Mohammed M. Alani

Guide to Cisco Routers Configuration

Becoming a Router Geek

Second Edition

Springer

Guide to Cisco Routers Configuration

Mohammed M. Alani

Guide to Cisco Routers Configuration

Becoming a Router Geek

Second Edition

Springer

Mohammed M. Alani
Al Khawarizmi International College
Abu Dhabi
United Arab Emirates

ISBN 978-3-319-54629-2 ISBN 978-3-319-54630-8 (eBook)
DOI 10.1007/978-3-319-54630-8

Library of Congress Control Number: 2017934622

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Since the first edition of this book, I have received many requests to do a second version that is extended and more thorough. With the research activities I got involved in, I could not do that very early. However, at one point, I needed to do some router configuration as a part of one of the courses I was teaching. I looked through the first edition of the book, and I could not find what I was looking for. This came as a wake-up call to me that a second edition is a must.

Cisco networking equipment still takes a major role in our lives. Many service providers, corporate networks, government networks, and others rely on Cisco devices, which basically means that we are relying on these devices on daily basis. All network administrators know that relying on Cisco devices gives them some relief in terms of reliability.

The uniqueness of this book lies in its cookbook-like way of writing that does not dive deep into theory and provides a ready-to-use reference for everyday configuration needs. My biggest concern when writing this book was to avoid complications and get directly to the simplified configuration steps.

The second edition of this book included many important topics that were missing from the first edition, such as MPLS, multicasting, GRE, HSRP, with many others. Some older topics like access-control lists were expanded to include more details like reflexive and timed access lists. The second edition included configuration steps for IPv6 versions for most of the protocols covered by the book. With the rapid increase of the use of IPv6, its configuration became an important skill to have in your arsenal.

An important addition to this edition of the book is the inclusion of training scenarios. In total, the second edition includes 61 training scenarios. These training scenarios were aimed to be a way of gaining skills by doing the tasks instead of reading about them only. Most of these tasks can be implemented using the currently available commercial network simulator.

This edition is divided into nine chapters: basic configuration, domestic duties, routing, advanced routing, WAN technologies, security, router management, remote connectivity, and tips. Each chapter explains in detail the steps required to configure different protocols on Cisco router and explains when would you need to invoke this procedure.

Intended Audience of the Guide:

- field network engineers engaging Cisco routers
- students working on Cisco routers in their laboratories
- laboratory instructors
- Cisco certification seekers
- Cisco networking academy students
- everyone working with or wanting to learn about Cisco router configuration

How to Use this Guide:

To make guide easier to use, different parts of the text were formatted differently. The meanings of these formats:

- `courier new` font is used for output of the router
- **`courier new bold`** font is used for commands input to the router
- *`courier new italic`* font is used for commands parameters that the reader have to choose.
- [ ] square brackets are used for optional commands.

Finally, I would like to express my sincere gratitude to my editors Wayne Wheeler and Simon Rees for making this project possible again. I would also like to extend my thanks to the readers of the first edition for their feedback that encouraged me to write the second expanded edition.

Abu Dhabi, United Arab Emirates Mohammed M. Alani
December 2016

Contents

Chapter 1
Starting Up a Cisco Router

Keywords Cisco · Router · Console · Basic configuration · IPv4 · IPv6

1.1 Connecting the Router

In order to configure the Cisco router to perform the network operation desired, the first thing to do is to connect the router through console connection. This connection is used to configure the router and it does not carry user data.

Most routers come with console cable that has one DB-9 end and the other end is RJ-45 connecter. These connectors can be seen in Fig. 1.1. The cable is usually a slim ribbon that looks different from the regular LAN twisted pair cables.

Leave the router off and connect the RJ-45 end of the cable to the port of the router labelled console. The other end, the DB-9 one, should be connected to the computer serial port. Most of the modern laptops do not have the legacy DB-9 serial port. Instead of the DB-9 serial port, a USB port can be used. This would require the use of a USB-to-DB9 adapter.

Some older routers come with console cables that have RJ-45 connectors in both of their ends. These cables come with RJ-45-to-DB9 adapters.

After connecting the console cable to the computer, using the DB-9 connecter or the USB, software for terminal emulation is required.

The first choice is HyperTerminal if you are using Windows XP. If you are using Windows Vista or Windows 7, there are several free alternatives such as PuTTY or Tera Term. If you are using Linux, you can use MiniCom or CuteCom.

For Mac, you can use MiniCom and ZTerm.

The basic settings that need to be done in the terminal emulation software are the following:

1. Bits per sec: 9600
2. Data bits: 8
3. Parity: none
4. Stop bits: 1
5. Flow control: none

M.M. Alani, *Guide to Cisco Routers Configuration*,
DOI 10.1007/978-3-319-54630-8_1

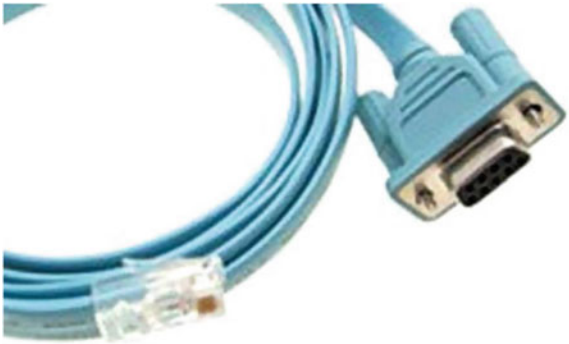


Fig. 1.1 Console cable

After connecting the cable and configuring the settings on the terminal emulation software, turn the router power on.

The screen should show the router boot-up process and information such as the IOS version, amount of memory available, and types of interfaces.

Eventually, you will end up at the user EXEC mode with the prompt ‘`Router>`’.

If the router was configured before and it has console password configured in it, you will be asked to input the password before getting to the user EXEC mode.

1.2 Basic Configuration

Basic configuration is simply what you need to configure on a router coming out of the box or surviving a configuration erase. Think of it as saying ‘Hi’ to the router.

The following steps are what we refer to as the basic router configuration:

1. Go to the global configuration mode and give the router a hostname:

   ```
   Router>enable
   Router#configure terminal
   Router(config)#hostname new-hostname
   ```

   This would change the hostname of the router from ‘`Router`’ to *new-hostname*. Keep in mind that this name follows the old file name rules (it should start with a letter, and should not contain spaces or symbols).
2. Set up enable/secret password:

   ```
   Router(config)#enable password your-password
   ```

   Or

   ```
   Router(config)#enable secret your-password
   ```

   This password will be required when you type ‘enable’ to go from user EXEC mode to privilege mode.
   The first one saves the password in plain text, while the second one saves the password in encrypted format.
   The first one is almost obsolete. It is more secure to use the second one.
   Remember that only one of them is required. If you set them both, the secret password would prevail.

3. Set up console password:
   ```
   Router(config)#line console 0
   Router(config-line)#password console-password
   Router(config-line)#login
   ```
   This password will be required when a console connection is made. It is the first password that an administrator will be asked for before entering any mode.
4. To prevent the router status messages from interrupting your writing, use the following command in the console line configuration mode:
   ```
   Router(config-line)#logging synchronous
   ```
5. If you plan to use Telnet, set up the Telnet password. If you do not intend to use Telnet in the near future, do not set it up.
   ```
   Router(config)#line vty 0 4
   Router(config-line)#password telnet-password
   Router(config-line)#login
   ```
   In some routers, vty 0 15 is used instead of vty 0 4, depending on the number of simultaneous Telnet sessions you want to allow. If you need only one, just write ‘`line vty 0`’.
6. Save the configuration from the RAM to the NVRAM.
   ```
   Router#copy running-configuration startup-configuration
   ```
   Or
   ```
   Router#copy run start
   ```

1.3 Interfaces Configuration

Before we dive into the interfaces settings, we need to understand the difference between an Interface and a Line. In the simplest distinction, interfaces are designed to carry user and routing data while lines are used for configuration purposes.

1.3.1 Configuring IPv4 Addresses

1. Assign IPv4 addresses to interfaces you plan to use:
   ```
   Router(config)#interface interface-type interface-number
   ```

```
Router(config-if)#ip address interface-ip-address subnetmask
```

where,

interface-type is the interface type such as ethernet, FastEthernet, or serial.

interface-number is the interface number like 0, 0/0, or 0/1/0.

interface-ip-address is the IPv4 address you want to assign to this interface.

subnetmask is the subnet mask of the network this interface is connected to.

