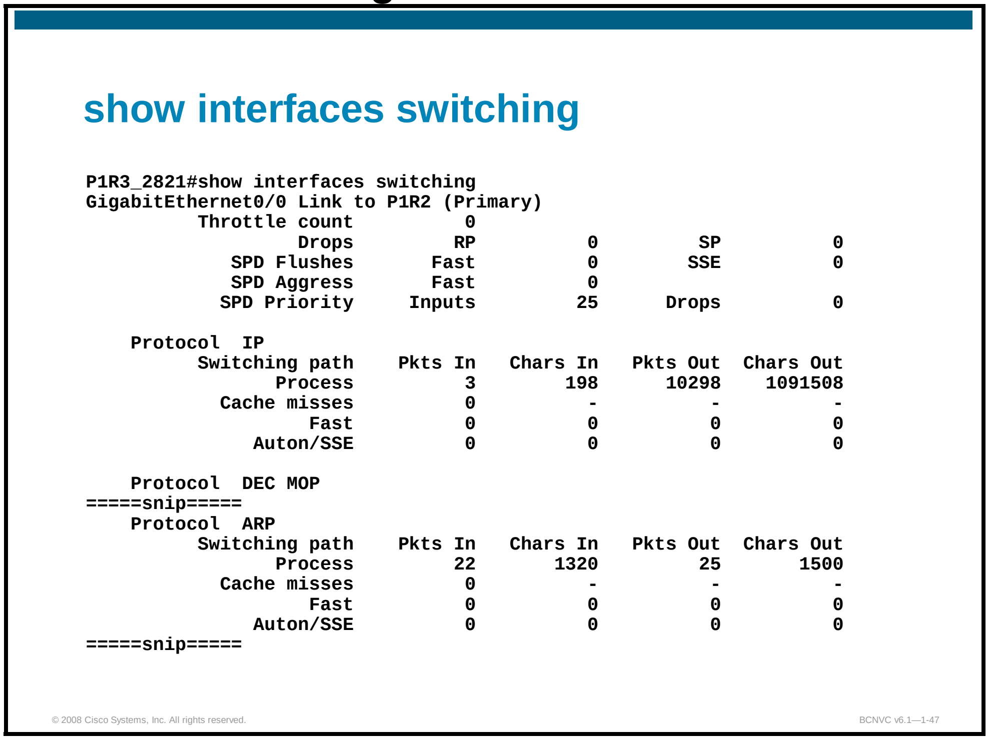
Configuring Interfaces on Layer 2 and Layer 3 Devices

- On some Layer 2 and Layer 3 devices, such as the Cisco 2800 integrated services router (ISR) used in the training network, Ethernet interfaces are Layer 2 switching only.
- Configuring these interfaces requires additional steps:
  - Configuring the VLAN
  - Configuring the Layer 2 interface
- If this interface is used as a routed port, a Layer 3 switch virtual interface (SVI) is required, which:
  - Ties a Layer 2 and Layer 3 VLAN together
  - Is a logical routable interface at Layer 3
  - May be assigned an IP address

BCNVC v6.1—1-41

Some useful Cisco IOS Software commands enable you to check the status of interfaces in Cisco IOS Software. Two useful commands are **show interface switching** and **show interface stats**. More detailed information regarding these commands is provided over the next two pages.

Show Interfaces Switching



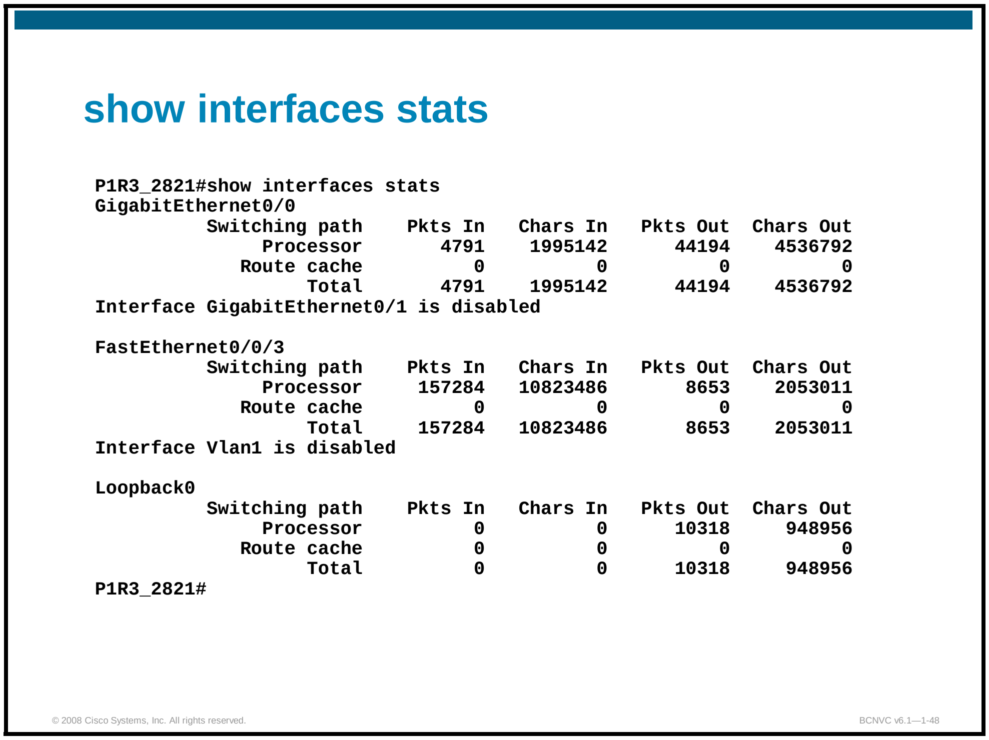
show interfaces switching

```
P1R3_2821#show interfaces switching
GigabitEthernet0/0 Link to P1R2 (Primary)
          Throttle count          0
                   Drops         RP          0         SP          0
             SPD Flushes       Fast          0        SSE          0
             SPD Aggress       Fast          0
            SPD Priority     Inputs         25      Drops          0

    Protocol  IP
          Switching path    Pkts In   Chars In   Pkts Out  Chars Out
                 Process          3        198      10298    1091508
            Cache misses          0          -          -          -
                    Fast          0          0          0          0
               Auton/SSE          0          0          0          0

    Protocol  DEC MOP
=====snip=====
    Protocol  ARP
          Switching path    Pkts In   Chars In   Pkts Out  Chars Out
                 Process         22       1320         25       1500
            Cache misses          0          -          -          -
                    Fast          0          0          0          0
               Auton/SSE          0          0          0          0
=====snip=====
```

The Cisco IOS command **show interface switching** provides useful information about the switching status of a router interface, either on an individual interface basis or over the whole router. The full command format is **show interface [int n/n] switching**, where an optional argument is the specific interface in question.

This sample output shows SPD activity, as well as other activity on that particular interface on the router. Note the references to autonomous/SSE switching—this applies only to the Cisco 7000-series with Silicon Switch Engine only (a product that is now discontinued). Fast switching refers to all packets that have not been process-switched, which includes optimum switching, NetFlow, and Cisco Express Forwarding.

Show Interfaces Stats



show interfaces stats

```
P1R3_2821#show interfaces stats
GigabitEthernet0/0
          Switching path    Pkts In   Chars In   Pkts Out  Chars Out
               Processor       4791    1995142      44194    4536792
             Route cache          0          0          0          0
                   Total       4791    1995142      44194    4536792
Interface GigabitEthernet0/1 is disabled

FastEthernet0/0/3
          Switching path    Pkts In   Chars In   Pkts Out  Chars Out
               Processor     157284   10823486       8653    2053011
             Route cache          0          0          0          0
                   Total     157284   10823486       8653    2053011
Interface Vlan1 is disabled

Loopback0
          Switching path    Pkts In   Chars In   Pkts Out  Chars Out
               Processor          0          0      10318     948956
             Route cache          0          0          0          0
                   Total          0          0      10318     948956
P1R3_2821#
```

© 2008 Cisco Systems, Inc. All rights reserved.　BCNVC v6.1—1-48

The Cisco IOS command **show interface stats** shows the number of packets and characters inbound and outbound on an individual router interface. The full command format is **show interface [int n/n] stats**, where an optional argument is the specific interface in question.

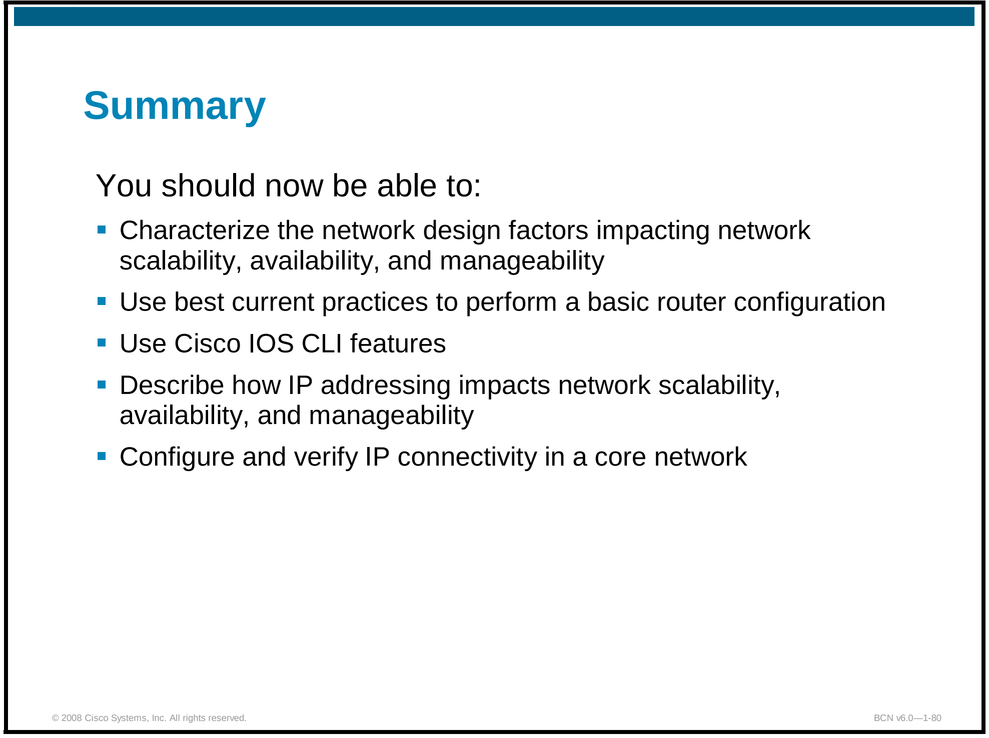
The output of interface switching differentiates between packets that go via the processor and those that have been processed via the route cache. This is useful to determine the level of process switching taking place on a router.

On a router that supports distributed switching, the output looks like the following:

```
P1R1>show interface stats
FastEthernet0/1/0
Switching path    Pkts In   Chars In   Pkts Out  Chars Out
Processor         207745   14075132     270885   21915788
Route cache          0          0          0          0
Distributed cache   93       9729          0          0
Total             207838   14084861     270885   21915788
```

Notice that packets that have been processed via the distributed cache are counted separately from those handled via the central route cache and the processor.

Summary



Summary

You should now be able to:

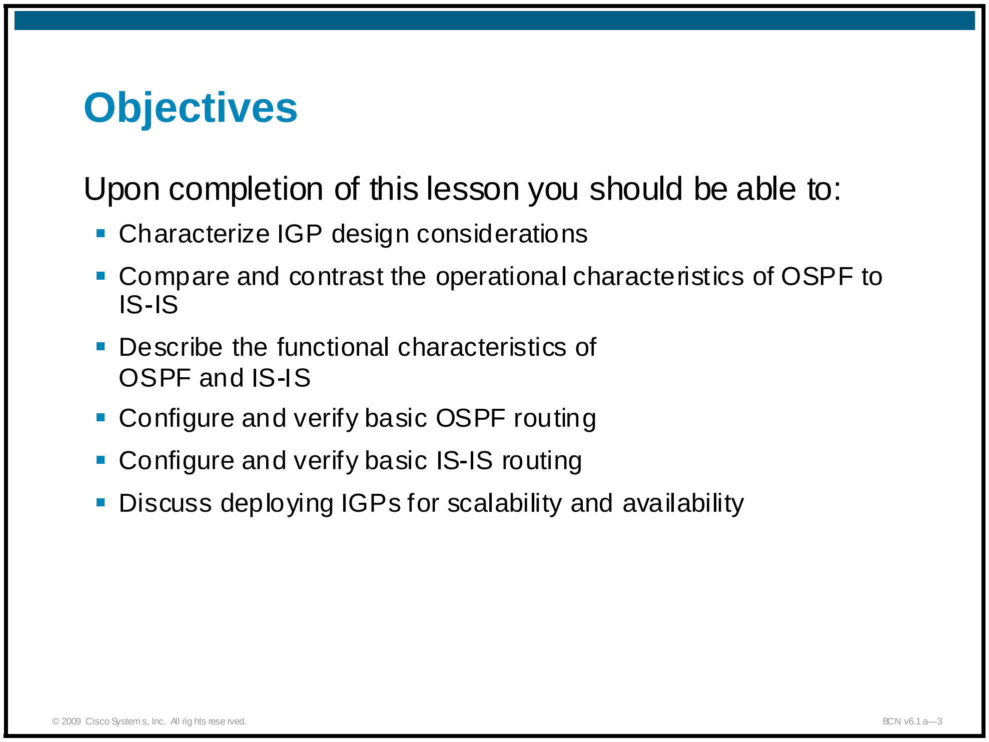
- Characterize the network design factors impacting network scalability, availability, and manageability
- Use best current practices to perform a basic router configuration
- Use Cisco IOS CLI features
- Describe how IP addressing impacts network scalability, availability, and manageability
- Configure and verify IP connectivity in a core network

BCN v6.0—1-80

Lesson 2

Implement Link-state Protocols (OSPF and IS-IS)

Objectives

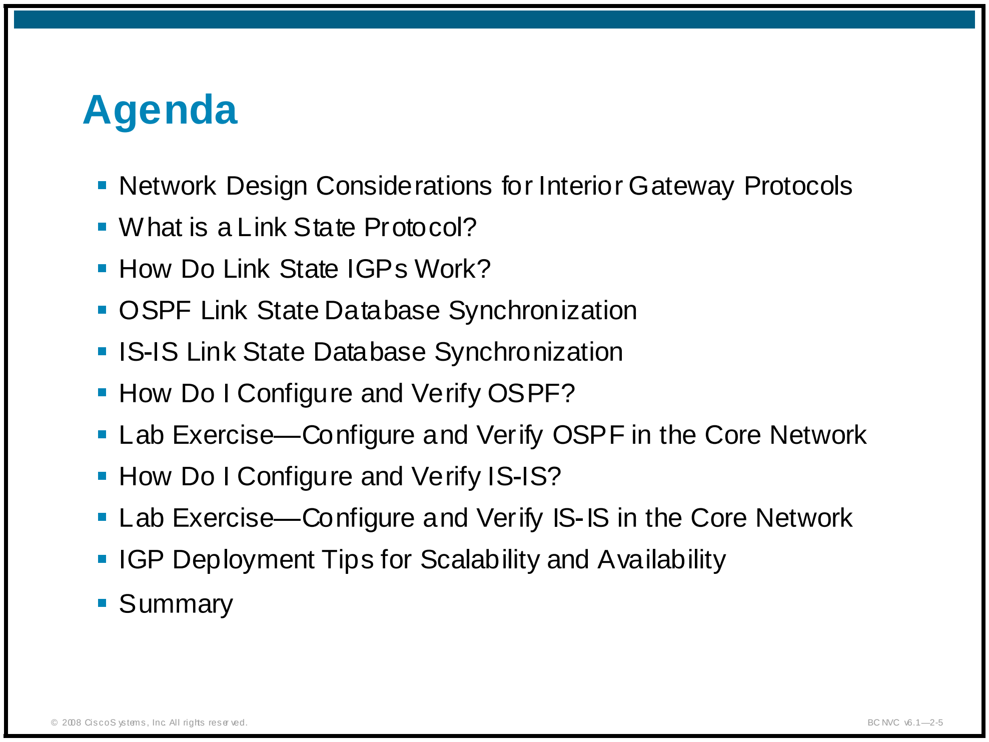


Objectives

Upon completion of this lesson you should be able to:

- Characterize IGP design considerations
- Compare and contrast the operational characteristics of OSPF to IS-IS
- Describe the functional characteristics of OSPF and IS-IS
- Configure and verify basic OSPF routing
- Configure and verify basic IS-IS routing
- Discuss deploying IGPs for scalability and availability

Agenda



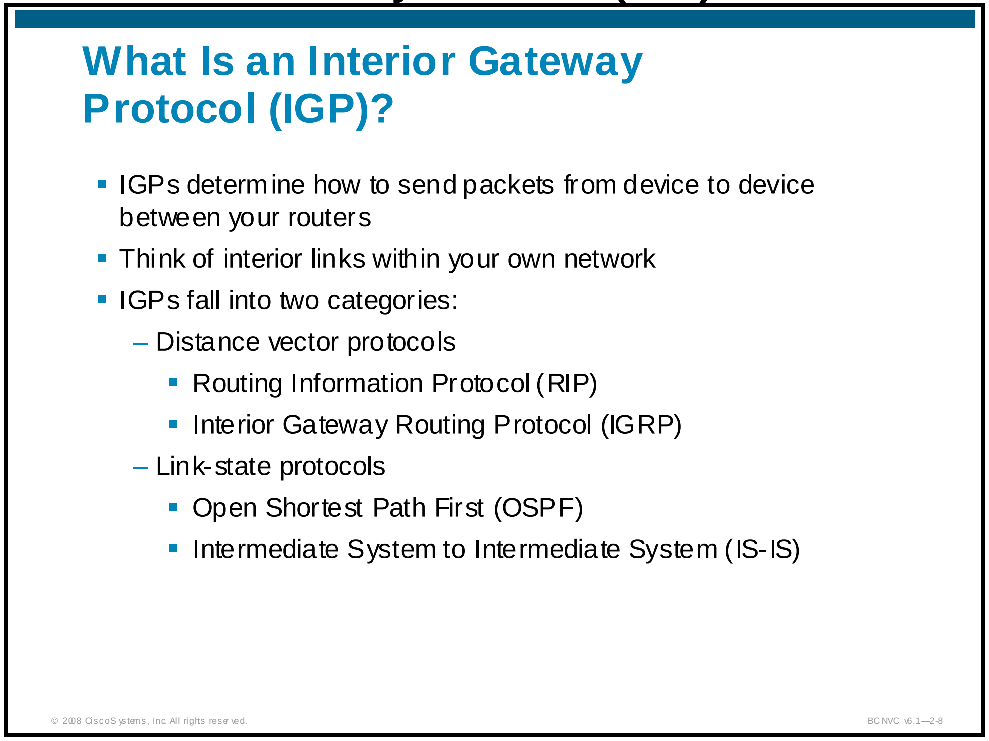
Agenda

- Network Design Considerations for Interior Gateway Protocols
- What is a Link State Protocol?
- How Do Link State IGPs Work?
- OSPF Link State Database Synchronization
- IS-IS Link State Database Synchronization
- How Do I Configure and Verify OSPF?
- Lab Exercise—Configure and Verify OSPF in the Core Network
- How Do I Configure and Verify IS-IS?
- Lab Exercise—Configure and Verify IS-IS in the Core Network
- IGP Deployment Tips for Scalability and Availability
- Summary

BCNVC v6.1—2-5

Network Design Considerations for Interior Gateway Protocols

What Is an Interior Gateway Protocol (IGP)?



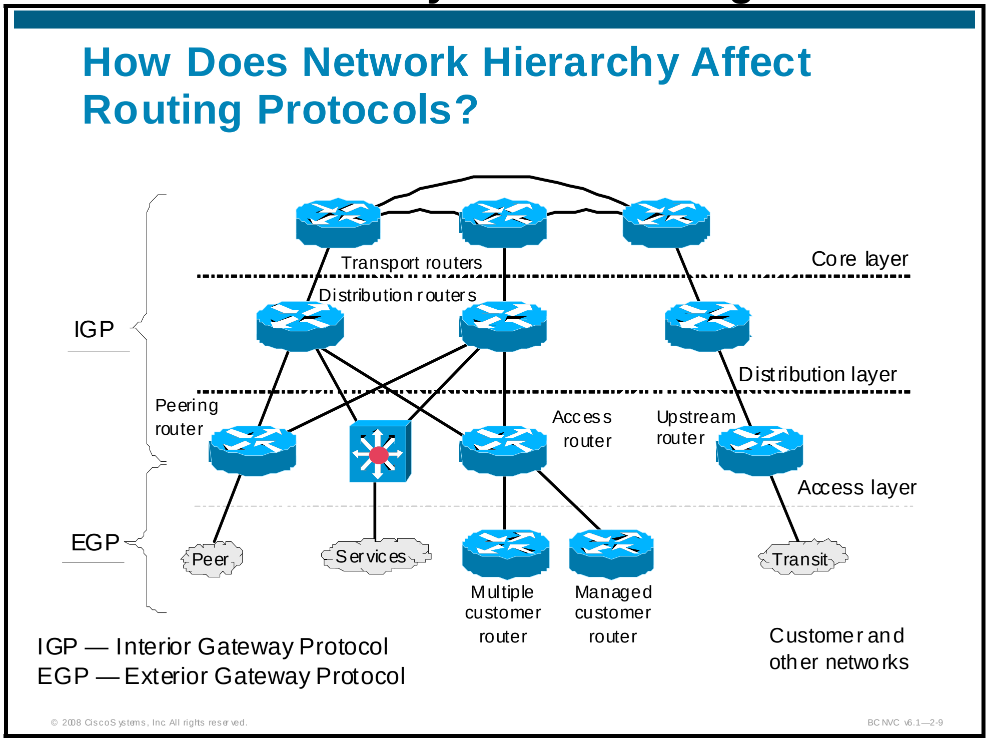
What Is an Interior Gateway Protocol (IGP)?

- IGPs determine how to send packets from device to device between your routers
- Think of interior links within your own network
- IGPs fall into two categories:
  - – Distance vector protocols
    - Routing Information Protocol (RIP)
    - Interior Gateway Routing Protocol (IGRP)
  - – Link-state protocols
    - Open Shortest Path First (OSPF)
    - Intermediate System to Intermediate System (IS-IS)

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—2-8

IGPs handle routing within an autonomous system (AS). Plainly, IGPs decide how to route packets between routers. These protocols keep track of how to get from one destination to another inside a network or set of networks that you administer (all of the networks you manage are usually just one AS). IGPs allow networks to communicate with each other.

How Does Network Hierarchy Affect Routing Protocols?



How Does Network Hierarchy Affect Routing Protocols?

IGP — Interior Gateway Protocol
EGP — Exterior Gateway Protocol

BCNVC v6.1—2-9

The physical topology of an internetwork is described by the complete set of routers and the networks that connect them. Networks also have a logical topology. Different routing protocols establish the logical topology in different ways.

Some routing protocols do not use a logical hierarchy. Such protocols use addressing to segregate specific areas or domains within a given internetworking environment and to establish a logical topology. For such non-hierarchical, or flat, protocols, no manual topology creation is required.

Other protocols require you to create an explicit hierarchical topology through establishing a backbone and logical areas. Open Shortest Path First (OSPF) and Intermediate System-Intermediate System (IS-IS) are examples of routing protocols that use a hierarchical structure. A general hierarchical network scheme is illustrated above. The explicit topology in a hierarchical scheme takes precedence over a topology created through addressing.

As routing protocols have improved, and as routers themselves have become more robust, it is now common practice to deploy routing protocols as follows:

- IGP carries infrastructure prefixes—backbone links and router loopback interface addresses.
- Interior Border Gateway Protocol (IBGP) carries customer-assigned address blocks, access-network address pools, and any other prefixes that do not need to be carried in an IGP. IBGP also is used to carry some or all of the Internet Route Table (depending on the Internet service provider internal policy).
- External Border Gateway Protocol (EBGP) carries prefixes and implements routing policy between ISPs.

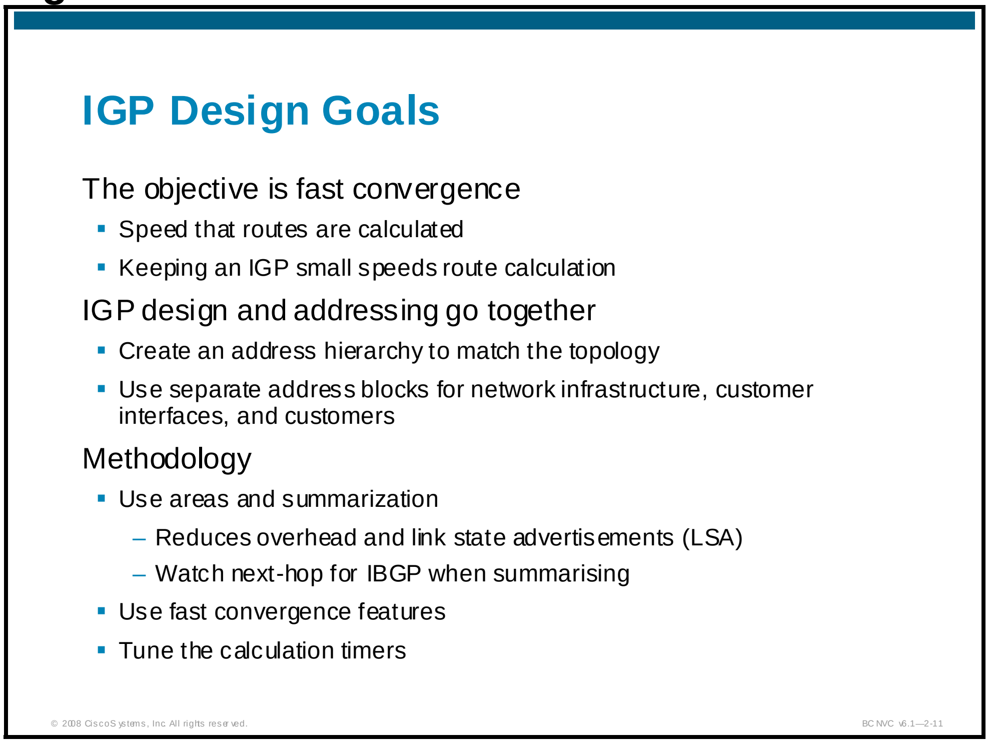
This current routing protocol model is very different from earlier models used in the infancy of the Internet, in which an IGP carried all prefixes in an ISP backbone and BGP was restricted to simply exchanging prefixes between different autonomous systems. The relative lack of scalability of IGPs and the great scalability now available in IBGP through route reflectors and confederations means that IBGP is an excellent tool for carrying prefixes across an ISP backbone.

A typical deployment scenario follows:

- Install all loopback and link addresses in a backbone into the IGP.
- All routers in the backbone participate in the IGP. Any routers that cannot participate in the IGP generally have static default routes pointing to those that can (these are typically access-aggregation devices).
- Configure IBGP across the entire ISP backbone. IBGP is configured to peer using the loopback interfaces. The routers can see each other because there is an entry in the forwarding table courtesy of the IGP.
- The ISP implements a policy for IBGP. All domestic prefixes (the ISP prefixes not in the IGP and address space assigned to its customers) are tagged with a certain community.
- Core routers or route reflectors have filters so that the clients receive only the routes that belong to this special community. This ensures that the core routers (or route reflectors) are the only ones carrying the full routing table and all other IBGP routers carry only the reduced list of prefixes.

This type of deployment makes for a very scalable network and allows the ISP to implement BGP on nearly every router device within its own backbone. This method is much preferred over having an IBGP island in the middle of a network and either pointing static routes or using redistribution from other routing protocols at the edge. The latter has proven risky and unreliable in many situations.

IGP Design Goals



IGP Design Goals

The objective is fast convergence

- Speed that routes are calculated
- Keeping an IGP small speeds route calculation

IGP design and addressing go together

- Create an address hierarchy to match the topology
- Use separate address blocks for network infrastructure, customer interfaces, and customers

Methodology

- Use areas and summarization
  - Reduces overhead and link state advertisements (LSA)
  - Watch next-hop for IBGP when summarising
- Use fast convergence features
- Tune the calculation timers

BCNVC v6.1—2-11

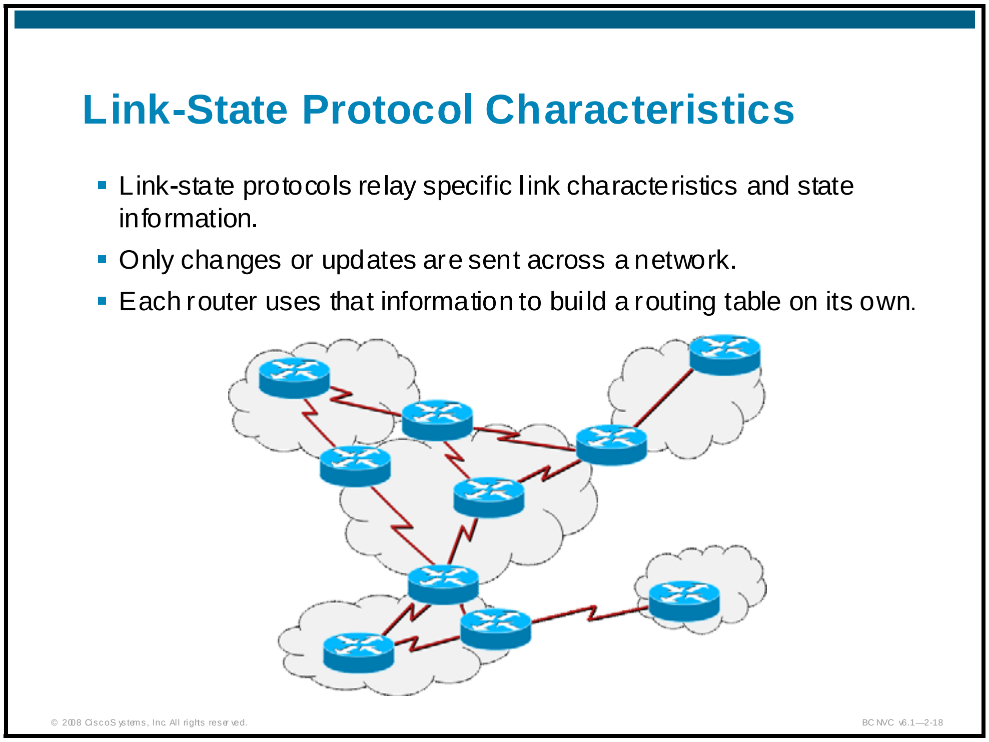
This lesson covers the details of setting up a configuration for two of the popular IGPs: OSPF and IS-IS. If these configuration guidelines are followed, there is very little else to say about IGP choice and configuration. The key to a scalable network is simple: keep the IGP small. BGP is designed to carry a large number of prefixes around an ISP backbone. IBGP is considered by some network engineers to be the interior routing protocol for their network.

IGP recommendations:

- Keep the IGP routing table as small as possible.
- The IGP should only have router loopbacks, backbone wide-area network (WAN) point-to-point link addresses, and network addresses of any local area networks (LANs) having an IGP running on them.
- Use inter-router authentication.
- Use summarization if possible.

What is a Link-State Protocol?

Link-State Protocol Characteristics



Link-State Protocol Characteristics

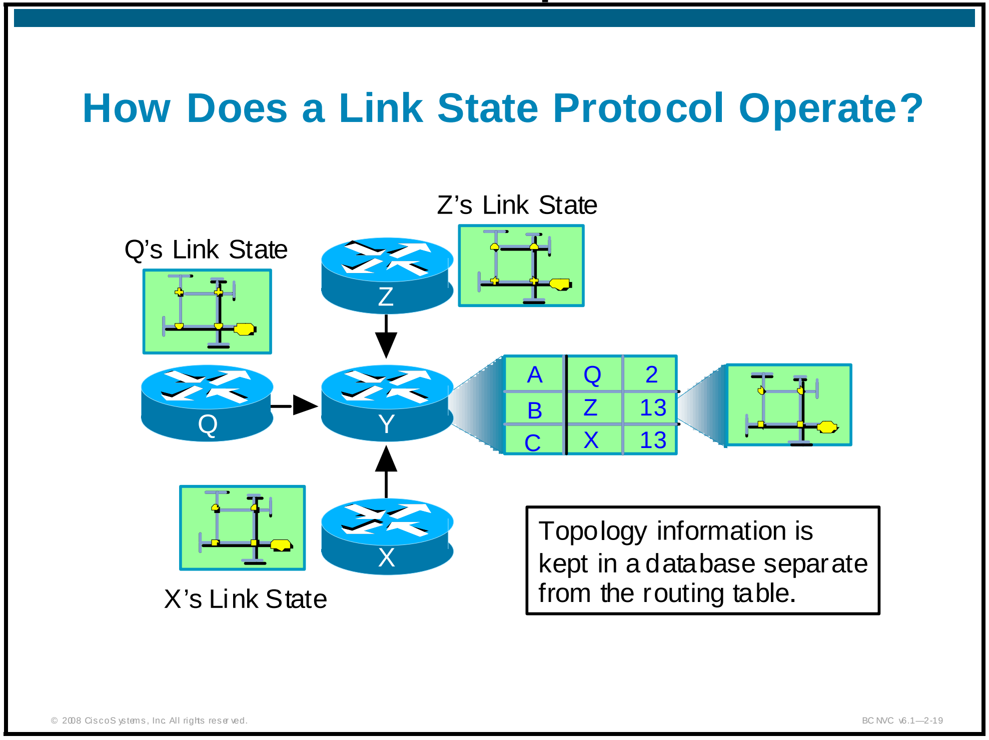
- Link-state protocols relay specific link characteristics and state information.
- Only changes or updates are sent across a network.
- Each router uses that information to build a routing table on its own.

BCMVC v6.1—2-18

Traditional distance-vector protocols relay information regarding their relative distance to a destination. Link-state protocols relay specific link characteristics and state information. Once adjacencies are formed and databases are synchronized, only changes or updates are sent across a network. Each router uses that information to build a routing table on its own.

Two commonly deployed link-state protocols are IS-IS and OSPF.

How Does a Link-state Protocol Operate?



How Does a Link State Protocol Operate?

A	Q	2
B	Z	13
C	X	13

Topology information is kept in a database separate from the routing table.

BCNVC v6.1—2-19

Routers participating in a link-state protocol are uniquely identified throughout a network with a router ID (which is some form of address). Link-state-protocol routers discover their adjacent neighbors with some form of Hello protocol. Once discovered, neighbors form a relationship to exchange and synchronize link-state database (LSDB) information between them by flooding messages. Each router also maintains a database of network information for a complete picture of the network.

The LSDB contains information regarding all links and routers within a logical area. A router has a separate LSDB for each area it belongs to. All routers belonging to the same area have identical databases.

The database relies on the Dijkstra Shortest Path First (SPF) algorithm to calculate a path tree. SPF calculation is performed separately for each area.

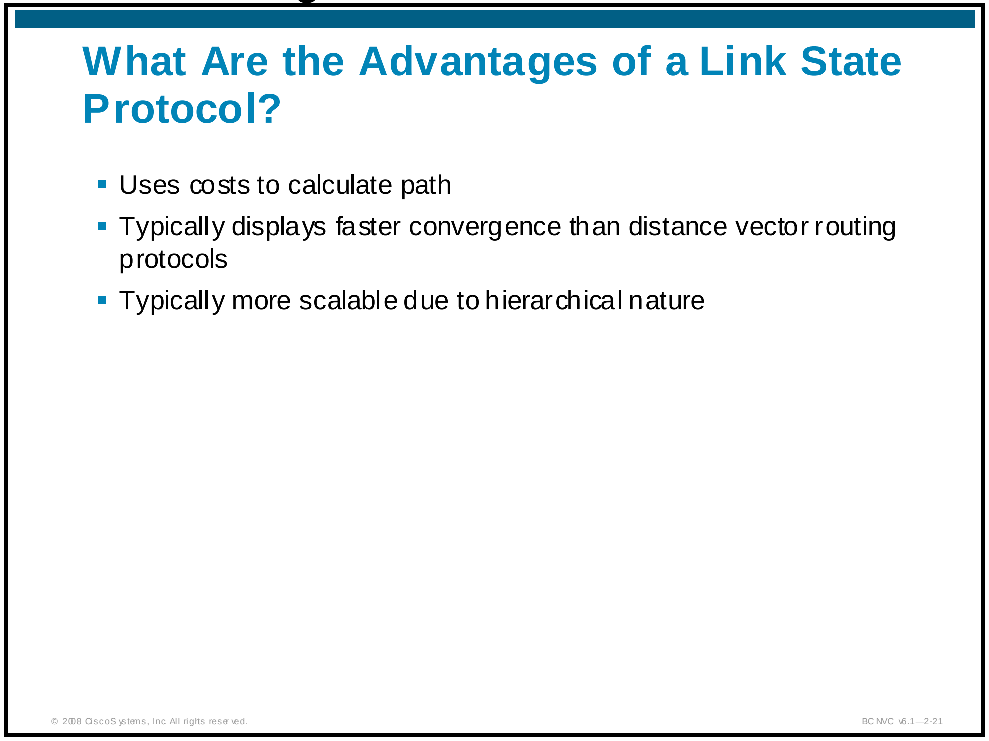
Link-state routing steps include:

1. Discover neighbors
2. Construct a link-state packet (LSP)
3. Distribute LSPs; acknowledge LSPs as needed
4. Compute routes using SPF
5. Flood new LSPs on network change
6. Recompute routing tables (all routers)

The LSDB contains information regarding all links and routers within a logical area. A router has a separate LSDB for each type of adjacency (Layers 1 and 2) it participates in.

All routers belonging to the same area have an identical database. SPF calculation is performed separately for each area.

What Are the Advantages of a Link State Protocol?



What Are the Advantages of a Link State Protocol?

- Uses costs to calculate path
- Typically displays faster convergence than distance vector routing protocols
- Typically more scalable due to hierarchical nature

BCNVC v6.1—2-21

A metric is a path cost that can be assigned by an administrator or calculated using link characteristics information. It is a more flexible numeric value than hop count, used by distance-vector routing protocols.

Because routing changes are propagated instantaneously rather than on a schedule, link-state protocols provide better convergence than distance-vector protocols.

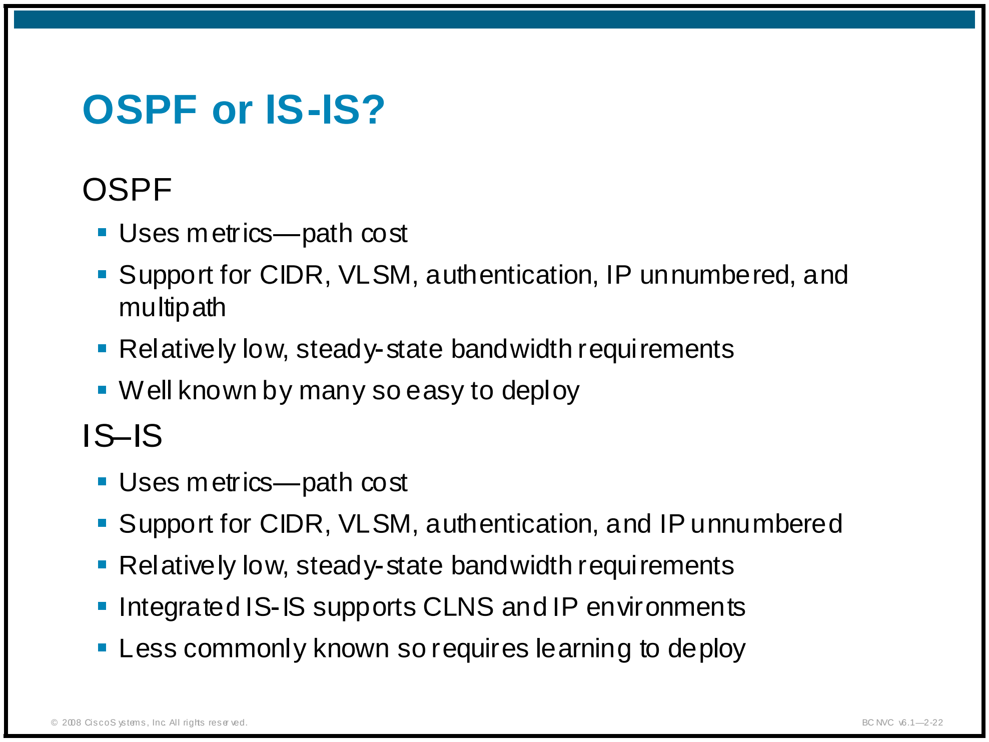
Link-state protocols deploy a logical hierarchy in their design. This hierarchy typically consists of two levels — a backbone level and another sublevel.

IS-IS: Layer 2 (backbone area), Layer 1 areas

OSPF: Area 0 (backbone area), regular areas

This hierarchy enables scalability by allowing summarizing and abstracting, thereby reducing information from lower level areas into the higher-level backbone area.

OSPF or IS-IS?



OSPF or IS-IS?

OSPF

- Uses metrics—path cost
- Support for CIDR, VLSM, authentication, IP unnumbered, and multipath
- Relatively low, steady-state bandwidth requirements
- Well known by many so easy to deploy

IS–IS

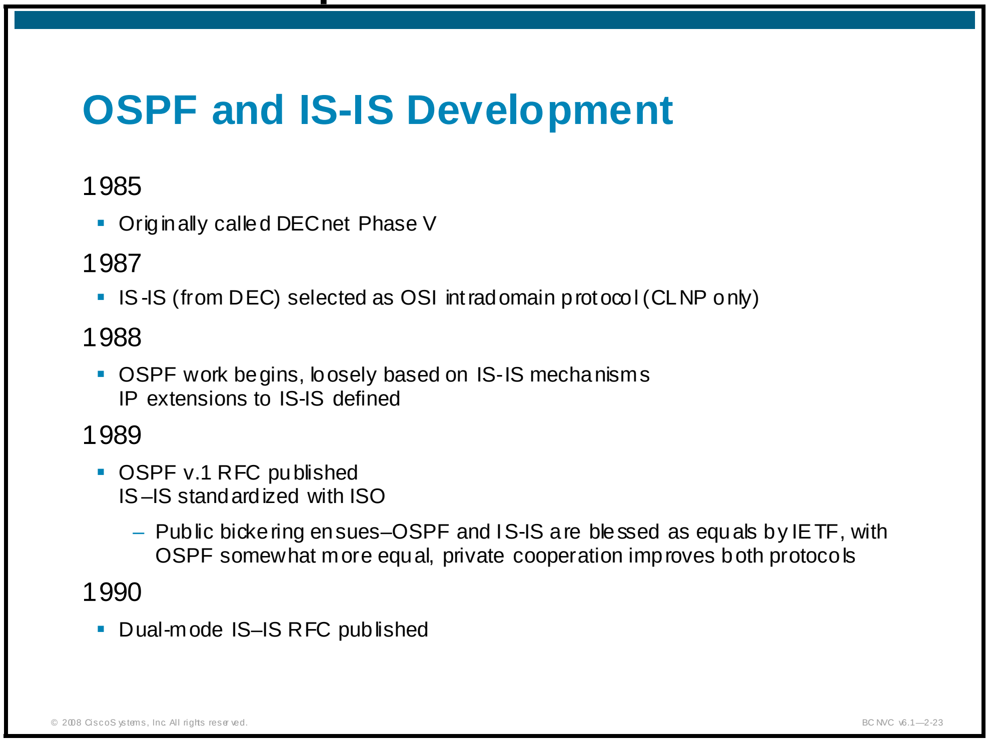
- Uses metrics—path cost
- Support for CIDR, VLSM, authentication, and IP unnumbered
- Relatively low, steady-state bandwidth requirements
- Integrated IS-IS supports CLNS and IP environments
- Less commonly known so requires learning to deploy

CIDR—classless interdomain routing

VLSM—variable-length subnet mask

There is little difference between OSPF and IS-IS for most network implementations today, and the choice between the two depends on which IGP staff has the most operational experience with. If you observe the spread of IS-IS around the Internet, you will see that as engineers leave the established ISPs and move to new jobs, they implement IS-IS as the IGP of choice simply because they are most familiar with it. A similar situation occurs with OSPF.

OSPF and IS-IS Development



OSPF and IS-IS Development

1985

- Originally called DECnet Phase V

1987

- IS-IS (from DEC) selected as OSI intradomain protocol (CLNP only)

1988

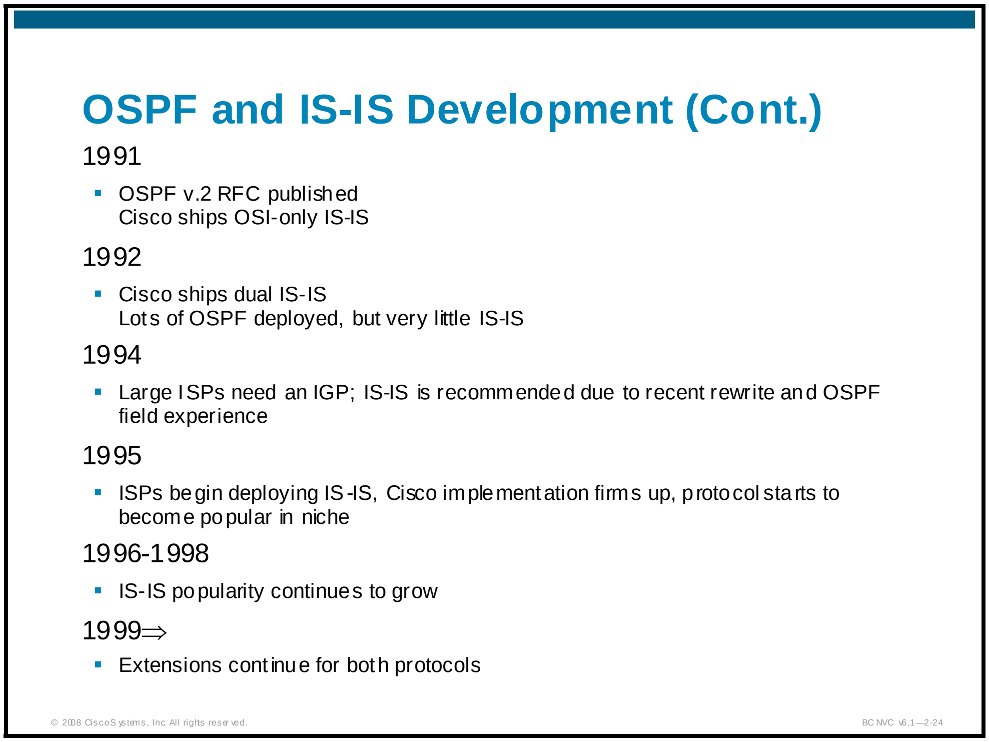
- OSPF work begins, loosely based on IS-IS mechanisms
  IP extensions to IS-IS defined

1989

- OSPF v.1 RFC published
  IS–IS standardized with ISO
  - Public bickering ensues–OSPF and IS-IS are blessed as equals by IETF, with OSPF somewhat more equal, private cooperation improves both protocols

1990

- Dual-mode IS–IS RFC published



OSPF and IS-IS Development (Cont.)

1991

- OSPF v.2 RFC published
  Cisco ships OSI-only IS-IS

1992

- Cisco ships dual IS-IS
  Lots of OSPF deployed, but very little IS-IS

1994

- Large ISPs need an IGP; IS-IS is recommended due to recent rewrite and OSPF field experience

1995

- ISPs begin deploying IS-IS, Cisco implementation firms up, protocol starts to become popular in niche

1996-1998

- IS-IS popularity continues to grow

1999⇒

- Extensions continue for both protocols

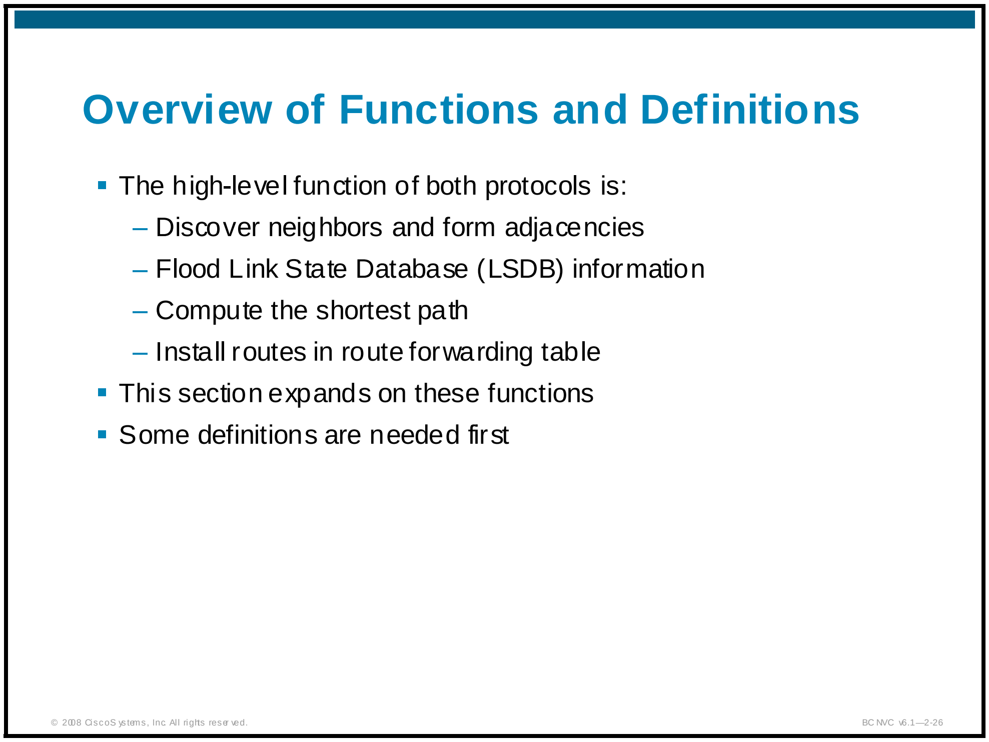
IS-IS was originally one of two protocols defined by the Open Systems Interconnect (OSI) to establish information exchange between network devices. It dictates how routers (also known as independent systems) share routing information.

End System to Intermediate System (ES-IS) specifies how end systems or hosts discover each other and other intermediate systems, and how they exchange information. It is defined in ISO 9542.

Because ES-IS is not used with IP-enabled IS-IS, ES-IS is not discussed in detail in this chapter. Any references in the remainder of this student guide to the network protocol IS-IS are intended to mean Integrated IS-IS or Dual IS-IS. Integrated IS-IS supports pure-IP environments, pure-OSI environments, and environments with both OSI and IP traffic. As further evidence of the IS-IS foundation in OSI, its packets are not encapsulated in IP or Connectionless Network Service (CLNS), but are encapsulated directly into the data-link layer.

How Do Link-State IGPs Work?

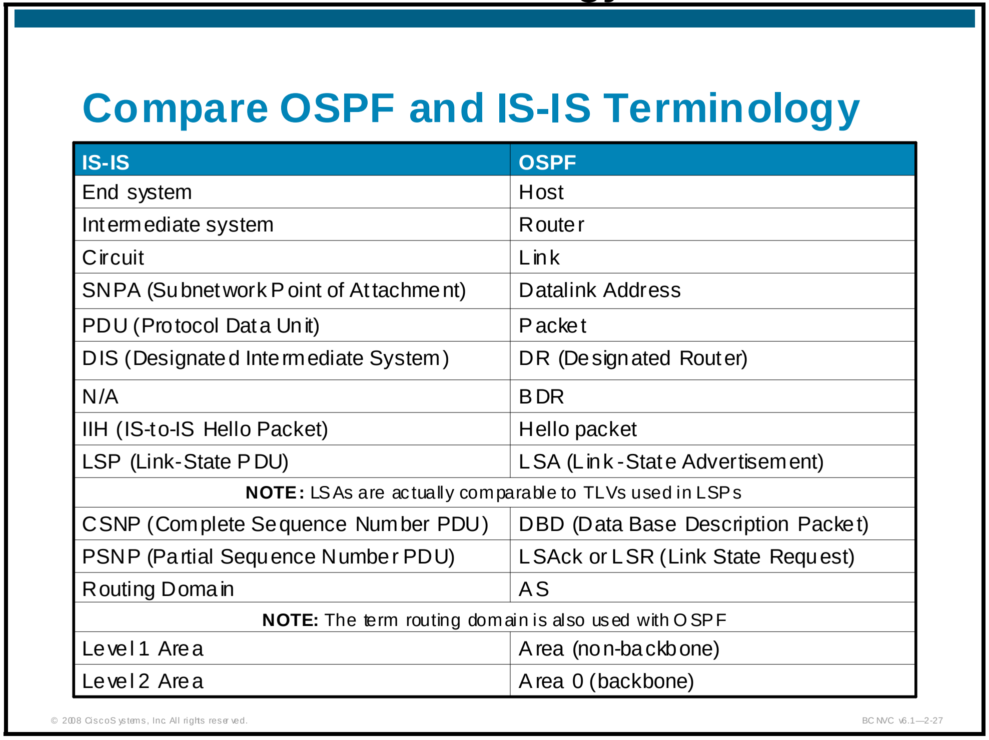
Overview of Functions and Definitions



Overview of Functions and Definitions

- The high-level function of both protocols is:
  - Discover neighbors and form adjacencies
  - Flood Link State Database (LSDB) information
  - Compute the shortest path
  - Install routes in route forwarding table
- This section expands on these functions
- Some definitions are needed first

Compare OSPF and IS-IS Terminology

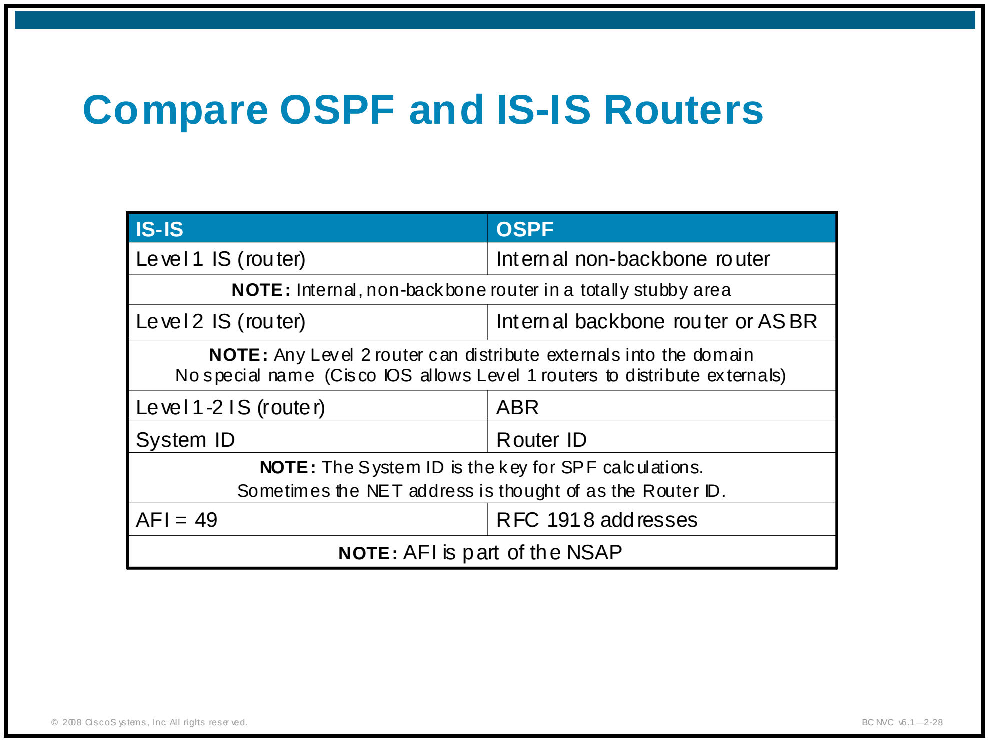


Compare OSPF and IS-IS Terminology

IS-IS	OSPF
End system	Host
Intermediate system	Router
Circuit	Link
SNPA (Subnetwork Point of Attachment)	Datalink Address
PDU (Protocol Data Unit)	Packet
DIS (Designated Intermediate System)	DR (Designated Router)
N/A	BDR
IIH (IS-to-IS Hello Packet)	Hello packet
LSP (Link-State PDU)	LSA (Link-State Advertisement)
NOTE: LSAs are actually comparable to TLVs used in LSPs	
CSNP (Complete Sequence Number PDU)	DBD (Data Base Description Packet)
PSNP (Partial Sequence Number PDU)	LSAck or LSR (Link State Request)
Routing Domain	AS
NOTE: The term routing domain is also used with OSPF	
Level 1 Area	Area (non-backbone)
Level 2 Area	Area 0 (backbone)

Term	Definition
Adjacency	A relationship formed between selected neighboring routers for the purpose of exchanging routing information. Not every pair of neighboring routers becomes adjacent.
Designated router (OSPF)/ Designated IS (IS-IS)	Each multi-access network that has at least two attached routers has a designated router (DR) or designated IS (DIS). The DR generates a link-state advertisement (LSA) for the multi-access network and has other special responsibilities in the running of the protocol. OSPF elects a backup DR (BDR) whereas IS-IS does not. The DIS generates a link-state protocol (LSP) data unit for the broadcast network (known as a virtual node or pseudonode) and has other special responsibilities in the running of the protocol.
End system (ES)	A device attached to the network that receives and transmits packets, but does not forward packets from one physical link (segment) to another. Also called "host."
Hello protocol	The part of the protocol used to establish and maintain neighboring relationships. The hello protocol can also dynamically discover neighboring routers.
Intermediate System (IS)	A device that connects multiple physical links (networks), forwarding packets between these links as needed. Also known as a "router."
IS-IS Hello packet (IIH)	The part of the protocol used to establish and maintain neighboring relationships. The IIH can also dynamically discover neighboring routers.
Level 1 area	An area containing Level 1 routers.
Level 2 area	An area containing Level 2 routers. A contiguous collection of Level 2–capable routers linking Level 1 areas is the IS-IS backbone.
Link-state advertisement	Describes the local state of the router or network. This includes the state of the router interfaces and adjacencies. Each link-state advertisement is flooded throughout the routing domain. The collected link-state advertisements of all routers and networks form the protocol topological database.
Link-state protocol (LSP) data units	Because IS-IS is based on CLNS rather than IP, information is exchanged between routers with protocol data units (PDUs). Link-state PDUs (LSPs) describe the local state of the router or network, including the state of router interfaces and adjacencies. Each link-state PDU is flooded throughout the area. The collected LSPs of all routers and networks form the protocol topology database.
Neighboring routers	Two routers that have interfaces to a common network. Neighbors are dynamically discovered with a hello protocol.
Network entity title (NET)	CLNS-based number assignment used to uniquely identify each intermediate system (aka router) on the internetwork. The NET consists of an area ID, system ID, and NSAP selector. Routers use this number to identify themselves when generating updates.
Router ID	A 32-bit number assigned to each router running the OSPF protocol. This number uniquely identifies the router within an autonomous system. Routers use this number to identify themselves when generating updates.
Routing domain	An IS-IS routing domain is a network in which all the routers run the Integrated IS-IS routing protocol to support intradomain exchange of routing information.
Subnetwork point of attachment (SNPA)	An SNPA is the point at which subnetwork services are provided. This is the equivalent of the Layer 2 address corresponding to the Layer 3 (NET or NSAP) address.
Topology/link-state database	A router has a separate link-state database for each area or level to which it belongs. All routers belonging to the same area have identical database. The SPF calculation is performed separately for each area and LSA/LSP flooding is bounded by area.

Compare OSPF and IS-IS Routers

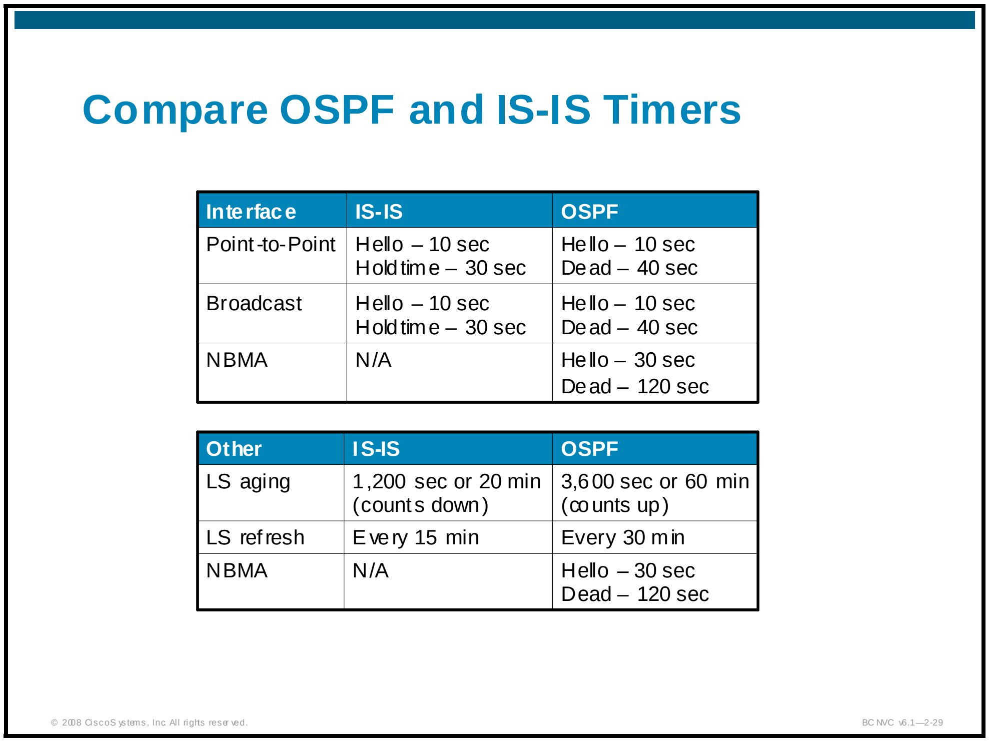


Compare OSPF and IS-IS Routers

IS-IS	OSPF
Level 1 IS (router)	Internal non-backbone router
NOTE: Internal, non-backbone router in a totally stubby area	
Level 2 IS (router)	Internal backbone router or ASBR
NOTE: Any Level 2 router can distribute externals into the domain No special name (Cisco IOS allows Level 1 routers to distribute externals)	
Level 1-2 IS (router)	ABR
System ID	Router ID
NOTE: The System ID is the key for SPF calculations. Sometimes the NET address is thought of as the Router ID.	
AFI = 49	RFC 1918 addresses
NOTE: AFI is part of the NSAP	

BCNVC v6.1—2-28

Compare OSPF and IS-IS Timers



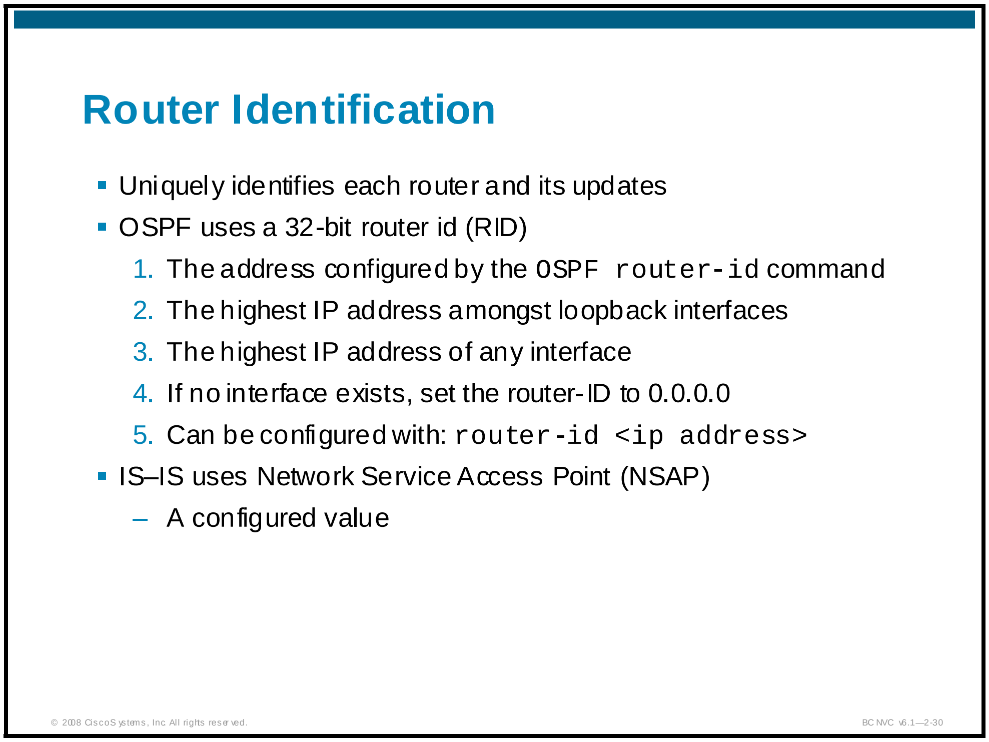
Compare OSPF and IS-IS Timers

Interface	IS-IS	OSPF
Point-to-Point	Hello – 10 sec Holdtime – 30 sec	Hello – 10 sec Dead – 40 sec
Broadcast	Hello – 10 sec Holdtime – 30 sec	Hello – 10 sec Dead – 40 sec
NBMA	N/A	Hello – 30 sec Dead – 120 sec

Other	IS-IS	OSPF
LS aging	1,200 sec or 20 min (counts down)	3,600 sec or 60 min (counts up)
LS refresh	Every 15 min	Every 30 min
NBMA	N/A	Hello – 30 sec Dead – 120 sec

BCNVC v6.1—2-29

Router Identification



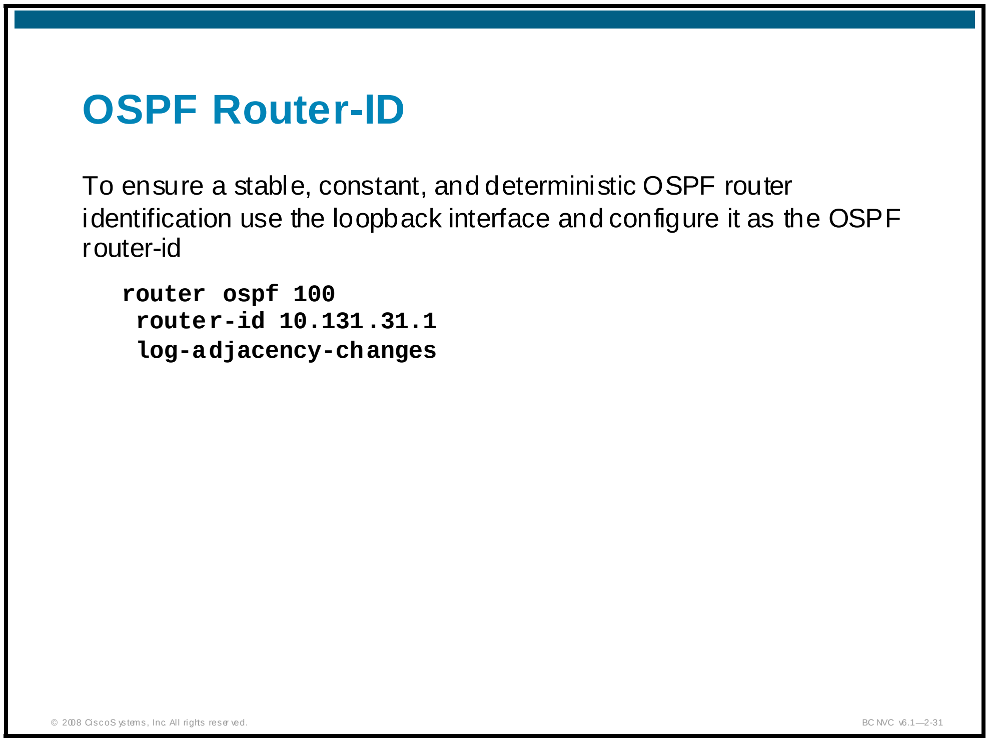
Router Identification

- Uniquely identifies each router and its updates
- OSPF uses a 32-bit router id (RID)
  1. The address configured by the `OSPF router-id` command
  2. The highest IP address amongst loopback interfaces
  3. The highest IP address of any interface
  4. If no interface exists, set the router-ID to 0.0.0.0
  5. Can be configured with: `router-id <ip address>`
- IS–IS uses Network Service Access Point (NSAP)
  – A configured value

BCNVC v6.1—2-30

Routers participating in a link-state protocol are uniquely identified throughout a network with some form of address, a router ID (RID), or network service access point (NSAP).

OSPF Router-ID



OSPF Router-ID

To ensure a stable, constant, and deterministic OSPF router identification use the loopback interface and configure it as the OSPF router-id

```
router ospf 100
 router-id 10.131.31.1
 log-adjacency-changes
```

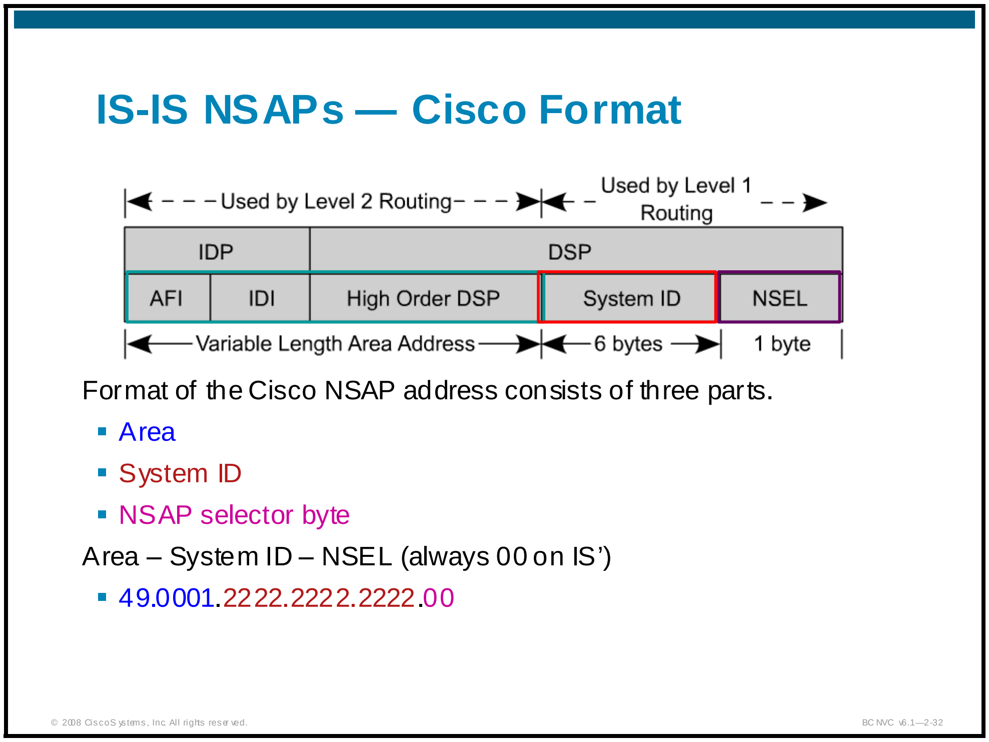
BCNVC v6.1—2-31

If a loopback interface is configured on a router, its IP address should be used as the router ID. This is important for ensuring stability and predictability in an ISP network.

OSPF chooses the designated router (DR) on a LAN as the device that has the highest IP address. If routers are added or removed from the LAN, or if a router gains an interface with a higher address than that of the existing DR, the DR likely will change if the DR or backup designated router (BDR) fails. This generally is undesirable in an ISP network because ISPs prefer to have the DR and BDR routers established deterministically. This change in DR and BDR can be avoided by ensuring that the loopback interface is configured and in use on all routers on a LAN.

If not configured, because of the ever-changing nature of an ISP network, this value can change, possibly resulting in operational confusion. Configuring and using a loopback interface ensures stability.

IS-IS NSAPs — Cisco Format



IS-IS NSAPs — Cisco Format

Format of the Cisco NSAP address consists of three parts.

- Area
- System ID
- NSAP selector byte

Area – System ID – NSEL (always 00 on IS')

- 49.0001.2222.2222.2222.00

NSAP prefixes are required for CLNS routing, including IP-only networks. Even in IP-only networks, IS-IS uses OSI addresses to identify the router, build the topology, build the SPF tree, and identify LSPs.

IS-IS NSAP Area – Cisco Format

Addresses starting with 49 (AFI=49) are considered private IP address, analogous to RFC 1918. These are routed by IS-IS and should not be advertised outside the IS-IS domain. All routers in the same area must have the same area address. An additional 2 bytes (high-order DSP) are added for the area ID.

IS-IS NSAP System ID – Cisco Format

OSI requires the IS-IS NSAP system ID to be the same number of bytes throughout a domain. Cisco fixes the system ID at 6 bytes. It is customary to use either a MAC address from the router or IP address of the loopback interface.

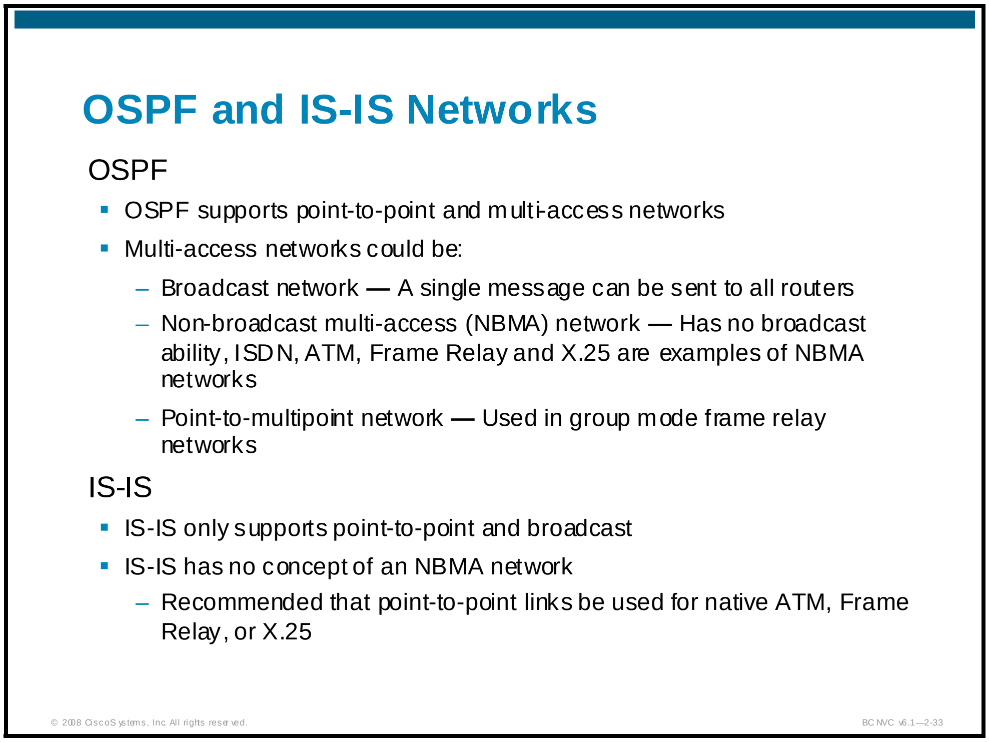
For example: 192.168.111.3 -> 192.168.111.003 -> 1921.6811.1003

IS-IS NSAP NSEL – Cisco Format

NSEL (NSAP Selector) is a service identifier loosely equivalent to a port or socket in TCP/IP. It is not used in routing decisions.

When NSEL=00, it identifies the device itself, that is, the network-level address. The NSAP with a NSEL=00 is known as a network entity title (NET). A NET is an NSAP with the NSEL set to 00.

OSPF and IS-IS Networks



OSPF and IS-IS Networks

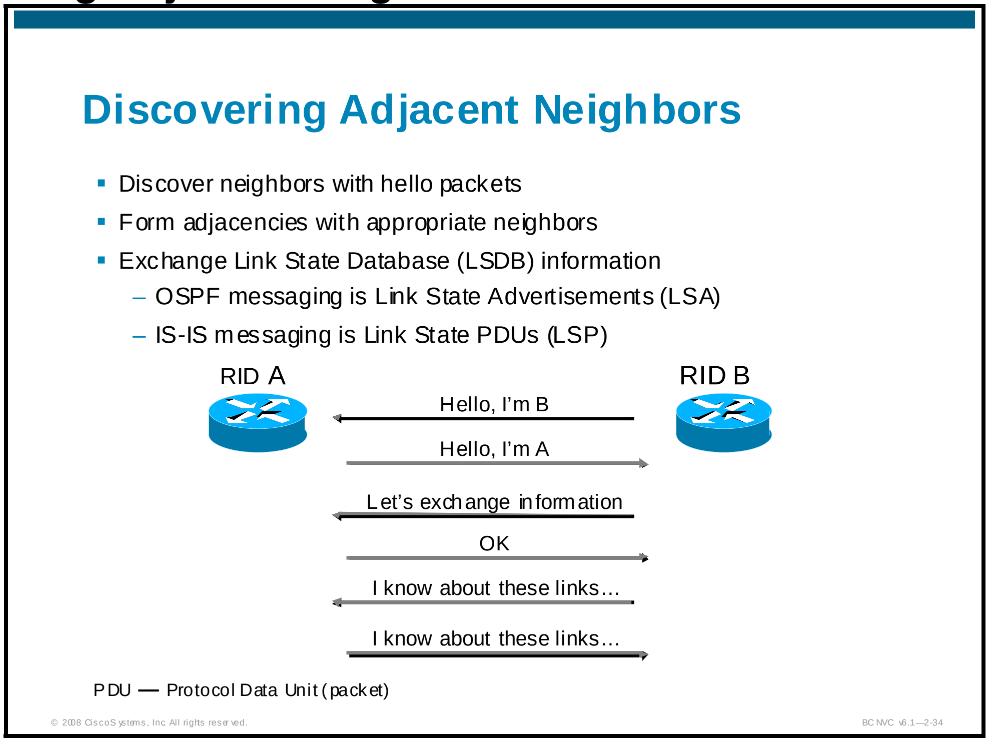
OSPF

- OSPF supports point-to-point and multi-access networks
- Multi-access networks could be:
  - Broadcast network — A single message can be sent to all routers
  - Non-broadcast multi-access (NBMA) network — Has no broadcast ability, ISDN, ATM, Frame Relay and X.25 are examples of NBMA networks
  - Point-to-multipoint network — Used in group mode frame relay networks

IS-IS

- IS-IS only supports point-to-point and broadcast
- IS-IS has no concept of an NBMA network
  - Recommended that point-to-point links be used for native ATM, Frame Relay, or X.25

Discovering Adjacent Neighbors



Discovering Adjacent Neighbors

- Discover neighbors with hello packets
- Form adjacencies with appropriate neighbors
- Exchange Link State Database (LSDB) information
  - OSPF messaging is Link State Advertisements (LSA)
  - IS-IS messaging is Link State PDUs (LSP)

RID A
RID B
Hello, I'm B
Hello, I'm A
Let's exchange information
OK
I know about these links…
I know about these links…

PDU — Protocol Data Unit (packet)

BCNVC v6.1—2-34

Link-state-protocol routers discover their adjacent neighbors with some form of hello protocol. The hello protocol forms adjacencies between routers and describes the optional capabilities.

OSPF Hello

OSPF multicasts to 224.0.0.5 on all router interfaces. The OSPF hello interval is 10 seconds on a LAN and 30 seconds with nonbroadcast multiaccess (NBMA).

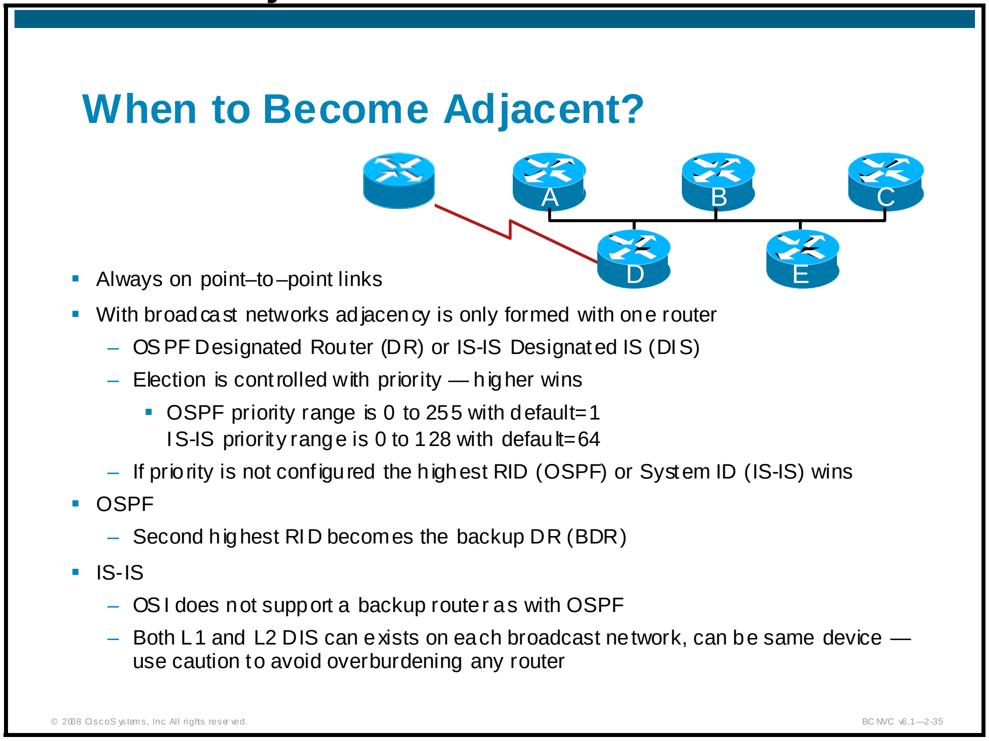
Once an adjacency is established, a router sends network information in a link-state advertisement (LSA) to its neighbor.

ISIS Hello

The IS-IS hello interval is 10 seconds. IS-IS hello packets (IIHs) contain router information including the sending router NET, hello interval, holdtime, PDU length, and priority. Unlike OSPF, hello interval and holdtime do not have to match between IS-IS devices for them to become adjacent.

Once an adjacency is established, a router sends network information in a link-state protocol data unit PDU to its neighbor.

When to Become Adjacent?



When to Become Adjacent?

- Always on point–to–point links
- With broadcast networks adjacency is only formed with one router
  - OSPF Designated Router (DR) or IS-IS Designated IS (DIS)
  - Election is controlled with priority — higher wins
    - OSPF priority range is 0 to 255 with default=1
      IS-IS priority range is 0 to 128 with default=64
  - If priority is not configured the highest RID (OSPF) or System ID (IS-IS) wins
- OSPF
  - Second highest RID becomes the backup DR (BDR)
- IS-IS
  - OSI does not support a backup router as with OSPF
  - Both L1 and L2 DIS can exists on each broadcast network, can be same device — use caution to avoid overburdening any router

BCNVC v6.1—2-35

OSPF

Discovered neighbors form a relationship to exchange and synchronize LSDB information.

Neighbors become adjacent when the:

- Underlying network is point-to-point
- Underlying network type is virtual link
- The router itself is the designated router
- The router itself is the backup designated router
- The neighboring router is the designated router
- The neighboring router is the backup designated router

In the case of a multiaccess network, because all routers are neighbors of each other, it doesn't make much sense to flood all LSAs to every router.

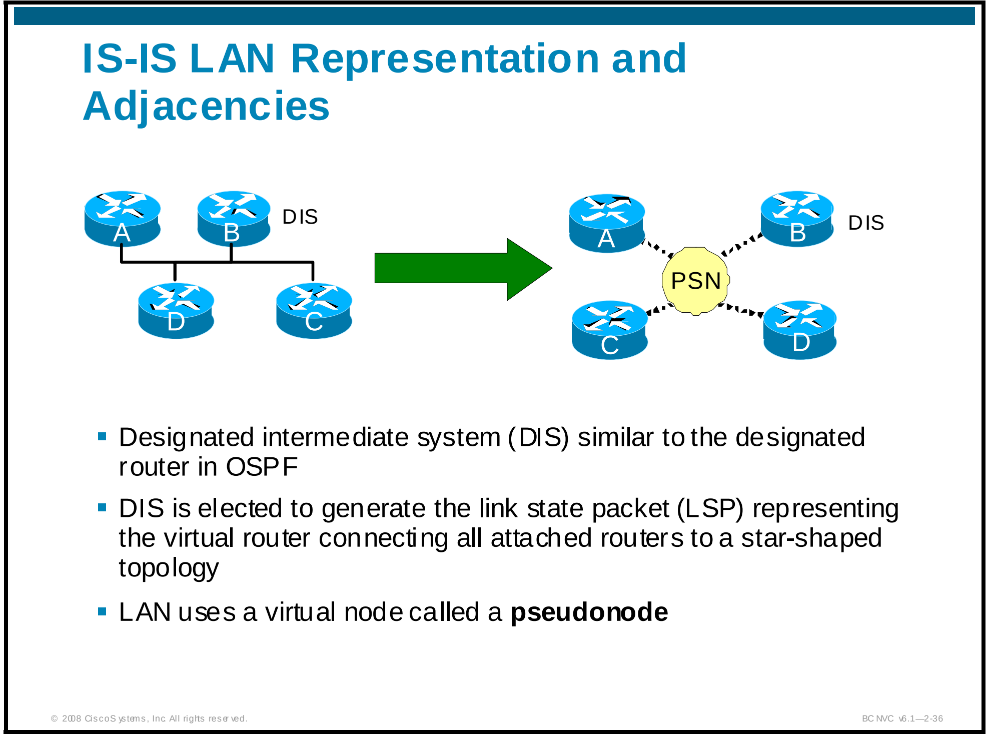
Therefore, to reduce OSPF traffic on multi-access links, not all routers flood LSAs. A designated router (DR) stores and distributes neighbor LSDBs. The DR is selected by priority and a backup designated router (BDR) is selected for redundancy.

IS-IS

IS-IS neighbors on a broadcast network establish an adjacency for each level of routing performed with the neighbor. Two adjacencies are built and maintained if the routers are both Layers 1 and 2 – one adjacency at Layer 1, another at Layer 2. Layers 1 and 2 neighbors never become adjacent because a mix of Layer 1 and Layer 2 devices may coexist on a broadcast network, and different neighbors may become adjacent for Layers 1 or 2. IS-IS routers send separate Layers 1 and 2 hello PDUs to support such dynamic relationships. A Layer 1/Layer 2 router on a broadcast network must send and receive both types of hello PDUs.

Routers on a point-to-point network can only become adjacent with each other. Even though they may both be Layer 1or Layer 2 routers, both adjacencies in this scenario are the same because there are only two neighbors. In this case, routers only send one hello PDU over a point-to-point network, which contains adequate information for both Layers 1 and 2 adjacencies to form, if necessary.

IS-IS LAN Representation and Adjacencies



IS-IS LAN Representation and Adjacencies

- Designated intermediate system (DIS) similar to the designated router in OSPF
- DIS is elected to generate the link state packet (LSP) representing the virtual router connecting all attached routers to a star-shaped topology
- LAN uses a virtual node called a **pseudonode**

BCNVC v6.1—2-36

IS-IS has potential drawbacks when running over a LAN. One of these drawbacks results from the fact that each router on the LAN needs to announce a link to every other router. This could result in an IS-IS router having a table containing n*(n–1) links. Another potential drawback is the fact that each router on the LAN reports the same list of end systems (ESs) to each other, resulting in an enormous amount of duplication.

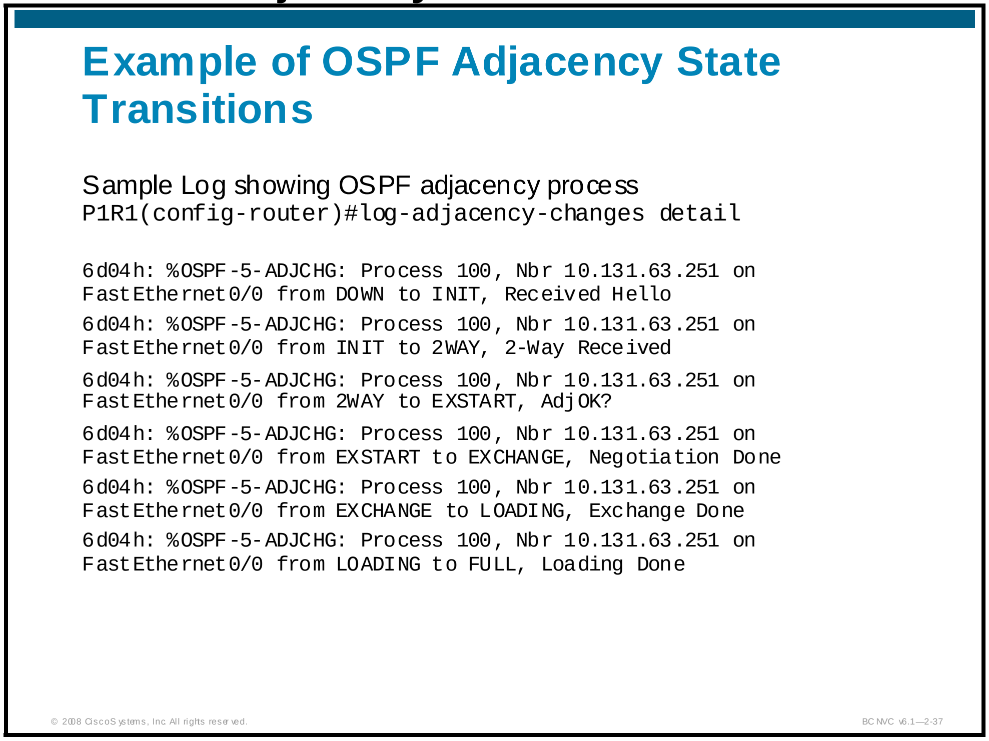
To combat these situations, IS-IS introduces a concept of virtual nodes, known as pseudonodes. A pseudonode is an IS on a link whose purpose is to reduce the number of full-mesh adjacencies required between nodes on a multi-access link. This node is called the designated IS (DIS). All routers on a multiaccess link, including the one elected the DIS, form adjacencies with the pseudonode instead of forming n*(n–1)–order adjacencies with each other in a full mesh. Only the pseudonode LSP includes the list of ESs on the LAN, eliminating the potential duplication problems.

The election process for the DIS is based on the interface priority; the default is 64. The node with the highest interface priority is elected the DIS. In the case of a tie in interface priorities, the router with the highest subnetwork point of attachment (SNPA) is selected. In IS-IS, the media access control (MAC) addresses are used as SNPAs on LANs. On nonbroadcast networks such as Frame Relay, the SNPA is the local data-link connection identifier (DLCI). In the case of multipoint Frame Relay scenarios that have the DLCI value, the highest system ID is used as a tiebreaker, independent of area ID.

You can influence this election by configuring the priority used by your router. You can configure these priorities for Level-1 and Level-2 elections separately. Use the following command to specify the value to use in the designated router election:

```
(config-if)#isis priority value {level-1 | level-2}
```

Example of OSPF Adjacency State Transitions



Example of OSPF Adjacency State Transitions

Sample Log showing OSPF adjacency process

```
P1R1(config-router)#log-adjacency-changes detail

6d04h: %OSPF-5-ADJCHG: Process 100, Nbr 10.131.63.251 on
FastEthernet0/0 from DOWN to INIT, Received Hello
6d04h: %OSPF-5-ADJCHG: Process 100, Nbr 10.131.63.251 on
FastEthernet0/0 from INIT to 2WAY, 2-Way Received
6d04h: %OSPF-5-ADJCHG: Process 100, Nbr 10.131.63.251 on
FastEthernet0/0 from 2WAY to EXSTART, AdjOK?
6d04h: %OSPF-5-ADJCHG: Process 100, Nbr 10.131.63.251 on
FastEthernet0/0 from EXSTART to EXCHANGE, Negotiation Done
6d04h: %OSPF-5-ADJCHG: Process 100, Nbr 10.131.63.251 on
FastEthernet0/0 from EXCHANGE to LOADING, Exchange Done
6d04h: %OSPF-5-ADJCHG: Process 100, Nbr 10.131.63.251 on
FastEthernet0/0 from LOADING to FULL, Loading Done
```

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—2-37

When OSPF adjacency is formed, a router goes through several state changes before it becomes fully adjacent with its neighbor. Following is an explanation of each state.

Down

This is the first OSPF neighbor state. It means that no information has been received from this neighbor, but hello packets can still be sent to the neighbor. If a router doesn't receive a hello packet from a neighbor within the RouterDeadInterval time (RouterDeadInterval = 4*HelloInterval by default), then the neighbor state changes from Full to Down.

Attempt

This state is only valid for neighbors in an NBMA environment. Attempt means that the router is sending hello packets to the neighbor, but has not yet received any information.

Init

This state specifies that the router has received a hello packet from its neighbor, but the receiving router ID was not included in the hello packet. When a router receives a hello packet from a neighbor, it should list the sender router ID in its hello packet as an acknowledgment that it received a valid packet.

2-Way

This designates that bidirectional communication has been established between two routers, that is, each router has seen the hello packet from the other router. In this state, a router decides whether to become adjacent with its neighbor. On broadcast media, a router becomes full only with the designated router (DR) and the backup designated router (BDR); it stays in the two-way state with all other neighbors.

Exstart

This is the first state in forming an adjacency. It is used to elect the master and slave, and to choose the initial sequence number for adjacency formation. The router with the higher router ID becomes the master, and as such, is the only router that can increment the sequence number.

Exchange

In the exchange state, OSPF routers exchange link-state advertisement (LSA) information. Each database description (DBD) packet has a sequence number that is explicitly acknowledged. Routers also send link-state request packets and link-state update packets (which contain the entire LSA) in this state.