2. By default, all router interfaces are shutdown. To turn on an interface, use the following command in the interface configuration mode:

   ```
   Router(config-if)#no shutdown
   ```

3. Repeat steps 1 and 2 for each interface you need.

1.3.2 Configuring IPv6 Addresses

1. Assign IPv4 addresses to interfaces you plan to use:

   ```
   Router(config)#interface interface-type interface-number
   Router(config-if)#ipv6 address ipv6-address/prefix-length
   ```

   where,

   interface-type is the interface type such as ethernet, FastEthernet, or serial.

   interface-number is the interface number like 0, 0/0, or 0/1/0.

   Ipv6-address is the IPv6 address you want to assign to this interface.

   Prefix-length is the IPv6 prefix length of the network this interface is connected to.

   You can use the following parameters for special settings:

   a.
      ```
      Router(config-if)#ipv6 address ipv6-address/prefix-length eui-64
      ```

      The EUI-64 parameter is used to tell the router to complete the rest of the IPv6 address using the EUI-64 rules. In this case, you need to give only the prefix instead of the *ipv6-address*.

   b.
      ```
      Router(config-if)#ipv6 address ipv6-address/prefix-length link-local
      ```

If you use the link-local parameter, the router will take the IPv6 address given in the command instead of the automatically generated link-local IPv6 address.

c.
   ```
   Router(config-if)#ipv6 address ipv6-address/prefix-length anycast
   ```
   Anycast parameter can be used to configure IPv6 anycast addresses.

2. Enable IPv6 on the interface:
   ```
   Router(config-if)#ipv6 enable
   ```
3. By default, all router interfaces are shutdown. To turn on an interface, use the following command in the interface configuration mode:
   ```
   Router(config-if)#no shutdown
   ```
4. Repeat steps 1, 2, and 3 for each interface you need.

1.3.3 Other Interface Settings

1. It is a very good practice to add interface descriptions. These descriptions are similar to remarks put into code of a program. It does not affect the operation of the interface in any way, but it gives information to the administrator viewing the configuration. This command should be written inside the interface configuration mode.
   ```
   Router(config-if)#description Write Your Own Description
   ```
   This description can be used in many useful ways such as writing the network name to which this interface is connected to, or writing the name of the other end of this link.
2. Configure bandwidth value on interfaces connected to other networks. The bandwidth value set in the following command does not affect the actual bandwidth of the link. It only changes the value of bandwidth used in best route calculation in dynamic routing protocols.
   ```
   Router(config-if)#bandwidth interface-bandwidth
   ```
   where *interface-bandwidth* is a number representing the bandwidth of the link in kilobits per second.

   It is highly advised that you set up the bandwidth of serial interfaces and even Ethernet interfaces when they are connected to other networks. If the interface is connected to a host or a group of hosts, this configuration is not necessary.

3. Setting the connection speed and duplex type in Ethernet, Fast Ethernet, and Gigabit Ethernet links is useful in some cases. By default, all interfaces auto-negotiate the speed and duplex settings. Sometimes, this negotiation does not accomplish the desired settings.

   ```
   Router(config-if)#duplex duplex-mode
   Router(config-if)#speed port-speed
   ```

   where,

   duplex-mode is the duplex mode of operation which can be `auto`, `half`, or `full`.

   port-speed is the speed of data transmission on port in megabits per second which can be `10`, `100`, `1000`, or `auto`.

1.4 Additional Basic Configuration

There are few other configurations that are useful but not necessary for the network to operate:

1. Setting a banner to be shown whenever someone tries to logon to the router configuration:

   ```
   Router(config)#banner motd #Your Message Here#
   ```

2. Encrypt the passwords such that they become non-comprehendible to anyone viewing them in the running-configuration.

   Secret password is already encrypted. All other passwords (vty, console, and auxiliary) are not. The command to encrypt them is as follows:

   ```
   Router(config)#service password-encryption
   ```

   There are two recommended methods to use this command. Because it is a service, it is not advised to keep it running all the time because it would consume processing power and memory. Thus, it can be used and turned off and the passwords will remain encrypted. One way to do this is to turn this command before setting up any passwords and turning it off after finishing the password set up commands using the following command:

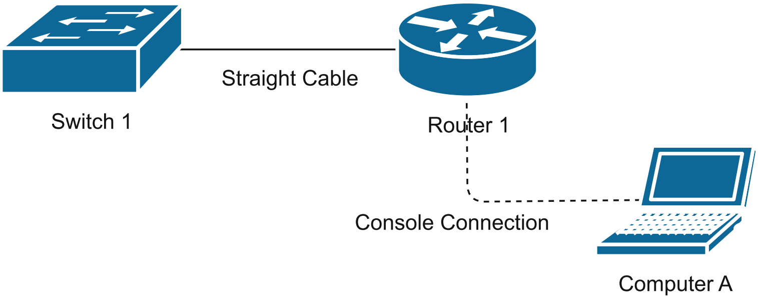
   ```
   Router(config)#no service password-encryption
   ```

   The second way to do it is after finishing the set up of all passwords, turn on password encryption, issue a ‘`show running-config`’ at the privilege mode, and then turn the password encryption off.

The encryption used here is very weak. The only purpose of it is to prevent people looking at the configuration from knowing the password.

1.5 Training Scenarios

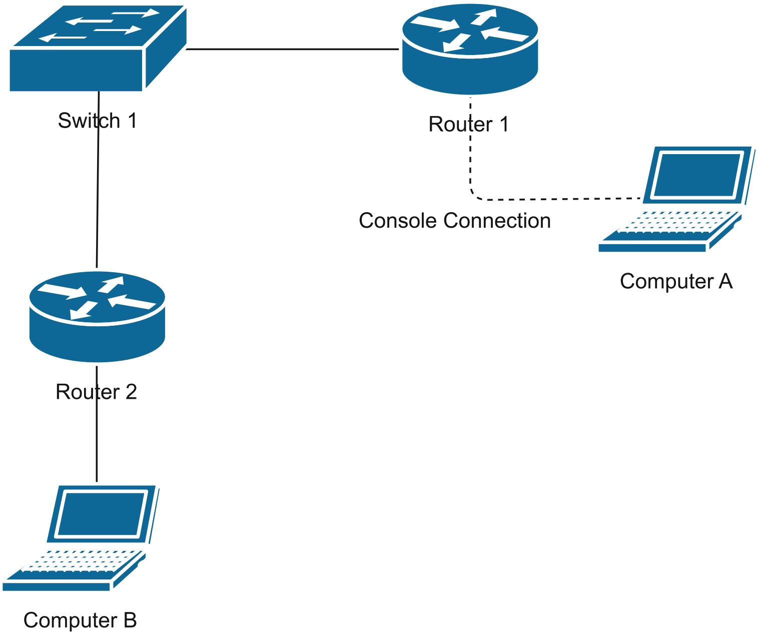
Scenario 1.1



Connect the network shown in the diagram above using a console connection to link PC A and Router 1 and a straight LAN cable to link the switch (port 1/1) to the router (interface FE0/0). Use the configuration parameters shown in the table below to do basic configuration on the router.

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	10.0.0.1/255.255.255.0

Scenario 1.2



Connect the network shown in the figure above. Router 2 and Computer B are already configured with the following settings:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.2/255.255.255.0
	Interface FE0/1 IP Address	10.0.0.1/255.255.255.0
Computer B	Ethernet Interface	10.0.0.2/255.255.255.0

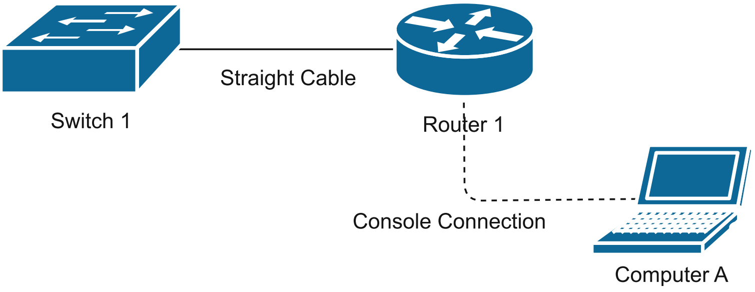
Make the console connection to Router 1 and do the following configuration steps:

1. Set up the FE0/0 interface with IP address 172.16.0.1 and subnet mask 255.255.255.0.
2. Set up the VTY Password to be MyCiscoVtY.

Now, move over to Computer B and set up a Telnet session to Router 1 using the IP address 172.16.0.1. In this Telnet session, do the following settings on Router 1:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	Interface FE0/1 IP Address	172.0.0.1/255.255.255.0

Scenario 1.3



Connect the network shown in the diagram above using a console connection to link PC A and Router 1 and a straight LAN cable to link the switch (port 1/1) to the router (interface FE0/0). Use the configuration parameters shown in the table below to do basic configuration on the router. Do not forget to enable IPv6 on the interface.

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IPv6 Address	3001::3:0:1/64

Chapter 2
Domestic Router Functions Configuration

Keywords Cisco · Router · DHCP · DHCP server · NAT · PAT · Inter-VLAN routing

2.1 How to Configure a Cisco Router as a DHCP Client

When would you need this: When your ISP gives you a dynamic IP address upon each connection or you need to configure the router to obtain its interface IP address automatically.

Special Requirements: None.

This is done using a single command:

```
Router(config-if)#ip address dhcp
```

Some service providers might ask you to use a client-id and/or a hostname of their own choice. This can be done by adding the following parameters to the command above:

```
Router(config-if)#ip address dhcp client-id interface-name hostname hostname
```

where

interface-name is the interface name that will be used for the *client-id* and

hostname is the hostname that will be used for the DHCP binding.

This hostname can be different from the one that was set for the router in the global configuration. You can use both of these parameters, one of them, or none of them.

If you need, use the ‘`ip nat outside`’ command at the interface and set up the rest of the NAT configuration as mentioned in the NAT and PAT configuration procedures in Sect. 2.4.

M.M. Alani, *Guide to Cisco Routers Configuration*,
DOI 10.1007/978-3-319-54630-8_2

2.2 How to Configure a Cisco Router as a DHCP Server

When would you need this: When using the router as a DHCP server to provide IP addresses and related information to DHCP clients.

Special Requirements: DHCP server software is supported for these series:
800, 1000, 1400, 1600, 1700 series (support for the Cisco 1700 series was added in Cisco IOS Release 12.0[2]T), 2500, 2600, 3600, 3800, MC3810, 4000, AS5100, AS5200, AS5300, 7000, 7100, 7200, MGX 8800 with an installed Route Processor Module, 12000, uBR900, uBR7200, Catalyst 5000 family switches with an installed Route Switch Module, Catalyst 6000 family switches with an installed MultiLayer Switch Feature Card, and Catalyst 8500.

The configuration steps are as follows:

1. Define the DHCP address pool:
   ```
   Router(config)#ip dhcp pool dhcp-pool-name
   Router(dhcp-config)#network network-address subnetmask
   ```
   where
   dhcp-pool-name is the DHCP pool name,
   network-address is the network address to be used by the DHCP pool, and
   subnetmask is the subnet mask for the network.
   You can replace the subnet mask by (*/prefix*) to provide the subnet mask.
2. Configure the parameters to be sent to the client:
   ```
   Router(dhcp-config)#dns-server dns-server-address
   ```
   To provide the DNS server IP address:
   ```
   Router(dhcp-config)#default-router default-gateway-address
   ```
   To provide the IP address of the default-gateway, which is usually the IP address of the router interface connected to the network.
   ```
   Router(dhcp-config)#domain-name domain
   ```
   To provide the name of the domain of the network (if in a domain environment):
   ```
   Router(dhcp-config)#netbios-name-server netbios-server-address
   ```
   To provide the IP address of the NetBIOS name server:
   ```
   Router(dhcp-config)#lease days hours minutes
   ```
   To define the lease time of the addresses given to the client. You can make it infinite, which is not advised, by using this command instead
   ```
   Router(dhcp-config)#lease infinite
   ```
   There is a large group of settings that you can configure to be sent to the clients, and I have only mentioned the most frequently used.
3. Configure the IP addresses to be excluded from the pool. This is usually done to avoid the conflicts caused by the DHCP with servers and printers. Remember to give all servers and network printers static IP addresses in the same range of the DHCP pool. Afterward, exclude these addresses from the pool to avoid conflicts.

`Router(config)#`**`ip dhcp excluded-address`** *`excluded-ip-address`*

Use the command in the previous form to exclude a single address. You can repeat it as many times as you see fit for the IP addresses you want to exclude. You can also use the same command to exclude a range of IP addresses all in a single command:

`Router(config)#`**`ip dhcp excluded-address`** *`start-ip-address end-ip-address`*

where

start-ip-address is the first address in the range to be excluded from the pool and *end-ip-address* is the last excluded address in the range.

4. Enable the DHCP service in the router:

   `Router(config)#`**`service dhcp`**

   To disable it, use

   `Router(config)#`**`no service dhcp`**

   Usually, the DHCP service is enabled by default on your router.

5. Use the following commands to check the DHCP operation on the router:

   `Router#`**`show ip dhcp binding`**

   This command shows the current bindings of addresses given to clients.

   `Router#`**`show ip dhcp server statistics`**

   This command shows the DHCP server statistics.

   `Router#`**`debug ip dhcp server`**

   This debug command is used to troubleshoot DHCP issues.

   Implementation notes:

   1. You can create a DHCP database agent that stores the DHCP binding database. A DHCP database agent is any host; for example, an FTP, TFTP, or RCP server that stores the DHCP bindings' database. You can configure multiple DHCP database agents, and you can configure the interval between database updates and transfers for each agent. To configure a database agent and database agent parameters, use the following command in global configuration mode:

      `Router(config)#`**`ip dhcp database`** *`URL`* `[`*`timeout seconds`* `|` *`write-delay seconds`*`]`

      An example *URL* is this

      ftp://user:password@192.168.0.3/router-dhcp

      If you choose not to configure a DHCP database agent, disable the recording of DHCP address conflicts on the DHCP server. To disable DHCP address conflict logging, use the following command in global configuration mode:

      `Router(config)#`**`no ip dhcp conflict logging`**

   2. DHCP service uses port 67 and 68. So, if you are using a firewall, remember to open these ports.

3. To clear DHCP server variables, use the following commands as needed:
   ```
   Router#clear ip dhcp server statistics
   Router#clear ip dhcp binding *
   ```
   If you want to clear a certain binding not all of them, replace the * in the previous command with the IP address to be cleared.

2.3 How to Configure a Cisco Router as a DHCP Server for IPv6

When would you need this: When using the router as a DHCP server to provide IPv6 in stateless and stateful configuration of DHCPv6.
Special Requirements: DHCPv6 support in IOS.

1. Create the DHCP pool:
   ```
   Router(config)#ipv6 dhcp pool pool-name
   ```
2. Configure the parameters you want to pass to the clients:
   ```
   Router(config-dhcp)#dns-server server-ipv6-address
   Router(config-dhcp)#domain-name domain
   ```
3. If you are working on a stateless address auto-configuration scenario, skip the next two steps and jump to 6.
4. Configure the IPv6 address prefix:
   ```
   Router(config-dhcp)#address prefix ipv6-address-prefix
   ```
   where the *ipv6-address-prefix* is the 64-bit hexadecimal network address prefix.
5. An optional step is to set up a link address prefix:
   ```
   Router(config-dhcp)#link-address ipv6-link-prefix
   ```
6. Enable DHCPv6 on the interface you want to be part of the DHCP process and assign a specific pool to the interface:
   ```
   Router(config-if)#ipv6 dhcp server pool-name
   ```
7. Check the address leases (in stateful addressing only):
   ```
   Router#show ipv6 dhcp lease
   ```

2.4 How to Configure DHCP Relay in Cisco Router

2.4.1 IPv4

If you have a DHCP server other than the router and you would like the router to pass the DHCP requests to this DHCP server laying outside the LAN, go to the LAN interface that does not have the DHCP server and type the following command:

```
Router(config-if)#ip helper-address dhcp-server-address
```

where *dhcp-server-address* is the IP address of the DHCP server located outside this LAN.

2.4.2 IPv6

If you have a DHCPv6 server other than the router and you would like the router to pass the DHCPv6 requests to this DHCPv6 server laying outside the LAN, go to the LAN interface that does not have the DHCPv6 server and type the following command:

```
Router(config-if)#ipv6 dhcp relay destination dhcp-server-ipv6-address
```

where *dhcp-server-ipv6-address* is the IPv6 address of the DHCP server located outside this LAN.

2.5 How to Configure NAT and PAT on a Cisco Router

When would you need this: When you want to connect a local network to the Internet and the available global IP addresses are less than the local IP addresses. This can also be used as an additional security feature.

Special Requirements: None.

There are two types of NAT that can be configured on a Cisco router: static and dynamic.

2.5.1 Static NAT Configuration

This type is used when you want to do one-to-one assignment of global (namely public) IP addresses to local IP addresses.