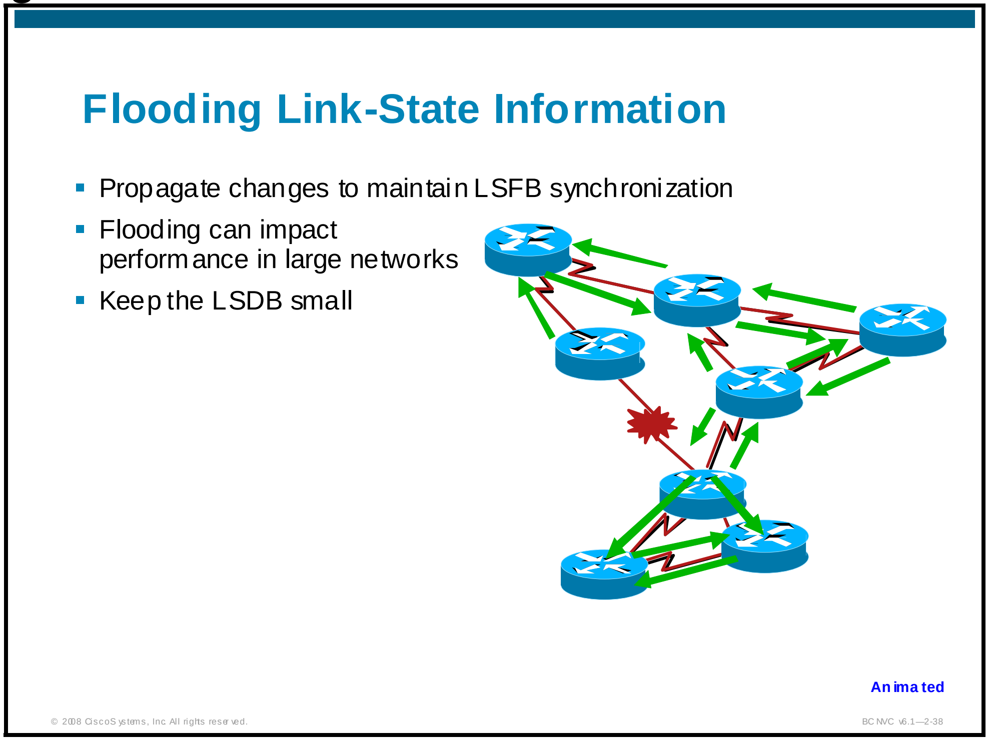
Loading

In the loading state, routers send link-state request packets. During the adjacency, if a router receives an outdated or missing LSA, it requests that LSA by sending a link-state request packet.

Full

Routers are fully adjacent with each other and the databases are fully synchronized. Full is the normal state for an OSPF router. If a router is stuck in another state, it indicates that there are problems in forming adjacencies. Except in a broadcast network, routers achieve full state only with the DR and BDR. Other routers always see each other as two-way.

Flooding Link-State Information



Flooding Link-State Information

- Propagate changes to maintain LSFB synchronization
- Flooding can impact performance in large networks
- Keep the LSDB small

Animated

BCNVC v6.1—2-38

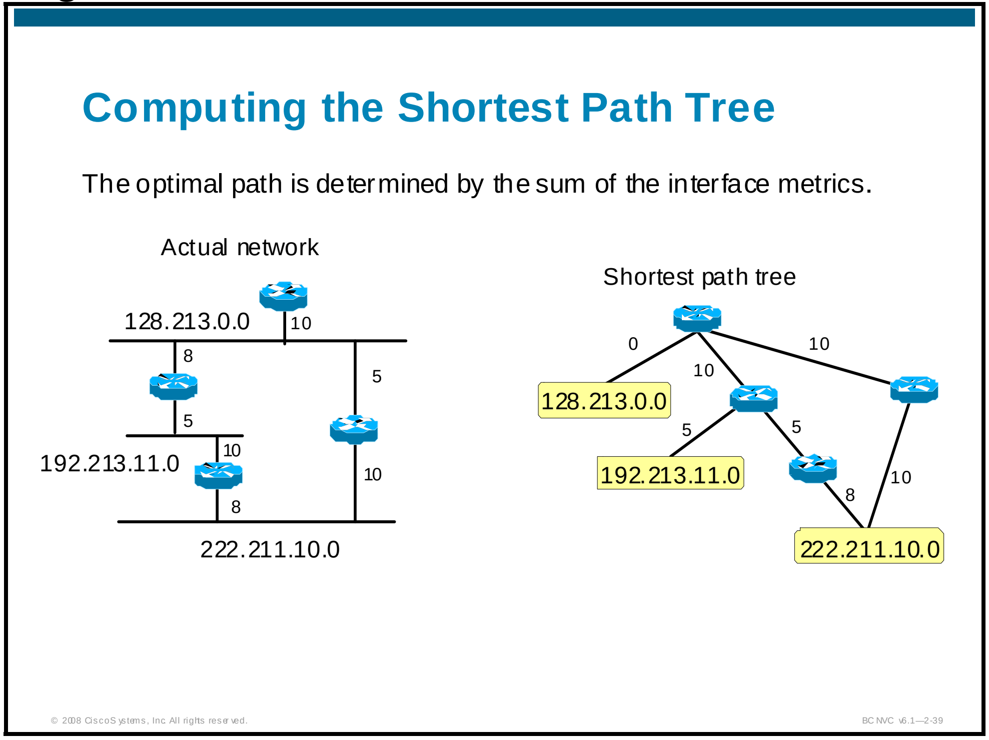
Network changes generate LSAs in the case of OSPF and LSPs in the case of IS-IS. In both cases these updates are flooded across a logical network area.

This is done to maintain LSDB consistency across all routers. The protocol remains relatively quiet during steady-state conditions.

- LSAs are refreshed every 30 minutes by default
- LSPs are refreshed every 15 minutes by default
- Otherwise, updates are only sent when there are changes

As a network grows, and more routers participate in flooding, the network loses the ability to scale. The network can be segmented into multiple areas to reduce the number of adjacencies. This minimizes the LSDB and optimizes convergence times within each area.

Computing the Shortest Path Tree



Computing the Shortest Path Tree

The optimal path is determined by the sum of the interface metrics.

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—2-39

Cost is a 16-bit positive number between 1 and 65,535 where a lower number is the more desirable metric. Cost is applied on all router link paths and route decisions are made on the total cost of a path. Metric is only relevant on an outbound path (route decisions are not made for inbound traffic). Cost is more flexible than hops used in distance-vector routing protocols since it can be administratively controlled.

OSPF

OSPF assigns cost to links based on the link bandwidth according to the following formula:

- Cost = 10^8 / bandwidth

If a network contains high-bandwidth links (155 Mbps or more), the automatic cost assignment does not work anymore (it would result in all costs being equal to 1). In this case, you have two options:

1. The OSPF costs can be set manually on each interface.

```
ip ospf cost <value>
```

2. The global OSPF cost equation can be changed to more appropriately calculate cost specific to the high-speed network. This feature was available as of 11.1CC, 11.2(5).

```
ospf auto-cost reference-bandwidth <reference-bandwidth-in
  Mbps->
```

- Default is 100 (for backward compatibility)

IS-IS

There are four routing metrics defined in ISO 10589/RFC 1142. Only the default metric is supported on Cisco routers.

1. Default metric: Understood by every IS in the domain. The metric value can be associated with any objective function of a circuit but is meant to measure the traffic handling capacity of the circuit. A higher default metric indicates a lower circuit capacity. The default value is 10.

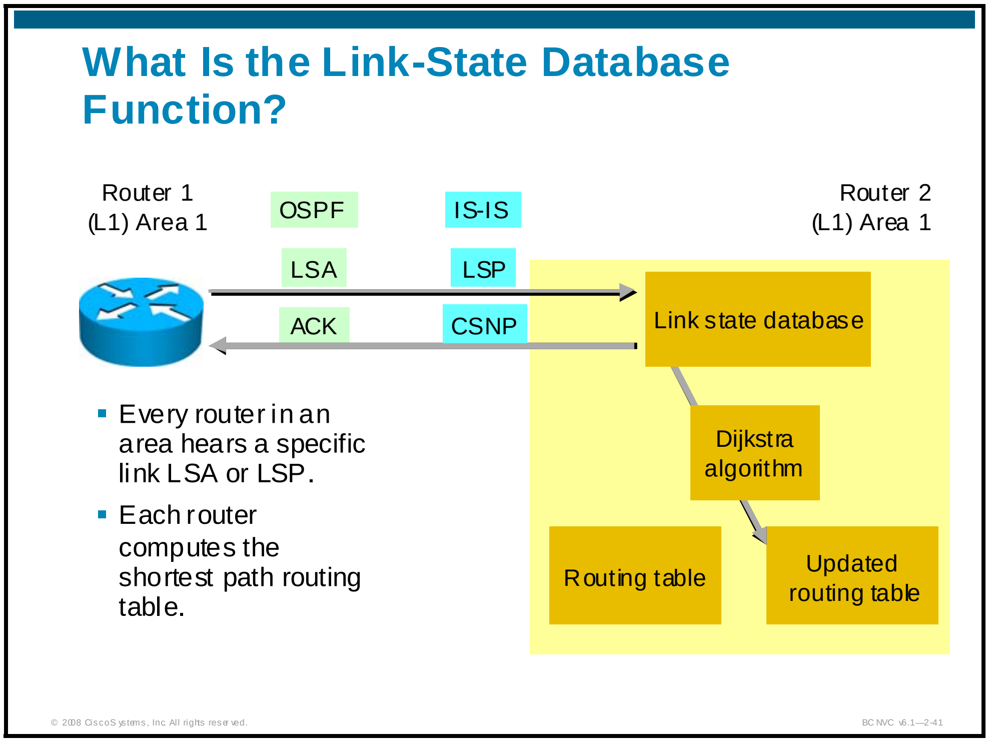
2. Delay metric: Measures the transit delay of a circuit.

3. Expense metric: Measures the monetary cost of using the associated circuit.

4. Error metric: Measures the residual error probability of the associated circuit.

OSPF Link State Database Synchronization

What Is the Link-State Database Function?



What Is the Link-State Database Function?

Router 1
(L1) Area 1

OSPF

IS-IS

Router 2
(L1) Area 1

LSA

LSP

ACK

CSNP

Link state database

Dijkstra algorithm

Routing table

Updated routing table

- Every router in an area hears a specific link LSA or LSP.
- Each router computes the shortest path routing table.

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—2-41

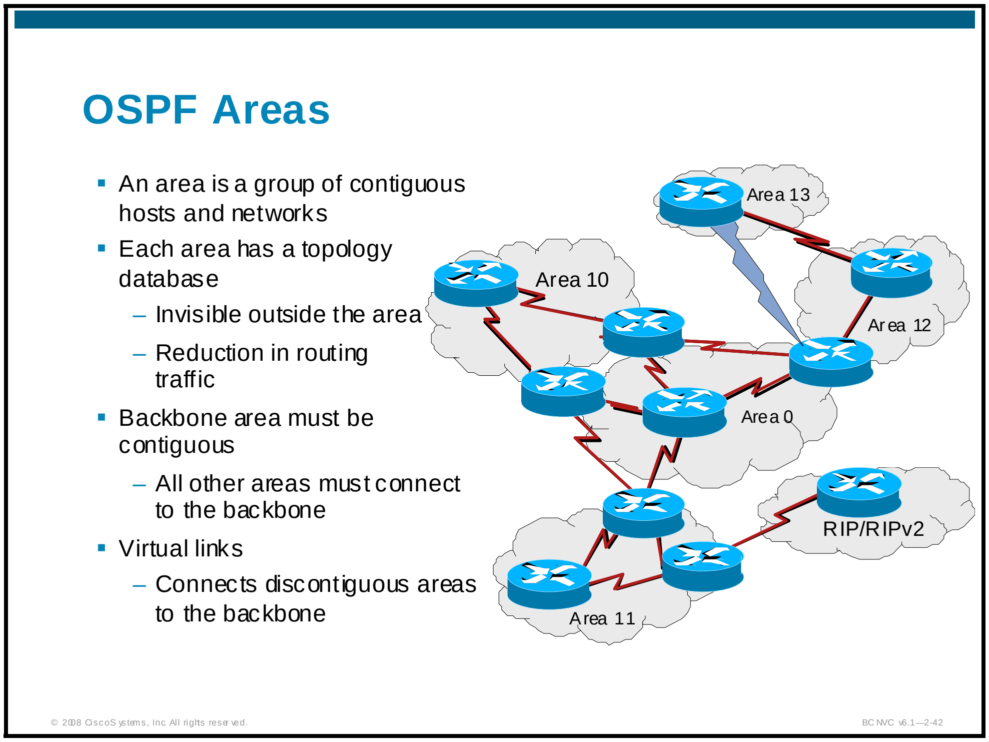
Note Djikstra's algorithm (named after its discover, E.W. Dijkstra) solves the problem of finding the shortest path from a point in a graph (the source) to a destination. You can find the shortest path from a given source to all points in a graph at the same time, hence this problem is sometimes called the single-source shortest-path problem.

A router has a separate LSDB for each area or level to which it belongs. All routers belonging to the same area, at the same level (Level 1 or Level 2) in the case of IS-IS should have identical databases. When a change occurs in the network, LSA or LSP flooding is bounded by area. The new LSAs or LSPs are loaded into the LSDB, and SPF calculation is performed to determine current best paths for the routing table. This activity occurs independently for each area.

There are many configurable timers that, when tuned wisely by informed administrators, can increase network convergence efficiency.

Configure and Verify OSPF

OSPF Areas



OSPF Areas

- An area is a group of contiguous hosts and networks
- Each area has a topology database
  - Invisible outside the area
  - Reduction in routing traffic
- Backbone area must be contiguous
  - All other areas must connect to the backbone
- Virtual links
  - Connects discontiguous areas to the backbone

BCNVC v6.1—2-42

As a network grows, and more routers participate in flooding, the network loses the ability to scale. You can segment an OSPF network into multiple areas to reduce the number of adjacencies. This keeps the LSA database minimized and optimizes convergence times within each area.

Link-state protocols deploy a logical hierarchy in their design. This usually consists of a backbone level and another sublevel, including:

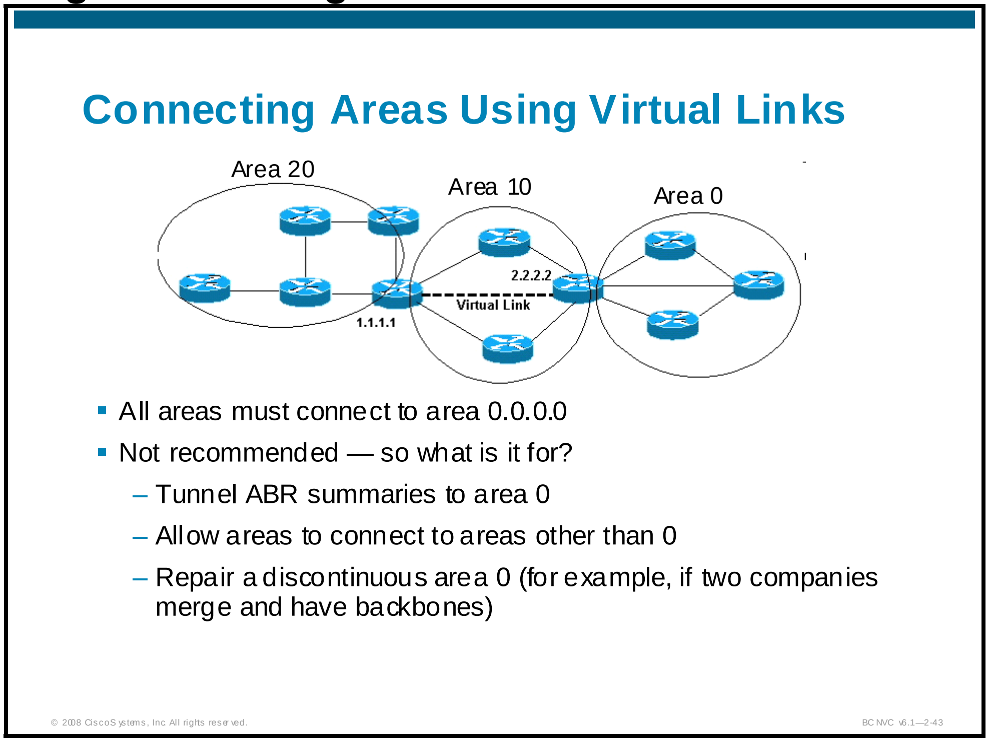
- OSPF: Backbone area (area 0), regular areas
- IS-IS: Layer 2 areas, Layer 1 areas

OSPF areas enable scalability by summarizing and abstracting, thereby reducing, information from lower level areas into the higher-level area.

OSPF uses a two-level hierarchical model:

- Areas defined with a 32-bit number (IP address format - Area 0.0.0.0).
- Areas can also be defined using a single decimal value - Area 0).
- 0.0.0.0 is reserved for the backbone area.
- All areas must connect to area 0.0.0.0.

Connecting Areas Using Virtual Links



Connecting Areas Using Virtual Links

- All areas must connect to area 0.0.0.0
- Not recommended — so what is it for?
  - Tunnel ABR summaries to area 0
  - Allow areas to connect to areas other than 0
  - Repair a discontinuous area 0 (for example, if two companies merge and have backbones)

BCNVC v6.1—2-43

All areas in an OSPF autonomous system must be physically connected to the backbone area (area 0). In some cases where this is not possible, you can use a virtual link to connect to the backbone through a non-backbone area. The area through which you configure the virtual link, known as a transit area, must have full routing information. The transit area cannot be a stub area.

The command to configure a virtual link is:

```
area <area-id> virtual-link <router-id>
```

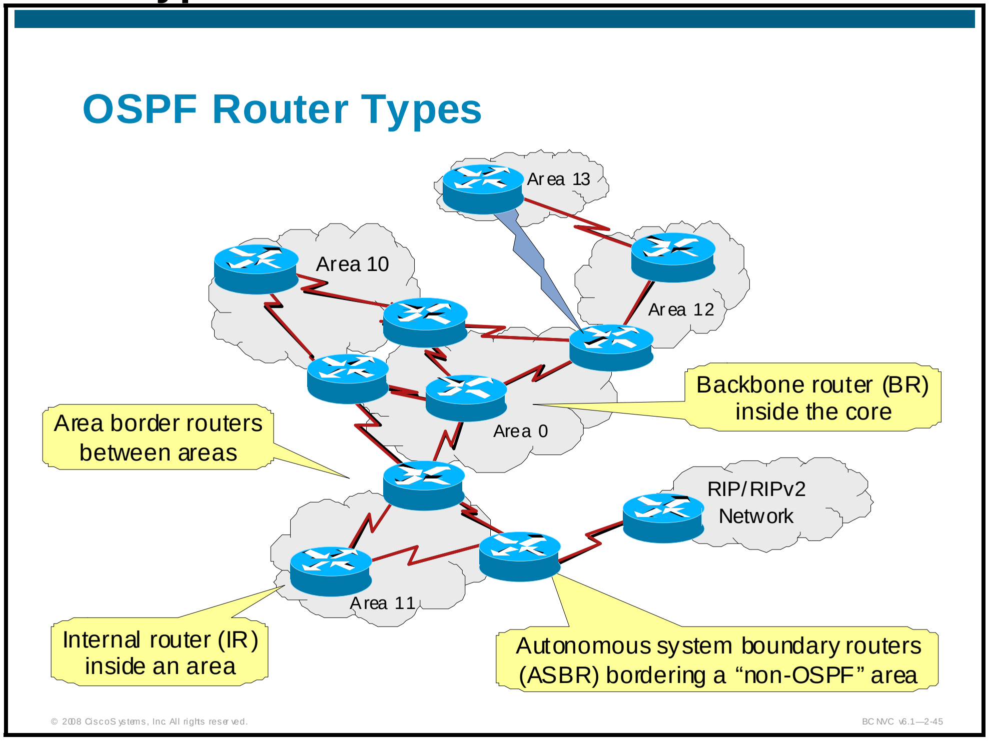
Enter the area ID assigned to the transit area (either a valid IP address or a decimal value) and the router ID associated with the virtual link neighbor. In the example topology, the virtual link connects area 20 to the backbone through area 10.

In this case, a virtual link is created between the routers with router ID 1.1.1.1 and router ID 2.2.2.2. To create the virtual link, configure the **area 10 virtual-link 2.2.2.2** subcommand on router 1.1.1.1 and the **area 10 virtual-link 1.1.1.1** subcommand on router 2.2.2.2.

A virtual link:

- May be required in backup scenarios
- Should be configured at each ABR
- Should use loopback interfaces

OSPF Router Types



OSPF Router Types

Area border routers between areas

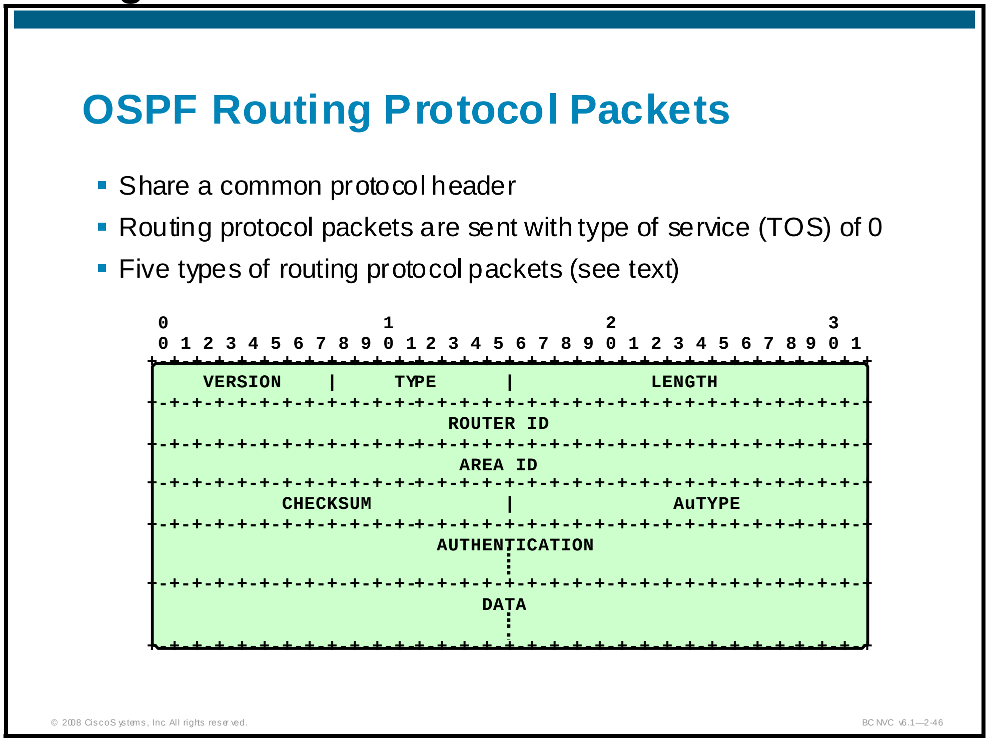
Backbone router (BR) inside the core

Internal router (IR) inside an area

Autonomous system boundary routers (ASBR) bordering a "non-OSPF" area

- Internal Router (IR) is inside an area.
- Backbone Router (BR) has at least one interface in backbone area 0.
- Area Border Router (ABR) is between two or more areas.
- Autonomous System Boundary Routers (ASBRs) connect to a different routing administration (routes are redistributed into OSPF).

OSPF Routing Protocol Packets



OSPF Routing Protocol Packets

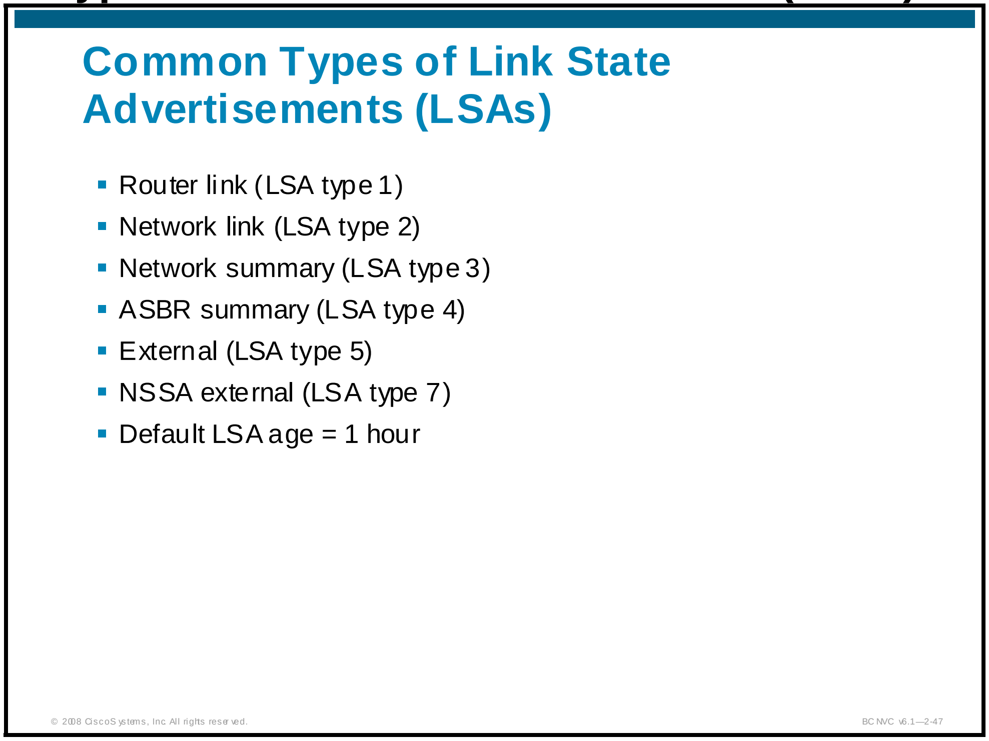
- Share a common protocol header
- Routing protocol packets are sent with type of service (TOS) of 0
- Five types of routing protocol packets (see text)

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   VERSION     |      TYPE     |             LENGTH            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          ROUTER ID                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           AREA ID                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           CHECKSUM            |             AuTYPE            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       AUTHENTICATION                          |
|                               :                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             DATA                              |
|                               :                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

BCNVC v6.1—2-46

Field	Description
Version.8 bits.	OSPF version number
Type. 8 bits.	OSPF request/reply code 1 = Hello. 2 = Database description. 3 = Link-state request. 4 = Link-state update. 5 = Link-state acknowledgment.
Length. 16 bits.	Size of the OSPF message including the OSPF header
Router ID. 32 bits.	The Router ID of the packet source
Area ID. 32 bits.	The area that this packet belongs to (OSPF packets are associated with a single area). Packets traveling over a virtual link are labeled with the backbone Area ID of 0.0.0.0
Checksum. 16 bits.	The standard IP checksum of the entire contents of the packet, starting with the OSPF packet header but excluding the 64 bit authentication field
AuType. 16 bits.	Identifies the authentication procedure to be used for the packet
Authentication. 64 bits.	Authentication 0 = None 1 = Simple password authentication 2 = Cryptographic authentication 3 - 65535 = Reserved

Common Types of Link State Advertisements (LSAs)



Common Types of Link State Advertisements (LSAs)

- Router link (LSA type 1)
- Network link (LSA type 2)
- Network summary (LSA type 3)
- ASBR summary (LSA type 4)
- External (LSA type 5)
- NSSA external (LSA type 7)
- Default LSA age = 1 hour

BCNVC v6.1—2-47

LSAs are packets that OSPF uses to advertise changes in the condition of a specific link to other OSPF routers. There are various types of link-state packets used by OSPF, each of which is generated for a different purpose and flooded in the network.

Following are the different types of LSA packets that can be generated by the source router and entered into the destination router LSA database.

Type 1 Router LSAs:

Router LSAs are generated by each router for each area it is in. These packets describe the state of the router links in an area to other OSPF devices in that area. They are only flooded within an area. The link-state ID is the originating router ID.

Type 2 Network LSAs:

Network LSAs are generated by Designated Routers (DR)s and describe the set of routers attached to a particular network. They are flooded in the area that contains the network. The link-state ID is the IP interface address of the DR.

Note Type of Service (ToS) has been removed from the OSPF specifications; however, most implementations in the field have yet to see this, so ToS fields remain for clarity.

Type 3 Summary LSAs for ABRs:

Summary LSAs are generated by Area Border Routers (ABRs) and describe inter-area routes to various networks. They can also be used for aggregating routes. The link-state ID is the destination network number.

Type 4 Summary LSAs for ASBRs:

Summary LSAs describe links to Autonomous System Border Routers (ASBRs) and are generated by Area Border Routers (ABRs). The link-state ID is the router ID of the described ASBR.

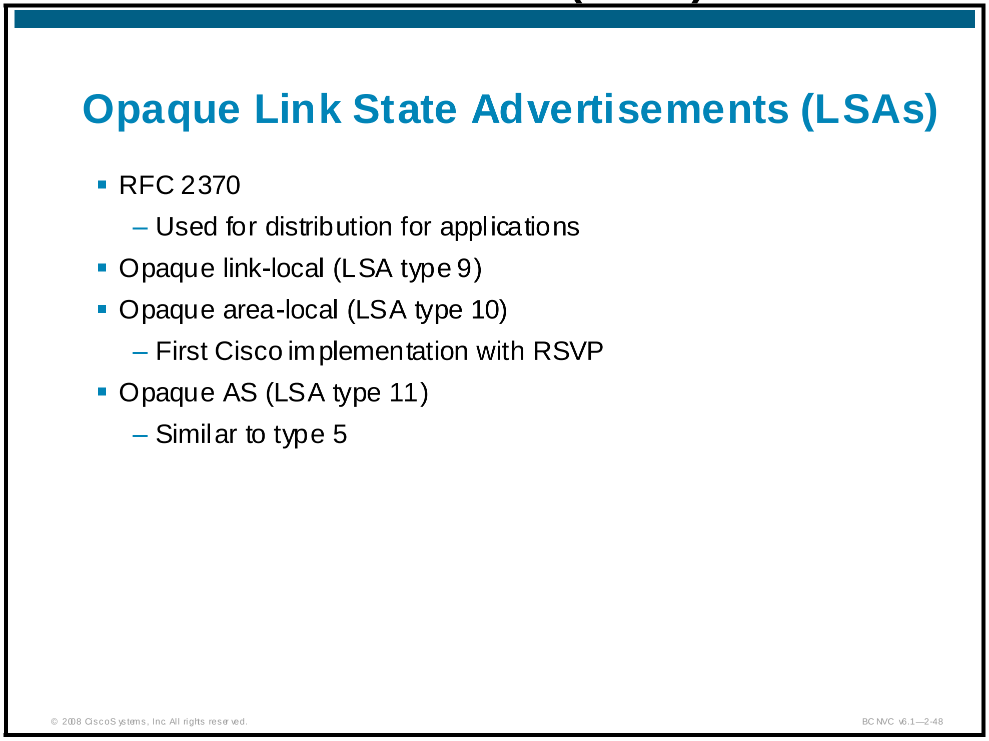
Type 5 Autonomous System External LSAs:

Type 5 LSAs are generated by Autonomous System Border Routers (ASBRs). They describe routes to destinations outside an autonomous system. They are flooded everywhere except stub areas. The link-state ID is the external network number.

Type 7 Not-So-Stubby Area (NSSA):

Type 7 LSAs are generated by ASBRs. They describe external routes connecting to an NSSA. Type 7 LSAs are converted into Type 5 LSAs by the ABR as the advertisement is propagated to the backbone. After they are converted to Type 5 LSAs, they are distributed to areas that can support Type 5 LSAs. Refer to RFC 1587 for further details on how this conversion occurs.

Opaque Link State Advertisements (LSAs)



Opaque Link State Advertisements (LSAs)

- RFC 2370
  - Used for distribution for applications
- Opaque link-local (LSA type 9)
- Opaque area-local (LSA type 10)
  - First Cisco implementation with RSVP
- Opaque AS (LSA type 11)
  - Similar to type 5

BCNVC v6.1—2-48

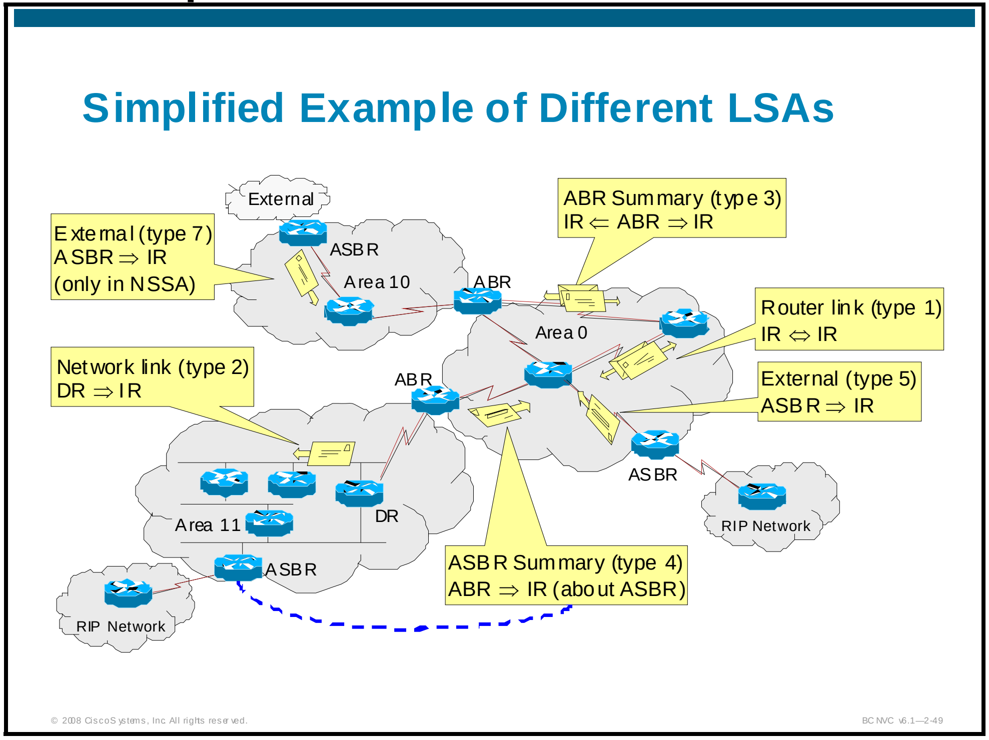
Type 9, 10, and 11 Opaque LSAs: Opaque LSAs may be used for distributing application-specific information through an OSPF domain. Type 9 LSAs are not flooded beyond the local (sub)network. Type 10 LSAs are not flooded beyond the borders of their associated area. Type 11 LSAs are flooded throughout an AS. The flooding scope of Type 11 LSAs is equivalent to that of AS-external (Type-5) LSAs. Multiprotocol Label Switching Traffic Engineering (MPLS-TE) functionality is implemented with Type 10 Opaque LSAs. For more information on opaque LSAs, please see RFC2370.

Note LSA Type 6 (MOSPF) is not supported on Cisco routers. A syslog message is generated whenever Cisco routers receive a Type 6 LSA. The following router configuration command can be configured under the router ospf process to ignore these syslog messages:

```
ospf ignore lsa mospf
```

Applicable Opaque LSAs are discussed in the MPLS section.

Simplified Example of Different LSAs



Simplified Example of Different LSAs

External (type 7)
ASBR ⇒ IR
(only in NSSA)

ABR Summary (type 3)
IR ⇐ ABR ⇒ IR

Router link (type 1)
IR ⇔ IR

Network link (type 2)
DR ⇒ IR

External (type 5)
ASBR ⇒ IR

ASBR Summary (type 4)
ABR ⇒ IR (about ASBR)

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—2-49

Router LSA (type 1)

- Describes the state and cost of the router links to the area
- All of the router links in an area must be described in a single LSA
- Flooded throughout the particular area and no more
- Router indicates whether it is an ASBR, ABR, or virtual link endpoint

Network LSA (type 2)

- Generated for every broadcast network
- Describes all routers attached to a network
- Only the designated router originates this LSA
- Flooded throughout the area and no more

Summary LSA (type 3 and type 4)

- Describes the destination outside the area but still in the AS
- Flooded throughout a single area
- Originated by an ABR
- Only intra-area routes are advertised to the backbone
- Type 4 is the information about the ASBR

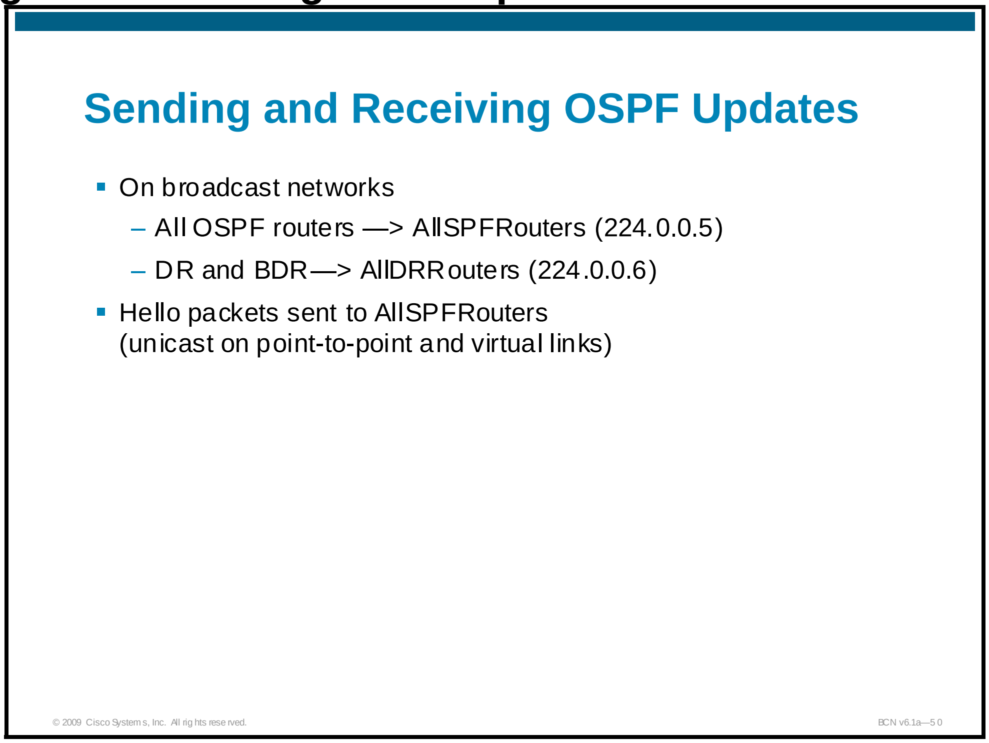
External LSA (type 5)

- Defines routes to a destination external to the AS
- Default route is also sent as an external route
- Two types of external LSA:
  - E1: Consider the total cost up to the external destination
  - E2: Considers only the cost of the outgoing interface to the external destination (Cisco default)

External LSA (type 7)

- Defines routes to destination external to the AS
- Flooded throughout NSSA areas ONLY
- Originated by an ASBR
- Converted to External (type 5) by the ABR before being sent to area 0

Sending and Receiving OSPF Updates

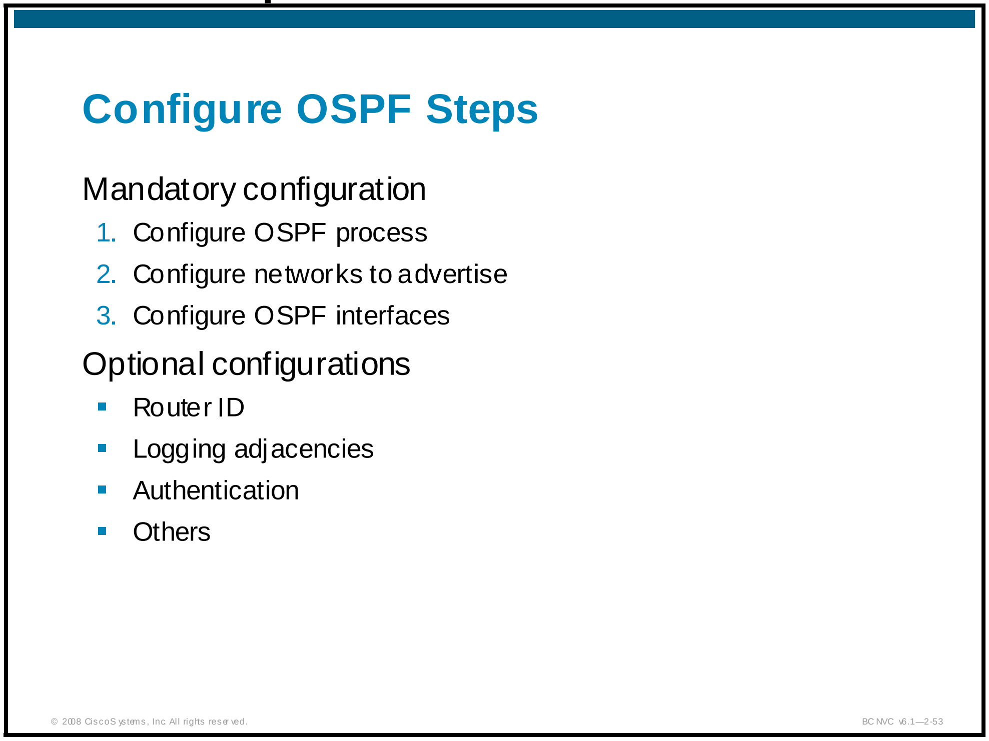


Sending and Receiving OSPF Updates

- On broadcast networks
  - All OSPF routers —> AllSPFRouters (224.0.0.5)
  - DR and BDR—> AllDRRouters (224.0.0.6)
- Hello packets sent to AllSPFRouters
  (unicast on point-to-point and virtual links)

Configure and OSPF Steps

Configure OSPF Steps



Configure OSPF Steps

Mandatory configuration

1. Configure OSPF process
2. Configure networks to advertise
3. Configure OSPF interfaces

Optional configurations

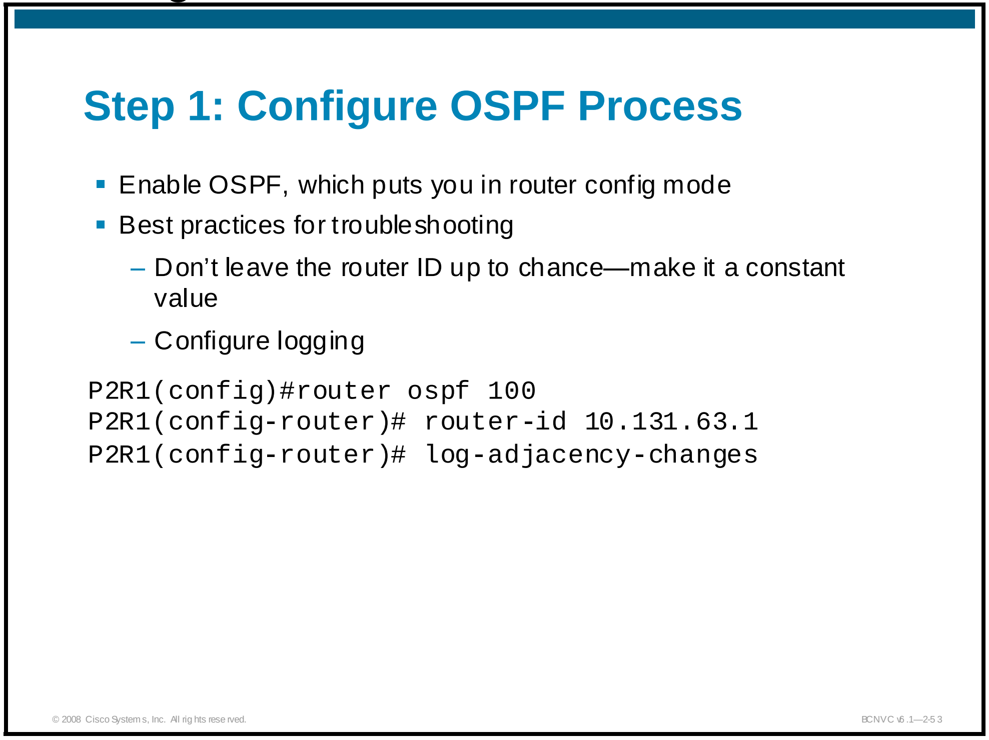
- Router ID
- Logging adjacencies
- Authentication
- Others

BCNVC v6.1—2-53

As with other routing protocols, enabling OSPF requires that you create an OSPF routing process, associate a range of IP addresses with the routing process, and assign area IDs to associate with that range of IP addresses.

Routing interface parameters are configurable parameters that include interface output cost, retransmission interval, interface transmit delay, router priority, router dead and hello intervals, and authentication key.

Step 1: Configure OSPF Process



Step 1: Configure OSPF Process

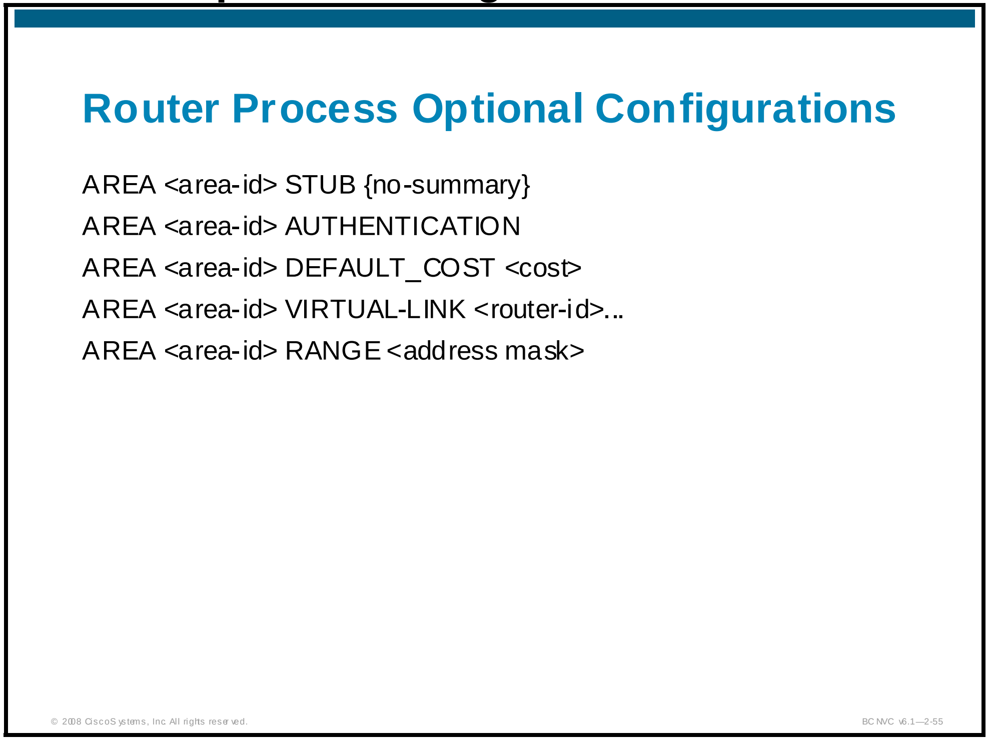
- Enable OSPF, which puts you in router config mode
- Best practices for troubleshooting
  - Don’t leave the router ID up to chance—make it a constant value
  - Configure logging

```
P2R1(config)#router ospf 100
P2R1(config-router)# router-id 10.131.63.1
P2R1(config-router)# log-adjacency-changes
```

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—2-53

Router Process Optional Configurations



Router Process Optional Configurations

AREA <area-id> STUB {no-summary}

AREA <area-id> AUTHENTICATION

AREA <area-id> DEFAULT_COST <cost>

AREA <area-id> VIRTUAL-LINK <router-id>...

AREA <area-id> RANGE <address mask>

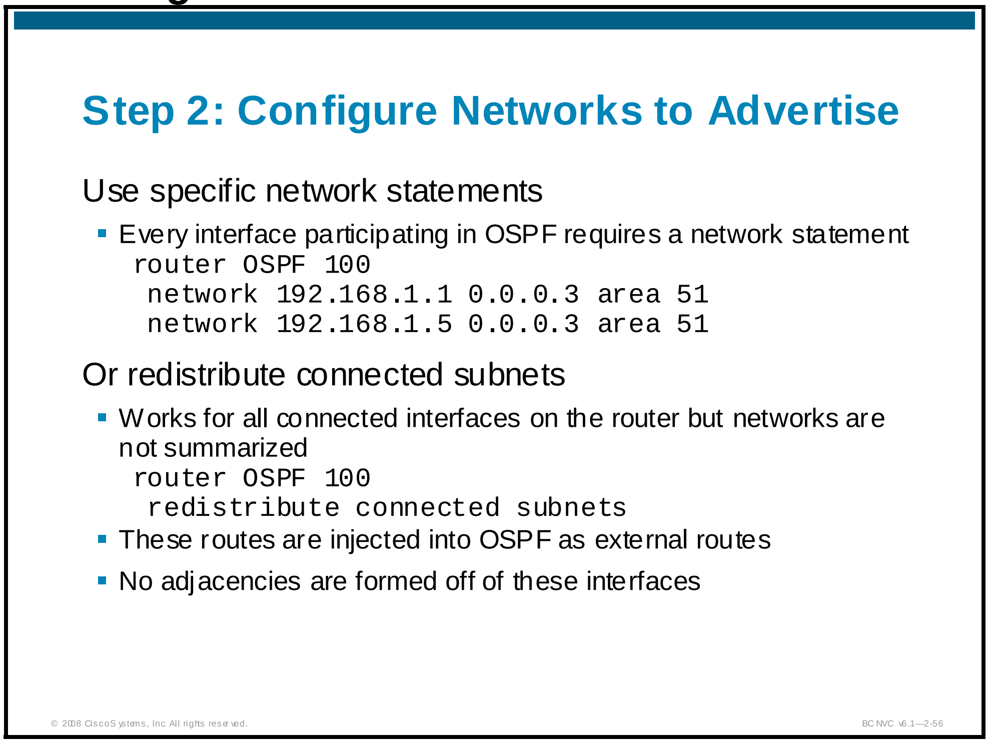
© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—2-55

`AREA <area-id> DEFAULT_COST`

To specify a cost for the default summary route sent into a stub or not so stubby area (NSSA), use the **area default-cost** command in router configuration mode.

Step 2: Configure Networks to Advertise



Step 2: Configure Networks to Advertise

Use specific network statements

- Every interface participating in OSPF requires a network statement

```
router OSPF 100
 network 192.168.1.1 0.0.0.3 area 51
 network 192.168.1.5 0.0.0.3 area 51
```

Or redistribute connected subnets

- Works for all connected interfaces on the router but networks are not summarized

```
router OSPF 100
 redistribute connected subnets
```

- These routes are injected into OSPF as external routes
- No adjacencies are formed off of these interfaces

BCNVC v6.1—2-56

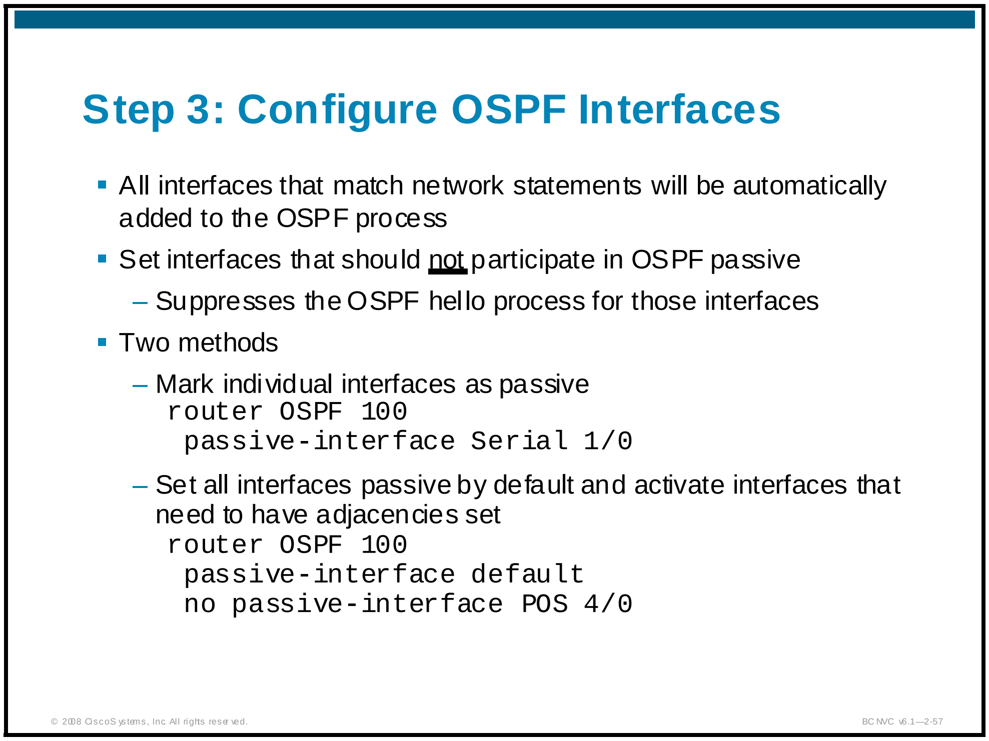
There are two methods of advertising a network:

- Network statement
- Redistribution

Best practices are to add an individual OSPF network statement for each infrastructure link.

Step 3: Configure OSPF Interfaces

Defining Non-OSPF Interfaces



Step 3: Configure OSPF Interfaces

- All interfaces that match network statements will be automatically added to the OSPF process
- Set interfaces that should not participate in OSPF passive
  - Suppresses the OSPF hello process for those interfaces
- Two methods
  - Mark individual interfaces as passive

```
router OSPF 100
 passive-interface Serial 1/0
```

  - Set all interfaces passive by default and activate interfaces that need to have adjacencies set

```
router OSPF 100
 passive-interface default
 no passive-interface POS 4/0
```

© 2008 Cisco Systems, Inc. All rights reserved.   BCMVC v6.1—2-57

To prevent other routers in a local network from learning about routes dynamically, you can keep routing update messages from being sent through a router interface. Keeping routing update messages from being sent through a router interface prevents other systems on the interface from learning about routes dynamically. This feature applies to all IP-based routing protocols except BGP.

OSPF and IS-IS behave somewhat differently. In OSPF, the interface address you specify as passive appears as a stub network in the OSPF domain. OSPF routing information is neither sent nor received through the specified router interface. In IS-IS, the specified IP addresses are advertised without actually running IS-IS on those interfaces.

In large networks, many of the distribution routers have more than 200 interfaces. Before the Default Passive Interface feature, there were two possibilities for obtaining routing information from these interfaces:

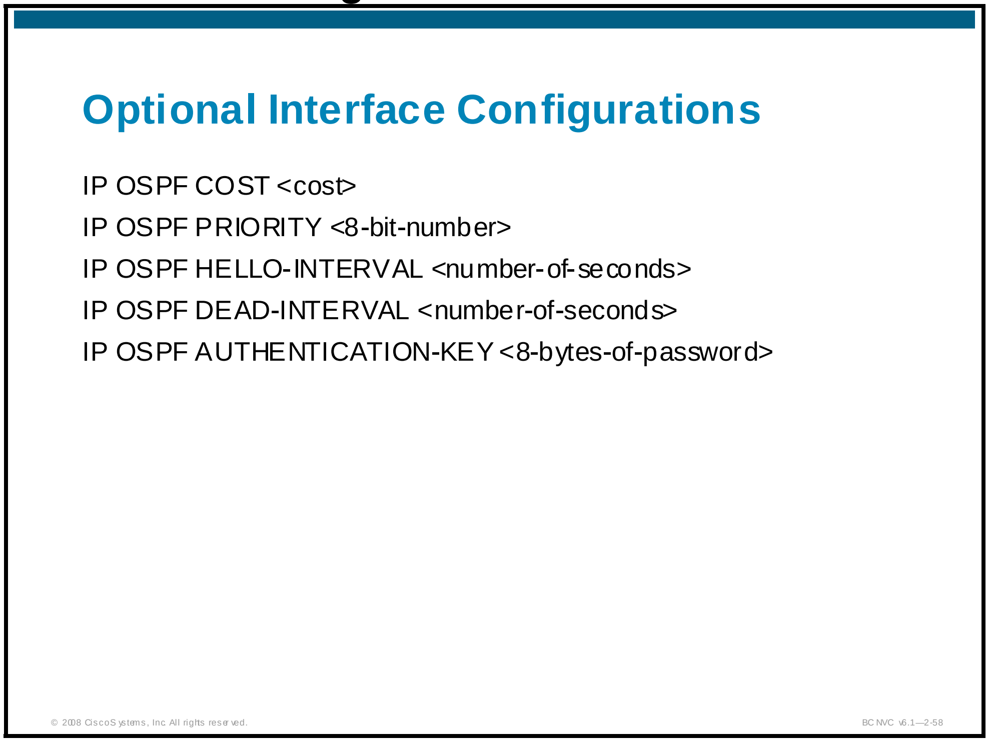
- Configure a routing protocol such as OSPF on backbone interfaces and redistribute connected interfaces.
- Configure the routing protocol on all interfaces and manually set most of them as passive.

Network managers may not always be able to summarize type-5 LSAs at the router level where redistribution occurs, as in the first possibility. Thus, a large number of type-5 LSAs can be flooded over a domain.

In the second possibility, large type-1 link-state LSAs might be flooded into an area. An area border router (ABR) creates type-3 LSAs, one for each type-1 LSA, and floods them to the backbone. It is possible, however, to have unique summarization at the ABR level, which injects just one summary route into the backbone, thereby reducing processing overhead.

The solution to this problem has been to configure the routing protocol on all interfaces and manually set the **passive-interface** command on the interfaces where adjacency was not desired. In some networks, this means coding 200 or more passive interface statements. With the Default Passive Interface feature, this problem is solved by allowing all interfaces to be set as passive by default using a single **passive-interface default** command, then configuring individual interfaces where adjacencies are desired using the **no passive-interface** command.

Optional Interface Configurations



Optional Interface Configurations

IP OSPF COST <cost>

IP OSPF PRIORITY <8-bit-number>

IP OSPF HELLO-INTERVAL <number-of-seconds>

IP OSPF DEAD-INTERVAL <number-of-seconds>

IP OSPF AUTHENTICATION-KEY <8-bytes-of-password>

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—2-58

DR and BDR selection

`ip ospf priority 100` (default 1)

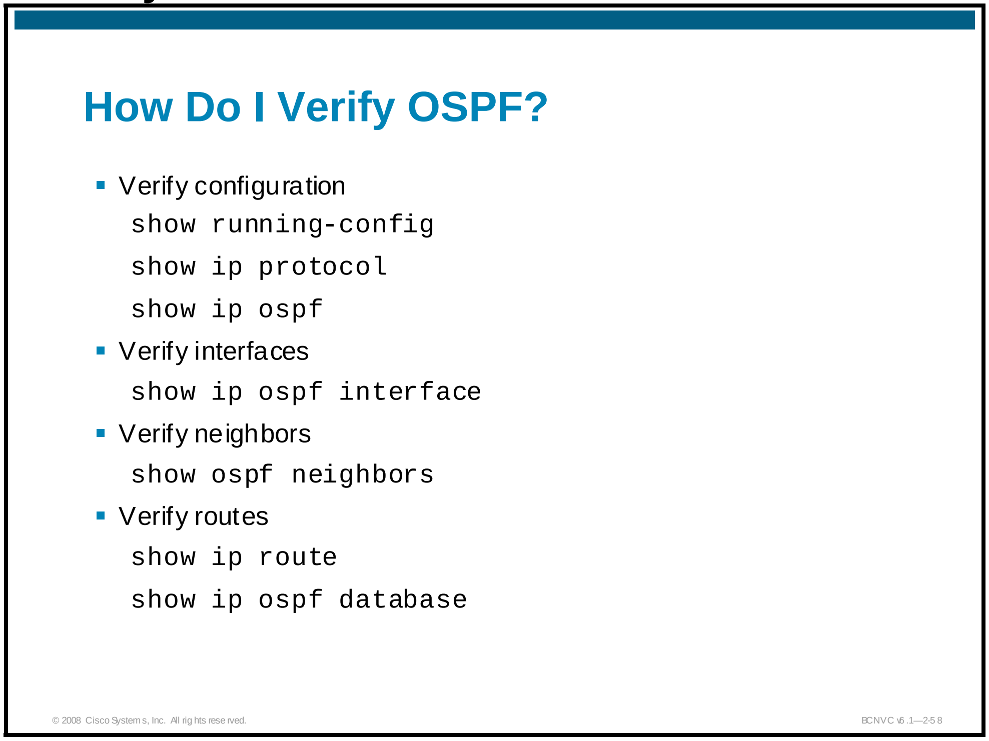
This feature should be in used in your OSPF network. Forcibly set your DR and BDR per segment so that they are known. Choose your most powerful, or most idle routers. Try to keep the DR or BDR limited to one segment each. Priority 0 means it never becomes elected.

Hello and dead timers

`ip ospf hello-interval 3` (the default is 10 seconds for a broadcast network, default is 30 seconds for non-broadcast networks)
`ip ospf dead-interval 15` (the default is four times the hello interval)

This allows faster network awareness of a failure, and can result in faster reconvergence, but requires more router central processing unit CPU use and generates more overhead.

How Do I Verify OSPF?



How Do I Verify OSPF?

- Verify configuration

```
show running-config
show ip protocol
show ip ospf
```

- Verify interfaces

```
show ip ospf interface
```

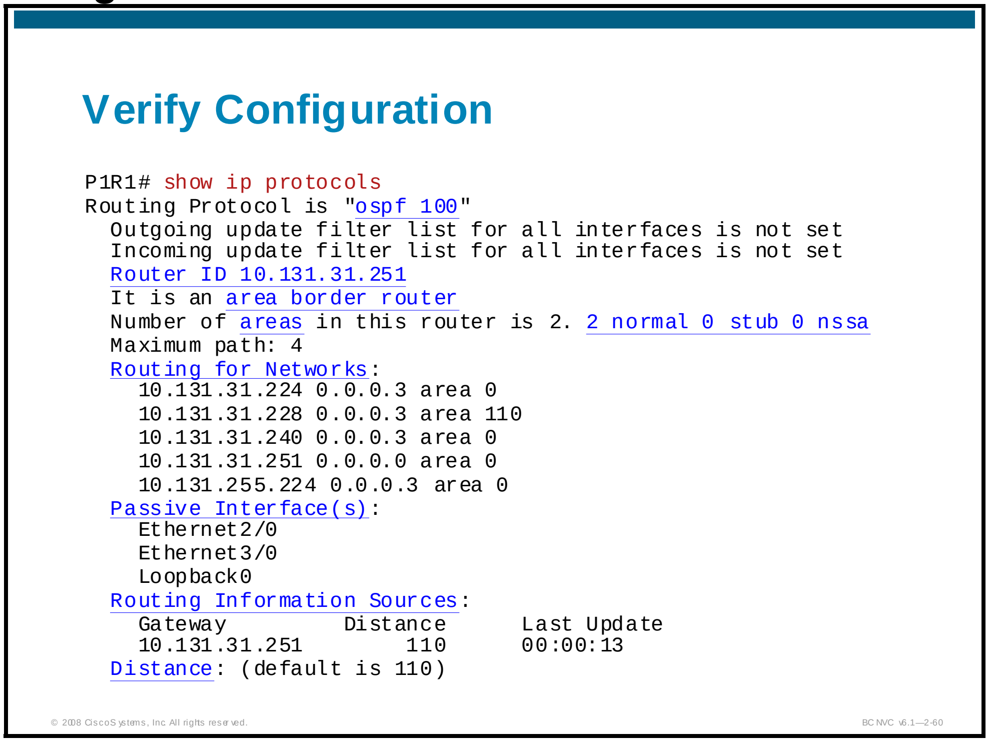
- Verify neighbors

```
show ospf neighbors
```

- Verify routes

```
show ip route
show ip ospf database
```

Verify Configuration



Verify Configuration

```
P1R1# show ip protocols
Routing Protocol is "ospf 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 10.131.31.251
  It is an area border router
  Number of areas in this router is 2. 2 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    10.131.31.224 0.0.0.3 area 0
    10.131.31.228 0.0.0.3 area 110
    10.131.31.240 0.0.0.3 area 0
    10.131.31.251 0.0.0.0 area 0
    10.131.255.224 0.0.0.3 area 0
  Passive Interface(s):
    Ethernet2/0
    Ethernet3/0
    Loopback0
  Routing Information Sources:
    Gateway         Distance      Last Update
    10.131.31.251        110      00:00:13
  Distance: (default is 110)
```

BCNVC v6.1—2-60

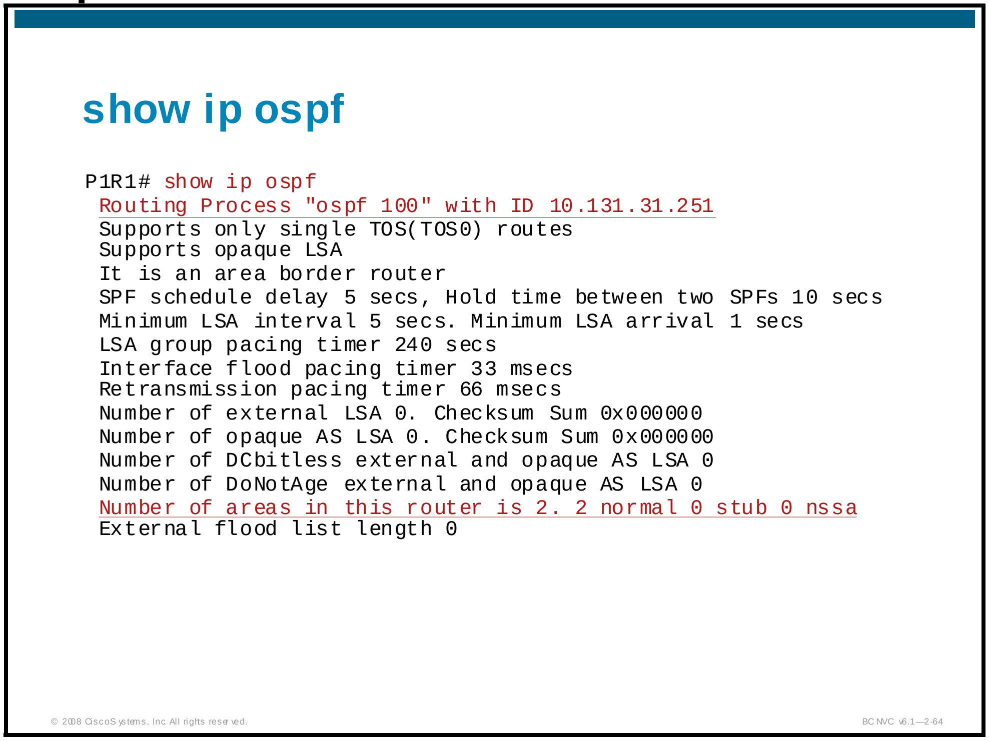
Show running-configuration is always a good way check a configuration. However, sometimes it is difficult to see your own mistakes. The **show ip protocols** command can give you a different view of your configuration.

It is easy to verify that you properly configured:

- OSPF process ID
- Router ID
- Network statements
- Passive interfaces
- Distance

Also, under the heading Routing Information Sources you can find the neighbors that are providing updates. Is this all you should have? The **show IP OSPF neighbors** command gives you additional neighbor information.

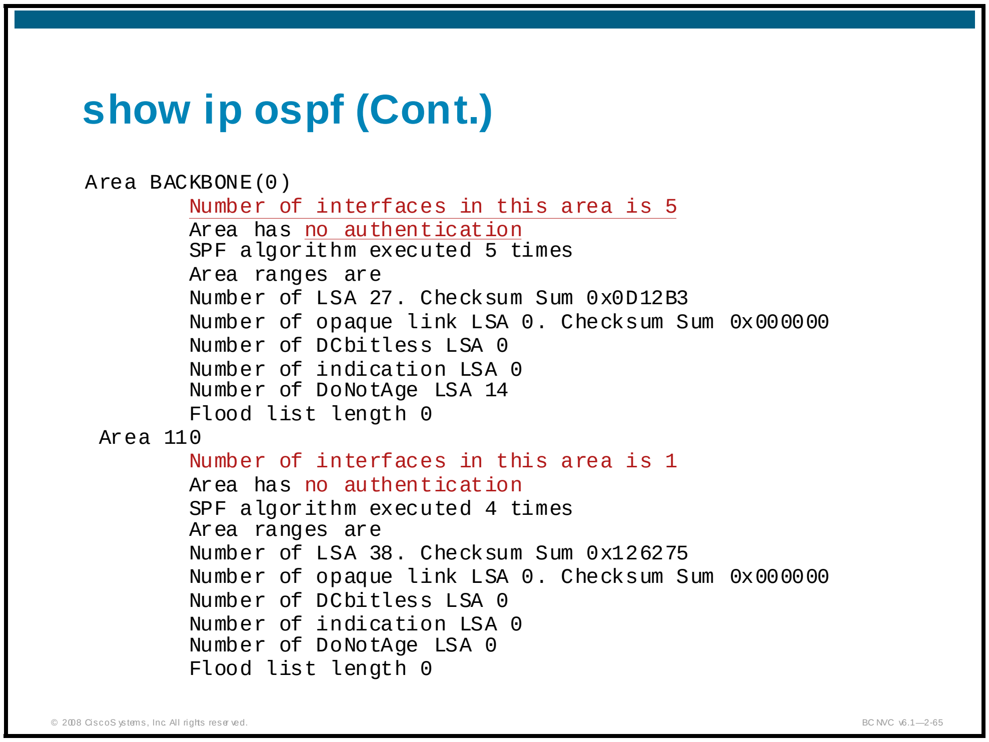
show ip ospf



show ip ospf

```
P1R1# show ip ospf
 Routing Process "ospf 100" with ID 10.131.31.251
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 It is an area border router
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 0. Checksum Sum 0x000000
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 2. 2 normal 0 stub 0 nssa
 External flood list length 0
```

BC NVC v6.1—2-64

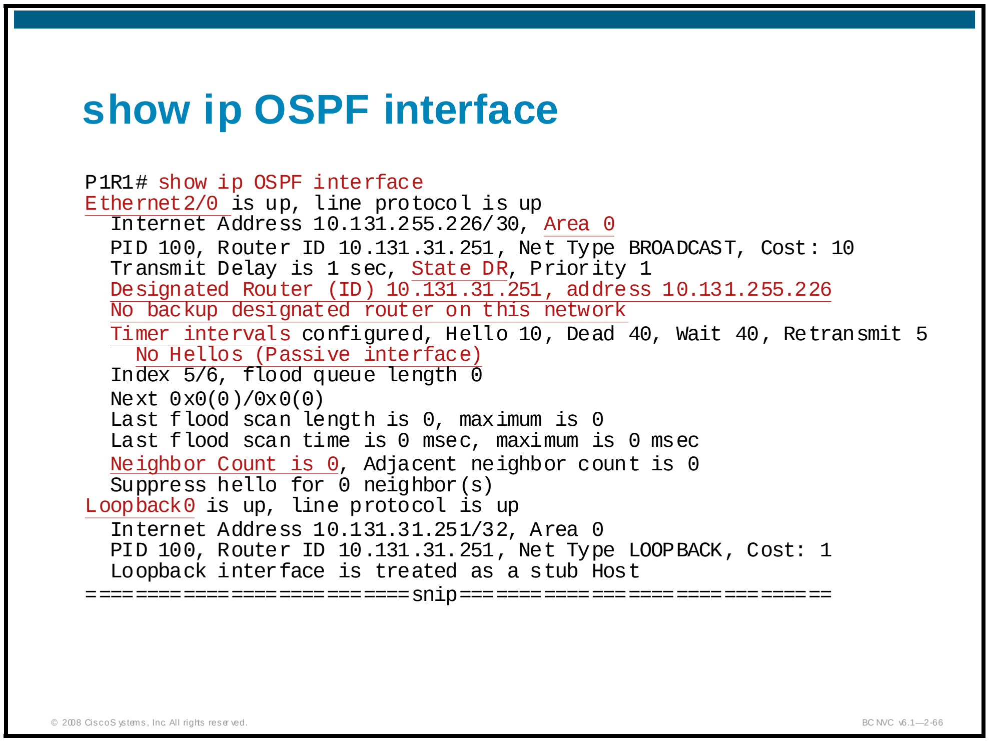


show ip ospf (Cont.)

```
Area BACKBONE(0)
        Number of interfaces in this area is 5
        Area has no authentication
        SPF algorithm executed 5 times
        Area ranges are
        Number of LSA 27. Checksum Sum 0x0D12B3
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 14
        Flood list length 0
 Area 110
        Number of interfaces in this area is 1
        Area has no authentication
        SPF algorithm executed 4 times
        Area ranges are
        Number of LSA 38. Checksum Sum 0x126275
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

BC NVC v6.1—2-65

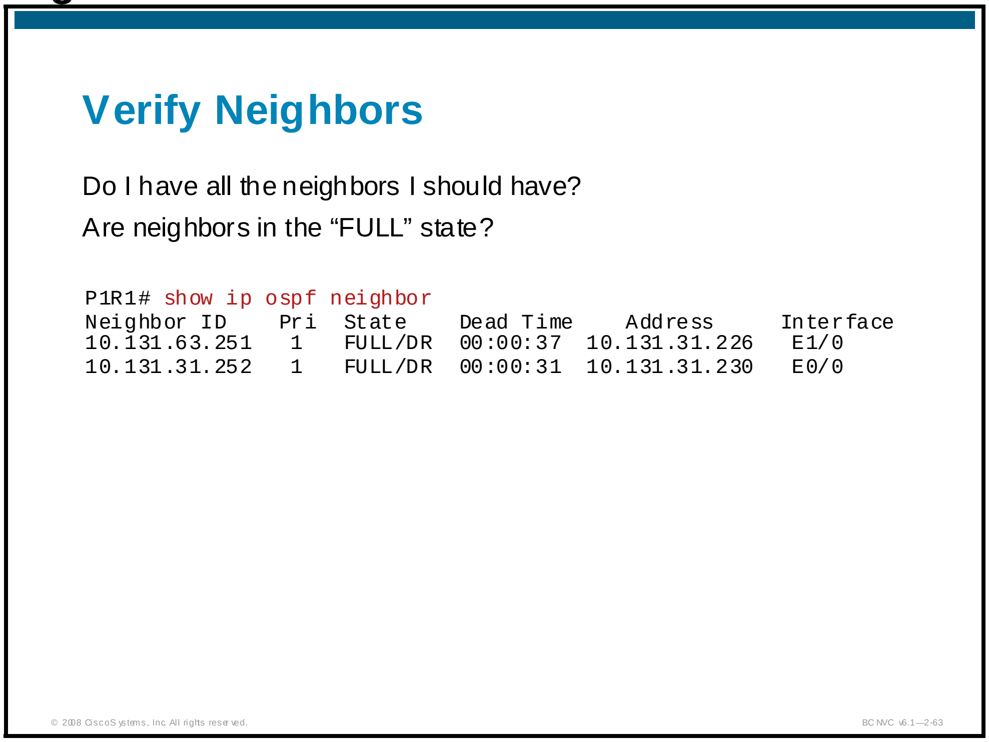
show ip OSPF interface



show ip OSPF interface

```
P1R1# show ip OSPF interface
Ethernet2/0 is up, line protocol is up
  Internet Address 10.131.255.226/30, Area 0
  PID 100, Router ID 10.131.31.251, Net Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 10.131.31.251, address 10.131.255.226
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    No Hellos (Passive interface)
  Index 5/6, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
Loopback0 is up, line protocol is up
  Internet Address 10.131.31.251/32, Area 0
  PID 100, Router ID 10.131.31.251, Net Type LOOPBACK, Cost: 1
  Loopback interface is treated as a stub Host
==========================snip============================
```

Verify Neighbors



Verify Neighbors

Do I have all the neighbors I should have?

Are neighbors in the “FULL” state?

```
P1R1# show ip ospf neighbor
Neighbor ID     Pri   State     Dead Time    Address       Interface
10.131.63.251    1    FULL/DR   00:00:37  10.131.31.226    E1/0
10.131.31.252    1    FULL/DR   00:00:31  10.131.31.230    E0/0
```

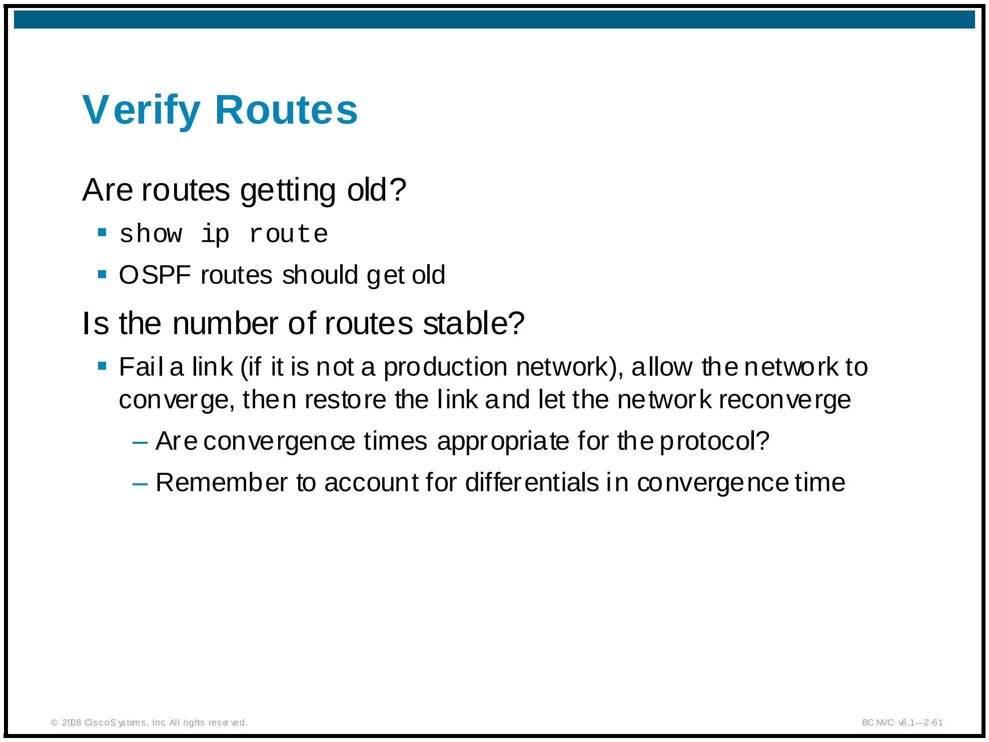
If you don’t have the appropriate neighbors, double-check your network statements to ensure the correct interfaces ended up in the correct areas.

One of the most common issues in OSPF is the failure of two neighboring routers to become adjacent. There can be many causes. The following items can be verified when troubleshooting adjacency issues:

- Make sure the network type is the same on all routers attached to a media.
- If the hello timers have been changed, make sure all routers have the same value for hello intervals and dead intervals on a given media.
- Make sure you have the same maximum transmission unit (MTU) on all routers attached to a media. If the routers change to the Exchange state and go no further, there may be an MTU mismatch.
- Make sure the authentication is set properly on both ends of a link.

Run **debug ip ospf adjacency** to find out more about the cause of a problem. This command should be executed on both sides of a link. Also, this debug command can be safely executed on a router without producing any problems.

Verify Routes



Verify Routes

Are routes getting old?

- `show ip route`
- OSPF routes should get old

Is the number of routes stable?

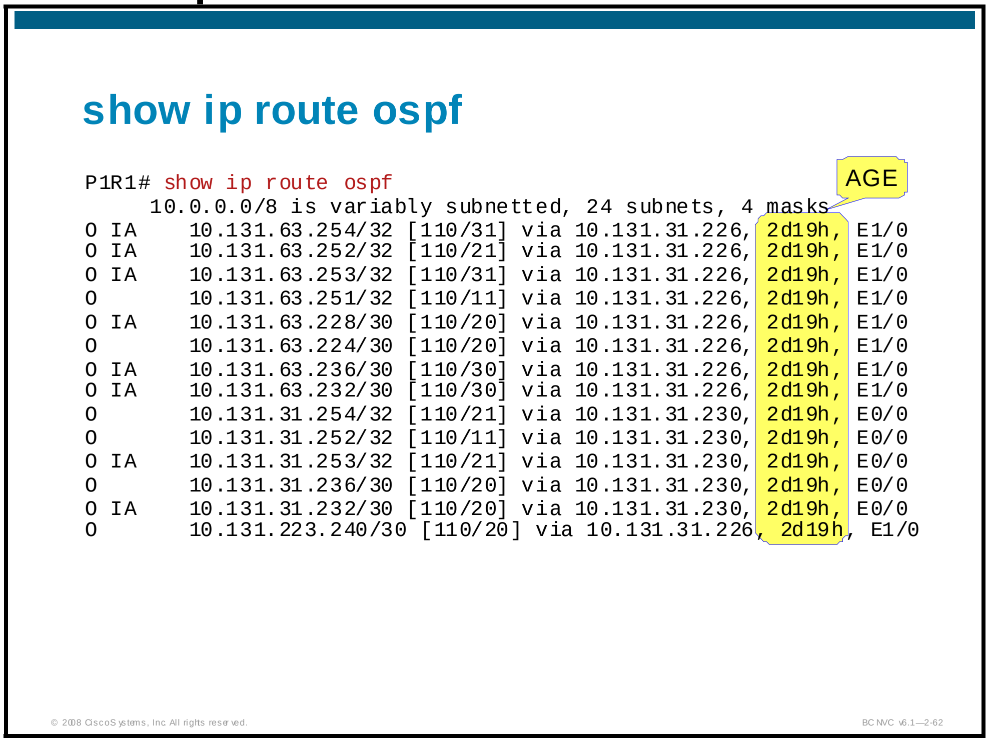
- Fail a link (if it is not a production network), allow the network to converge, then restore the link and let the network reconverge
  - Are convergence times appropriate for the protocol?
  - Remember to account for differentials in convergence time

OSPF routes should be as old as router uptime after a reboot. If not and you can’t find a failed link, there is a configuration error.

If the routes' ages are not appropriate, the change is a failure and should be reverted. The network is unstable and while you might have raw connectivity, performance will probably suffer and the network may fail altogether under load.

When no links change state, the number of routes should be stable. If you can’t account for each convergence event by finding a failed or flapping link, then you’ve got a route loop situation and the change should be reverted.

show ip route ospf



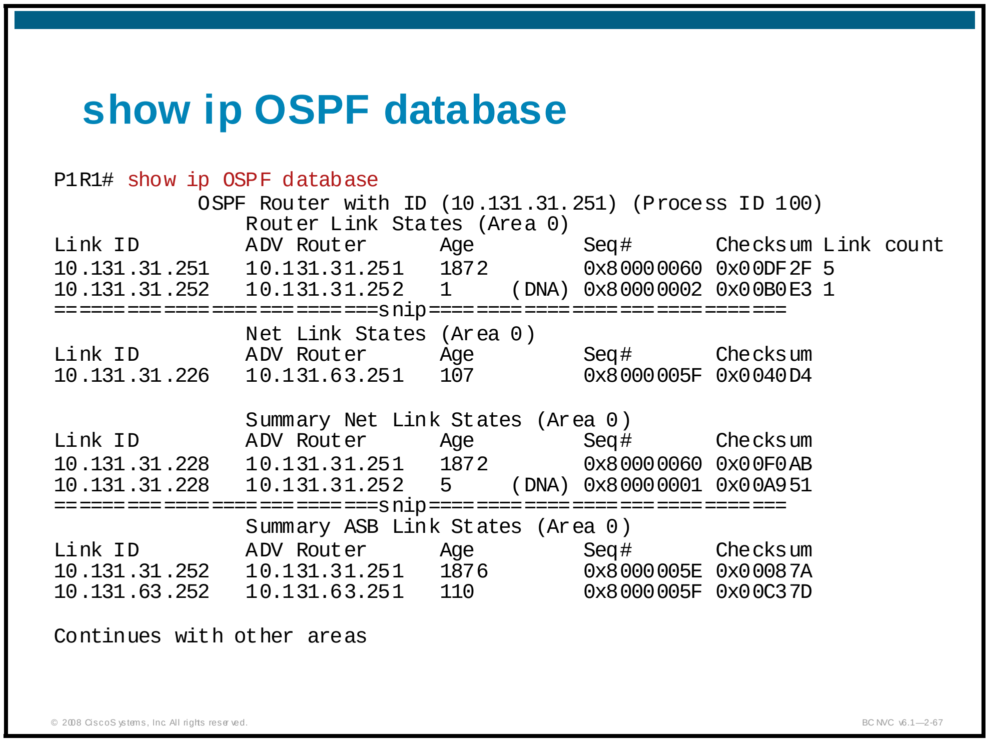
show ip route ospf

AGE

```
P1R1# show ip route ospf
     10.0.0.0/8 is variably subnetted, 24 subnets, 4 masks
O IA    10.131.63.254/32 [110/31] via 10.131.31.226, 2d19h, E1/0
O IA    10.131.63.252/32 [110/21] via 10.131.31.226, 2d19h, E1/0
O IA    10.131.63.253/32 [110/31] via 10.131.31.226, 2d19h, E1/0
O       10.131.63.251/32 [110/11] via 10.131.31.226, 2d19h, E1/0
O IA    10.131.63.228/30 [110/20] via 10.131.31.226, 2d19h, E1/0
O       10.131.63.224/30 [110/20] via 10.131.31.226, 2d19h, E1/0
O IA    10.131.63.236/30 [110/30] via 10.131.31.226, 2d19h, E1/0
O IA    10.131.63.232/30 [110/30] via 10.131.31.226, 2d19h, E1/0
O       10.131.31.254/32 [110/21] via 10.131.31.230, 2d19h, E0/0
O       10.131.31.252/32 [110/11] via 10.131.31.230, 2d19h, E0/0
O IA    10.131.31.253/32 [110/21] via 10.131.31.230, 2d19h, E0/0
O       10.131.31.236/30 [110/20] via 10.131.31.230, 2d19h, E0/0
O IA    10.131.31.232/30 [110/20] via 10.131.31.230, 2d19h, E0/0
O       10.131.223.240/30 [110/20] via 10.131.31.226, 2d19h, E1/0
```

BCNVC v6.1—2-62

show ip OSPF database



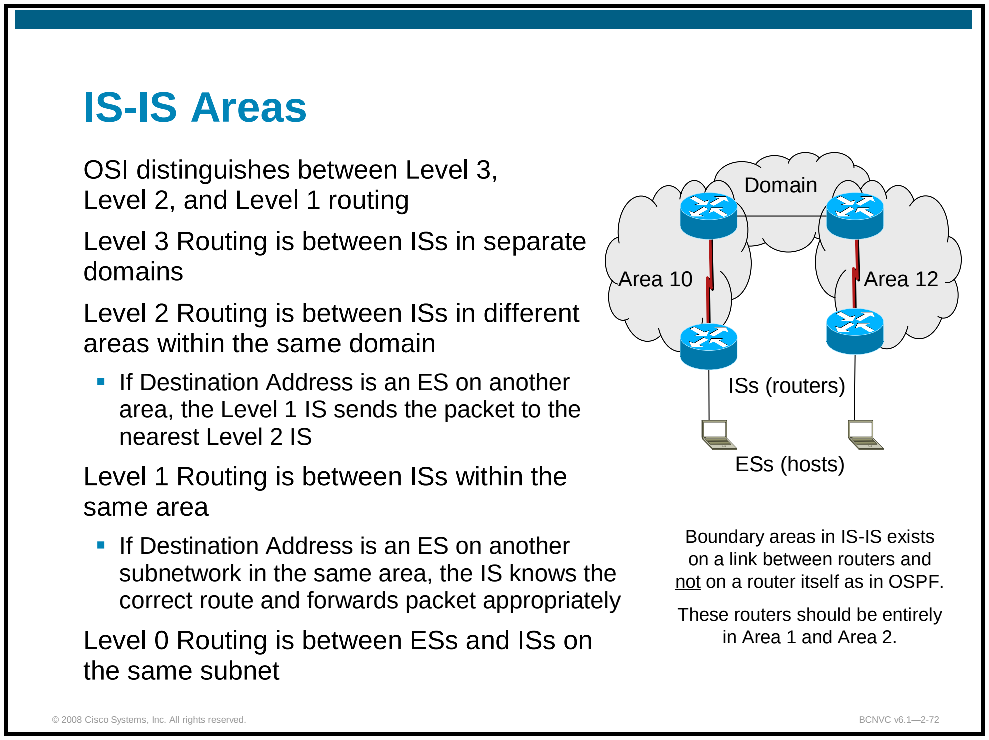
show ip OSPF database

```
P1R1# show ip OSPF database
            OSPF Router with ID (10.131.31.251) (Process ID 100)
                Router Link States (Area 0)
Link ID         ADV Router      Age         Seq#       Checksum Link count
10.131.31.251   10.131.31.251   1872        0x80000060 0x00DF2F 5
10.131.31.252   10.131.31.252   1     (DNA) 0x80000002 0x00B0E3 1
===========================snip=============================
                Net Link States (Area 0)
Link ID         ADV Router      Age         Seq#       Checksum
10.131.31.226   10.131.63.251   107         0x8000005F 0x0040D4

                Summary Net Link States (Area 0)
Link ID         ADV Router      Age         Seq#       Checksum
10.131.31.228   10.131.31.251   1872        0x80000060 0x00F0AB
10.131.31.228   10.131.31.252   5     (DNA) 0x80000001 0x00A951
===========================snip=============================
                Summary ASB Link States (Area 0)
Link ID         ADV Router      Age         Seq#       Checksum
10.131.31.252   10.131.31.251   1876        0x8000005E 0x00087A
10.131.63.252   10.131.63.251   110         0x8000005F 0x00C37D