1. Establish static translation between an inside local address and an inside global address:

   ```
   Router(config)#ip nat inside source static local-ip-address global-ip-address
   ```

   where
   local-ip-address is the (inside) local address and
   global-ip-address is the (inside) global address.
2. Specify the local interface (the interface connected to the internal network). This is done by going to the interface configuration mode and issuing:

   ```
   Router(config-if)#ip nat inside
   ```

3. Specify the global interface (the interface connected to the external network). This is done by going to the interface configuration mode and issuing:

   ```
   Router(config-if)#ip nat outside
   ```

2.5.2 Dynamic NAT Configuration

This type is used when you want the router to do the mapping dynamically. This method is useful when you have too many global and local addresses and you do not want to do the mapping manually, or when the number of global addresses available is less than the local addresses.

This would lead us to two different scenarios:

A. The number of global IP addresses is more than one and it is equal or less than the local addresses.

   1. Define a pool of global addresses that would be employed in the translation:

      ```
      Router(config)#ip nat pool pool-name first-public-address last-public-address netmask public-subnetmask
      ```

      where
      pool-name is the name of the pool,
      first-public-address is the starting IP address of the pool,
      last-public-address is the end IP address of the pool, and
      public-subnetmask is the subnet mask of the network that the pool is part of (i.e., the global network).
   2. Define the range of local addresses permitted to participate in the translation using an access-list:

      ```
      Router(config)#access-list access-list-number permit local-network-address wildcard-mask
      ```

      where
      access-list-number is the number of the access-list, which is usually a standard accesslist; thus, the number can be any number from 1 to 99;
      local-network-address is the network address of the local network or the starting IP address of the range; and
      wildcard-mask is the wildcard mask used to define the range.
      You can issue more than one access-list sentence in the same access-list to define the specific IP address range(s). If you are not familiar with wildcard masks, refer to the note in section.
   3. Associate the pool and the local range in a dynamic NAT translation command:

      ```
      Router(config)#ip nat inside source list access-list-number pool nat-pool-name [overload]
      ```

      where
      access-list-number is the number of the access-list,
      nat-pool-name is the name of the global pool, and

overload : This parameter must be used when you have global IP addresses less than local IP addresses (this type of NAT is also known as Port Address Translation, PAT).

4. Specify the local interface. This is done by going to the interface configuration mode and issuing:
   `Router(config-if)#`**`ip nat inside`**
5. Specify the global interface. This is done by going to the interface configuration mode and issuing:
   `Router(config-if)#`**`ip nat outside`**

B. The other scenario is when there is only one global IP address and a group of local IP addresses.
In this case, the only global IP address is assigned to the interface connected to the global network.

1. Define the range of local addresses permitted to participate in the translation using an access-list:
   `Router(config)#`**`access-list`** *`access-list-number`* **`permit`** *`local-network-address wildcard-mask`*
   where
   access-list-number is the number of the access-list, which is usually a standard accesslist; thus, the number can be any number from 1 to 99,
   local-network-address is the network address of the local network or the starting IP address of the range, and
   wildcard-mask is the wildcard mask used to define the range.
   You can issue more than one access-list sentence in the same access-list to define the specific IP address range(s). If you are not familiar with wildcard masks, refer to the note in Section.
2. Associate the pool and the local range in a dynamic NAT translation command:
   `Router(config)#`**`ip nat inside source list`** *`access-list-number`* **`interface`** *`interface-type interface-number`* **`overload`**
   where
   access-list-number is the number of the access-list,
   interface-type is the type of the interface that has the global IP address (e.g., serial or Ethernet), and
   interface-number is the number of the interfaces.
   An example of the interface type and number is serial 0 or Ethernet 0/0.
3. Specify the local interface. This is done by going to the interface configuration mode and issuing:
   `Router(config-if)#`**`ip nat inside`**
4. Specify the global interface. This is done by going to the interface configuration mode and issuing:
   `Router(config-if)#`**`ip nat outside`**

2.5.3 Troubleshooting Commands

1. To show the current translations performed by NAT
   ```
   Router#show ip nat translation
   ```
   Note that these translations have a certain lifetime. They do not remain in the list forever. If you need to test your NAT configuration, ping to an outside host from an inside host and look for the translations immediately.
2. To show the static translations of NAT:
   ```
   Router#show ip nat static
   ```
3. To watch the instantaneous interactions of NAT:
   ```
   Router#debug ip nat
   ```

2.5.4 Disabling NAT

To disable NAT, you need to do the following steps:

1. Disable NAT on the local and global interfaces:
   ```
   Router(config-if)#no ip nat inside
   ```
   on the local, and
   ```
   Router(config-if)#no ip nat outside
   ```
   on the global interface.
2. Clear the contents of the translation table:
   ```
   Router#clear ip nat translations
   ```
3. Remove the NAT assignment command by preceding it with a ‘no’. For example,
   ```
   Router(config)#no ip nat inside source list access-list-number interface interface-type interface-number overload
   ```
4. Remove the access-list, if any, by putting ‘no’ ahead of the command:
   ```
   Router(config)#no access-list access-list-number
   ```

2.5.5 NAT-PT Configuration for IPv6

When would you need this: When you have IPv6-only devices that need to communicate with IPv4-only devices.
Special Requirements: None.

NAT-PT, where PT stands for Protocol Translation, is a tunneling protocol that is used to translate IPv6 into IPv4 and vice versa.

NAT-PT can operate in one of the three modes: static, dynamic, and Port Address Translation.

Before configuring NAT-PT, you need to enable IPv6 routing on the translation router using this command:

```
Router(config)#ipv6 unicast-routing
```

1. In static configuration, an IPv6 address is statically mapped into an IPv4 address using the following command:

   ```
   Router(config)#ipv6 nat v6v4 source ipv6-address ipv4-address
   ```

   where
   ipv6-address is the IPv6 address assigned to the IPv6-only host and
   ipv4-address is the IPv4 address assigned to the IPv4-only host.
   The previous command needs to be configured once for every address.
   In a similar fashion, we need to identify the reversed mapping from IPv4 to IPv6 using the following command:

   ```
   Router(config)#ipv6 nat v6v4 source ipv4-address ipv6-address
   ```

   where
   ipv6-address is the IPv6 address assigned to the IPv6-only host and
   ipv4-address is the IPv4 address assigned to the IPv4-only host.
   The next step is to enable IPv6 NAT on the IPv4 interface:

   ```
   Router(config-if)#ipv6 nat
   ```

   On the IPv6 interface, you need to assign an IPv6 address and enable IPv6 just as explained in Sect. 1.3.2.
2. In dynamic configuration, you will need to configure translation in both ways: IPv6-to-IPv4 and IPv4-to-IPv6. For the first option, IPv6-to-IPv4, you will need to identify the IPv6 addresses using an access-list and map it to an IPv4 address pool to be used in the translation.
   First, we identify the pool of IPv4 addresses using the command:

   ```
   Router(config)#ipv6 nat v6v4 pool pool-name start-address end-address prefix-length prefix-length
   ```

   where
   pool-name is the name of the NAT pool,
   start-address and end-address are the first and last addresses in the pool, and
   prefix-length is the prefix length of the IPv4 network.
   Next, we create a named access-list to identify the range of IPv6 addresses that are allowed to participate in the translation. This is done using the following commands:

   ```
   Router(config)#ipv6 access-list acl-name
   Router(config-ipv6-acl)#permit ip ipv6-source-prefix/prefix-length any
   ```

   where

acl-name is the name of the access-list,

ipv6-source-prefix is the IPv6 prefix address of the hosts that are allowed to use this NAT translation, and

prefix-length is the IPv6 network prefix length.

Repeat the last command as many times as you need to include all the addresses you want to participate in the translation.

The last step is to configure the mapping using the following command:

```
Router(config)#ipv6 nat v6v4 source list acl-name pool pool-name
```

where

acl-name is the name of the access-list identified in the previous step and

pool-name is the name of the NAT pool we identified earlier.

In the second part, we will need to identify the IPv4-to-IPv6 mapping using similar commands to the ones used before but exchanging IPv4 and IPv6 addresses.

First, we identify the pool of IPv6 addresses using the command:

```
Router(config)#ipv6 nat v6v4 pool pool-name start-address end-address prefix-length prefix-length
```

where

pool-name is the name of the NAT pool,

start-address and end-address are the first and last addresses in the pool, and

prefix-length is the prefix length of the IPv6 network.

Next, we create a numbered (or named) access-list to identify the range of IPv4 addresses that are allowed to participate in the translation. This is done using the following commands:

```
Router(config)#access-list acl-number permit ip ipv4-network-address wildcard-mask
```

where

acl-number is the number of the access-list. The number should be within the range 1-99 because it is a standard ACL;

ipv4-network-address is the IPv4 network that includes the hosts that are allowed to use this NAT translation; and

wildcard-mask is the wildcard mask that identifies the range.

Repeat the last command as many times as you need to include all the addresses you want to participate in the translation using the same access-list number.

The last step is to configure the mapping using the following command:

```
Router(config)#ipv6 nat v6v4 source list acl-number pool pool-name
```

where

acl-name is the name of the access-list identified in the previous step and

pool-name is the name of the NAT pool we identified earlier.

3. Port Address Translation is configured in an identical manner to the previous case of dynamic mapping with the exception of one small difference. In the mapping command, add the word **overload** at the end after the pool name.

4. For verification purposes, use the following commands:
   `Router#`**`show ipv6 nat translations`**
   `Router#`**`clear ipv6 nat translation *`**
   `Router#`**`debug ipv6 nat detail`**

2.6 How to Configure Inter-VLAN Routing on a Cisco Router

When would you need this: When you want to perform routing between different VLANs.
Special Requirements: You have to make sure that your router supports the frame tagging technology used between the switches.

Before jumping into the router configuration, you have to configure a port in the switch that will be connected to the router to be a trunk port. Your choice of VLAN tagging method configured on the switch (ISL or 802.1Q, 802.10, or LANE) will be the same that you will have to configure the router to operate by.

What will be done in this procedure is creating logical interfaces inside the single physical interface (on the router) that will be linking the switch to the router.

These logical interfaces will be treated as separate interfaces in the routing decisions.

1. Remove the IP address from the physical interface and turn it on:
   `Router(config-if)#`**`no ip address`**
   `Router(config-if)#`**`no shutdown`**
2. Create a logical interface to be assigned to one of the VLANs:
   `Router(config-if)#`**`int fastethernet`** *`interface-number.subinterface-number`*
   You can change the 'fastethernet' to the type you have and the *interfacenumber* with the physical interface number that you are using.
   subinterface-number represents the logical interface number (not number of logical interfaces). You can use any number here, but it is a good practice to use the same number of the VLAN that you will assign to this logical interface. For example, for the logical interface that you will use for VLAN 5, use '`int fastethernet 0/0.5`'. This way, you will easily know which interface refers to which VLAN.
3. Assign the logical interface to a VLAN number:
   `Router(config-subif)#`**`encapsulation`** *`encapsulation-type vlan-number`*

where
encapsulation-type is the encapsulation type you are using for the VLANs (e.g., `isl` or `dot1q` which is 802.1Q) and
vlan-number is the VLAN number that this logical interface will be assigned to.

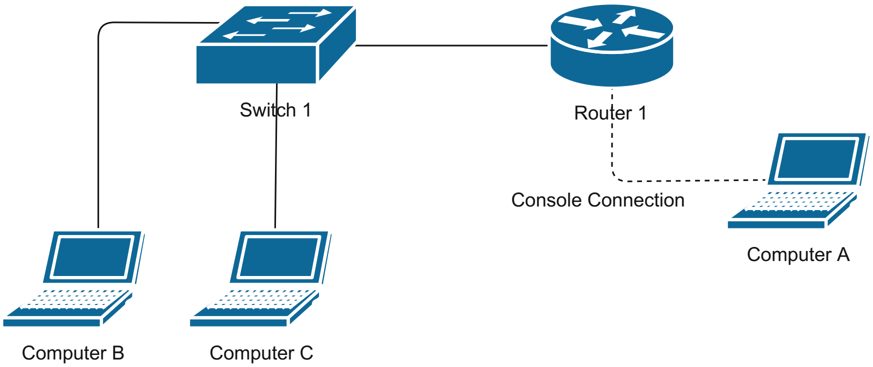
4. Assign an IP address to the logical interface:
   `Router(config-subif)#`**`ip address`** *`ip-address subnetmask`*
   where *ip-address* and *subnetmask* are the IP address and the subnet mask, respectively, you want to use. Remember to give the logical interface an IP address that is within the range of the available IP addresses in the VLAN you assigned it to. This logical interface will be the gateway to the hosts connected to this VLAN.
   Repeat the steps 2–4 for each VLAN that you want to participate in the inter-VLAN routing.
5. Configure static or dynamic routing in the way you need it. Treat the logical interfaces the exact same way you treat the physical interfaces when doing the routing.
   If you want some VLANs (i.e., networks) not to participate in the routing, you can either not include them in the routing protocol or not assign a logical interface for them.
6. You can configure access-lists in the way you find appropriate to filter the traffic going from one VLAN to another and apply them to the logical interfaces the same way you apply them to physical interfaces.

Implementation notes:

1. If you plan to let routing updates go through the router from one VLAN to another, it is necessary to turn off split-horizon. Split-horizon technology forbids the update coming from one interface to go out the same interface.
   Split-horizon can be turned off using the following command issued in the physical interface:
   `Router(config-if)#`**`no ip split-horizon`**
2. Most switches support trunks on FastEthernet or faster interfaces, and do not support the old Ethernet with 10 Mbps.

2.7 Training Scenarios

Scenario 2.1



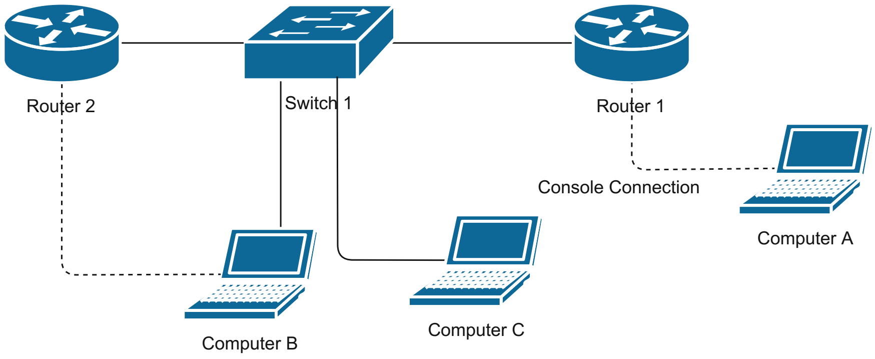
Connect the network shown in the figure above. Use Computer A and the console connection to make the following configuration:

1. Router 1 basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.15.1/24

2. Router 1 DHCP server configuration:
   Now, set up Router 1 to be a DHCP server with the following parameters:
   DHCP Pool Name: routergeek
   DHCP Network Address: 192.168.15.0/24
   Excluded Addresses: 192.168.15.1 to 192.168.15.10
   Default-Gateway Address: 192.168.15.1
   DNS Server Address: 192.168.15.1
3. Client settings:
4. Set up Computer B and Computer C to be DHCP clients using the ‘obtain IP address automatically’ setting.

Scenario 2.2



Connect the network shown in the figure above and do the following settings:

1. Using the console connection to Computer A, configure Router 1 with the following settings:

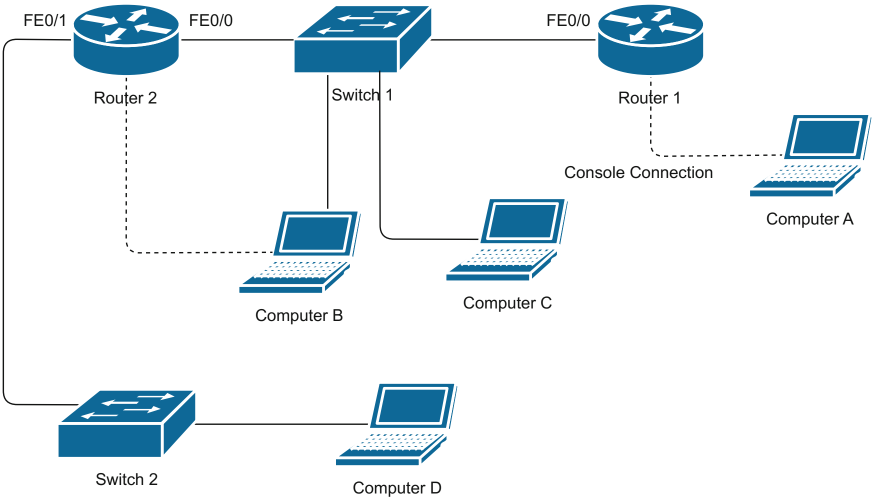
Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.255.254/16

2. Set up Router 1 to be a DHCP server:
   DHCP Pool Name: TheDHCP
   DHCP Network Address: 172.16.0.0/16
   Excluded Addresses: 172.16.255.1 to 172.16.255.254
   Default-Gateway Address: 172.16.255.254
   DNS Server Address: 172.16.255.254
3. Set up Computer B and Computer C to be DHCP clients using the ‘obtain IP address automatically’ setting.
4. Using the console connection with Computer B, do the following configuration on Router 2:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	SecondConsole
	Secret Password	Cisco
	VTY Password	SecondCiscoVTY