Continues with other areas
```

BCNVC v6.1—2-67

Configure and Verify IS-IS

IS-IS Areas



IS-IS Areas

OSI distinguishes between Level 3, Level 2, and Level 1 routing

Level 3 Routing is between ISs in separate domains

Level 2 Routing is between ISs in different areas within the same domain

- If Destination Address is an ES on another area, the Level 1 IS sends the packet to the nearest Level 2 IS

Level 1 Routing is between ISs within the same area

- If Destination Address is an ES on another subnetwork in the same area, the IS knows the correct route and forwards packet appropriately

Level 0 Routing is between ESs and ISs on the same subnet

Boundary areas in IS-IS exists on a link between routers and not on a router itself as in OSPF.

These routers should be entirely in Area 1 and Area 2.

BCNVC v6.1—2-72

IS-IS is one of two popular Interior Gateway Protocols (IGP) used on the large service-provider networks that are interconnected to form the global Internet. The other popular IGP is the Open Shortest Path First (OSPF) protocol. The Border Gateway Protocol (BGP) is used for inter-domain router between network domains (or autonomous systems). While the protocol specifies 4 levels, IGP deployments are only concerned with a two-level routing hierarchy:

- Routing within areas (Level 1)
- Routing between areas (Level 2)

IS-IS inherits the following ISO classification and definition of the two basic types of net-work nodes:

- End systems
- Intermediate systems

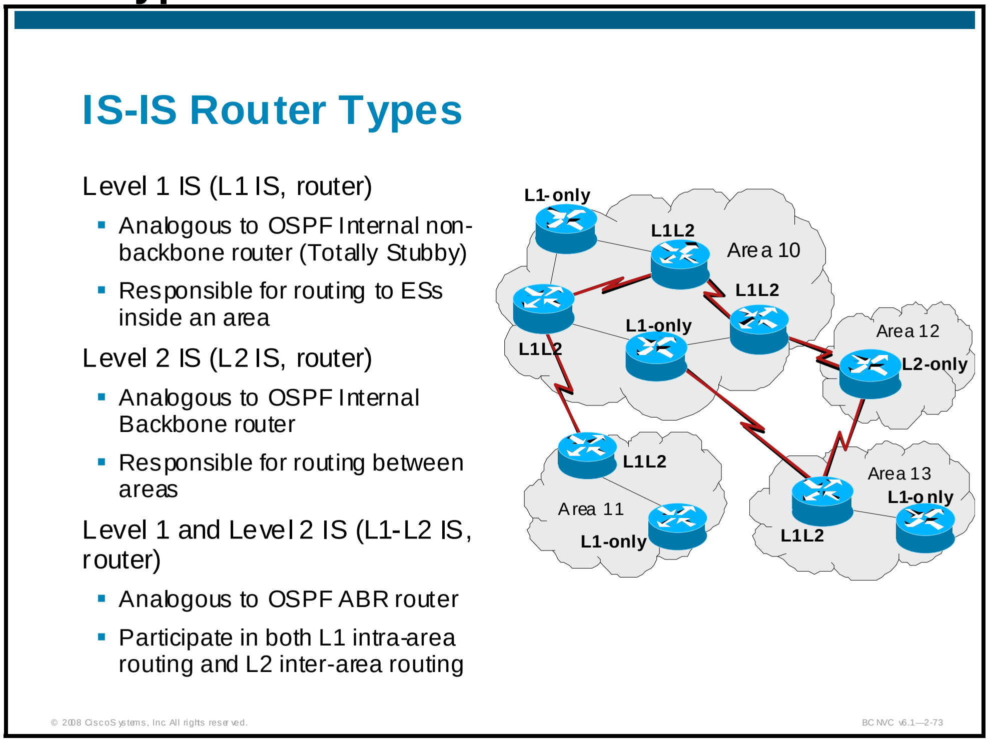
End systems are hosts in a network that typically do not have extensive routing capabilities. Intermediate systems refer to routers whose primary function is to route packets.

Network nodes are interconnected by links. Again, in IS-IS, only two basic links types are of practical relevance:

- Point-to-point links
- Broadcast links

Point-to-point links interconnect pairs of nodes, while broadcast type links are multipoint and can interconnect more than two nodes at the same time.

IS-IS Router Types



IS-IS Router Types

Level 1 IS (L1 IS, router)

- Analogous to OSPF Internal non-backbone router (Totally Stubby)
- Responsible for routing to ESs inside an area

Level 2 IS (L2 IS, router)

- Analogous to OSPF Internal Backbone router
- Responsible for routing between areas

Level 1 and Level 2 IS (L1-L2 IS, router)

- Analogous to OSPF ABR router
- Participate in both L1 intra-area routing and L2 inter-area routing

BCNVC v6.1—2-73

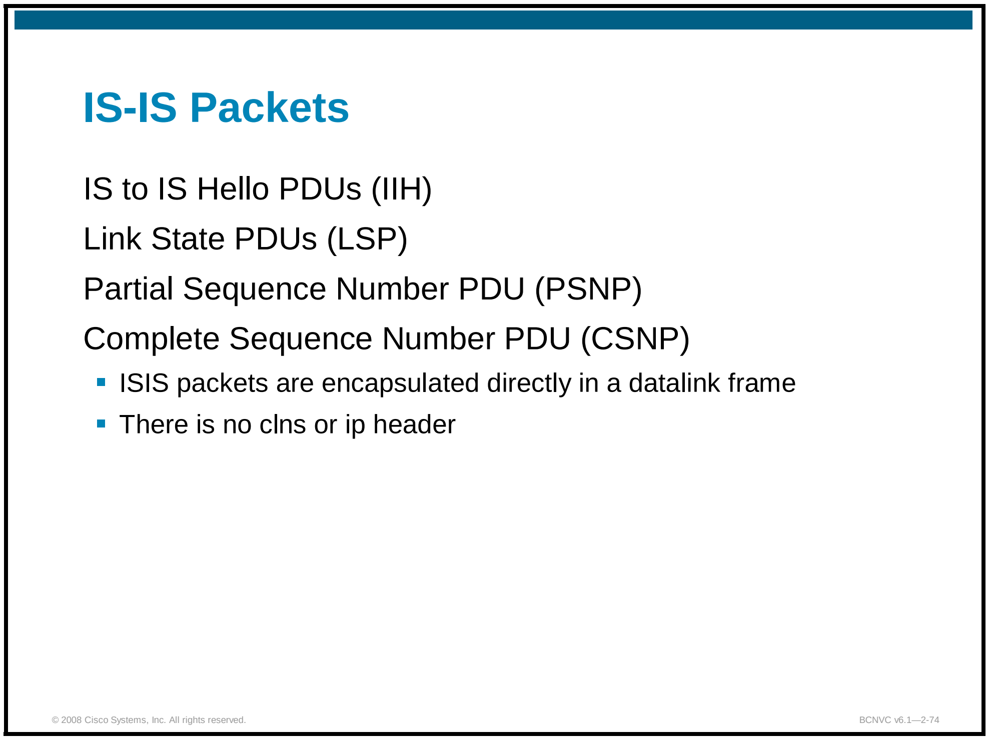
IS-IS areas provide a means for scaling routing in the IS-IS domain. Regular IS-IS areas and the backbone interconnecting them are organized into a two-level routing hierarchy. Routing within an area is referred to as Level 1 routing. Routing between the respective areas in a domain is referred to as Level 2 routing. It is interesting to note that, although Level 1 routing is restricted only to the confines of each area, Level 2 routing occurs within the stretch of the backbone, which can overlap well into any area based on configuration of the routers.

IS-IS routers can be Level 1 only (L1), Level 2 only (L2), or both Level 1 and Level 2 (Level 1–2), based on their configuration. The configuration of a router determines the type of adjacency (Level 1 or Level 2) that it can form with its neighbors, regardless of the type of link. This, in turn, determines the level of routing (Level 1 or Level 2) that a router can participate in.

In the default mode of operation, Cisco routers are Level 1–2 and can form any kind of adjacency with their neighbors. A router in one area can form only a Level 2 adjacency with a router in another area, so only Level 2 routing occurs between them. However, depending on their configuration, two routers in the same area can form a Level 1 adjacency or both Level 1 and Level 2 adjacencies with each other.

Typically, routers that are Level 2, by virtue of their connectivity to the backbone, also engage in Level 1 routing within their respective areas, making them Level 1–2 routers. Level 1–2 routers facilitate access to other areas for Level 1–only routers in the area. Level 1–2 routers flag their connectivity to the backbone in their Level 1 routing advertisements.

IS-IS Packets



IS-IS Packets

IS to IS Hello PDUs (IIH)

Link State PDUs (LSP)

Partial Sequence Number PDU (PSNP)

Complete Sequence Number PDU (CSNP)

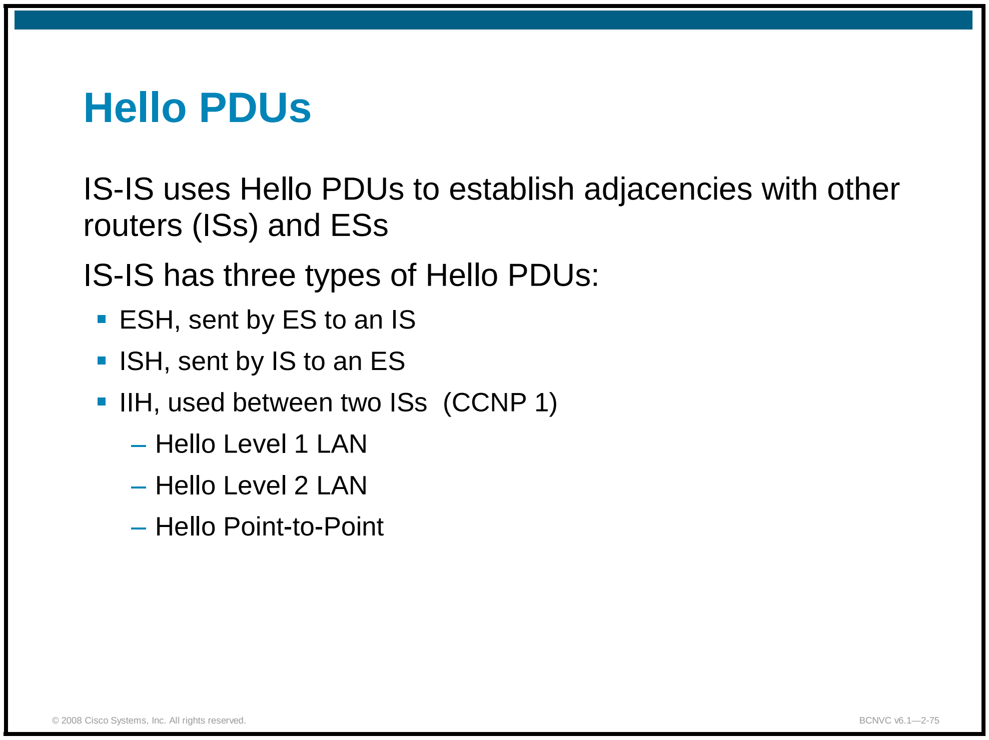
- ISIS packets are encapsulated directly in a datalink frame
- There is no clns or ip header

BCNVC v6.1—2-74

As a link-state protocol, IS-IS works by gathering reliable and complete information about the routing environment through the use of special packets known as Link-state Protocol Data Units (LSPs). A protocol data unit (PDU) also means a packet. Each router generates an LSP, which captures local link-state information describing connected links, neighbor routers, IP subnets, related metric information, and so forth. Copies of the LSP are distributed to all routers in a specific area through a process referred to as flooding. Ultimately, all routers in an area obtain every other router's LSP and synchronize their databases. Because the area link-state database is used for only intra-area routing (also referred to as Level 1 routing), it is called the Level 1 link-state database. The Level 2 routers interconnected into the backbone similarly maintain a Level 2 link-state database through the exchange of Level 2 LSPs. Best paths though the network are resolved by running the SPF algorithm over the information in the Level 1 and Level 2 databases separately.

These will be discussed in the following pages.

Hello PDUs



Hello PDUs

IS-IS uses Hello PDUs to establish adjacencies with other routers (ISs) and ESs

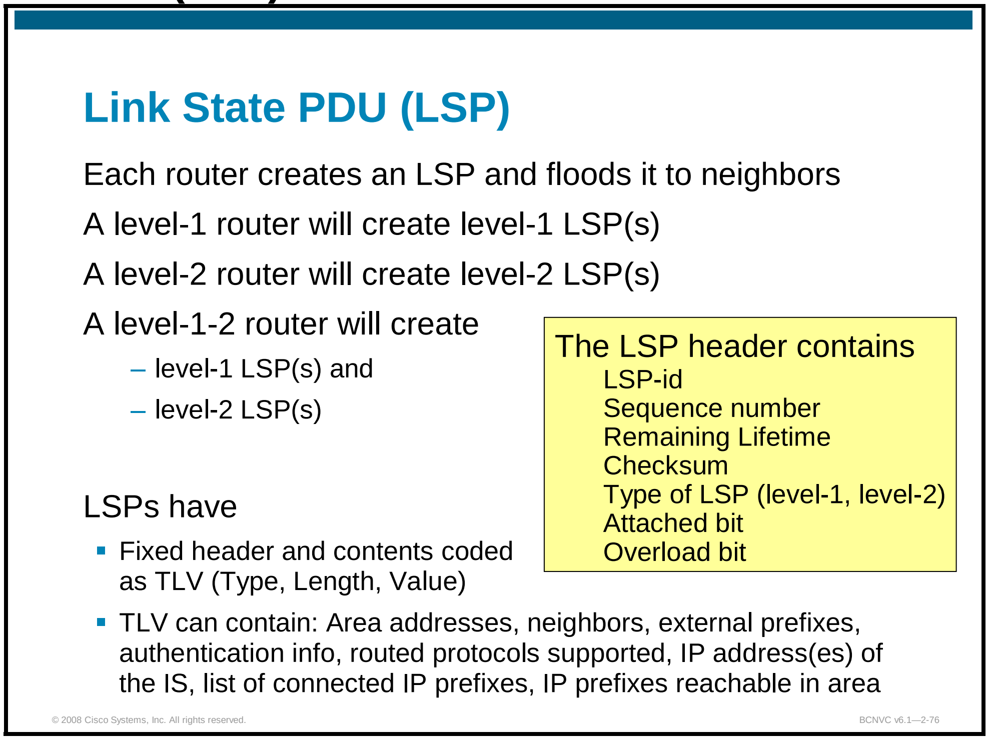
IS-IS has three types of Hello PDUs:

- ESH, sent by ES to an IS
- ISH, sent by IS to an ES
- IIH, used between two ISs (CCNP 1)
  - Hello Level 1 LAN
  - Hello Level 2 LAN
  - Hello Point-to-Point

BCNVC v6.1—2-75

Formation and maintenance of adjacencies between IS-IS routers take place through the exchange of special packets, referred to as hellos. Routers need to form both ES-IS and IS-IS adjacencies over either point-to-point or broadcast links. Even though ES-IS is not necessary for IP routing, IS-IS adjacency formation on point-to-point links is dependent on ES-IS adjacency detection on such links. Therefore, Cisco IOS Software enables the ES-IS protocol even if IS-IS is enabled for only IP routing. ES-IS uses end-system hellos (ESHs) and intermediate-system hellos (ISHs) for ES-IS adjacencies, while IS-IS uses intermediate system-to-intermediate system hellos (IIHs).

Link State PDU (LSP)



Link State PDU (LSP)

Each router creates an LSP and floods it to neighbors

A level-1 router will create level-1 LSP(s)

A level-2 router will create level-2 LSP(s)

A level-1-2 router will create

- level-1 LSP(s) and
- level-2 LSP(s)

The LSP header contains

LSP-id
Sequence number
Remaining Lifetime
Checksum
Type of LSP (level-1, level-2)
Attached bit
Overload bit

LSPs have

- Fixed header and contents coded as TLV (Type, Length, Value)
- TLV can contain: Area addresses, neighbors, external prefixes, authentication info, routed protocols supported, IP address(es) of the IS, list of connected IP prefixes, IP prefixes reachable in area

BCNVC v6.1—2-76

Each type of IS-IS packet is made up of a packet header and a number of optional variable-length fields referred to as Type-Length-Value (TLV) fields. The fields of each packet type vary slightly from each other, consisting of the generic fields and packet type specific fields.

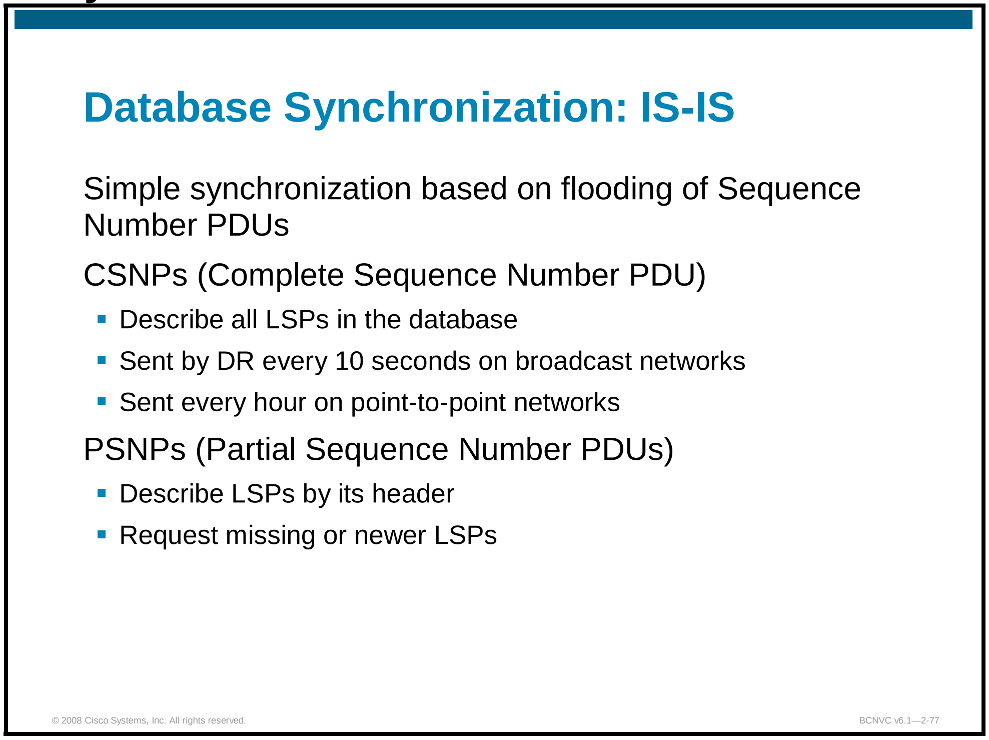
The generic header fields are described as follows:

- Intradomain Routing Protocol Discriminator— This is the network layer identifier assigned to IS-IS, as specified by ISO 9577. Its value is 0x83 in hexadecimal.
- Length Indicator— This specifies the length of the packet header fields in octets (bytes).
- Version/Protocol ID Extension— Currently this field has a value of 1.
- Version— The value of this field is 1.
- Reserved— These are unused bits; this field is set to 0.
- Maximum Area Addresses— This field includes values between 1 and 254 for the actual number. 0 implies a maximum of three addresses per area.

The TLV fields are so named because each is described by the following three attributes:

- Type— A 1-byte field containing a number code. ISO 10589 uses the word Code in place of Type. However, Type seems to be preferred in IETF and Cisco literature on IS-IS.
- Length— A 1-byte field that specifies the total length of TLVs of that type in the packet.
- Value— Content of the TLV. Typically, the value is made up of repeated blocks of similar information.

Database Synchronization: IS-IS



Database Synchronization: IS-IS

Simple synchronization based on flooding of Sequence Number PDUs

CSNPs (Complete Sequence Number PDU)

- Describe all LSPs in the database
- Sent by DR every 10 seconds on broadcast networks
- Sent every hour on point-to-point networks

PSNPs (Partial Sequence Number PDUs)

- Describe LSPs by its header
- Request missing or newer LSPs

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—2-77

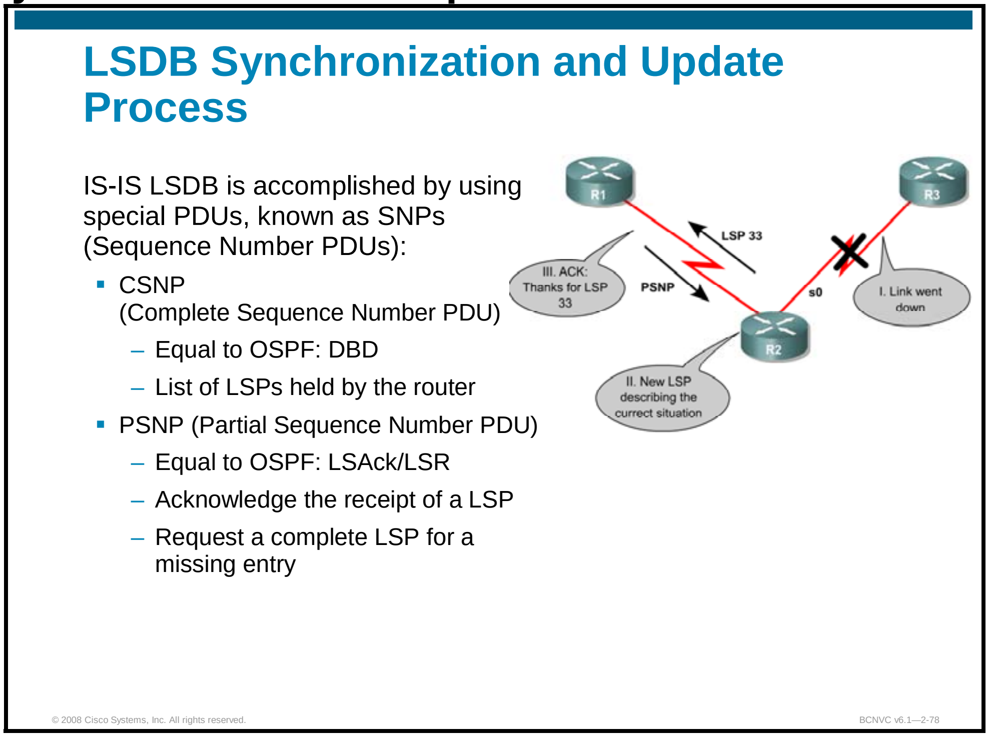
In addition to hello PDUs, there are two types of SNPs:

- Partial sequence number PDUs (PSNP)— Contain summaries of only a subset of known LSPs and solicit newer versions of a complete LSP or acknowledge receipt of LSPs
- Complete sequence number PDUs (CSNP)— Contain summaries of all LSPs known by the issuing router

Each LSP summary in a CSNP or PSNP consists of the following attributes from the header of the original LSP:

- Remaining lifetime
- LSP ID
- LSP sequence number
- LSP checksum

LSDB Synchronization and Update Process



LSDB Synchronization and Update Process

IS-IS LSDB is accomplished by using special PDUs, known as SNPs (Sequence Number PDUs):

- CSNP
  (Complete Sequence Number PDU)
  - Equal to OSPF: DBD
  - List of LSPs held by the router
- PSNP (Partial Sequence Number PDU)
  - Equal to OSPF: LSAck/LSR
  - Acknowledge the receipt of a LSP
  - Request a complete LSP for a missing entry

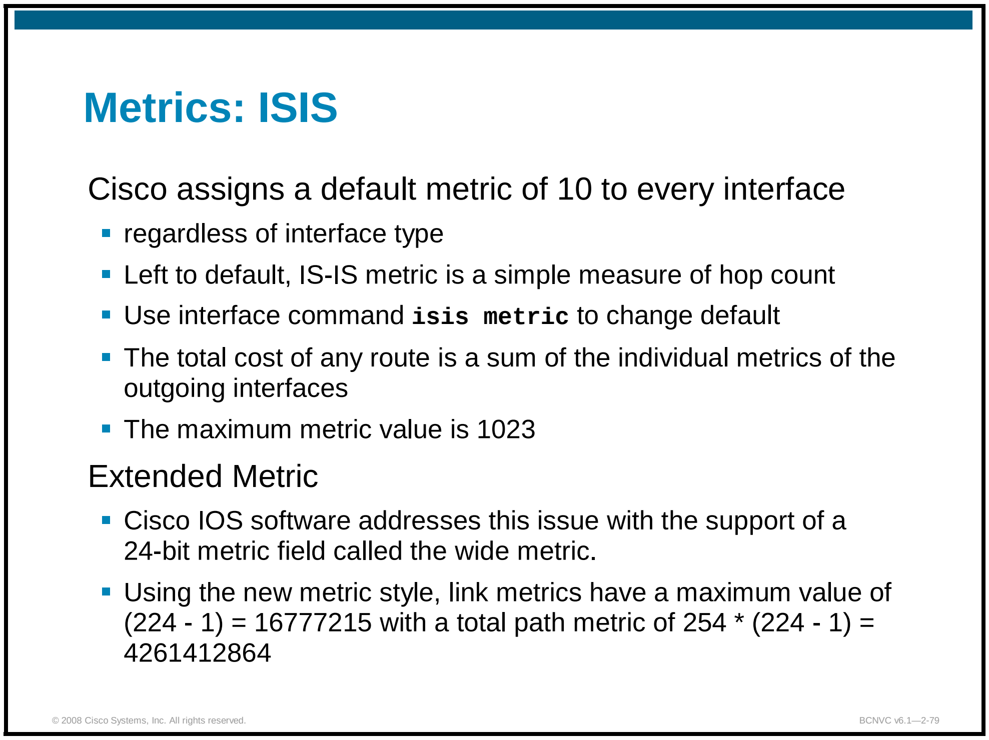
R1
R3
LSP 33
III. ACK: Thanks for LSP 33
PSNP
s0
I. Link went down
R2
II. New LSP describing the currect situation

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—2-78

Shown is an example of how CSNPs and PSNPs are used in the update process.

Metrics: ISIS



Metrics: ISIS

Cisco assigns a default metric of 10 to every interface

- regardless of interface type
- Left to default, IS-IS metric is a simple measure of hop count
- Use interface command **`isis metric`** to change default
- The total cost of any route is a sum of the individual metrics of the outgoing interfaces
- The maximum metric value is 1023

Extended Metric

- Cisco IOS software addresses this issue with the support of a 24-bit metric field called the wide metric.
- Using the new metric style, link metrics have a maximum value of (224 - 1) = 16777215 with a total path metric of 254 * (224 - 1) = 4261412864

BCNVC v6.1—2-79

The original IS-IS specification defines four different types of metrics. Cost, being the default metric, is supported by all routers. Delay, expense, and error are optional metrics. The delay metric measures transit delay, the expense metric measures the monetary cost of link utilization, and the error metric measures the residual error probability associated with a link. The Cisco implementation uses cost only.

While some routing protocols calculate the link metric automatically based on bandwidth (OSPF) or bandwidth/delay (Enhanced Interior Gateway Routing Protocol [EIGRP]), there is no automatic calculation for IS-IS. Using old-style metrics, an interface cost is between 1 and 63 (6 bit metric value). All links use the metric of 10 by default. The total cost to a destination is the sum of the costs on all outgoing interfaces along a particular path from the source to the destination, and the least-cost paths are preferred.

The total path metric was limited to 1023. This small metric value proved insufficient for large networks and provided too little granularity for new features such as Traffic Engineering and other applications, especially with high bandwidth links.

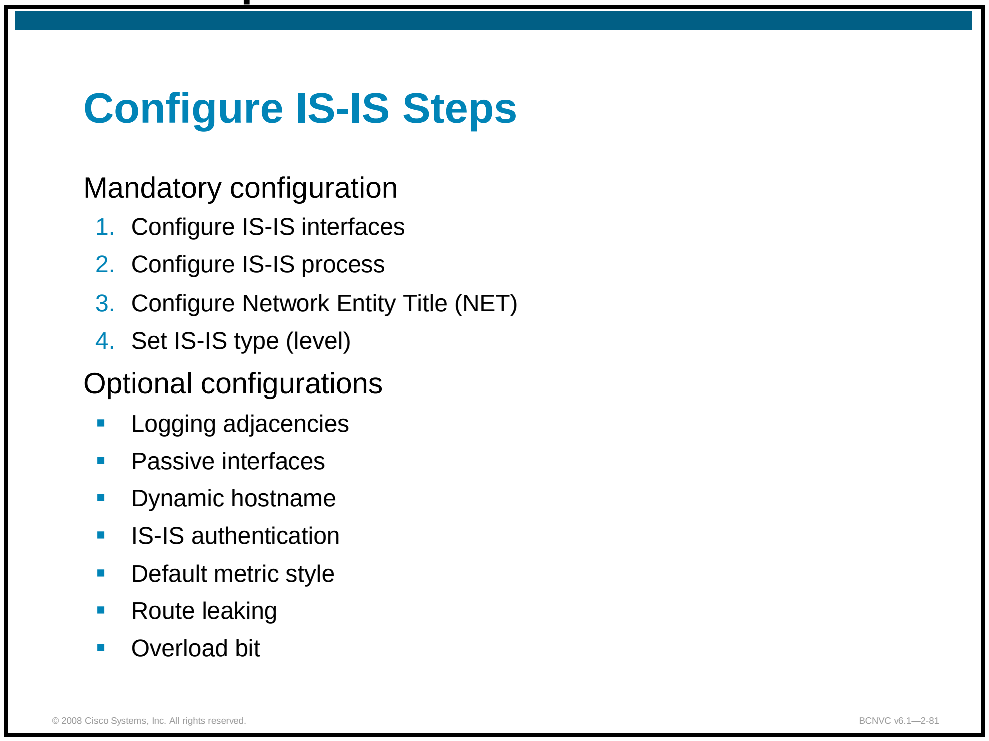
Cisco IOS Software addresses this issue with the support of a 24-bit metric field, the so-called "wide metric". Using the new metric style, link metrics now have a maximum value of 16777215 (224-1) with a total path metric of 4261412864 (254 x 224).

Deploying IS-IS in the IP network with wide metrics is recommended to enable finer granularity and to support future applications such as Traffic Engineering.

Running different metric styles within one network poses one serious problem: Link-state protocols calculate loop-free routes because all routers (within one area) calculate their routing table based on the same link-state database. This principle is violated if some routers look at old-style (narrow), and some at new-style (wider) TLVs. However, if the same interface cost is used for both the old- and new-style metrics, then the SPF will compute a loop-free topology.

Configure and Verify IS-IS

Configure IS-IS Steps



Configure IS-IS Steps

Mandatory configuration

1. Configure IS-IS interfaces
2. Configure IS-IS process
3. Configure Network Entity Title (NET)
4. Set IS-IS type (level)

Optional configurations

- Logging adjacencies
- Passive interfaces
- Dynamic hostname
- IS-IS authentication
- Default metric style
- Route leaking
- Overload bit

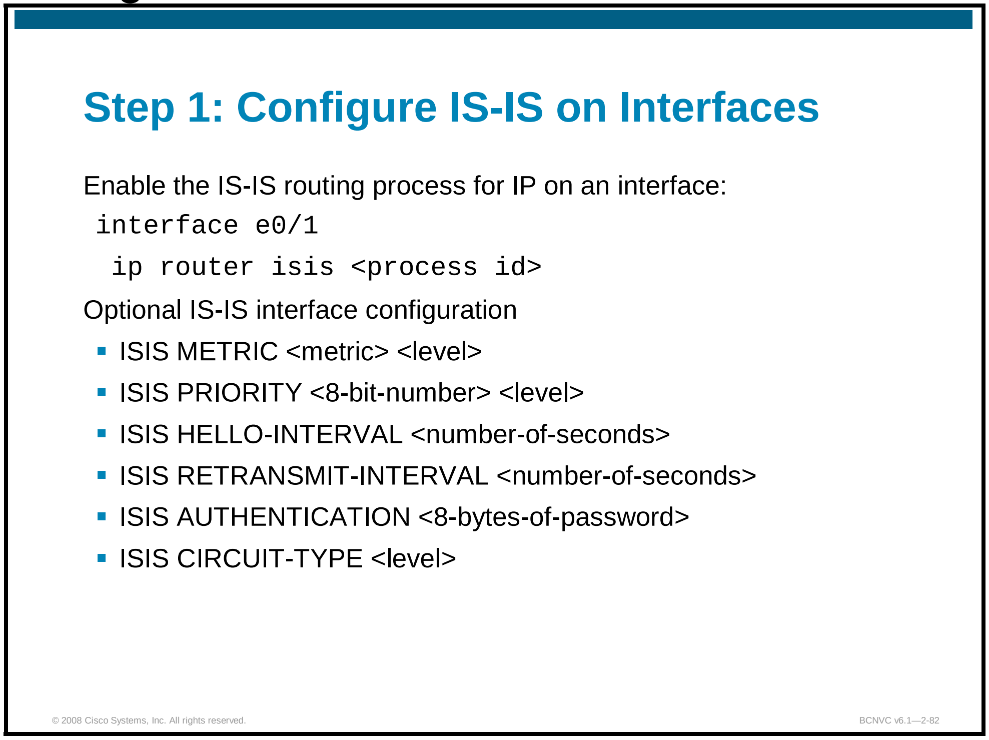
BCNVC v6.1—2-81

As with other routing protocols, enabling IS-IS requires that you create an IS-IS routing process, specify the router ID (in the form of a network entity title [NET]) to be associated with the local router, identify the type or level of IS-IS routing to be performed, and configure IS-IS interface parameters. Users should note that before the IS-IS routing process is useful, a NET must be assigned with the net command and some interfaces must have IS-IS enabled.

To satisfy the above requirements smoothly, configure the interfaces first, then configure the IS-IS process(es).

Routing interface parameters—Configurable parameters include IS-IS routing, interface output metric, retransmission interval, interface transmit delay, router priority, hello intervals, and authentication.

Step 1: Configure IS-IS on Interfaces



Step 1: Configure IS-IS on Interfaces

Enable the IS-IS routing process for IP on an interface:

```
 interface e0/1
  ip router isis <process id>
```

Optional IS-IS interface configuration

- ISIS METRIC <metric> <level>
- ISIS PRIORITY <8-bit-number> <level>
- ISIS HELLO-INTERVAL <number-of-seconds>
- ISIS RETRANSMIT-INTERVAL <number-of-seconds>
- ISIS AUTHENTICATION <8-bytes-of-password>
- ISIS CIRCUIT-TYPE <level>

To configure an interface to participate in an IS-IS routing process, use the **ip router isis** <process id> command in interface configuration mode. To remove an interface, use the **no** form of this command.

An interface cannot be part of more than one IS-IS process or area, except when an associated routing process performs both Level 1 and Level 2 routing. On media (such as WAN media, for example) where subinterfaces are supported, different subinterfaces can be configured for different IS-IS areas.

Optional IS-IS Interface Configurations

IS-IS metric (cost)—When setting up a new IS-IS process, backbone-wide metrics should be used. The original IS-IS used narrow metrics (six bits), which allows only 63 different values. Wide metrics are 32 bits, obviously providing much more scope and flexibility. Wide metrics should be set as the default in any ISP template for IS-IS. IS-IS has a uniform value of 10 for the link cost. So, to make different links have different costs, configure the IS-IS metric manually. Having only six bits is very restrictive, especially with the larger backbones, so the 32-bit metric makes more sense from the start.

DIS selection— IS-IS priority 100 (the default is 64).

Use this feature in your IS-IS network. Forcibly set your DIS per broadcast segment so that they are known. Choose your most powerful, or most idle routers. Try to keep the DIS limited to one segment per router.

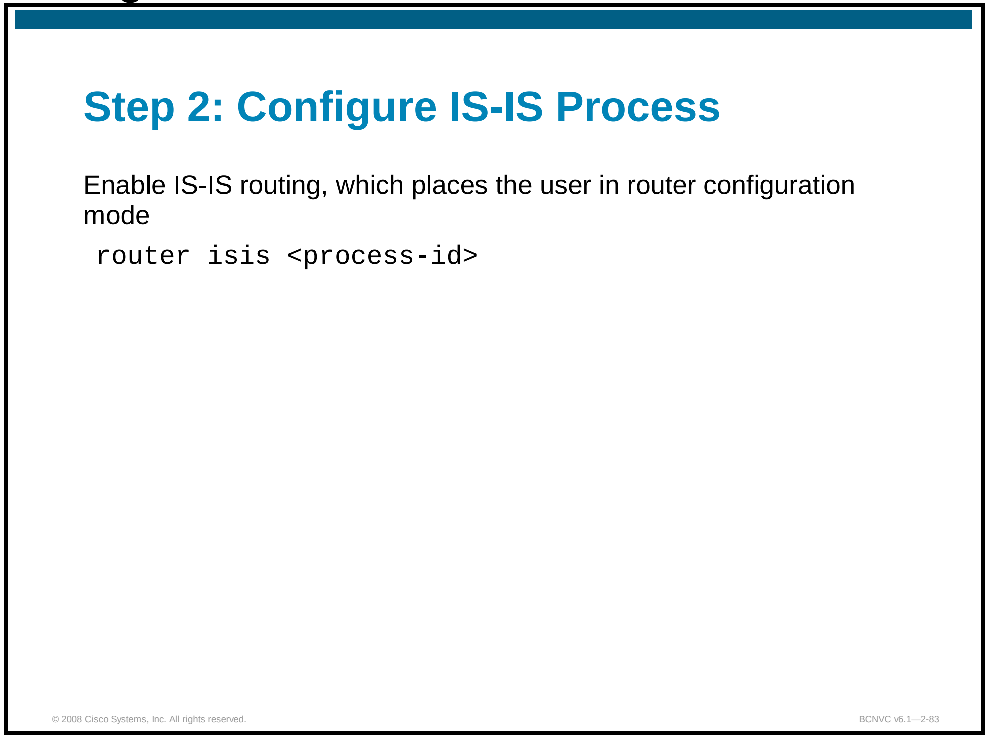
Hello/retransmit timers— **isis hello-interval 5** (the default is 10).

isis retransmit-interval 3 (the default is five).

This allows for faster network awareness of a failure, and can result in faster reconvergence, but requires more router CPU use and generates more overhead.

isis circuit-type—allows you to control the type of adjacencies formed on an interface (the default is L1/2).

Step 2: Configure IS-IS Process



Step 2: Configure IS-IS Process

Enable IS-IS routing, which places the user in router configuration mode

```
router isis <process-id>
```

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—2-83

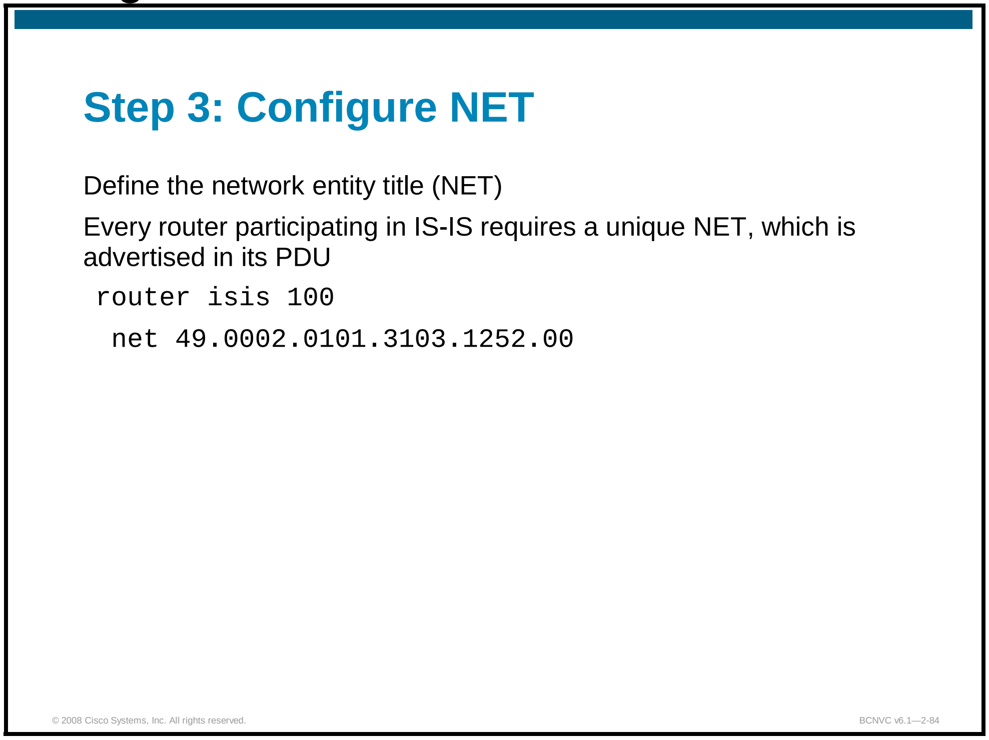
To configure an IS-IS routing process, use the **router isis** <process id> command in global configuration mode. To remove an IS-IS process, use the **no** form of this command.

Unlike other routing protocols, enabling IS-IS requires that you create an IS-IS routing process and assign it to a specific interface, rather than to a network. You can specify more than one IS-IS routing process per Cisco device, using multi-area IS-IS configuration.

In general, each routing process corresponds to an area. By default, the first instance of the routing process configured performs both intra-area (Level 1) and interarea (Level 2) routing. You can configure additional router instances, which are automatically treated as Level 1 areas. Routing parameters for each instance of the IS-IS routing process must be configured individually.

You can configure at most only one IS-IS routing process to perform Level 2 (interarea) routing. A particular Level type can be set per IS-IS routing instance, using the **is-type** command, which is discussed later in this chapter.

Step 3: Configure NET



Step 3: Configure NET

Define the network entity title (NET)

Every router participating in IS-IS requires a unique NET, which is advertised in its PDU

```
router isis 100
 net 49.0002.0101.3103.1252.00
```

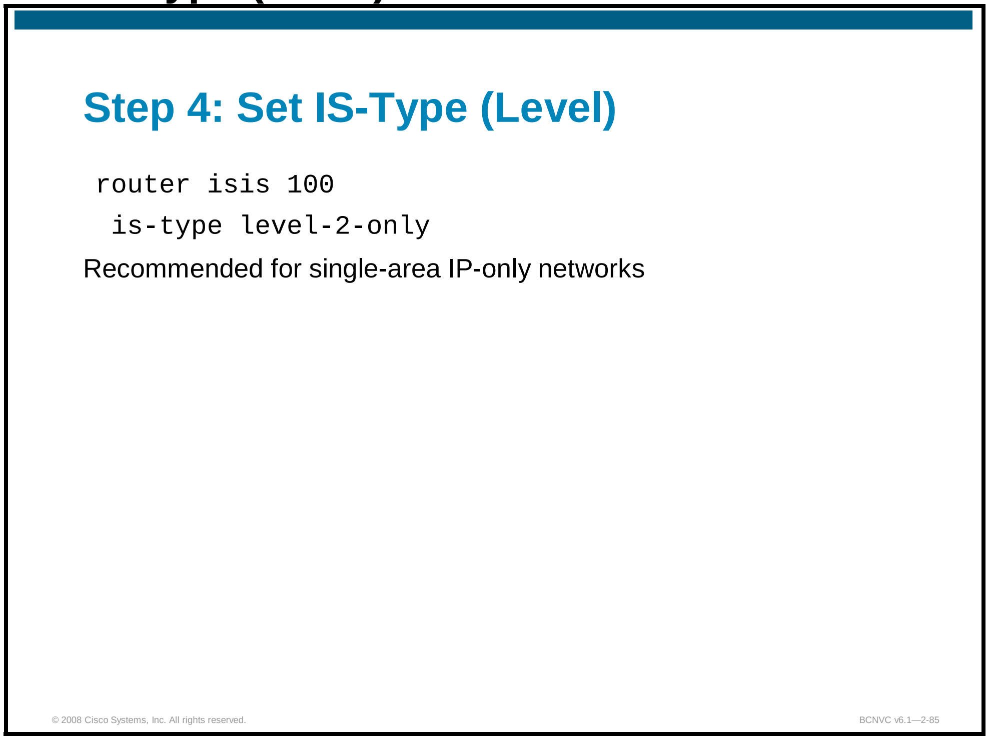
© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—2-84

To configure an IS-IS network entity title (NET) for a Connectionless Network Service (CLNS) routing process, use the **net** command in router configuration mode. To remove a NET, use the **no** form of this command.

Under most circumstances, only one NET must be configured. A maximum of three NETs per router are allowed. In rare circumstances, it is possible to configure two or three NETs. In such a case, the area that this router is in has three area addresses. There is still only one area, but it has an additional maximum of three area addresses. Configuring multiple NETs can be temporarily useful in the case of network reconfiguration where multiple areas are merged, or where one area is split into additional areas. Multiple area addresses enable you to renumber an area individually as needed. If you are configuring multi area IS-IS, the area ID must be unique, but the system ID portion of the NET must be the same for all IS-IS routing process instances.

Step 4: Set IS-Type (Level)



Step 4: Set IS-Type (Level)

```
router isis 100
 is-type level-2-only
```

Recommended for single-area IP-only networks

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—2-85

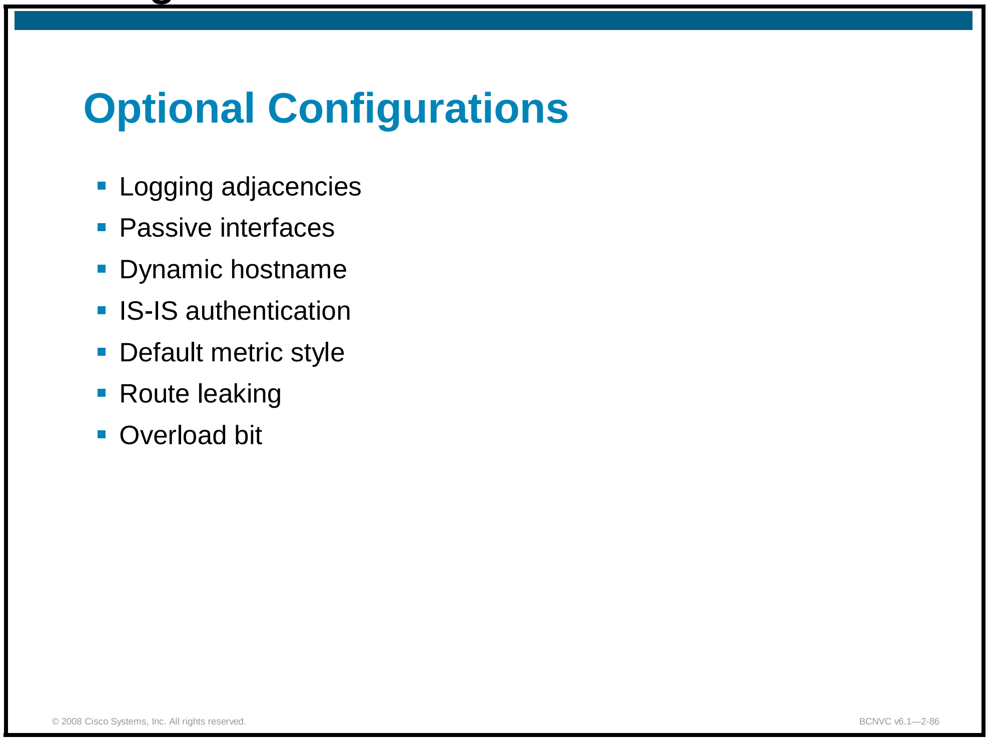
To configure the routing level for an instance of the IS-IS routing process, use the **is-type** command in router configuration mode. To reset the default value, use the **no** form of this command.

```
is-type [level-1 | level-1-2 | level-2-only]
```

In conventional IS-IS configurations, the router acts as both a Level 1 (intra-area) and a Level 2 (interarea) router.

In multi-area IS-IS configurations, the first instance of the IS-IS routing process configured is by default a Level 1-2 (intra-area and interarea) router. The remaining instances of the IS-IS process configured by default are Level 1 routers. You can also use the **is-type** command to configure Level 2 routing for an area, but it must be the only instance of the IS-IS routing process configured for Level 2 on the Cisco device. A Cisco router can support a maximum of 29 IS-IS processes.

Optional Configurations

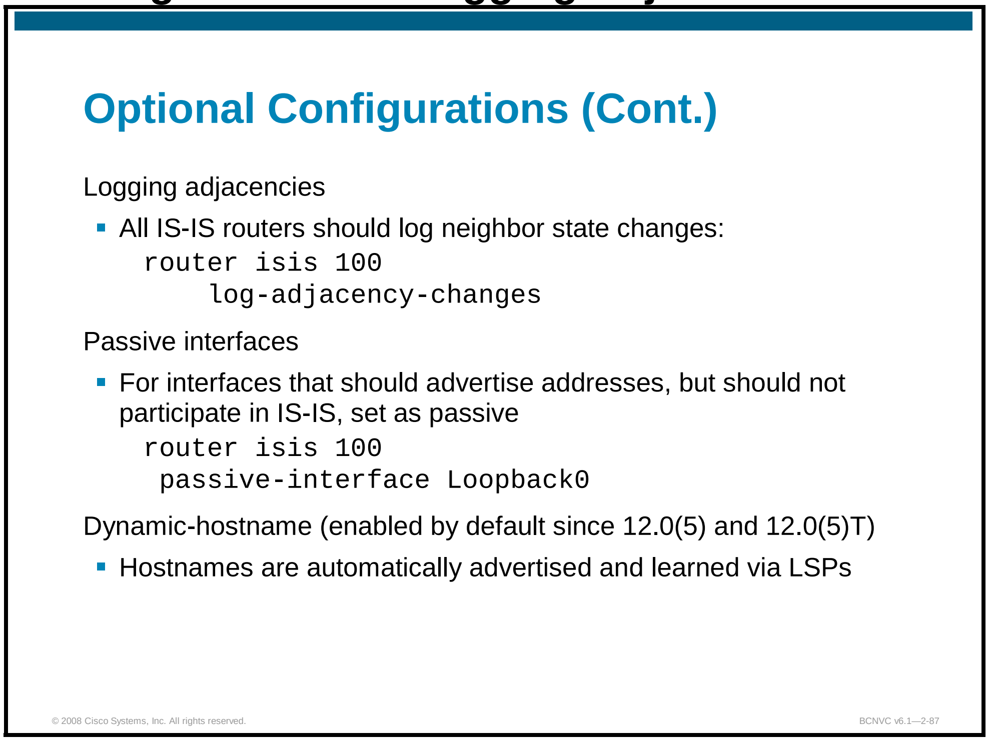


Optional Configurations

- Logging adjacencies
- Passive interfaces
- Dynamic hostname
- IS-IS authentication
- Default metric style
- Route leaking
- Overload bit

BCNVC v6.1—2-86

Optional Configurations — Logging Adjacencies



Optional Configurations (Cont.)

Logging adjacencies

- All IS-IS routers should log neighbor state changes:

```
router isis 100
    log-adjacency-changes
```

Passive interfaces

- For interfaces that should advertise addresses, but should not participate in IS-IS, set as passive

```
router isis 100
 passive-interface Loopback0
```

Dynamic-hostname (enabled by default since 12.0(5) and 12.0(5)T)

- Hostnames are automatically advertised and learned via LSPs

BCNVC v6.1—2-87

To generate a log message when an IS-IS adjacency changes state (up or down), use the **log-adjacency-changes** command in router configuration mode.

This may be very useful when monitoring large networks. Messages are logged using the system error message facility, and are of the form:

```
%CLNS-5-ADJCHANGE: ISIS: Adjacency to 0000.0000.0034 (Serial0) Up, new
adjacency
%CLNS-5-ADJCHANGE: ISIS: Adjacency to 0000.0000.0034 (Serial0) Down, hold time
expired
```

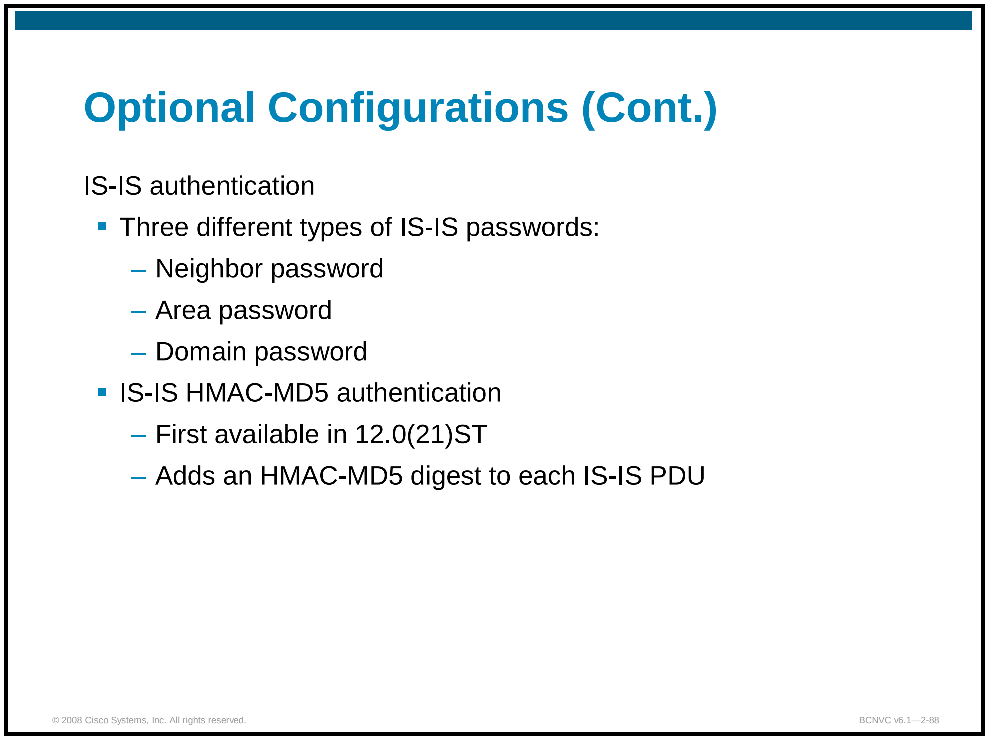
Passive Interface

To prevent other routers on a local network from learning about routes dynamically, you can keep routing update messages from being sent through a router interface. Keeping routing update messages from being sent through a router interface prevents other systems on the interface from learning about routes dynamically. This feature applies to all IP-based routing protocols except BGP.

OSPF and IS-IS behave somewhat differently. In OSPF, the interface address you specify as passive appears as a stub network in the OSPF domain. OSPF routing information is neither sent nor received through the specified router interface. In IS-IS, the specified IP addresses are advertised without actually running IS-IS on those interfaces.

Optional Configurations — dynamic-hostname

Since Cisco IOS version 12.0(5) and 12.0(5)T, a new feature known as dynamic hostname (discovery) has been added to automatically advertise and learn router hostnames via LSPs. These hostnames can be seen in **show isis hostname**, **show neighbor** and **show topology** command output.



Optional Configurations (Cont.)

IS-IS authentication

- Three different types of IS-IS passwords:
  - Neighbor password
  - Area password
  - Domain password
- IS-IS HMAC-MD5 authentication
  - First available in 12.0(21)ST
  - Adds an HMAC-MD5 digest to each IS-IS PDU

BCNVC v6.1—2-88

IS-IS Authentication

IS-IS allows for authentication to be established for several different levels of security, including:

Neighbor password —Configured per interface with the **isis password** <password> {level-1 | level-2} command, this password is contained in hello PDUs, and can prevent adjacencies from forming when password reliability is not achieved.

Area password —Layer 1 or Layer 1/Layer 2 same-type routers in the same area can implement a password. Configuration is performed in the IS-IS router configuration, with the **area-password** <password> command. If some routers are being set up to support area authentication, then all routers in that area or level must support that authorization. This password is contained in the Layer 1 LSPs and SNPs, and it can prevent Layer 1 LSPs from being accepted into all routers.

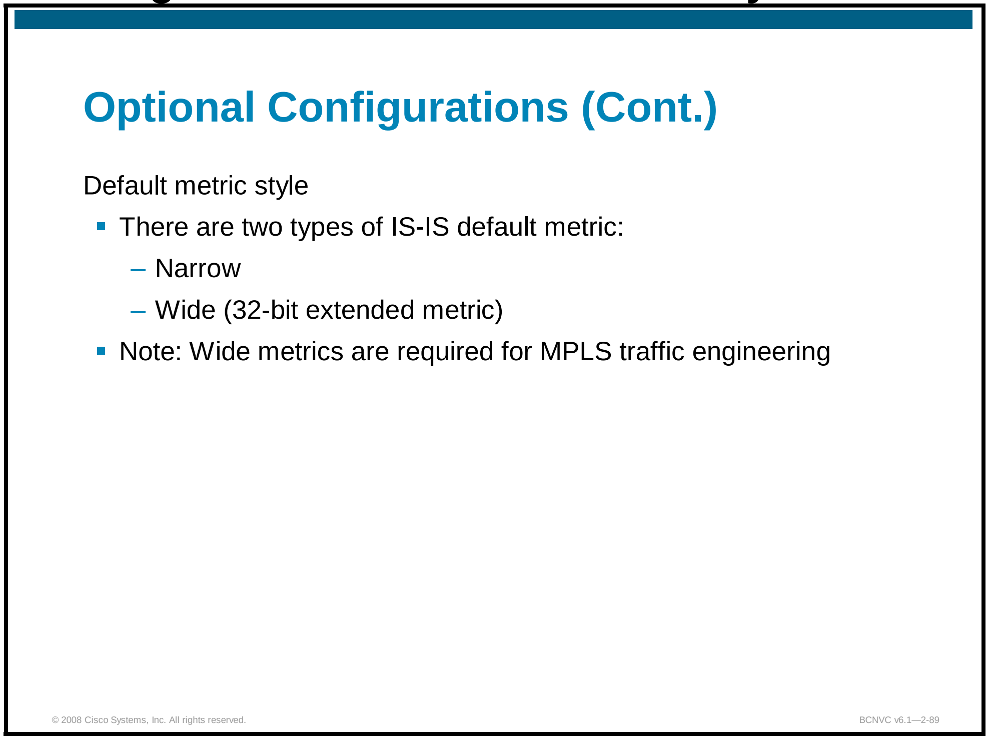
Domain password —Layer 2 or Layer 1/2 routers in the same IS-IS routing domain can share a password. Configuration is performed in the IS-IS router configuration with the **domain-password** <password> command. The same password must be used on all L2 routers. The password is contained in the L2 LSPs and SNPs, and can cause some routing loops if not configured correctly.

IS-IS HMAC-MD5 authentication—The IS-IS HMAC-MD5 authentication feature adds an HMAC-MD5 digest to each IS-IS PDU. HMAC is a mechanism for message authentication codes using cryptographic hash functions. The digest allows authentication at the IS-IS routing protocol level, which prevents unauthorized routing messages from being injected into the network routing domain.

IS-IS has five packet types: link-state packet (LSP), LAN hello, serial hello, CSNP, and PSNP. The IS-IS HMAC-MD5 authentication or the clear text password authentication can be applied to all five types of PDU. The authentication can be enabled on different IS-IS levels independently. The interface-related PDUs (LAN hello, serial hello, CSNP, and PSNP) can be enabled with authentication on different interfaces, with different levels and different passwords.

The HMAC-MD5 mode cannot be mixed with the clear text mode on the same authentication scope (LSP or interface). However, you can use one mode for LSP and another mode for some interfaces, for example. If mixed modes are intended, different keys should be used for different modes in order not to compromise the encrypted password in the PDUs.

Optional Configurations — Default Metric Style



Optional Configurations (Cont.)

Default metric style

- There are two types of IS-IS default metric:
  - Narrow
  - Wide (32-bit extended metric)
- Note: Wide metrics are required for MPLS traffic engineering

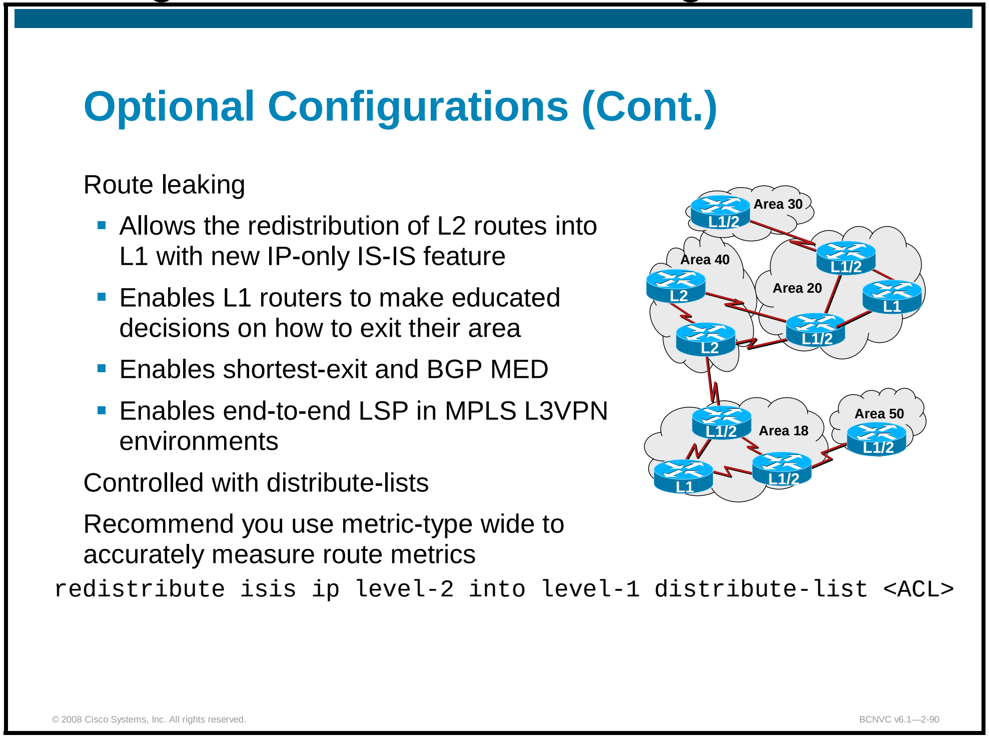
© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—2-89

An enhancement to the default metric is now supported that allows you to incorporate 32-bit “wide” metrics. This increases the maximum range of link metric to 16,777,215, and path metric up to 4,261,412,864. Two major benefits of the “wide metric” include:

- Support of MPLS traffic engineering
- Allowance of finer granularity in setting and maintaining network policies

The default in Cisco routers is currently old-style metrics. Wide metric support can be configured with the router configuration **metric-style wide** command. It is critical that you deploy one metric style consistently across an autonomous system to maintain a loop-free topology.

Optional Configurations — Route Leaking



Optional Configurations (Cont.)

Route leaking

- Allows the redistribution of L2 routes into L1 with new IP-only IS-IS feature
- Enables L1 routers to make educated decisions on how to exit their area
- Enables shortest-exit and BGP MED
- Enables end-to-end LSP in MPLS L3VPN environments

Controlled with distribute-lists

Recommend you use metric-type wide to accurately measure route metrics

```
redistribute isis ip level-2 into level-1 distribute-list <ACL>
```

BCNVC v6.1—2-90

Without route leaking, each Layer 1 router looks for the LSP with a set ATT bit to recognize its Layer 1/2 router for forwarding default gateway traffic. If there are multiple Layer 1/2 routers in an area, each router uses metrics to select one best default path. This is similar to not-so-stubby areas (NSSAs) in OSPF networks.

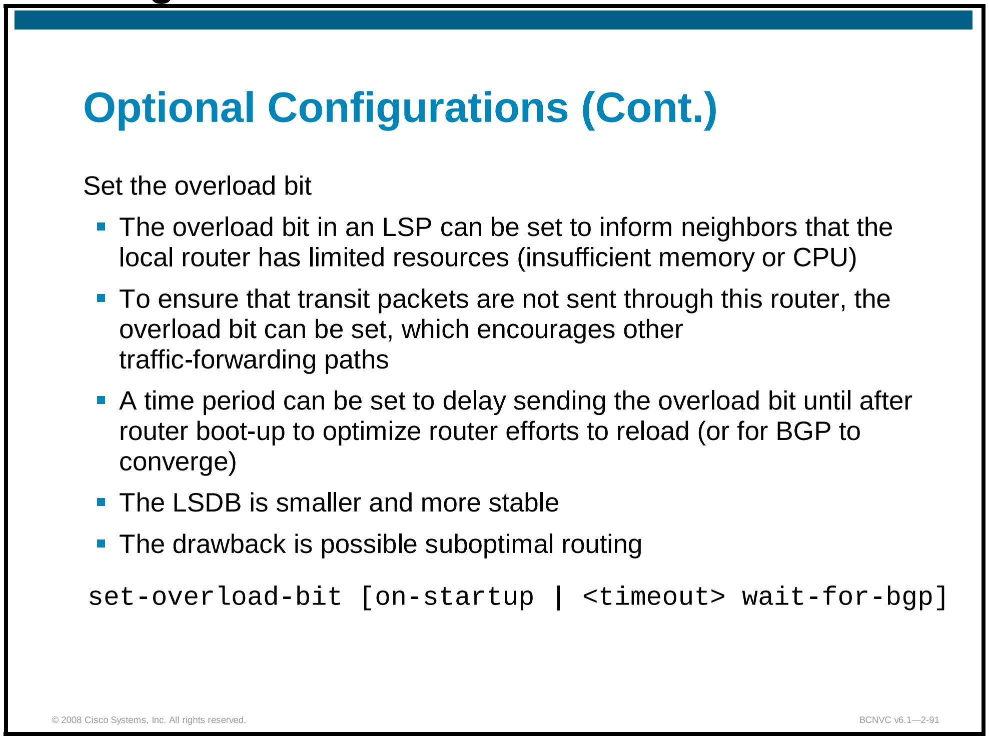
With route leaking, the Layer 2 routes can pass through the Layer 1/2 router and each Layer 1 router can collect these LSPs in their LSDB, and make accurate and best forwarding decisions by destination rather than by sending all external-to-the-area traffic to a default exit. It is recommended that you configure wide metrics when implementing route leaking to accurately measure route value.

In IS-IS LSP exchanges, the up/down bit is used to indicate whether or not the route defined in the TLV has been leaked. If the up/down bit is set to 0 the route was originated within that Layer 1 area. If the up/down bit is set to 1, the route has been redistributed into the area from Layer 2. The up/down bit is used to prevent routing information and forwarding loops. An Layer 1/Layer 2 router does not re-advertise into Layer 2 any Layer 1 routes that have the up/down bit set.

To configure route leaking, two steps must be completed. First, define which addresses should be leaked into the Level-1 area by configuring an access-list in global configuration mode. Then IS-IS route leaking is enabled with the **redistribute isis ip level-2 into level-1 distribute-list** <access-list number> router configuration command. Note that this leaking should be controlled by applying the access list to the **redistribute** command with the distribute-list keyword, and access-list must be an extended ip access-list (numbered between 100 and 199, and 2000 to 2699).

```
RouterC(config)#access-list 100 permit 10.131.31.0 0.0.0.255
RouterC(config)#access-list 100 deny all
RouterC(config)#router isis 150
RouterC(config-router)#redistribute isis ip level-2 into level-1
distribute-list 100
```

Optional Configurations — Set the Overload Bit



Optional Configurations (Cont.)

Set the overload bit

- The overload bit in an LSP can be set to inform neighbors that the local router has limited resources (insufficient memory or CPU)
- To ensure that transit packets are not sent through this router, the overload bit can be set, which encourages other traffic-forwarding paths
- A time period can be set to delay sending the overload bit until after router boot-up to optimize router efforts to reload (or for BGP to converge)
- The LSDB is smaller and more stable
- The drawback is possible suboptimal routing

```
set-overload-bit [on-startup | <timeout> wait-for-bgp]
```

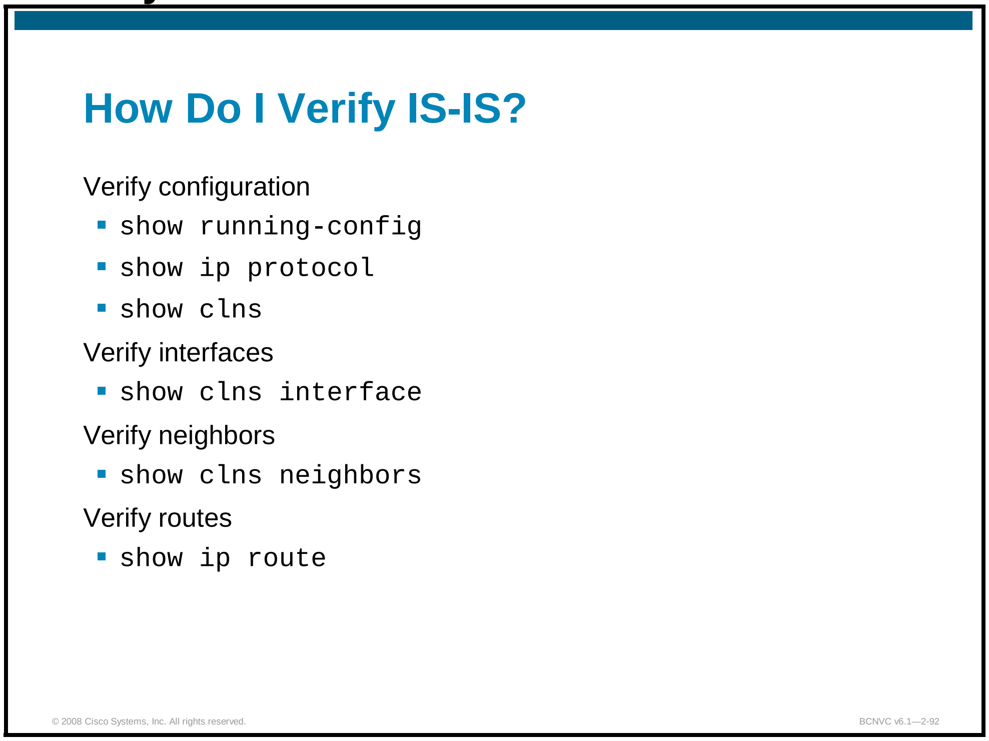
© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—2-91

During startup or in periods of network turbulence, a router may be overloaded with work. During this time it is not uncommon for the IGP to converge before BGP. When this happens, BGP traffic could be black-holed to the overloaded router. This is of particular importance to MPLS networks where Label Distribution Protocol (LDP) must reconverge before the Label Switched Path (LSP) can be re-established.

If an overload bit is set to wait for BGP and BGP fails to notify the router, the overload bit automatically clears after 10 minutes.

You can configure BGP parameters in conjunction with this IGP configuration. In BGP configuration, set the maximum initial delay before sending BGP updates (the default is two minutes).

How Do I Verify IS-IS?



How Do I Verify IS-IS?

Verify configuration

- `show running-config`
- `show ip protocol`
- `show clns`

Verify interfaces

- `show clns interface`

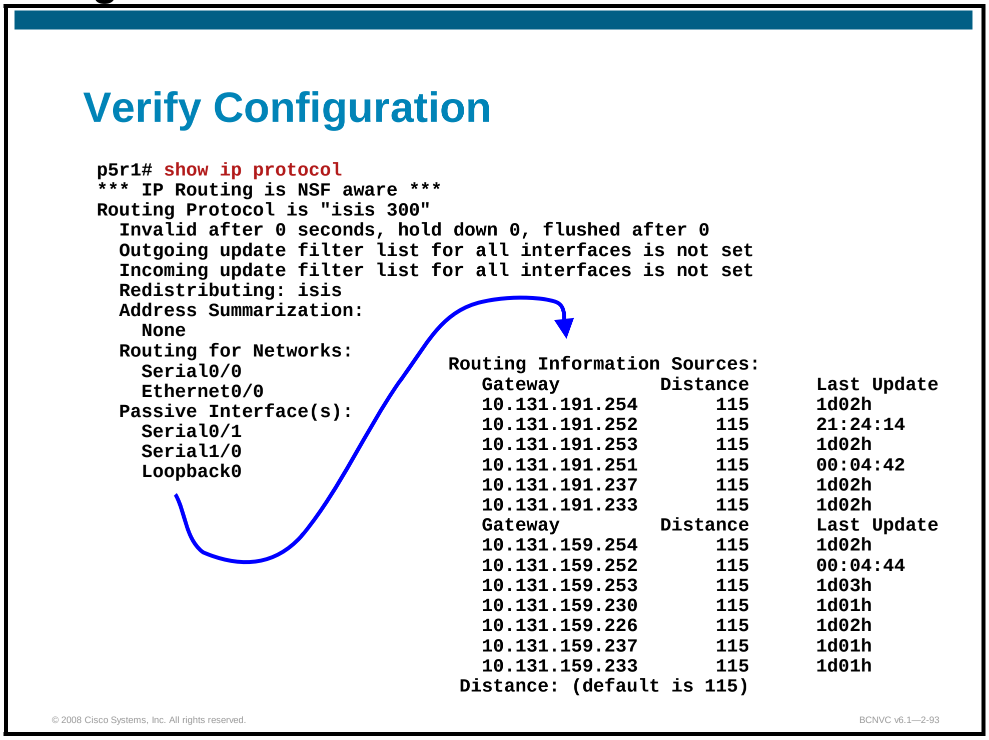
Verify neighbors

- `show clns neighbors`

Verify routes

- `show ip route`

Verify Configuration



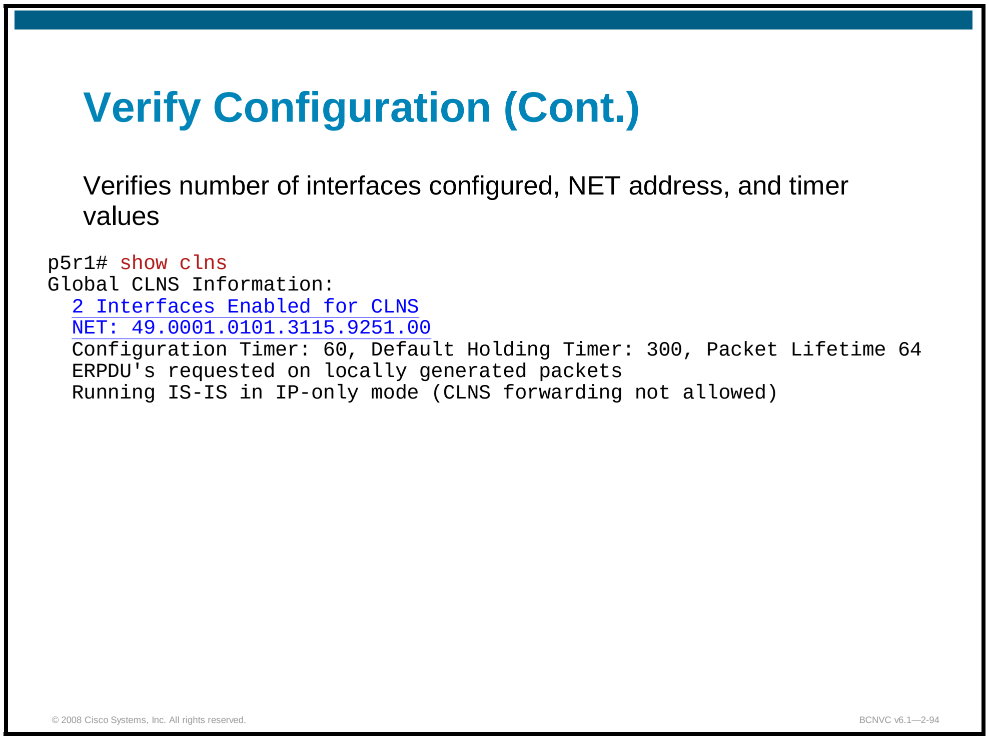
Verify Configuration

```
p5r1# show ip protocol
*** IP Routing is NSF aware ***
Routing Protocol is "isis 300"
  Invalid after 0 seconds, hold down 0, flushed after 0
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: isis
  Address Summarization:
    None
  Routing for Networks:
    Serial0/0
    Ethernet0/0
  Passive Interface(s):
    Serial0/1
    Serial1/0
    Loopback0
```

```
Routing Information Sources:
   Gateway          Distance      Last Update
   10.131.191.254        115      1d02h
   10.131.191.252        115      21:24:14
   10.131.191.253        115      1d02h
   10.131.191.251        115      00:04:42
   10.131.191.237        115      1d02h
   10.131.191.233        115      1d02h
   Gateway          Distance      Last Update
   10.131.159.254        115      1d02h
   10.131.159.252        115      00:04:44
   10.131.159.253        115      1d03h
   10.131.159.230        115      1d01h
   10.131.159.226        115      1d02h
   10.131.159.237        115      1d01h
   10.131.159.233        115      1d01h
 Distance: (default is 115)
```

BCNVC v6.1—2-93

Show running-configuration always provides a good way to check the configuration. However, sometimes it is difficult to see your own mistakes. The **show ip protocols** command can give you a different view of your configuration.



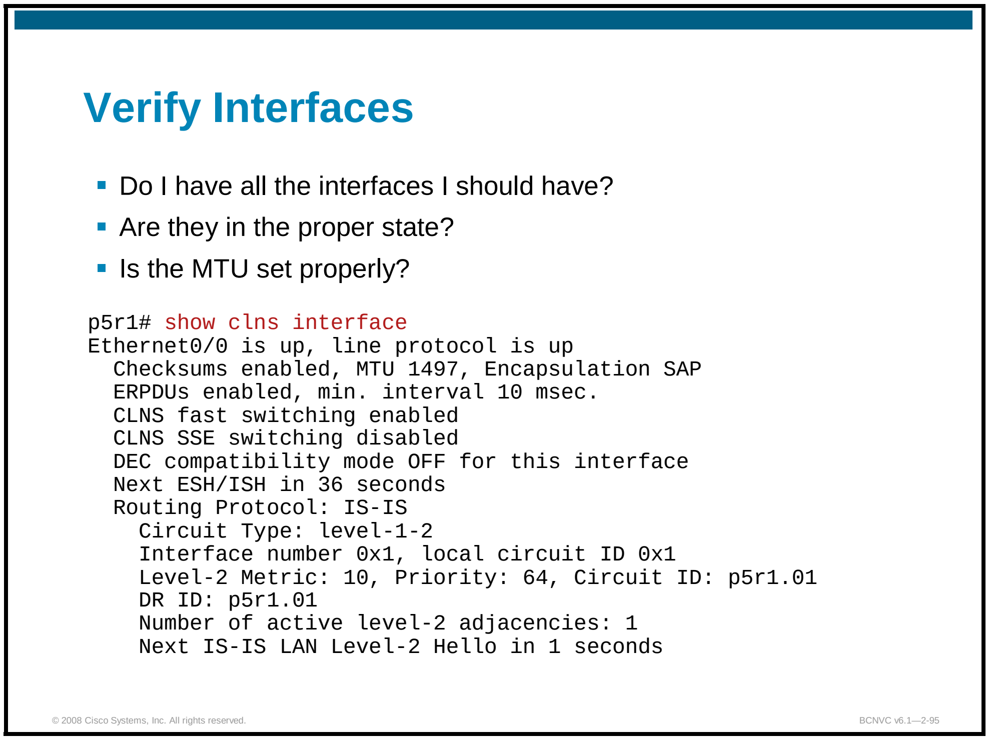
Verify Configuration (Cont.)

Verifies number of interfaces configured, NET address, and timer values

```
p5r1# show clns
Global CLNS Information:
  2 Interfaces Enabled for CLNS
  NET: 49.0001.0101.3115.9251.00
  Configuration Timer: 60, Default Holding Timer: 300, Packet Lifetime 64
  ERPDU's requested on locally generated packets
  Running IS-IS in IP-only mode (CLNS forwarding not allowed)
```

BCNVC v6.1—2-94

Verify Interfaces

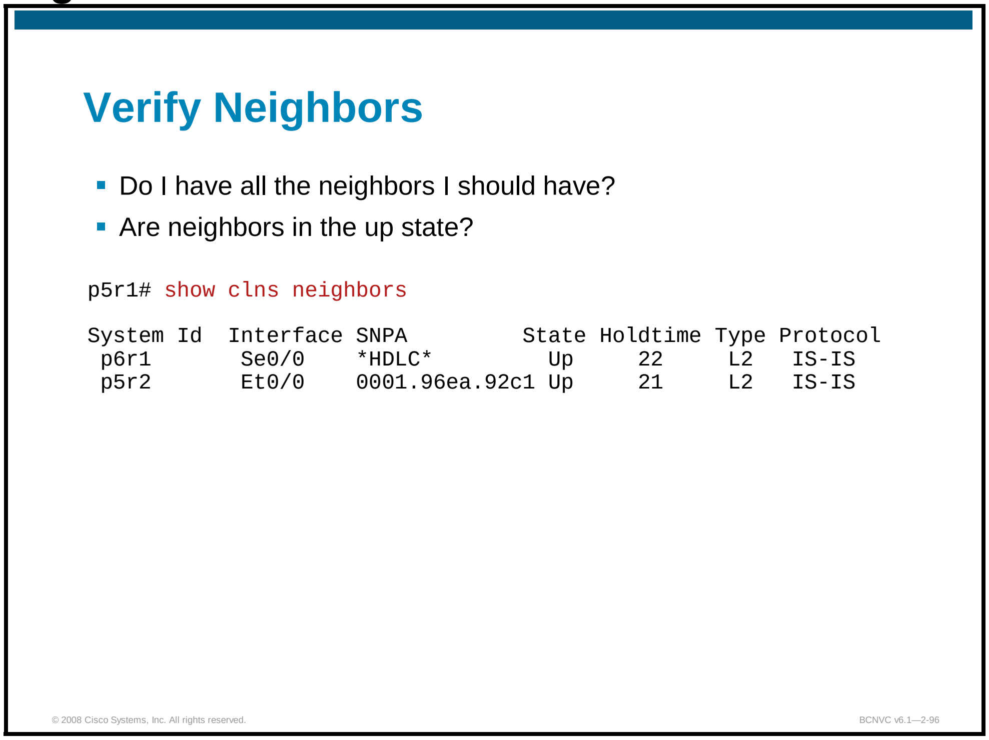


Verify Interfaces

- Do I have all the interfaces I should have?
- Are they in the proper state?
- Is the MTU set properly?

```
p5r1# show clns interface
Ethernet0/0 is up, line protocol is up
  Checksums enabled, MTU 1497, Encapsulation SAP
  ERPDUs enabled, min. interval 10 msec.
  CLNS fast switching enabled
  CLNS SSE switching disabled
  DEC compatibility mode OFF for this interface
  Next ESH/ISH in 36 seconds
  Routing Protocol: IS-IS
    Circuit Type: level-1-2
    Interface number 0x1, local circuit ID 0x1
    Level-2 Metric: 10, Priority: 64, Circuit ID: p5r1.01
    DR ID: p5r1.01
    Number of active level-2 adjacencies: 1
    Next IS-IS LAN Level-2 Hello in 1 seconds
```

Verify Neighbors



Verify Neighbors

- Do I have all the neighbors I should have?
- Are neighbors in the up state?

```
p5r1# show clns neighbors

System Id  Interface SNPA          State Holdtime Type Protocol
 p6r1       Se0/0     *HDLC*         Up     22      L2   IS-IS
 p5r2       Et0/0     0001.96ea.92c1 Up     21      L2   IS-IS
```

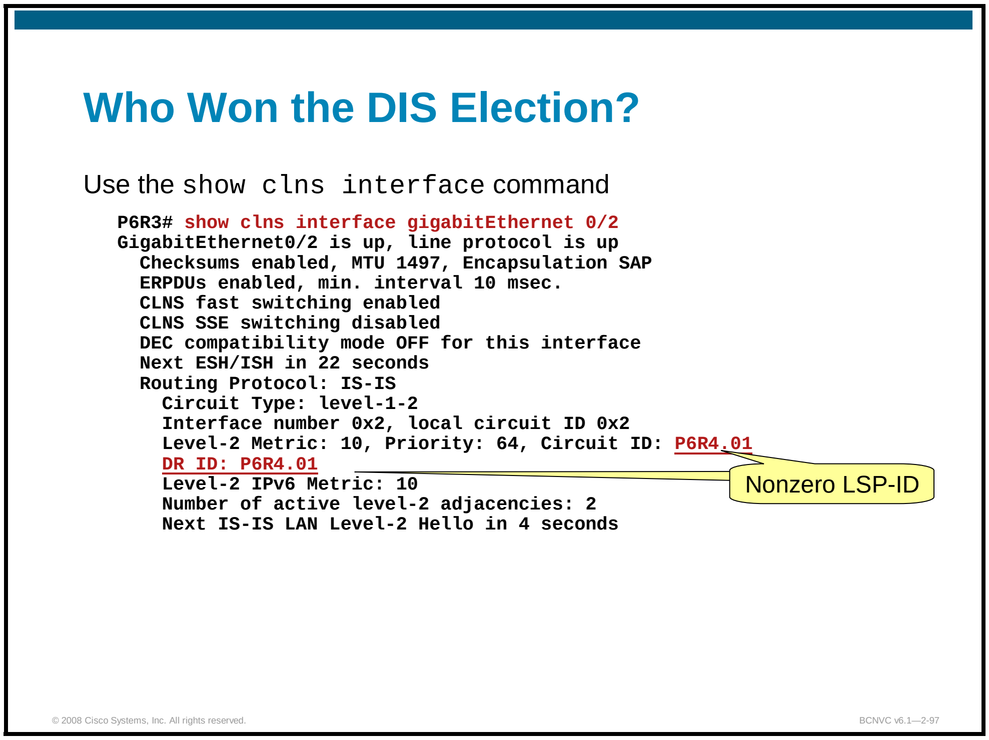
BCNVC v6.1—2-96

If you don’t have the appropriate neighbors, double-check your network statements to ensure the correct interfaces ended up in IS-IS.

One of the most common issues in IS-IS is the failure of two neighboring routers to become adjacent. There can be many causes. The following items can be verified when troubleshooting adjacency issues:

- Make sure the network type is the same on all routers attached to a media.
- Make sure you have the same maximum transmission unit (MTU) on all routers attached to a media. If the routers change to the INIT state and go no further, there may be an MTU mismatch.
- Make sure the authentication is set properly on both ends of the link.
- Run **debug isis adj-packets** to find out more about the cause of the problem. This command should be executed on both sides of a link. This debug command can be safely executed on a router without producing any problems.

Who Won the DIS Election?



Who Won the DIS Election?

Use the show `clns interface` command

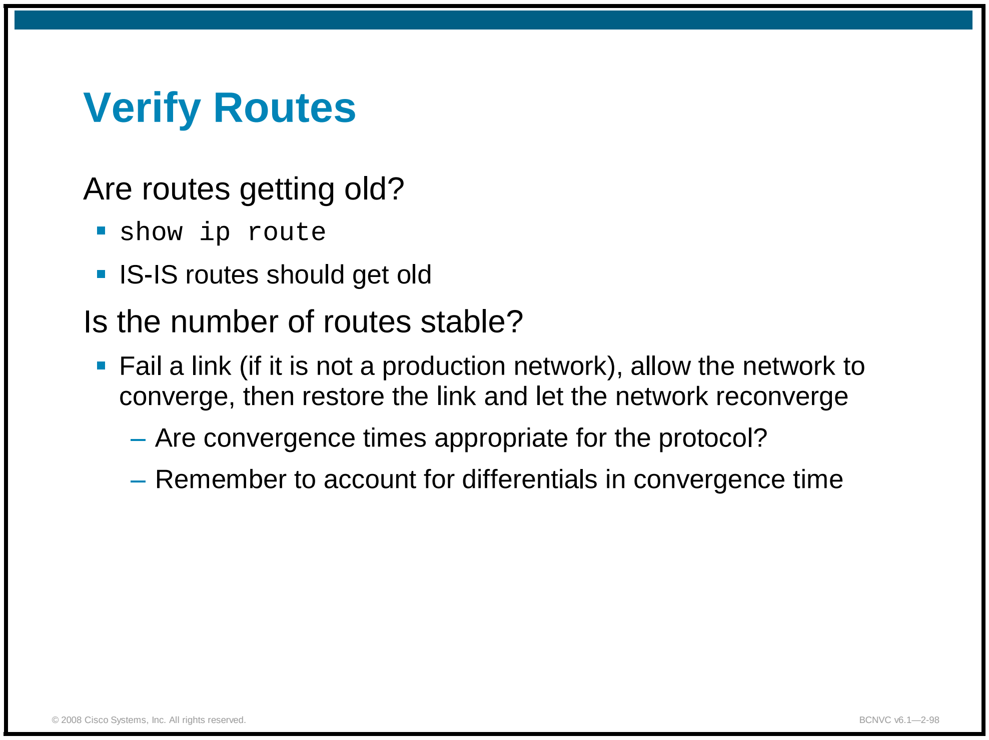
```
P6R3# show clns interface gigabitEthernet 0/2
GigabitEthernet0/2 is up, line protocol is up
  Checksums enabled, MTU 1497, Encapsulation SAP
  ERPDUs enabled, min. interval 10 msec.
  CLNS fast switching enabled
  CLNS SSE switching disabled
  DEC compatibility mode OFF for this interface
  Next ESH/ISH in 22 seconds
  Routing Protocol: IS-IS
    Circuit Type: level-1-2
    Interface number 0x2, local circuit ID 0x2
    Level-2 Metric: 10, Priority: 64, Circuit ID: P6R4.01
    DR ID: P6R4.01
    Level-2 IPv6 Metric: 10
    Number of active level-2 adjacencies: 2
    Next IS-IS LAN Level-2 Hello in 4 seconds
```

Nonzero LSP-ID

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—2-97

The router with the highest system ID becomes the Designated Intermediate System (DIS). The circuit ID highlighted above as P6R4 01 is from the designated router (P6R4). The circuit ID is a one-octet number that a router uses to uniquely identify an IS-IS interface.

Verify Routes



Verify Routes

Are routes getting old?

- `show ip route`
- IS-IS routes should get old

Is the number of routes stable?

- Fail a link (if it is not a production network), allow the network to converge, then restore the link and let the network reconverge
  - Are convergence times appropriate for the protocol?
  - Remember to account for differentials in convergence time

© 2008 Cisco Systems, Inc. All rights reserved.

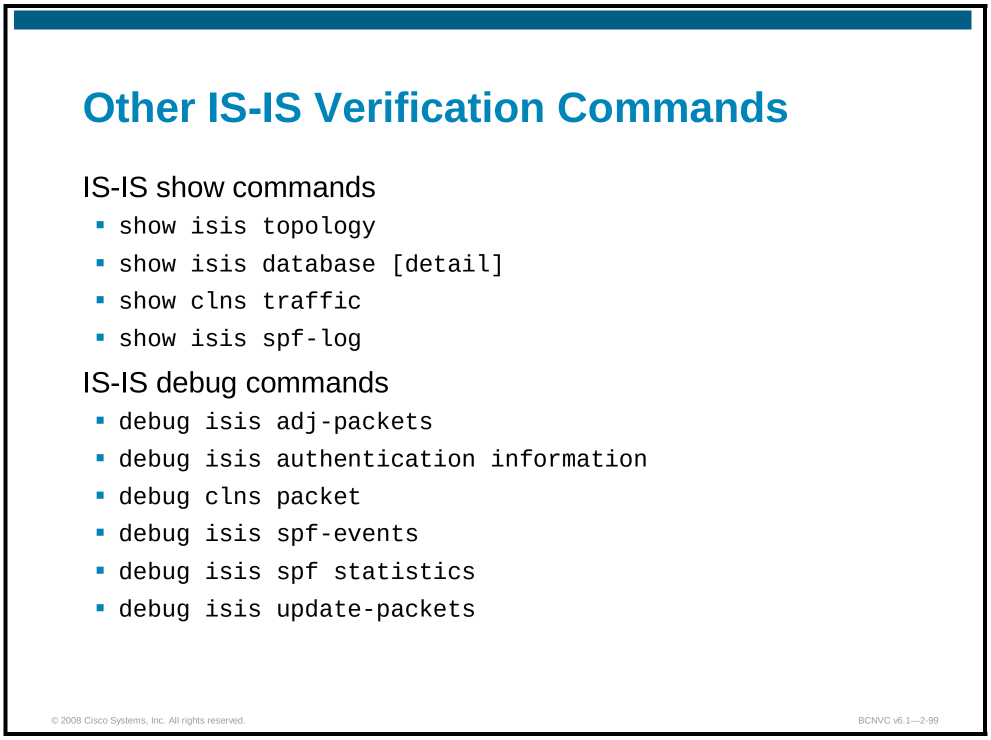
BCNVC v6.1—2-98

Routes should be as old as router uptime after a reboot. If not and you can't find a failed link, there is a configuration error.

If the routes ages are not appropriate, the change is a failure and should be reverted. The network is unstable and while you might have raw connectivity, performance will probably suffer and the network may fail altogether under load.

When no links are changing state, the number of routes should be stable. If you can't account for each convergence event by finding a failed or flapping link, then there is a route loop situation and the change should be reverted.

Other IS-IS Verification Commands



Other IS-IS Verification Commands

IS-IS show commands

- `show isis topology`
- `show isis database [detail]`
- `show clns traffic`
- `show isis spf-log`

IS-IS debug commands

- `debug isis adj-packets`
- `debug isis authentication information`
- `debug clns packet`
- `debug isis spf-events`
- `debug isis spf statistics`
- `debug isis update-packets`

debug isis adj-packets displays information on all adjacency-related activity such as Hello packets sent and received and IS-IS adjacencies going up and down.

debug isis authentication information enables IS-IS debugging authentication.

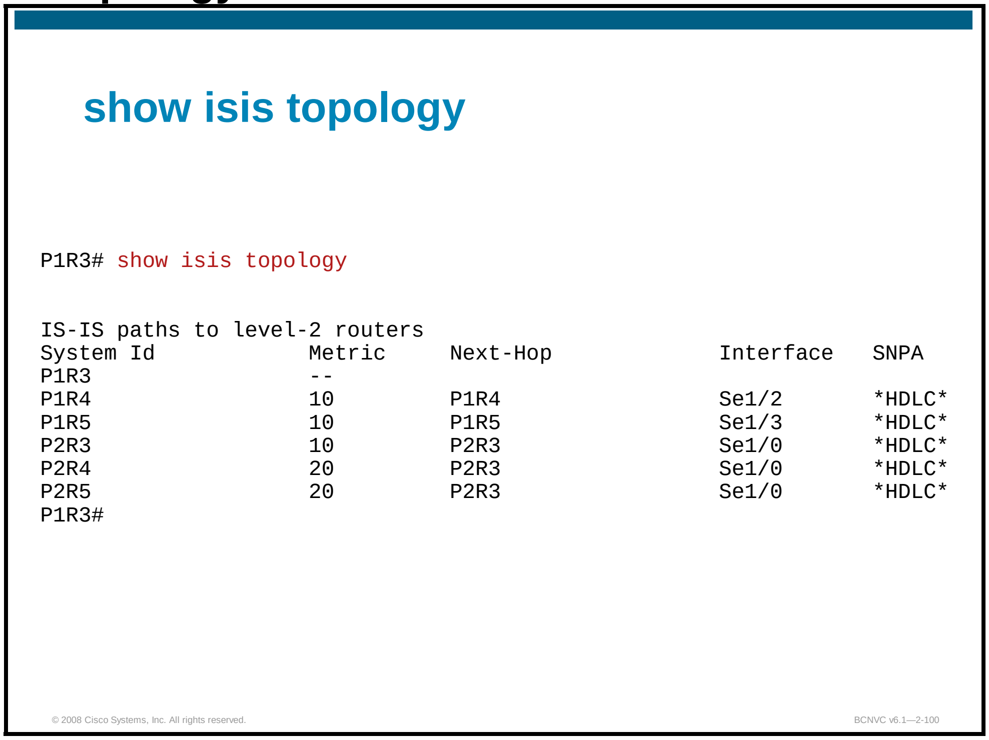
debug clns packet displays information about packet receipt and forwarding to the next interface.

debug isis spf-events displays a log of significant events during an IS-IS SPF computation.

debug isis spf statistics displays statistical information about building routes between IS-IS routers.

debug isis update-packets displays various sequence number PDUs and link-state packets that are detected by a router.

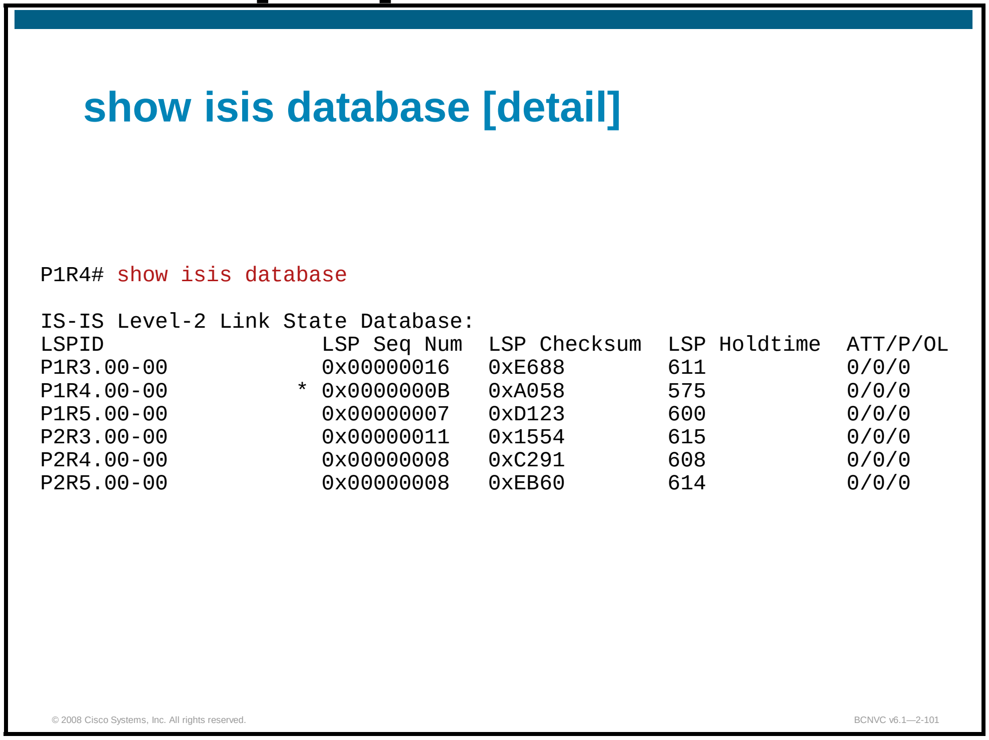
show isis topology



show isis topology