5. Set up the FastEthernet interface 0/0 on Router 2 to obtain its IP address settings through DHCP (i.e., DHCP client).

Scenario 2.3



Connect the network shown in the figure above and do the following settings:

1. Using the console connection to Computer A, configure Router 1 with the following settings:

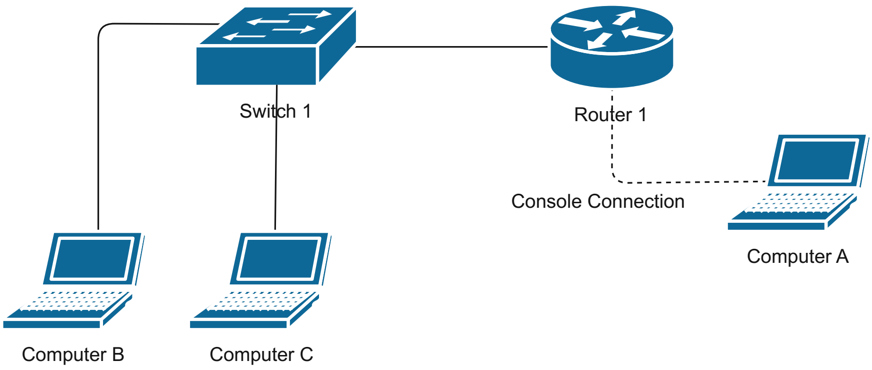
Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.254/24

2. Set up Router 1 to be a DHCP server:
   DHCP Pool Name: TheDHCP
   DHCP Network Address: 172.16.0.0/24
   Excluded Addresses: 172.16.0.240 to 172.16.255.254
   Default-Gateway Address: 172.16.0.254
   DNS Server Address: 172.16.0.254
3. Set up Computer B and Computer C to be DHCP clients using the ‘obtain IP address automatically’ setting.
4. Using the console connection with Computer B, do the following configuration on Router 2:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	SecondConsole
	Secret Password	Cisco
	VTY Password	SecondCiscoVTY
	FastEthernet 0/1 IP Address	172.16.1.254/24

5. Set up the FastEthernet interface 0/0 on Router 2 to obtain its IP address settings through DHCP (i.e., DHCP client).
6. Configure Router 2 to act as a DHCP relay to pass the DHCP requests to Router 1 (the DHCP server).
7. Configure Computer D to obtain IP address settings through DHCP.

Scenario 2.4



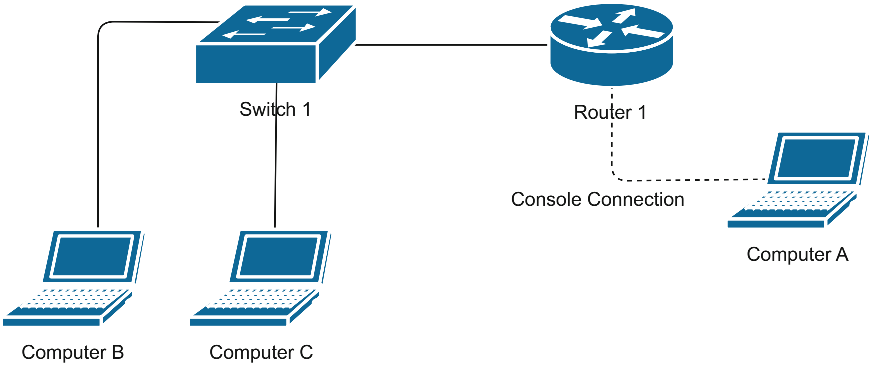
Connect the network shown in the figure above. Use Computer A and the console connection to make the following configuration:

1. Router 1 basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IPv6 Address	2001::FEFE:1/64

2. Set up Router 1 to be a stateless DHCP server with the following parameters:
   DHCP Pool Name: routergeek
   DNS Server Address: 2001::FEFE:1
   Domain Name: routergeek.org
3. Set up Computer B and Computer C to IPv6 stateless auto-configuration clients.

Scenario 2.5



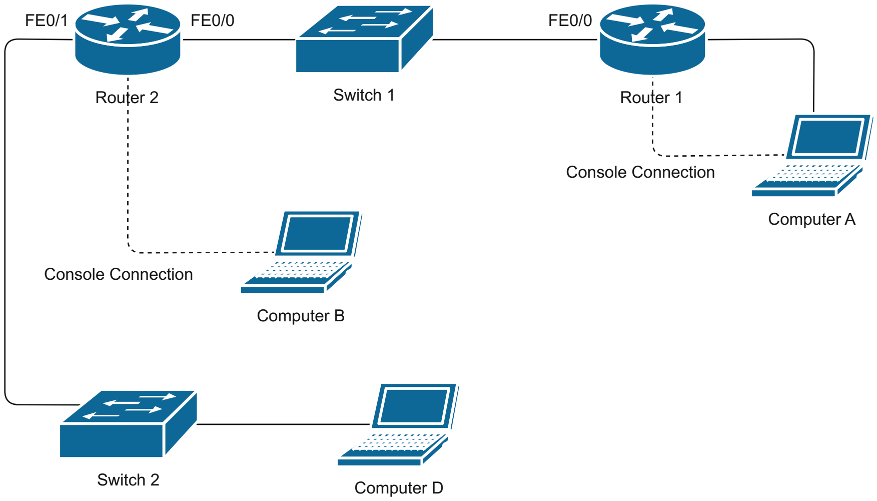
Connect the network shown in the figure above. Use Computer A and the console connection to make the following configuration:

4. Router 1 basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IPv6 Address	2001::FEFE:1/64

5. Set up Router 1 to be a stateless DHCP server with the following parameters:
   DHCP Pool Name: routergeek
   DNS Server Address: 2001::FEFE:1
   Address Prefix: 2001::/64
   Domain Name: routergeek.org
   Excluded IPv6 address: 2001::FEFE:1
6. Set up Computer B and Computer C to IPv6 stateful DHCP clients.

Scenario 2.6



Connect the network shown in the figure above. Do the following configuration:

1. On Router 1, use the console link to Computer A to change the following settings:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.0.254/24
	Interface FE0/1 IP Address	10.0.0.1/24

2. On Computer A, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.168.0.100
	Subnet Mask	255.255.255.0
	Default-Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254

3. On Router 2, using the console link to Computer B, change the following settings:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	10.0.0.2/24
	Interface FE0/1 IP Address	172.16.0.254/24

4. On Computer D, change the following settings:

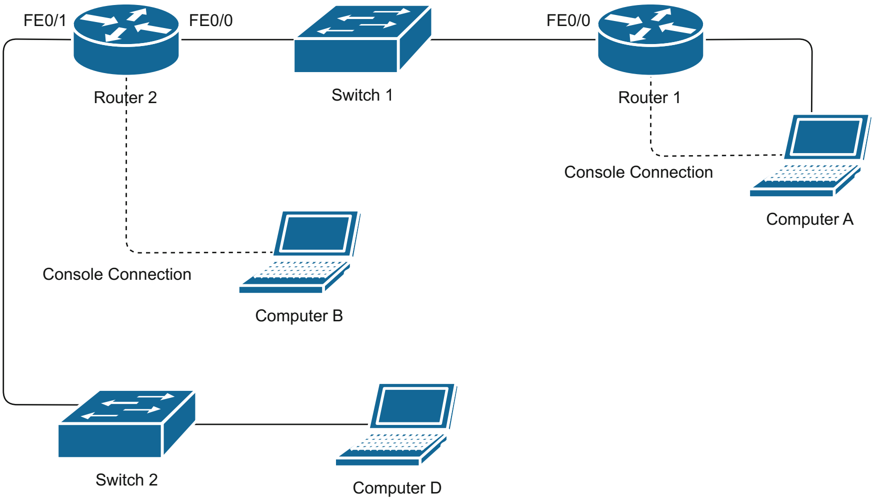
Device	Parameter	Value
Computer D	IP Address	172.16.0.10
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

5. Set up static NAT on Router 1 with the following parameters:
   Internal Address: 192.168.0.100
   External Address: 10.0.0.100
6. Set up the following static routing (detailed in Chap. 3) commands to assure delivery of data from Computer A to Computer D and back:

   a. On Router 1:
   ```
   ip route 172.16.0.0 255.255.255.0 10.0.0.2
   ```
   b. On Router 2:
   ```
   ip route 192.168.0.0 255.255.255.0 10.0.0.1
   ```

7. For testing purposes, install WireShark on Computer D. Then, do a PING command from Computer A to Computer D. On computer D, observe the source IP address of the PING message.