```
P1R3# show isis topology


IS-IS paths to level-2 routers
System Id            Metric     Next-Hop             Interface   SNPA
P1R3                 --
P1R4                 10         P1R4                 Se1/2       *HDLC*
P1R5                 10         P1R5                 Se1/3       *HDLC*
P2R3                 10         P2R3                 Se1/0       *HDLC*
P2R4                 20         P2R3                 Se1/0       *HDLC*
P2R5                 20         P2R3                 Se1/0       *HDLC*
P1R3#
```

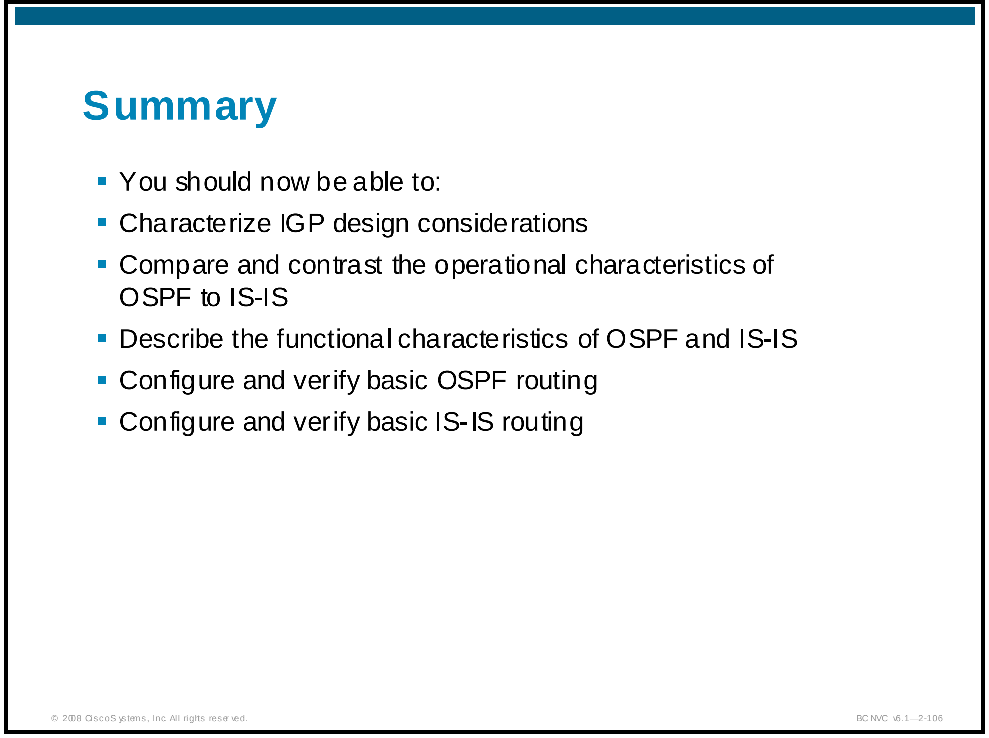
show isis database [detail]



show isis database [detail]

```
P1R4# show isis database

IS-IS Level-2 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
P1R3.00-00            0x00000016   0xE688        611           0/0/0
P1R4.00-00          * 0x0000000B   0xA058        575           0/0/0
P1R5.00-00            0x00000007   0xD123        600           0/0/0
P2R3.00-00            0x00000011   0x1554        615           0/0/0
P2R4.00-00            0x00000008   0xC291        608           0/0/0
P2R5.00-00            0x00000008   0xEB60        614           0/0/0
```

Summary



Summary

- You should now be able to:
- Characterize IGP design considerations
- Compare and contrast the operational characteristics of OSPF to IS-IS
- Describe the functional characteristics of OSPF and IS-IS
- Configure and verify basic OSPF routing
- Configure and verify basic IS-IS routing

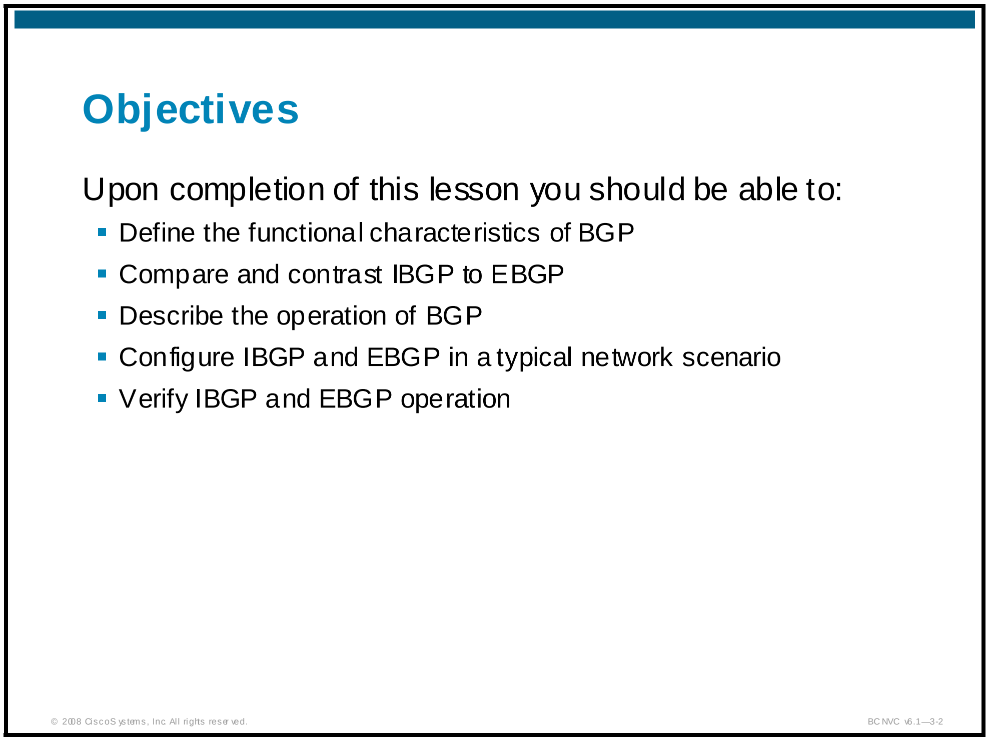
BCNVC v6.1—2-106

The preceding sections gave brief examples of configuring each of the popular IGPs to operate in an ISP backbone. Questions are often asked about which IGP is better or which is easier to configure. Hopefully these examples have shown that there is little difference when it comes to configuration, and the benefits in an ISP backbone usually come down to the scalability of each IGP and the familiarity that the operators have with them.

Lesson 3

Configure BGP

Objectives

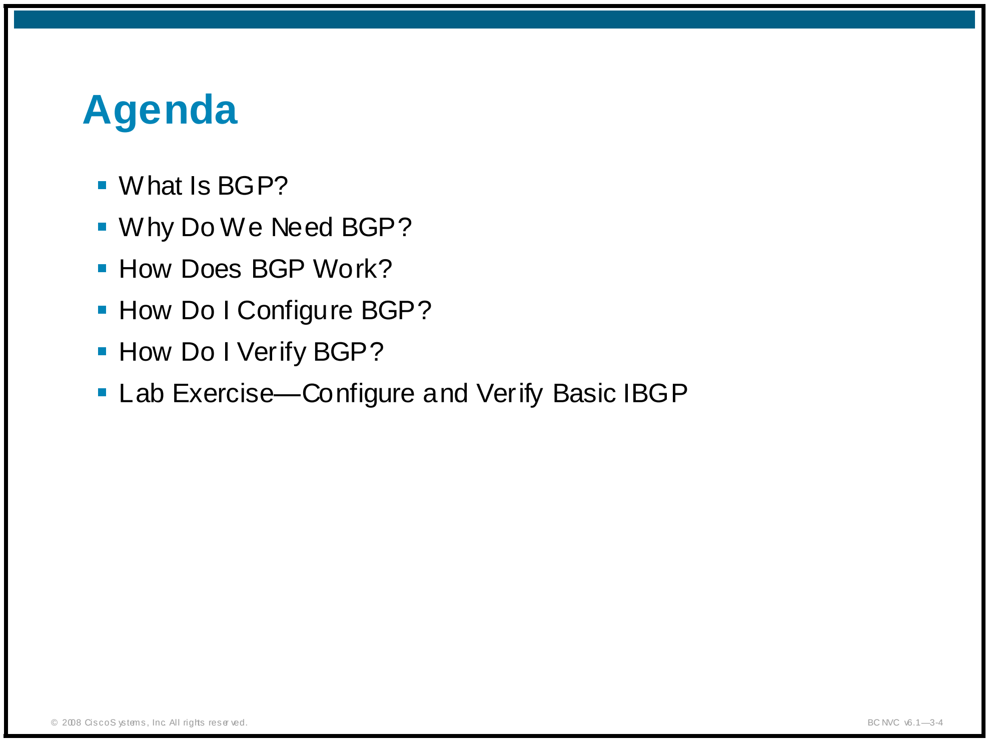


Objectives

Upon completion of this lesson you should be able to:

- Define the functional characteristics of BGP
- Compare and contrast IBGP to EBGP
- Describe the operation of BGP
- Configure IBGP and EBGP in a typical network scenario
- Verify IBGP and EBGP operation

Agenda



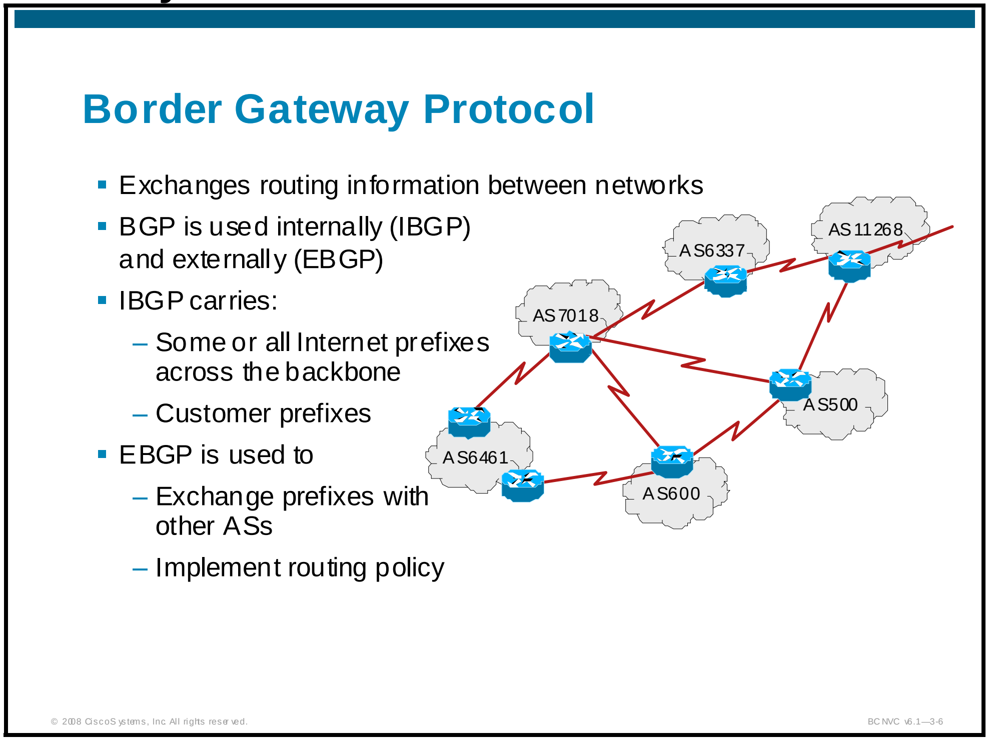
Agenda

- What Is BGP?
- Why Do We Need BGP?
- How Does BGP Work?
- How Do I Configure BGP?
- How Do I Verify BGP?
- Lab Exercise—Configure and Verify Basic IBGP

BCNVC v6.1—3-4

What Is BGP?

Border Gateway Protocol



Border Gateway Protocol

- Exchanges routing information between networks
- BGP is used internally (IBGP) and externally (EBGP)
- IBGP carries:
  - Some or all Internet prefixes across the backbone
  - Customer prefixes
- EBGP is used to
  - Exchange prefixes with other ASs
  - Implement routing policy

BCNVC v6.1—3-6

Border gateway protocol (BGP) replaced the original exterior gateway protocol (EGP), and attempts to address the most serious of EGP problems. Like EGP, BGP is an interdomain routing protocol created for use in the Internet core routers. Unlike EGP, BGP was designed to prevent routing loops in arbitrary topologies and to allow policy-based route selection. BGP was co-authored by a Cisco founder, and Cisco continues to be very involved in BGP development. The latest revision of BGP, BGP4, was designed to handle the scaling problems of the growing Internet.

BGP is an exterior gateway routing protocol that exchanges routing information between networks.

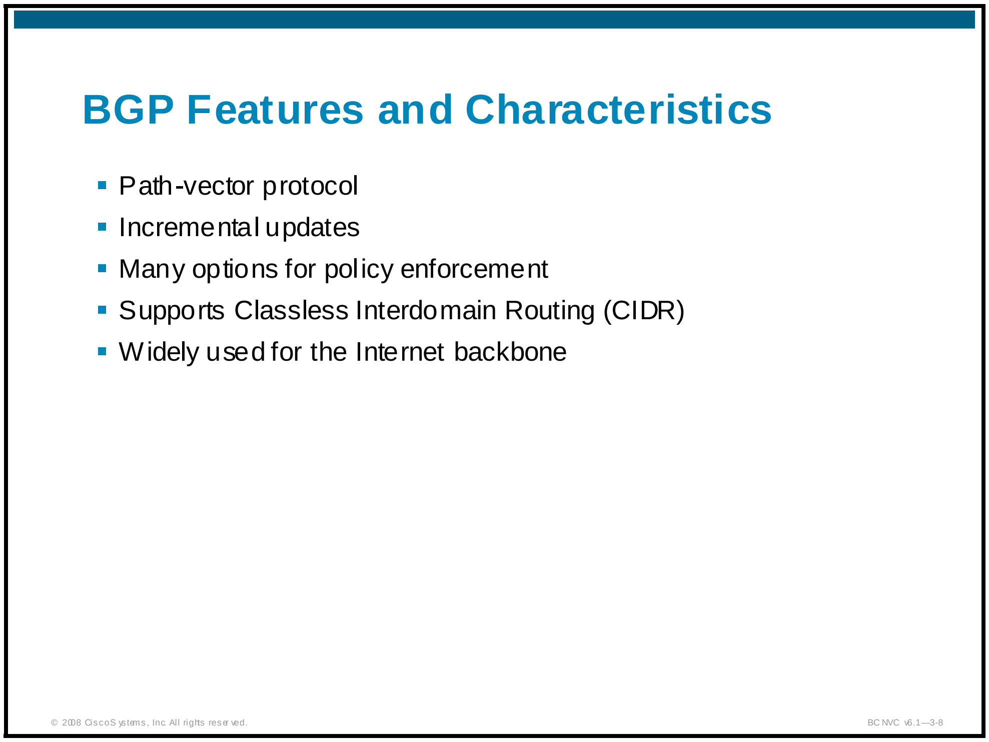
- BGP uses TCP port 179 to transmit routing updates, and is therefore a reliable protocol.
- BGP requires an Internal Routing Protocol (IGP) to establish infrastructure address reachability.

Good practice is to keep IGPs and BGP separate. IGP should NOT be used for carrying Internet prefixes or customer prefixes; that is the role of BGP.

Your network will not scale if you:

- Distribute BGP prefixes into an IGP
- Distribute IGP routes into BGP
- Use an IGP to carry customer prefixes

BGP Features and Characteristics



BGP Features and Characteristics

- Path-vector protocol
- Incremental updates
- Many options for policy enforcement
- Supports Classless Interdomain Routing (CIDR)
- Widely used for the Internet backbone

BCNVC v6.1—3-8

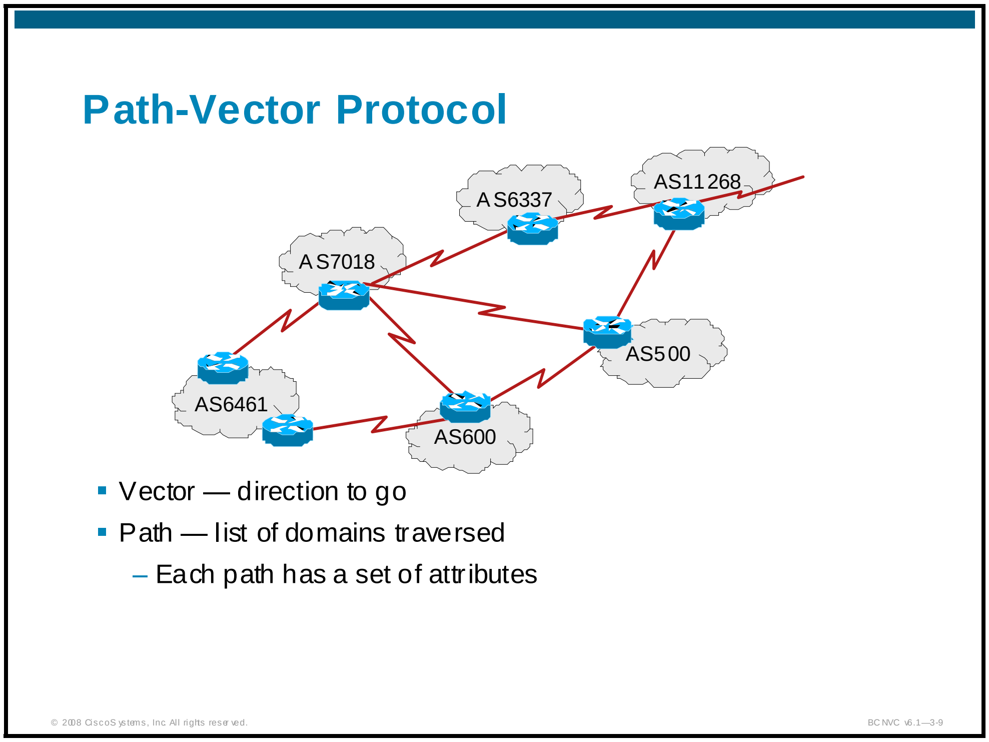
BGP characteristics:

- Path-vector protocol with enhancements
- Acquires neighbors (peers)
- Agrees on autonomous system (AS) numbers and timers
- Keeps track of neighbors (keepalives every 60 seconds)
- Exchanges reachability information
  - Initially routers exchange the entire table
  - Only updates are sent later
  - Updates consist of only network prefix, prefix length, and attributes
  - Reliable updates: it relies on TCP
- Keeps alternative routes
- Insures loop-free routing (between autonomous systems)

Message types

- Open (BGP version, my AS, holdtime, router ID)
- Update (withdrawn routes, advertised route)
- Notification (errors), originator closes the connection
- Keepalives
- Updates

Path-Vector Protocol



Path-Vector Protocol

- Vector — direction to go
- Path — list of domains traversed
  - Each path has a set of attributes

BCNVC v6.1—3-9

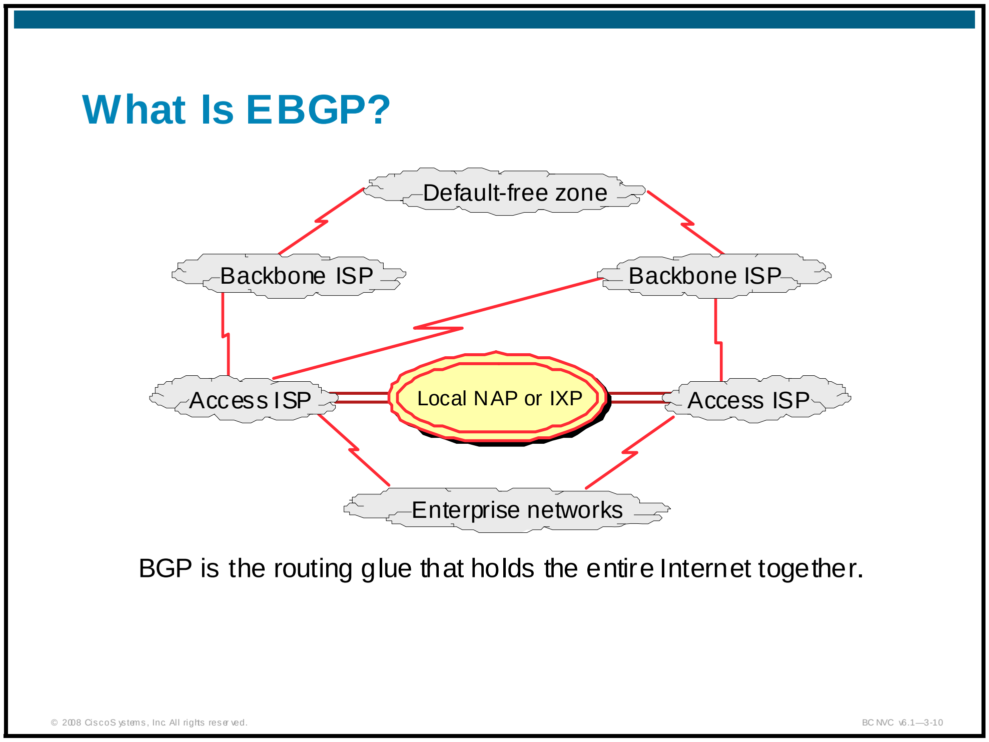
BGP is classed as a path-vector routing protocol (see RFC 1322). The term “path-vector protocol” is intentionally similar to the term “distance-vector protocol,” in which a border router receives from each of its neighbors a vector that contains distances paths to a set of destinations. The path, expressed in terms of the domains (or confederations) traversed so far, is carried in a special path attribute that records the sequence of routing domains through which the reachability information has passed. This path attribute (AS-PATH) is also used to suppress routing loops.

BGP defines a route as a pairing between a destination and the attributes of the path to that destination.

As required of any interdomain protocol, BGP allows policy-based metrics to override distance-based metrics and enables each autonomous system to independently define its routing policies with little or no global coordination. Path attributes make this possible. Some common attributes are:

- AS path (a list of autonomous systems that a route has traversed)
- Next hop
- Origin
- Local preference
- Multi-exit discriminator
- Communities

What Is EBGP?



What Is EBGP?

BGP is the routing glue that holds the entire Internet together.

BGP is used to connect ISPs. It is used for exterior routing, and calculates loop-free paths between autonomous systems. BGP is the routing protocol of choice for connecting to the Internet for service providers with more than one connection to the Internet. It is a way for ISPs to exchange routing information and to define and control the peering relationship they have with each other.

The Internet hierarchy has different types of ISPs, which results in different peering relationships.

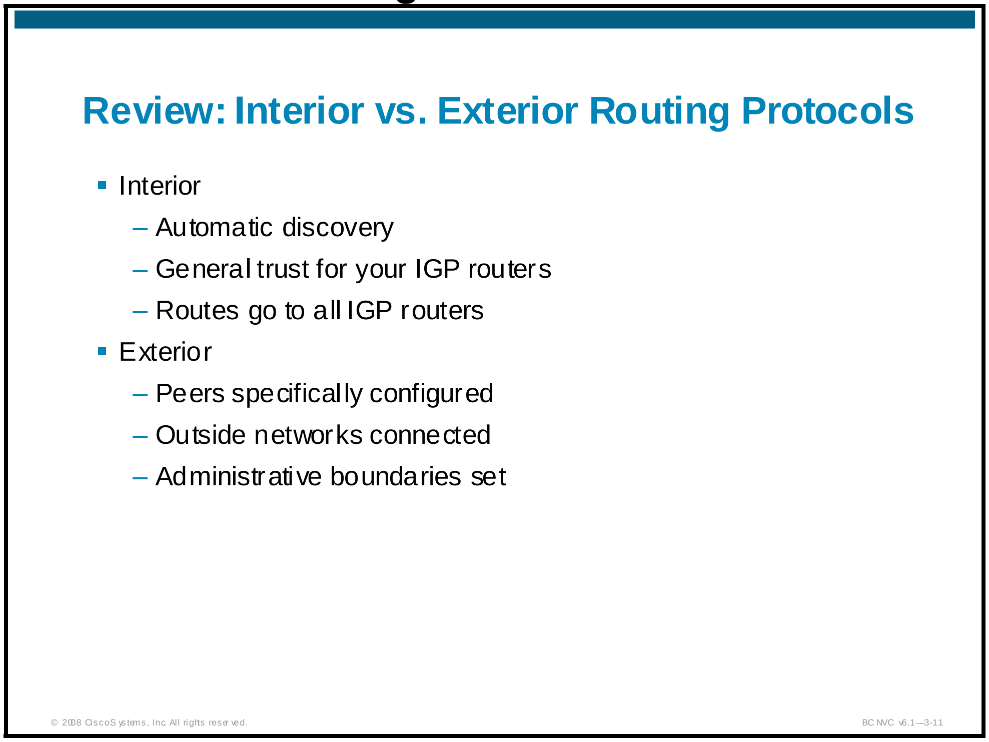
At the top is the default-free zone. These are the top-level ISPs; their routers have explicit routing information about the rest of the Internet, and do not have a default route. If they don’t know how to find it, it doesn’t exist.

Then there are backbone and access ISPs. This hierarchy is sometimes also referred to as Tier 1, Tier 2 and Tier 3 ISPs, but this terminology is very ill-defined and can be politically risky, for example, you may refer to a company as Tier 3 but they consider themselves to be Tier 1.

Enterprise networks may use BGP to interconnect to the ISPs, particularly if they are multihomed; that is, connected to more than one service provider.

Network access providers (NAPs) and Internet exchange points (IXPs) provide interconnectivity between ISPs. They provide a valuable service in the Internet and you should be aware of their existence. Because NAP and IXP interconnectivity is generally switched and not routed, this course doe not address IXPs and NAPs explicitly.

Interior vs. Exterior Routing Protocols



Review: Interior vs. Exterior Routing Protocols

- Interior
  - Automatic discovery
  - General trust for your IGP routers
  - Routes go to all IGP routers
- Exterior
  - Peers specifically configured
  - Outside networks connected
  - Administrative boundaries set

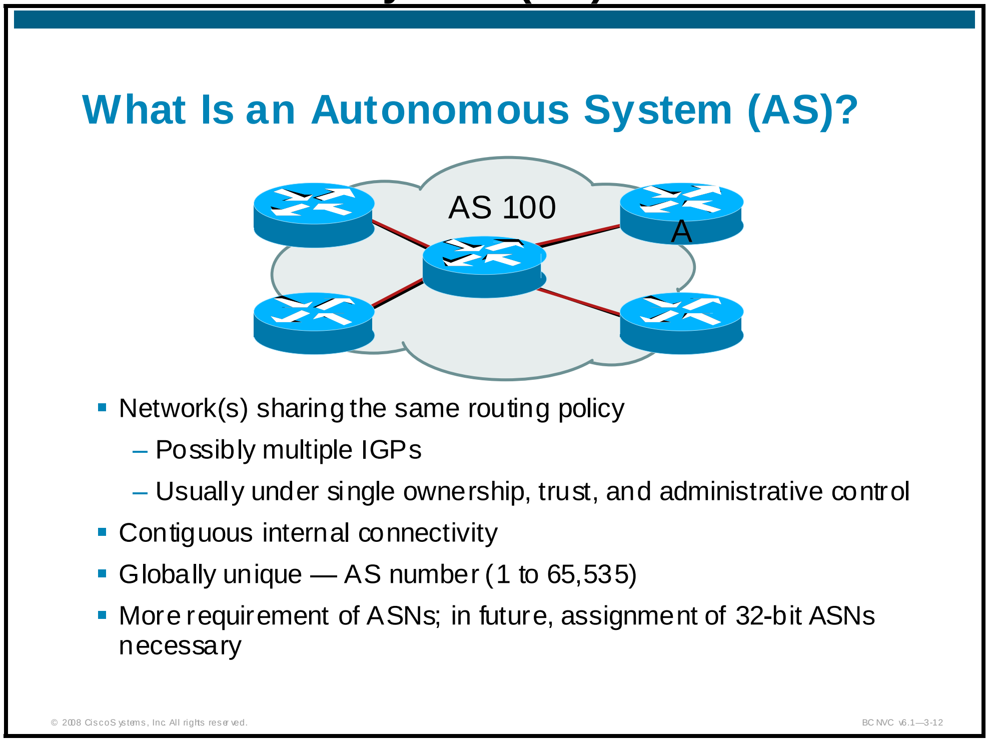
Why do we not just use an IGP for Internet connectivity? Along with the fact that IGPs are not designed to carry this many routes, some other fundamental differences exist between IGPs and BGP.

First, in an IGP, there is an inherent trust in the configuration and performance of IGP neighbors because usually one person, or group of persons, in an organization controls all of the routers. With this inherent trust comes ease-of-use mechanisms, such as automatic neighbor discovery.

In the case of EGPs, and BGP in particular, sessions require manual configuration. Filtering of routes between BGP peers is common practice, as are BGP session passwords and other security and policy enforcement techniques.

Sometimes even a large enterprise network can have multiple controlling bodies—or may simply have more routes than an IGP can easily handle. For the very largest enterprise networks, BGP may be introduced to assist the IGP.

What Is an Autonomous System (AS)?



What Is an Autonomous System (AS)?

- Network(s) sharing the same routing policy
  - Possibly multiple IGPs
  - Usually under single ownership, trust, and administrative control
- Contiguous internal connectivity
- Globally unique — AS number (1 to 65,535)
- More requirement of ASNs; in future, assignment of 32-bit ASNs necessary

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—3-12

An autonomous system is roughly equivalent to an ISP. But what does this really mean? An AS is a routing domain under a single administrative control insofar as the rest of the Internet is concerned. This is not to say an AS must run a single IGP. It may run many. The AS may even use BGP to help scale the IGP within a network. However, eventually, when it connects to the Internet, it does so through a single, globally unique identifier called an "AS number."

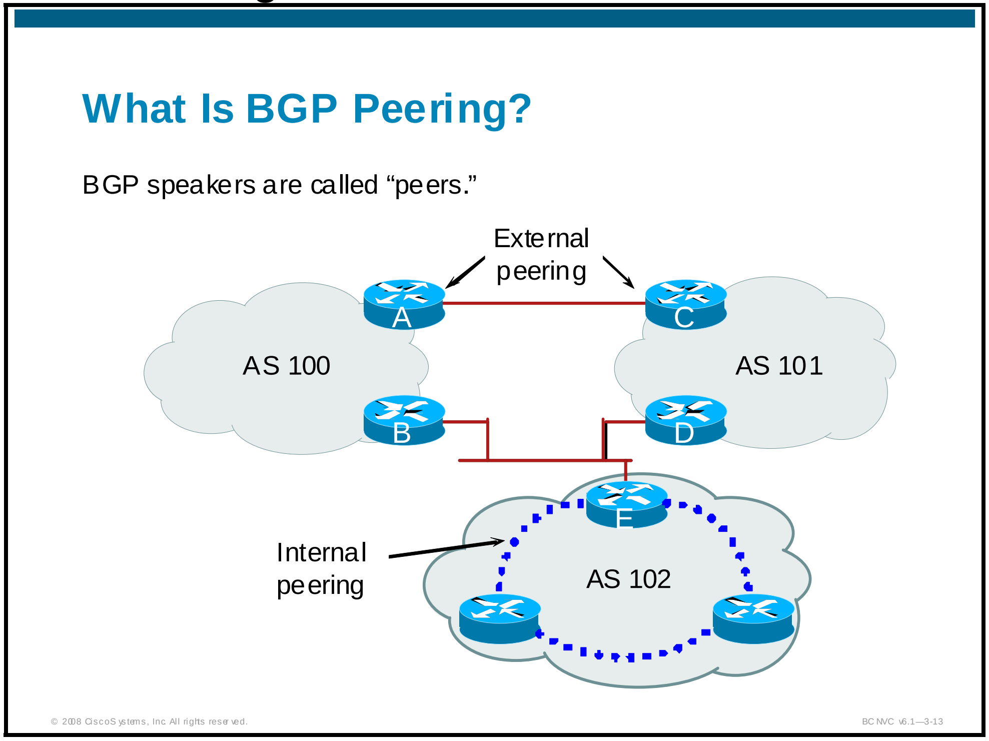
Autonomous System Numbers

To uniquely identify an autonomous system, a number, ranging from 1 to 65,535, is assigned from a centralized authority so there are no duplicate numbers. These numbers are referred to as AS numbers. The American Registry for Internet Numbers (ARIN) is responsible for tracking and assigning AS numbers. ARIN charges a fee to organizations wishing to obtain an AS number to cover the administrative costs associated with managing AS number registrations and assignments.

A range of AS numbers (64,512 to 65,535) are reserved for private use. These are roughly equivalent to the private IP address space, such as network 10.0.0.0. These AS numbers are never used on the public Internet. They can be used in private networks and translated before being advertised to the Internet.

As the number of available official ASNs is decreasing, in *RFC 4893 BGP Support for Four-octet AS Number Space*, the usage and transition to 32-bit ASN is defined. The discussion of migration strategies and use of the 32-bit ASN is outside the scope of this class. For the sake of simplicity we will use 16-bit ASN throughout the remainder of the class.

What Is BGP Peering?



What Is BGP Peering?

BGP speakers are called “peers.”

BCNVC v6.1—3-13

Generally speaking, the terms "neighbor" and "peer" are interchangeable. The word "peer" originates from the concept of peer-to-peer networking. A neighbor or peer is a router that another router has established a BGP session with. Neighbors and peers exchange BGP updates according to their administrator-established routing policies.

Peering occurs when a router exchanges routes with another BGP device. There are two types of peering sessions:

Internal peers (IBGP) - An internal peer is a BGP-speaking neighbor with the same AS number.

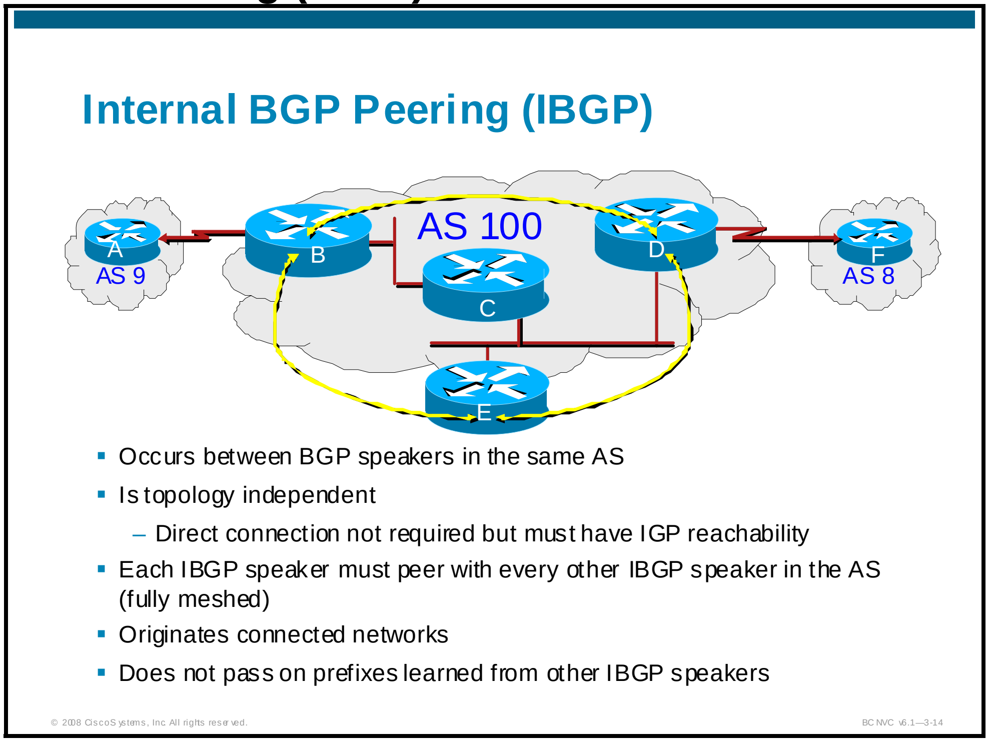
External peers (EBGP) - External peers have different AS numbers. An external peer passes on all the best routes it knows or has learned from any other peer to all other directly connected external peers. This means that EBGP is considered a "gossipy" protocol; routers speaking EBGP pass on everything they know to their neighbors unless you install a gag (a route filter).

Private peering occurs when two autonomous systems agree to connect to one another. Private peering relationships may be based on a financial arrangement or may be for an exchange of services. The decision to peer may be based on several factors:

- Between equivalent sizes of service providers (for example, Tier 2 to Tier 2)
- Shared-cost private interconnection, equal traffic flows
- No-cost peering

A router that peers with another AS is called an Autonomous System Border Router (ASBR).

Internal BGP Peering (IBGP)



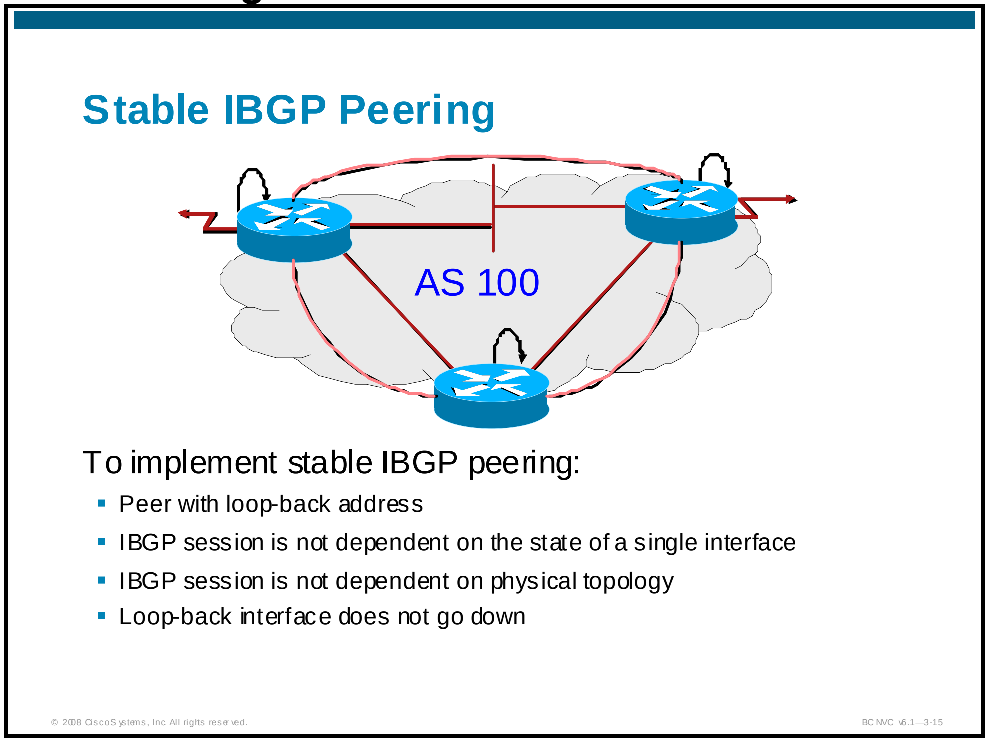
Internal BGP Peering (IBGP)

AS 100
A
AS 9
B
C
D
E
F
AS 8

- Occurs between BGP speakers in the same AS
- Is topology independent
  - Direct connection not required but must have IGP reachability
- Each IBGP speaker must peer with every other IBGP speaker in the AS (fully meshed)
- Originates connected networks
- Does not pass on prefixes learned from other IBGP speakers

BCNVC v6.1—3-14

Stable IBGP Peering



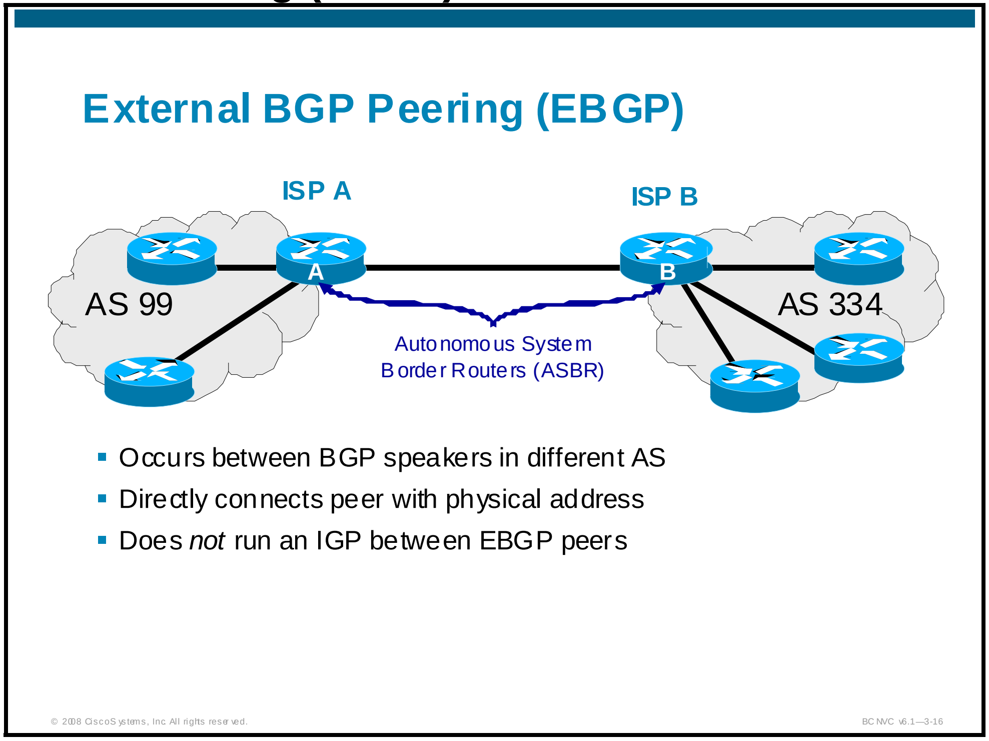
Stable IBGP Peering

To implement stable IBGP peering:

- Peer with loop-back address
- IBGP session is not dependent on the state of a single interface
- IBGP session is not dependent on physical topology
- Loop-back interface does not go down

BCNVC v6.1—3-15

External BGP Peering (EBGP)

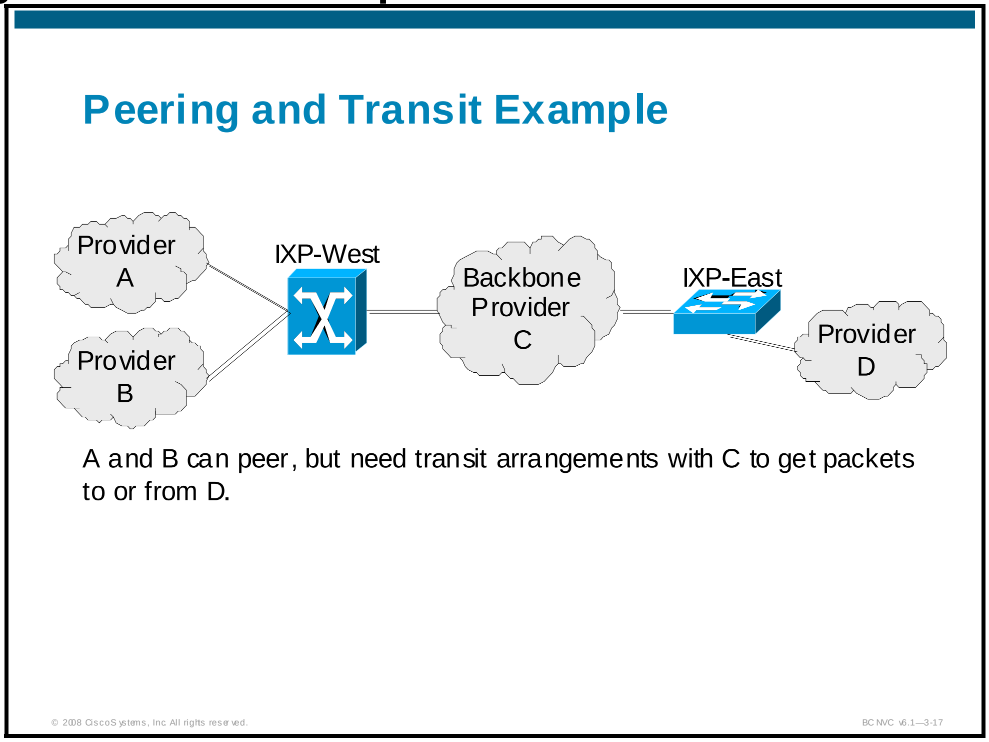


External BGP Peering (EBGP)

- Occurs between BGP speakers in different AS
- Directly connects peer with physical address
- Does *not* run an IGP between EBGP peers

BC NVC v6.1—3-16

Peering and Transit Example



Peering and Transit Example

A and B can peer, but need transit arrangements with C to get packets to or from D.

BCNVC v6.1—3-17

When two autonomous systems peer, they can provide access to all their other peers or to none of their peers. This is often established as part of each autonomous system’s routing policy.

Peering may be done across exchange points (called public peering) if it is convenient, of mutual benefit, or technically feasible. Exchange points are usually used if there is fee-based peering, or if there are unequal traffic flows.

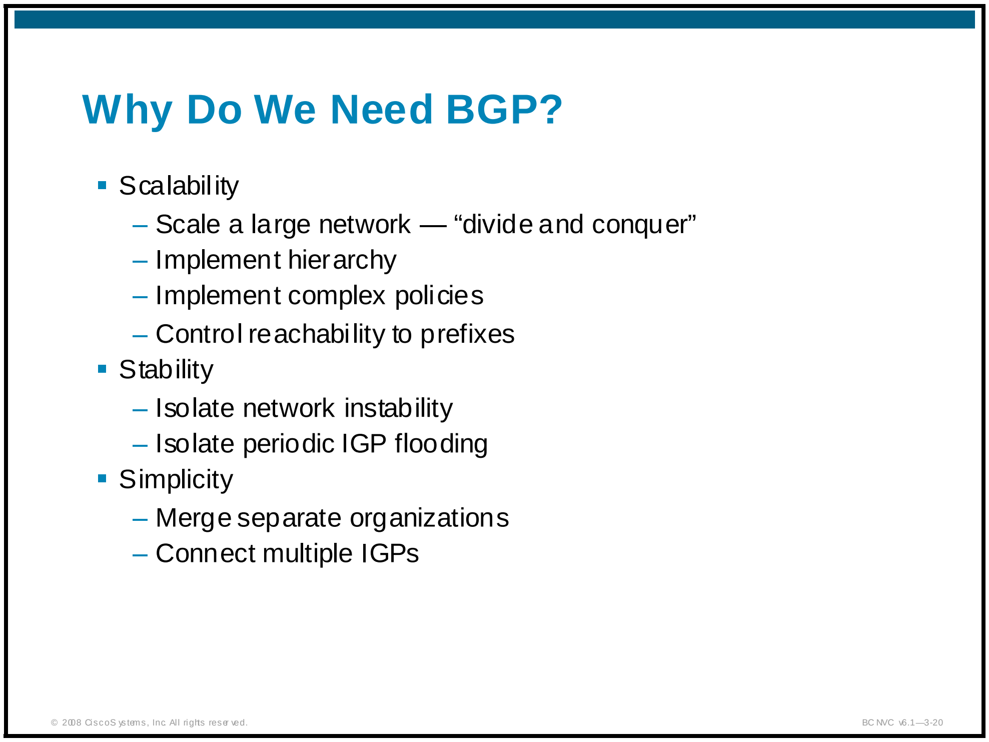
Transit

Transit is when traffic that originates outside your autonomous system, and is destined for a network outside your AS, is permitted to route through your AS. Transit peering describes an arrangement where an EBGP peer is permitted to communicate with your other EBGP peers. The most common use of this occurs when an ISP allows their customers using BGP to access all their other customers using BGP.

Non-transit

Non-transit is when you provide one EBGP peer access to your network, but not to any other EBGP peers you might have. This is useful for when a customer is connected to two ISP networks, and wishes to have each ISP customers use their own connections to reach him. The two ISPs use a non-transit policy, while providing a transit policy for all their customers. All customers can reach each other, but the ISPs cannot use the other ISP to reach another ISP.

Why Do We Need BGP?



Why Do We Need BGP?

- Scalability
  - Scale a large network — “divide and conquer”
  - Implement hierarchy
  - Implement complex policies
  - Control reachability to prefixes
- Stability
  - Isolate network instability
  - Isolate periodic IGP flooding
- Simplicity
  - Merge separate organizations
  - Connect multiple IGPs

BCMVC v6.1—3-20

Three factors are critical to good networking: scalability, stability, and simplicity. BGP provides the tools to facilitate large network deployments.

Scalability

Scalability derives from good planning -- even if your network is small today, it could grow very large. Good planning early on, including a strategy to scale your network and routing policy using features such as route reflectors, peer groups, and community-based policy, is critical.

Stability

Stability is important because BGP routing activity is visible to the entire Internet. At best, an unstable network may give you intermittent connectivity and poor performance on your routers while CPU resources are wasted on unnecessary route computations. However, it can be worse -- some ISPs penalize you for instability by taking your routes off the Internet using a process called BGP damping.

Simplicity

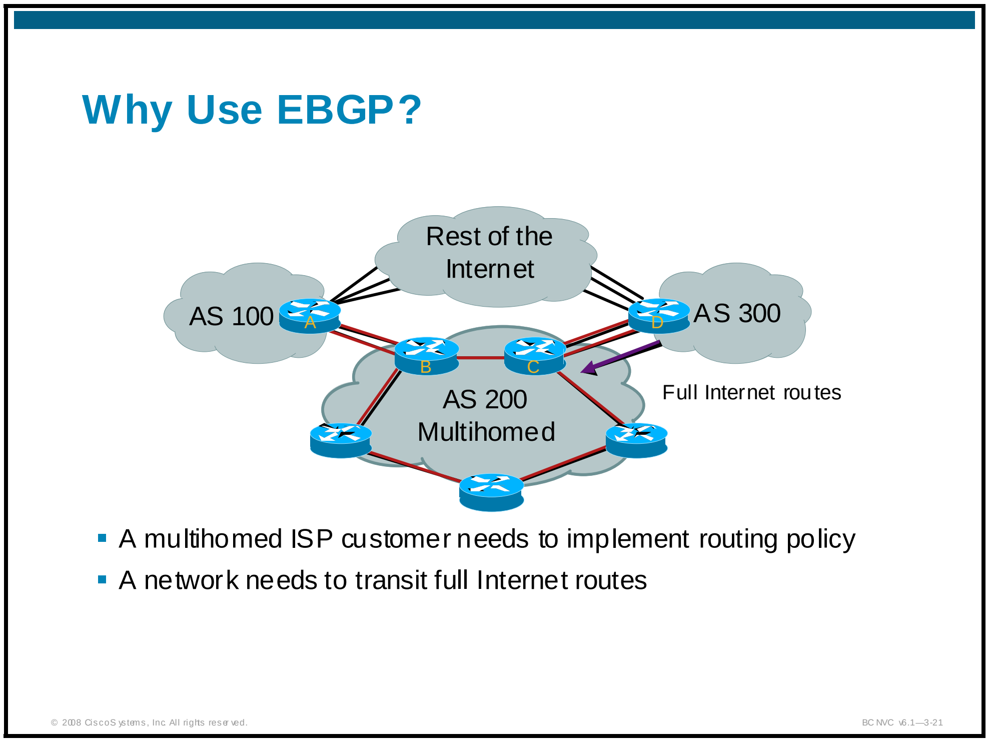
Networks consisting of more than a few routers usually have more than one administrator. Think of your colleagues when doing BGP gymnastics that use every knob and feature you can find. Make sure your configurations and policies are easy to follow and, most importantly, consistent. A short two-page document describing how policy is implemented in your network can be very beneficial.

BGP allows you to scale a network. It provides a strict boundary to IGP behavior, such as inherent-trust, flooding, and possible routing storms due to bugs or misconfiguration.

BGP can allow you to isolate areas of your enterprise network from problems in other areas. It can allow you to join two recently merged enterprises.

Finally, you can simply take one large, overworked IGP, and shrink it into several smaller IGP deployments, and connect them all via BGP.

Why Use EBGP?



Why Use EBGP?

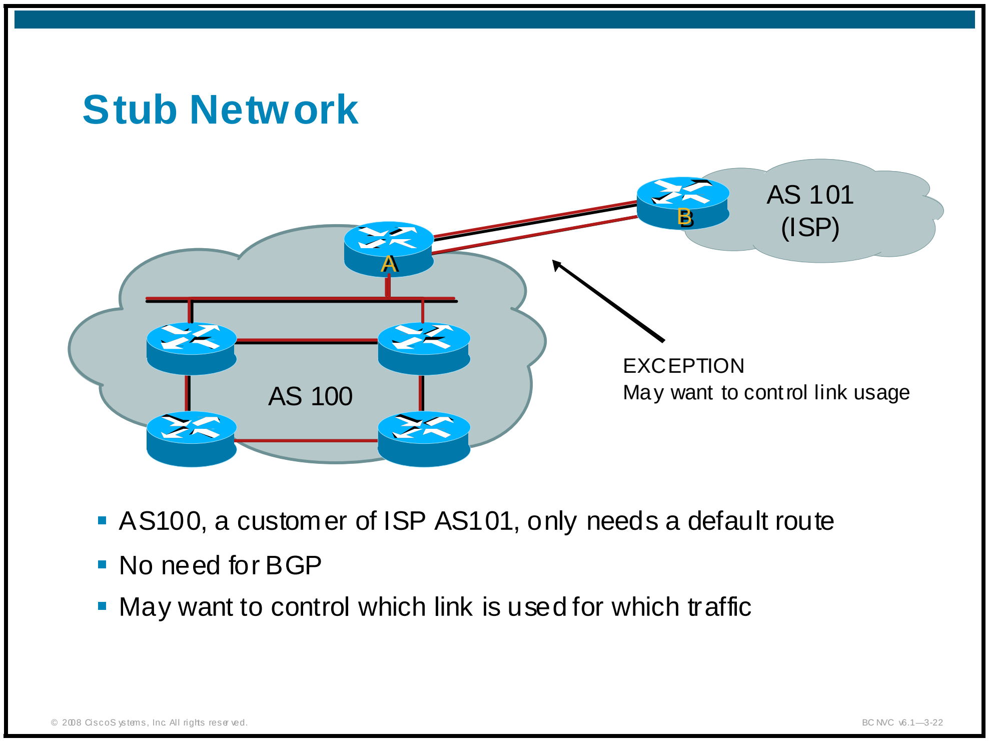
- A multihomed ISP customer needs to implement routing policy
- A network needs to transit full Internet routes

BCNVC v6.1—3-21

For companies connecting to the Internet, BGP can be used to achieve multihoming, which is connecting to more than one ISP for performance or reliability reasons. BGP provides the tools that give you rough control over which ISP is used to reach which destinations.

ISPs may need to pass a full set of Internet routes to their customers -- especially those who are multihomed. If you want to feed someone a full set of Internet routes, BGP is your only choice.

Stub Network



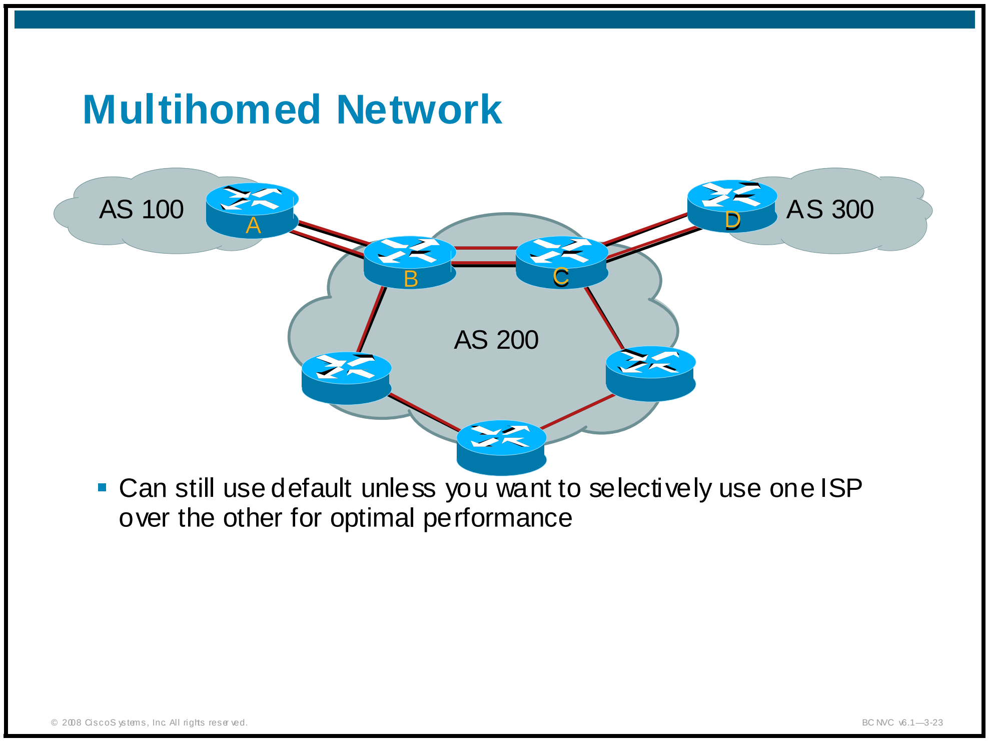
Stub Network

- AS100, a customer of ISP AS101, only needs a default route
- No need for BGP
- May want to control which link is used for which traffic

Consider first an enterprise that only connects to the Internet via a single ISP, a “stub” network. In this case there is no need for BGP. The ISP advertises the stub network and the policy is confined within ISP policy. The enterprise only needs a default route to the border.

Even if this network wishes to connect via one or more links to the same ISP, it is not necessary to use BGP. The only time it might make sense to use BGP in this scenario is if AS100 wants to control the link it uses to reach a particular Internet destination.

Multihomed Network



Multihomed Network

- Can still use default unless you want to selectively use one ISP over the other for optimal performance

Many situations are possible:

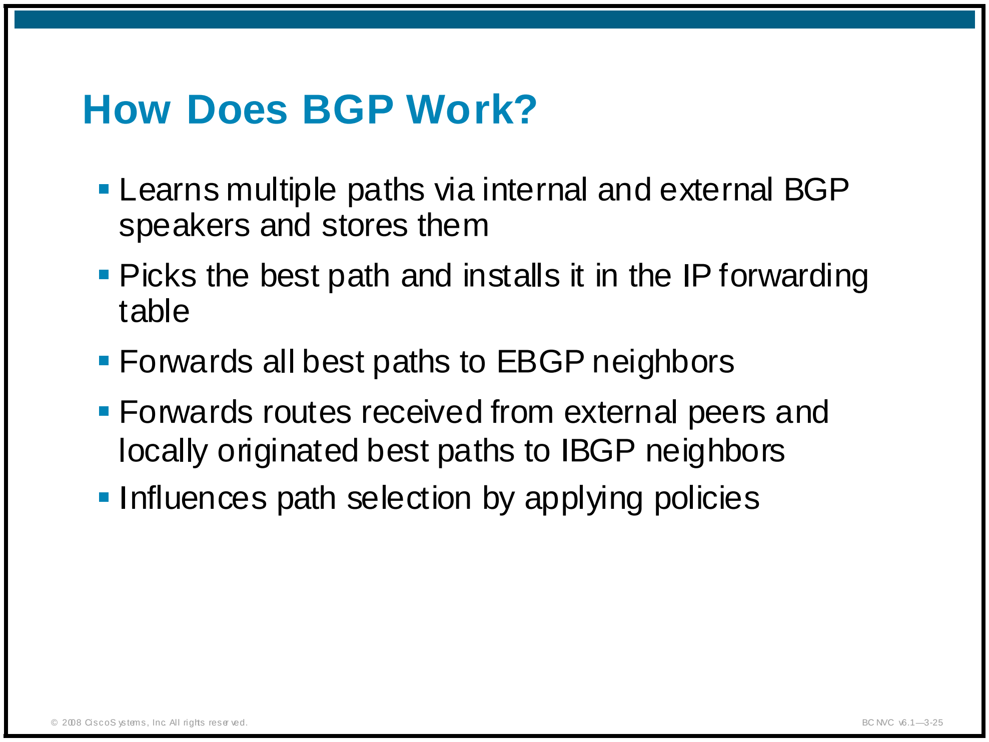
- Multiple links to the same ISP—without BGP
- Secondary is only for backup—without BGP
- Loadshare exists between primary and secondary— without BGP
- Selectively use different ISPs—need BGP

With multihoming, there are several different scenarios to consider. The simplest is having multiple links to the same ISP. One link might be a backup link, or you may want to loadshare between two links.

Further, there may be another ISP. The customer may not want to rely on a single ISP, either to the keep their ISPs competitive, or to remove a single point of failure or loadshare.

None of the above scenarios requires the use of BGP. It is only when you want to selectively use either of the ISPs (or links to a single ISP) that you need to consider using BGP.

How Does BGP Work?



How Does BGP Work?

- Learns multiple paths via internal and external BGP speakers and stores them
- Picks the best path and installs it in the IP forwarding table
- Forwards all best paths to EBGP neighbors
- Forwards routes received from external peers and locally originated best paths to IBGP neighbors
- Influences path selection by applying policies

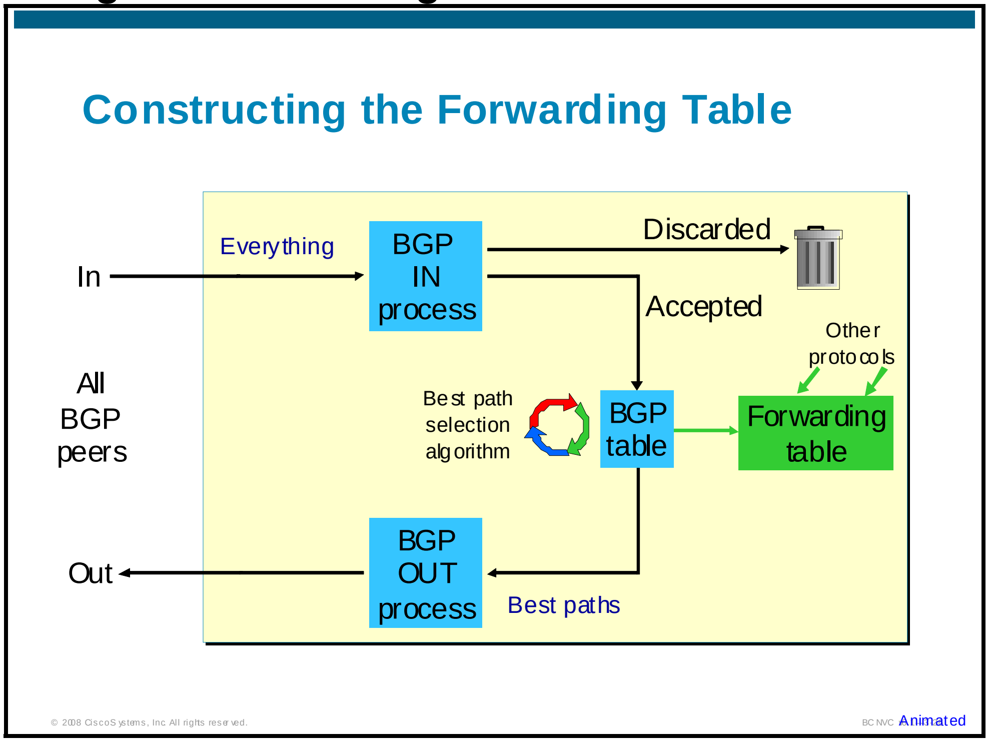
The basic operation of BGP is to receive routes from neighbors, choose the best one, install it in the forwarding table, and send the best route on to EBGP neighbors.

Why only EBGP neighbors? To ensure a loop-free internal BGP environment. BGP assumes the only routes learned from a neighbor are those it has itself learned from EBGP. EBGP is protected from loops by an attribute called AS path.

To ensure any BGP router has complete routing information, it is necessary for all BGP routers in an AS to have a IBGP peering sessions with each other. With anything but a very small number of routers, this presents a serious scaling issue. There are technologies called BGP confederations and route reflectors that address this problem.

Routing policy controls the way traffic flows into and out of an AS by impacting, either through filters or BGP attribute modification, the best path decisions.

Constructing the Forwarding Table



Constructing the Forwarding Table

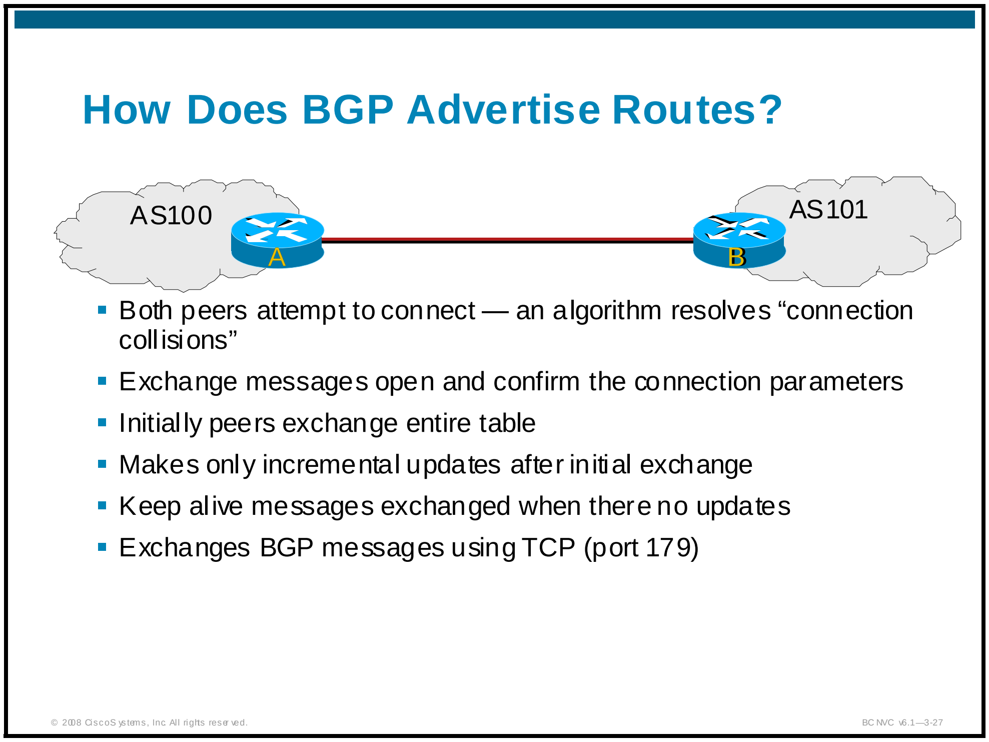
BC NVC Animated

1. BGP Intelligent Network (IN) process
2. BGP learns multiple paths via internal and external BGP speakers.
3. Paths are sent to the BGP in process where the routes are tested for a variety of criteria. Those that meet the prerequisites (access lists, policy, and other tests) are sent to the BGP table.
4. When there are multiple routes to the same destination, the BGP path selection algorithm chooses the best path and flags it. Single routes are always the best path.
5. Best paths are installed in the forwarding table and sent to the “BGP out” process.
6. At the BGP out process the routes are tested for a variety of criteria. Prefixes that pass those tests are announced to peers.

When dealing with BGP, there are two distinct routing tables that you should be aware of, the forwarding table (show ip route), and the BGP table (show ip bgp). The routing table consists of the chosen best route to reach a network for any routing protocol, be that RIP, Open Shortest Path First (OSPF), Interior Gateway Routing Protocol (IGRP), static, connected, or BGP-derived networks. The BGP table consists of routes received only via BGP, and may contain several duplicate ways to reach the same network (referred to as prefix). As updates are received via BGP, the best route to a prefix may change, and this change is reflected in the main routing table. You can analyze the different routing tables by using the appropriate show command.

Because these tables are logically separate, it follows that just because a route is in the forwarding table doesn't mean that it is in the BGP table, and hence is not be advertised via BGP. For routes to appear in the BGP table, they must be redistributed into BGP. However, this is considered quite dangerous, as it does not afford a great deal of control. It is much safer to redistribute static routes, as they must be manually entered into the router in order to be advertised.

How Does BGP Advertise Routes?



How Does BGP Advertise Routes?

- Both peers attempt to connect — an algorithm resolves "connection collisions"
- Exchange messages open and confirm the connection parameters
- Initially peers exchange entire table
- Makes only incremental updates after initial exchange
- Keep alive messages exchanged when there no updates
- Exchanges BGP messages using TCP (port 179)

BCNVC v6.1—3-27

A peering session only includes two routers. Both routers can attempt to connect to the other on TCP port 179 (make sure this is not blocked by a firewall). If both connections occur at the same time, there is a well-defined mechanism to resolve the collision.

Route advertisements are not sent out on regular intervals as in other protocols. A full table exchange is sent out when BGP is first started, and then only incremental updates are sent when changes occur in the topology.

BGP uses an UPDATE message to advertise routes. It is important to remember that BGP routers listen for routes. A BGP router cannot forward a packet if it has not heard a route. Also, if a route is not being advertised, it is also not possible to forward packets.

EBGP peers advertise all known EBGP routes to all other EBGP peers. IBGP peers advertise routes they originate, plus EBGP routes, to other IBGP peers. A IBGP router never advertises other IBGP peer routes to any other IBGP peer.

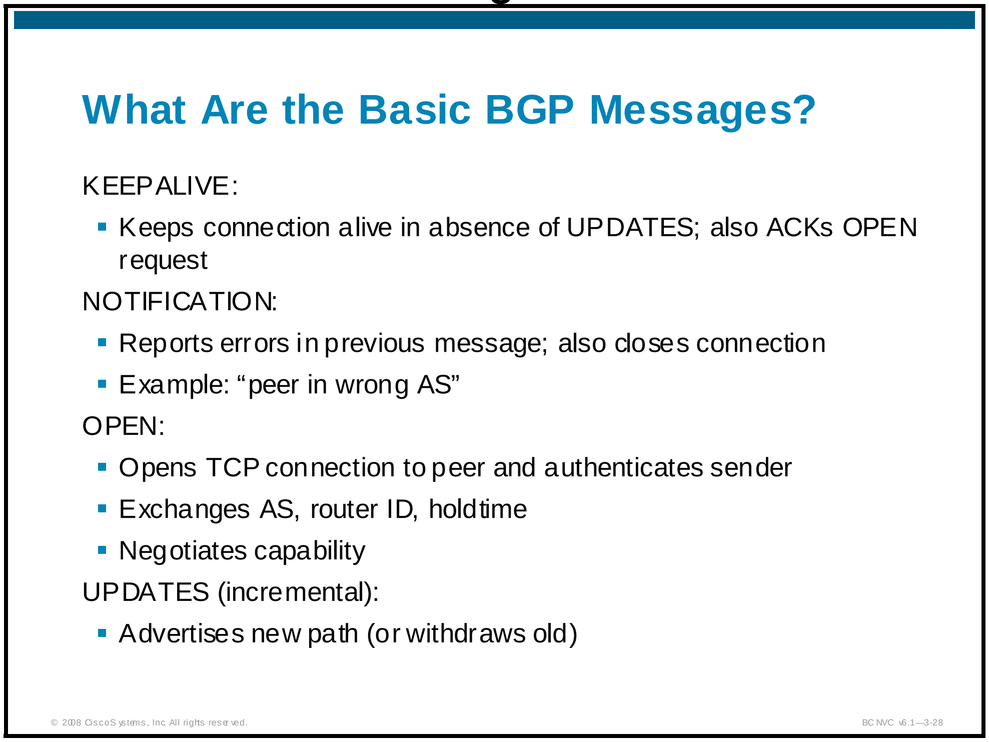
Once BGP sends a route to a peer, it assumes the peer keeps it unless:

- A replacement route is sent—implicit withdraw of old route
- The route is withdrawn—explicit withdraw
- The BGP session goes down (keepalive failure)

When a BGP route becomes unreachable, BGP sends UPDATE messages that contain withdrawal information, requesting that other BGP routers remove those routes from their tables.

When there are no updates sent for more than 60 seconds (default timer), a keepalive packet is sent to the neighbor, in order to keep the session established. If neither a keepalive or update packet is received within the holdtime (peers use the lower of the holdtimes contained in the two open messages), the session is closed, and a holdtime expired notification is sent.

What Are the Basic BGP Messages



What Are the Basic BGP Messages?

KEEPALIVE:

- Keeps connection alive in absence of UPDATES; also ACKs OPEN request

NOTIFICATION:

- Reports errors in previous message; also closes connection
- Example: “peer in wrong AS”

OPEN:

- Opens TCP connection to peer and authenticates sender
- Exchanges AS, router ID, holdtime
- Negotiates capability

UPDATES (incremental):

- Advertises new path (or withdraws old)

BCNVC v6.1—3-28

BGP information is contained in the payload of a TCP protocol data unit. BGP uses the following message types:

- KEEPALIVE
- NOTIFICATION
- OPEN
- UPDATE

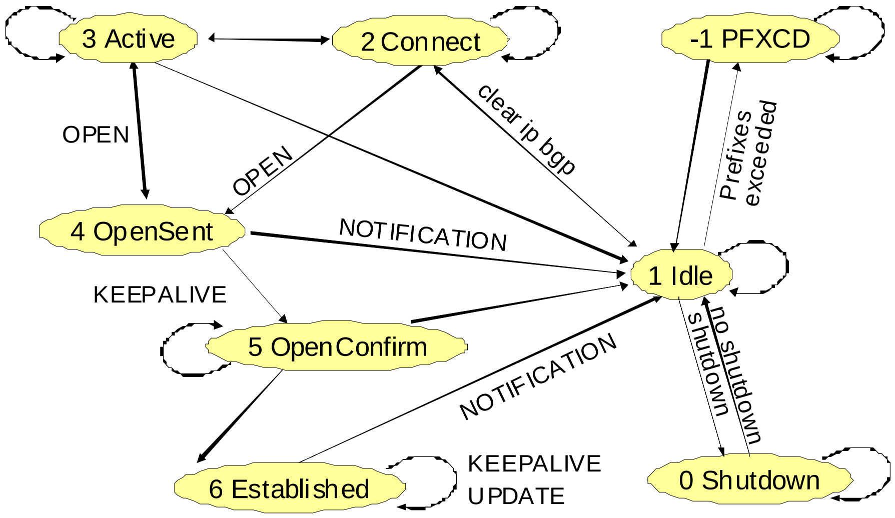
Plain BGP has four message types (there are more that can occur if agreed by peers through capability negotiation; for example; route refresh request).

The open message is used to exchange AS, ID, and timer (holdtime) information. The AS and IP address in an incoming message must match those configured on the local router, in order for a session to be established. There is a BGP finite state machine, which takes sessions from the idle through to the established state.

A notification is sent when there is a problem with the session. Examples include a corrupted update, incorrect AS, or invalid attribute. The session is always closed upon receiving a notification. Sending and receiving notifications is a normal part of the capability negotiation process. This is the process by which peers can negotiate optional BGP features, such as route refresh, or outbound route filtering (ORF) (prefix list exchange). It is done this way to maintain backward compatibility with routers that do not support capabilities negotiation.

BGP States

BGP States



BCNVC v6.1—3-29

Eight possible states exist in BGP’s finite state machine, as follows:

- **-1. prefix-exceeded**: If the neighbor has been configured to limit the prefixes received and if the received prefix count exceeds maximum, the session is held in this state until a **clear ip bgp** command is applied to this neighbor. This feature safeguards against the large number of routes that may be received peers as a result of configuration errors such as applying an incorrect route-forwarding policy or not applying any policy at all. The classic example is a small ISP that inadvertently transits all Internet routes between two large ISPs.

- **0. Shutdown**: Sessions may be administratively shut down using the **neighbor** {ip-address |peer-group-name} **shutdown BGP router configuration** command. In this state, all incoming connection requests are refused, and no connections are attempted.

- **1. Idle**: After configuration via the **neighbor** {ip-address | peer-group-name} **remote- as number BGP** subcommand, sessions begin in the Idle state. At this point, the router periodically (based on an exponentially growing connect-retry-timer) initiates a TCP connection on port 179 to its neighbor and moves to the Connect state. While in the Idle state, the router also listens on port 179 for incoming TCP connections. If one arrives from the listed neighbor IP address, the session also moves to the Connect state. If clear ip bgp {ip-address | peer-group-name} is executed during this session, the connect-retry-timer is reset.

- **2. Connect**: At this point, the session waits for the TCP connection to succeed. If it does, the local router sends an OPEN message to its peer and moves to the OpenSent state. If the connection fails, the session moves to the Active state. If the connect-retry-timer expires, it is reset and a new transport connection is initiated. In response to clear ip bgp {ip-address | peer-group-name} associated with this session, it returns to the Idle state. If a valid (from the correct remote-IP on the correct port) incoming TCP connection attempt is received, this session moves to the Connect state. An OPEN message is sent to the remote neighbor, and the session moves to the OpenSent state.

- **3. Active**: The router generally reaches this state because a transport connection has failed. It stays in the Active state until the connect-retry-timer expires, at which point it initiates another connection and moves to the Connect state. In this state, the router also continues to listen for an incoming BGP session. If a valid connection is made, the router sends an OPEN message and transitions to the OpenSent state.
- **4. OpenSent**: The router has sent an OPEN message and waits for an OPEN message from its peer. Once received, the OPEN message is checked for the following: acceptable remote-as (as per neighbor {ip-address | peer-group-name} remote-as number configuration); acceptable version number (2, 3, or 4; default is 4 unless the bgp {ip-address | peer-group- name} version value is configured). Any errors in the OPEN message result in a notification being sent and a change to the Idle state. If no errors arise, a KEEPALIVE message is sent and the router sets the holdtime as the minimum of its locally configured holdtime and the holdtime in the open message received from its peer. If the holdtime expires, the router sends a NOTIFICATION (holdtime expired), and the session reverts to Idle state. If the underlying transport connection is closed, the session reverts to Active state.
- **5. Open Confirm**: The router waits for a KEEPALIVE, which signals that no notifications are expected as a result of the Open message and the session can move to the Established state. If the router does not receive a KEEPALIVE within the negotiated holdtime, it sends a NOTIFICATION (holdtimer expired) and reverts to Idle state. Similarly, if the router receives a NOTIFICATION (usually as a result of a problem with the Open message), the state reverts to Idle.
- **6. Established**: Once in this state, the routers generally exchange UPDATE messages. If there is no UPDATE within the holdtime, a KEEPALIVE is sent unless the negotiated holdtime is zero, in which case no keepalives are necessary. If an UPDATE or KEEPALIVE contains errors, a NOTIFICATION is sent and the state shifts to Idle.

Shutting Down a BGP neighbor

Sometimes intermittent physical-layer problems can cause packet loss and routing instability at ISP gateways. Sometimes the best course of action is to shut down the BGP neighbor. This leaves the link up (to allow debugging with ping), but eliminates any production traffic from going over the link. The command to deactivate a BGP link is:

```
neighbor {ip-address | peer-group-name} shutdown
```

Limiting Prefixes

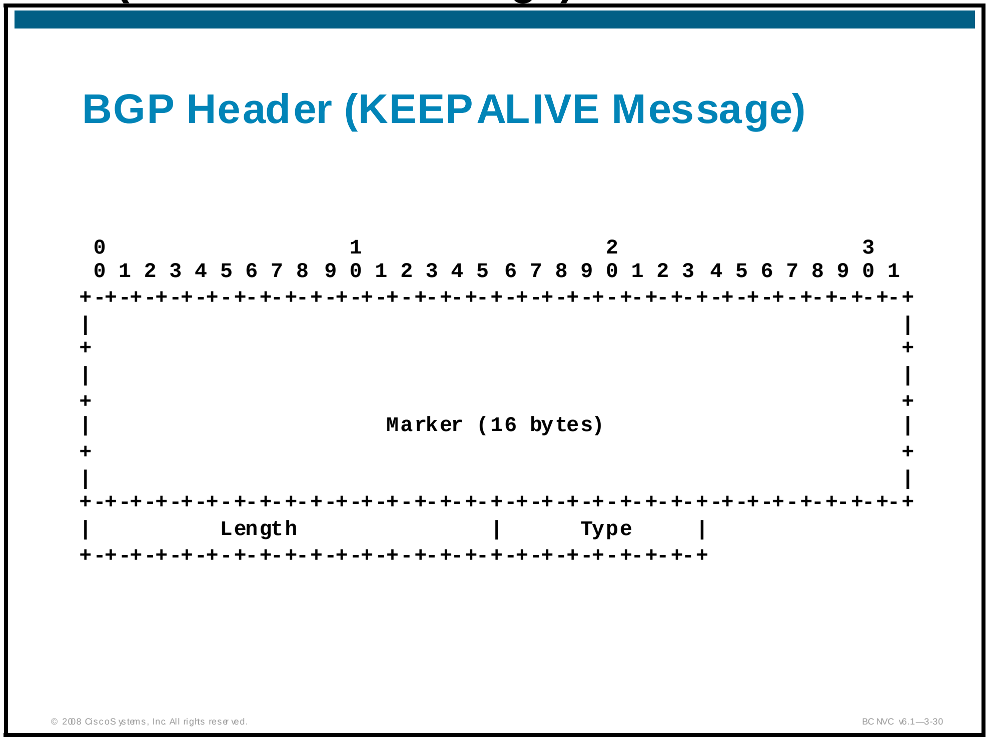
To control how many prefixes can be received from a neighbor, and the action to take when the limit is reached, use the following command:

```
neighbor {ip-address | peer-group-name} maximum-prefix maximum
[threshold] [warning only]
```

ip-address | peer-group-name	IP address or Name of a BGP peer group name of the neighbor
maximum	Maximum number of prefixes allowed from this neighbor.
threshold	(Optional) Integer specifying at what percentage of maximum the router starts to generate a warning message. The range is from 1 to 100; the default is 75%.
warning-only	(Optional) Allows the router to generate log message when the maximum is exceeded, instead of terminating the peering.

Caution Once a neighbor is placed in the prefix exceeded state, it is held in this state until a **clear ip bgp** command is applied to the neighbor.

BGP Header (KEEPALIVE Message)



BGP Header (KEEPALIVE Message)

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                                                               +
|                           Marker (16 bytes)                   |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Length               |      Type     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—3-30

All BGP messages have a 19-byte header that identifies packet type and packet length.

The header includes a 16-byte marker for synchronization, which allows BGP to figure out where a new packet starts. The BGP RFC also describes a way to use the marker for authentication, however, Cisco’s implementation does not use this mechanism. Instead, Cisco uses MD5 at the TCP level. Therefore, on Cisco routers, the marker is just filled with 1’s.

Type: 1 = OPEN, 2 = UPDATE, 3 = NOTIFICATION, 4 = KEEPALIVE

KEEPALIVE Message

Since there is no timer for route updates (updates happen dynamically on an incremental basis), keepalive messages are exchanged to be certain that a BGP session stays up and functional. This message consists of only the header.

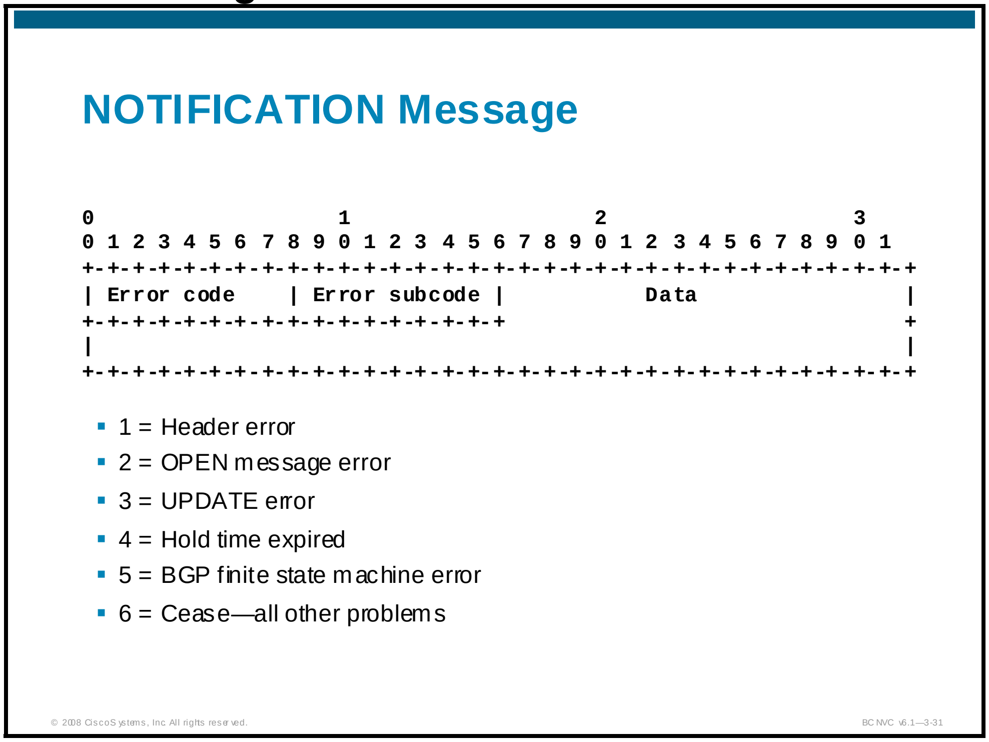
The keepalive interval is 60 seconds. The timer counts down to zero and then sends out another keepalive.

The hold-down timer indicates how long a router waits between hearing messages from its neighbor. The hold-down timer defaults to 180 seconds, but can be reconfigured. The timer starts at zero and counts its way up to the hold-down timer value. If either a keepalive or update message is not received in that time, then the router declares the peering session dead, places all routes learned from that peer into a ‘damped’ state, and attempts to reset the session.

Algorithm:

- The defaults are 180 seconds for holdtime and 60 seconds for the keepalive timer. If the user has configured a value, then that overrides the defaults.
- The holdtime is the minimum of what the router receives in the OPEN message and the default (or configured) value. If this results in 0, then the keepalive timer is set to 0 (don’t send/expire).
- If the holdtime is not 0, then it is set to at least 3 seconds. In this case, the keepalive timer is at least one second or the minimum between the default (or configured) value, holdtime/3 and (holdtime-1).

Notification Message



NOTIFICATION Message

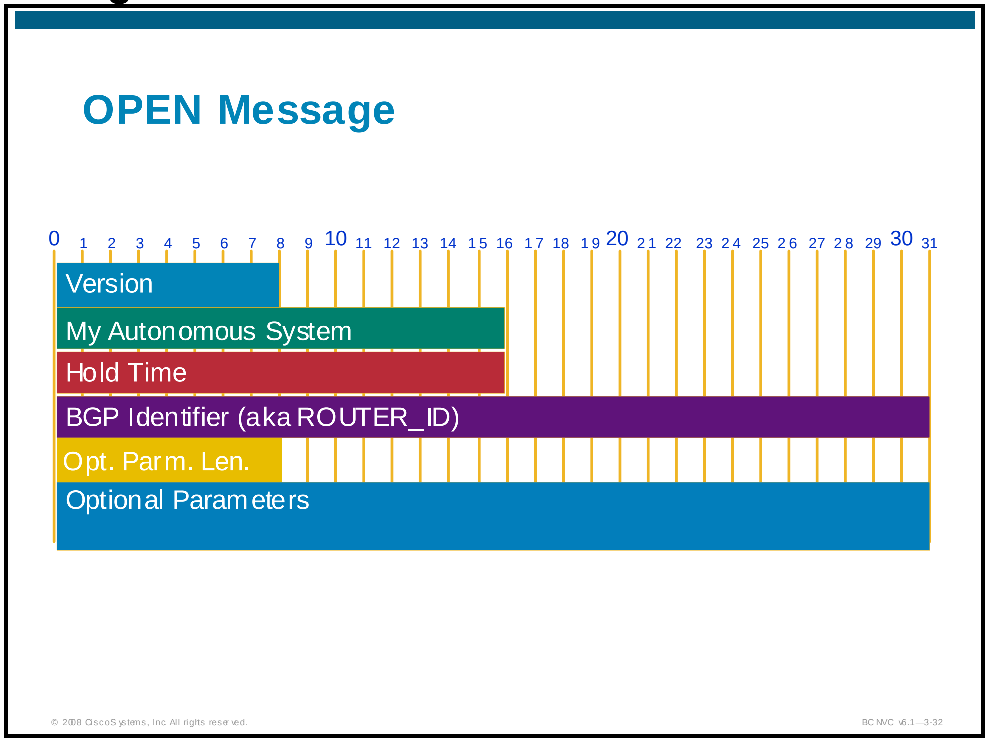
```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Error code    | Error subcode |           Data                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- 1 = Header error
- 2 = OPEN message error
- 3 = UPDATE error
- 4 = Hold time expired
- 5 = BGP finite state machine error
- 6 = Cease—all other problems

BCNVC v6.1—3-31

NOTIFICATION messages are sent when there is an error with a session. When a NOTIFICATION is sent or received, the associated BGP session is always closed down. It is usually restarted again about a minute later, except in certain cases such as prefix-exceeded notifications.

Open Message


OPEN Message

Version

My Autonomous System

Hold Time

BGP Identifier (aka ROUTER_ID)

Opt. Parm. Len.

Optional Parameters

BCNVC v6.1—3-32

BGP learns and exchanges path information regarding a route to a given destination network by keeping lists of AS numbers, and associating them with destination networks. This is why AS numbers should be unique. BGP uses the AS-path (a list of all the autonomous systems that the route passes through to reach the destination) to prevent routing loops. A BGP speaker does not accept a route already containing its own ASN in the AS path. This is how routing loops are prevented.

The BGP router ID also is used as the last step of the BGP path-selection process. Another reason to ensure that the loopback interface is configured and has an IP address is that if there is no loopback, the router ID is the highest IP address configured on the box at the time the BGP process was started. (This is potentially problematic because, if the router is gaining more interfaces or more activated connections with IP addresses assigned to them, the router ID can potentially change if the router is rebooted or the BGP process is restarted.)

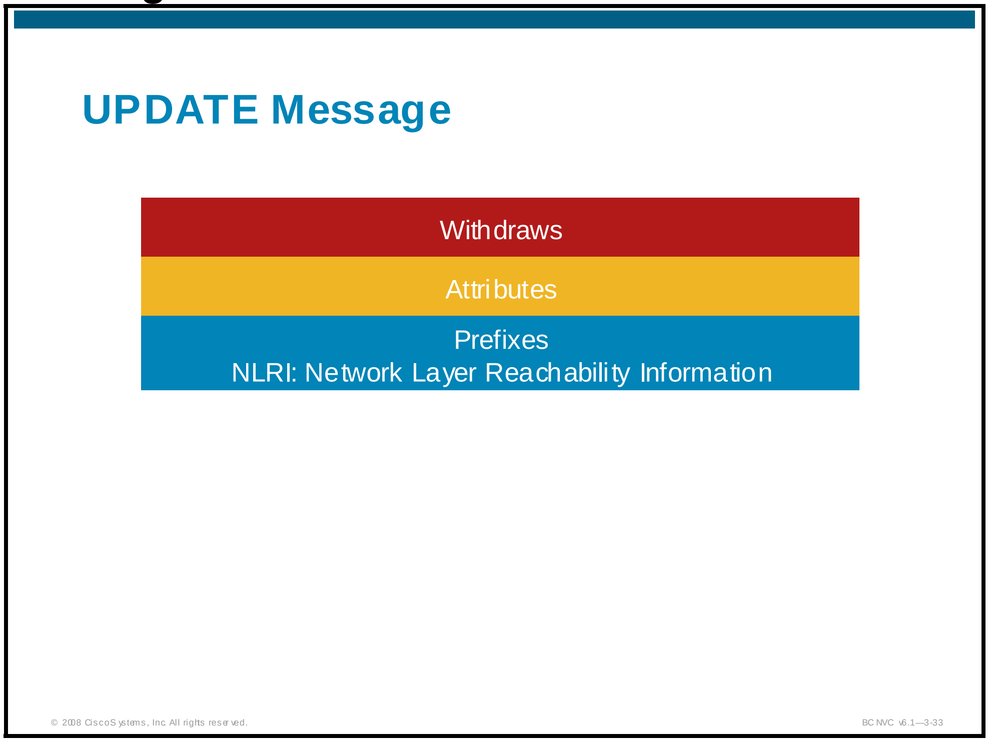
The BGP identifier is the highest IP address in the router (loopbacks are preferred over physical interfaces). If OSPF is redistributing into BGP, then the OSPF router ID is used instead. The ROUTER_ID is used as a tie-breaker in the BGP best-path algorithm. You can explicitly set the router ID using the **bgp router-id** command.

Optional parameters are used to negotiate MD5 TCP encryption for the session, or to negotiate additional capabilities such as MBGP, VPN address families, outbound route filtering (ORF), and route refresh.

Router ID best practices:

- Use the manually configured address
- Use the router ID of the OSPF process into which BGP is redistributing routes
- Use the loopback interface with the highest IP address
- Use the physical interface with the highest IP address

Update Message



UPDATE Message

Withdraws

Attributes

Prefixes
NLRI: Network Layer Reachability Information

A BGP update message begins with a list of routes to explicitly withdraw. These are usually routes that were advertised earlier during the session.

Following the list of withdraws are the attributes associated with the new prefixes being advertised in this update. The attributes include AS path, multi-exit discriminator (MED), community, and many others.

Note that BGP 4, which is the version in use today, is classless. Routes include both a network and a mask. In BGP RFC nomenclature, these routes are called Network-Layer Reachability Information (NLRI).

If, in the list of prefixes, there appears a prefix that was sent earlier, the earlier prefix is assumed to be implicitly withdrawn, and replaced by the new advertisement.

BGP Network Layer Reachability Information

The UPDATE message contains NLRI, which includes prefixes, masks, paths, and attributes.

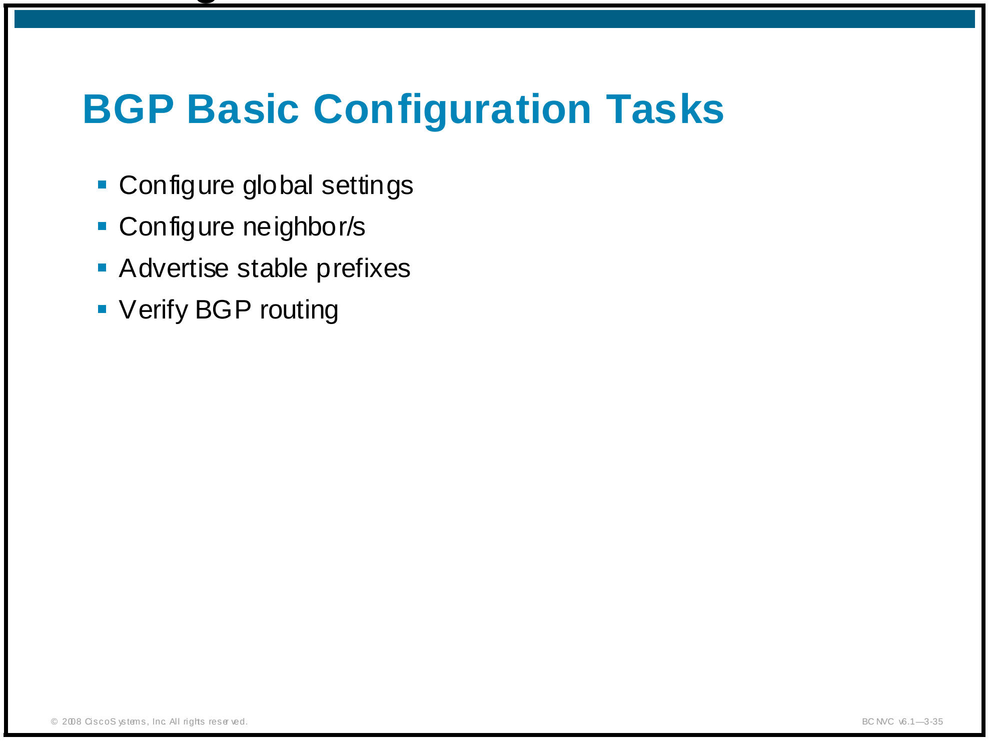
The NLRI is exchanged between BGP routers using UPDATE messages. An NLRI is composed of a LENGTH and a PREFIX. The length is a mask in classless interdomain routing (CIDR) notation (/25) specifying the number of network bits, and the prefix is the network address for that subnet.

The NLRI is unique to BGP version 4 and allows BGP to carry supernetting information and perform aggregation.

Multiple NLRIs are included in an UPDATE message.

Configuring BGP

BGP Basic Configuration Tasks



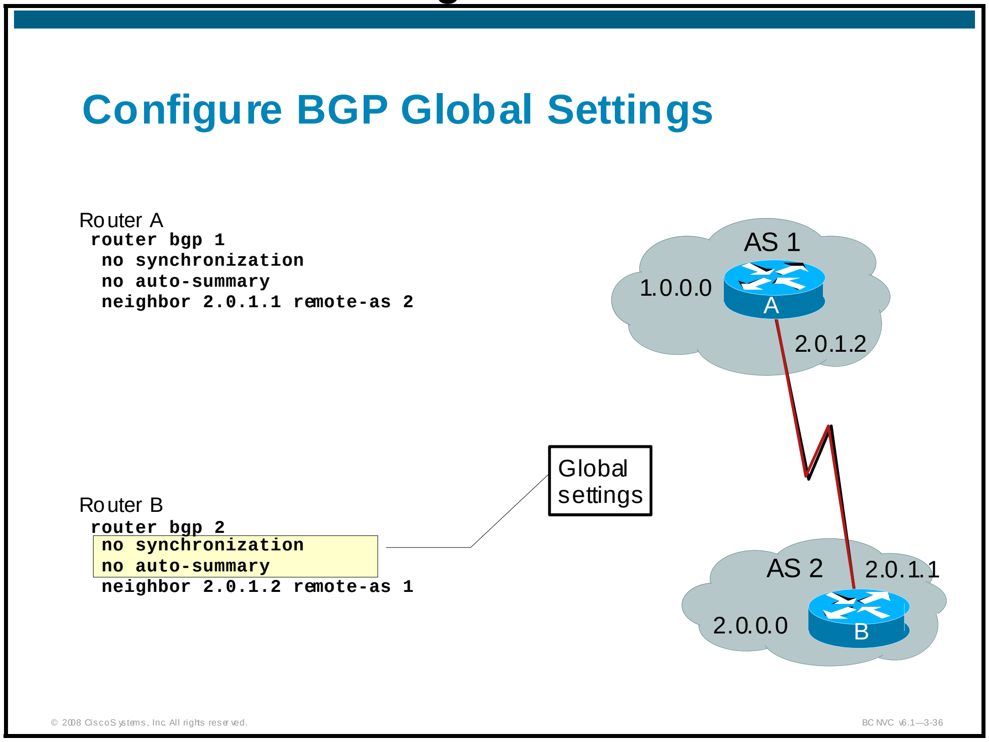
BGP Basic Configuration Tasks

- Configure global settings
- Configure neighbor/s
- Advertise stable prefixes
- Verify BGP routing

This section introduces the basics of BGP configuration. It demonstrates how to configure basic BGP global settings, configure IBGP and EBGP neighbors, advertise routes, and verify BGP routing.

The major differences between IBGP peers and EBGP peers are covered in the BGP Routing Policy module.

Configure BGP Global Settings



Configure BGP Global Settings

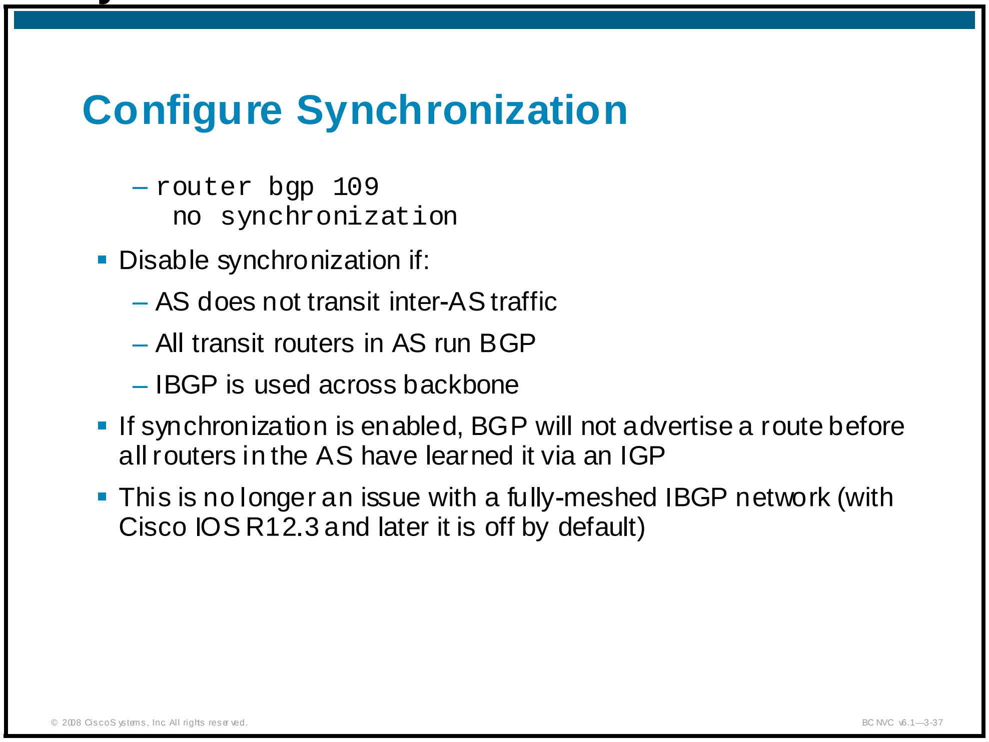
Router A

```
router bgp 1
 no synchronization
 no auto-summary
 neighbor 2.0.1.1 remote-as 2
```

Router B

```
router bgp 2
 no synchronization
 no auto-summary
 neighbor 2.0.1.2 remote-as 1
```

Configure Synchronization

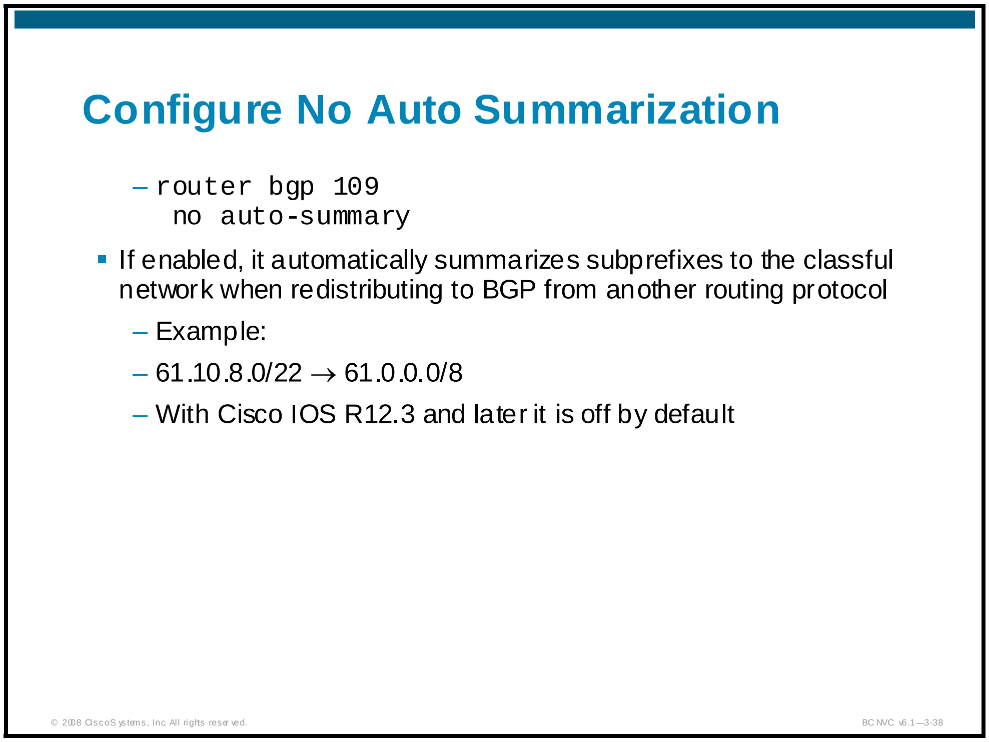


Configure Synchronization

```
– router bgp 109
    no synchronization
```

- Disable synchronization if:
  - AS does not transit inter-AS traffic
  - All transit routers in AS run BGP
  - IBGP is used across backbone
- If synchronization is enabled, BGP will not advertise a route before all routers in the AS have learned it via an IGP
- This is no longer an issue with a fully-meshed IBGP network (with Cisco IOS R12.3 and later it is off by default)

Configure No Auto Summarization



Configure No Auto Summarization

-
  ```
  router bgp 109
   no auto-summary
  ```
- If enabled, it automatically summarizes subprefixes to the classful network when redistributing to BGP from another routing protocol
  - Example:
  - 61.10.8.0/22 → 61.0.0.0/8
  - With Cisco IOS R12.3 and later it is off by default

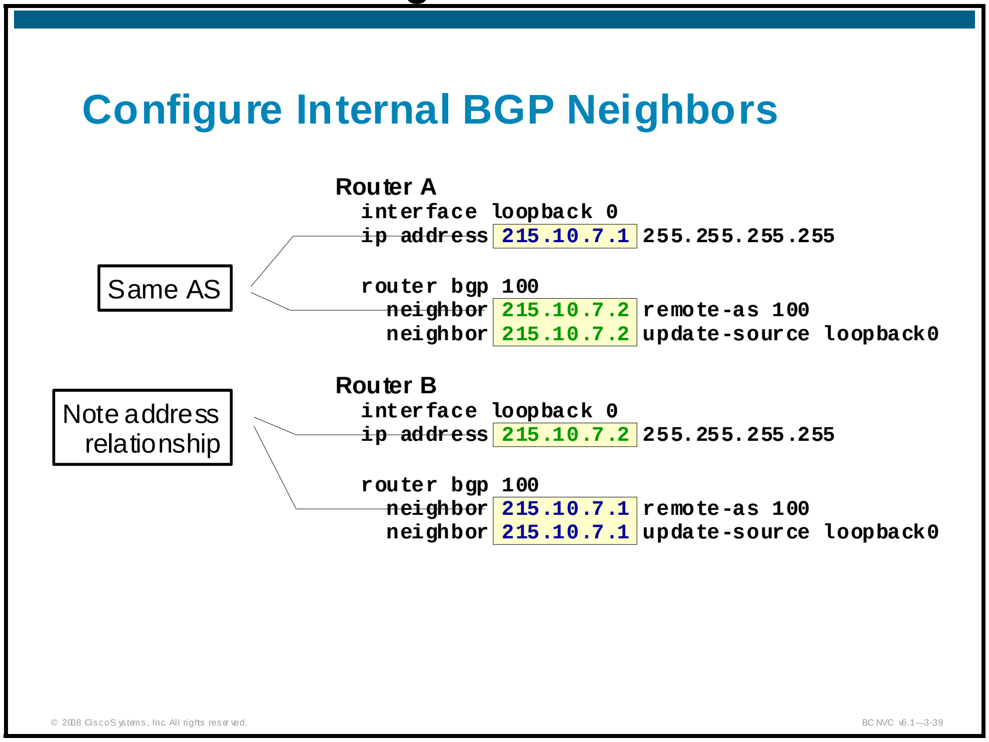
BC NVC v6.1—3-38

In the past, Internet authorities allocated IP address space in Class A, B, or C sized chunks. Therefore, even though network operators broke these allocations into subnets within their networks, it nearly always made sense to automatically aggregate these subnets into the corresponding Class A, B, or C networks before sending them out via EBGP.

This scenario is no longer true. The various address allocation authorities now allocate address-space out of what were traditionally Class A networks. Portions of a Class A may be allocated to many providers. If you auto-aggregate this address space and send it out to the Internet you will become very unpopular among the ISP community, because this would severely damage global Internet routing.

Eventually, **no auto-summary** will become the default in Cisco IOS software. However, for now, please manually disable auto-summary in all BGP configurations.

Configure Internal BGP Neighbors



Configure Internal BGP Neighbors

Same AS

Note address relationship

Router A

```
  interface loopback 0
  ip address 215.10.7.1 255.255.255.255

  router bgp 100
    neighbor 215.10.7.2 remote-as 100
    neighbor 215.10.7.2 update-source loopback0
```

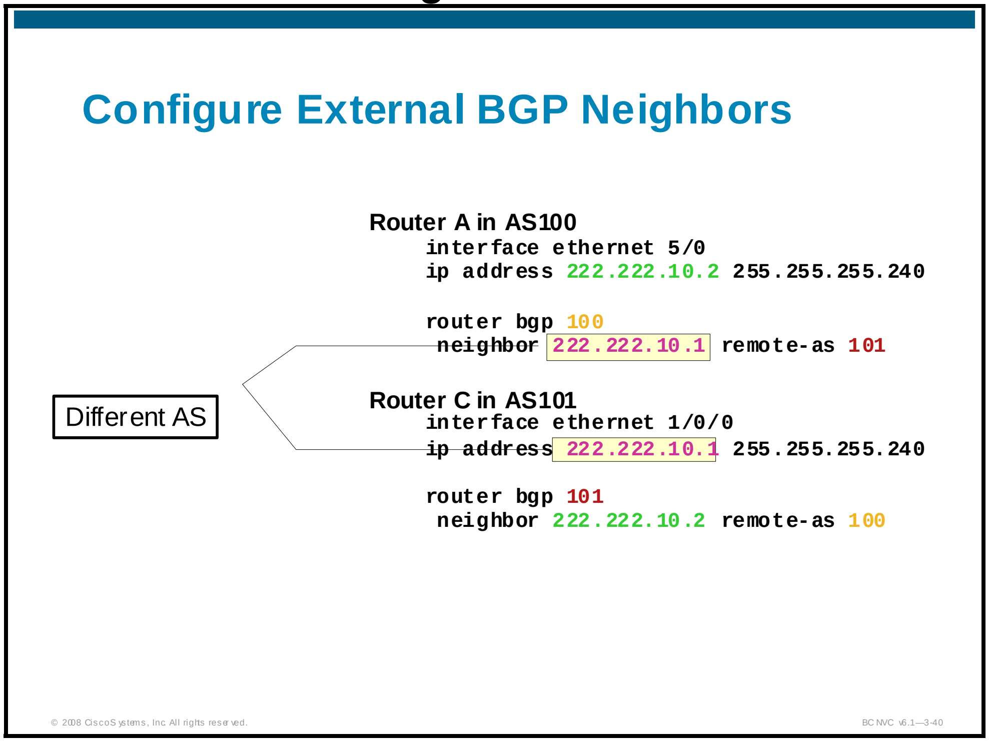
Router B

```
  interface loopback 0
  ip address 215.10.7.2 255.255.255.255

  router bgp 100
    neighbor 215.10.7.1 remote-as 100
    neighbor 215.10.7.1 update-source loopback0
```

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—3-39

To enable BGP routing, establish a BGP routing process: (router bgp as-number). Then define neighbors: (neighbor [ip-address | peer-group-name] remote-as as-number).

Configure External BGP Neighbors



Configure External BGP Neighbors

Different AS

Router A in AS100

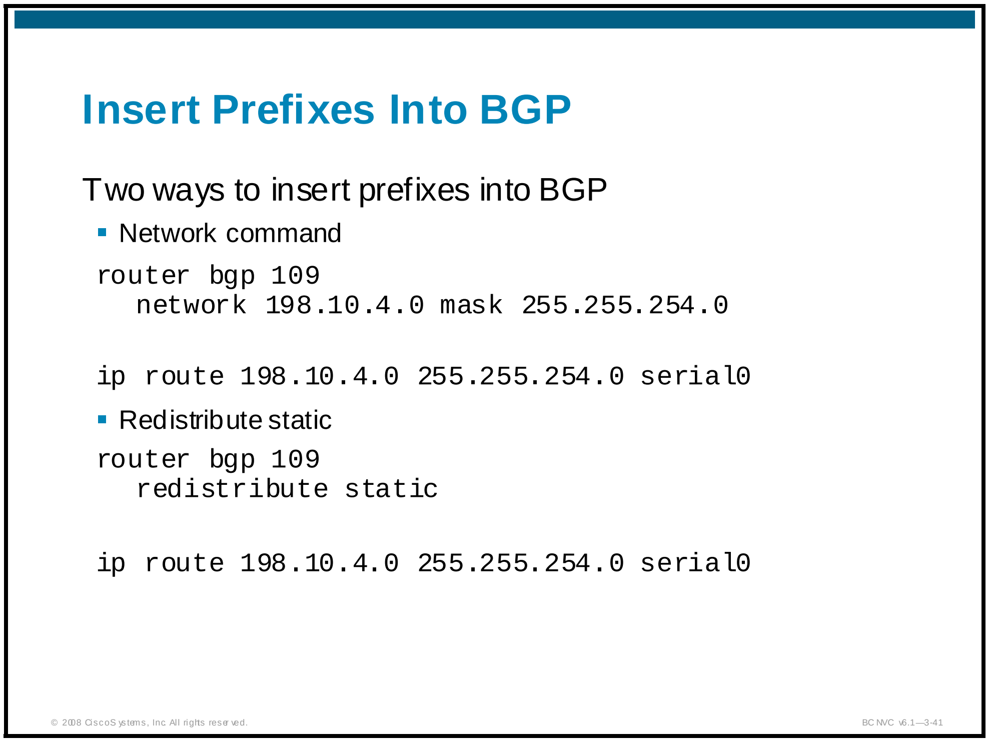
```
     interface ethernet 5/0
     ip address 222.222.10.2 255.255.255.240

     router bgp 100
      neighbor 222.222.10.1 remote-as 101
```

Router C in AS101

```
     interface ethernet 1/0/0
     ip address 222.222.10.1 255.255.255.240

     router bgp 101
      neighbor 222.222.10.2 remote-as 100
```

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—3-40

Insert Prefixes Into BGP



Insert Prefixes Into BGP

Two ways to insert prefixes into BGP

- Network command

```
router bgp 109
  network 198.10.4.0 mask 255.255.254.0

ip route 198.10.4.0 255.255.254.0 serial0
```

- Redistribute static

```
router bgp 109
  redistribute static

ip route 198.10.4.0 255.255.254.0 serial0
```

BCNVC v6.1—3-41

The network statement informs BGP what prefixes it is permitted to announce. The optional mask statement causes aggregation of all CIDR blocks smaller than the network-mask combination into the larger supernet. All routes in the IP table that fall within this range are advertised as originating from that router.

Caveats to inserting prefixes into BGP with the **network** command:

- Matching route must exist in the routing table before the network is announced
- Origin must be IGP

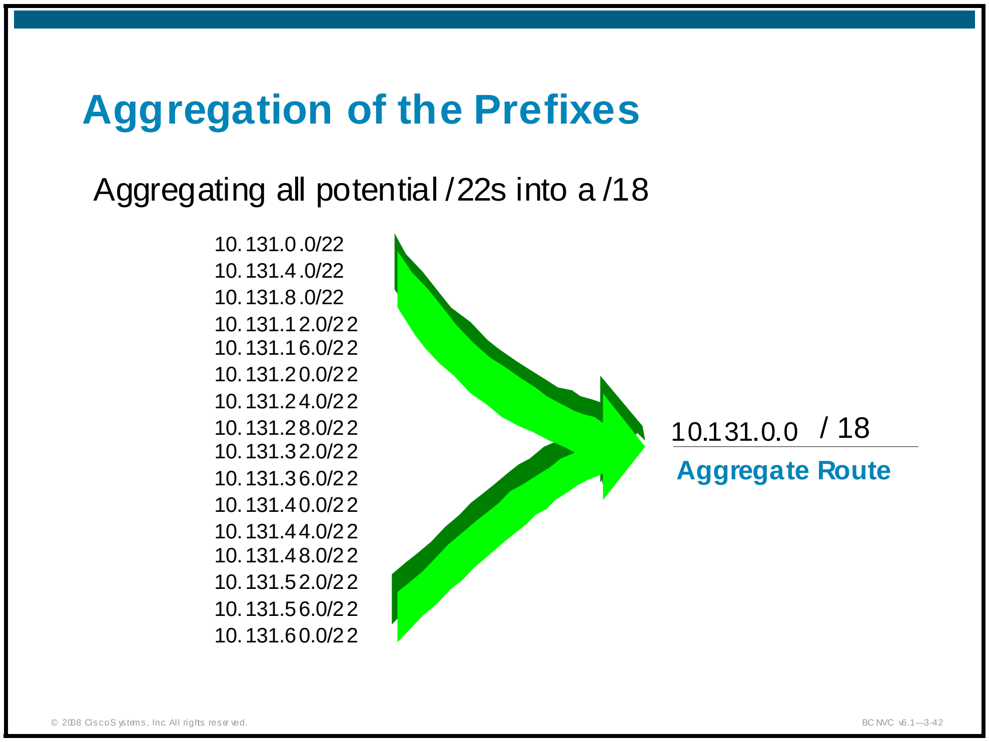
Use care when employing redistribution. Redistribute <routing-protocol> means everything inserted into the IP routing table by <routing-protocol> is transferred into the current routing protocol.

- Will not scale if uncontrolled
- Best avoided if at all possible
- Normally used with route-maps and under tight administrative control

Caveats to inserting prefixes into BGP with the **redistribute static** command.

- Static route must exist before the **redistribute** command will work
- Origin must be incomplete

Aggregation of the Prefixes



Aggregation of the Prefixes

Aggregating all potential /22s into a /18

10.131.0.0/22
10.131.4.0/22
10.131.8.0/22
10.131.12.0/22
10.131.16.0/22
10.131.20.0/22
10.131.24.0/22
10.131.28.0/22
10.131.32.0/22
10.131.36.0/22
10.131.40.0/22
10.131.44.0/22
10.131.48.0/22
10.131.52.0/22
10.131.56.0/22
10.131.60.0/22

10.131.0.0 / 18

Aggregate Route

BCNVC v6.1—3-42

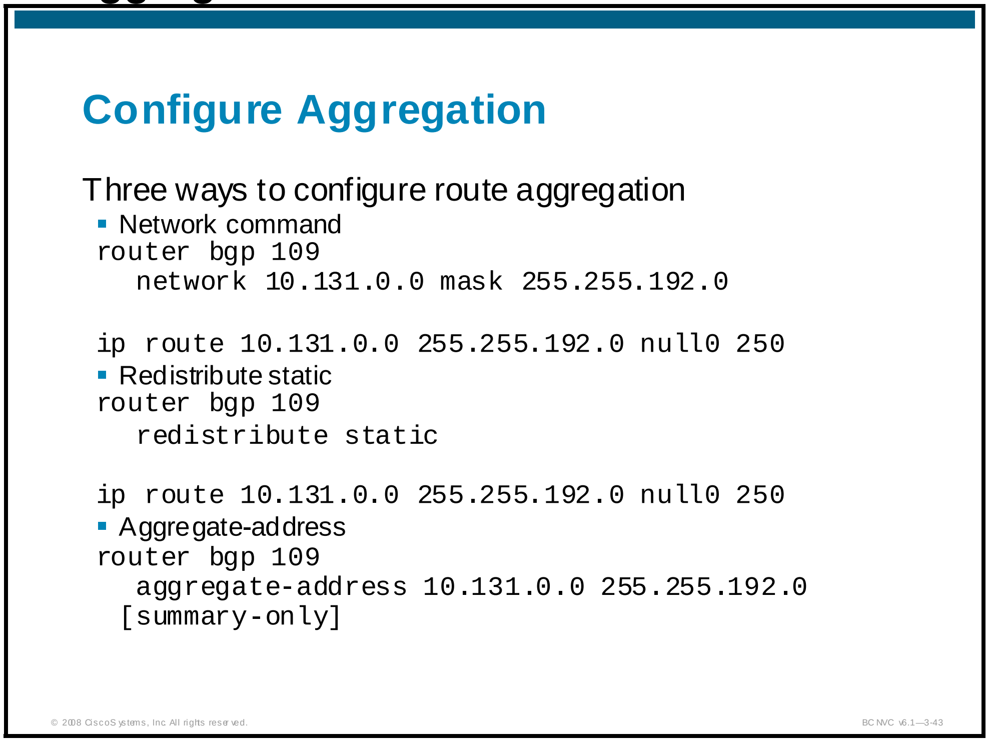
Aggregation means announcing the address block only, not subprefixes. This is the same as summarization in OSPF.

ISPs receive an address block from a regional registry or upstream provider. This address block should be announced to the Internet as an aggregate. Subprefixes of address block should *not* be announced to the Internet unless there are special circumstances.

Note	A more specific prefix can be leaked in the case where you want a specific interface to be used. This is discussed in a later chapter.

ISPs that do not aggregate are held in poor regard by the Internet community.

Configure Aggregation



Configure Aggregation

Three ways to configure route aggregation

- Network command

```
router bgp 109
   network 10.131.0.0 mask 255.255.192.0

ip route 10.131.0.0 255.255.192.0 null0 250
```

- Redistribute static

```
router bgp 109
   redistribute static

ip route 10.131.0.0 255.255.192.0 null0 250
```

- Aggregate-address

```
router bgp 109
   aggregate-address 10.131.0.0 255.255.192.0
  [summary-only]
```

Consider the above example.

Note You must have a route inside the aggregate CIDR block before the network is announced. A way to kick-start the announcement is to statically route the entire aggregate prefix to Null0. Statically routing large blocks of network allocations to Null0 is often referred to as "nailing down announcements." Packets are only sent here if there is not a more-specific match in the routing table. Setting the distance to 250 ensures this is a last resort static. This static route guarantees that you will announce this aggregate to your peers regardless of the status of the network, even in the case of router reloads, down circuits, or IGP mistakes. Your neighbors will appreciate that you do not send them frequent BGP updates, known as "flaps." Frequent flaps can cause your AS to be damped, but even worse, flaps have a negative affect on the Internet, as they consume router CPU resources and can even force a router reload if route-flaps are received in high enough succession.

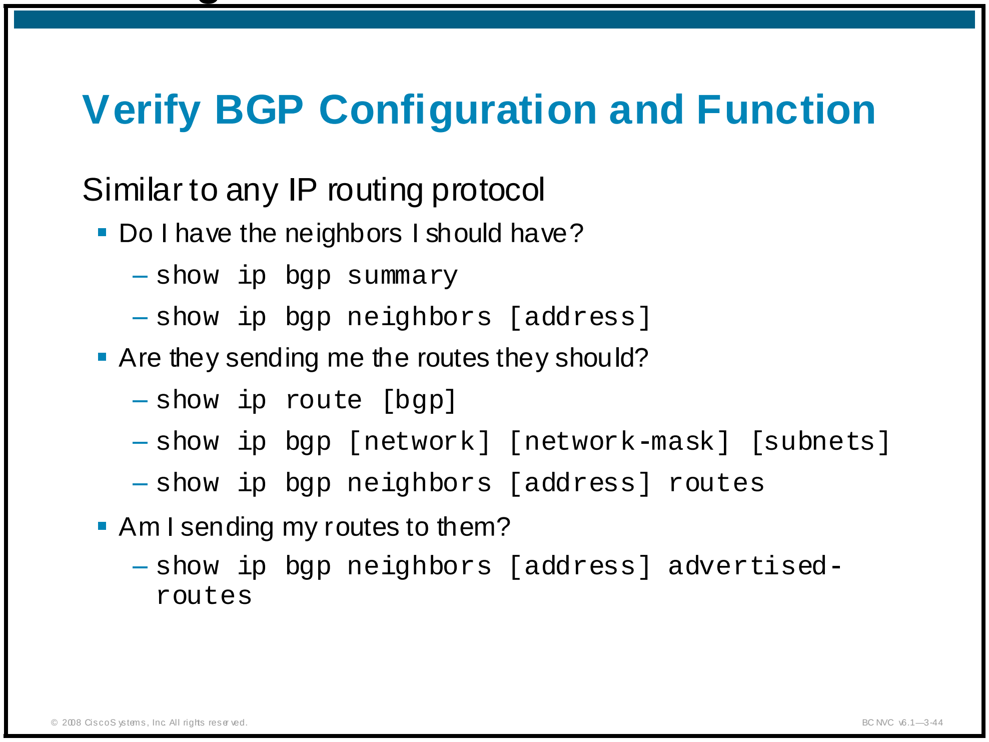
The **network** command is the easiest and preferred way of generating an aggregate. An interesting effect of adding this command is that this prefix shows up in the BGP table as being routed to Null0, but it may not be advertised. This is one of a few exceptions where a prefix that shows up in the BGP table is not be advertised to its peers.

Use care with redistribution using the **redistribute static** command.

Pertinent details of the **aggregate-address** command are:

- The **optional** {summary-only} keyword ensures that only the summary is announced even if a more specific prefix exists in the routing table
- This command does not require a pull-up route

Verify BGP Configuration and Function



Verify BGP Configuration and Function

Similar to any IP routing protocol

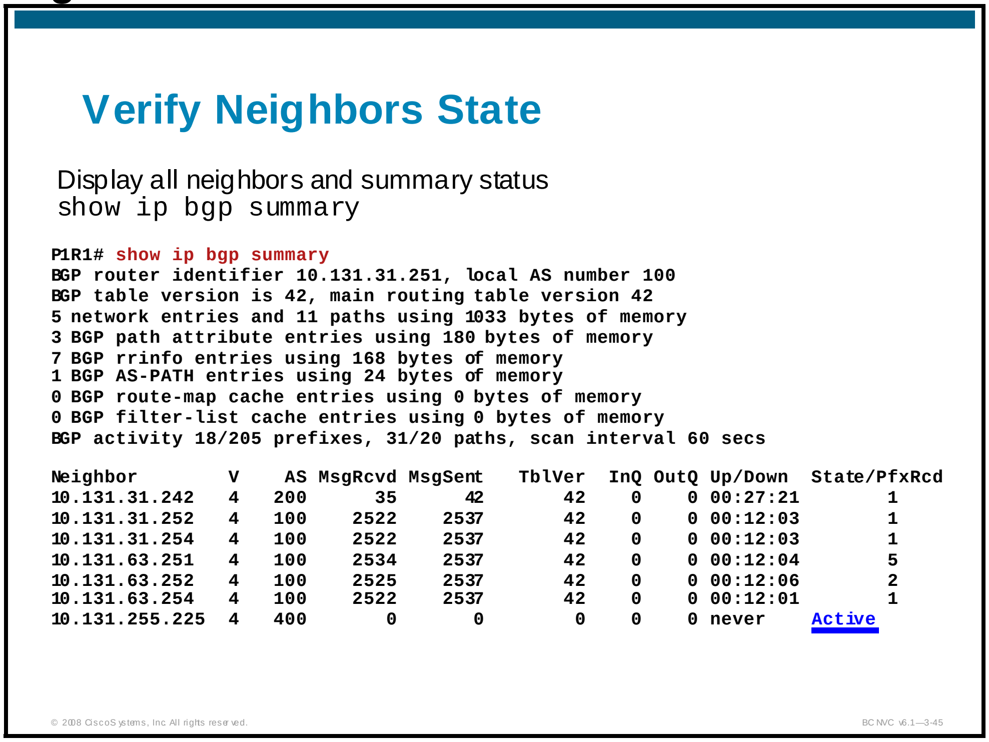
- Do I have the neighbors I should have?
  - `show ip bgp summary`
  - `show ip bgp neighbors [address]`
- Are they sending me the routes they should?
  - `show ip route [bgp]`
  - `show ip bgp [network] [network-mask] [subnets]`
  - `show ip bgp neighbors [address] routes`
- Am I sending my routes to them?
  - `show ip bgp neighbors [address] advertised-routes`

BCNVC v6.1—3-44

Common BGP commands and descriptions

Command	Description
show ip bgp summary	Display the status of all BGP connections.
show ip bgp neighbors [address] [received-routes | routes | advertised-routes | {paths *regular-expression*} | dampened-routes]	Display detailed information on the connections neighbors. (Optional) neighbor address; if omitted all neighbors are displayed (Optional) received routes from the neighbor (soft-reset) (Optional) all routes that are received and accepted (Optional) all the routes has advertised to the neighbor (Optional) used to match the paths received (Optional) dampened routes to the neighbor specified
show ip route [[ip-address [mask] [longer-prefixes]] | [protocol [process-id]] | [list access-list-number | access-list-name]]	Display the current state of the routing table (Optional) address to display information about (Optional) argument for a subnet mask (Optional) only routes matching the ip-address and mask pair (Optional) name of a routing protocol; use one of the following keywords: connected, static, summary, BGP, EGP, EIGRP, HELLO, IGRP, ISIS, OSPF, or RIP (Optional) number used to identify a process of the specified protocol (Optional) to filter output by an access list name or number (Optional) filters output based on the specified access list number (Optional) filters output based on the specified access list name
show ip bgp [*network*] [*network-mask*] [subnets]	Display the contents of the BGP routing table. (Optional) entered to display a particular network (Optional) displays all BGP routes matching the address and mask (Optional) displays the route and more specific routes

Verify Neighbors State



Verify Neighbors State

Display all neighbors and summary status

`show ip bgp summary`

```
P1R1# show ip bgp summary
BGP router identifier 10.131.31.251, local AS number 100
BGP table version is 42, main routing table version 42
5 network entries and 11 paths using 1033 bytes of memory
3 BGP path attribute entries using 180 bytes of memory
7 BGP rrinfo entries using 168 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP activity 18/205 prefixes, 31/20 paths, scan interval 60 secs

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.131.31.242   4   200      35      42       42    0    0 00:27:21        1
10.131.31.252   4   100    2522    2537       42    0    0 00:12:03        1
10.131.31.254   4   100    2522    2537       42    0    0 00:12:03        1
10.131.63.251   4   100    2534    2537       42    0    0 00:12:04        5
10.131.63.252   4   100    2525    2537       42    0    0 00:12:06        2
10.131.63.254   4   100    2522    2537       42    0    0 00:12:01        1
10.131.255.225  4   400       0       0        0    0    0 never    Active
```

BCNVC v6.1—3-45

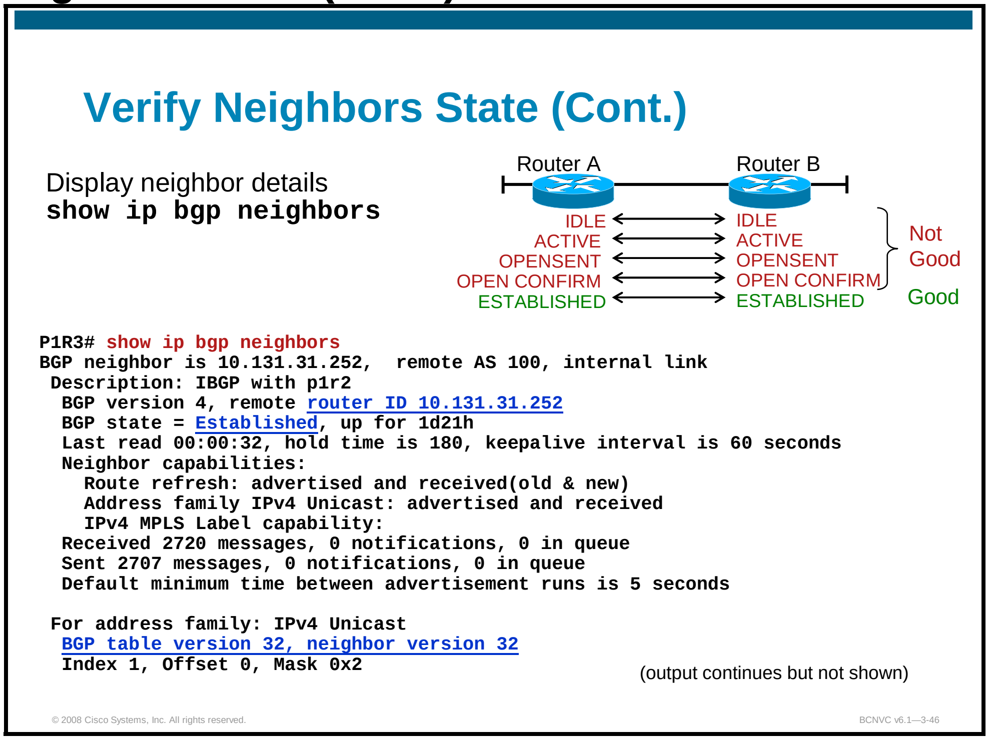
This command provides you with a quick and easy way to view your configuration and the operational state of your peers.

With this output you can check the following:

- Neighbor IP address
- Neighbor AS number
- Neighbor state
- Whether messages are being exchanged
- If prefixes are being received
- What BGP version is being used
- If table version is stable or incrementally high which indicates instability

If all of this looks good then the only thing remaining is to verify that a router has the routes it should have, and that it advertises your own routes.

Verify Neighbors State (Cont.)



Verify Neighbors State (Cont.)

Display neighbor details
show ip bgp neighbors

```
P1R3# show ip bgp neighbors
BGP neighbor is 10.131.31.252,  remote AS 100, internal link
 Description: IBGP with p1r2
  BGP version 4, remote router ID 10.131.31.252
  BGP state = Established, up for 1d21h
  Last read 00:00:32, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received(old & new)
    Address family IPv4 Unicast: advertised and received
    IPv4 MPLS Label capability:
  Received 2720 messages, 0 notifications, 0 in queue
  Sent 2707 messages, 0 notifications, 0 in queue
  Default minimum time between advertisement runs is 5 seconds

 For address family: IPv4 Unicast
  BGP table version 32, neighbor version 32
  Index 1, Offset 0, Mask 0x2
```

(output continues but not shown)

BCNVC v6.1—3-46

Anything other than "state = established" indicates that the peers are not up.

The remote router ID is the highest IP address on that router (or the highest loopback interface, if there is one). Make sure there is IGP reachability to that address (and vice versa for the router ID).

Notice the table version number: Each time the table is updated by new incoming information, the table version number increments. A table version number that continually increments is an indication that a route is flapping, thereby causing routes to be updated continually.

Note When you make a configuration change with respect to a neighbor for which a peer relationship has been established, be sure to reset the BGP session with that neighbor. To reset the session, at the system prompt, issue the **clear ip bgp** EXEC command specifying the IP address of that neighbor.

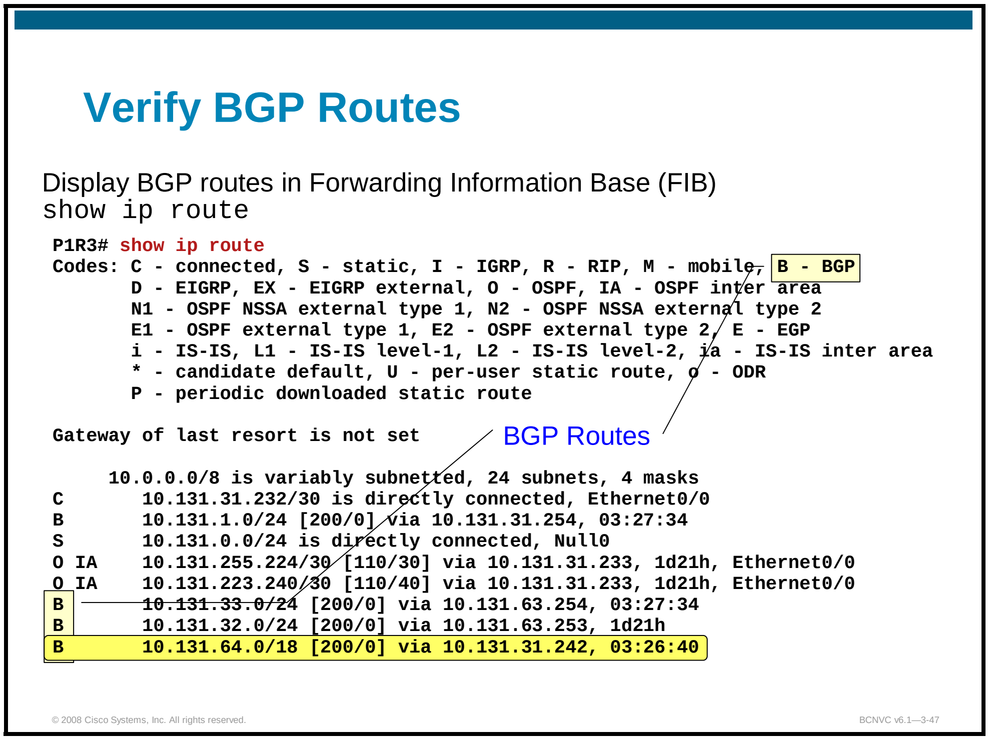
Everything happens between each pair of neighbors

- Idle STATE:
  - Does not try to create a TCP session
  - Idles for 20 seconds after configuration or clear
  - Listens if a neighbor tries to establish a TCP session
- Connect STATE:
  - Neighbor establishes a TCP connection
  - Is fast and difficult to see
- Active STATE:
  - Tries to actively establish a TCP connection

- Makes continued attempts
- Waits according to the ConnectRetry Timer (with –50 percent jitter) and tries again, if it does not succeed
- 30 seconds for EBGP, 10 seconds for IBGP

- OpenSent STATE:
  - Has already established TCP connection
  - Has sent an OPEN message
- OpenReceive STATE:
  - Has received an OPEN message from a neighbor
- Established STATE:
  - Has received an initial KEEPALIVE
- Now, the router can start sending and receiving UPDATES
  - Cisco sends another KEEPALIVE after the initial set of updates have been sent
- Admin Idle STATE:
  - State defined by Cisco
  - Configurable
  - [no] neighbor <neighbor> shutdown
  - Never transitions to connect state
  - Never listens to incoming TCP connections

Verify BGP Routes



Verify BGP Routes

Display BGP routes in Forwarding Information Base (FIB)
`show ip route`

```
P1R3# show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/8 is variably subnetted, 24 subnets, 4 masks
C       10.131.31.232/30 is directly connected, Ethernet0/0
B       10.131.1.0/24 [200/0] via 10.131.31.254, 03:27:34
S       10.131.0.0/24 is directly connected, Null0
O IA    10.131.255.224/30 [110/30] via 10.131.31.233, 1d21h, Ethernet0/0
O IA    10.131.223.240/30 [110/40] via 10.131.31.233, 1d21h, Ethernet0/0
B       10.131.33.0/24 [200/0] via 10.131.63.254, 03:27:34
B       10.131.32.0/24 [200/0] via 10.131.63.253, 1d21h
B       10.131.64.0/18 [200/0] via 10.131.31.242, 03:26:40
```

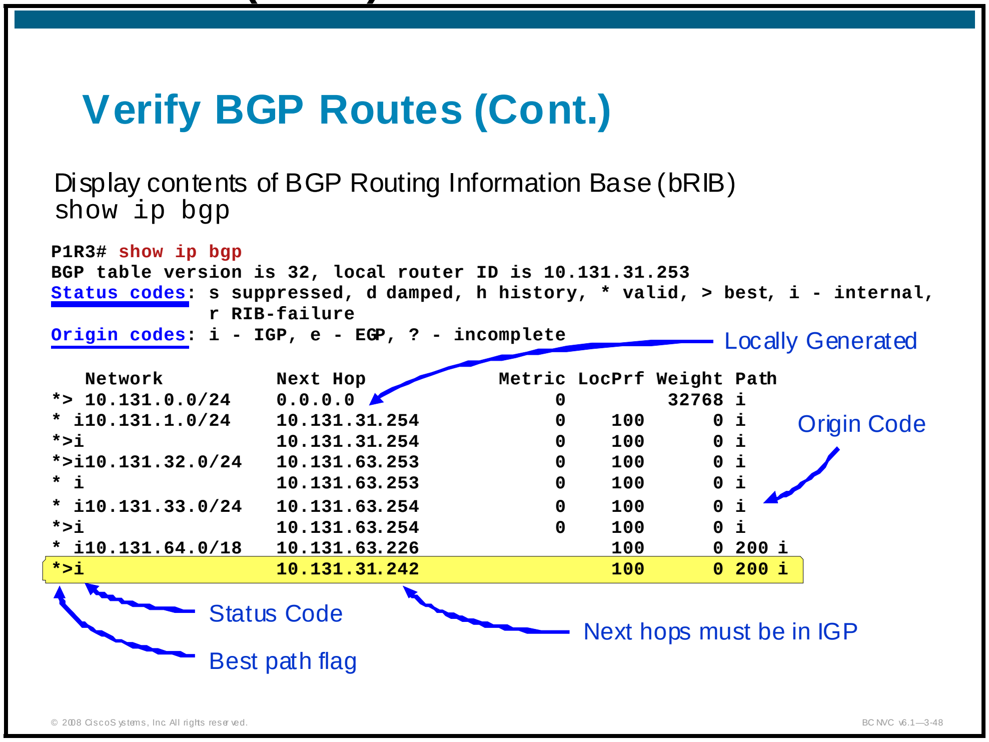
BGP Routes

BCNVC v6.1—3-47

Recall that the BGP best path is installed in the Forwarding Information Base (FIB) from the BGP Routing Information Base. When you see BGP routes in the forwarding table you know that some things are working.

Viewing the BGP table provides more detail.

Verify BGP Routes (Cont.)



Verify BGP Routes (Cont.)

Display contents of BGP Routing Information Base (bRIB)
`show ip bgp`

```
P1R3# show ip bgp
BGP table version is 32, local router ID is 10.131.31.253
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 10.131.0.0/24    0.0.0.0                  0         32768 i
* i10.131.1.0/24    10.131.31.254            0    100      0 i
*>i                 10.131.31.254            0    100      0 i
*>i10.131.32.0/24   10.131.63.253            0    100      0 i
* i                 10.131.63.253            0    100      0 i
* i10.131.33.0/24   10.131.63.254            0    100      0 i
*>i                 10.131.63.254            0    100      0 i
* i10.131.64.0/18   10.131.63.226                 100      0 200 i
*>i                 10.131.31.242                 100      0 200 i
```

BCNVC v6.1—3-48

Notice that any locally generated entry, such as 10.131.0.0/24, has a next hop of 0.0.0.0.

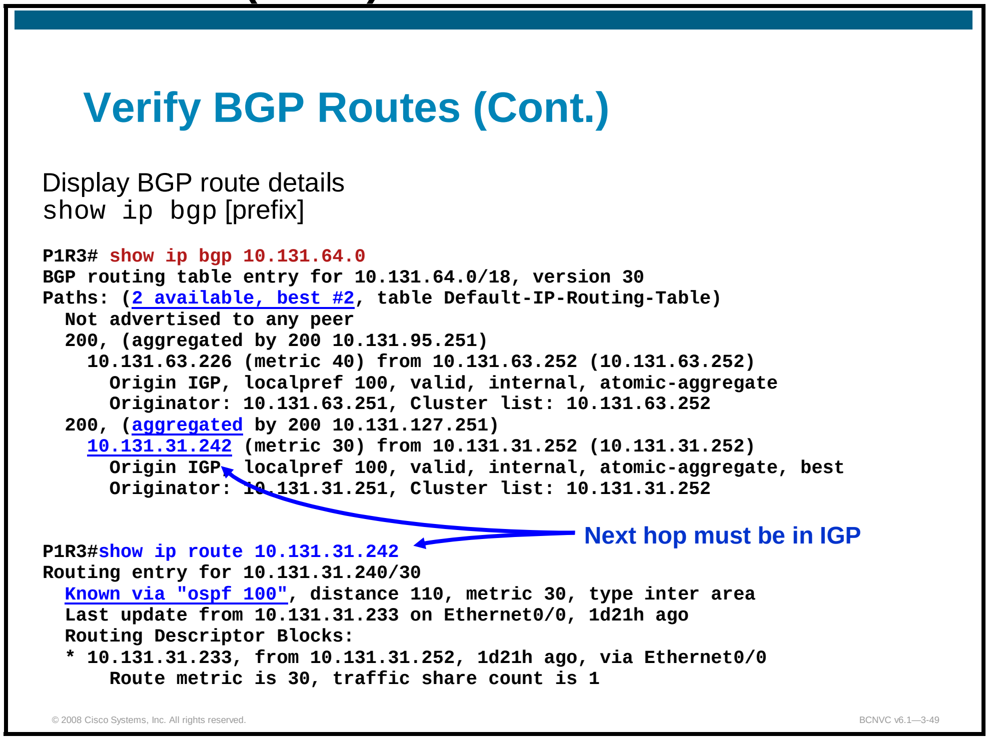
The letter “i” at the beginning of a line means that the entry was learned via an internal BGP peer. The letter “i” at the end of a line indicates that the path information comes from an IGP.

You would read the last prefix above as follows: network 10.131.64.0/18 has been learned from two sources via path 200, next hops can be 10.131.63.226 or 10.131.31.242, the next hop of 10.131.31.242 is the best path.

The “>” symbol indicates that BGP has chosen the best route based on the decision steps described in a later chapter. If there are multiple paths to a prefix, BGP picks only the one route that it determines to be the best route. It installs this route in the IP routing table and advertises it to other BGP peers. Note the next hop attribute of 10.131.31.242, which is the EBGP next hop carried in the IGP.

Note	Next-hop, AS path, metric, local preference, and weight are discussed in a later chapter.

Verify BGP Routes (Cont.)



Verify BGP Routes (Cont.)

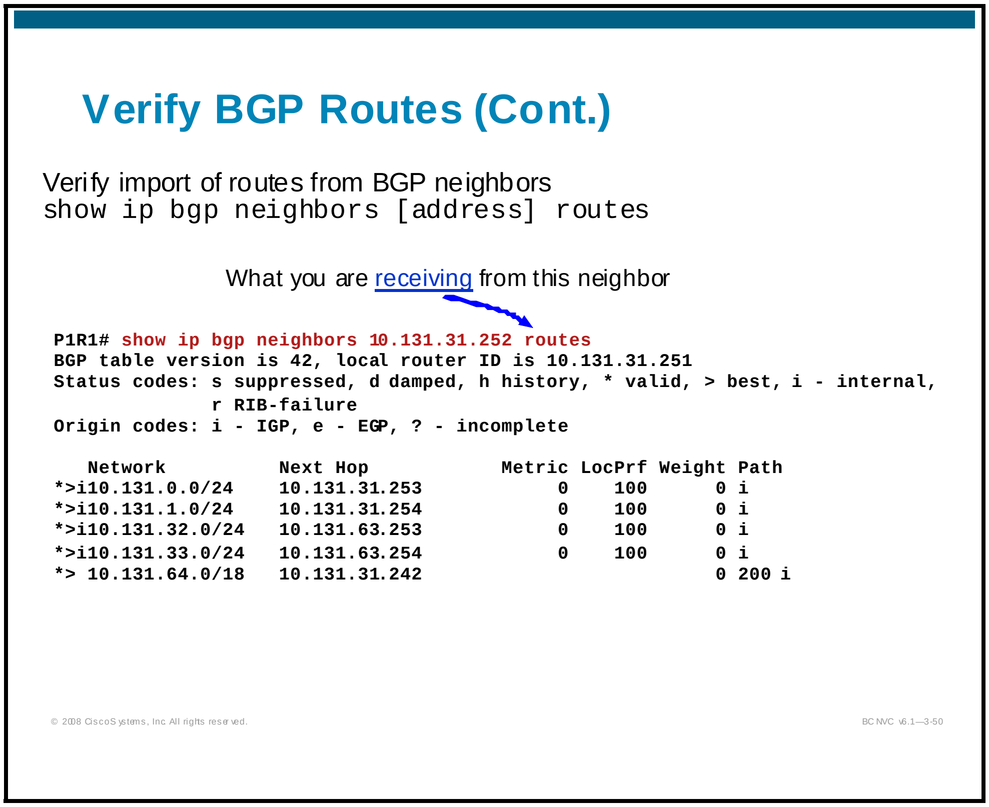
Display BGP route details
`show ip bgp` [prefix]

```
P1R3# show ip bgp 10.131.64.0
BGP routing table entry for 10.131.64.0/18, version 30
Paths: (2 available, best #2, table Default-IP-Routing-Table)
  Not advertised to any peer
  200, (aggregated by 200 10.131.95.251)
    10.131.63.226 (metric 40) from 10.131.63.252 (10.131.63.252)
      Origin IGP, localpref 100, valid, internal, atomic-aggregate
      Originator: 10.131.63.251, Cluster list: 10.131.63.252
  200, (aggregated by 200 10.131.127.251)
    10.131.31.242 (metric 30) from 10.131.31.252 (10.131.31.252)
      Origin IGP, localpref 100, valid, internal, atomic-aggregate, best
      Originator: 10.131.31.251, Cluster list: 10.131.31.252
```

Next hop must be in IGP

```
P1R3#show ip route 10.131.31.242
Routing entry for 10.131.31.240/30
  Known via "ospf 100", distance 110, metric 30, type inter area
  Last update from 10.131.31.233 on Ethernet0/0, 1d21h ago
  Routing Descriptor Blocks:
  * 10.131.31.233, from 10.131.31.252, 1d21h ago, via Ethernet0/0
      Route metric is 30, traffic share count is 1
```

BCNVC v6.1—3-49



Verify BGP Routes (Cont.)

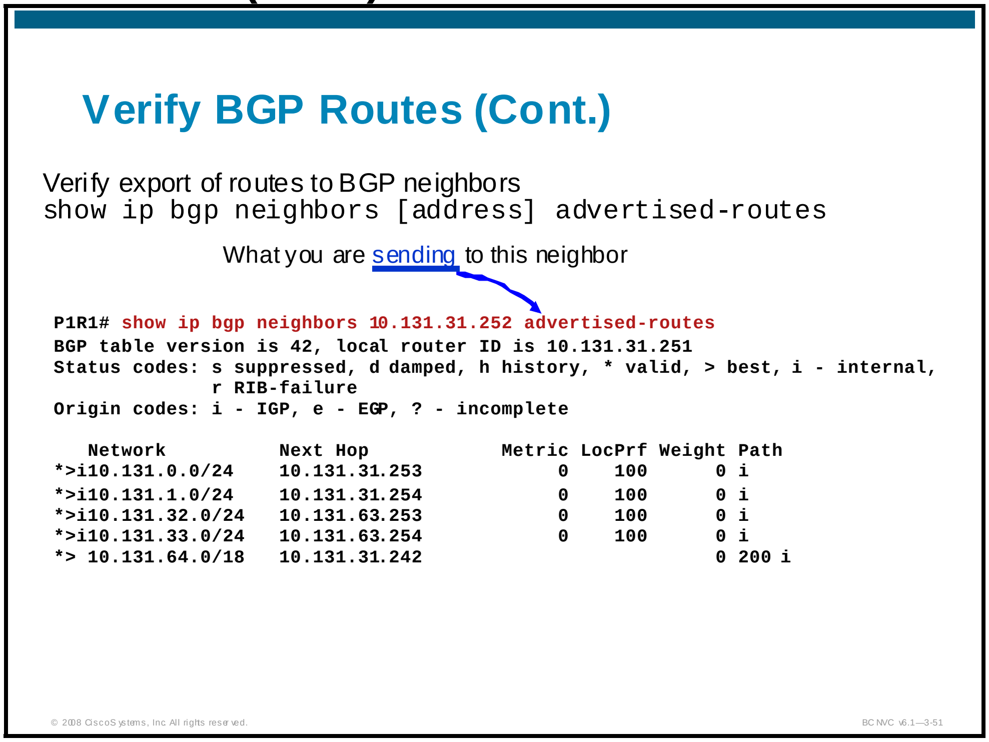
Verify import of routes from BGP neighbors
`show ip bgp neighbors [address] routes`

What you are receiving from this neighbor

```
P1R1# show ip bgp neighbors 10.131.31.252 routes
BGP table version is 42, local router ID is 10.131.31.251
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i10.131.0.0/24    10.131.31.253            0    100      0 i
*>i10.131.1.0/24    10.131.31.254            0    100      0 i
*>i10.131.32.0/24   10.131.63.253            0    100      0 i
*>i10.131.33.0/24   10.131.63.254            0    100      0 i
*> 10.131.64.0/18   10.131.31.242                          0 200 i
```

BCNVC v6.1—3-50

Verify BGP Routes (Cont.)



Verify BGP Routes (Cont.)

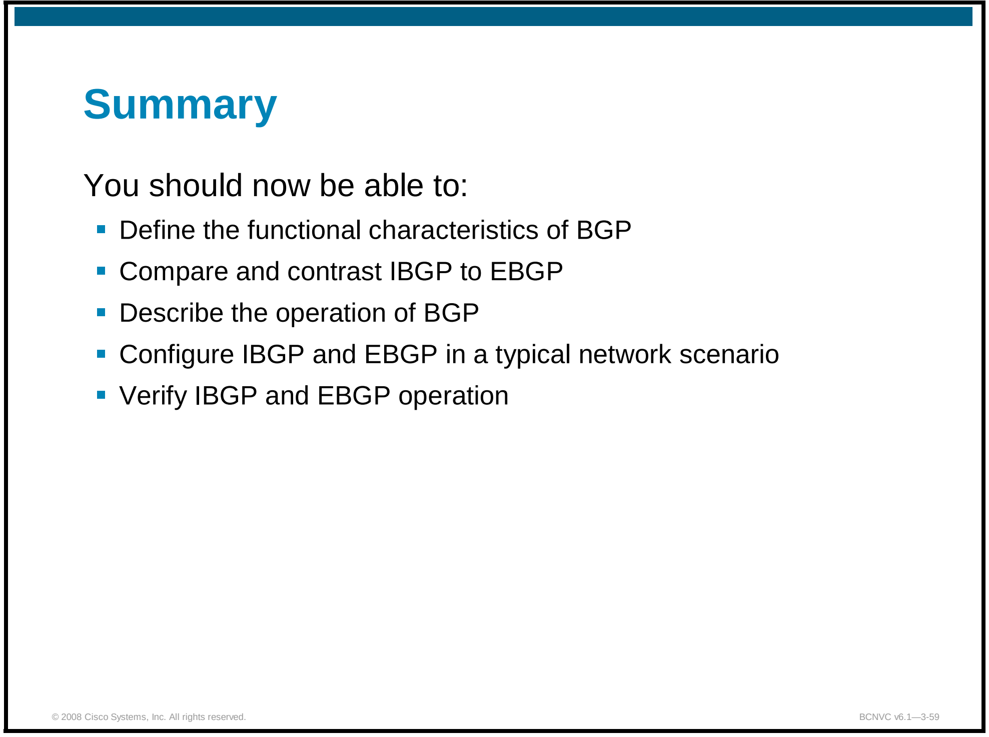
Verify export of routes to BGP neighbors

`show ip bgp neighbors [address] advertised-routes`

What you are sending to this neighbor

```
P1R1# show ip bgp neighbors 10.131.31.252 advertised-routes
BGP table version is 42, local router ID is 10.131.31.251
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i10.131.0.0/24    10.131.31.253            0    100      0 i
*>i10.131.1.0/24    10.131.31.254            0    100      0 i
*>i10.131.32.0/24   10.131.63.253            0    100      0 i
*>i10.131.33.0/24   10.131.63.254            0    100      0 i
*> 10.131.64.0/18   10.131.31.242                          0 200 i
```

Summary



Summary

You should now be able to:

- Define the functional characteristics of BGP
- Compare and contrast IBGP to EBGP
- Describe the operation of BGP
- Configure IBGP and EBGP in a typical network scenario
- Verify IBGP and EBGP operation

BCNVC v6.1—3-59

This module has been an introduction to BGP. This module has covered the following major questions and answers:

What Is BGP?

BGP is used to scale routing to Internet dimensions. It is predominantly used to connect ISPs, which are distinguished by the use of a globally unique Autonomous System number.

Why Do We Need EBGP?

To hide an IGP, multihome, or for suboptimal routing.

You do not have to use BGP to connect to the Internet. BGP can be useful to an enterprises that multihomes. It can also be useful for scaling enterprise networks, in particular to scale beyond the ability of most IGP implementations to carry only a few thousand routes. It can provide better autonomy between departments in an Enterprise than is possible with an IGP.

How Does BGP Work?

We’ve seen how BGP operates over the reliable transport of TCP (port 179) and uses incremental updates. Two types of BGP session are used: external BGP between autonomous systems, and internal BGP within an AS. Internal BGP requires all routers to by fully meshed, and therefore presents some scaling problems.

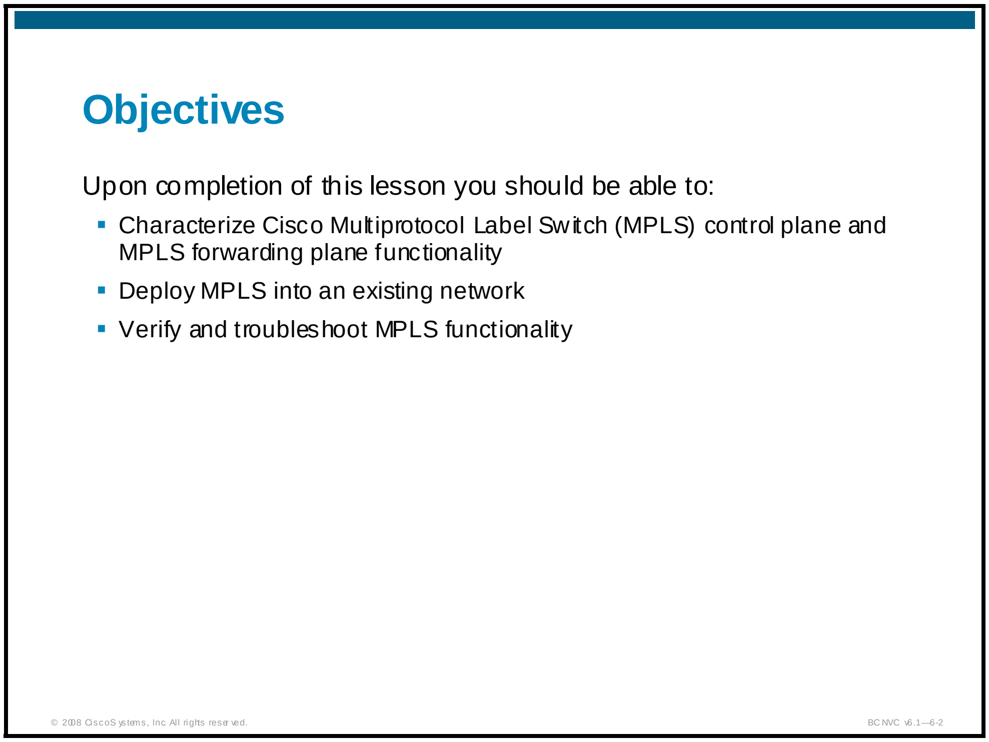
How Do I Configure BGP?

Simple to implement, complex to understand.

Lesson 4

Implement and Troubleshoot MPLS

Objectives

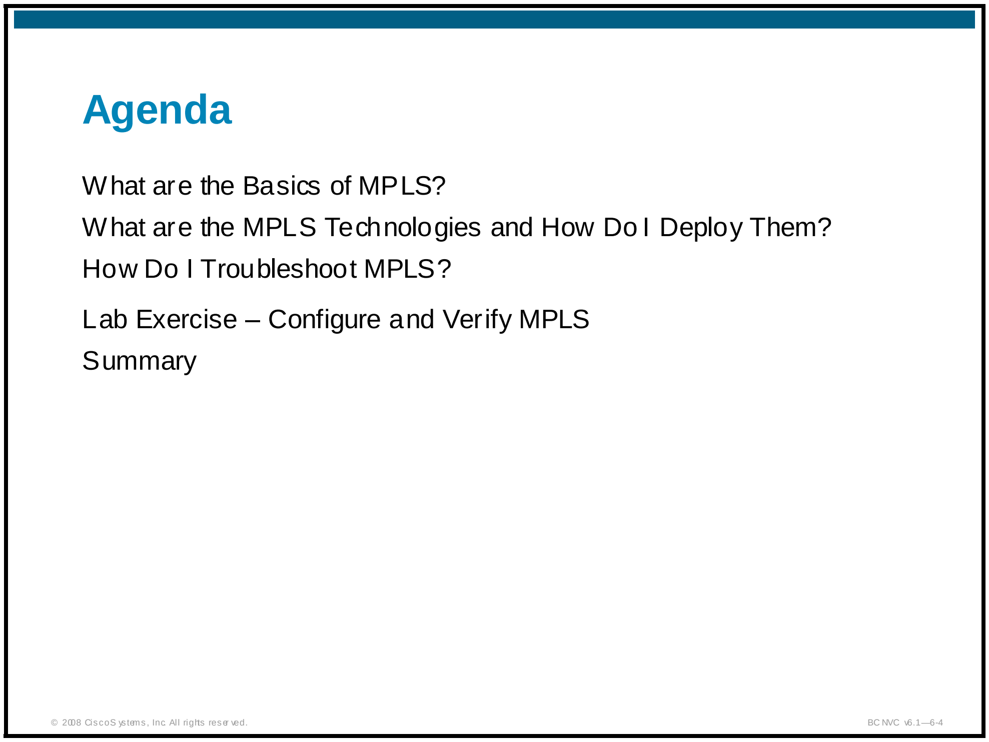


Objectives

Upon completion of this lesson you should be able to:

- Characterize Cisco Multiprotocol Label Switch (MPLS) control plane and MPLS forwarding plane functionality
- Deploy MPLS into an existing network
- Verify and troubleshoot MPLS functionality

Agenda



Agenda

What are the Basics of MPLS?

What are the MPLS Technologies and How Do I Deploy Them?

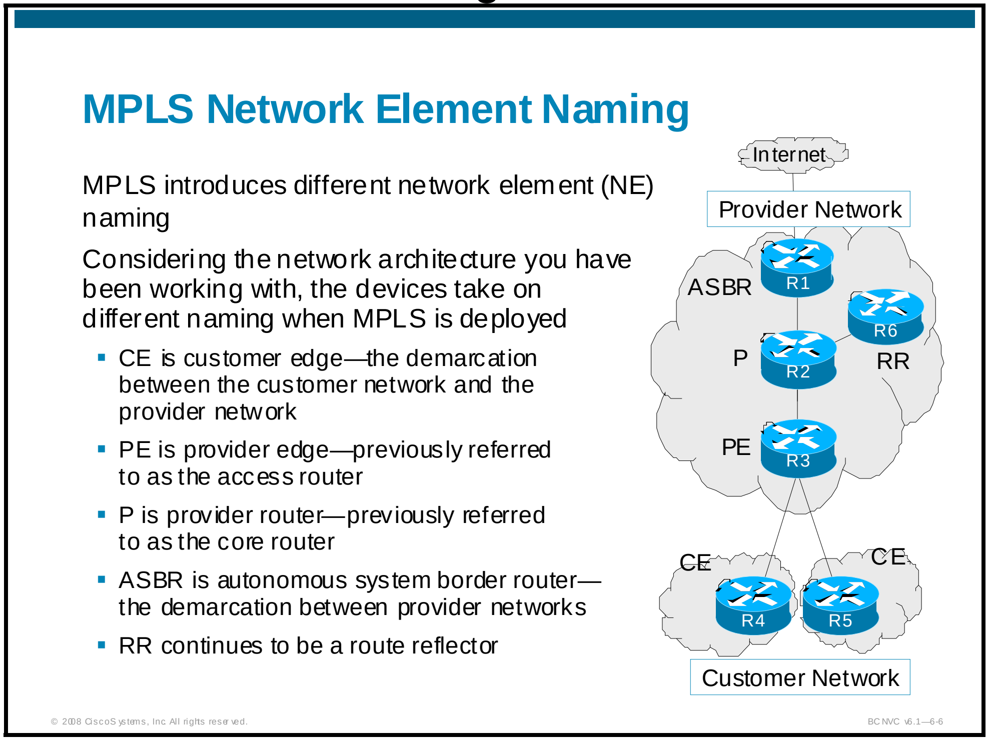
How Do I Troubleshoot MPLS?

Lab Exercise – Configure and Verify MPLS

Summary

What Are the Basics of MPLS?

MPLS Network Element Naming



MPLS Network Element Naming

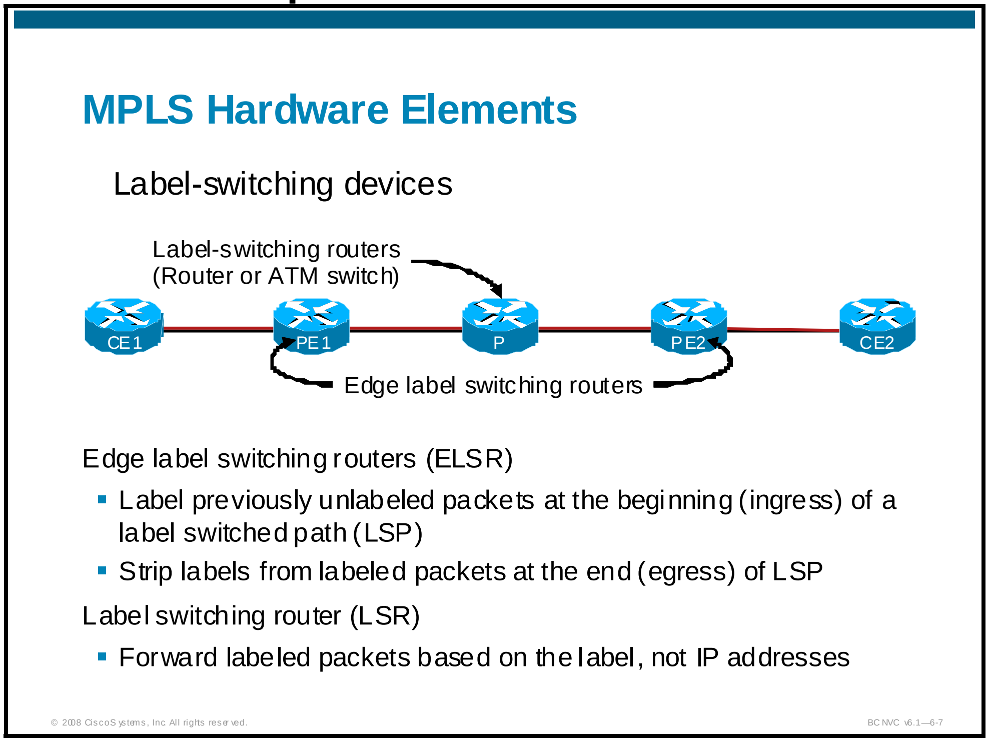
MPLS introduces different network element (NE) naming

Considering the network architecture you have been working with, the devices take on different naming when MPLS is deployed

- CE is customer edge—the demarcation between the customer network and the provider network
- PE is provider edge—previously referred to as the access router
- P is provider router—previously referred to as the core router
- ASBR is autonomous system border router—the demarcation between provider networks
- RR continues to be a route reflector

BCNVC v6.1—6-6

MPLS Hardware Components


CE1
PE1
P
PE2
CE2

MPLS Hardware Elements

Label-switching devices

Label-switching routers
(Router or ATM switch)

Edge label switching routers

Edge label switching routers (ELSR)

- Label previously unlabeled packets at the beginning (ingress) of a label switched path (LSP)
- Strip labels from labeled packets at the end (egress) of LSP

Label switching router (LSR)

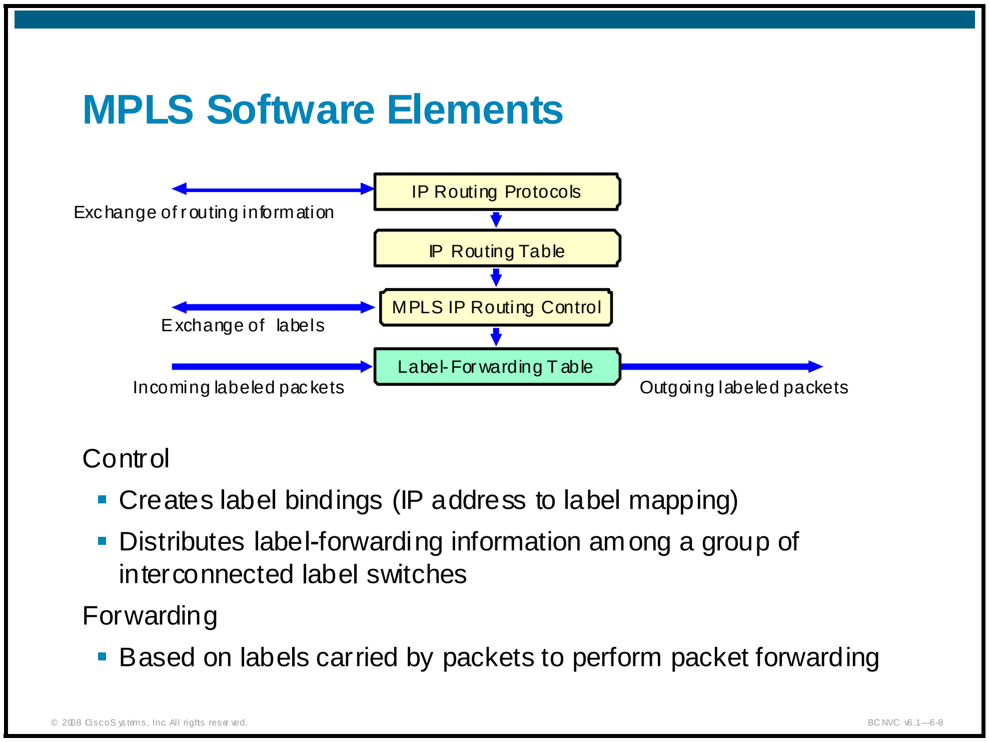
- Forward labeled packets based on the label, not IP addresses

BCNVC v6.1—6-7

LSR Types

LSR Type	Actions Performed by This LSR Type
LSR	Forwards labeled packets.
Edge-LSR	Can receive an IP packet, perform Layer 3 lookups, and impose a label stack before forwarding the packet into the LSR domain. Can receive a labeled packet, remove labels, perform Layer 3 lookups, and forward the IP packet toward its next-hop.
ATM-LSR	Runs MPLS protocols in the control plane to set up ATM virtual circuits. Forwards labeled packets as ATM cells.
ATM edge-LSR	Can receive a labeled or unlabeled packet, segment it into ATM cells, and forward the cells toward the next-hop ATM-LSR. Can receive ATM cells from an adjacent ATM-LSR, reassemble these cells into the original packet, and then forward the packet as a labeled or unlabeled packet.

MPLS Software Elements



Exchange of routing information
IP Routing Protocols
IP Routing Table
Exchange of labels
MPLS IP Routing Control
Incoming labeled packets
Label-Forwarding Table
Outgoing labeled packets

MPLS Software Elements

Control

- Creates label bindings (IP address to label mapping)
- Distributes label-forwarding information among a group of interconnected label switches

Forwarding

- Based on labels carried by packets to perform packet forwarding

MPLS relies on two principal components: forwarding and control. The forwarding component uses labels carried by packets and the label-forwarding information maintained by an LSR to perform packet forwarding. The control component is responsible for maintaining correct label-forwarding information among a group of interconnected label switches (LSRs).

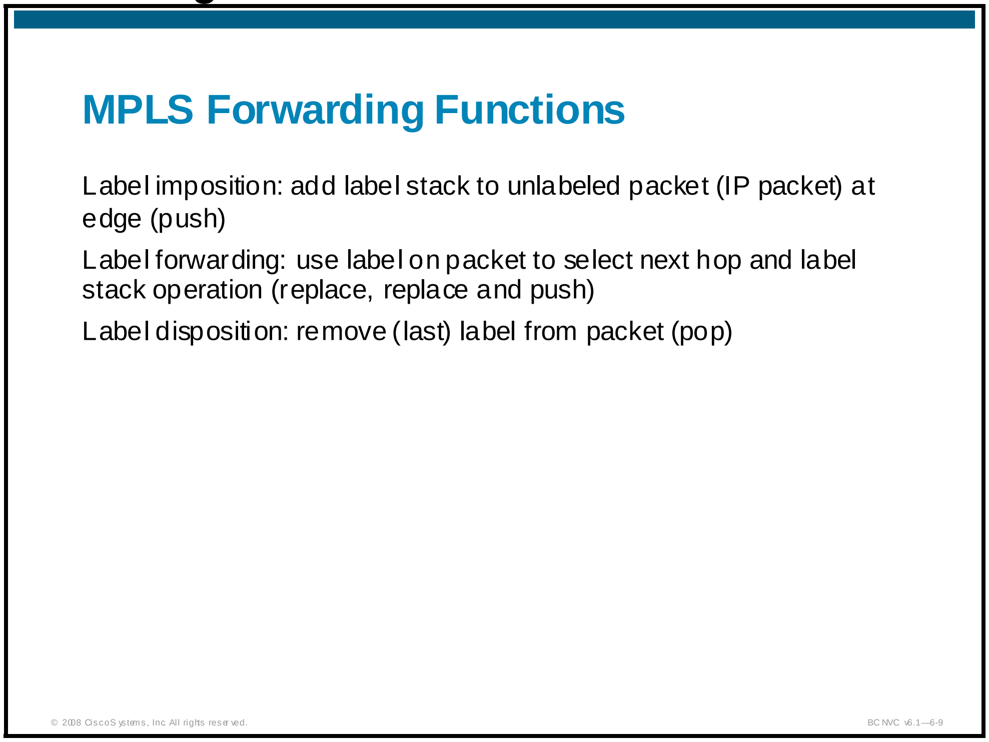
The MPLS architecture is split into two major components:

- Control plane – Used to exchange Layer 3 routing information and labels
- Data plane – Used to forward the actual packets based on labels

The control plane uses a number of features depending on the application being used (unicast IP routing, traffic engineering, and MPLS VPN).

The key portion of the control plane is Label Distribution Protocol (LDP). LDP builds the Label Forwarding Information Base (LFIB) so that the router can switch labels.

MPLS Forwarding Functions



MPLS Forwarding Functions

Label imposition: add label stack to unlabeled packet (IP packet) at edge (push)

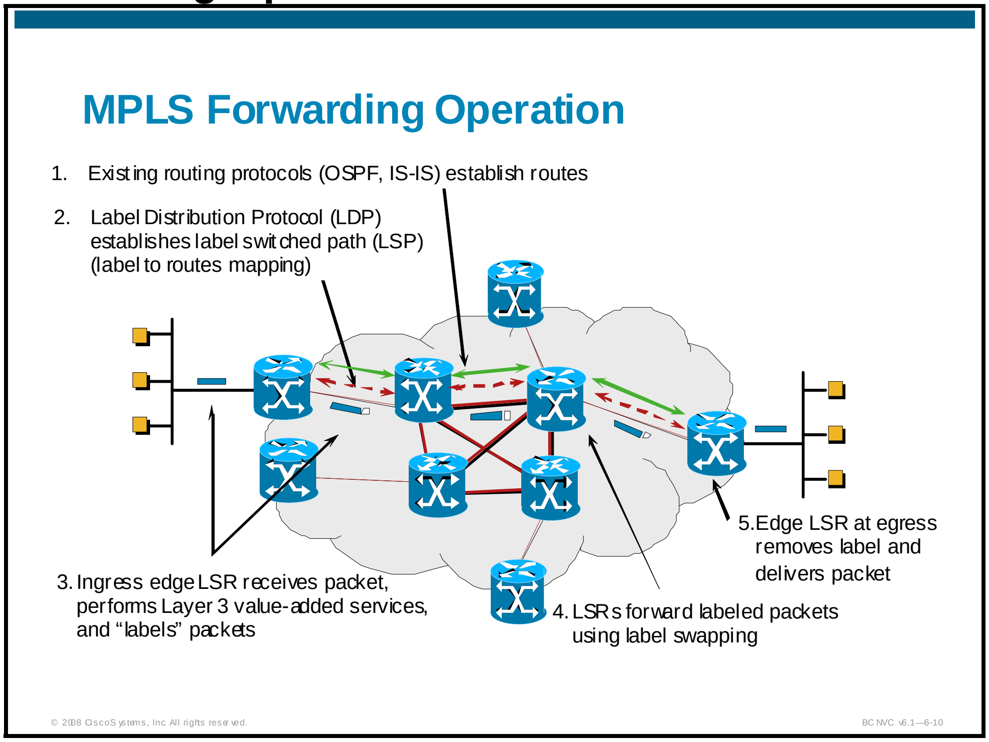
Label forwarding: use label on packet to select next hop and label stack operation (replace, replace and push)

Label disposition: remove (last) label from packet (pop)

BCNVC v6.1—6-9

An edge-LSR is a router that performs either label imposition (sometimes also referred to as push action) or label disposition (also called pop action) at the edge of an MPLS network. Label imposition is the act of prepending a label, or a stack of labels, to a packet in the ingress point (in respect of the traffic flow from source to destination) of the MPLS domain. Label disposition is the reverse and is the act of removing the last label from a packet at the egress point before it is forwarded to a neighbor that is outside the MPLS domain.

MPLS Forwarding Operation



MPLS Forwarding Operation

1. Existing routing protocols (OSPF, IS-IS) establish routes
2. Label Distribution Protocol (LDP) establishes label switched path (LSP) (label to routes mapping)
3. Ingress edge LSR receives packet, performs Layer 3 value-added services, and “labels” packets
4. LSRs forward labeled packets using label swapping
5. Edge LSR at egress removes label and delivers packet

BCMVC v6.1—6-10

Here is how MPLS carries traffic:

1. A routing protocol such as OSPF, EIGRP, or IS-IS determines the Layer 3 topology. A router builds a routing table as it “listens” to the network. A Cisco router or IP+ATM switch can have a routing function inside that does this. All devices in the network build the Layer 3 topology.

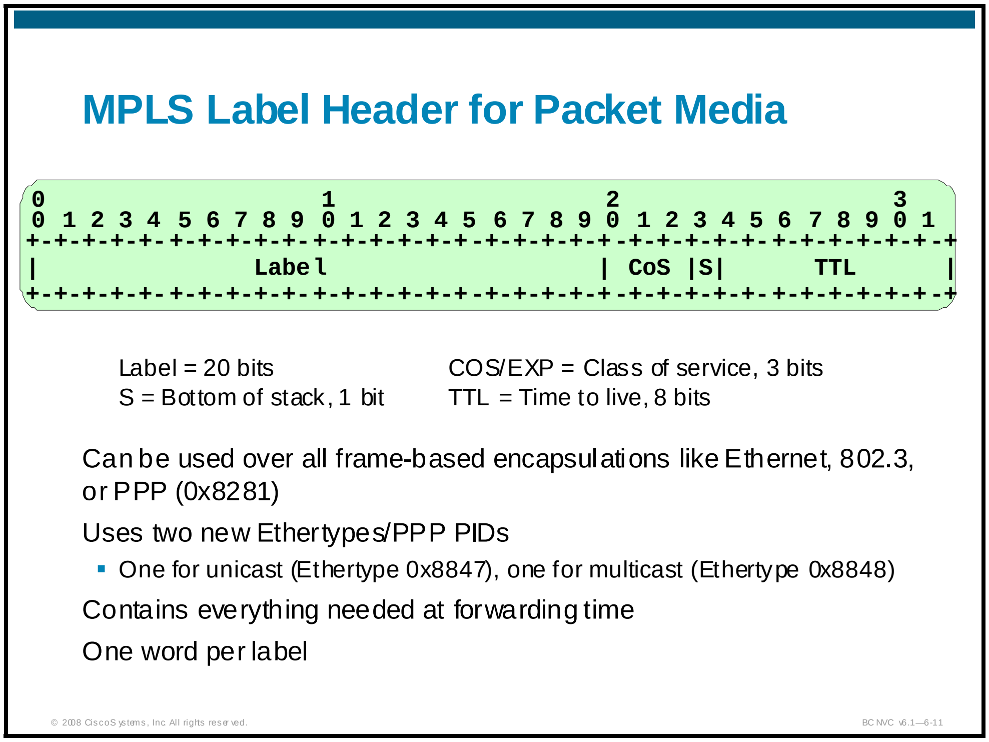
2. The LDP establishes label values for each device according to the routing topology, to preconfigure maps to destination points. Unlike ATM PVCs where the VPI/VCIs are manually assigned, labels are assigned automatically by LDP. This is called a label switched path (LSP).

3. An ingress packet enters the edge LSR. The LSR performs a Layer 3 lookup, does any Layer 3 value-added services required, including quality of service (QoS), bandwidth management, and so forth. It then applies a label to the packet based on the information in the forwarding tables.

4. The core LSR reads the labels on each packet on the ingress interface, and based on information in the label, sends the packet out the appropriate egress interface with a new label.

5. The egress edge LSR strips the label and sends the packet to its destination.

MPLS Label Header for Packet Media



MPLS Label Header for Packet Media

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                Label                  | CoS |S|      TTL      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Label = 20 bits	COS/EXP = Class of service, 3 bits
S = Bottom of stack, 1 bit	TTL = Time to live, 8 bits

Can be used over all frame-based encapsulations like Ethernet, 802.3, or PPP (0x8281)

Uses two new Ethertypes/PPP PIDs

- One for unicast (Ethertype 0x8847), one for multicast (Ethertype 0x8848)

Contains everything needed at forwarding time

One word per label

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—6-11

Reserved Labels 0-15

- A label value of 0 represents the "IPv4 explicit null label". It indicates that the label stack must be popped, and the forwarding of the packet then be based on the IPv4 header. This is useful in keeping exp bits safe until they reach the egress router. It is used in MPLS-based QoS.
- A value of 1 represents the "router alert label". The use of this label is analogous to the use of the "Router Alert Option" in IP packets (that is, ping with record route option).
- A value of 2 represents the "IPv6 explicit null label". It indicates that the label stack must be popped, and packet forwarding must then be based on the IPv6 header.
- A value of 3 represents the "implicit null label". This is a label that an LSR may assign and distribute, but which never actually appears in the encapsulation. It indicates the LSR pops the top label from the stack and forwards the rest of the packet (labeled or unlabeled) through the outgoing interface (as per the entry in LFIB). Although this value may never appear in the encapsulation, it needs to be specified in the LDP, so a value is reserved.

The three EXP bits carry the original IP precedence value.

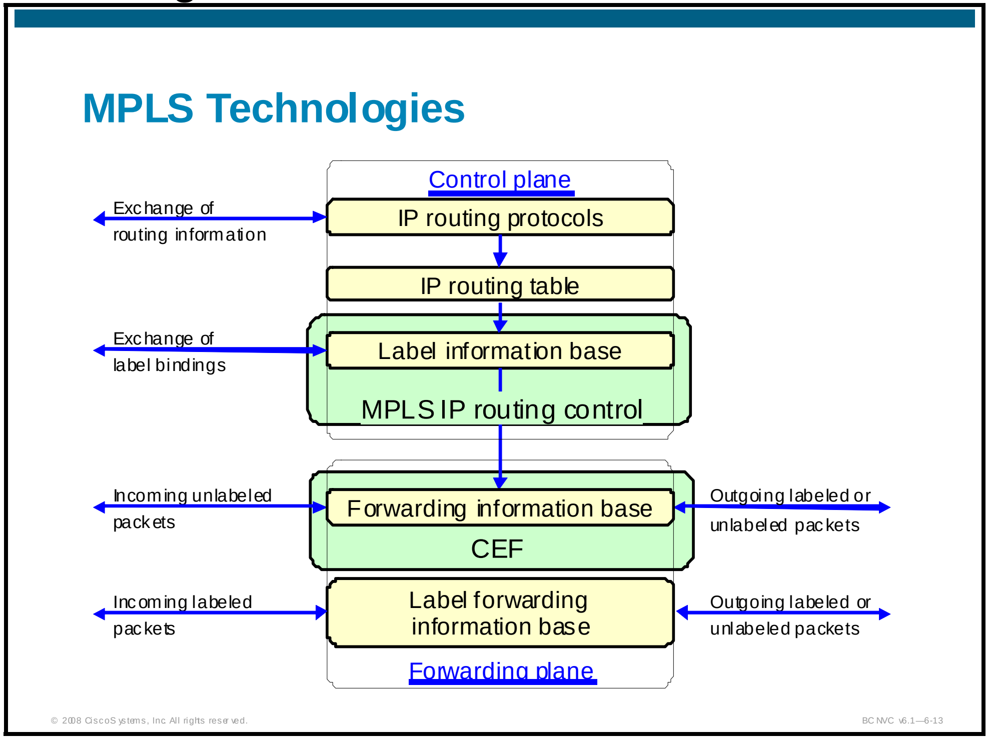
The single "S" bit indicates that this is the bottom label in the stack. MPLS has the ability to use a label stack (more than one label assigned to a packet – for VPNs and traffic engineering). If this 'S' bit is set to 1 then it indicates that this is the bottom of the stack.

Note Label stacks are discussed in later in this course.

The time-to-live (TTL) field is used to prevent indefinite looping of packets, just as it is used in a normal IP packet.

What Are the MPLS Technologies and How Do I Deploy Them?

MPLS Technologies



MPLS Technologies

Control plane
Exchange of routing information
IP routing protocols
IP routing table
Exchange of label bindings
Label information base
MPLS IP routing control
Incoming unlabeled packets
Forwarding information base
CEF
Outgoing labeled or unlabeled packets
Incoming labeled packets
Label forwarding information base
Outgoing labeled or unlabeled packets
Forwarding plane

BCNVC v6.1—6-13

Essential to MPLS is the notion of binding between a label and network layer routes. The control component creates label bindings, stored in the Label Information Base (LIB), and then distributes the label-binding information among LSRs using an LDP.

To support destination-based routing with MPLS, an LSR participates in routing protocols and constructs its LFIB by using the information that it receives from these protocols. In this way, it operates much like a router. A label binding associates a destination subnet to a locally significant label. Labels are locally significant because they are replaced at each hop.

An LFIB contains label data consisting of an incoming label, an outgoing label, an outgoing interface, and Layer 2 adjacency information.

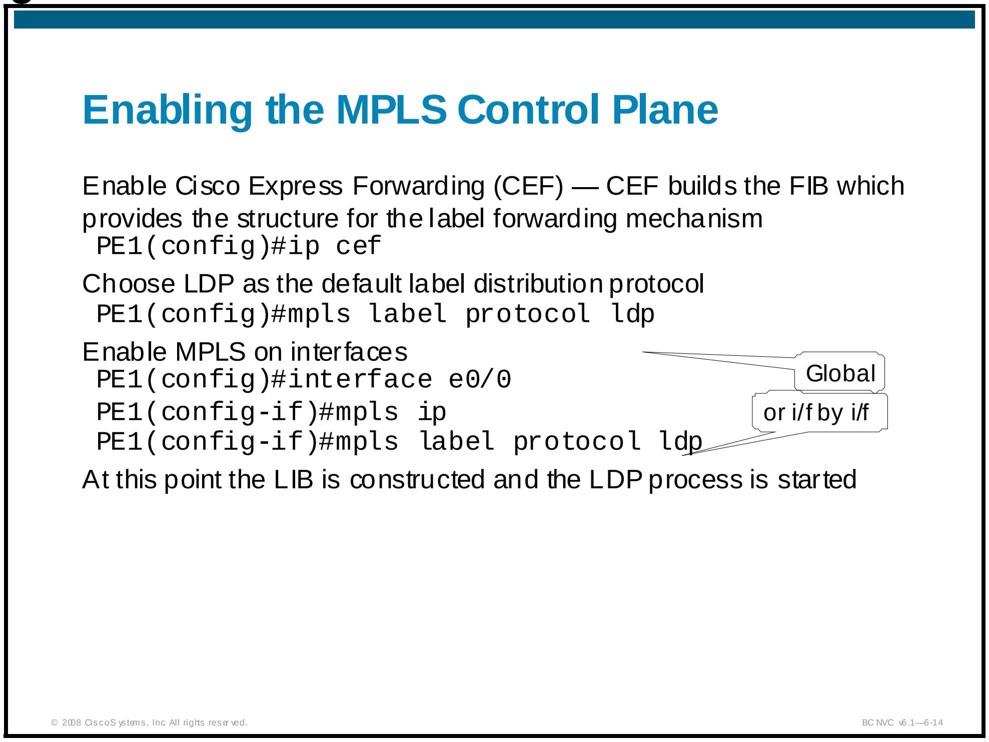
- LFIB is indexed by incoming label
- LFIB could be either per LSR or per interface

Cisco IOS label forwarding code is based on Cisco Express Forwarding (CEF)

- Maintenance of label rewrite structures in LFIB
- Recursive route resolution
- IP to label switching (label imposition) path

An LSR must distribute and use labels for LSR peers to correctly forward a frame. LSRs distribute labels using an LDP. Whenever an LSR discovers a neighbor LSR, the two establish a TCP connection to transfer label bindings.

Enabling the MPLS Control Plane



Enabling the MPLS Control Plane

Enable Cisco Express Forwarding (CEF) — CEF builds the FIB which provides the structure for the label forwarding mechanism

```
 PE1(config)#ip cef
```

Choose LDP as the default label distribution protocol

```
 PE1(config)#mpls label protocol ldp
```

Enable MPLS on interfaces

```
 PE1(config)#interface e0/0
 PE1(config-if)#mpls ip
 PE1(config-if)#mpls label protocol ldp
```

At this point the LIB is constructed and the LDP process is started

BCNVC v6.1—6-14

The label switching control plane refers to the mechanisms used to create and support the label switching functions along a label switched path (LSP).