Scenario 2.7



Connect the network shown in the figure above. Do the following configuration:

1. On Router 1, use the console link to Computer A to change the following settings:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.0.254/24
	Interface FE0/1 IP Address	10.0.0.1/24

2. On Computer A, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.168.0.10
	Subnet Mask	255.255.255.0
	Default-Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254

3. On Router 2, using the console link to Computer B, change the following settings:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	10.0.0.2/24
	Interface FE0/1 IP Address	172.16.0.254/24

4. On Computer D, change the following settings:

Device	Parameter	Value
Computer D	IP Address	172.16.0.10
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

5. Set up dynamic NAT on Router 1 with the following parameters:
   NAT Pool Name: geekNAT
   Internal Addresses: 192.168.0.0 0.0.0.15
   External Address Range: 10.0.0.100–10.0.0.120
6. Set up the following static routing (detailed in Chap. 3) commands to assure delivery of data from Computer A to Computer D and back:

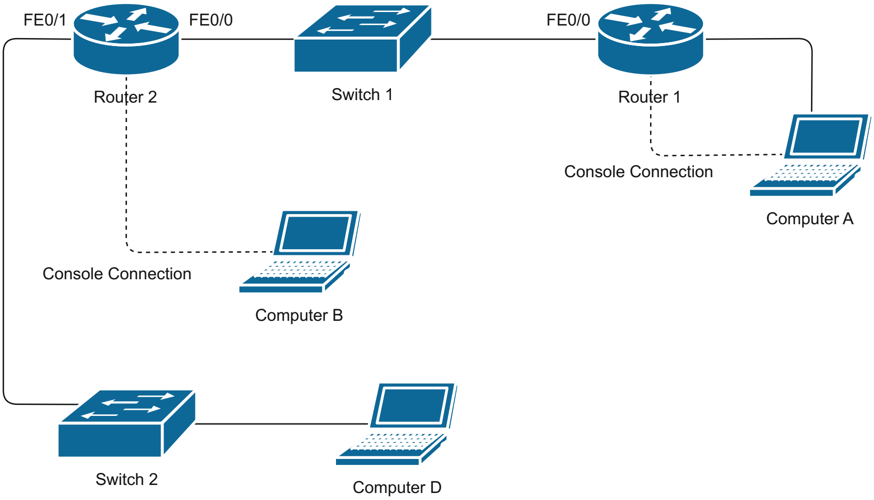
   c. On Router 1:
   ```
   ip route 172.16.0.0 255.255.255.0 10.0.0.2
   ```
   d. On Router 2:
   ```
   ip route 192.168.0.0 255.255.255.0 10.0.0.1
   ```

7. For testing purposes, install WireShark on Computer D. Then, do a PING command from Computer A to Computer D. On computer D, observe the source IP address of the PING message.
8. Change the settings of Computer A according to the following:

Device	Parameter	Value
Computer A	IP Address	192.168.0.100
	Subnet Mask	255.255.255.0
	Default-Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254

9. Enable WireShark on Computer D again. Then, do a PING command from Computer A to Computer D. On computer D, observe the source IP address of the PING message. This time, you should see the source as 192.168.0.100 because the source range is out of the acceptable NAT range.

Scenario 2.8



Connect the network shown in the figure above. Do the following configuration:

1. On Router 1, use the console link to Computer A to change the following settings:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.0.254/24
	Interface FE0/1 IP Address	10.0.0.1/30

2. On Computer A, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.168.0.10
	Subnet Mask	255.255.255.0
	Default-Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254

3. On Router 2, using the console link to Computer B, change the following settings:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	10.0.0.2/30
	Interface FE0/1 IP Address	172.16.0.254/24

4. On Computer D, change the following settings:

Device	Parameter	Value
Computer D	IP Address	172.16.0.10
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

5. Set up dynamic NAT with overload on Router 1 with the following parameters:
   Internal Addresses: 192.168.0.0 0.0.0.31
   External Address: Interface 10.0.0.1
6. Set up the following static routing (detailed in Chapter 3) commands to assure delivery of data from Computer A to Computer D and back:

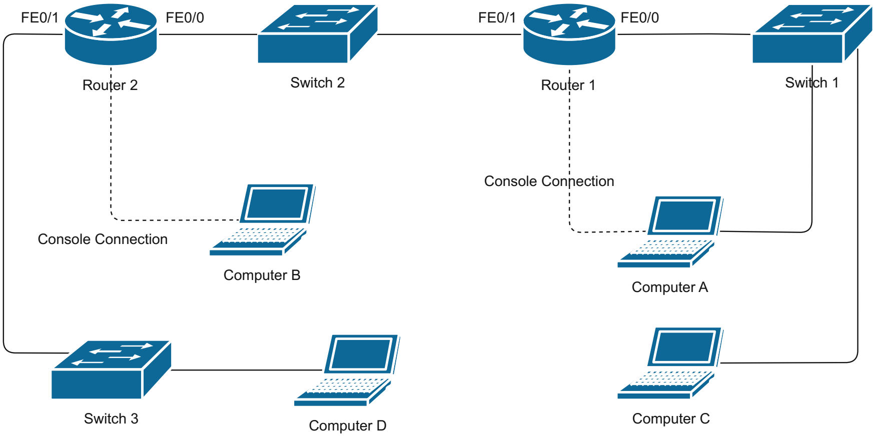
   e. On Router 1:
   ```
   ip route 172.16.0.0 255.255.255.0 10.0.0.2
   ```
   f. On Router 2:
   ```
   ip route 192.168.0.0 255.255.255.0 10.0.0.1
   ```

7. For testing purposes, install WireShark on Computer D. Then, do a PING command from Computer A to Computer D. On computer D, observe the source IP address of the PING message.
8. Change the settings of Computer A according to the following:

Device	Parameter	Value
Computer A	IP Address	192.168.0.100
	Subnet Mask	255.255.255.0
	Default-Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254

9. Enable WireShark on Computer D again. Then, do a PING command from Computer A to Computer D. On computer D, observe the source IP address of the PING message. This time, you should see the source as 192.168.0.100 because the source range is out of the acceptable NAT range.

Scenario 2.9



Connect the network shown in the figure above. Do the following configuration:

1. On Router 1, use the console link to Computer A to change the following settings:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.0.254/24
	Interface FE0/1 IP Address	10.0.0.1/25

2. On Computer A, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.168.0.10
	Subnet Mask	255.255.255.0
	Default-Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254

3. On Computer C, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.168.0.12
	Subnet Mask	255.255.255.0
	Default-Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254

4. On Router 2, using the console link to Computer B, change the following settings:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	10.0.0.2/25
	Interface FE0/1 IP Address	172.16.0.254/24

5. On Computer D, change the following settings:

Device	Parameter	Value
Computer D	IP Address	172.16.0.10
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

6. Set up dynamic NAT with overload on Router 1 with the following parameters:
   NAT Pool Name: geekNAT
   Internal Addresses: 192.168.0.0 0.0.0.15
   External Addresses: 10.0.0.5 to 10.0.0.8
7. Set up the following static routing (detailed in Chap. 3) commands to assure delivery of data from Computer A to Computer D and back:

   g. On Router 1:
   ```
   ip route 172.16.0.0 255.255.255.0 10.0.0.2
   ```
   h. On Router 2:
   ```
   ip route 192.168.0.0 255.255.255.0 10.0.0.1
   ```

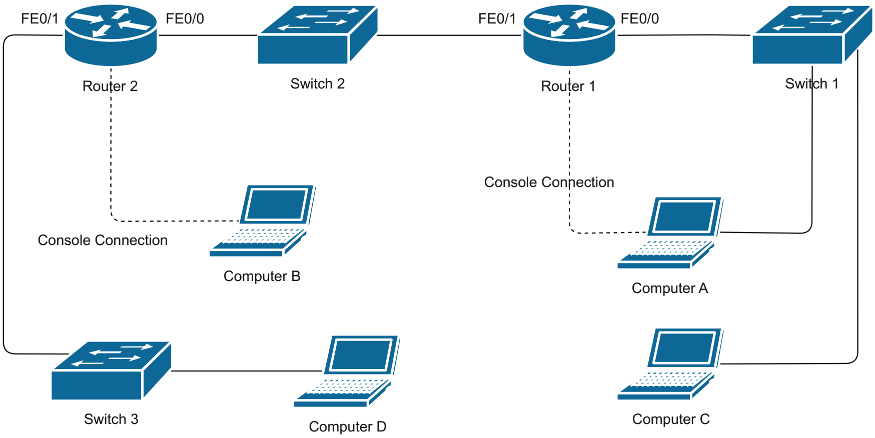
8. For testing purposes, install WireShark on Computer D. Then, do a PING command from Computer A to Computer D. And on Computer C, do a PING from Computer C to Computer D. On computer D, observe the source IP addresses of the PING messages.