When configuring an MPLS network there are three tables that are imperative for MPLS operation:

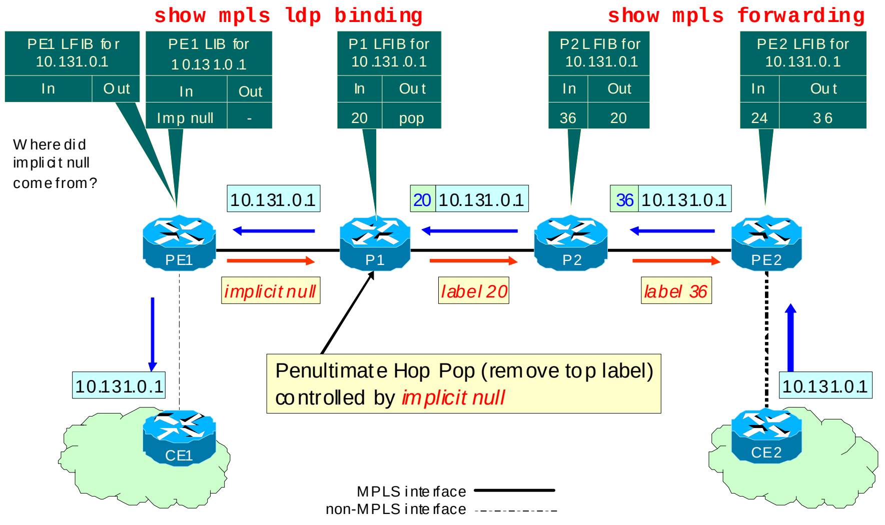
- Forwarding information base (CEF)
- Label information base (LIB / TIB)
- Label forwarding base (LFIB/TFIB)

Following is the control plane process that occurs when MPLS is enabled:

1. Build the LIB
2. Bind local labels to prefixes
3. Discover LDP neighbors
4. Establish LDP sessions
5. Advertise labels to and receive labels from remote LDP neighbors
6. Bind remote labels to prefixes
7. Update LIB
8. Update LFIB and FIB with labels from next hops

Control and Forwarding Planes in Action

Control and Forwarding Planes in Action



`show mpls ldp binding`

`show mpls forwarding`

PE1 LFIB for 10.131.0.1

In	Out

PE1 LIB for 10.131.0.1

In	Out
Imp null	-

P1 LFIB for 10.131.0.1

In	Out
20	pop

P2 LFIB for 10.131.0.1

In	Out
36	20

PE2 LFIB for 10.131.0.1

In	Out
24	36

Where did implicit null come from?

10.131.0.1

20 10.131.0.1

36 10.131.0.1

PE1

P1

P2

PE2

implicit null

label 20

label 36

10.131.0.1

10.131.0.1

Penultimate Hop Pop (remove top label) controlled by *implicit null*

CE1

CE2

MPLS interface

non-MPLS interface

© 2008 Cisco Systems, Inc. All rights reserved.

BCMVC v6.1—6-27

LSRs use LDP to exchange an IP prefix to label bindings. A LIB stores these bindings, which builds the FIB entries in ingress edge-LSRs as well as LFIB in all MPLS nodes.

The **mpls ip interface configuration** command enables MPLS on a frame-mode interface. In Cisco IOS software supporting LDP, the desired LDP must be selected using the **mpls label-distribution** command. These commands start LDP on the specified interface. LDP finds other LSRs attached to the same subnet through LDP hello packets sent as UDP packets to broadcast or multicast IP addresses. When the neighboring LSRs are discovered, a LDP session is established using TCP as the transport protocol to ensure the reliable delivery of label mappings.

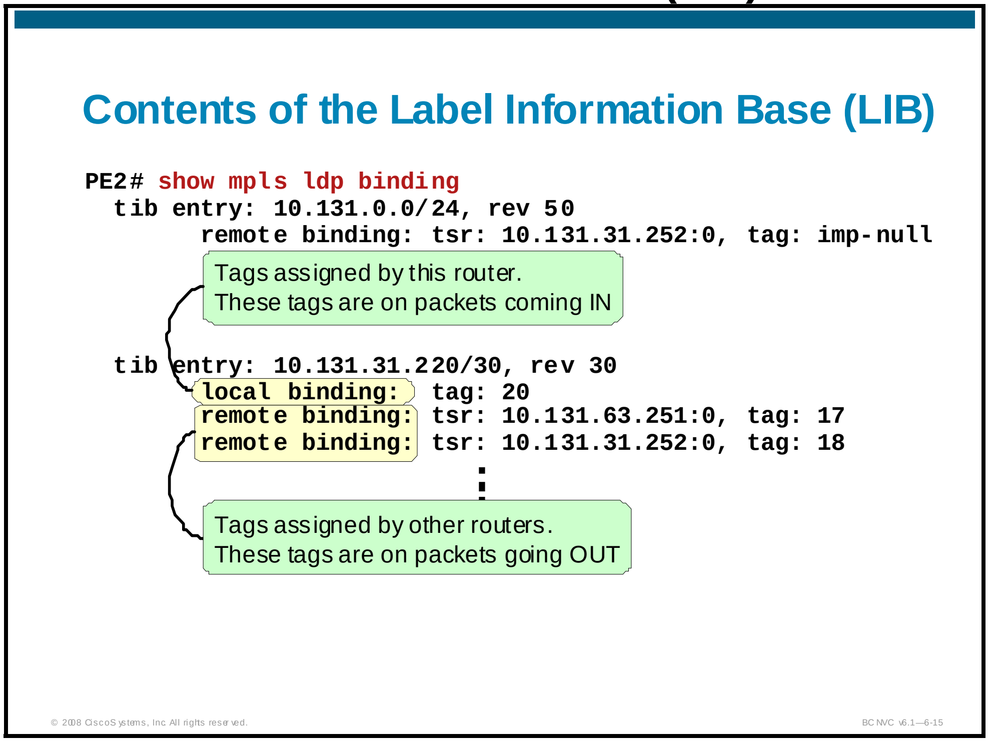
The Cisco IOS implementation of LSRs on frame-mode interfaces assigns labels to IP prefixes as soon as they appear in the routing table, even though the LSR may not have received a corresponding label from its downstream neighbor, because it can always perform a Layer 3 lookup if needed. The router works in independent control allocation mode, as opposed to ordered control allocation, where a device assigns labels only to those prefixes where a downstream label already exists in the LIB.

When running MPLS over frame-mode interfaces, a Cisco router immediately propagates allocated labels to its LDP neighbors. This distribution method is called unsolicited downstream distribution, as opposed to downstream on demand distribution, where the upstream routers explicitly ask the downstream routers for specific labels.

A Cisco router acting as an LSR stores all label mappings received from its LDP neighbors. This storage method is called liberal retention mode as opposed to conservative retention mode where the LSR stores only labels received from its next hop downstream routers. The liberal retention mode uses more memory but enables instantaneous LDP convergence following the routing protocol convergence after a failure in the network.

After the LSRs in an MPLS network have exchanged label mappings, the ingress LSR can label the incoming data packets. The ingress LSR inserts a label stack header between the Layer 2 header and the IP header. For unicast-destination-only IP routing, the label stack header usually contains only one label, but MPLS also supports stacked labels used by other MPLS applications, such as traffic engineering or VPNs. The labeled packets are distinguished from the unlabeled IP packets by using different Ethertype codes on LAN media and a different PPP field value.

Contents of the Label Information Base (LIB)



Contents of the Label Information Base (LIB)

```
PE2# show mpls ldp binding
  tib entry: 10.131.0.0/24, rev 50
        remote binding: tsr: 10.131.31.252:0, tag: imp-null

  tib entry: 10.131.31.220/30, rev 30
        local binding:  tag: 20
        remote binding: tsr: 10.131.63.251:0, tag: 17
        remote binding: tsr: 10.131.31.252:0, tag: 18
                                 :
```

Tags assigned by this router.
These tags are on packets coming IN

Tags assigned by other routers.
These tags are on packets going OUT

Upon startup of MPLS (enabled by the commands from the previous page), the MPLS process binds labels to all routes (except BGP routes) in the FIB.

Once LDP neighbor sessions are established, labels are exchanged and you will see local and remote label bindings.

The LIB holds all labels that are sent and received. When you examine label allocation and distribution you notice that neighbors send labels even if they are not considered when routing packets.

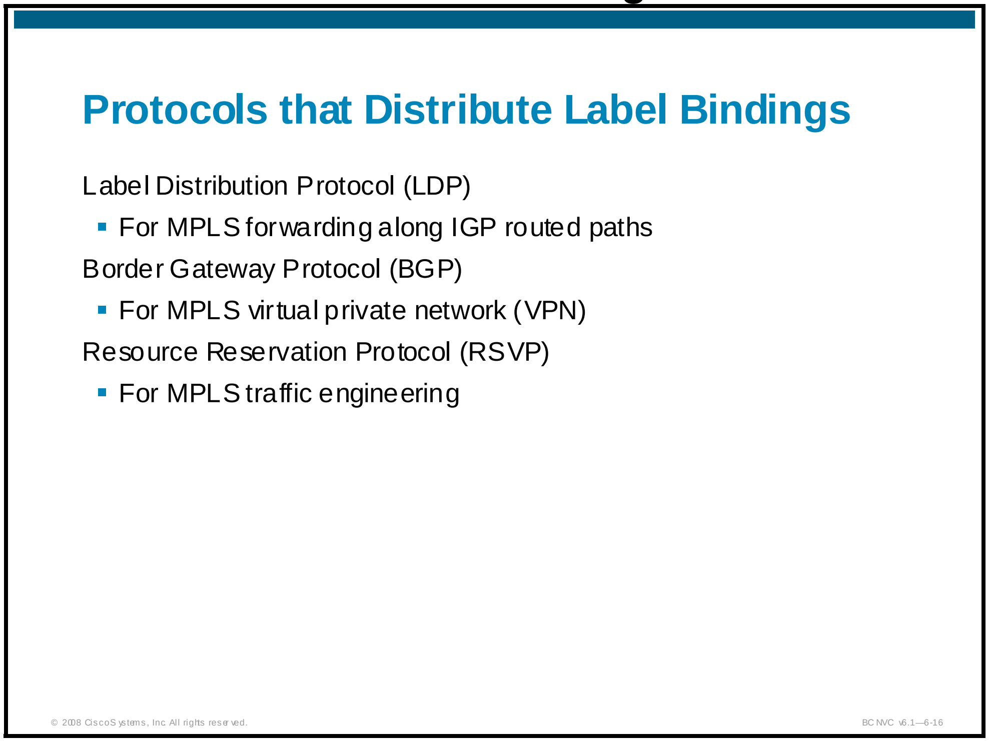
A router stores a complete list of labels for a prefix. In the event of failure, the router already has alternate labels, making convergence much faster.

The copy of the LIB above indicates that the LIB has three entries for network 10.131.31.220/30.

- The first of these entries displays the local binding for the network. This is the tag (20) that is distributed to all neighbors.
- The second entry indicates the tag value from 10.131.63.251 with a label of 17.
- The third entry displays the tag value from 10.131.31.252 with a tag of 18.

This explains how the LIB is actually built; rules governing exchange of labels and label allocation are discussed later in this chapter.

Protocols that Distribute Label Bindings



Protocols that Distribute Label Bindings

Label Distribution Protocol (LDP)

- For MPLS forwarding along IGP routed paths

Border Gateway Protocol (BGP)

- For MPLS virtual private network (VPN)

Resource Reservation Protocol (RSVP)

- For MPLS traffic engineering

Each LSR in a network makes an independent local decision regarding which label value to use to represent a Forwarding Equivalence Class (FEC). A FEC is a group of packets that are forwarded in the same manner, over the same path, and with the same forwarding treatment. This association is known as label binding. Each LSR informs its neighbors of the label bindings it has made. This awareness of label bindings by neighboring routers and switches is facilitated by the following protocols:

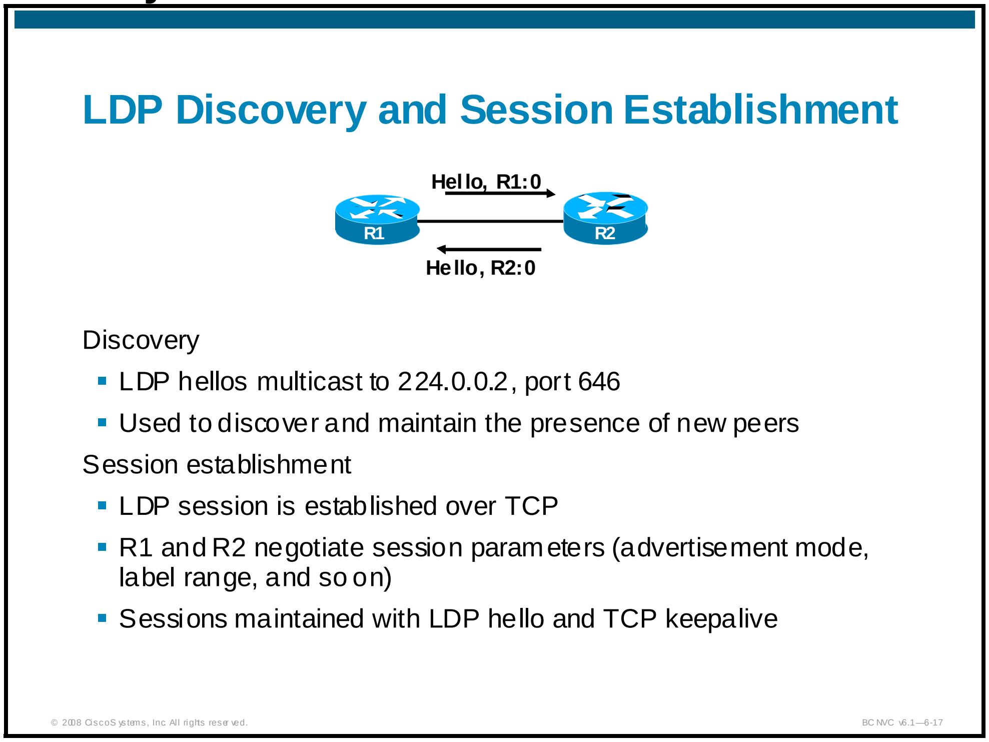
- LDP—Supports MPLS forwarding along normally routed paths
- Resource Reservation Protocol (RSVP)—Supports MPLS traffic engineering
- Border Gateway Protocol (BGP)—Supports MPLS VPNs

MPLS LDP provides a standard methodology for hop-by-hop (dynamic label) distribution in an MPLS network by assigning labels to routes that have been chosen by the underlying Interior Gateway Protocol (IGP). The resulting labeled paths, called LSPs, forward label traffic across an MPLS backbone to particular destinations. These capabilities enable service providers to implement Cisco MPLS-based IP VPNs and IP+ATM services across multivendor MPLS networks.

LDP allows LSRs to request, distribute, and release label prefix binding information to peer routers in a network. LDP enables LSRs to discover potential peers and establish LDP sessions with those peers to exchange label binding information.

An LDP label binding is an association between a destination prefix and a label. The label used in a label binding is allocated from a set of possible labels called a label space.

LDP Discovery and Session Establishment



LDP Discovery and Session Establishment

Hello, R1:0
R1
R2
Hello, R2:0

Discovery

- LDP hellos multicast to 224.0.0.2, port 646
- Used to discover and maintain the presence of new peers

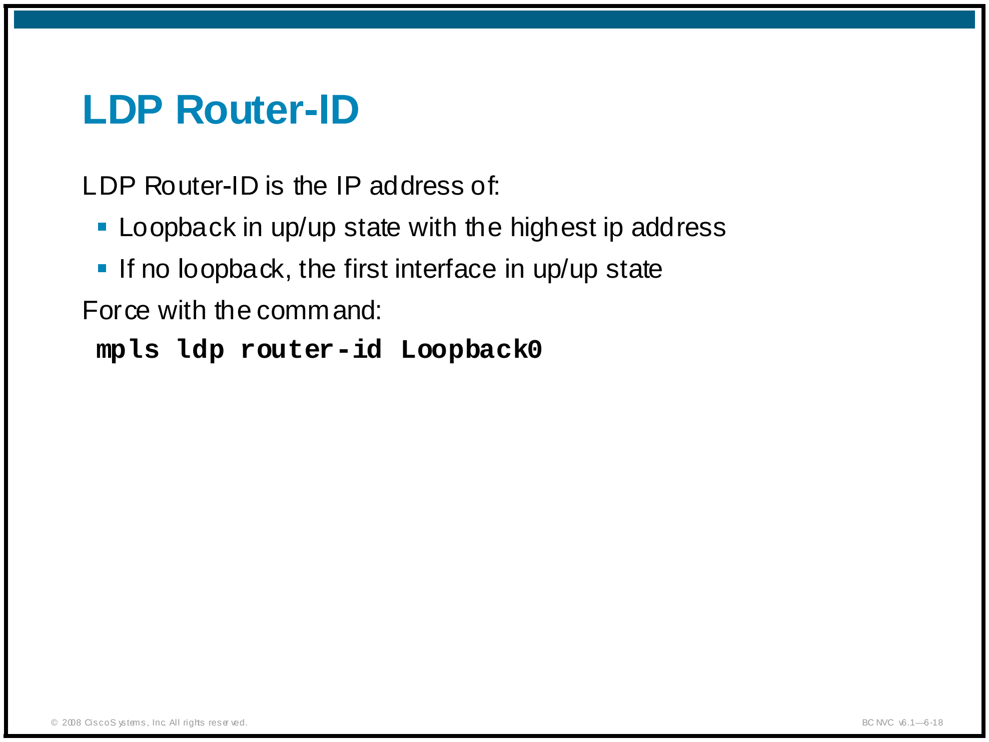
Session establishment

- LDP session is established over TCP
- R1 and R2 negotiate session parameters (advertisement mode, label range, and so on)
- Sessions maintained with LDP hello and TCP keepalive

The discovery mechanisms differ in that the IP destination address used for link Hellos differs; TDP uses broadcast confined to the subnet, LDP uses the “all routers on this subnet” multicast group.

LDP specifies a backoff mechanism to throttle session establishment attempts between potential LDP peers that cannot agree on session parameters.

LDP Router-ID



LDP Router-ID

LDP Router-ID is the IP address of:

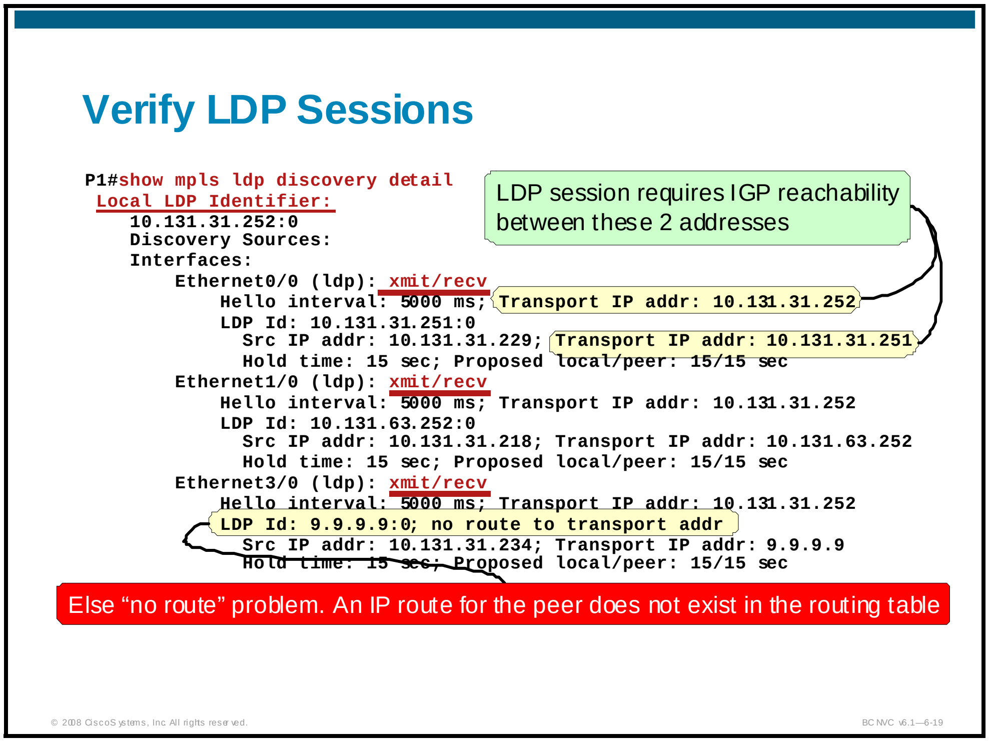
- Loopback in up/up state with the highest ip address
- If no loopback, the first interface in up/up state

Force with the command:

```
mpls ldp router-id Loopback0
```

Like OSPF and BGP, LDP uses the highest numbered loopback address for the LSR (TSR) identifier. If there are no loopback addresses LDP uses the highest IP address defined within the device. This can be manually configured and is discussed later in the course.

Verify LDP Sessions



Verify LDP Sessions

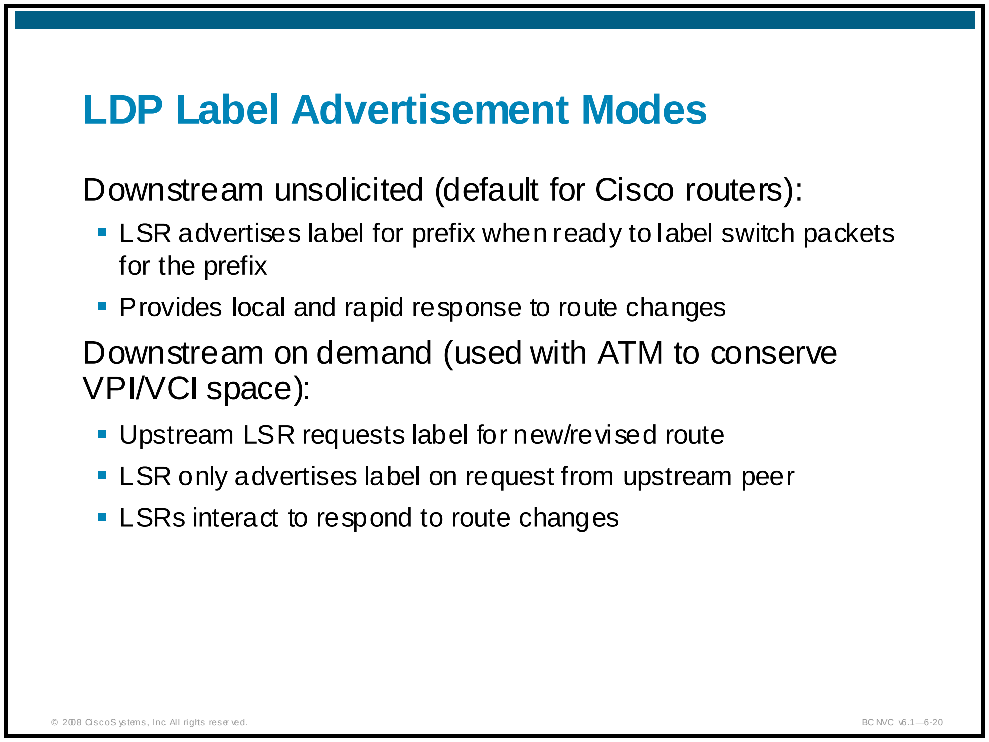
```
P1#show mpls ldp discovery detail
 Local LDP Identifier:
    10.131.31.252:0
    Discovery Sources:
    Interfaces:
        Ethernet0/0 (ldp): xmit/recv
            Hello interval: 5000 ms; Transport IP addr: 10.131.31.252
            LDP Id: 10.131.31.251:0
              Src IP addr: 10.131.31.229; Transport IP addr: 10.131.31.251
              Hold time: 15 sec; Proposed local/peer: 15/15 sec
        Ethernet1/0 (ldp): xmit/recv
            Hello interval: 5000 ms; Transport IP addr: 10.131.31.252
            LDP Id: 10.131.63.252:0
              Src IP addr: 10.131.31.218; Transport IP addr: 10.131.63.252
              Hold time: 15 sec; Proposed local/peer: 15/15 sec
        Ethernet3/0 (ldp): xmit/recv
            Hello interval: 5000 ms; Transport IP addr: 10.131.31.252
            LDP Id: 9.9.9.9:0; no route to transport addr
              Src IP addr: 10.131.31.234; Transport IP addr: 9.9.9.9
              Hold time: 15 sec; Proposed local/peer: 15/15 sec
```

LDP session requires IGP reachability between these 2 addresses

Else “no route” problem. An IP route for the peer does not exist in the routing table

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—6-19

LDP Label Advertisement Modes



LDP Label Advertisement Modes

Downstream unsolicited (default for Cisco routers):

- LSR advertises label for prefix when ready to label switch packets for the prefix
- Provides local and rapid response to route changes

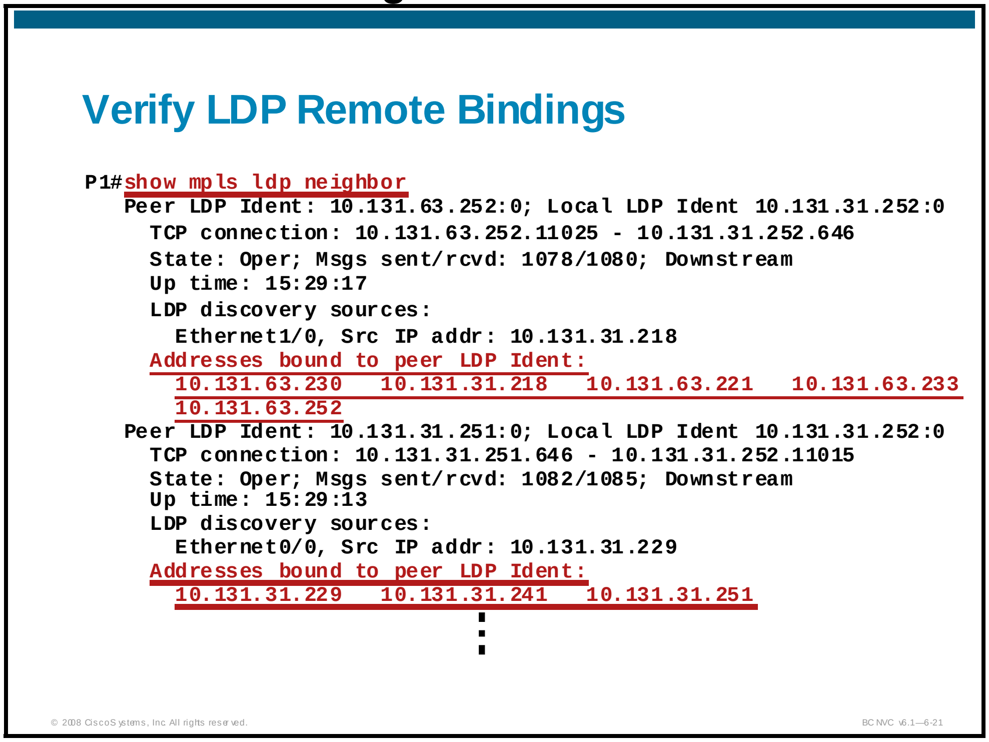
Downstream on demand (used with ATM to conserve VPI/VCI space):

- Upstream LSR requests label for new/revised route
- LSR only advertises label on request from upstream peer
- LSRs interact to respond to route changes

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—6-20

TDP and LDP both support “downstream unsolicited” and “downstream on demand” label distribution. The patterns of message exchange for these label distribution methods are identical for both protocols. The label distribution mechanisms differ in that the TDP bind message can carry multiple label bindings, whereas the LDP label message can carry only a single label binding.

Verify LDP Remote Bindings

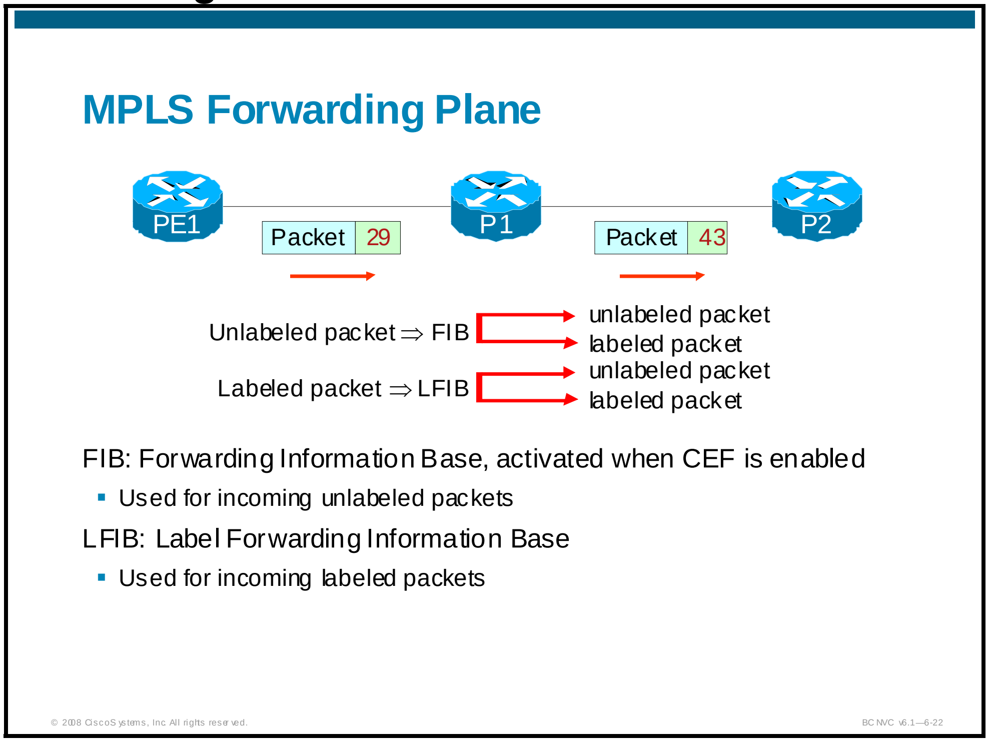


Verify LDP Remote Bindings

```
P1#show mpls ldp neighbor
   Peer LDP Ident: 10.131.63.252:0; Local LDP Ident 10.131.31.252:0
     TCP connection: 10.131.63.252.11025 - 10.131.31.252.646
     State: Oper; Msgs sent/rcvd: 1078/1080; Downstream
     Up time: 15:29:17
     LDP discovery sources:
       Ethernet1/0, Src IP addr: 10.131.31.218
     Addresses bound to peer LDP Ident:
       10.131.63.230   10.131.31.218   10.131.63.221   10.131.63.233
       10.131.63.252
   Peer LDP Ident: 10.131.31.251:0; Local LDP Ident 10.131.31.252:0
     TCP connection: 10.131.31.251.646 - 10.131.31.252.11015
     State: Oper; Msgs sent/rcvd: 1082/1085; Downstream
     Up time: 15:29:13
     LDP discovery sources:
       Ethernet0/0, Src IP addr: 10.131.31.229
     Addresses bound to peer LDP Ident:
       10.131.31.229   10.131.31.241   10.131.31.251
                                 .
                                 .
                                 .
```

BCNVC v6.1—6-21

MPLS Forwarding Plane



MPLS Forwarding Plane

FIB: Forwarding Information Base, activated when CEF is enabled

- Used for incoming unlabeled packets

LFIB: Label Forwarding Information Base

- Used for incoming labeled packets

BCMVC v6.1—6-22

The label switching forwarding plane consists of two primary data structures: the FIB and the LFIB.

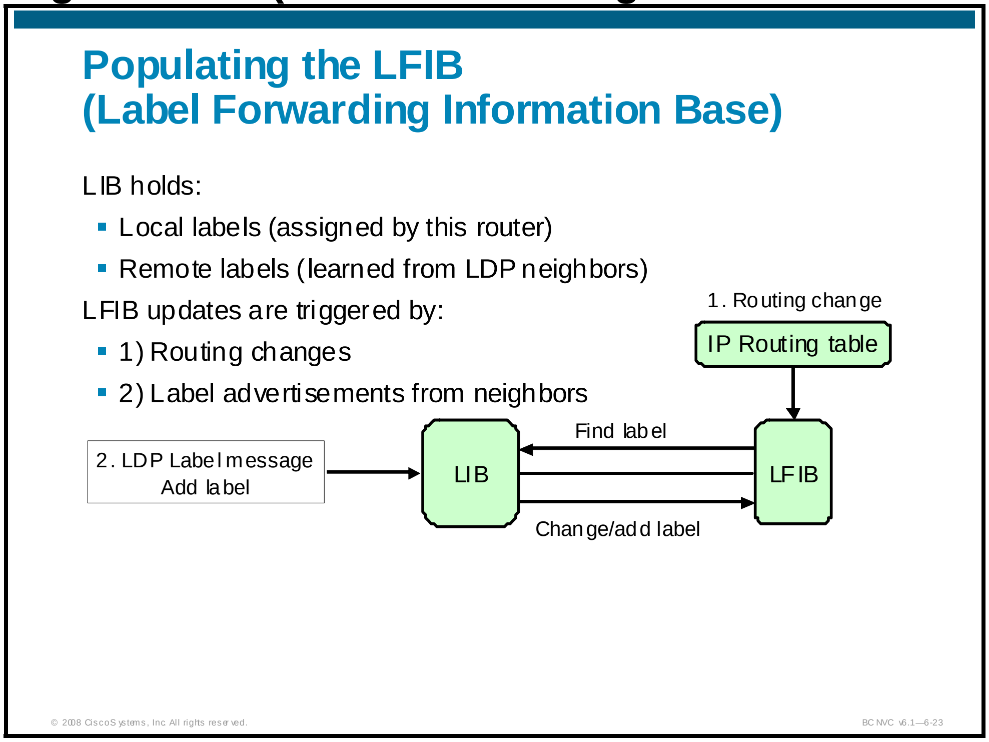
For incoming unlabeled packets:

- The FIB is created when CEF is enabled on a router. CEF or distributed CEF (dCEF) is required for MPLS. It is a fast switching technique that is topology driven and useful in very large networks.
- The FIB may be used to forward the outbound packet as either labeled or unlabeled.
- When a packet has a label imposed, the FIB obtains the appropriate label information and applies it to the packet.

For incoming labeled packets:

- The LFIB is used when a labeled packet is received. The label is used as an index into the LFIB to determine the appropriate outgoing label that is used.
- The outbound packet may be forwarded as either labeled or unlabeled. Unlabeled means that the MPLS type code is removed from the packet, any labels are removed, and the packet is forwarded in its native format.

Populating the LFIB (Label Forwarding Information Base)



Populating the LFIB (Label Forwarding Information Base)

LIB holds:

- Local labels (assigned by this router)
- Remote labels (learned from LDP neighbors)

LFIB updates are triggered by:

- 1) Routing changes
- 2) Label advertisements from neighbors

1. Routing change

IP Routing table

2. LDP Label message
Add label

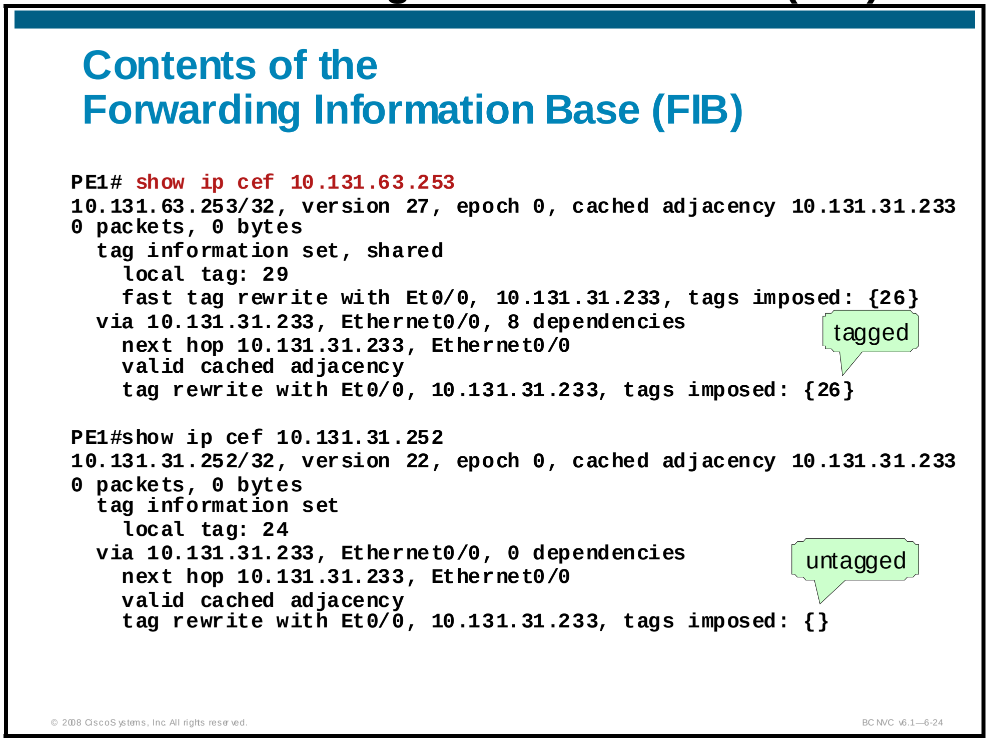
LIB

Find label

Change/add label

LFIB

Contents of the Forwarding Information Base (FIB)



Contents of the Forwarding Information Base (FIB)

```
PE1# show ip cef 10.131.63.253
10.131.63.253/32, version 27, epoch 0, cached adjacency 10.131.31.233
0 packets, 0 bytes
  tag information set, shared
    local tag: 29
    fast tag rewrite with Et0/0, 10.131.31.233, tags imposed: {26}
  via 10.131.31.233, Ethernet0/0, 8 dependencies
    next hop 10.131.31.233, Ethernet0/0
    valid cached adjacency
    tag rewrite with Et0/0, 10.131.31.233, tags imposed: {26}

PE1#show ip cef 10.131.31.252
10.131.31.252/32, version 22, epoch 0, cached adjacency 10.131.31.233
0 packets, 0 bytes
  tag information set
    local tag: 24
  via 10.131.31.233, Ethernet0/0, 0 dependencies
    next hop 10.131.31.233, Ethernet0/0
    valid cached adjacency
    tag rewrite with Et0/0, 10.131.31.233, tags imposed: {}
```

BCMVC v6.1—6-24

The FIB is populated by CEF. The command to see this database is **show ip cef**.

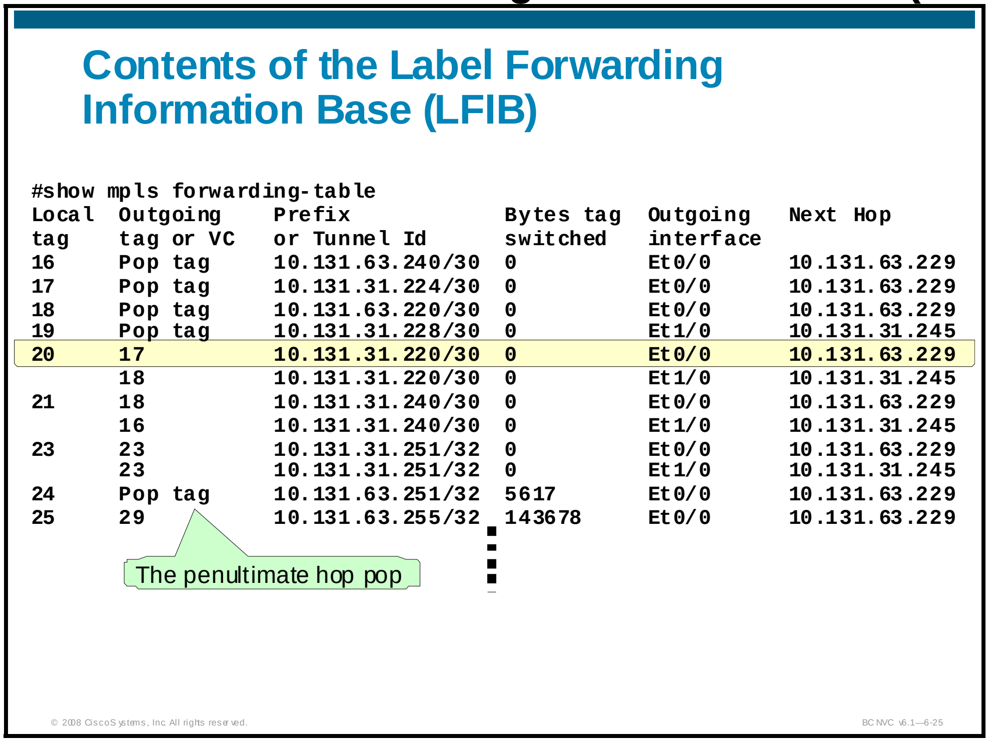
The CEF database shows the prefix for the destination network, the next hop, and the interface that the packet is sent out.

This “next hop label forwarding entry” is used when forwarding a labeled packet. It contains the following information:

- Next hop of the packet
- Operation to perform on the label stack of the packet; could be one of the following:
  - Replace the label at the top of the label stack with a specified new label
  - Pop the label stack
  - Replace the label at the top of the label stack with a specified new label, and then push one or more specified new labels onto the label stack.
- It may also contain:
  - Data link encapsulation to use when transmitting the packet
  - Way to encode the label stack when transmitting the packet
  - Any other information needed in order to properly dispose of the packet

If the next hop of the packet is the current LSR, then the label stack operation MUST be “pop the stack” (remove the label).

Contents of the Label Forwarding Information Base (LFIB)



Contents of the Label Forwarding Information Base (LFIB)

```
#show mpls forwarding-table
Local  Outgoing    Prefix            Bytes tag  Outgoing    Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
16     Pop tag     10.131.63.240/30  0          Et0/0       10.131.63.229
17     Pop tag     10.131.31.224/30  0          Et0/0       10.131.63.229
18     Pop tag     10.131.63.220/30  0          Et0/0       10.131.63.229
19     Pop tag     10.131.31.228/30  0          Et1/0       10.131.31.245
20     17          10.131.31.220/30  0          Et0/0       10.131.63.229
       18          10.131.31.220/30  0          Et1/0       10.131.31.245
21     18          10.131.31.240/30  0          Et0/0       10.131.63.229
       16          10.131.31.240/30  0          Et1/0       10.131.31.245
23     23          10.131.31.251/32  0          Et0/0       10.131.63.229
       23          10.131.31.251/32  0          Et1/0       10.131.31.245
24     Pop tag     10.131.63.251/32  5617       Et0/0       10.131.63.229
25     29          10.131.63.255/32  143678     Et0/0       10.131.63.229
```

The penultimate hop pop

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—6-25

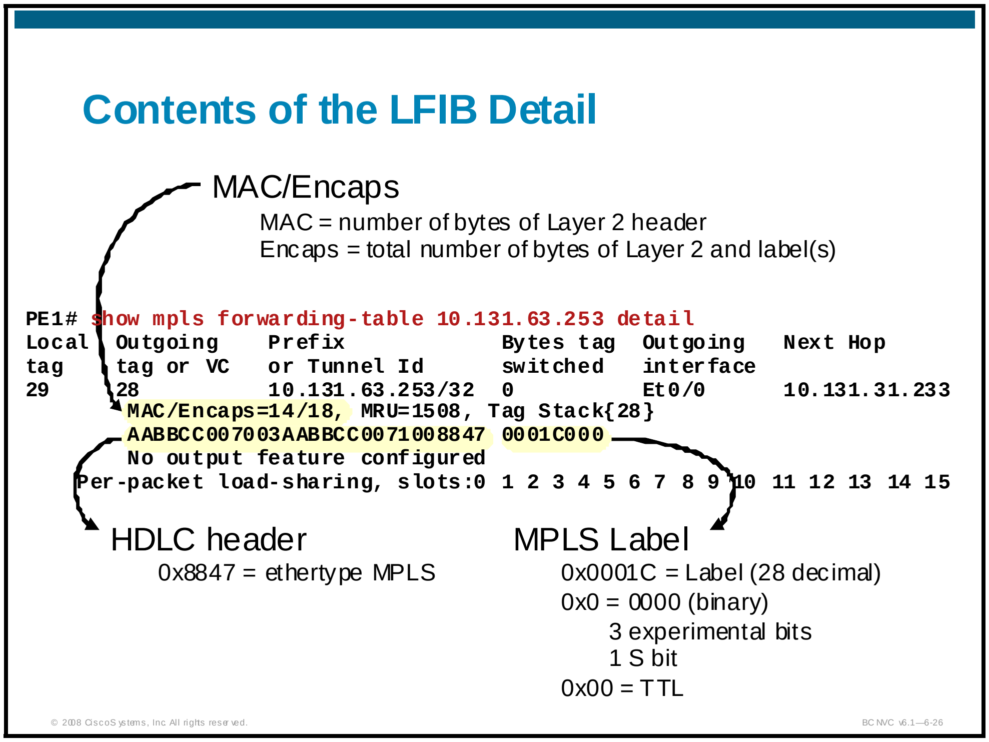
The LFIB contains the tag used for a particular destination prefix.

Notice in the figure above the LFIB for the destination network 10.131.31.220/30.

- The local tag matches the tag specified in the LIB
- The outgoing tag matches the route to Ethernet 0/0

This indicates that any packets (to be label-switched) come into the router with label 20. It changes to a label value of 17 and is sent out interface Ethernet 0/0.

Contents of the LFIB Detail



Contents of the LFIB Detail

MAC/Encaps

MAC = number of bytes of Layer 2 header
Encaps = total number of bytes of Layer 2 and label(s)

```
PE1# show mpls forwarding-table 10.131.63.253 detail
Local  Outgoing    Prefix             Bytes tag  Outgoing    Next Hop
tag    tag or VC   or Tunnel Id       switched   interface
29     28          10.131.63.253/32   0          Et0/0       10.131.31.233
        MAC/Encaps=14/18, MRU=1508, Tag Stack{28}
        AABBCC007003AABBCC0071008847 0001C000
        No output feature configured
    Per-packet load-sharing, slots:0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
```

HDLC header

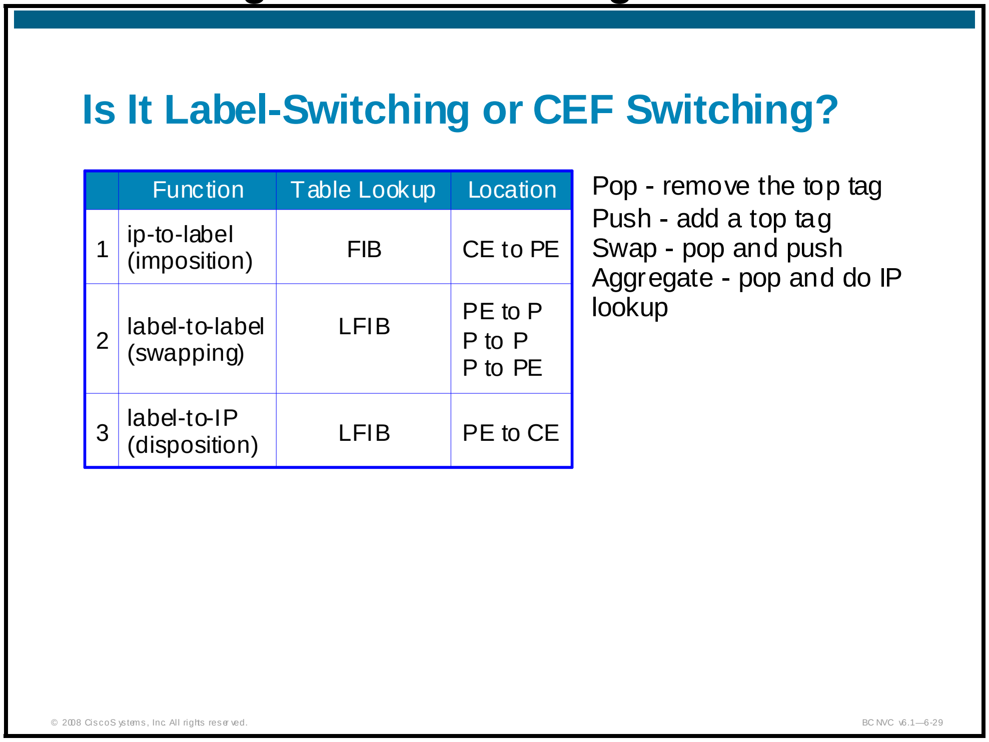
0x8847 = ethertype MPLS

MPLS Label

0x0001C = Label (28 decimal)
0x0 = 0000 (binary)
3 experimental bits
1 S bit
0x00 = TTL

A detailed view of the LFIB provides additional information. Here you can see the encapsulation details.

Is It Label-Switching or CEF Switching?

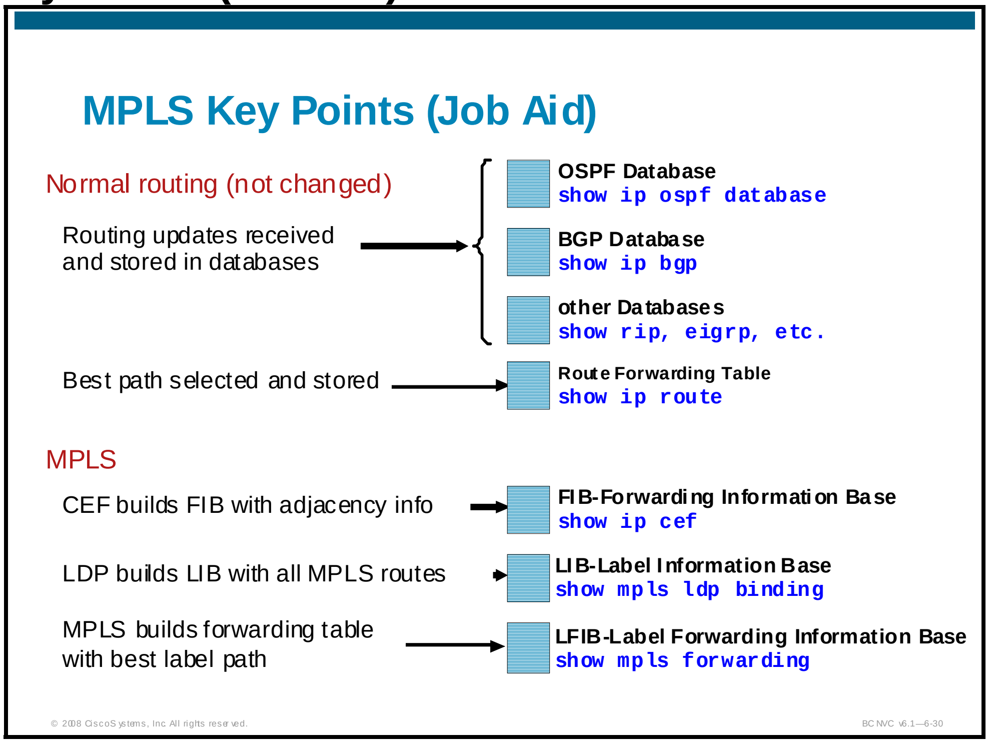


Is It Label-Switching or CEF Switching?

	Function	Table Lookup	Location
1	ip-to-label (imposition)	FIB	CE to PE
2	label-to-label (swapping)	LFIB	PE to P P to P P to PE
3	label-to-IP (disposition)	LFIB	PE to CE

Pop - remove the top tag
Push - add a top tag
Swap - pop and push
Aggregate - pop and do IP lookup

MPLS Key Points (Job Aid)



MPLS Key Points (Job Aid)

Normal routing (not changed)

Routing updates received and stored in databases

OSPF Database
`show ip ospf database`

BGP Database
`show ip bgp`

other Databases
`show rip, eigrp, etc.`

Best path selected and stored

Route Forwarding Table
`show ip route`

MPLS

CEF builds FIB with adjacency info

FIB-Forwarding Information Base
`show ip cef`

LDP builds LIB with all MPLS routes

LIB-Label Information Base
`show mpls ldp binding`

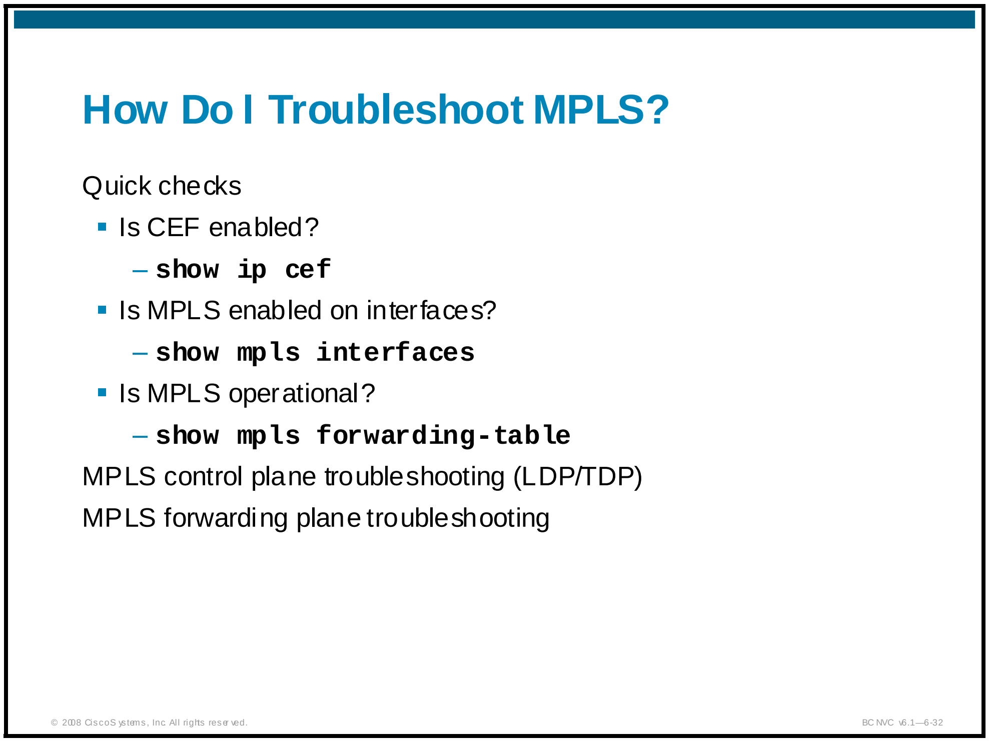
MPLS builds forwarding table with best label path

LFIB-Label Forwarding Information Base
`show mpls forwarding`

BCNVC v6.1—6-30

How Do I Troubleshoot MPLS?

Quick Checks



How Do I Troubleshoot MPLS?

Quick checks

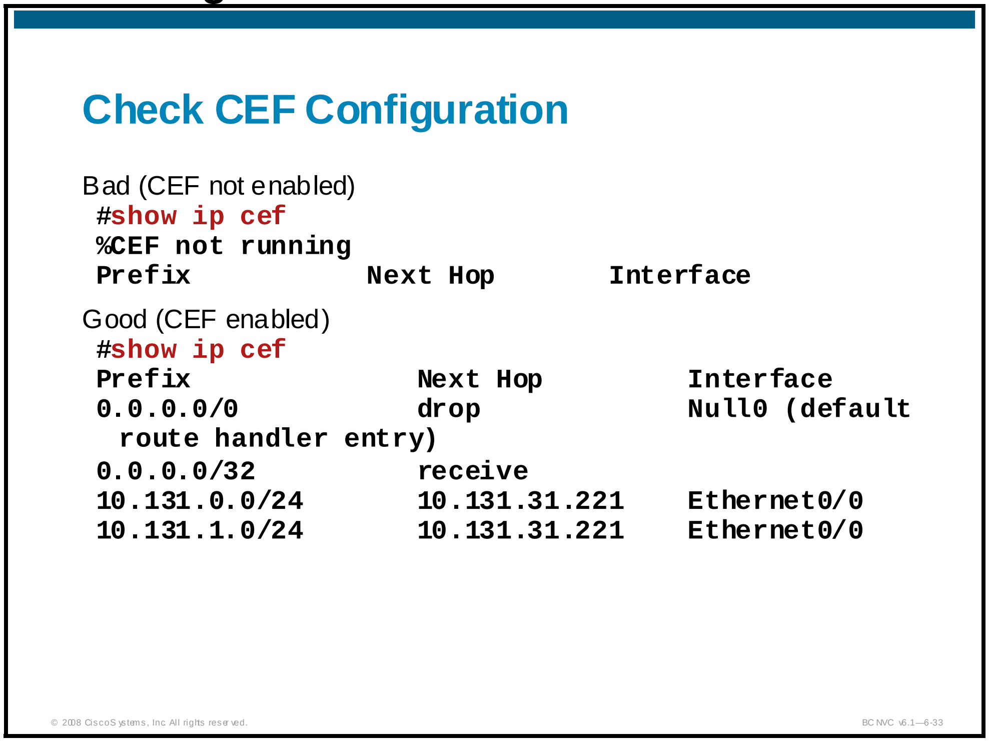
- Is CEF enabled?
  - **`show ip cef`**
- Is MPLS enabled on interfaces?
  - **`show mpls interfaces`**
- Is MPLS operational?
  - **`show mpls forwarding-table`**

MPLS control plane troubleshooting (LDP/TDP)

MPLS forwarding plane troubleshooting

BCNVC v6.1—6-32

Check CEF Configuration



Check CEF Configuration

Bad (CEF not enabled)

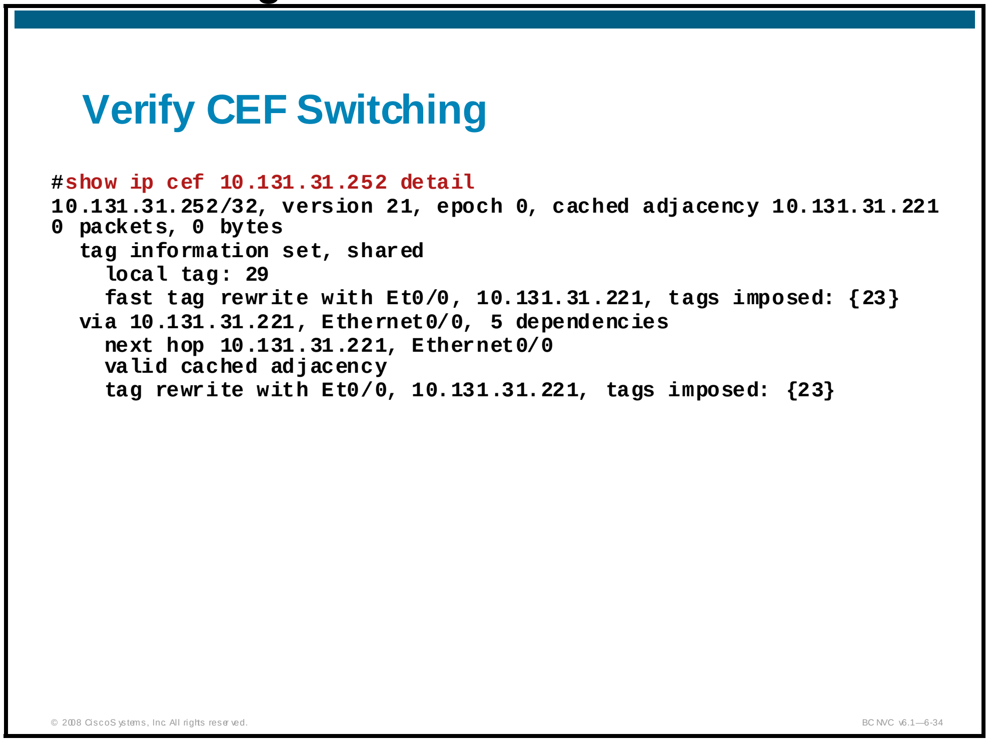
```
#show ip cef
%CEF not running
Prefix            Next Hop         Interface
```

Good (CEF enabled)

```
#show ip cef
Prefix                Next Hop          Interface
0.0.0.0/0             drop              Null0 (default
  route handler entry)
0.0.0.0/32            receive
10.131.0.0/24         10.131.31.221     Ethernet0/0
10.131.1.0/24         10.131.31.221     Ethernet0/0
```

BCNVC v6.1—6-33

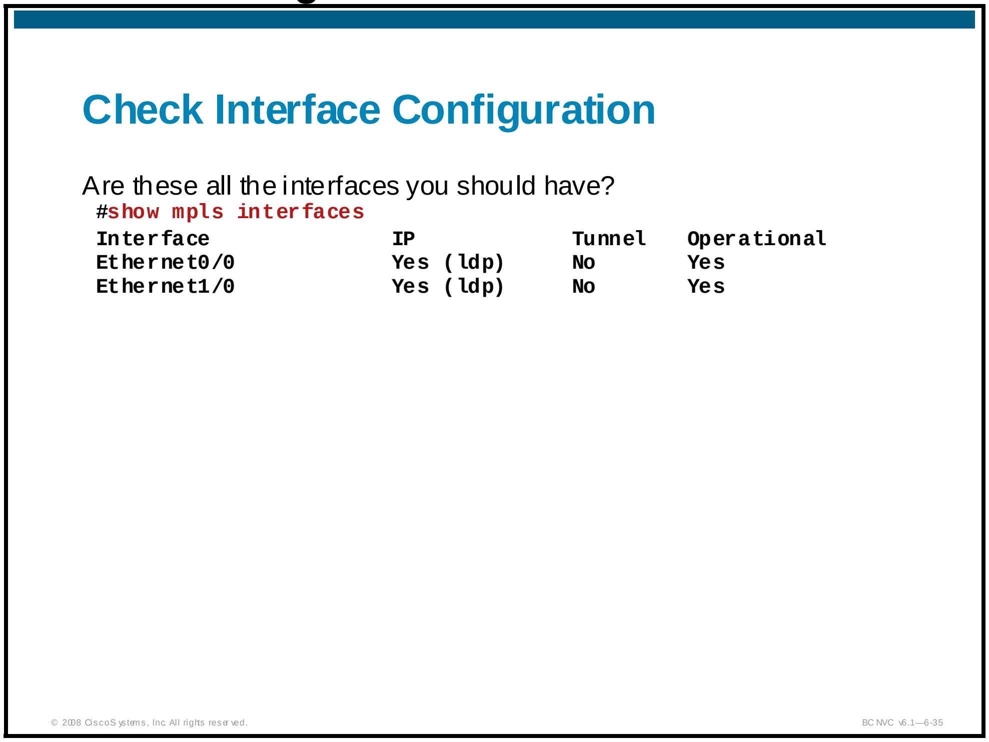
Verify CEF Switching



Verify CEF Switching

```
#show ip cef 10.131.31.252 detail
10.131.31.252/32, version 21, epoch 0, cached adjacency 10.131.31.221
0 packets, 0 bytes
  tag information set, shared
    local tag: 29
    fast tag rewrite with Et0/0, 10.131.31.221, tags imposed: {23}
  via 10.131.31.221, Ethernet0/0, 5 dependencies
    next hop 10.131.31.221, Ethernet0/0
    valid cached adjacency
    tag rewrite with Et0/0, 10.131.31.221, tags imposed: {23}
```

Check Interface Configuration



Check Interface Configuration

Are these all the interfaces you should have?

```
#show mpls interfaces
Interface              IP            Tunnel   Operational
Ethernet0/0            Yes (ldp)     No       Yes
Ethernet1/0            Yes (ldp)     No       Yes
```

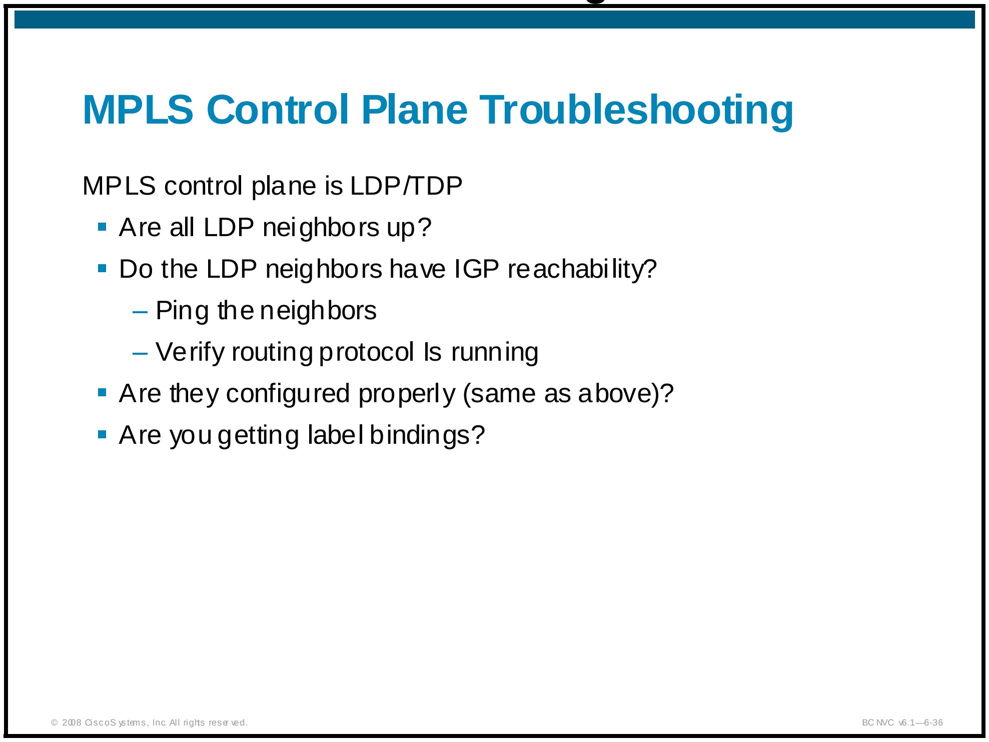
© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—6-35

Make sure that MPLS is enabled on appropriate interfaces and that a label distribution protocol is running on the requested interfaces using the **show running-config** command.

Also use the **show mpls interfaces** command. Field descriptions are:

- IP Field—Shows that MPLS IP is configured for an interface. LDP appears in parenthesis to the right of the IP status.
- Tunnel Field—Indicates the capability of traffic engineering on the interface.
- Operational Field—”Yes” if labeled packets can be sent over this interface. Labeled packets can be sent over an interface if an MPLS protocol is configured on the interface and required Layer 2 negotiations have occurred.

MPLS Control Plane Troubleshooting

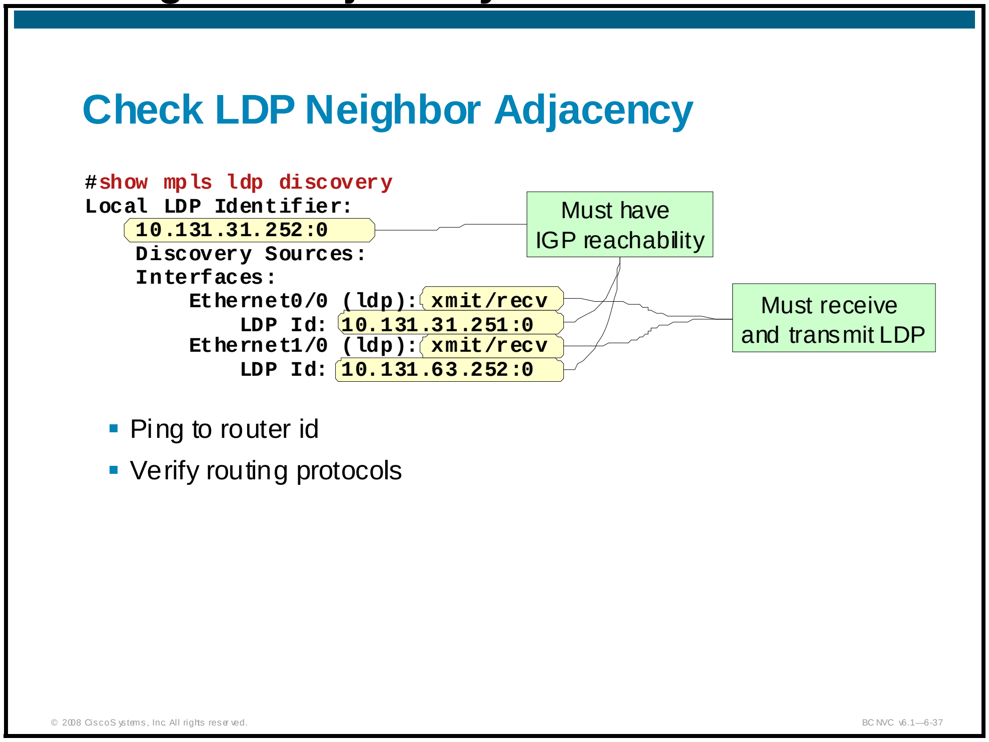


MPLS Control Plane Troubleshooting

MPLS control plane is LDP/TDP

- Are all LDP neighbors up?
- Do the LDP neighbors have IGP reachability?
  - Ping the neighbors
  - Verify routing protocol Is running
- Are they configured properly (same as above)?
- Are you getting label bindings?

Check LDP Neighbor Adjacency



Check LDP Neighbor Adjacency

```
#show mpls ldp discovery
Local LDP Identifier:
    10.131.31.252:0
    Discovery Sources:
    Interfaces:
        Ethernet0/0 (ldp): xmit/recv
            LDP Id: 10.131.31.251:0
        Ethernet1/0 (ldp): xmit/recv
            LDP Id: 10.131.63.252:0
```

Must have IGP reachability

Must receive and transmit LDP

- Ping to router id
- Verify routing protocols

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—6-37

The **show mpls ldp discovery** command displays the discovered neighbors.

If any of the presumed neighbors is missing and cannot be pinged, a connectivity problem exists and the label distribution protocol cannot run. If label distribution protocol is running correctly, it should assign one label per forwarding equivalent class.

Note If the router ID for the label distribution protocol cannot be reached from the global routing table, the neighbor relationship is not established.

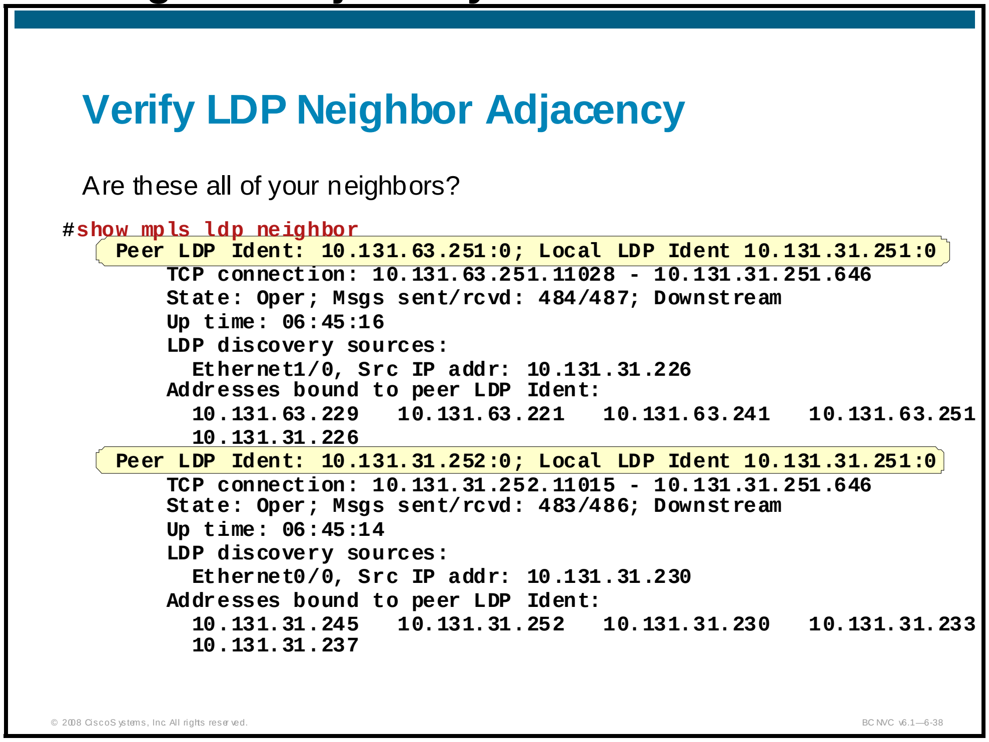
1. Enter the **show mpls ldp discovery** command to determine the identifier of the LSR
2. Determine the LDP identifier of the LSRs
3. Check the interfaces field. This field displays the interfaces engaging in LDP discovery activity:

- *xmit* indicates that the interface is transmitting LDP discovery hello packets
- *recv* indicates that the interface is receiving LDP discovery hello packets

If either xmit or recv do not appear, then:

- Make sure the interfaces are configured for LDP at both ends
- There is not an access list block

Verify LDP Neighbor Adjacency



Verify LDP Neighbor Adjacency

Are these all of your neighbors?

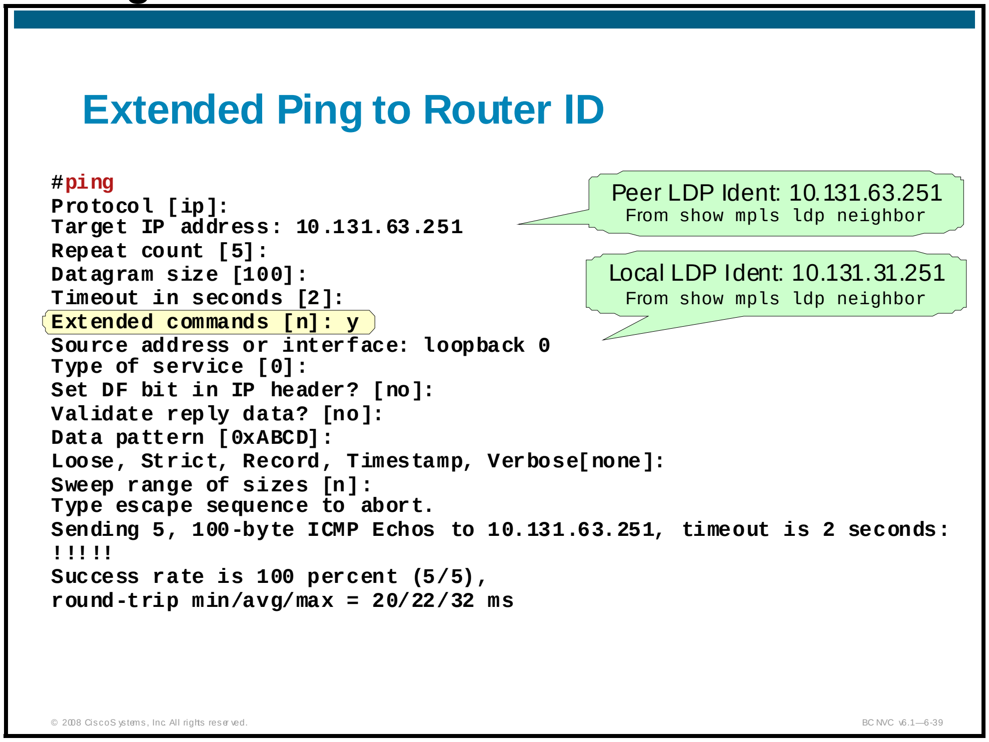
```
#show mpls ldp neighbor
  Peer LDP Ident: 10.131.63.251:0; Local LDP Ident 10.131.31.251:0
        TCP connection: 10.131.63.251.11028 - 10.131.31.251.646
        State: Oper; Msgs sent/rcvd: 484/487; Downstream
        Up time: 06:45:16
        LDP discovery sources:
          Ethernet1/0, Src IP addr: 10.131.31.226
        Addresses bound to peer LDP Ident:
          10.131.63.229   10.131.63.221   10.131.63.241   10.131.63.251
          10.131.31.226
  Peer LDP Ident: 10.131.31.252:0; Local LDP Ident 10.131.31.251:0
        TCP connection: 10.131.31.252.11015 - 10.131.31.251.646
        State: Oper; Msgs sent/rcvd: 483/486; Downstream
        Up time: 06:45:14
        LDP discovery sources:
          Ethernet0/0, Src IP addr: 10.131.31.230
        Addresses bound to peer LDP Ident:
          10.131.31.245   10.131.31.252   10.131.31.230   10.131.31.233
          10.131.31.237
```

© 2008 Cisco Systems, Inc. All rights reserved. BCMVC v6.1—6-38

The **show mpls ldp neighbor command** displays the LDP identifiers of the local and remote routers, the IP addresses and the TCP port numbers between which the LDP connection is established, the connection uptime and the interfaces through which the LDP neighbor was discovered, as well as all the interface IP addresses used by the LDP neighbor.

Note The LDP identifier is determined in the same way as the OSPF or BGP identifier (unless controlled by the **mpls ldp router-id** command) - the highest IP address of all loopback interfaces is used. If no loopback interfaces are configured on the router, the LDP identifier becomes the highest IP address of any interface that was operational at the LDP process startup time.

Extended Ping to Router ID



Extended Ping to Router ID

```
#ping
Protocol [ip]:
Target IP address: 10.131.63.251
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: loopback 0
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.131.63.251, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5),
round-trip min/avg/max = 20/22/32 ms
```

Peer LDP Ident: 10.131.63.251
From `show mpls ldp neighbor`

Local LDP Ident: 10.131.31.251
From `show mpls ldp neighbor`

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—6-39

A TCP connection must be up between each pair of neighboring routers. The connection is used to exchange label bindings, which builds the LIB and - based on the routing table - the LFIB.

Verify that each LSR can ping the LDP identifier (LDP ID) of its neighbor.

LDP tries by default to use the highest numbered loopback address to establish communications, that is, a node attempts to use its highest-numbered loopback address for LDP peering. If there are no loopback interfaces then it uses its highest IP address.

This can be manually overridden with the command **(config)#mpls ldp router-id <interface>.**

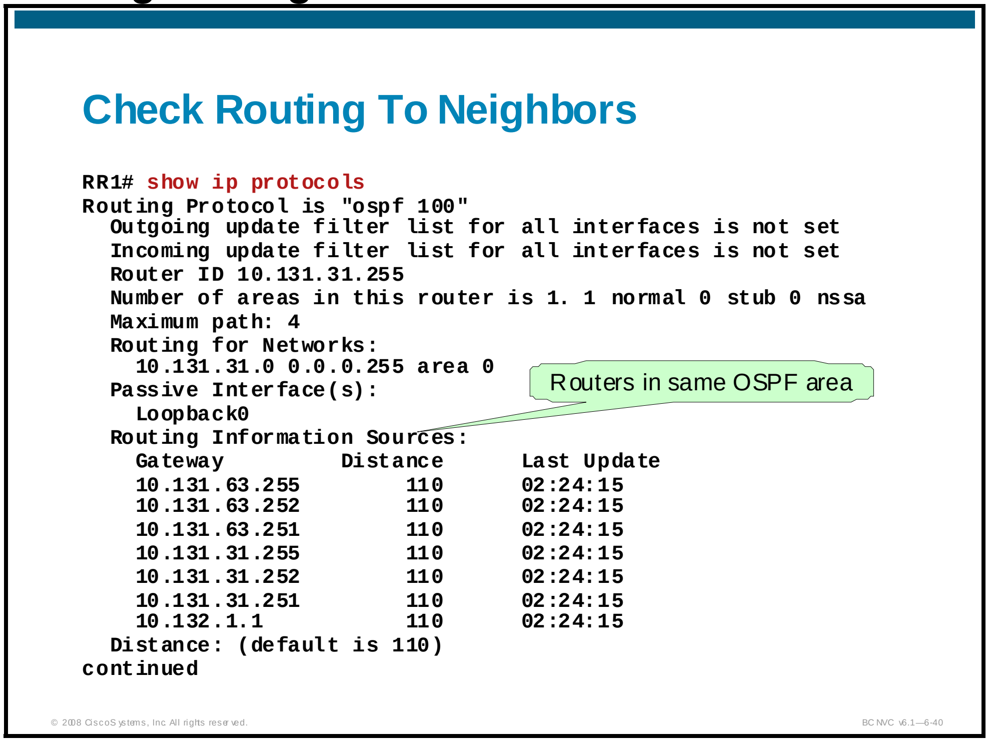
This command tells the node which interface address to use when trying to establish LDP neighbor relations.

Because typically loopback addresses are used, the easiest way to verify this is with an extended ping command as shown in the figure above.

If this connectivity is not available then LDP can not establish a neighbor relationship (assuming that loopbacks are used).

If this is not successful, verify that the appropriate IGP network is being advertised.

Check Routing to Neighbors

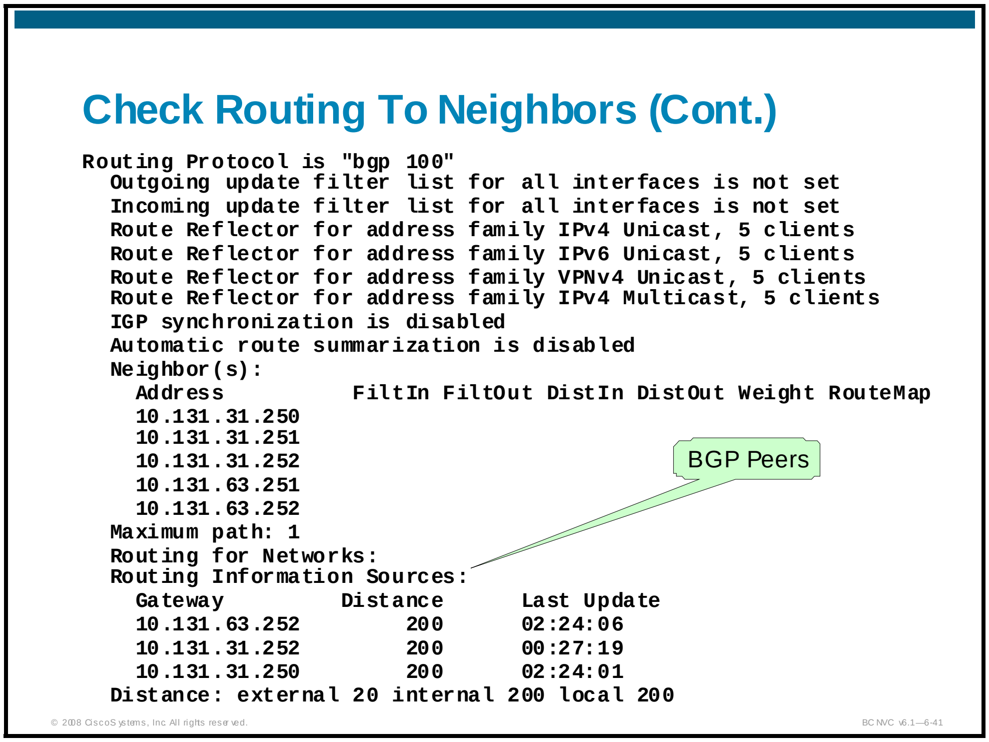


Check Routing To Neighbors

```
RR1# show ip protocols
Routing Protocol is "ospf 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 10.131.31.255
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    10.131.31.0 0.0.0.255 area 0
  Passive Interface(s):
    Loopback0
  Routing Information Sources:
    Gateway         Distance      Last Update
    10.131.63.255        110      02:24:15
    10.131.63.252        110      02:24:15
    10.131.63.251        110      02:24:15
    10.131.31.255        110      02:24:15
    10.131.31.252        110      02:24:15
    10.131.31.251        110      02:24:15
    10.132.1.1           110      02:24:15
  Distance: (default is 110)
continued
```

Routers in same OSPF area

© 2008 Cisco Systems, Inc. All rights reserved. BC NVC v6.1—6-40



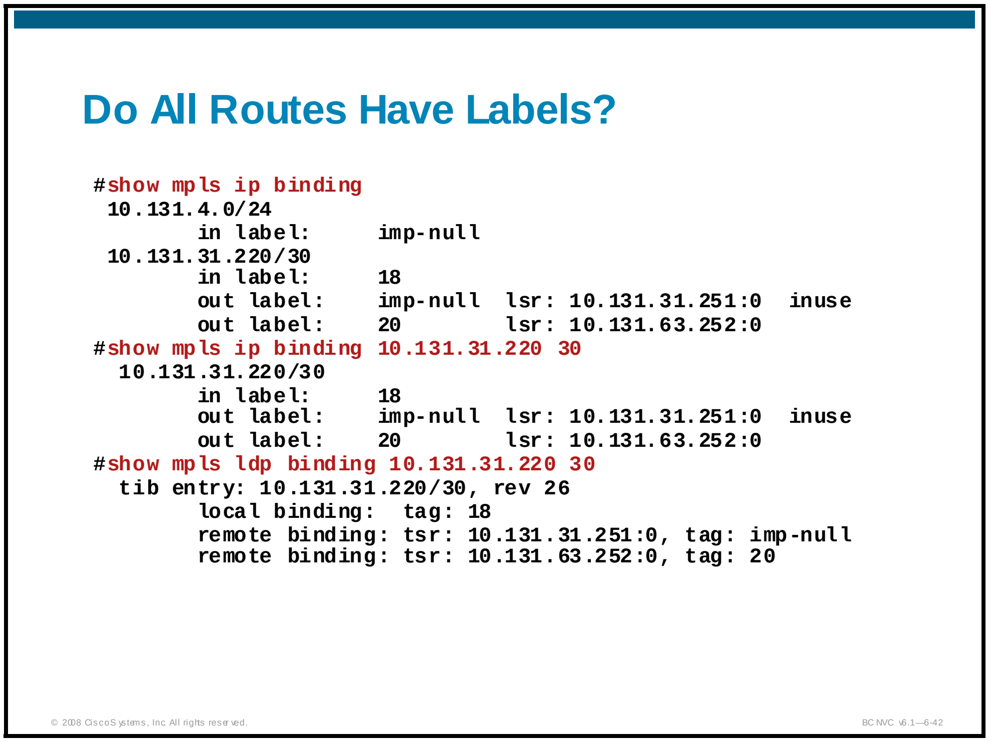
Check Routing To Neighbors (Cont.)

```
Routing Protocol is "bgp 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Route Reflector for address family IPv4 Unicast, 5 clients
  Route Reflector for address family IPv6 Unicast, 5 clients
  Route Reflector for address family VPNv4 Unicast, 5 clients
  Route Reflector for address family IPv4 Multicast, 5 clients
  IGP synchronization is disabled
  Automatic route summarization is disabled
  Neighbor(s):
    Address          FiltIn FiltOut DistIn DistOut Weight RouteMap
    10.131.31.250
    10.131.31.251
    10.131.31.252
    10.131.63.251
    10.131.63.252
  Maximum path: 1
  Routing for Networks:
  Routing Information Sources:
    Gateway         Distance      Last Update
    10.131.63.252        200      02:24:06
    10.131.31.252        200      00:27:19
    10.131.31.250        200      02:24:01
  Distance: external 20 internal 200 local 200
```

BGP Peers

© 2008 Cisco Systems, Inc. All rights reserved. BC NVC v6.1—6-41

Do All Routes Have Labels?



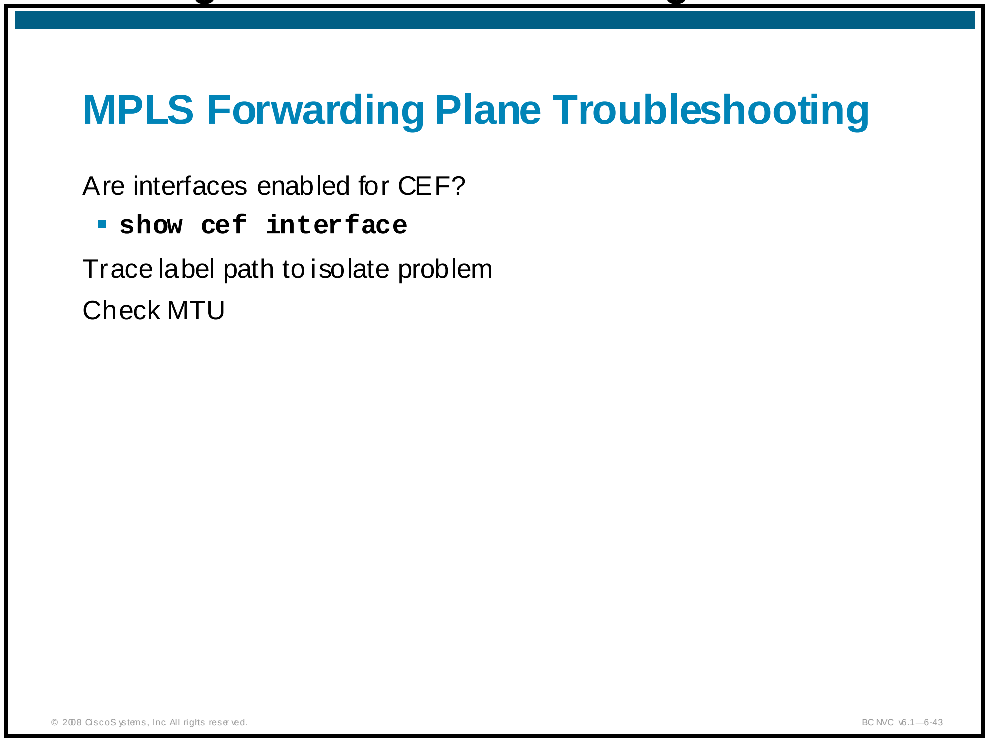
Do All Routes Have Labels?

```
#show mpls ip binding
 10.131.4.0/24
        in label:     imp-null
 10.131.31.220/30
        in label:     18
        out label:    imp-null  lsr: 10.131.31.251:0  inuse
        out label:    20        lsr: 10.131.63.252:0
#show mpls ip binding 10.131.31.220 30
  10.131.31.220/30
        in label:     18
        out label:    imp-null  lsr: 10.131.31.251:0  inuse
        out label:    20        lsr: 10.131.63.252:0
#show mpls ldp binding 10.131.31.220 30
  tib entry: 10.131.31.220/30, rev 26
        local binding:  tag: 18
        remote binding: tsr: 10.131.31.251:0, tag: imp-null
        remote binding: tsr: 10.131.63.252:0, tag: 20
```

Label bindings are labels associated with a particular destination. You can see them using one of the following commands (depending on which Cisco IOS version and which label distribution protocol you are using).

Note that labels for each forwarding class are established at each LSR, even if they are not on the preferred (shortest) path. In this case, a packet destined to 10.10.10.4/32 can go by 10.10.10.1 (with label 22) or 10.10.10.6 (with label 24). The LSR chooses the first solution because it is the shortest one. This decision is made using the standard IP routing table (which in this case was built using OSPF).

MPLS Forwarding Plane Troubleshooting



MPLS Forwarding Plane Troubleshooting

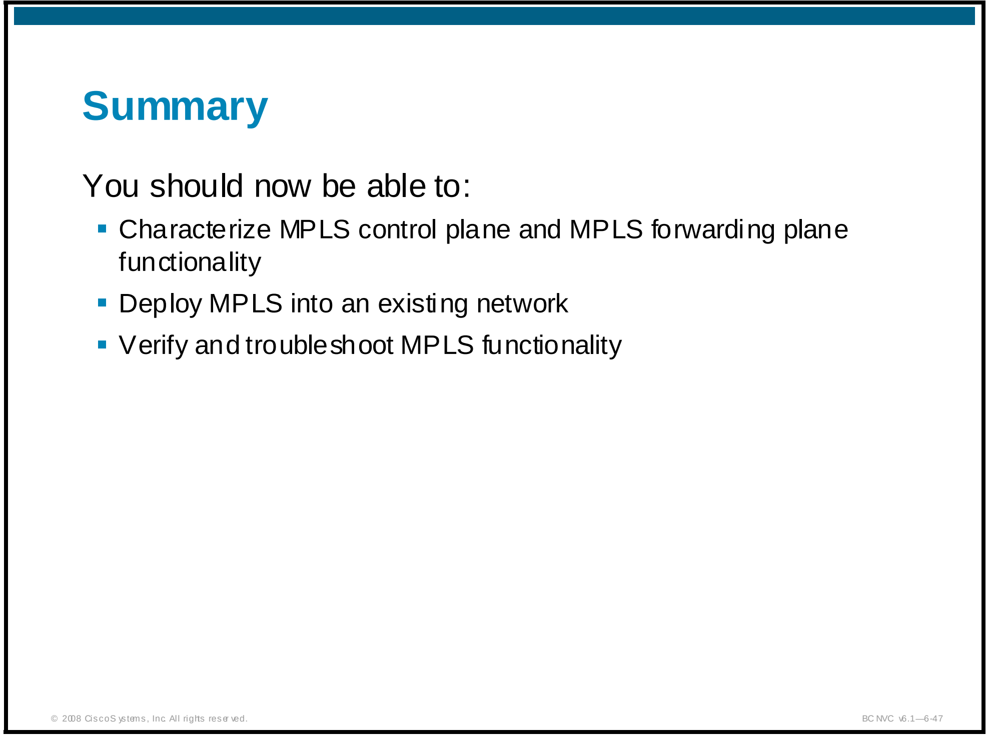
Are interfaces enabled for CEF?

- **`show cef interface`**

Trace label path to isolate problem

Check MTU

Summary



Summary

You should now be able to:

- Characterize MPLS control plane and MPLS forwarding plane functionality
- Deploy MPLS into an existing network
- Verify and troubleshoot MPLS functionality

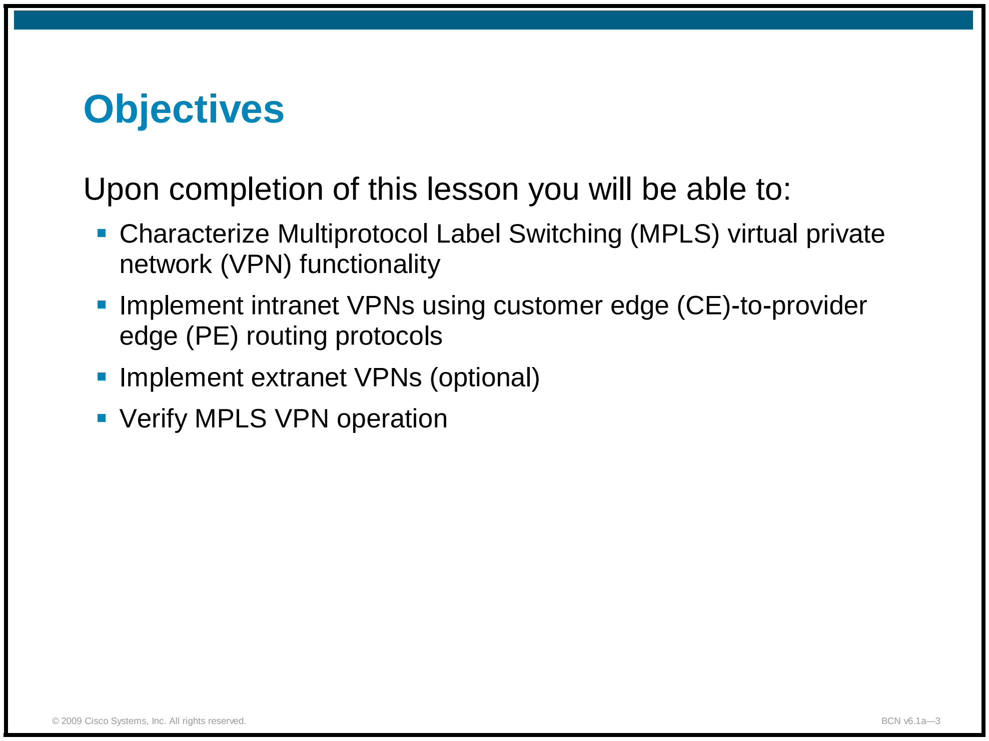
BCNVC v6.1—6-47

- MPLS allows flexible packet classification and network resource optimization
- Labels may be distributed by different protocols
  - — LDP, TDP, RSVP, BGP, others
- Different distribution protocols may coexist in the same LSR
- Labels have local (LSR) significance
  - — No need for global (domain) label allocation or numbering
- Decouples IP packet forwarding from the information carried in the IP header of the packet
- Provides multiple routing paradigms (destination-based, explicit routing, VPN, multicast, (CoS) over a common forwarding algorithm (label swapping)
- Facilitates integration of ATM and IP—from control plane point of view an MPLS-capable ATM switch looks like a router

Lesson 5

Implement Intranet and Extranet MPLS VPNs

Objectives

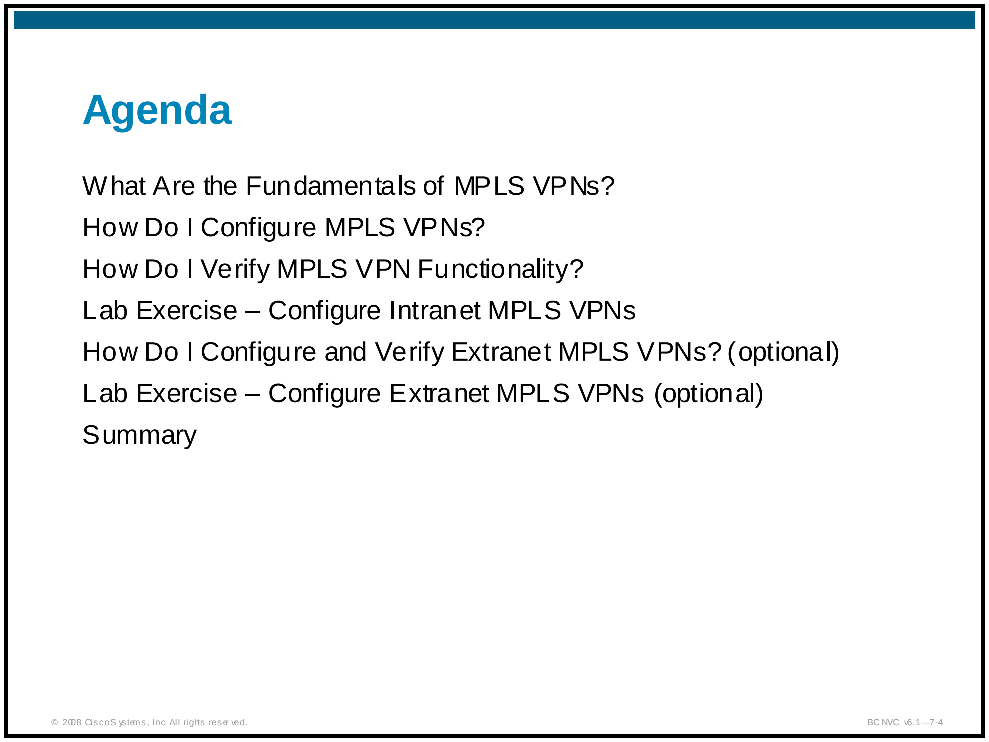


Objectives

Upon completion of this lesson you will be able to:

- Characterize Multiprotocol Label Switching (MPLS) virtual private network (VPN) functionality
- Implement intranet VPNs using customer edge (CE)-to-provider edge (PE) routing protocols
- Implement extranet VPNs (optional)
- Verify MPLS VPN operation

Agenda



Agenda

What Are the Fundamentals of MPLS VPNs?

How Do I Configure MPLS VPNs?

How Do I Verify MPLS VPN Functionality?

Lab Exercise – Configure Intranet MPLS VPNs

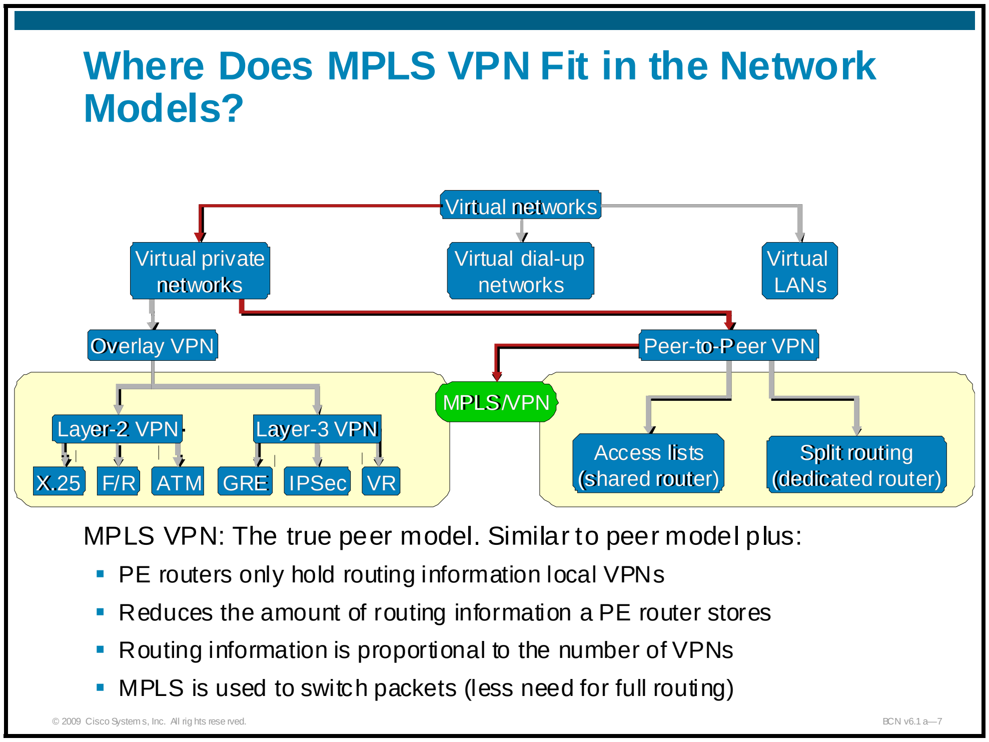
How Do I Configure and Verify Extranet MPLS VPNs? (optional)

Lab Exercise – Configure Extranet MPLS VPNs (optional)

Summary

What Are the Fundamentals of MPLS VPNs?

Where Does MPLS VPN Fit in the Network Models?



Where Does MPLS VPN Fit in the Network Models?

MPLS VPN: The true peer model. Similar to peer model plus:

- PE routers only hold routing information local VPNs
- Reduces the amount of routing information a PE router stores
- Routing information is proportional to the number of VPNs
- MPLS is used to switch packets (less need for full routing)

BCN v6.1 a—7

A VPN is an IP network infrastructure that delivers private network services over a public infrastructure.

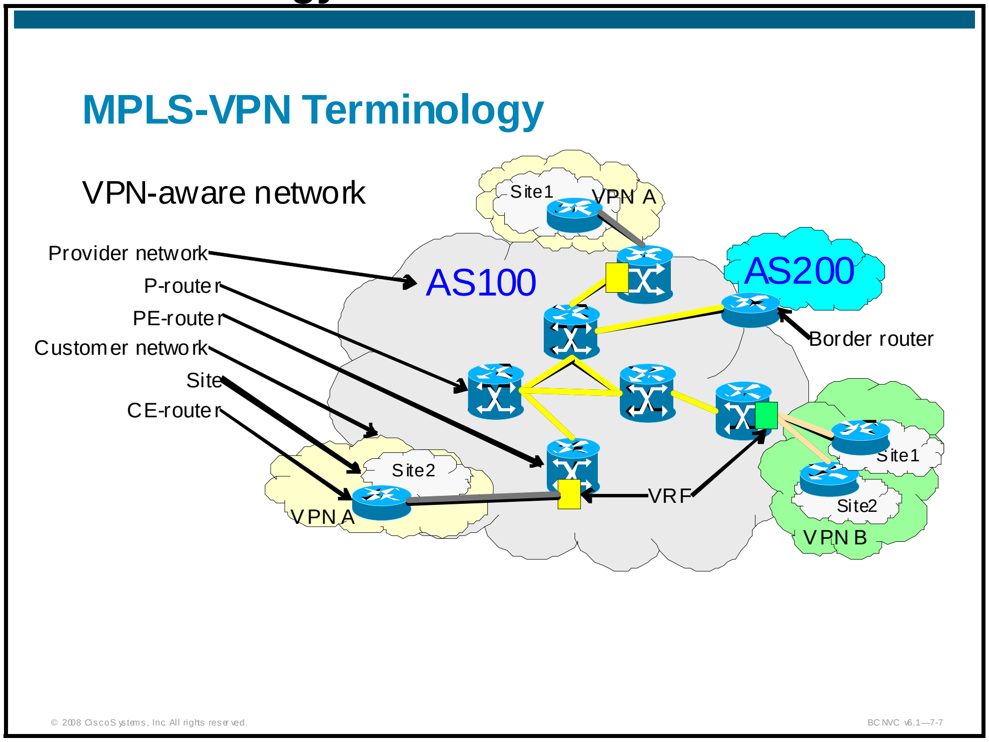
The overlay VPN model, most commonly used in a service provider network, dictates that the design and provisioning of virtual circuits across the backbone must be complete prior to any traffic flow. In the case of an IP network, this means that even though the underlying technology is connectionless, it requires a connection-oriented approach to provision the service.

From a service provider point of view, the scaling issues of an overlay VPN model are felt most when having to manage and provision a large number of circuits or tunnels between customer devices. From a customer point of view, the Interior Gateway Protocol (IGP) design is typically extremely complex and also difficult to manage.

On the other hand, the peer-to-peer VPN models, before MPLS VPNs, suffered from lack of isolation between the customers and the need for coordinated IP address space between them.

With the introduction of Multiprotocol Label Switching (MPLS), which combines the benefits of Layer 2 switching with Layer 3 routing, it is possible to construct a technology that combines the benefits of an overlay VPN (such as security and isolation among customers) with the benefits of simplified routing that a peer-to-peer VPN implementation brings. MPLS VPN results in simpler customer routing and somewhat simpler service provider provisioning, and makes possible a number of topologies that are hard to implement in either the overlay or peer-to-peer VPN models. MPLS also adds the benefits of a connection-oriented approach to the IP routing paradigm, through the establishment of label-switched paths, which are created based on topology information rather than traffic flow.

MPLS-VPN Terminology



MPLS-VPN Terminology

VPN-aware network

Provider network

P-router

PE-router

Customer network

Site

CE-router

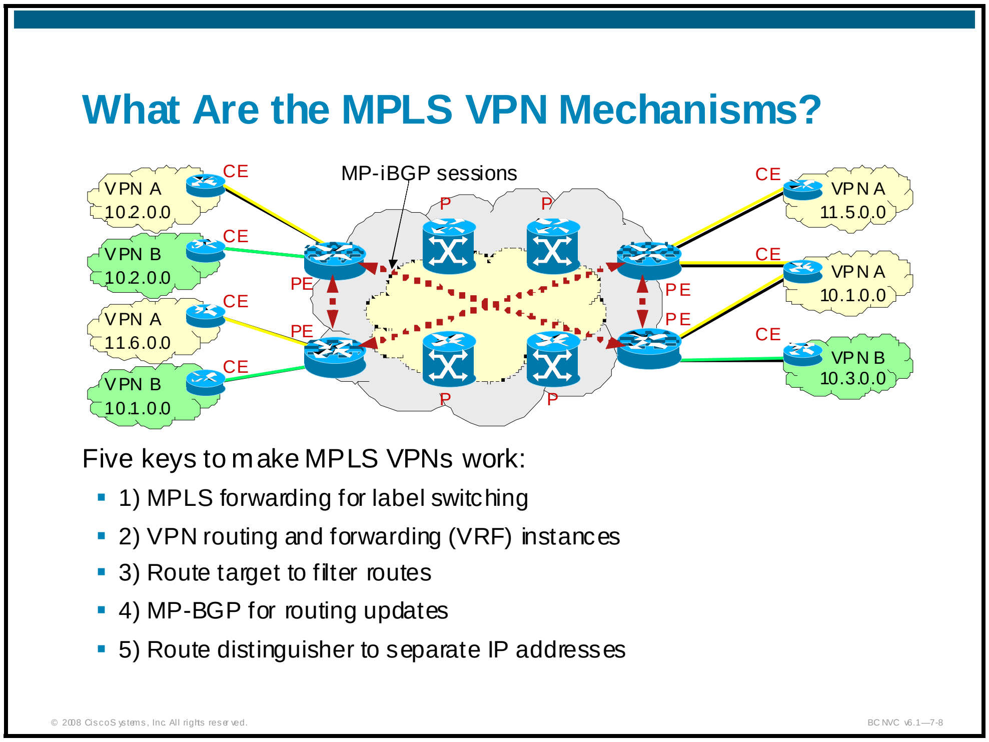
BCNVC v6.1—7-7

A VPN contains customer devices attached to the CE-routers. These customer devices use VPNs to exchange information (routing updates) between devices. Only the PE-routers are aware of the VPNs.

MPLS-VPN Terminology and Definitions

Term	Definition
Provider Network (P-Network)	The backbone under control of a service provider
Customer Network (C-Network)	Network under customer control
CE-router	Customer edge router. Part of the customer network and interfaces to a PE-router
Site	Set of (sub)networks part of the customer network and co-located. A site is connected to the VPN backbone through one or more PE-to-CE links
PE-router	Provider edge router. Part of the provider network and interfaces to CE-routers
P-router	Provider (core) router, without knowledge of VPN
Border router	Provider edge router interfacing to other provider networks
VRF	VPN routing and forwarding instance
Extended Community	BGP attribute used to identify a route-origin, route-target
Site of Origin Identifier (SOO)	64 bits identifying the site where the route originated
route target	64 bits identifying the VRFs that should receive the route
Route Distinguisher	Attributes of each route used to uniquely identify prefixes among VPNs (64 bits). RD is VRF-based (not VPN-based)
VPN-IPv4 addresses	Normal IP address including the 64-bit route distinguisher and the 32-it IP address
Routing table and FIB table	Populated by routing protocol contexts
VPN-Aware network	A provider backbone where MPLS PN is deployed

What Are the MPLS VPN Mechanisms?



What Are the MPLS VPN Mechanisms?

Five keys to make MPLS VPNs work:

- 1) MPLS forwarding for label switching
- 2) VPN routing and forwarding (VRF) instances
- 3) Route target to filter routes
- 4) MP-BGP for routing updates
- 5) Route distinguisher to separate IP addresses

BCNVC v6.1—7-8

There are five principal technologies that make it possible to build MPLS-based VPNs:

1. MPLS forwards packets between PEs (across a service provider backbone)

2. Each PE has multiple VPN routing and forwarding instances (VRFs)

3. Route targets (RTs) define route policy

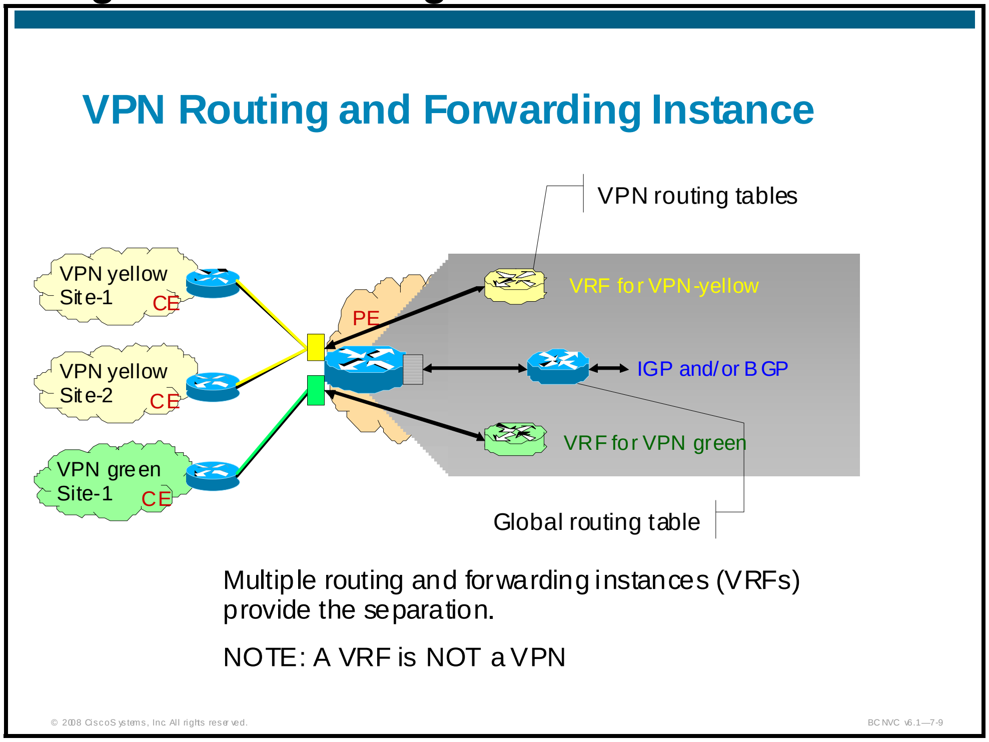
4. Multiprotocol Border Gateway Protocol (MP-BGP) between PEs carries CE routing information

5. Route distinguishers (RDs) uniquely identify IP addresses

Network topology factors:

- The VPN backbone is composed of MPLS label switch routers (LSRs).
- P-routers (core LSRs) are the MPLS backbone. They do not run BGP and do not have any VPN knowledge.
- PE-routers (edge LSRs) are faced to customer edge (CE) routers and distribute VPN information (VPN IPv4 addresses, extended community, label) through MP-BGP to other PE-routers
- PE-routers use MPLS to the core and plain IP to CE-routers
- P- and PE-routers share a common IGP
- PE-routers are MP-IBGP fully meshed

VPN Routing and Forwarding Instance



VPN Routing and Forwarding Instance

Multiple routing and forwarding instances (VRFs) provide the separation.

NOTE: A VRF is NOT a VPN

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—7-9

The VPN routing and forwarding (VRF) table is a key element in the MPLS VPN technology. VRFs exist only at the source of a VPN. While this is most commonly the PE-router, there are some deployment scenarios, such as inter-AS, where VRFs may be found at an Autonomous System Boundary Router (ASBR). A VRF is a routing table instance, and more than one VRF can exist on a PE. A VPN can contain one or more VRFs on a PE. The VRF contains routes that should be available to a particular set of sites. VRFs use Cisco Express Forwarding (CEF) technology; therefore the router must be CEF-enabled.

A VRF is associated with the following elements:

- IP routing table
- Derived forwarding table, based on CEF
- A set of interfaces that use the derived forwarding table
- A set of routing protocols and routing peers that inject information into the VRF

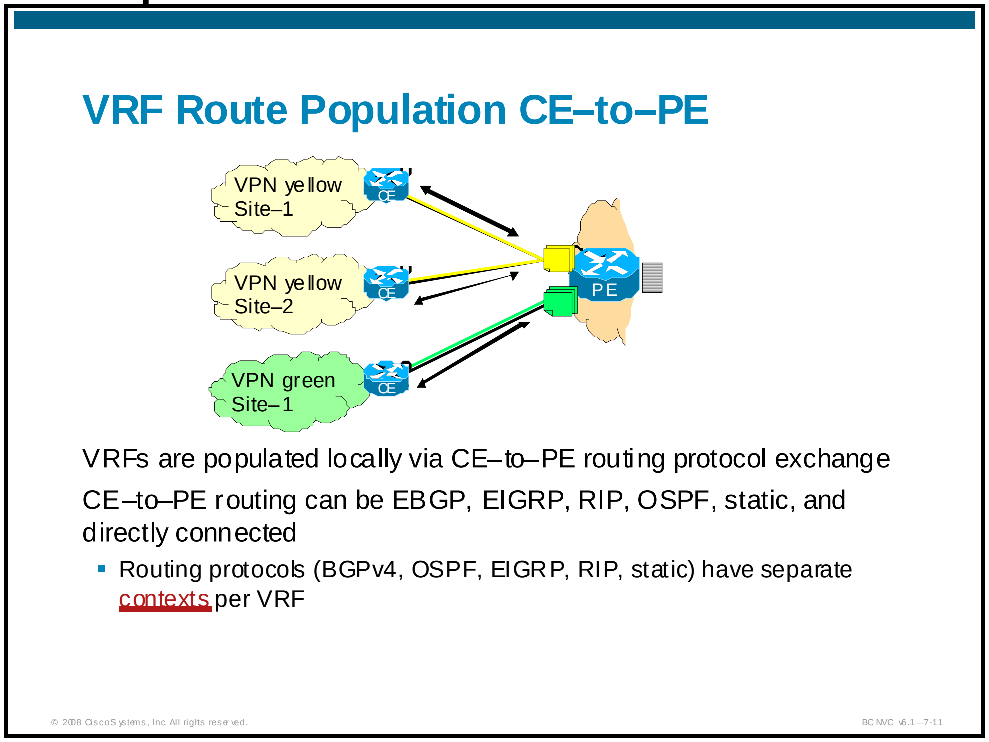
Each PE maintains one or more VRFs. MPLS VPN software looks up a particular packet's IP destination address in the appropriate VRF only if that packet arrived directly through an interface that is associated with that VRF. The so-called “color” MPLS label tells the destination PE to check the VRF for the appropriate VPN so that it can deliver the packet to the correct CE and finally to the local host machine.

PE and CE-routers exchange routing information through EBGP, RIP, OSPF, EIGRP, or static routing. CE-routers run standard routing software. PE-routers maintain separate routing tables:

- The global routing table contains all PE and P routes populated by the VPN backbone IGP
- VRF (VPN routing and forwarding) table is associated with one or more directly connected sites (CEs). VRF are associated to interfaces (sub, virtual, or tunnel).

Interfaces may share the same VRF if the connected sites share the same routing information.

VRF Route Population CE-to-PE



VRF Route Population CE–to–PE

VRFs are populated locally via CE–to–PE routing protocol exchange

CE–to–PE routing can be EBGP, EIGRP, RIP, OSPF, static, and directly connected

- Routing protocols (BGPv4, OSPF, EIGRP, RIP, static) have separate contexts per VRF

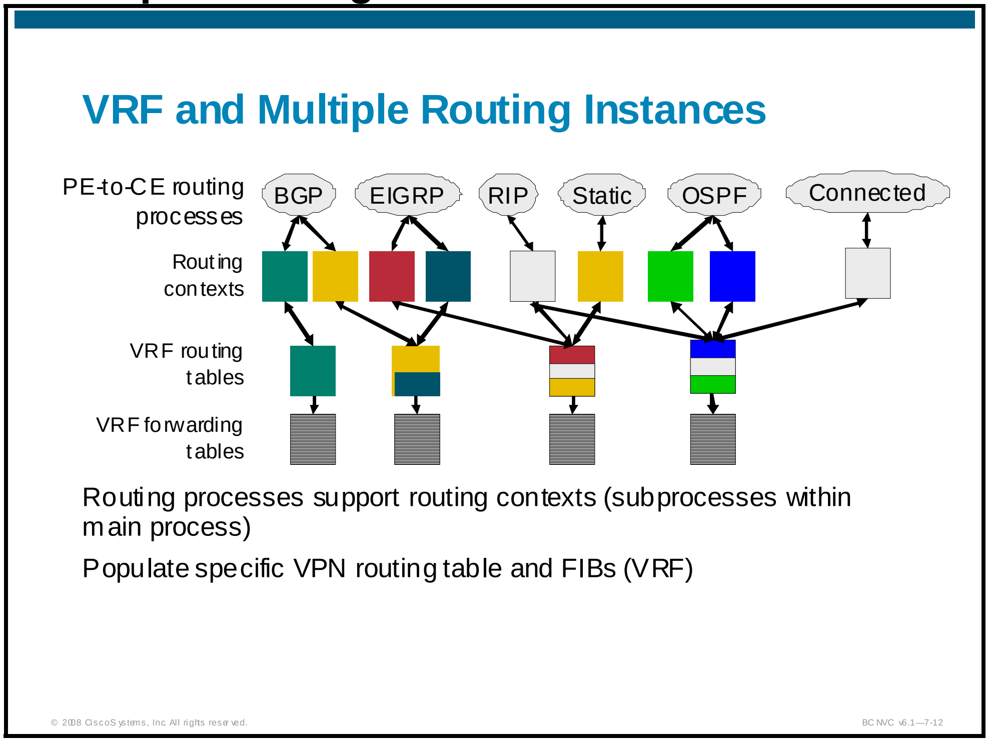
BCNVC v6.1—7-11

VRF is populated locally through PE and CE routing protocol exchange. The routes that a PE receives from CE-routers are installed in the appropriate VRF. The following routing protocols are supported:

- RIP, OSPF, BGPv4, EIGRP, and static routing
- Routing protocol context (BGPv4, RIP, and EIGRP)
- Separate process for OSPF

Note An OSPF process must be configured for each VRF. Before 12.3(4)T, the number of routing processes on a router was limited to 32 total processes; but in a VRF situation you could use only 28 OSPF processes due to other required routing processes (IGP, static, connected, and BGP). Since 12.3(4)T, 32 OSPF processes are supported per VRF. The total number of OSPF processes is therefore limited only by the available resources of the PE-router.

VRF and Multiple Routing Instances



VRF and Multiple Routing Instances

Routing processes support routing contexts (subprocesses within main process)

Populate specific VPN routing table and FIBs (VRF)

BCMVC v6.1—7-12

The overlapping addresses, usually resulting from usage of private IP addresses in customer networks, are one of the major obstacles to successful deployment of peer-to-peer VPN implementations. The MPLS VPN technology provides an elegant solution to the dilemma: Each VPN has its own routing and forwarding table in the router, so any customer or site that belongs to that VPN is provided access only to the set of routes contained within that table. Any PE-router in an MPLS VPN network thus contains a number of per-VPN routing tables and a global routing table that is used to reach other routers in the provider network, as well as external globally reachable destinations (for example, the rest of the Internet). Effectively, a number of virtual routers are created in a single physical router,

The relationship between virtual private networks and VPN routing and forwarding tables as explained in the previous paragraph is a slight simplification of the actual relationship between these two concepts. Nevertheless, it is true in cases where each site (or customer) belongs only to one VPN.

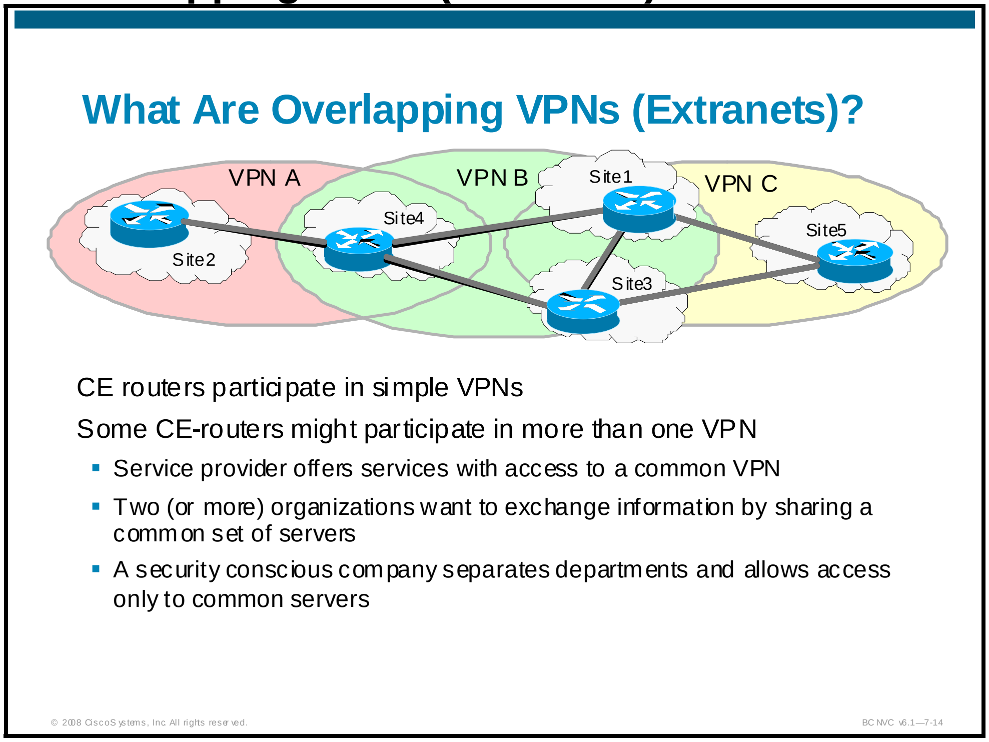
The concept of virtual routers allows the customers to use either global or private IP address space in each VPN. Each customer site belongs to a particular VPN, so the only requirement is that the address space be unique within that VPN. Uniqueness of addresses is not required among VPNs except where two VPNs that share the same private address space want to communicate.

More structures are associated with each virtual router than just the virtual IP routing table:

- A forwarding table that is derived from the routing table and is based on CEF technology.
- A set of interfaces that use the derived forwarding table.
- Rules that control the import and export of routes to and from the VPN routing table. These rules were introduced to support overlapping VPNs and are explained later in this chapter.

- A set of routing protocols and peers, which inject information into the VPN routing table. This includes static routing.
- Router variables associated with the routing protocol that is used to populate the VPN routing table.

What Are Overlapping VPNs (Extranets)?



What Are Overlapping VPNs (Extranets)?

CE routers participate in simple VPNs

Some CE-routers might participate in more than one VPN

- Service provider offers services with access to a common VPN
- Two (or more) organizations want to exchange information by sharing a common set of servers
- A security conscious company separates departments and allows access only to common servers

BCNVC v6.1—7-14

Each VPN is associated with one or more VRFs. A VRF consists of an IP routing table, a derived CEF table, a set of interfaces that use the forwarding table, and a set of rules and routing protocol parameters that control the information that is included in the routing table.

A one-to-one relationship does not necessarily exist between customer sites and VPNs. A given site can be a member of multiple VPNs, as shown. However, a site can associate with only one VRF. A VRF contains all the routes from the VPNs of which it is a member.

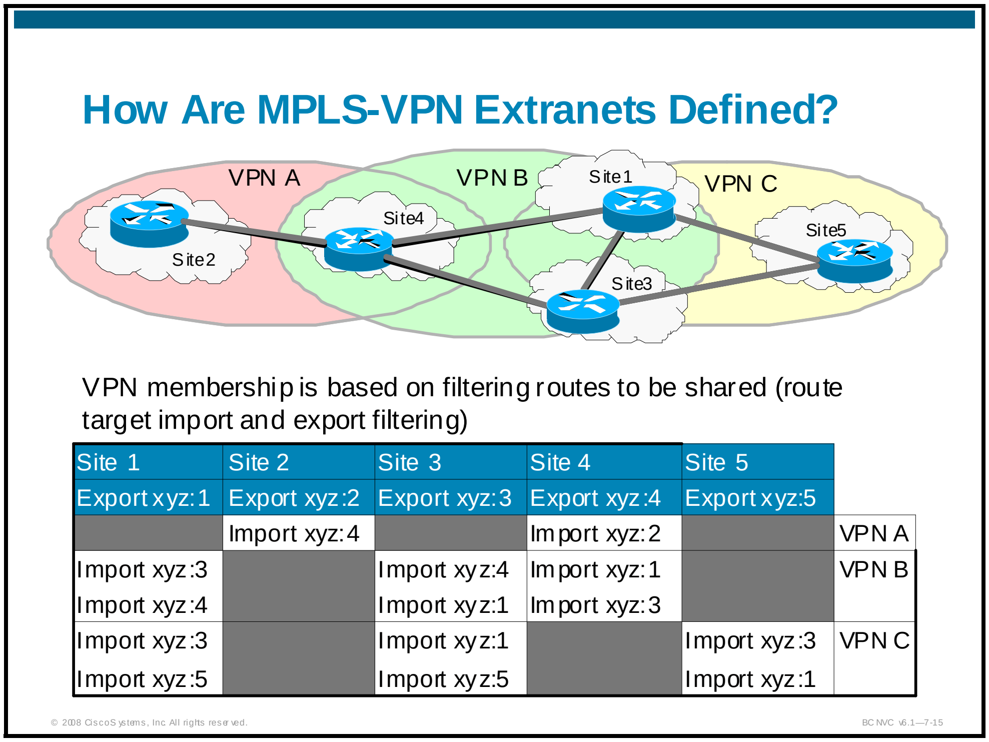
Packet forwarding information is stored in the IP routing table and the CEF table for each VRF. A separate set of routing and CEF tables is maintained for each VRF. These tables prevent information from being forwarded outside a VPN, and also prevent packets that are outside a VPN from being forwarded to a router within the VPN.

The figure shows five customer sites communicating within three VPNs. The VPNs can communicate with the following sites: VPN A---sites 2 and 4, VPN B---sites 1, 3, and 4, VPN C---sites 1, 3, and 5

Typical usages for extranet VPNs are:

- A service provider offers management services and allows customers to access this VPN
- Companies that use MPLS VPN to implement both intranet and extranet services. In this scenario each company participating in the extranet VPN would probably deploy a security mechanism on its CE-routers to prevent other companies participating in the VPN from gaining access to other sites in the customer VPN.
- Some security-conscious companies might decide to deploy limited visibility between different departments in the same organization because of security reasons. Extranet VPNs might be used as a solution in this case.

How Are MPLS-VPN Extranets Defined?



How Are MPLS-VPN Extranets Defined?

VPN membership is based on filtering routes to be shared (route target import and export filtering)

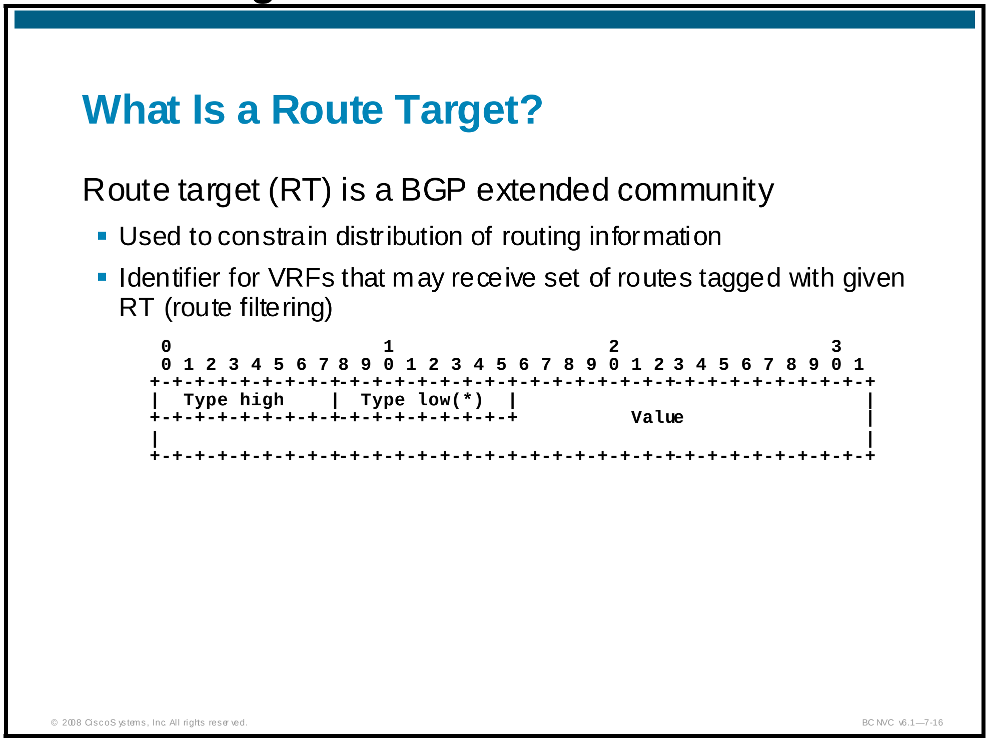
Site 1	Site 2	Site 3	Site 4	Site 5	
Export xyz:1	Export xyz:2	Export xyz:3	Export xyz:4	Export xyz:5	
	Import xyz:4		Import xyz:2		VPN A
Import xyz:3 Import xyz:4		Import xyz:4 Import xyz:1	Import xyz:1 Import xyz:3		VPN B
Import xyz:3 Import xyz:5		Import xyz:1 Import xyz:5		Import xyz:3 Import xyz:1	VPN C

BCNVC v6.1—7-15

There is no one-to-one mapping between VPN and VRF, for the router to know which routes need to be inserted into which VRF the route target is used. Every VPN route is tagged with one or more route targets when it is exported from a VRF (to be offered to other VRFs). You can also associate a set of route targets to a VRF, and all routes tagged with at least one of those route targets are inserted into the VRF.

The route target is the closest approximation to a VPN identifier in the MPLS VPN architecture. In most VPN topologies, you can equate them, but in other topologies (usually a central services topology), a single VPN might need more than one route target for successful implementation.

What Is a Route Target?



What Is a Route Target?

Route target (RT) is a BGP extended community

- Used to constrain distribution of routing information
- Identifier for VRFs that may receive set of routes tagged with given RT (route filtering)

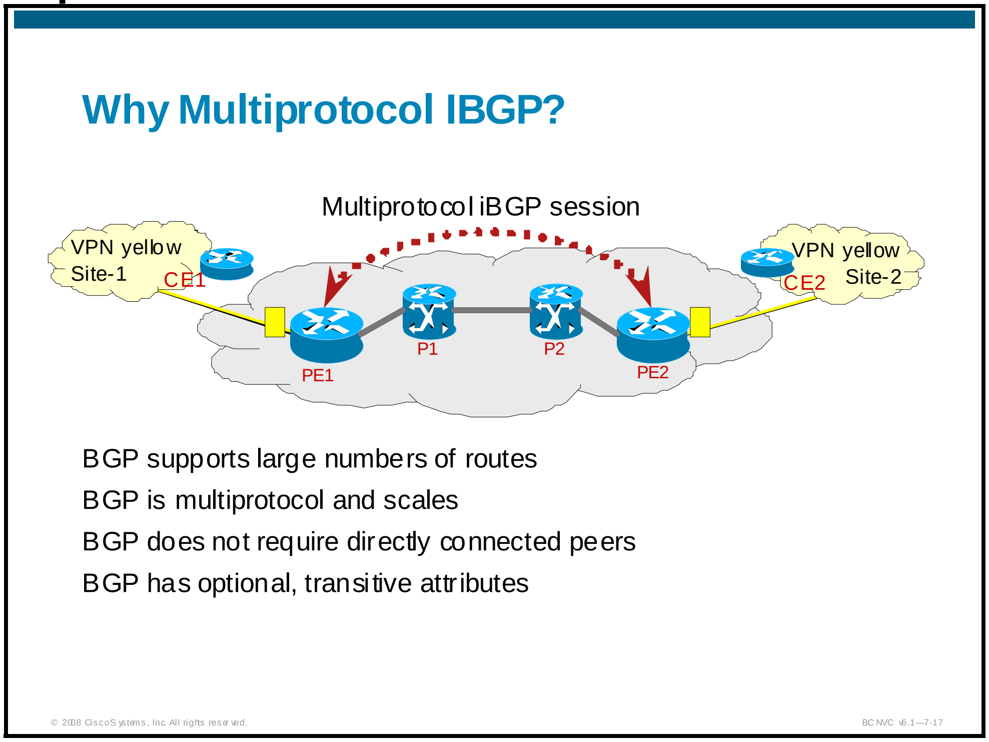
```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type high    |  Type low(*)  |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+          Value                |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

© 2008 Cisco Systems, Inc. All rights reserved. BCMVC v6.1—7-16

The BGP extended communities draft specifies two new communities defined as the route target and the route origin. The route origin, referred to as the SOO by the Cisco implementation, prevents routing loops between sites, and the route target extended community defines the import and export policies that a particular VRF uses.

A route target is a 64-bit BGP extended community attached to a BGP route. The MP-IBGP update propagates the extended community along with other BGP attributes between PE-routers, and its value determines to which VRF or set of VRFs to import the route. Careful definition of the route target extended community values provides the flexibility to provision many different VPN topologies.

Why Multiprotocol IBGP?



Why Multiprotocol IBGP?

- BGP supports large numbers of routes
- BGP is multiprotocol and scales
- BGP does not require directly connected peers
- BGP has optional, transitive attributes

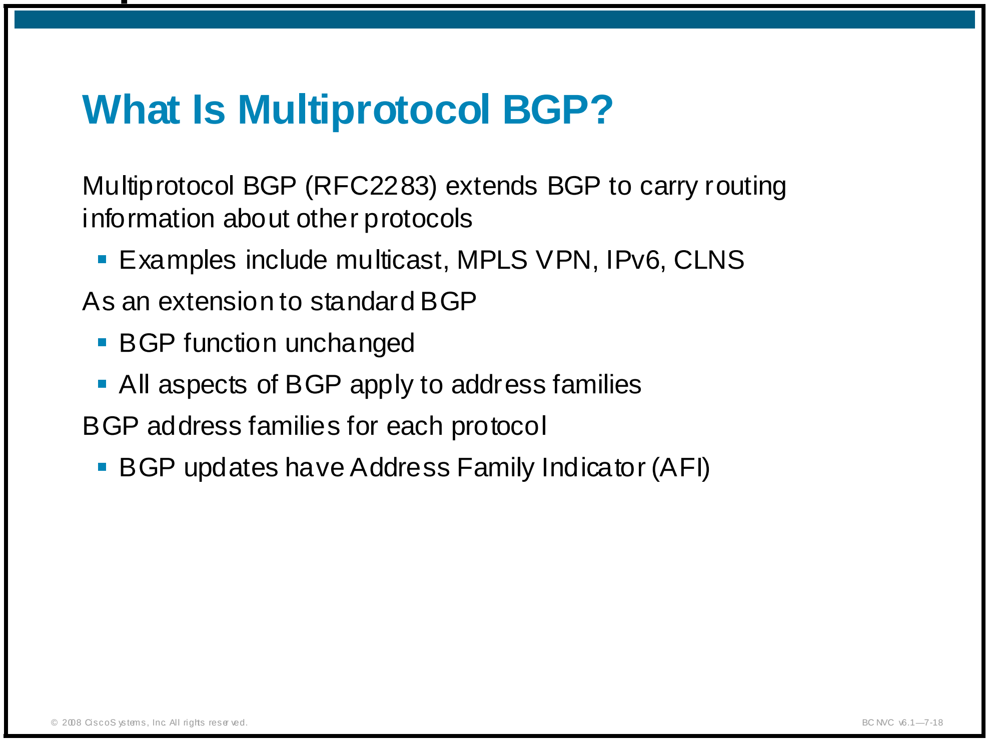
IP subnets advertised by CE-routers to PE-routers are augmented with a 64-bit prefix called a route distinguisher to make them unique. The resulting 96-bit addresses are then exchanged between the PE-routers using a special address family of MP-BGP. There were several reasons for choosing BGP as the routing protocol used to transport VPN routes.

The number of VPN routes in a network can become very large. BGP is the only routing protocol that can support a very large number of routes.

BGP, EIGRP, and IS-IS are the only routing protocols that are multiprotocol by design (all of them can carry routing information for a number of different address families). IS-IS and EIGRP, however, do not scale to the same number of routes as BGP. BGP is also designed to exchange information between routers that are not directly connected. This BGP feature supports keeping VPN routing information out of the provider core routers (P-routers).

BGP can carry any information attached to a route as an optional BGP attribute. You can define additional attributes that are transparently forwarded by any BGP-router that does not understand them. This property of BGP makes propagation of route targets between PE-routers extremely simple.

What Is Multiprotocol BGP?



What Is Multiprotocol BGP?

Multiprotocol BGP (RFC2283) extends BGP to carry routing information about other protocols

- Examples include multicast, MPLS VPN, IPv6, CLNS

As an extension to standard BGP

- BGP function unchanged
- All aspects of BGP apply to address families

BGP address families for each protocol

- BGP updates have Address Family Indicator (AFI)

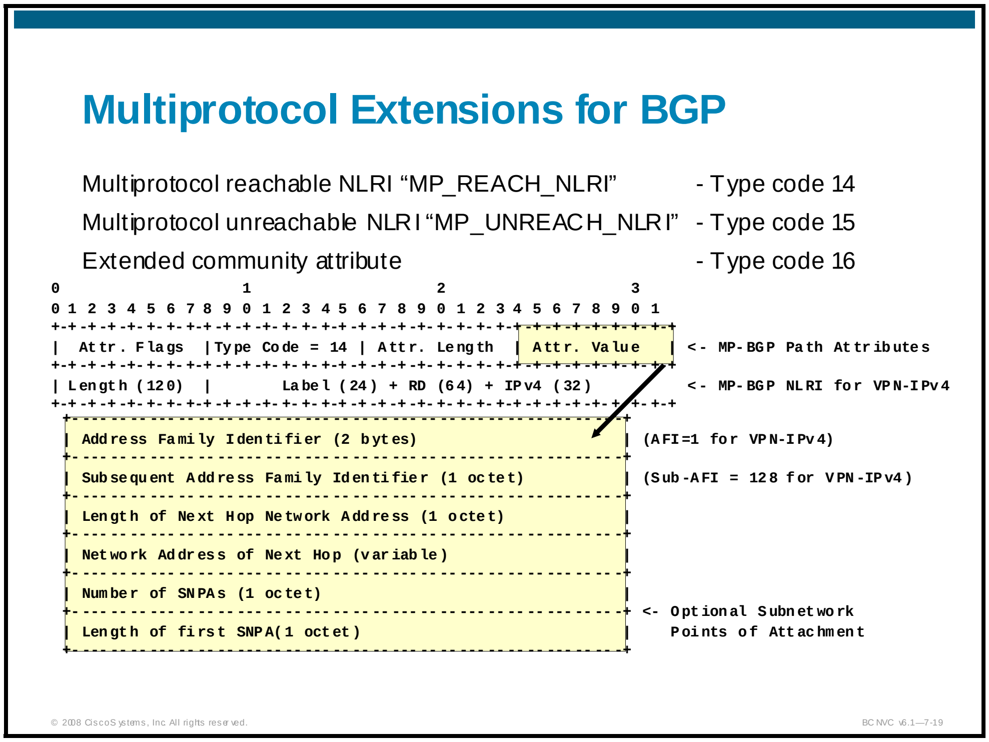
BCNVC v6.1—7-18

Multiprotocol BGP (MP–BGP) is defined in RFC 2283. This RFC defines extensions to the existing BGP protocol that allow it to carry more than just IPv4 route prefixes. Examples of some of the new types of routing information include (but are not limited to):

- IPv4 prefixes for unicast routing
- IPv4 prefixes for multicast RPF checking
- IPv6 prefixes for unicast routing

Because MP–BGP is an extension to the existing BGP protocol, the same basic rules apply to path selection, path validation, and so on.

Multiprotocol Extensions for BGP



Multiprotocol Extensions for BGP

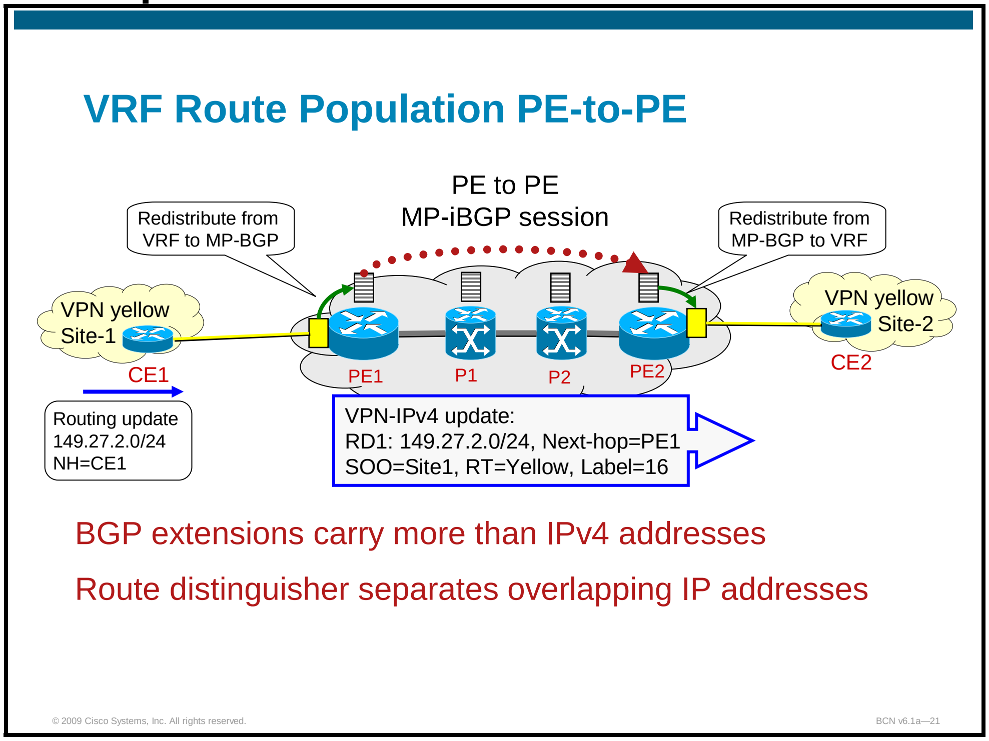
Multiprotocol reachable NLRI “MP_REACH_NLRI” - Type code 14

Multiprotocol unreachable NLRI “MP_UNREACH_NLRI” - Type code 15

Extended community attribute - Type code 16

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Attr. Flags  |Type Code = 14 | Attr. Length  | Attr. Value   | <- MP-BGP Path Attributes
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length (120)  |       Label (24) + RD (64) + IPv4 (32)            <- MP-BGP NLRI for VPN-IPv4
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 +---------------------------------------------------------+
 | Address Family Identifier (2 bytes)                     | (AFI=1 for VPN-IPv4)
 +---------------------------------------------------------+
 | Subsequent Address Family Identifier (1 octet)          | (Sub-AFI = 128 for VPN-IPv4)
 +---------------------------------------------------------+
 | Length of Next Hop Network Address (1 octet)            |
 +---------------------------------------------------------+
 | Network Address of Next Hop (variable)                  |
 +---------------------------------------------------------+
 | Number of SNPAs (1 octet)                               |
 +---------------------------------------------------------+ <- Optional Subnetwork
 | Length of first SNPA(1 octet)                           |    Points of Attachment
 +---------------------------------------------------------+
```

VRF Route Population PE-to-PE



VRF Route Population PE-to-PE

Routing update
149.27.2.0/24
NH=CE1

VPN-IPv4 update:
RD1: 149.27.2.0/24, Next-hop=PE1
SOO=Site1, RT=Yellow, Label=16

BGP extensions carry more than IPv4 addresses

Route distinguisher separates overlapping IP addresses

PE-routers distribute local VPN information across the MPLS VPN backbone

- Through the use of MP-IBGP and redistribution from VRF
- Receiving PE imports routes into attached VRFs

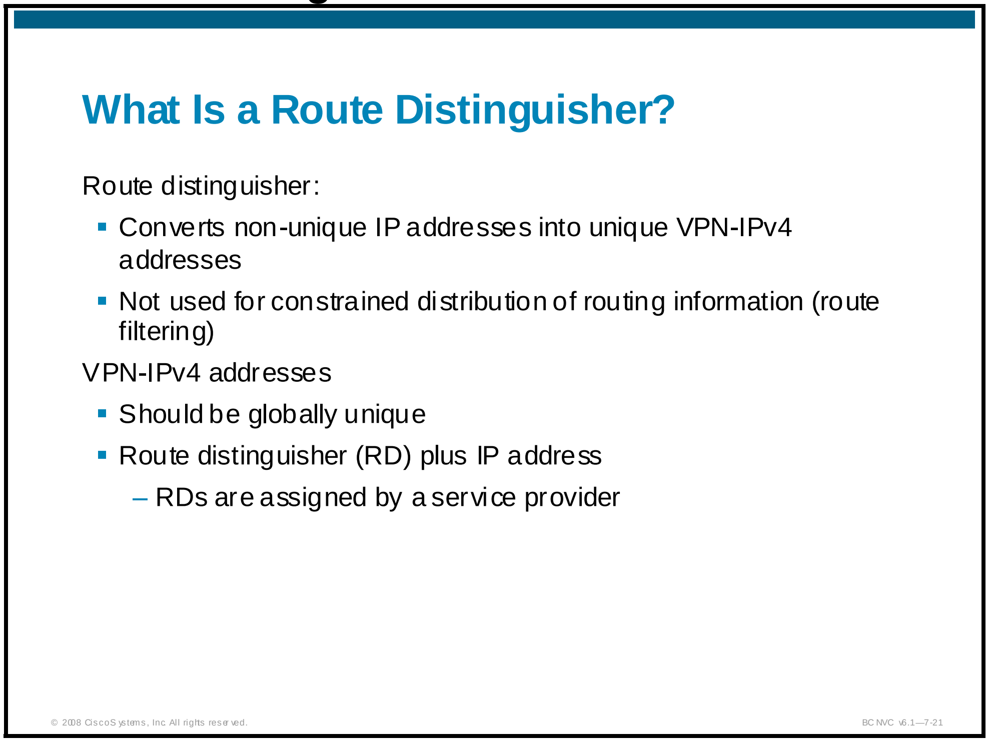
The routes the PE receives through the backbone IGP are installed in the global routing table.

MPLS-based VPNs use BGP to communicate between PEs to facilitate customer routes. This is made possible through extensions to BGP that carry addresses other than IPv4 addresses. A notable extension is the route distinguisher (RD).

The MPLS label is part of a BGP routing update. The routing update also carries the addressing and reachability information. When the RD is unique across the MPLS VPN network, proper connectivity is established even if different customers use non-unique IP addresses.

Since MPLS forwards traffic based on labels, you can use it to bind VPN IP routes to label switched routes. Since the MPLS switches read labels, not packet headers, they bypass the fact that the interior routers have no knowledge of the actual IP addresses within the underlying packet.

What Is a Route Distinguisher?



What Is a Route Distinguisher?

Route distinguisher:

- Converts non-unique IP addresses into unique VPN-IPv4 addresses
- Not used for constrained distribution of routing information (route filtering)

VPN-IPv4 addresses

- Should be globally unique
- Route distinguisher (RD) plus IP address
  - RDs are assigned by a service provider

BCNVC v6.1—7-21

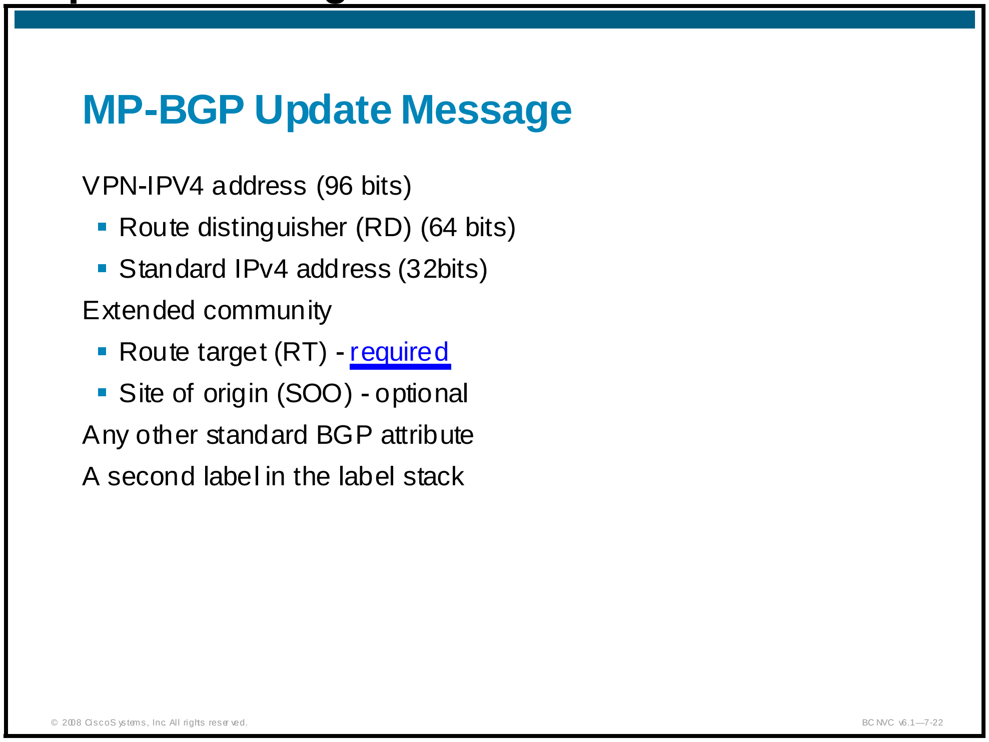
The purpose of a route distinguisher (RD) is to make prefix values unique across a backbone. IP limits the size of an address to 32 bits in the packet header. The RD adds 64 bits in front, creating a VPNv4 address (in routing tables only). The VPNv4 address (RD plus IP address) must be a globally unique value to avoid conflict with other prefixes.

Selecting RD values can be complex. Some of the issues that impact RD value selection are:

- Scaling — Each unique RD creates a BGP database entry. Prefixes to the same locations that have different RDs are replicated for each RD, causing an expansion of the BGP table.
- Topology— Are you using hub and spoke? If so, then you may want to further investigate RD assignments.

RDs and RTs are only for route exchange between PEs running BGP. To run MPLS VPNs appropriately, PEs must exchange routing information with more fields than usual for IPv4 routes; that extra information includes (but is not limited to) RDs and RTs.

MP-BGP Update Message



MP-BGP Update Message

VPN-IPV4 address (96 bits)

- Route distinguisher (RD) (64 bits)
- Standard IPv4 address (32bits)

Extended community

- Route target (RT) - required
- Site of origin (SOO) - optional

Any other standard BGP attribute

A second label in the label stack

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—7-22

An MP-BGP update message comprises the following elements:

VPN IPv4 address (96 bits)

RD (64 bits)

- Makes the IPv4 route globally unique
- RD is configured in the PE for each VRF
- RD may be related to a site or a VPN
- Standard IPv4 address (32 bits)

Extended community

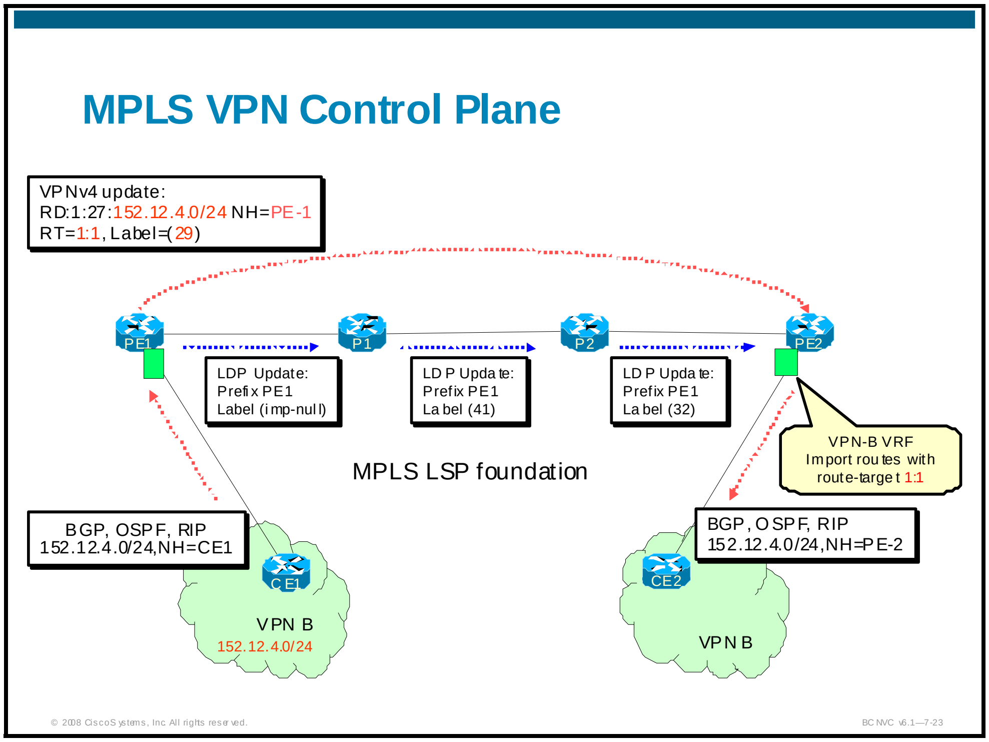
- RT: identifies the set of sites to which a route must be advertised
- Site of origin (SOO): identifies one or more routers where the route has been originated (site) which is used to extend the capability of AS-Path in loop detection in Hub and Spoke topologies with overlapping IP addresses

BGP attributes such as: local preference, MED, next-hop, AS_PATH, standard community

A second label identifying:

- The outgoing interface
- The VRF where a lookup must be done (aggregate label)

MPLS VPN Control Plane



MPLS VPN Control Plane

VPNv4 update:
RD:1:27:152.12.4.0/24 NH=PE-1
RT=1:1, Label=(29)

LDP Update:
Prefix PE1
Label (imp-null)

LDP Update:
Prefix PE1
Label (41)

LDP Update:
Prefix PE1
Label (32)

VPN-B VRF
Import routes with
route-target 1:1

MPLS LSP foundation

BGP, OSPF, RIP
152.12.4.0/24,NH=CE1

BGP, OSPF, RIP
152.12.4.0/24,NH=PE-2

© 2008 Cisco Systems, Inc All rights reserved.

BCNVC v6.1—7-23

1. The MPLS VPN Control Plane relies on MPLS as the foundation. So the first step is to build a label switched path (LSP) from PE to PE.

- All routers (P and PE) run an IGP and label distribution protocol
- Each P- and PE-router has routes for the backbone nodes and a label is associated to each route
  - Think of LDP as the IGP for labels
  - Unique label for each IGP route - locally significant
- Implicit-null says “Remove the top label, I don’t need it” (penultimate hop pop)
  - Improves performance

2. CE-routers advertise a route to PE1

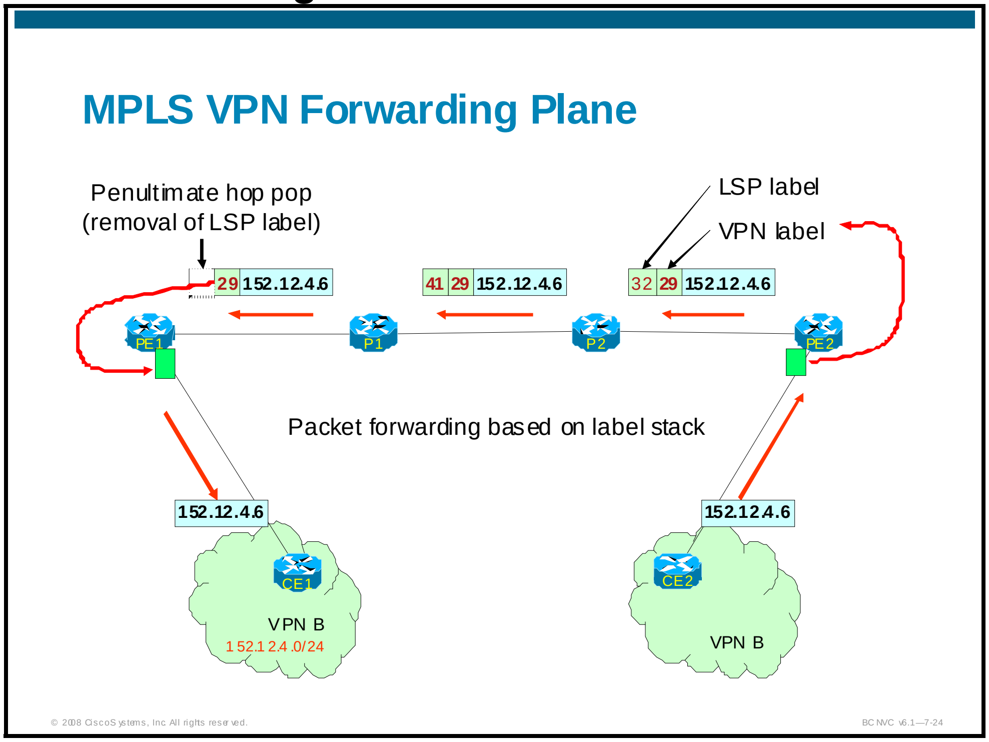
3. PE1 translates into a VPNv4 route

- Assigns a RD, SOO, and RT based on configuration and installs route in VRF
  - Rewrites next-hop attribute (to PE loopback)
  - Assigns a label based on VRF and/or interface
  - Sends MP-IBGP update to all PE neighbors

4. PE2 router translates to IPv4 and advertises it to CE2

- Inserts the route into the VRF identified by the RT
- The label associated to the VPNv4 address is set on packets forwarded towards the destination

MPLS VPN Forwarding Plane



MPLS VPN Forwarding Plane

Penultimate hop pop (removal of LSP label)

LSP label

VPN label

Packet forwarding based on label stack

BCNVC v6.1—7-24

MPLS-VPN uses TWO labels for each packet going to a VPN destination.

The top label is the LDP tag derived from an IGP route corresponding to a PE address (exit point of a VPN route). PE addresses are MP-BGP next-hops of VPN routes.

The second label is the MP-BGP label. It corresponds to the VPN route and identifies the outgoing interface or routing table to be used in order to reach the VPN destination.

In the global tables, PE-routers store IGP routes and associated labels

- Label distributed through LDP

In the VRFs, PE-routers store VPN routes, and associated labels are placed in the LFIB

- Labels are distributed through MP-BGP

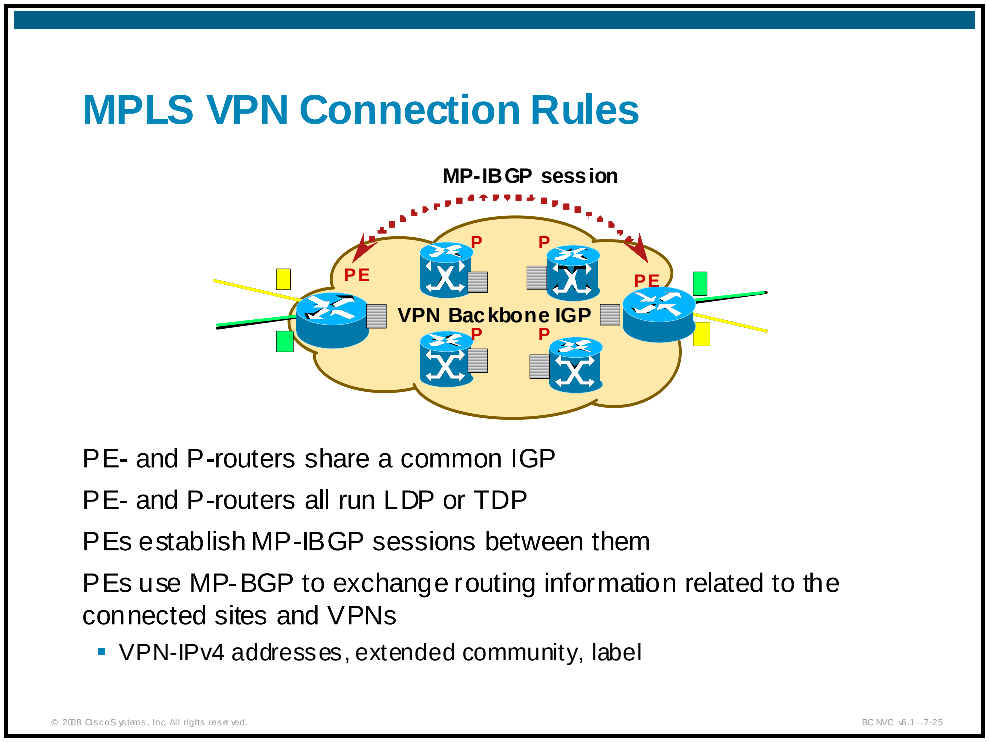
MPLS nodes forward packets based on the top label

- P-routers do not have BGP (or VPN) knowledge
- No VPN routing information

The diagram above depicts the process:

1. Ingress PE1 receives normal IP packets
2. PE1 router performs IP longest match from the VPN FIB, finds IBGP next-hop and imposes a stack of labels <IGP, VPN>
3. The last P-router performs a penultimate hop pop (removes the top tag)
4. Egress PE2 router uses the VPN label to select the VPN/CE interface for packet forwarding
5. VPN label is removed and the packet is routed toward the VPN site

MPLS VPN Connection Rules



MPLS VPN Connection Rules

PE- and P-routers share a common IGP

PE- and P-routers all run LDP or TDP

PEs establish MP-IBGP sessions between them

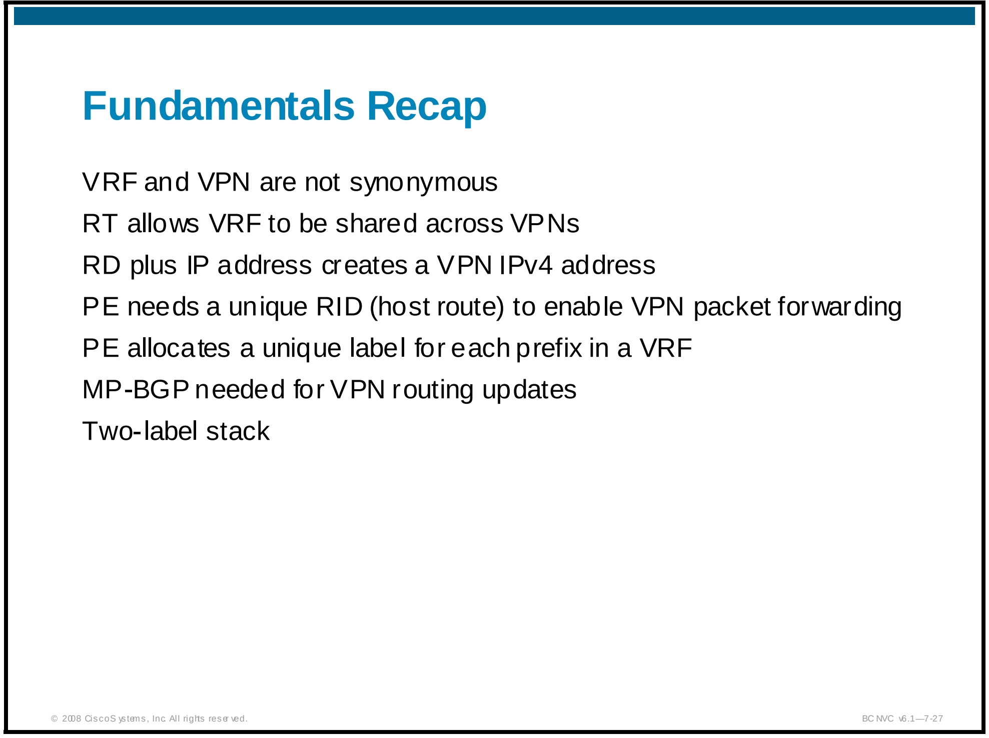
PEs use MP-BGP to exchange routing information related to the connected sites and VPNs

- VPN-IPv4 addresses, extended community, label

BCNVC v6.1—7-25

- The global routing table is populated by IGP protocols
- PE-routers may contain the BGP Internet routes and VPN routes simultaneously
- BGP-4 (IPv4) routes go into global routing table
- MP-BGP (VPN IPv4) routes go into VRFs (contexts)
- LDP must be run on all PE- and P-routers

Fundamentals Recap



Fundamentals Recap

VRF and VPN are not synonymous

RT allows VRF to be shared across VPNs

RD plus IP address creates a VPN IPv4 address

PE needs a unique RID (host route) to enable VPN packet forwarding

PE allocates a unique label for each prefix in a VRF

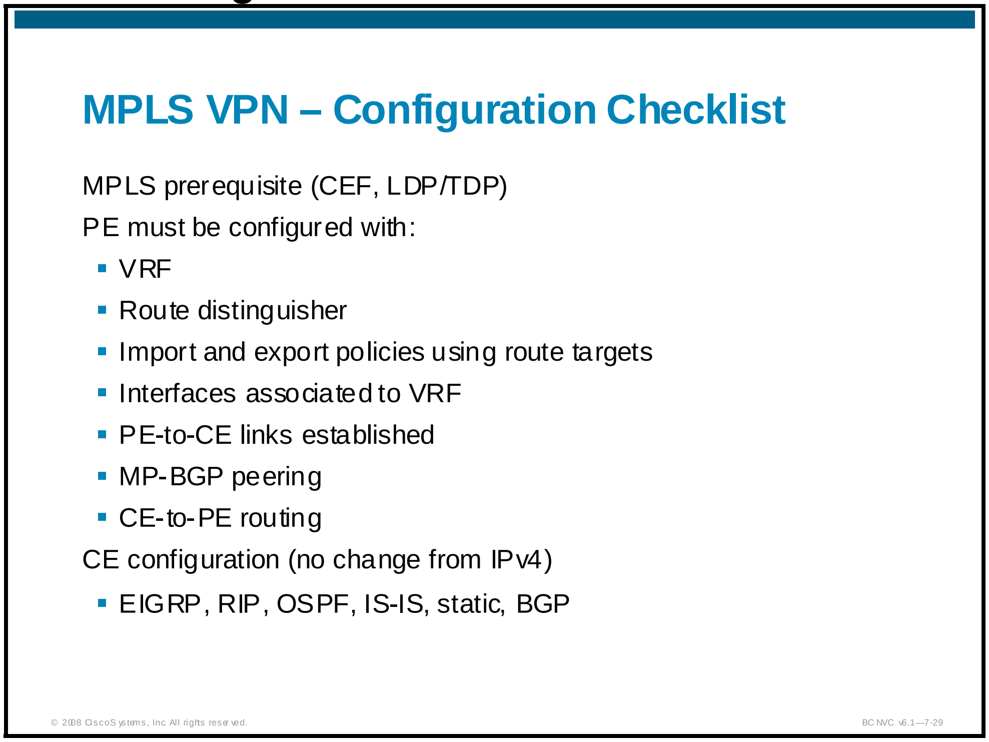
MP-BGP needed for VPN routing updates

Two-label stack

BCNVC v6.1—7-27

How Do I Configure MPLS VPNs?

MPLS VPN – Configuration Checklist



MPLS VPN – Configuration Checklist

MPLS prerequisite (CEF, LDP/TDP)

PE must be configured with:

- VRF
- Route distinguisher
- Import and export policies using route targets
- Interfaces associated to VRF
- PE-to-CE links established
- MP-BGP peering
- CE-to-PE routing

CE configuration (no change from IPv4)

- EIGRP, RIP, OSPF, IS-IS, static, BGP

The checklist above shows the minimum configuration items needed for MPLS VPNs.

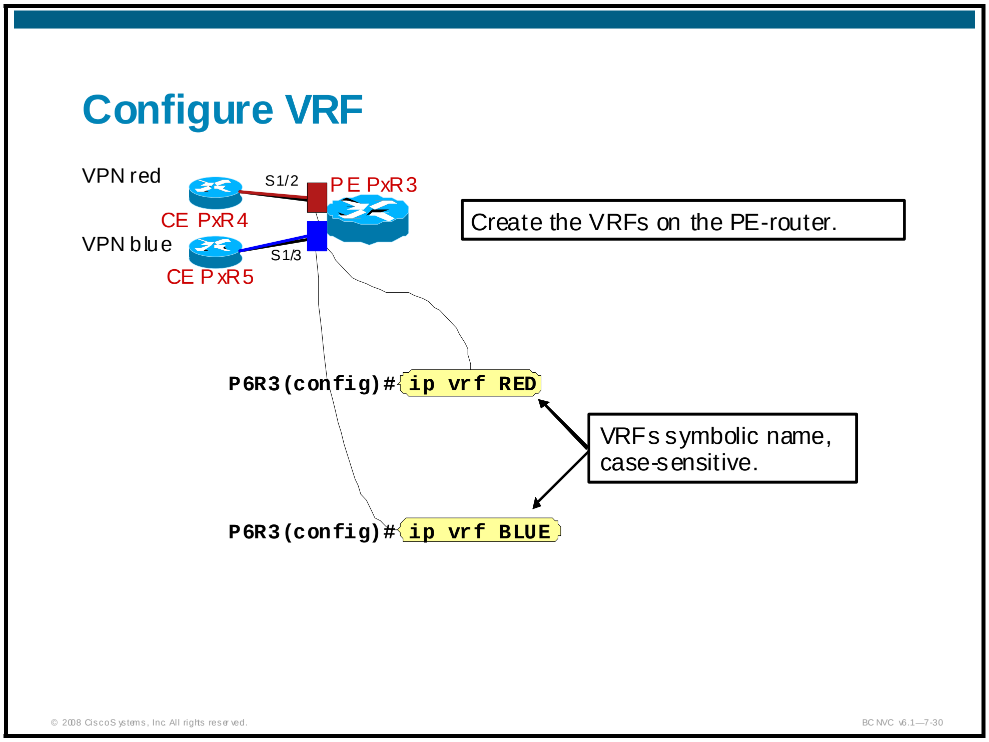
There may be access lists, reverse path forwarding, BGP filtering, OSPF areas, and other items in a configuration.

A provider-managed CE may have a quality of service (QoS) access list also.

The CE may be customer-managed (the service provider cannot access it) or service-provider-managed (easier to troubleshoot because a service provider can access it).

Two hundred to three hundred line configurations are not unusual and can be hard to troubleshoot. Try to keep these basics in mind.

Configure VRF



Configure VRF

Create the VRFs on the PE-router.

```
P6R3(config)#ip vrf RED
```

VRFs symbolic name, case-sensitive.

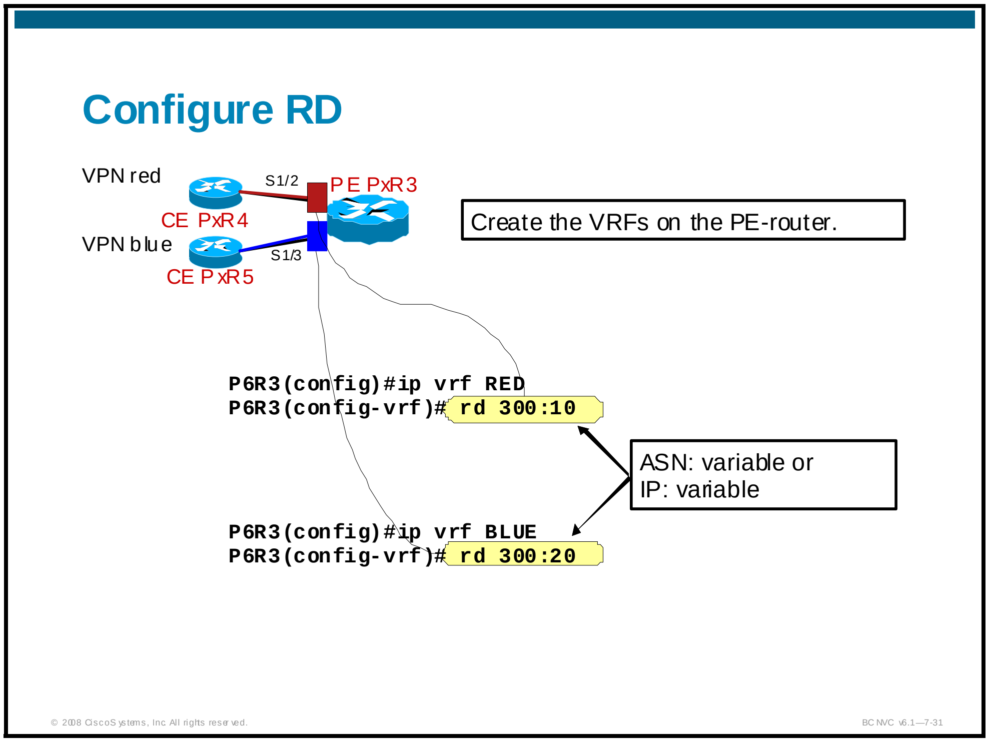
```
P6R3(config)#ip vrf BLUE
```

A VRF is given a symbolic name. Select a value that makes sense to your organization or application.

Create one VRF for each VPN connected using this command:

ip vrf <VPN routing/forwarding instance name>

Configure RD



Configure RD

Create the VRFs on the PE-router.

```
P6R3(config)#ip vrf RED
P6R3(config-vrf)# rd 300:10
```

ASN: variable or
IP: variable

```
P6R3(config)#ip vrf BLUE
P6R3(config-vrf)# rd 300:20
```

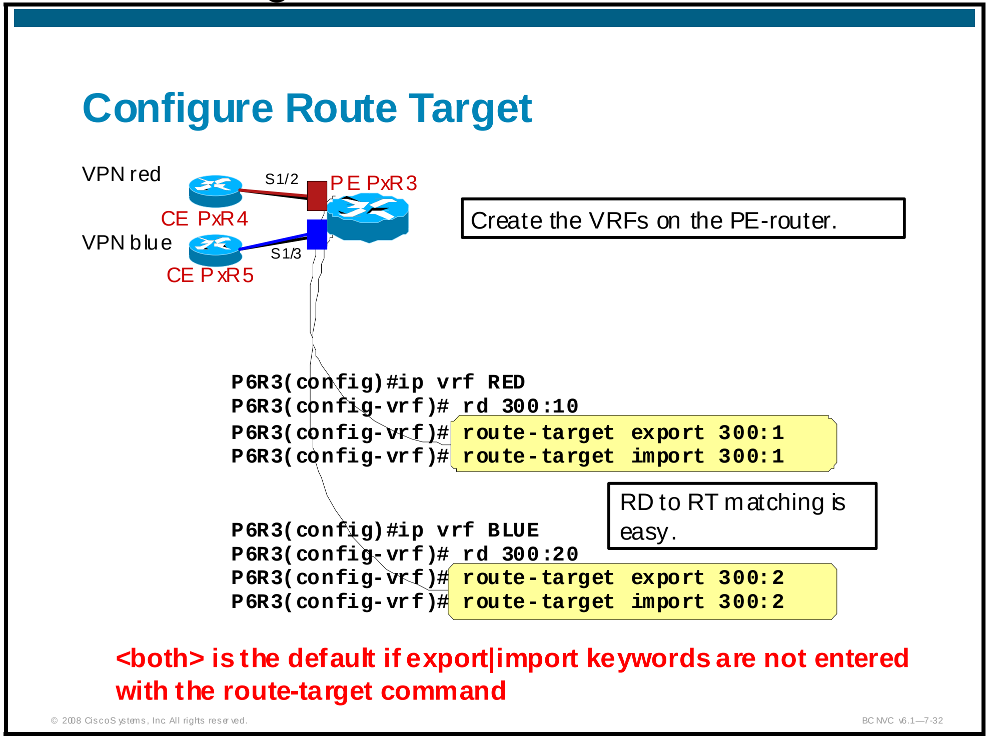
BCNVC v6.1—7-31

Specify the correct route distinguisher for the VPN. This extends the IP address.

```
rd <VPN route distinguisher>
```

- ASN:nn— Convention is to use the ASN for the first value and any logical scheme for the second.
- A common practice is to use the same RD for the same VPN in all PEs (not required).

Configure Route Target



Configure Route Target

Create the VRFs on the PE-router.

```
P6R3(config)#ip vrf RED
P6R3(config-vrf)# rd 300:10
P6R3(config-vrf)# route-target export 300:1
P6R3(config-vrf)# route-target import 300:1

P6R3(config)#ip vrf BLUE
P6R3(config-vrf)# rd 300:20
P6R3(config-vrf)# route-target export 300:2
P6R3(config-vrf)# route-target import 300:2
```

RD to RT matching is easy.

<both> is the default if export|import keywords are not entered with the route-target command

BCNVC v6.1—7-32

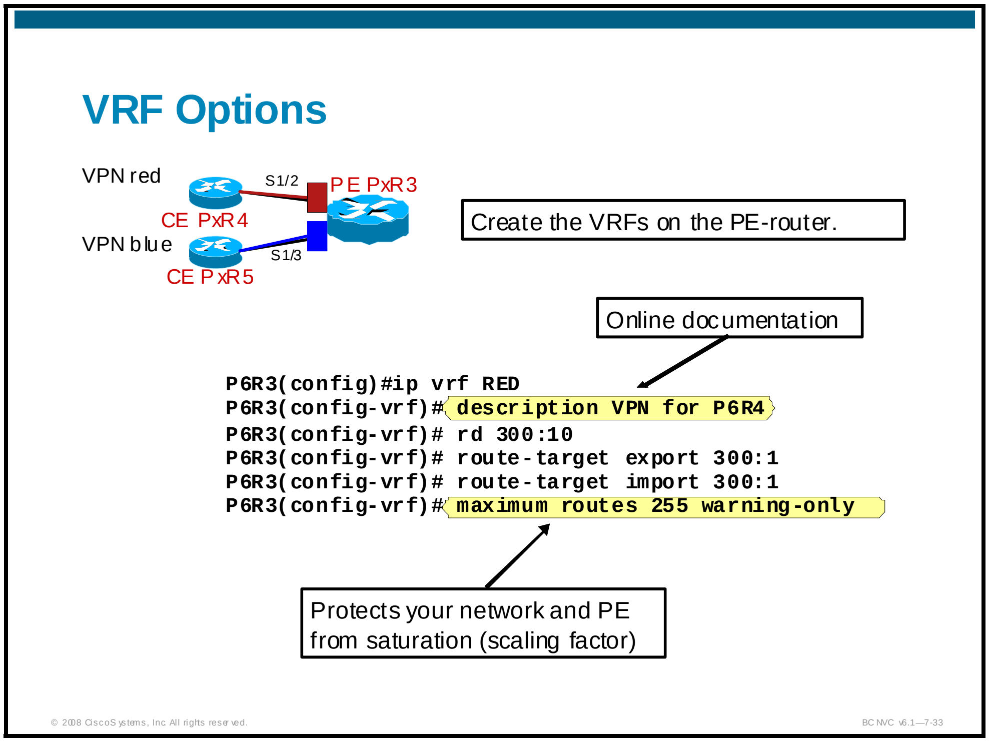
Set up the import and export properties for the MP-BGP extended communities. These are used for filtering the import and export process.

```
route-target [export|import|both] <target VPN extended
community>
```

Matching the values for RD and RT makes it easier to keep track of the values, but is not required.

Note In our examples and lab exercises, we use different values in order to make it clear which value is associated to the different entities.

VRF Options



VRF Options

Create the VRFs on the PE-router.

Online documentation

```
P6R3(config)#ip vrf RED
P6R3(config-vrf)# description VPN for P6R4
P6R3(config-vrf)# rd 300:10
P6R3(config-vrf)# route-target export 300:1
P6R3(config-vrf)# route-target import 300:1
P6R3(config-vrf)# maximum routes 255 warning-only
```

Protects your network and PE from saturation (scaling factor)

BCNVC v6.1—7-33

- Descriptions are always recommended. You may know what something is, but another person may not.
- To prevent a CE from flooding you with routing updates, use the max-path setting in BGP

Maximum Routes

To limit the maximum number of routes in a VRF to prevent a PE-router from importing too many routes, use the **maximum routes** command in VRF configuration sub mode. To remove the limit on the maximum number of routes allowed, use the no form of this command.

maximum routes limit {warn threshold | warn-only}

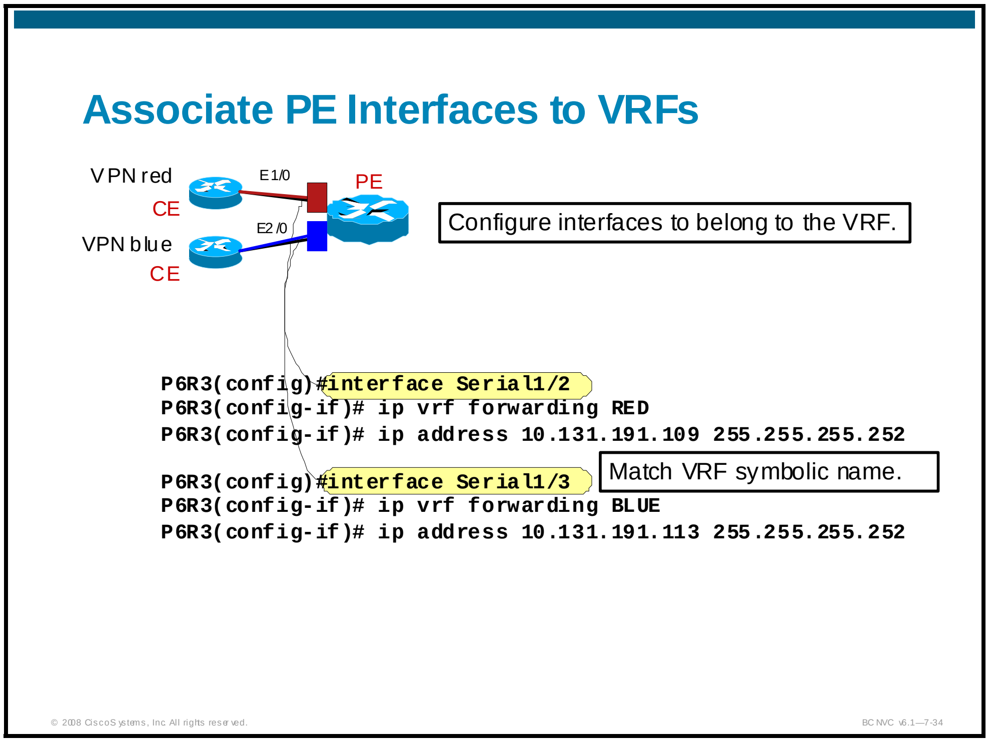
Syntax description

limit	Specifies the maximum number of routes allowed in a VRF. You may select from 1 to 4,294,967,295 routes to be allowed in a VRF.
warn threshold	Rejects routes when the threshold limit is reached. The threshold limit is a percentage of the limit specified, from 1 to 100.
warn-only	Issues a syslog error message when the maximum number of routes allowed for a VRF exceeds the threshold. However, additional routes are still allowed.

In the following example, the route distinguisher ASN is 100, and the maximum number of VRF routes allowed is set to 1000. When the maximum routes for the VRF reaches 1000, the router issues a syslog error message, but continues to accept new VRF routes.

```
ip vrf vrf1
rd 100:1
route-target import 100:1
maximum routes 1000 warn-only
```

Associate PE Interfaces to VRFs



Associate PE Interfaces to VRFs

Configure interfaces to belong to the VRF.

```
P6R3(config)#interface Serial1/2
P6R3(config-if)# ip vrf forwarding RED
P6R3(config-if)# ip address 10.131.191.109 255.255.255.252

P6R3(config)#interface Serial1/3
P6R3(config-if)# ip vrf forwarding BLUE
P6R3(config-if)# ip address 10.131.191.113 255.255.255.252
```

Match VRF symbolic name.

BCNVC v6.1—7-34

After you define all relevant VRFs on the PE-router, you must tell the PE-router which interfaces belong to which VRF and, therefore, should populate the VRF with routes from connected sites. More than one interface can belong to the same VRF.

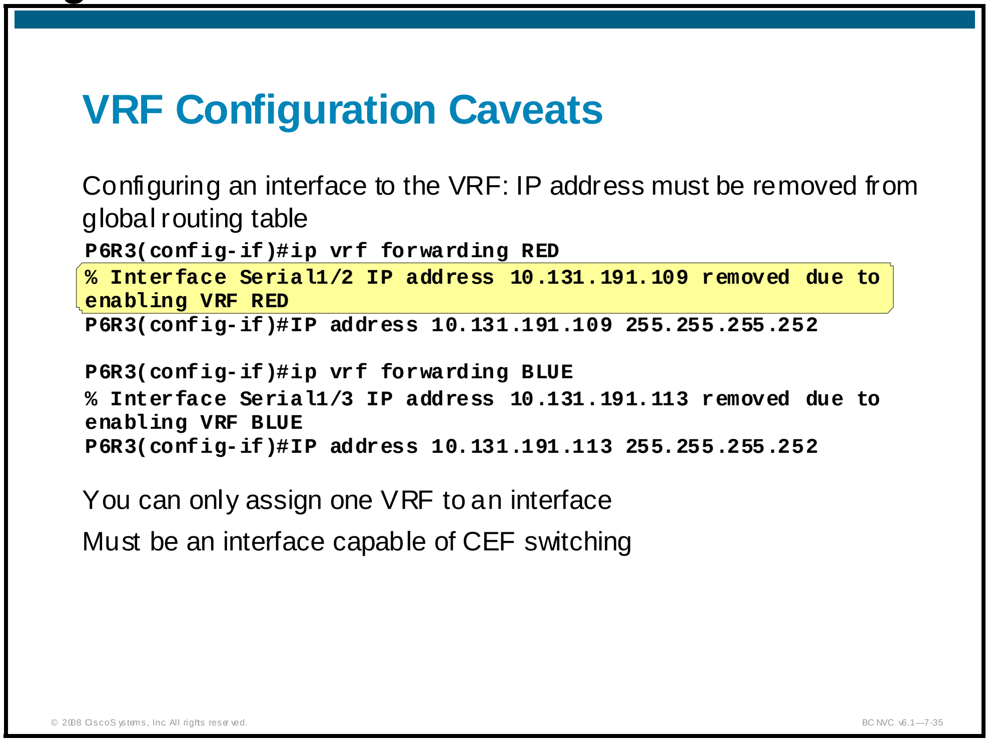
Do this by using the **ip vrf forwarding interface-mode** command, which associates the interface with the named VRF. Both main and subinterfaces can be defined within a VRF. Configure the forwarding details for the respective interfaces using the following command.

ip vrf forwarding <VPN routing/forwarding instance name>

Remember to set the IP address after doing this.

Depending on the PE-to-CE routing protocol, you can configure static routes or routing protocols (RIP, OSPF, EIGRP, or BGP) between the PE and CE devices.

VRF Configuration Caveats



VRF Configuration Caveats

Configuring an interface to the VRF: IP address must be removed from global routing table

```
P6R3(config-if)#ip vrf forwarding RED
% Interface Serial1/2 IP address 10.131.191.109 removed due to
enabling VRF RED
P6R3(config-if)#IP address 10.131.191.109 255.255.255.252

P6R3(config-if)#ip vrf forwarding BLUE
% Interface Serial1/3 IP address 10.131.191.113 removed due to
enabling VRF BLUE
P6R3(config-if)#IP address 10.131.191.113 255.255.255.252
```

You can only assign one VRF to an interface

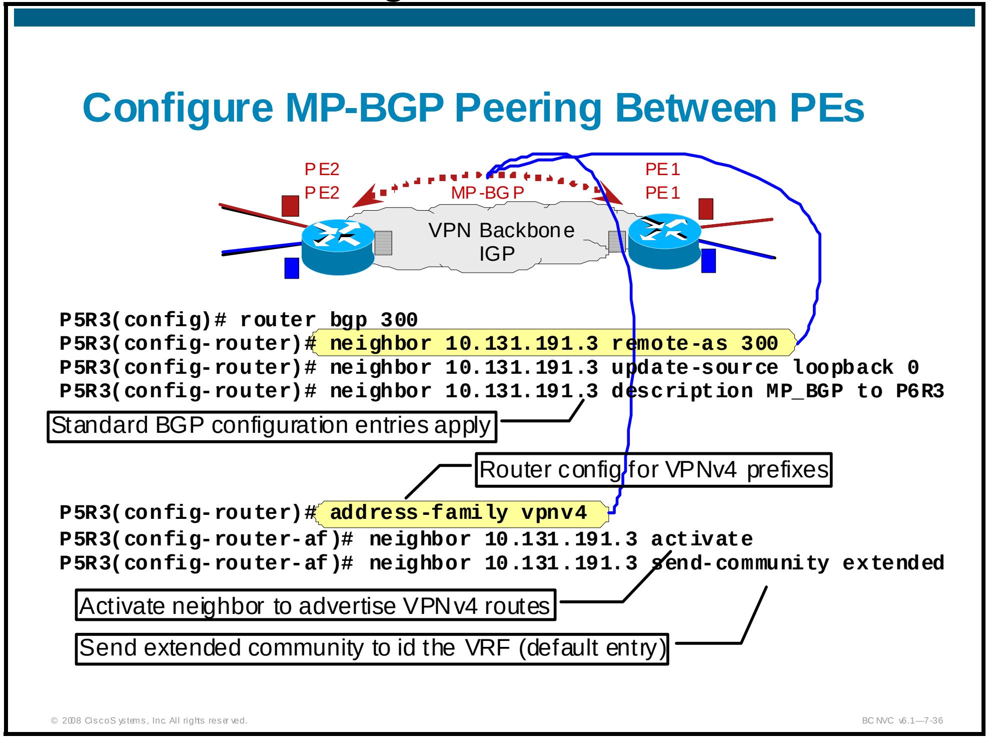
Must be an interface capable of CEF switching

BCNVC v6.1—7-35

When an interface is associated with a particular VRF, its IP address is removed from the global routing table and from the interface. This is because an assumption is made that the address is not valid across multiple routing tables and should be reconfigured after the interface is given membership to a VRF.

Only interfaces that run CEF switching can be associated with VRFs because the CEF switching mechanism is a necessary prerequisite for successful MPLS VPN data forwarding as label imposition is achieved through the CEF switching path.

Configure MP-BGP Peering Between PEs



Configure MP-BGP Peering Between PEs

```
P5R3(config)# router bgp 300
P5R3(config-router)# neighbor 10.131.191.3 remote-as 300
P5R3(config-router)# neighbor 10.131.191.3 update-source loopback 0
P5R3(config-router)# neighbor 10.131.191.3 description MP_BGP to P6R3
```

Standard BGP configuration entries apply

Router config for VPNv4 prefixes

```
P5R3(config-router)# address-family vpnv4
P5R3(config-router-af)# neighbor 10.131.191.3 activate
P5R3(config-router-af)# neighbor 10.131.191.3 send-community extended
```

Activate neighbor to advertise VPNv4 routes

Send extended community to id the VRF (default entry)

BGP configuration requires several steps and various configuration commands. It must be configured for any PE-to-PE MP-IBGP sessions across the MPLS VPN backbone, and for any PE-to-CE EBGP sessions for customers that want to run BGP with the service provider.

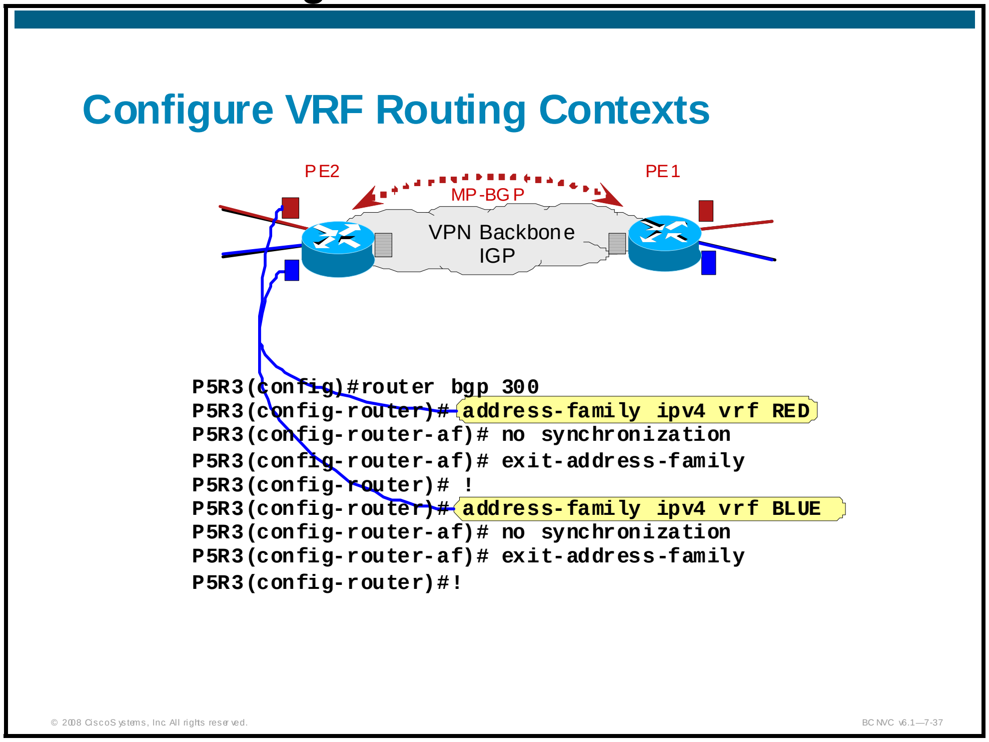
As part of the MP-BGP specification (RFC 2283), create an address-family to allow BGP to carry protocols other than IPv4. Within the MPLS VPN architecture, this address-family is the VPN-IPv4 address and BGP must be told that this type of address-family is carried by one of its sessions.

The default behavior when a BGP session is configured on a Cisco router is to activate the session to carry IPv4 unicast prefixes. This might represent a problem in a pure MPLS VPN environment where BGP is used solely to carry VPN-IPv4. A new command reverses this behavior so that the activation of any BGP sessions, whether IPv4 or VPN-IPv4, does not occur by default.

There are several ways to configure MP-BGP between PE-routers, such as using a route reflector or confederation. The method used here—direct neighbor configuration—is the simplest and least scalable.

- Declare the different neighbors.
- Enter the address-family VPNv4 mode and complete the following steps:
  - Activate the neighbors.
  - Specify that extended community must be used. This is mandatory.

Configure VRF Routing Contexts

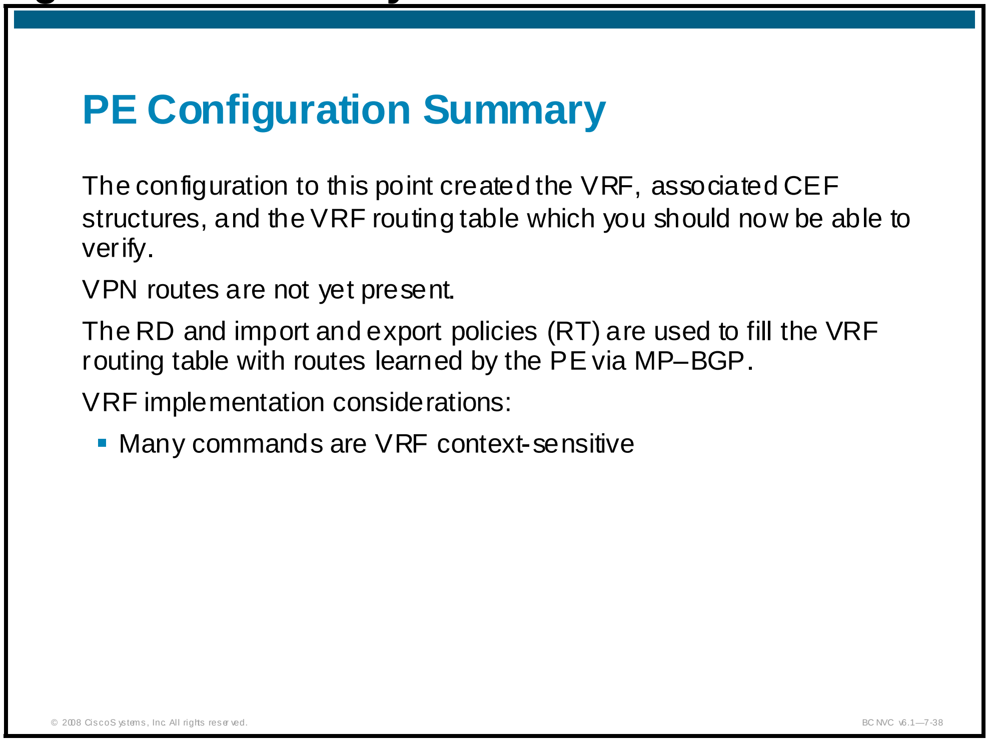


Configure VRF Routing Contexts

```
P5R3(config)#router bgp 300
P5R3(config-router)# address-family ipv4 vrf RED
P5R3(config-router-af)# no synchronization
P5R3(config-router-af)# exit-address-family
P5R3(config-router)# !
P5R3(config-router)# address-family ipv4 vrf BLUE
P5R3(config-router-af)# no synchronization
P5R3(config-router-af)# exit-address-family
P5R3(config-router)#!
```

Standard BGP entries apply here. Modifications depend on routing requirements of the connected CE.

PE Configuration Summary



PE Configuration Summary

The configuration to this point created the VRF, associated CEF structures, and the VRF routing table which you should now be able to verify.

VPN routes are not yet present.

The RD and import and export policies (RT) are used to fill the VRF routing table with routes learned by the PE via MP–BGP.

VRF implementation considerations:

- Many commands are VRF context-sensitive

VRF Implementation Considerations

When implementing VPNs and VRFs, keep the following considerations in mind:

A local VRF interface on a PE is not considered a directly connected interface in a traditional sense. For example, when you configure a Fast Ethernet interface on a PE to participate in a particular VRF/VPN, the interface no longer shows up as a directly connected interface when you issue a **show ip route** command. To see that interface in a routing table, you must issue a **show ip route vrf vrf_name** command.

The global routing table and per-VRF routing table are independent entities. Cisco IOS commands apply to IP routing in a global routing table context. For example, `show ip route` and other EXEC-level show commands—as well as utilities such as ping, traceroute, and Telnet---all use global IP routing table.

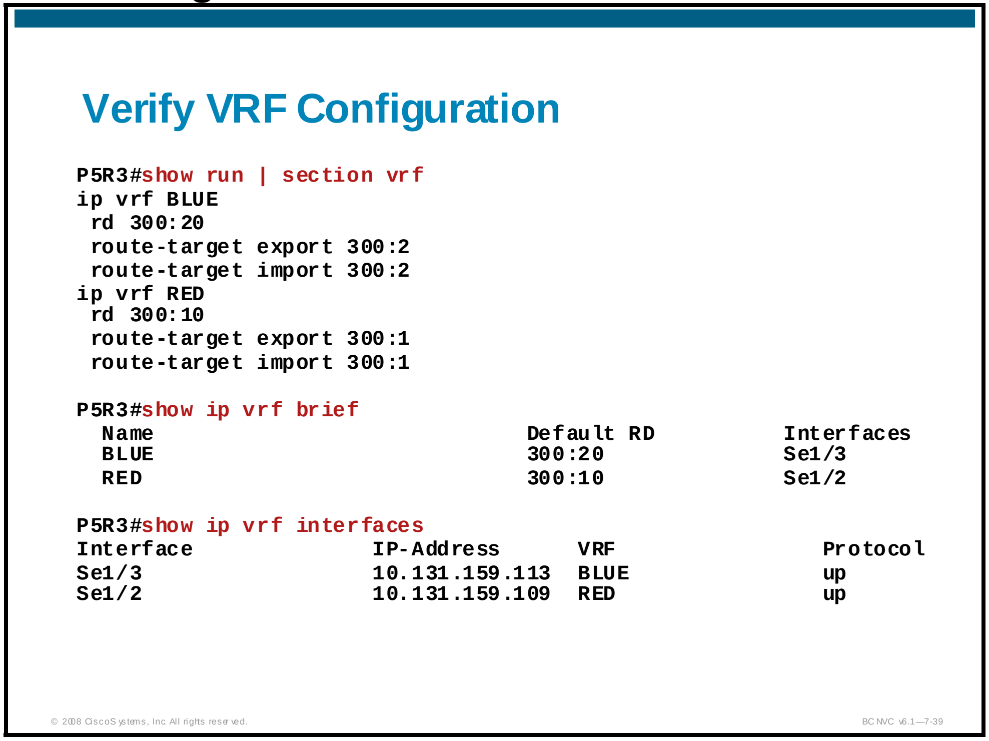
You can issue a standard Telnet command from a CE-router to connect to a PE-router. However, from that PE, you must issue the following command to connect from the PE to the CE:

telnet CERouterName /vrf vrf_name

Similarly, you can use the **traceroute** and **ping** commands in a VRF context.

The MPLS VPN backbone relies on the appropriate IGP that is configured for MPLS, for example, EIGRP or OSPF. When you issue a **show ip route** command on a PE, you see the IGP-derived routes connecting the PEs together. Contrast that with the **show ip route vrf VRF_name** command, which displays routes connecting customer sites in a particular VPN.

Verify VRF Configuration

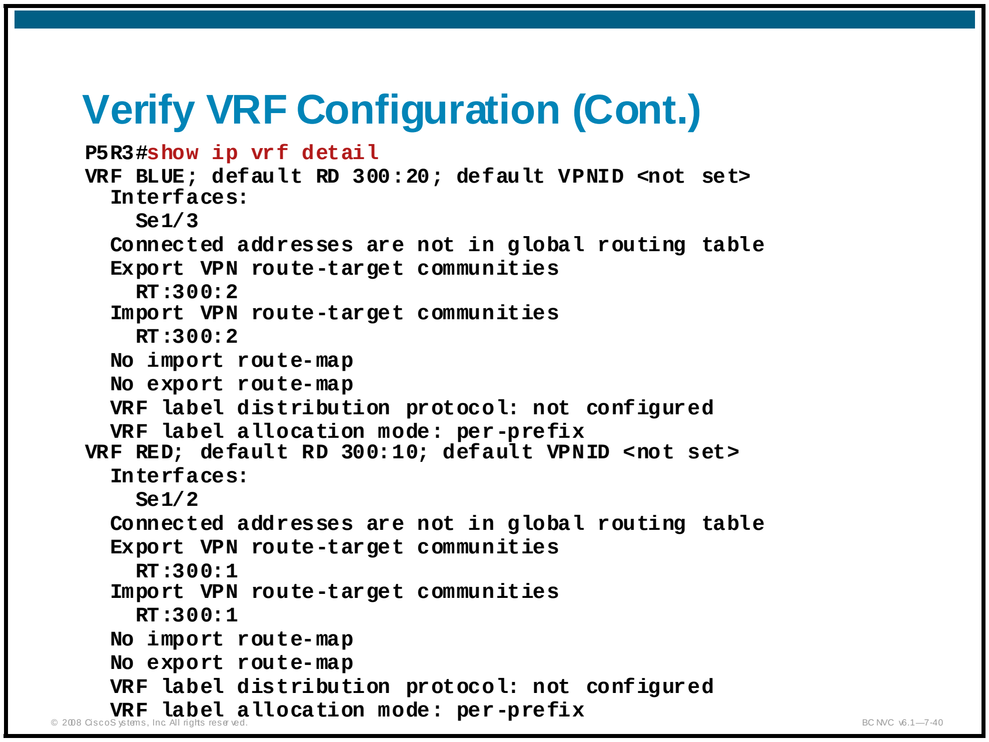


Verify VRF Configuration

```
P5R3#show run | section vrf
ip vrf BLUE
 rd 300:20
 route-target export 300:2
 route-target import 300:2
ip vrf RED
 rd 300:10
 route-target export 300:1
 route-target import 300:1

P5R3#show ip vrf brief
  Name                             Default RD          Interfaces
  BLUE                             300:20              Se1/3
  RED                              300:10              Se1/2

P5R3#show ip vrf interfaces
Interface              IP-Address      VRF                 Protocol
Se1/3                  10.131.159.113  BLUE                up
Se1/2                  10.131.159.109  RED                 up
```

BC NVC v6.1—7-39

Check that all of your configuration is as expected.



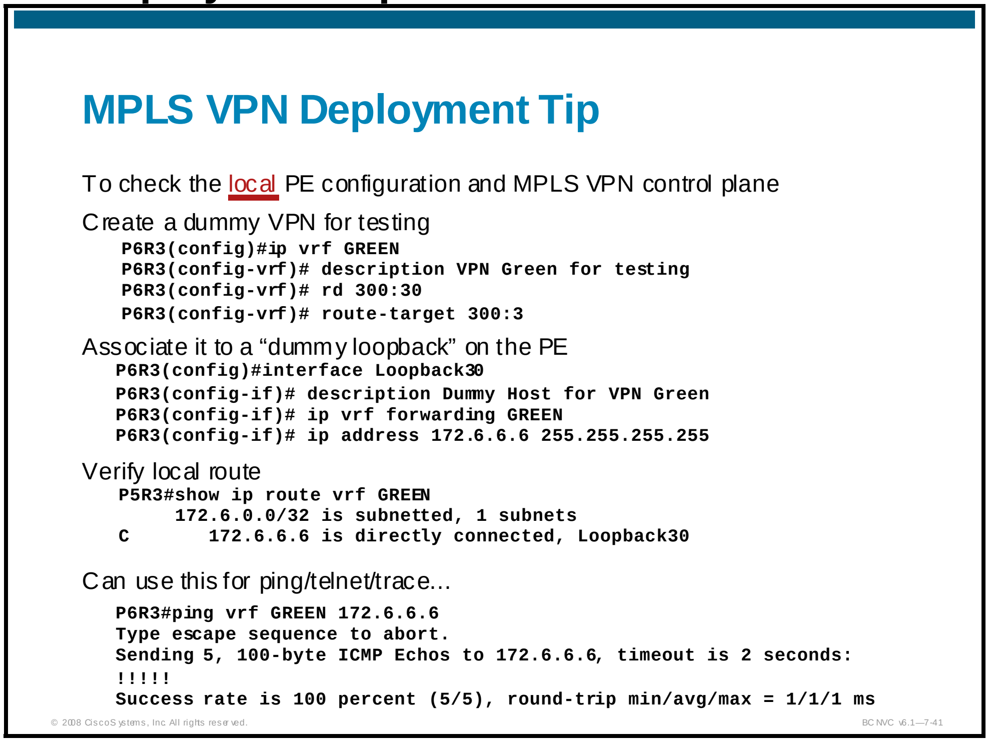
Verify VRF Configuration (Cont.)

```
P5R3#show ip vrf detail
VRF BLUE; default RD 300:20; default VPNID <not set>
  Interfaces:
    Se1/3
  Connected addresses are not in global routing table
  Export VPN route-target communities
    RT:300:2
  Import VPN route-target communities
    RT:300:2
  No import route-map
  No export route-map
  VRF label distribution protocol: not configured
  VRF label allocation mode: per-prefix
VRF RED; default RD 300:10; default VPNID <not set>
  Interfaces:
    Se1/2
  Connected addresses are not in global routing table
  Export VPN route-target communities
    RT:300:1
  Import VPN route-target communities
    RT:300:1
  No import route-map
  No export route-map
  VRF label distribution protocol: not configured
  VRF label allocation mode: per-prefix
```

BC NVC v6.1—7-40

The details keyword provides additional information. Much of this output would be useful for troubleshooting; for example, if a policy was applied that blocks traffic.

MPLS VPN Deployment Tip



MPLS VPN Deployment Tip

To check the local PE configuration and MPLS VPN control plane

Create a dummy VPN for testing

```
P6R3(config)#ip vrf GREEN
P6R3(config-vrf)# description VPN Green for testing
P6R3(config-vrf)# rd 300:30
P6R3(config-vrf)# route-target 300:3
```

Associate it to a “dummy loopback” on the PE

```
P6R3(config)#interface Loopback30
P6R3(config-if)# description Dummy Host for VPN Green
P6R3(config-if)# ip vrf forwarding GREEN
P6R3(config-if)# ip address 172.6.6.6 255.255.255.255
```

Verify local route

```
P5R3#show ip route vrf GREEN
     172.6.0.0/32 is subnetted, 1 subnets
C       172.6.6.6 is directly connected, Loopback30
```

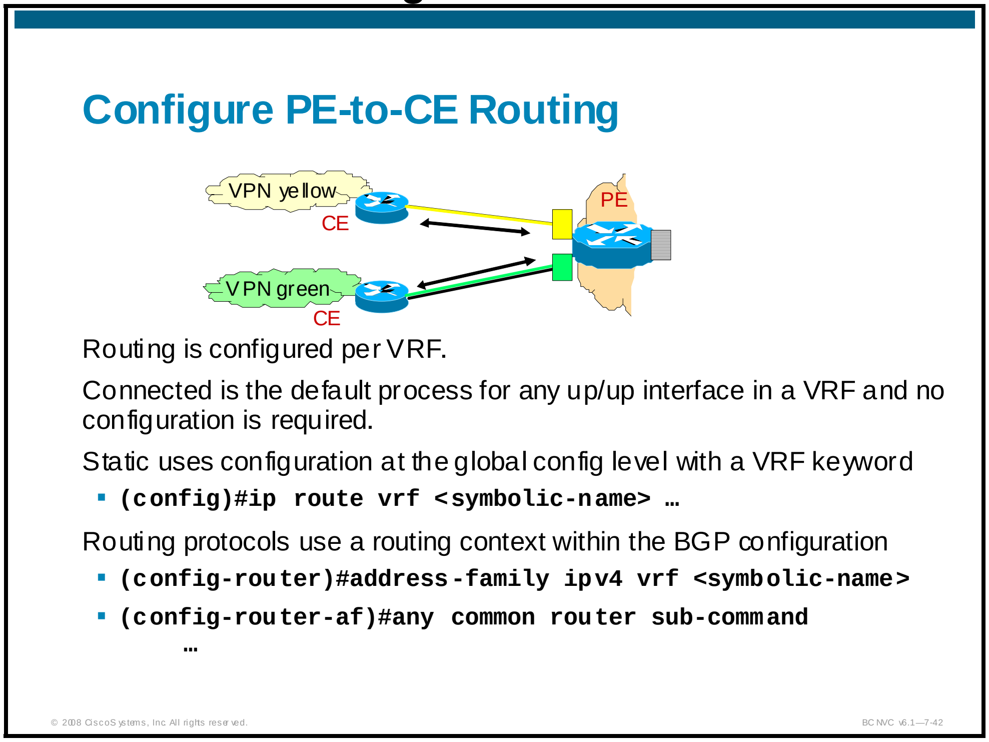
Can use this for ping/telnet/trace…

```
P6R3#ping vrf GREEN 172.6.6.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.6.6.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

It is not uncommon to get confused about your MPLS VPN configuration, especially when you are first starting. If the VPN is not functional, try to determine if there is a problem with the PE, the CE, or the links. This deployment tip eliminates any dependence on configuration outside of the PE-router.

Additionally, if other PE-routers have a similar configuration, then end-to-end tests can be performed.

Configure PE-to-CE Routing



Configure PE-to-CE Routing

Routing is configured per VRF.

Connected is the default process for any up/up interface in a VRF and no configuration is required.

Static uses configuration at the global config level with a VRF keyword

- **(config)#ip route vrf <symbolic-name> …**

Routing protocols use a routing context within the BGP configuration

- **(config-router)#address-family ipv4 vrf <symbolic-name>**
- **(config-router-af)#any common router sub-command**

  …

BCNVC v6.1—7-42

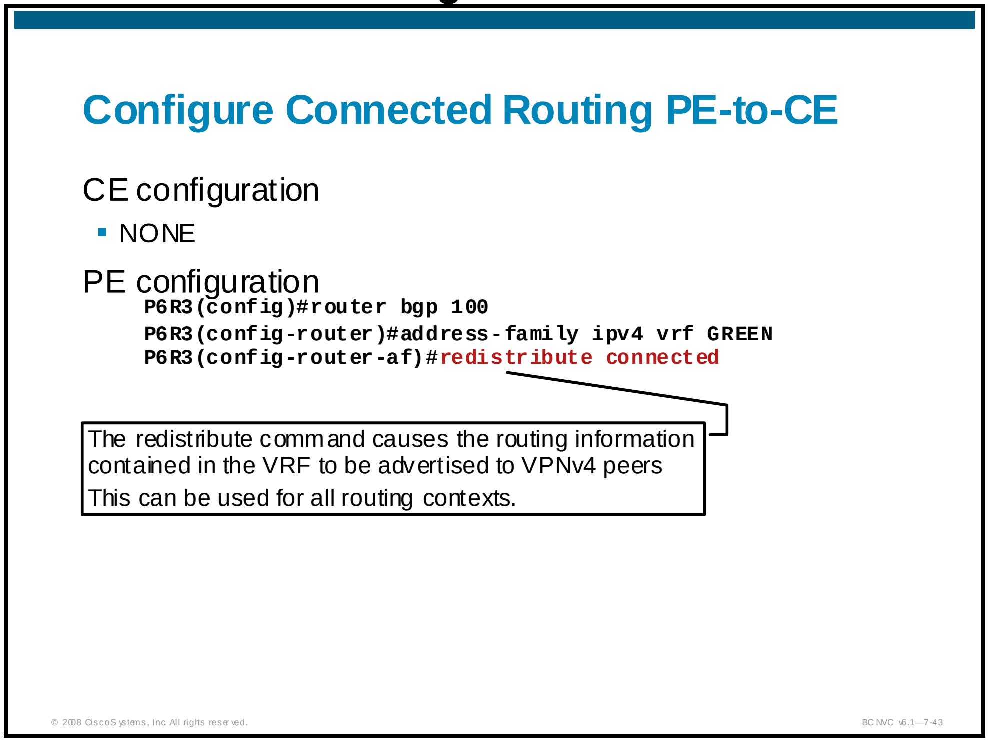
Static and dynamic routing protocols can be configured to distribute IP prefixes between a customer edge (CE) router at a VPN site and the associated provider edge (PE) router.

The routes statically defined or dynamically learned over a particular interface are inserted into the associated virtual routing and forwarding (VRF) tables within the PEs. Routes learned over a particular VRF interface are inserted only into the associated VRF routing table. The routes are not inserted into the global routing table or other VRF routing tables unless they are imported using the associated RT.

Standardizing on a PE-to-CE routing protocol facilitates administration and provides homogeneity. Choose among the following routing protocols:

- Connected
- Static routing
- RIP
- EBGP-4
- OSPF
- EIGRP

Configure Connected Routing PE-to-CE



Configure Connected Routing PE-to-CE

CE configuration

- NONE

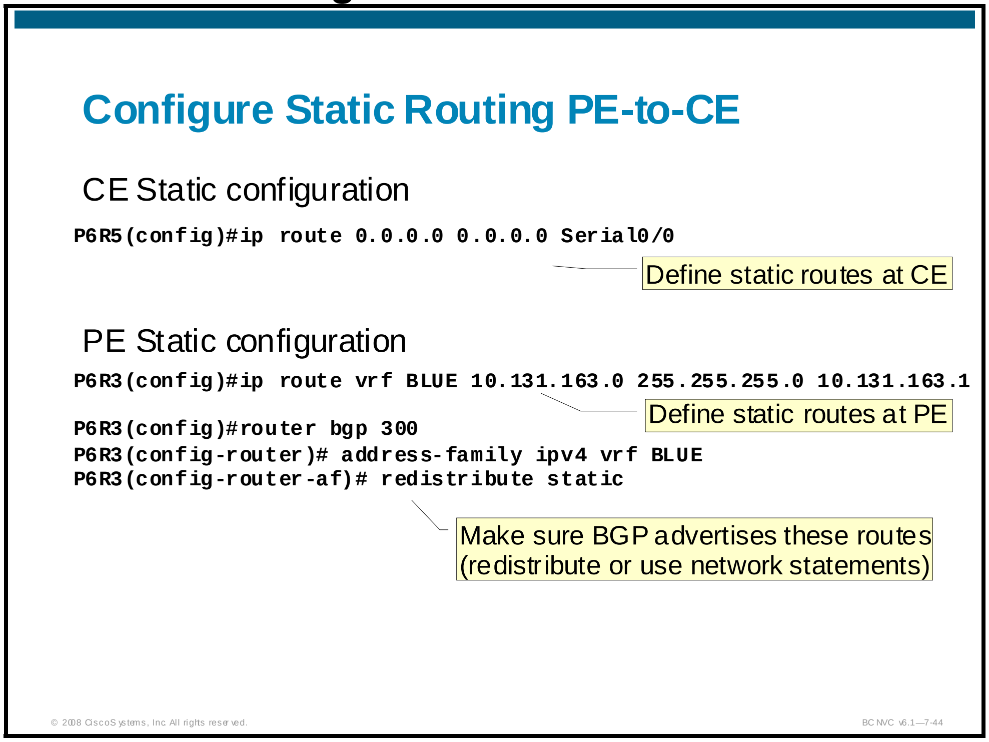
PE configuration

```
P6R3(config)#router bgp 100
P6R3(config-router)#address-family ipv4 vrf GREEN
P6R3(config-router-af)#redistribute connected
```

The redistribute command causes the routing information contained in the VRF to be advertised to VPNv4 peers

This can be used for all routing contexts.

Configure Static Routing PE-to-CE



Configure Static Routing PE-to-CE

CE Static configuration

```
P6R5(config)#ip route 0.0.0.0 0.0.0.0 Serial0/0
```

Define static routes at CE

PE Static configuration

```
P6R3(config)#ip route vrf BLUE 10.131.163.0 255.255.255.0 10.131.163.1

P6R3(config)#router bgp 300
P6R3(config-router)# address-family ipv4 vrf BLUE
P6R3(config-router-af)# redistribute static
```

Define static routes at PE

Make sure BGP advertises these routes (redistribute or use network statements)

Static routing is simple, stable, and does not require much router resources. However, static routing does not provide dynamic rerouting and requires additional configuration for every new route both on the CE (except when a default route can be used) and the PE. Use static routing whenever possible.

Static routes can be configured on the CE-router pointing to the PE VRF interface. This can be a default route to the rest of the customer VPN.

Specific VRF-based static routes can also be configured on the PE-router pointing to the CE-router by way of the VRF interface and IP address.

Static routing implies performing the following two configuration tasks:

- Configure one/many static routes on the CE to reach other VPN sites
- Configure one/many static routes in the PE VRF to reach the different subnetworks of the attached VPN site
- Route summarization for connected interfaces (PE-to-CE links) is recommended.

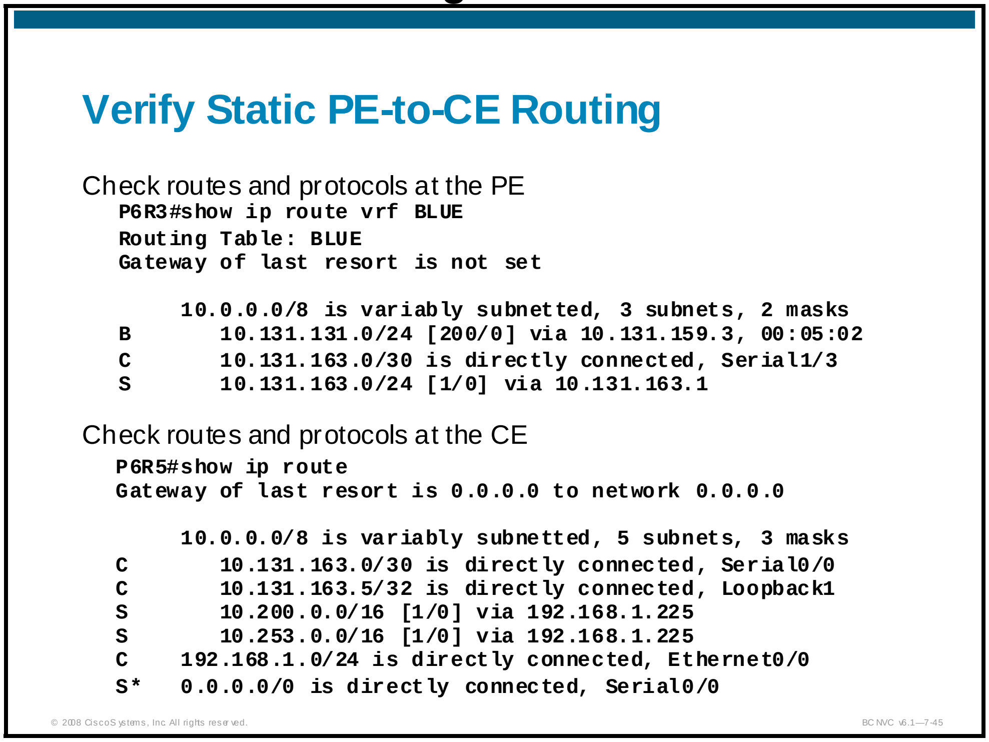
Network Stability with Static Routing

Static routing requires no protection mechanisms to maintain network stability.

- Routing instability from the CE—None
- Number of routing updates coming from the CE (number of routes)—Controlled on a manual basis

Note When a PE-to-CE link fails, the static route associated to the interface is removed and an MP-IBGP route update is sent by the PE to every MP-IBGP peer. To prevent this, use the `permanent` keyword when configuring the static route.

Verify Static PE-to-CE Routing



Verify Static PE-to-CE Routing

Check routes and protocols at the PE

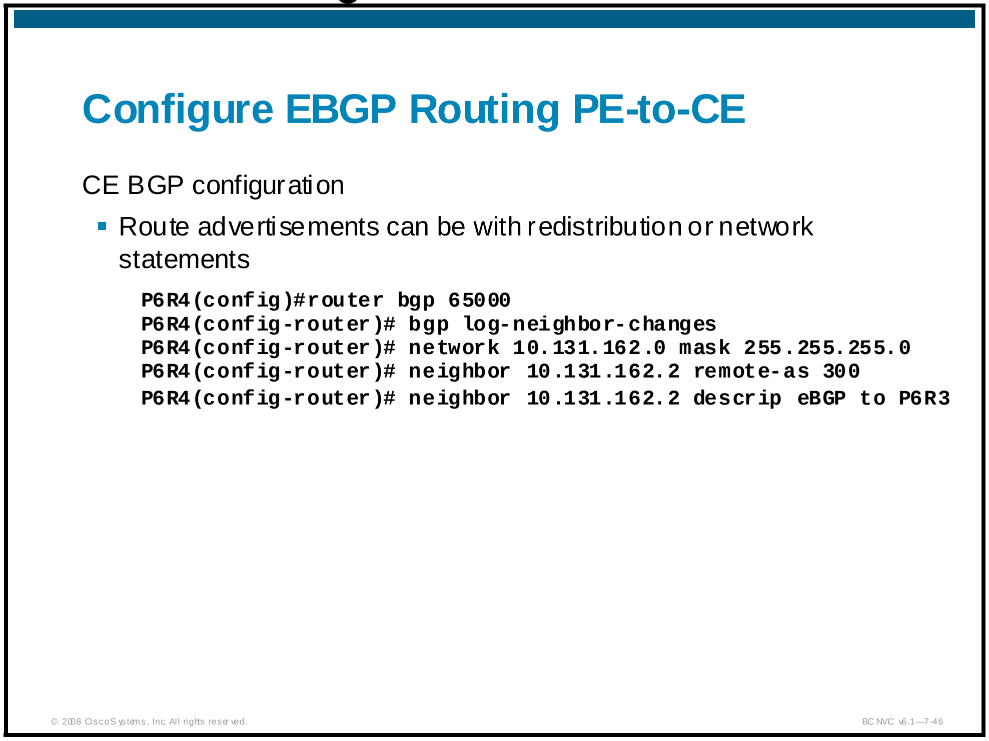
```
P6R3#show ip route vrf BLUE
Routing Table: BLUE
Gateway of last resort is not set

     10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
B       10.131.131.0/24 [200/0] via 10.131.159.3, 00:05:02
C       10.131.163.0/30 is directly connected, Serial1/3
S       10.131.163.0/24 [1/0] via 10.131.163.1
```

Check routes and protocols at the CE

```
P6R5#show ip route
Gateway of last resort is 0.0.0.0 to network 0.0.0.0

     10.0.0.0/8 is variably subnetted, 5 subnets, 3 masks
C       10.131.163.0/30 is directly connected, Serial0/0
C       10.131.163.5/32 is directly connected, Loopback1
S       10.200.0.0/16 [1/0] via 192.168.1.225
S       10.253.0.0/16 [1/0] via 192.168.1.225
C    192.168.1.0/24 is directly connected, Ethernet0/0
S*   0.0.0.0/0 is directly connected, Serial0/0
```

BCNVC v6.1—7-45

Configure EBGP Routing PE-to-CE



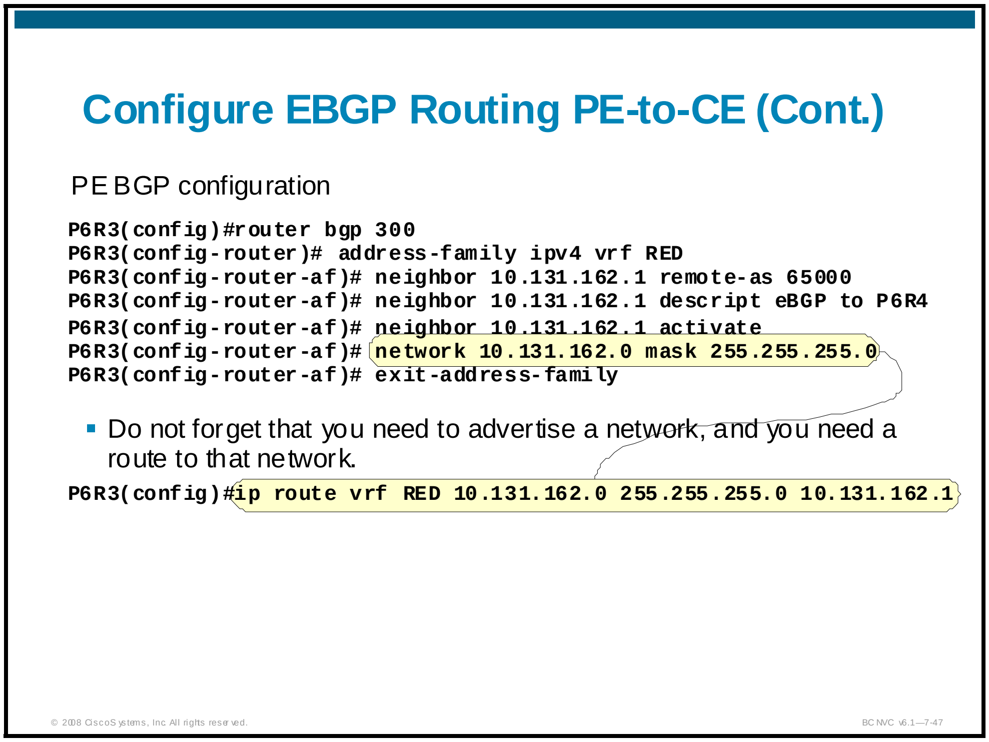
Configure EBGP Routing PE-to-CE

CE BGP configuration

- Route advertisements can be with redistribution or network statements

```
P6R4(config)#router bgp 65000
P6R4(config-router)# bgp log-neighbor-changes
P6R4(config-router)# network 10.131.162.0 mask 255.255.255.0
P6R4(config-router)# neighbor 10.131.162.2 remote-as 300
P6R4(config-router)# neighbor 10.131.162.2 descrip eBGP to P6R3
```

BCNVC v6.1—7-46



Configure EBGP Routing PE-to-CE (Cont.)

PE BGP configuration

```
P6R3(config)#router bgp 300
P6R3(config-router)# address-family ipv4 vrf RED
P6R3(config-router-af)# neighbor 10.131.162.1 remote-as 65000
P6R3(config-router-af)# neighbor 10.131.162.1 descript eBGP to P6R4
P6R3(config-router-af)# neighbor 10.131.162.1 activate
P6R3(config-router-af)# network 10.131.162.0 mask 255.255.255.0
P6R3(config-router-af)# exit-address-family
```

- Do not forget that you need to advertise a network, and you need a route to that network.

```
P6R3(config)#ip route vrf RED 10.131.162.0 255.255.255.0 10.131.162.1
```

BCMVC v6.1—7-47

Connecting CE- to PE-routers using EBGP allows the continuity of BGP policies between customer sites. BGP attributes such as AS_PATH, aggregator, and community can be propagated across the MPLS network.

In addition, no IGP-to-BGP redistribution is necessary, and based on standard BGP controls, the PE-router can limit the number of IP prefixes the BGP CE is allowed to announce (by means of the **neighbor maximum-prefix** command in conjunction with the **maximum routes** command for each VRF configuration).

The **address-family ipv4** command replaces the **match nlri** and **set nlri** commands.

Network Stability with EBGP

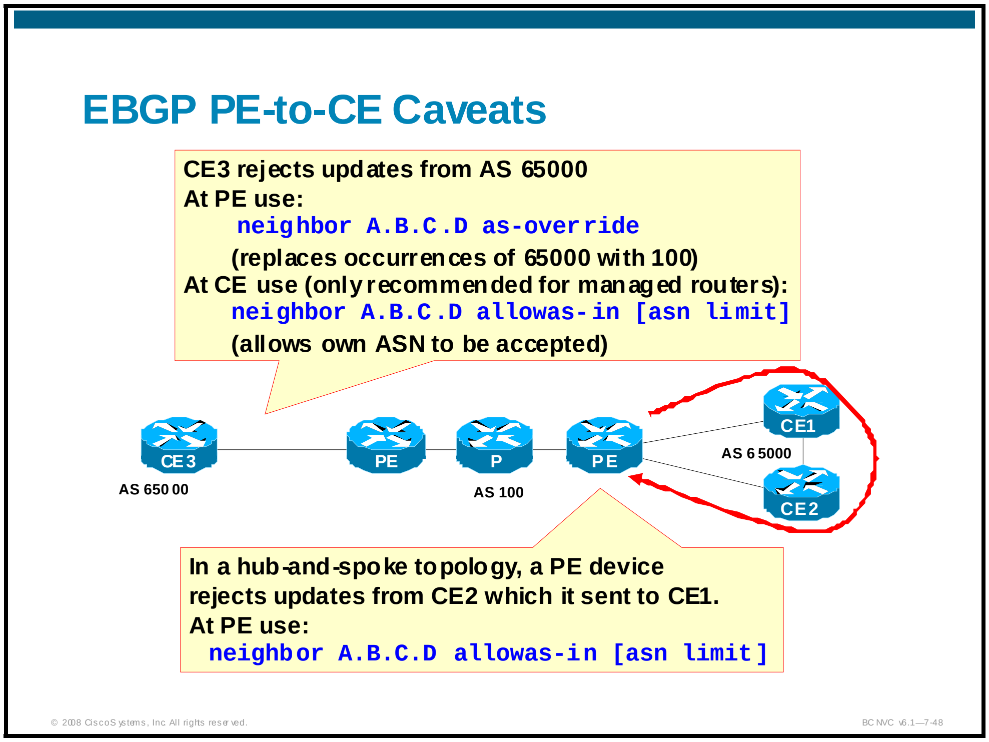
To prevent instability caused by a CE-router flooding routes into a PE-router, use the **maximum routes** command for each VRF that limits the number of routes the PE-router is allowed to receive.

```
maximum routes limit [warn threshold | warn-only]
```

```
[no] maximum routes
```

Use this command either as a warning or to stop further routes from being learned. This command is similar to the **BGP neighbor maximum-prefix** command.

EBGP PE-to-CE Caveats



EBGP PE-to-CE Caveats

CE3 rejects updates from AS 65000
At PE use:

```
neighbor A.B.C.D as-override
```

(replaces occurrences of 65000 with 100)
At CE use (only recommended for managed routers):

```
neighbor A.B.C.D allowas-in [asn limit]
```

(allows own ASN to be accepted)

In a hub-and-spoke topology, a PE device rejects updates from CE2 which it sent to CE1.
At PE use:

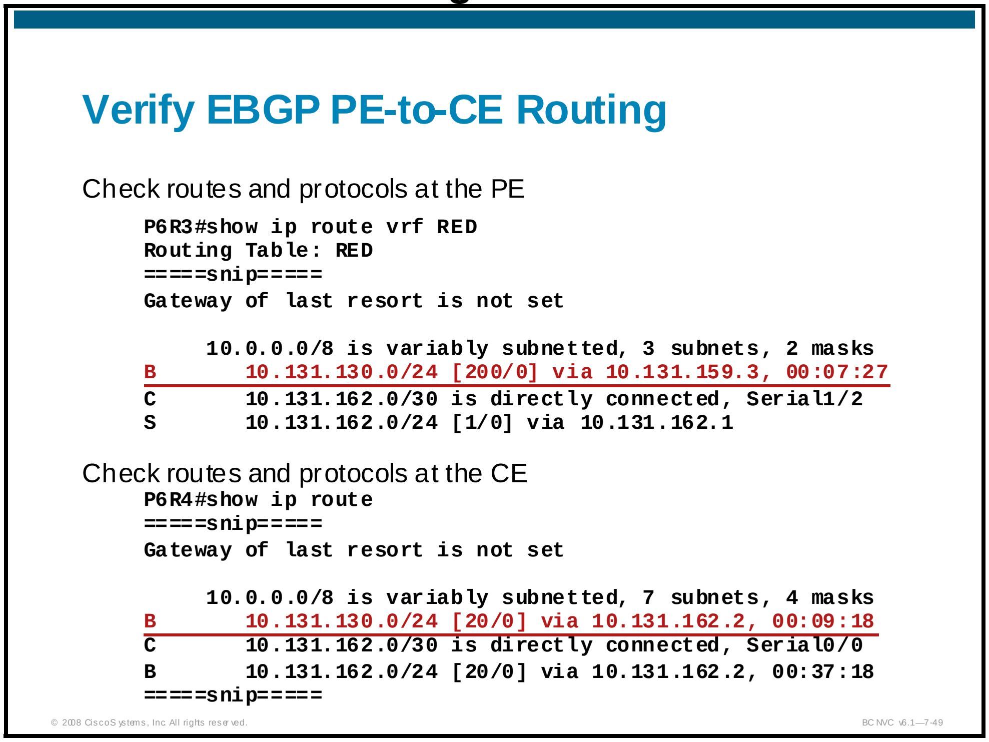
```
neighbor A.B.C.D allowas-in [asn limit]
```

BCNVC v6.1—7-48

Here are two points to remember when configuring BGP:

- Use **as-override** when multiple VPN sites use the same AS number.
- Use **allowas-in** when using a hub-and-spoke topology with BGP PE-CE on the hub.

Verify EBGP PE-to-CE Routing



Verify EBGP PE-to-CE Routing

Check routes and protocols at the PE

```
P6R3#show ip route vrf RED
Routing Table: RED
=====snip=====
Gateway of last resort is not set

     10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
B       10.131.130.0/24 [200/0] via 10.131.159.3, 00:07:27
C       10.131.162.0/30 is directly connected, Serial1/2
S       10.131.162.0/24 [1/0] via 10.131.162.1
```

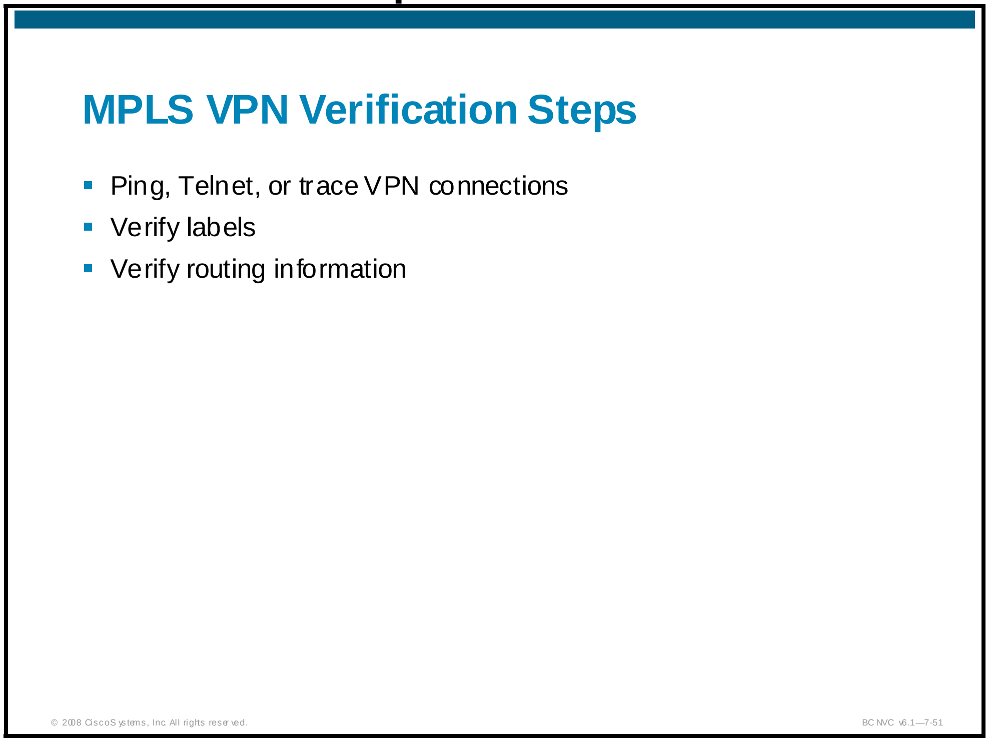
Check routes and protocols at the CE

```
P6R4#show ip route
=====snip=====
Gateway of last resort is not set

     10.0.0.0/8 is variably subnetted, 7 subnets, 4 masks
B       10.131.130.0/24 [20/0] via 10.131.162.2, 00:09:18
C       10.131.162.0/30 is directly connected, Serial0/0
B       10.131.162.0/24 [20/0] via 10.131.162.2, 00:37:18
=====snip=====
```

BCNVC v6.1—7-49

How Do I Verify MPLS VPN Functionality?

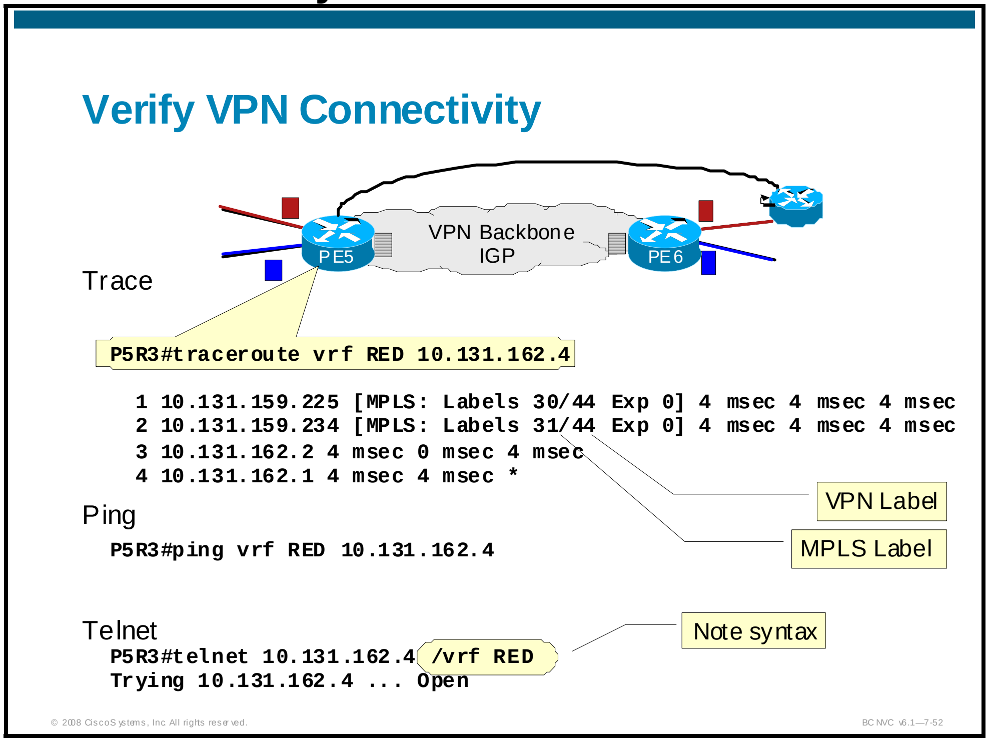
MPLS VPN Verification Steps



MPLS VPN Verification Steps

- Ping, Telnet, or trace VPN connections
- Verify labels
- Verify routing information

Verify VPN Connectivity



Verify VPN Connectivity

Trace

```
P5R3#traceroute vrf RED 10.131.162.4

  1 10.131.159.225 [MPLS: Labels 30/44 Exp 0] 4 msec 4 msec 4 msec
  2 10.131.159.234 [MPLS: Labels 31/44 Exp 0] 4 msec 4 msec 4 msec
  3 10.131.162.2 4 msec 0 msec 4 msec
  4 10.131.162.1 4 msec 4 msec *
```

VPN Label

MPLS Label

Ping

```
P5R3#ping vrf RED 10.131.162.4
```

Telnet

```
P5R3#telnet 10.131.162.4 /vrf RED
Trying 10.131.162.4 ... Open
```

Note syntax

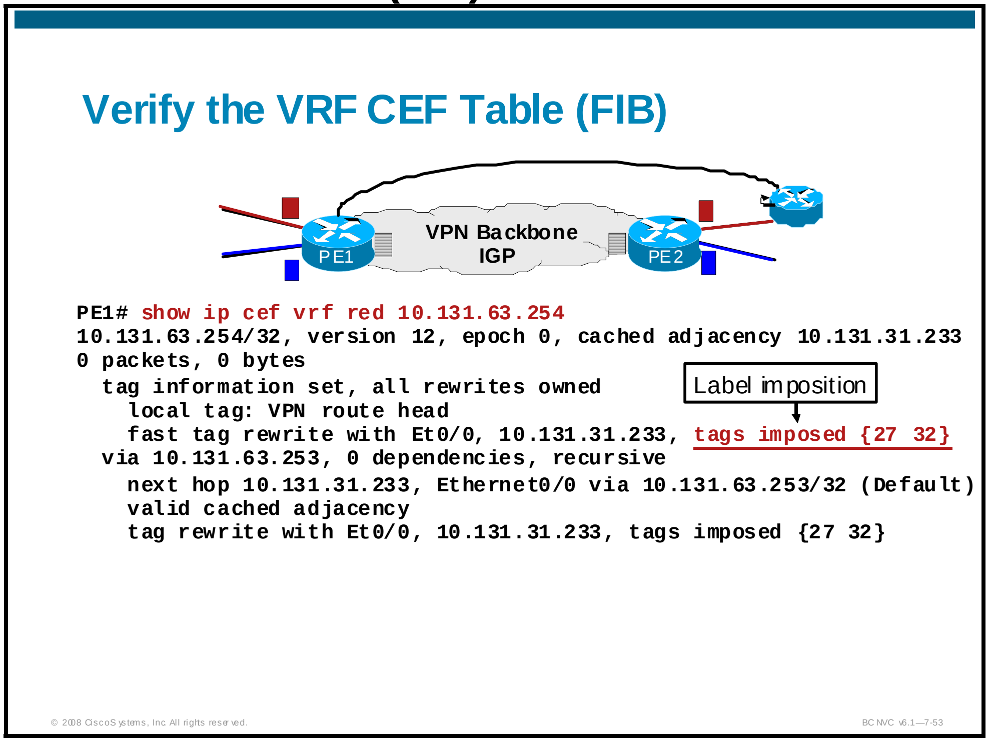
© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—7-52

Note Before establishing an MPLS VPN, you must be able to ping PE to PE.

Follow these steps to verify the neighbor MPLS interface connections:

1. Enter the **ping** command to verify that the connection is up between the PE- and the CE-routers
2. Enter the MPLS-aware **traceroute vrf** command to verify the MPLS labels are set.
3. Verify that the interfaces that appear in the **traceroute** command output displays the correct cross-connect addresses.

Verify the VRF CEF Table (FIB)



Verify the VRF CEF Table (FIB)

```
PE1# show ip cef vrf red 10.131.63.254
10.131.63.254/32, version 12, epoch 0, cached adjacency 10.131.31.233
0 packets, 0 bytes
  tag information set, all rewrites owned
    local tag: VPN route head
    fast tag rewrite with Et0/0, 10.131.31.233, tags imposed {27 32}
  via 10.131.63.253, 0 dependencies, recursive
    next hop 10.131.31.233, Ethernet0/0 via 10.131.63.253/32 (Default)
    valid cached adjacency
    tag rewrite with Et0/0, 10.131.31.233, tags imposed {27 32}
```

Label imposition

© 2008 Cisco Systems, Inc. All rights reserved. BCNVC v6.1—7-53

MPLS VPN uses a two-level label stack. One of the labels is used to identify the VRF, and is setup between the two PE ATM switch routers. The other label (on the top of the stack) is the “backbone” label, and is setup by the standard MPLS network.

Follow these steps to verify the labels on MPLS VPN interface connections:

- Enter the **traceroute VRF [vrf-name] ip-address** command to verify the transport addresses.

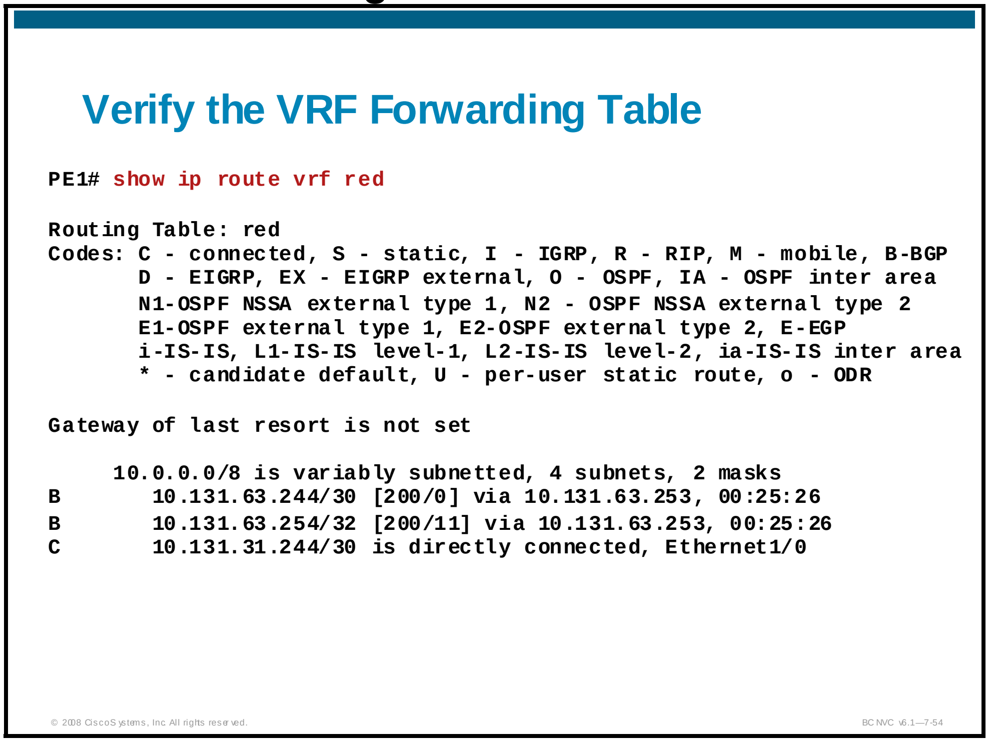
Note This command only works with an MPLS-aware traceroute, and only if the backbone routers are configured to propagate and generate IP TTL information.

- Enter the **show ip cef vrf** command with the VRF name and summary keyword to display a summary of the CEF table associated with a VRF.

The **show ip cef vrf** command with the VRF name and detail keyword displays greater detail of the CEF table associated with a VRF.

- Verify:
  - — That the tag information set with the local tag field confirms that the labels are used effectively.
  - — That the fast tag rewrite field displays a stack of (at least) two labels that are used for VPN destinations.

Verify the VRF Forwarding Table



Verify the VRF Forwarding Table

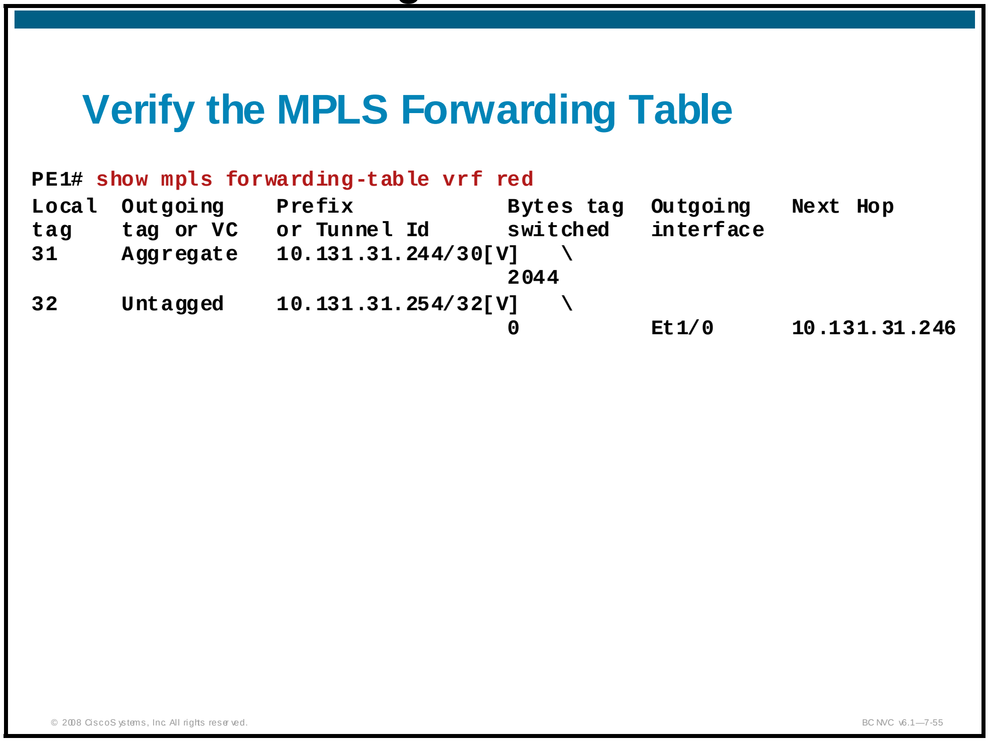
```
PE1# show ip route vrf red

Routing Table: red
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B-BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1-OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1-OSPF external type 1, E2-OSPF external type 2, E-EGP
       i-IS-IS, L1-IS-IS level-1, L2-IS-IS level-2, ia-IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR

Gateway of last resort is not set

     10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
B       10.131.63.244/30 [200/0] via 10.131.63.253, 00:25:26
B       10.131.63.254/32 [200/11] via 10.131.63.253, 00:25:26
C       10.131.31.244/30 is directly connected, Ethernet1/0
```

BCNVC v6.1—7-54

Follow these steps to verify the routing tables for MPLS VPN interface connections:

- To check routing tables or routing protocol databases, use the same commands you would use to check the global routing table. For example, enter `show ip route vrf` with the VRF-name to display only the MPLS VPN connections.
- Check the destination for a particular address by using `show ip route vrf` with the VRF-name and IP-address variables.

Verify the MPLS Forwarding Table



Verify the MPLS Forwarding Table

```
PE1# show mpls forwarding-table vrf red
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
31     Aggregate   10.131.31.244/30[V]    \
                                     2044
32     Untagged    10.131.31.254/32[V]    \
                                     0          Et1/0      10.131.31.246
```

© 2008 Cisco Systems, Inc. All rights reserved.

BCNVC v6.1—7-55

Check the MPLS LFIB for a specific VPN with `show mpls forwarding-table vrf <name>`

Label Types

Untagged Label

An untagged label strips out any label present in the packet to make it a pure IP packet and ships it to the corresponding outgoing interface as indicated in the tag forwarding information base (TFIB) (there is no IP table lookup).

[V] refers to the fact that this is a VPN prefix.

If “untagged” appears on any provider (P) router when there are VPNs defined in the network, it clearly indicates the problem as a P-router removes all the labels prematurely, resulting in the loss of VPN information. The provider edge (PE) device then drops the packet, since the PE does not have the routing information in the global IP routing table for this packet.

Pop Label

A pop label is an implicit-null; Only the top label is removed. All other labels in the label stack are preserved. The outgoing packet is still an MPLS packet.

Aggregate Label

The aggregate label is:

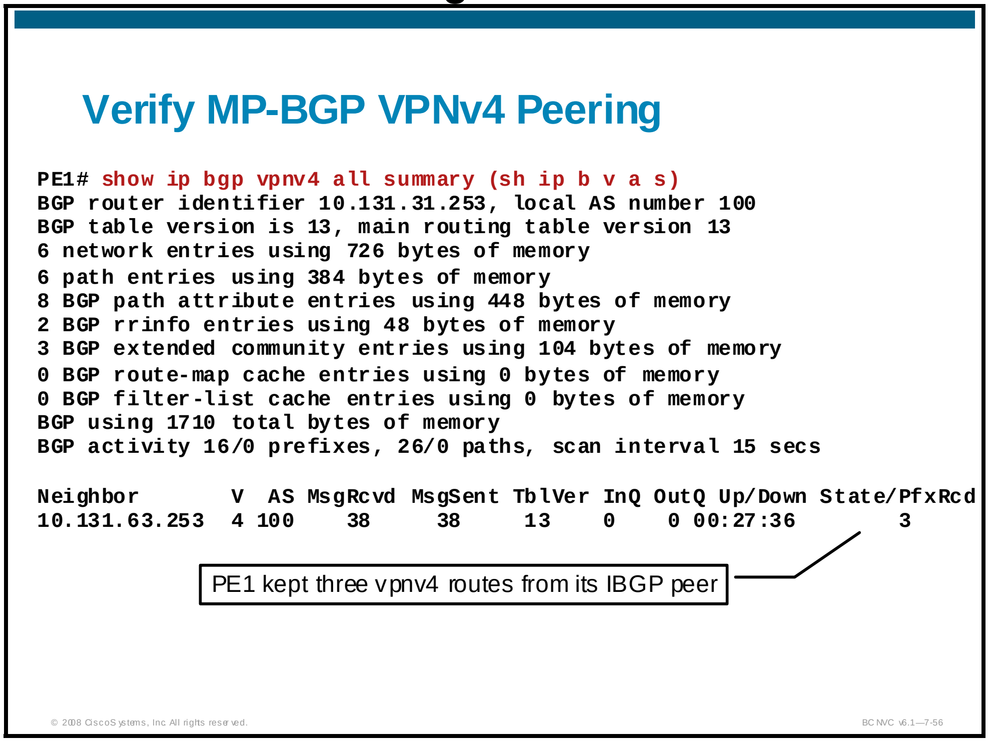
- Used in VPNs for directly connected subnets and aggregate routes.
- A place holder to indicate that the arriving MPLS packet should be switched through an IP lookup

Upon receiving a packet that has an aggregate outgoing label, the PE-router pops the label from the packet and forwards the packet up to the IP layer for further processing, because such a packet cannot be forwarded directly from MPLS.

PE-to-CE links always have aggregate labels, except Point-to-Point Protocol (PPP) on PE-to-CE links that have aggregate labels for /30 (for the PE-to-CE interface), and a untagged labels for /32 (for the host route).

Why is further processing required by the IP layer (or, how are aggregate labels so different)? In the output of the label forwarding information base (LFIB) there is no outgoing interface entry. If the packet is destined to .246 (CE side), then the PE should forward the packet to the CE. But if the packet is destined to .245 (PE side), then the PE should respond to the packet. Whether the packet is destined to the PE or the CE is determined by IP.

Verify MP-BGP VPNv4 Peering



Verify MP-BGP VPNv4 Peering

```
PE1# show ip bgp vpnv4 all summary (sh ip b v a s)
BGP router identifier 10.131.31.253, local AS number 100
BGP table version is 13, main routing table version 13
6 network entries using 726 bytes of memory
6 path entries using 384 bytes of memory
8 BGP path attribute entries using 448 bytes of memory
2 BGP rrinfo entries using 48 bytes of memory
3 BGP extended community entries using 104 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 1710 total bytes of memory
BGP activity 16/0 prefixes, 26/0 paths, scan interval 15 secs

Neighbor        V  AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down State/PfxRcd
10.131.63.253   4 100    38      38     13    0    0 00:27:36        3
```

PE1 kept three vpnv4 routes from its IBGP peer

BCNVC v6.1—7-56

- Check the MP-BGP routing information exchanged with MP-BGP peers with the command **show ip bgp vpnv4 all summary.**
- Verify PE-to-CE routing.

Check the routing protocol used on the CE using show commands, and apply them to the correct VRF.

For BGP, enter **show ip bgp vpnv4 vrf <name>**.

For RIP, enter **show ip rip database vrf <name> .**

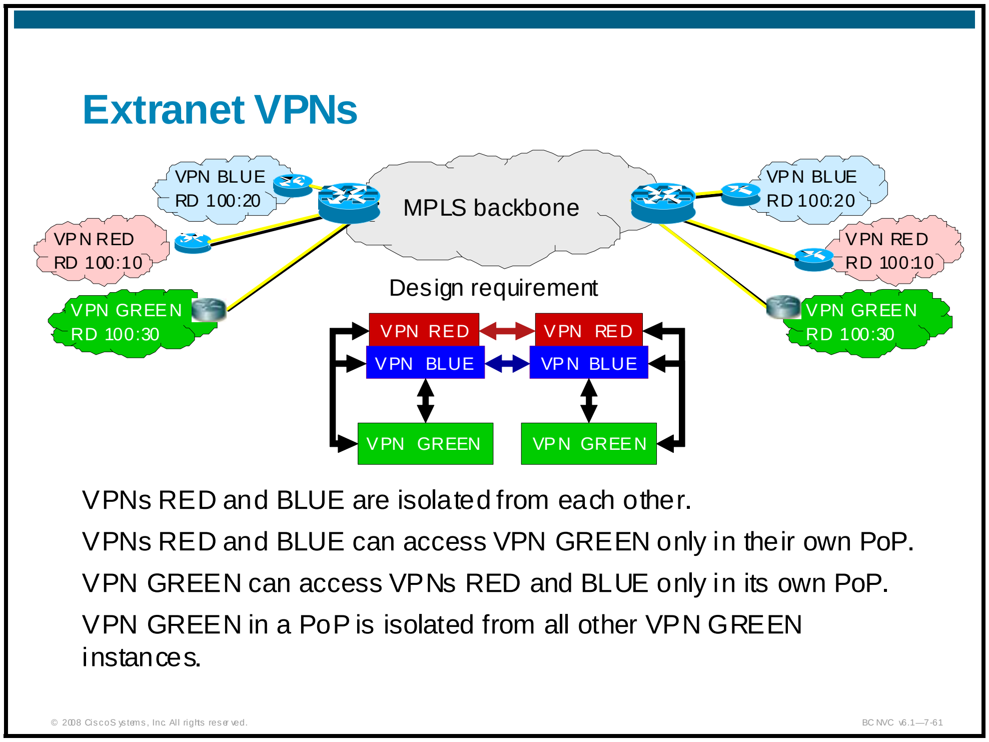
For OSPF, enter **show ip ospf [process-id area-id] database <process number>**.

For EIGRP, enter **show ip eigrp vrf <name> topology**.

- Verify:
  - — That the routing table is correct (from a customer point of view), or determine what is missing from the routing table.
  - — That BGP is up and working, or determine which neighbor is missing.

How Do I Configure and Verify Extranet VPNs?

Extranet VPNs



Extranet VPNs

VPNs RED and BLUE are isolated from each other.

VPNs RED and BLUE can access VPN GREEN only in their own PoP.

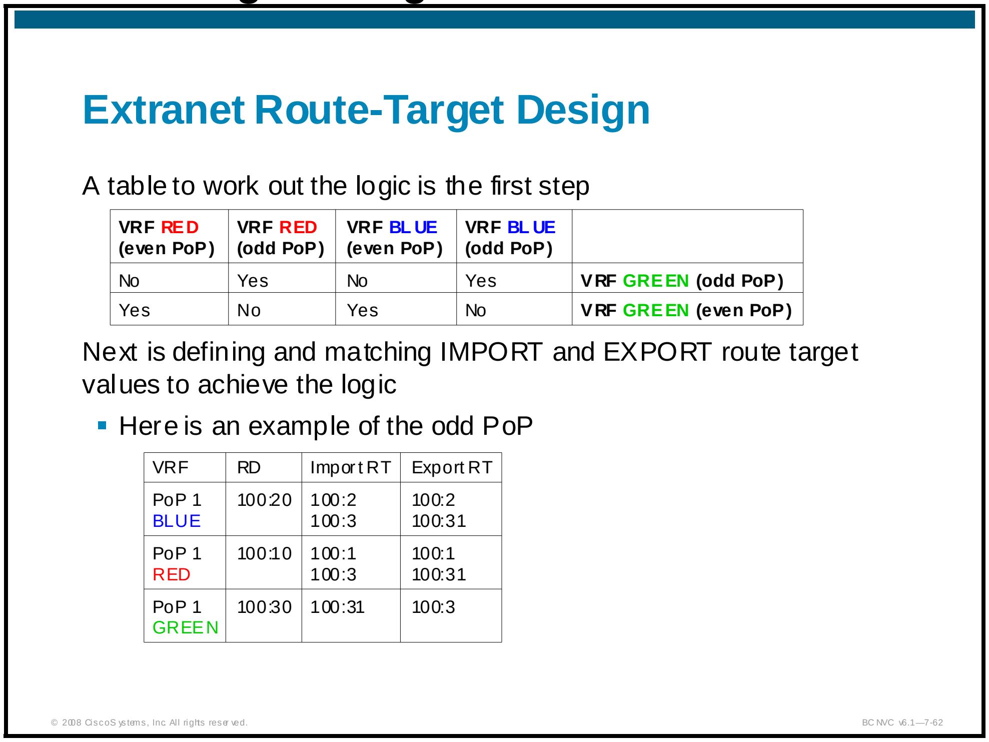
VPN GREEN can access VPNs RED and BLUE only in its own PoP.

VPN GREEN in a PoP is isolated from all other VPN GREEN instances.

BCNVC v6.1—7-61

The following scenario depicts a hypothetical situation where there is a management entity; assume a server of some type, at each local PoP. The network operator desires that local clients be able to access this server but not remote clients. This requirement is pictured below in the design requirements. The management VPN GREEN is reachable by VPNs RED and BLUE in the local PoP but remote clients in VPNs RED and BLUE do not have connectivity.

Extranet Route-Target Design



Extranet Route-Target Design

A table to work out the logic is the first step

VRF RED (even PoP)	VRF RED (odd PoP)	VRF BLUE (even PoP)	VRF BLUE (odd PoP)	
No	Yes	No	Yes	**VRF GREEN (odd PoP)**
Yes	No	Yes	No	**VRF GREEN (even PoP)**

Next is defining and matching IMPORT and EXPORT route target values to achieve the logic

- Here is an example of the odd PoP

VRF	RD	Import RT	Export RT
PoP 1 BLUE	100:20	100:2 100:3	100:2 100:31
PoP 1 RED	100:10	100:1 100:3	100:1 100:31
PoP 1 GREEN	100:30	100:31	100:3

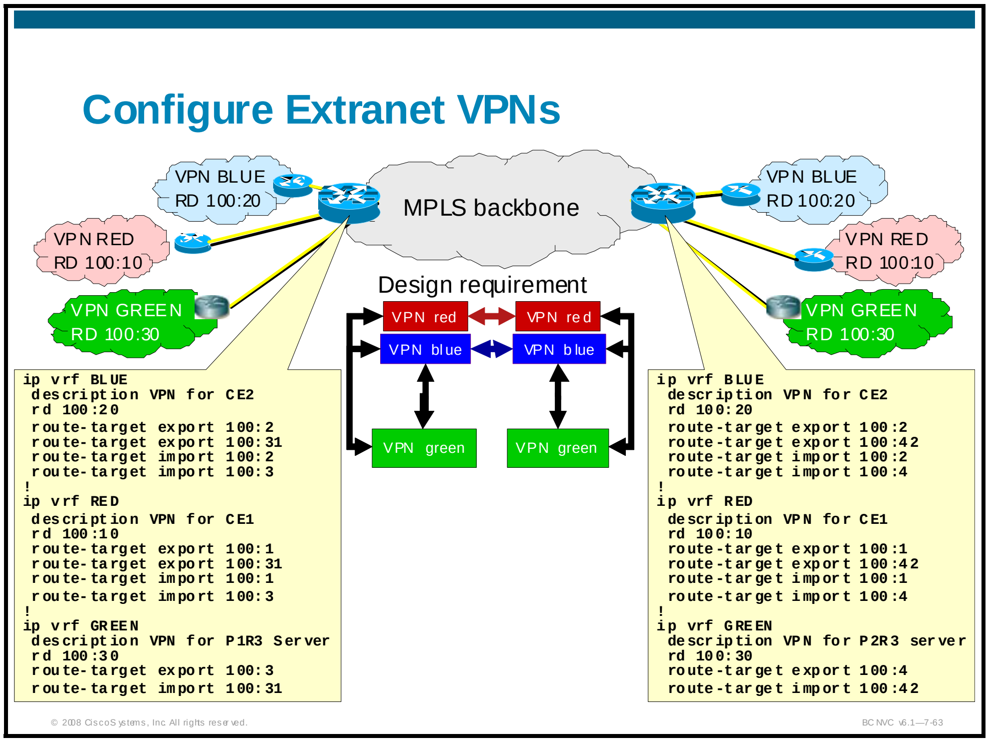
BCNVC v6.1—7-62

Determine the design requirements of the MPLS VPN interconnectivity. A chart such as the following is recommended. The following shows the definition for PoP 3 (left side of graphic). PoP 4 is incomplete and may be used as an exercise.

VRF	Route Distinguisher	Import Route Target	Export Route Target
PoP 3 BLUE	100:20	100:2 100:3	100:2 100:31
PoP 3 RED	100:10	100:1 100:3	100:1 100:31
PoP 3 GREEN	100:30	100:31	100:3
PoP 4 BLUE	100:20		
PoP 4 RED	100:10		
PoP 4 GREEN	100:30		

Using this data assign the import and export RTs to the appropriate VRFs.

Configure Extranet VPNs

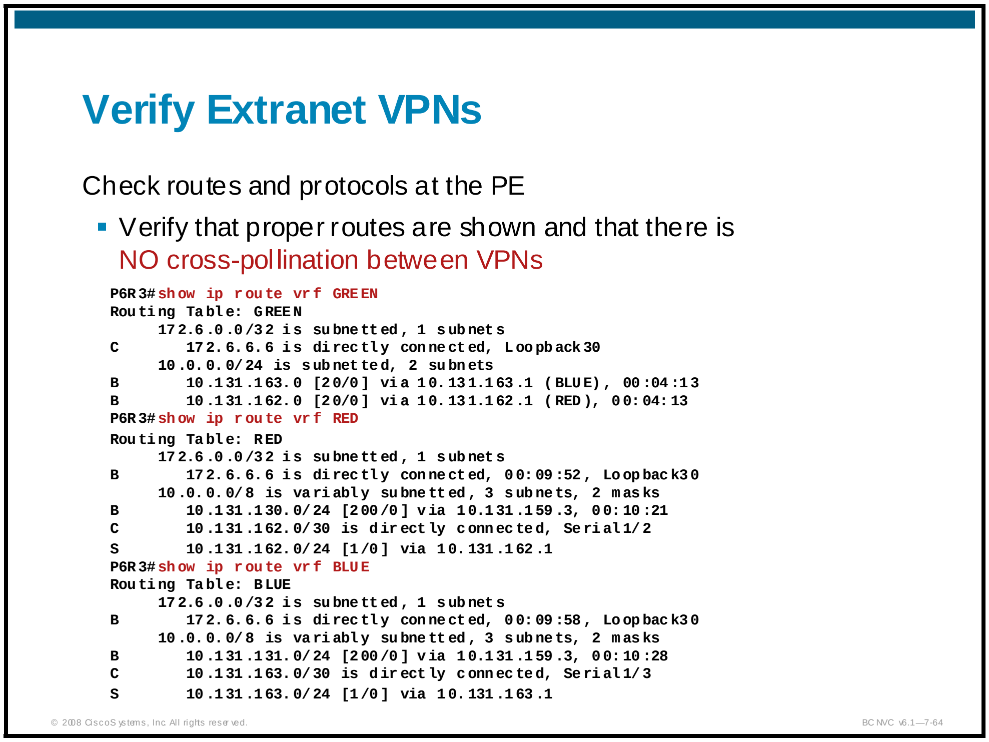


Configure Extranet VPNs

```
ip vrf BLUE
 description VPN for CE2
 rd 100:20
 route-target export 100:2
 route-target export 100:31
 route-target import 100:2
 route-target import 100:3
!
ip vrf RED
 description VPN for CE1
 rd 100:10
 route-target export 100:1
 route-target export 100:31
 route-target import 100:1
 route-target import 100:3
!
ip vrf GREEN
 description VPN for P1R3 Server
 rd 100:30
 route-target export 100:3
 route-target import 100:31
```

```
ip vrf BLUE
 description VPN for CE2
 rd 100:20
 route-target export 100:2
 route-target export 100:42
 route-target import 100:2
 route-target import 100:4
!
ip vrf RED
 description VPN for CE1
 rd 100:10
 route-target export 100:1
 route-target export 100:42
 route-target import 100:1
 route-target import 100:4
!
ip vrf GREEN
 description VPN for P2R3 server
 rd 100:30
 route-target export 100:4
 route-target import 100:42
```

Verify Extranet VPNs



Verify Extranet VPNs

Check routes and protocols at the PE

- Verify that proper routes are shown and that there is NO cross-pollination between VPNs

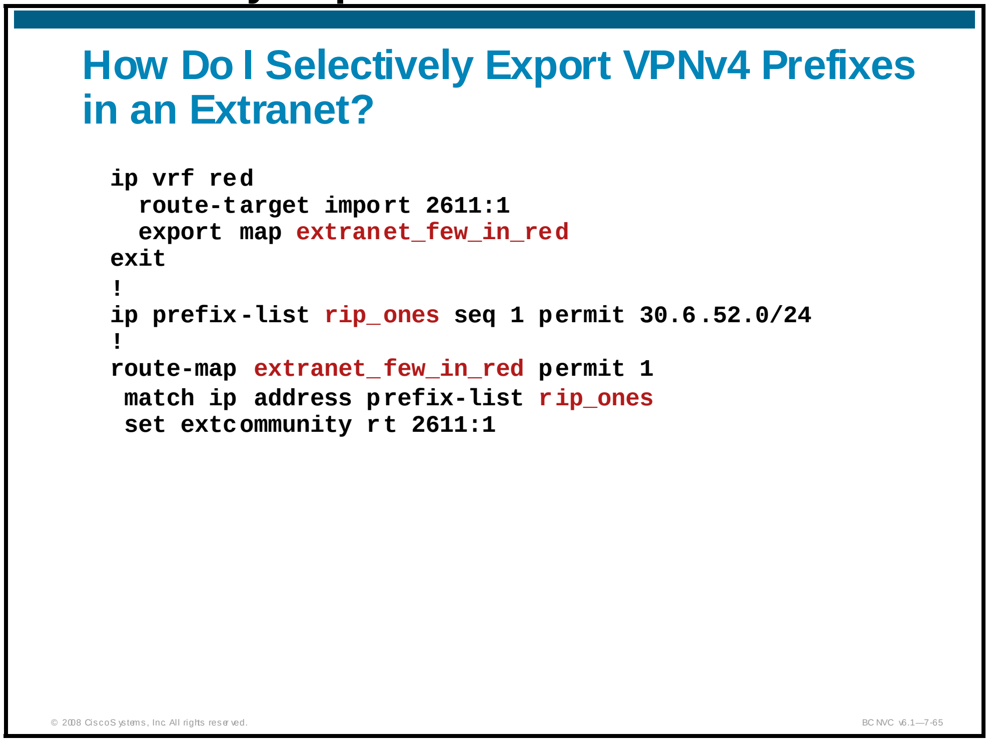
```
P6R3#show ip route vrf GREEN
Routing Table: GREEN
     172.6.0.0/32 is subnetted, 1 subnets
C       172.6.6.6 is directly connected, Loopback30
     10.0.0.0/24 is subnetted, 2 subnets
B       10.131.163.0 [20/0] via 10.131.163.1 (BLUE), 00:04:13
B       10.131.162.0 [20/0] via 10.131.162.1 (RED), 00:04:13
P6R3#show ip route vrf RED
Routing Table: RED
     172.6.0.0/32 is subnetted, 1 subnets
B       172.6.6.6 is directly connected, 00:09:52, Loopback30
     10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
B       10.131.130.0/24 [200/0] via 10.131.159.3, 00:10:21
C       10.131.162.0/30 is directly connected, Serial1/2
S       10.131.162.0/24 [1/0] via 10.131.162.1
P6R3#show ip route vrf BLUE
Routing Table: BLUE
     172.6.0.0/32 is subnetted, 1 subnets
B       172.6.6.6 is directly connected, 00:09:58, Loopback30
     10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
B       10.131.131.0/24 [200/0] via 10.131.159.3, 00:10:28
C       10.131.163.0/30 is directly connected, Serial1/3
S       10.131.163.0/24 [1/0] via 10.131.163.1
```

BCNVC v6.1—7-64

Verify the configuration using the same techniques as with an intranet VPN. In addition to confirming that the routing tables are correctly populated, use ping, trace, and Telnet commands to confirm IP reachability to the prefixes you should reach, and also to those that you should not be able to reach.

Caution	Misconfiguration of route-target values is the weak link in MPLS VPN security. Verify that no cross-pollination has occurred between VPNs.

How Do I Selectively Export VPNv4 Prefixes in an Extranet?



How Do I Selectively Export VPNv4 Prefixes in an Extranet?

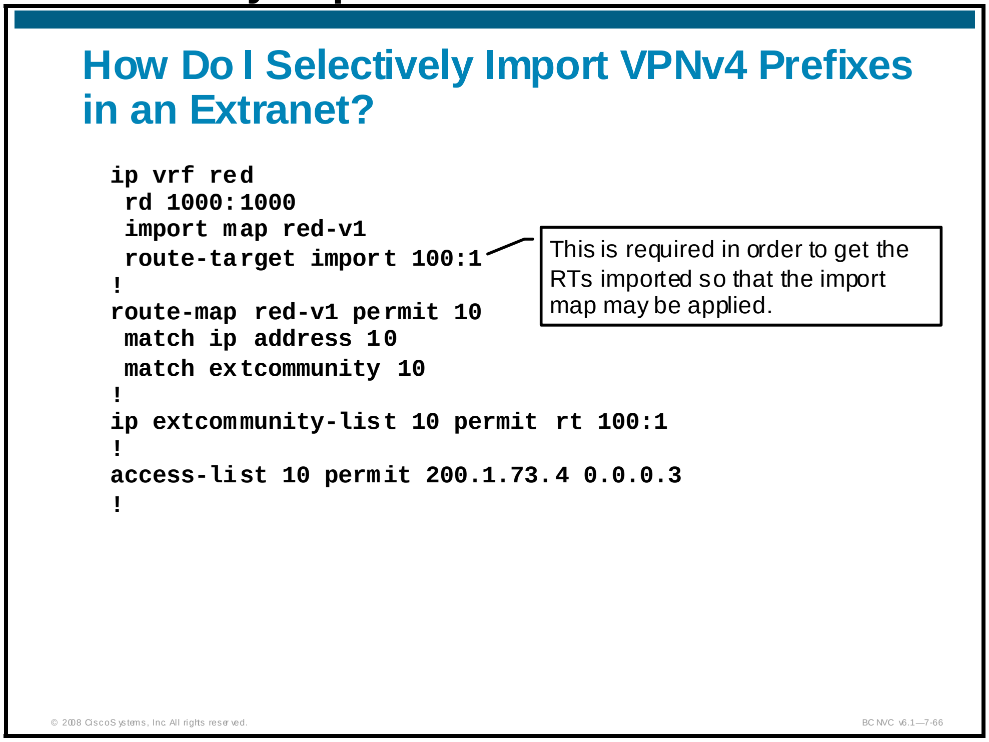
```
ip vrf red
  route-target import 2611:1
  export map extranet_few_in_red
exit
!
ip prefix-list rip_ones seq 1 permit 30.6.52.0/24
!
route-map extranet_few_in_red permit 1
 match ip address prefix-list rip_ones
 set extcommunity rt 2611:1
```

BCNVC v6.1—7-65

Remember that the presence of route-target export <rt> is not required. Another way to add export RTs is to use an export map. This export map can either append extra export route-targets to the existing RTs (with the additive keyword) or it can change export route-targets as is shown in the example (no additive keyword).

Omitting the `additive` keyword allows you to override the configured set of route-targets associated with a specific route.

How Do I Selectively Import VPNv4 Prefixes in an Extranet?



How Do I Selectively Import VPNv4 Prefixes in an Extranet?

```
ip vrf red
 rd 1000:1000
 import map red-v1
 route-target import 100:1
!
route-map red-v1 permit 10
 match ip address 10
 match extcommunity 10
!
ip extcommunity-list 10 permit rt 100:1
!
access-list 10 permit 200.1.73.4 0.0.0.3
!
```

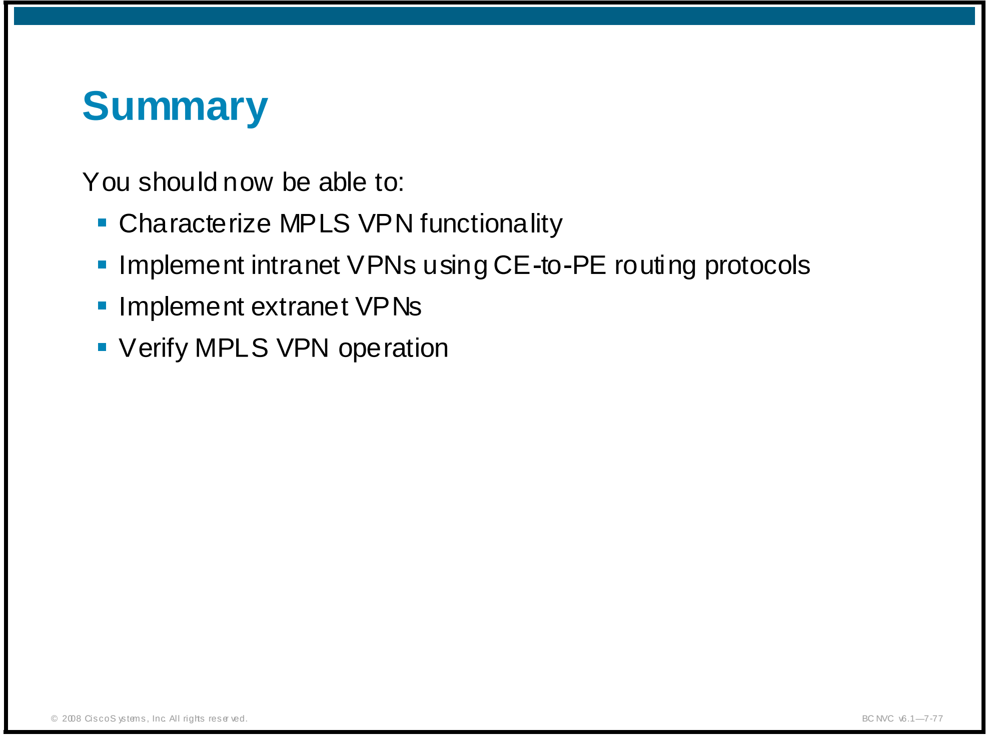
This is required in order to get the RTs imported so that the import map may be applied.

BCNVC v6.1—7-66

Remember to configure the route-target import <value> along with the import map. Its presence is mandatory in order to make the import map work. This is because routes, with that route-target, are required to be accepted before the import-map can be applied.

When a PE receives all the VPNv4 routes from the remote MP-IBGP peer, the BGP process filters prefixes based on the import route-target configured in all VRFs. In the absence of the import statement, all relevant routes are denied, and import-map does not have anything to work with.

Summary



Summary

You should now be able to:

- Characterize MPLS VPN functionality
- Implement intranet VPNs using CE-to-PE routing protocols
- Implement extranet VPNs
- Verify MPLS VPN operation

BCNVC v6.1—7-77