9. Change the settings of Computer A according to the following:

Device	Parameter	Value
Computer A	IP Address	192.168.0.100
	Subnet Mask	255.255.255.0
	Default-Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254

10. Enable WireShark on Computer D again. Then, do a PING command from Computer A to Computer D. On computer D, observe the source IP address of the PING message. This time, you should see the source as 192.168.0.100 because the source range is out of the acceptable NAT range.

Scenario 2.10



Connect the network shown in the figure above. Do the following configuration:

1. On Router 1, use the console link to Computer A to change the following settings:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	2001::FEFE:1/64
	Interface FE0/1 IP Address	10.0.0.1/25

2. On Computer A, change the following settings:

Device	Parameter	Value
Computer A	IP v6 Address	2001::FEFE:5
	Prefix Length	64
	Default-Gateway	2001::FEFE:1
	DNS Server 1	2001::FEFE:1
	DNS Server 2	2001::FEFE:1

3. On Computer C, change the following settings:

Device	Parameter	Value
Computer C	IP Address	2001::FEFE:6
	Prefix Length	64
	Default-Gateway	2001::FEFE:1
	DNS Server 1	2001::FEFE:1
	DNS Server 2	2001::FEFE:1

4. On Router 2, using the console link to Computer B, change the following settings:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	10.0.0.2/25
	Interface FE0/1 IP Address	172.16.0.254/24

5. On Computer D, change the following settings:

Device	Parameter	Value
Computer D	IP Address	172.16.0.10
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

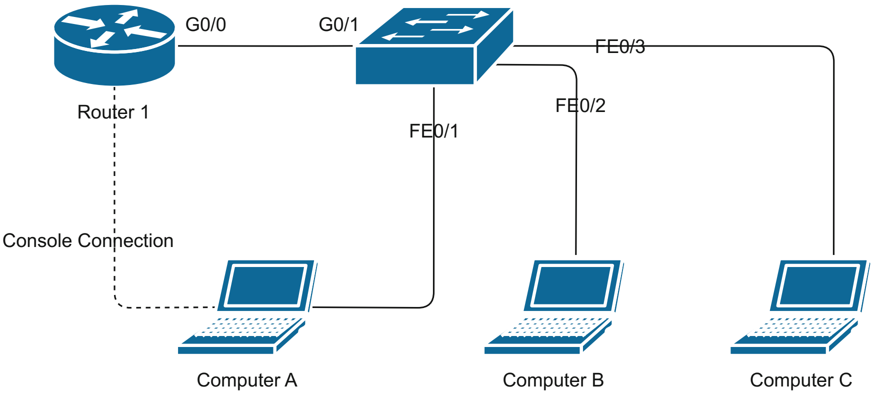
6. Set up dynamic NAT with overload on Router 1 with the following parameters:
NAT Pool Name: geekNAT
Internal Addresses: 2001::FEFE:0 /120
External Addresses: 10.0.0.5 to 10.0.0.8

7. Set up the following static routing (detailed in Chapter 3) commands to assure delivery of data from Computer A to Computer D and back:
   i. On Router 1:
   `ip route 172.16.0.0 255.255.255.0 10.0.0.2`
   j. On Router 2:
   `ipv6 route 2001::FEFE:1/64 10.0.0.1`
8. For testing purposes, install WireShark on Computer D. Then, do a PING command from Computer A to Computer D. And on Computer C, do a PING from Computer C to Computer D. On computer D, observe the source IP addresses of the PING messages.
9. Change the settings of Computer A according to the following:

Device	Parameter	Value
Computer A	IP Address	2001::FEFE:FE01
	Prefix Length	64
	Default-Gateway	2001::FEFE:1
	DNS Server 1	2001::FEFE:1
	DNS Server 2	2001::FEFE:1

10. Enable WireShark on Computer D again. Then, do a PING command from Computer A to Computer D. On computer D, observe the source IP address of the PING message. This time, you should see the source as 2001::FEFE:FE01 because the source range is out of the acceptable NAT range.

Scenario 2.11



Connect the network in the figure above. Make sure to use a switch that has at least one Gigabit Ethernet port. Follow the configuration steps outlines below:

1. On Router 1, change the following settings:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	cisco
	VTY Password	CiscoVTY

2. On Switch 1, create the following VLANs:

VLAN Number	VLAN Name	Ports to be assigned to VLAN
10	Accounts	FE0/2
20	IT	FE0/1
30	Managers	FE0/3

3. Configure port G0/1 on Switch 1 to act as trunk port with 802.1Q encapsulation.
4. On the Computers, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.168.20.100
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.20.254
	DNS Server 1	192.168.20.254
	DNS Server 2	192.168.20.254
Computer B	IP Address	192.168.10.100
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.10.254
	DNS Server 1	192.168.10.254
	DNS Server 2	192.168.10.254
Computer C	IP Address	192.168.30.100
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.30.254
	DNS Server 1	192.168.30.254
	DNS Server 2	192.168.30.254

5. For testing, PING from Computer A to B and C. The PING should not succeed.
6. Create three subinterfaces on G0/0 in Router 1 with the following parameters:

Subinterface Number	VLAN Assigned	IP Address Assigned	Encapsulation
10	10	192.168.10.254/24	dot1q
20	20	192.168.20. 254/24	dot1q
30	30	192.168.30. 254/24	dot1q

7. For testing, PING from Computer A to B and C. The PING should succeed now.

Chapter 3
Routing Protocols Configuration

Keywords Cisco · Router · Static routing · Dynamic routing · Default route · EIGRP · RIP · RIPv2 · OSPF · RIPng · Multicast · OSPFv3 · HSRP · GLBP · VRRP

3.1 Static Routing

In static routing, there is no routing information exchange. Each router is configured to use a specific route or routes to deliver user data to specific destinations.

3.1.1 How to Configure Static Routing in Cisco Routers

When would you need this: When you want to route user data to a destination through a specific path or you have a small internetwork and you do not want to bother the router with dynamic routing traffic.

Special Requirements: None.

The static route gives you more control by deciding which specific path the user data will take from a specific source to a specific destination. Configuring a single static route is done through one command on the router. The path to be taken by the packets can be identified by either the next-hop address or the exit interface.

```
Router(config)#ip route destination-network-address subnetmask next-hop-address
```

Or

```
Router(config)#ip route destination-network-address subnetmask exit-interface
```

where

destination-network-address is the network address that you want to deliver the data to.

M.M. Alani, *Guide to Cisco Routers Configuration*,
DOI 10.1007/978-3-319-54630-8_3

next-hop-address is the IP address of the neighboring interface that would help the router to deliver packets to the destination address.

subnetmask is the subnetmask of the destination network.

exit-interface is the type and number of the interface on which the packets should be forwarded to be delivered to the destination.

In IPv6, the commands are slightly different:

```
Router(config)#ipv6 route destination-network-prefix/prefix-length next-hop-address
```

Or

```
Router(config)#ipv6 route destination-network-prefix/prefix-length exit-interface
```

where

destination-network-prefix/prefix-length is the IPv6 network prefix and prefix-length that you want to deliver the data to.

next-hop-address is the IP address of the neighboring interface that would help the router to deliver packets to the destination address.

exit-interface is the type and number of the interface on which the packets should be forwarded to be delivered to the destination.

If you're using next-hop link-local address:

```
Router(config)#ipv6 route destination-network-prefix/network-prefix exit-interface next-hop-link-local-address
```

where

destination-network-prefix/prefix-length is the IPv6 network prefix and prefix-length that you want to deliver the data to.

exit-interface is the type and number of the interface on which the packets should be forwarded to be delivered to the destination.

next-hop-link-local-address is the link-local IPv6 address of the next hop.

You will need to repeat the static route configuration for every destination network you want the router to be able to communicate to. If the packet is supposed to pass through more than one router, you cannot check whether your configuration works until you configure all the routers in the path for two-way routing paths. Otherwise, you will not be able to reach the destination and back.

Generally, using the *next-hop-address* saves the router some processing of ARP requests for each packet destined to a different network. So, whenever you can, use *next-hop-ip-address* instead of *exit-interface*.

3.1.2 How to Configure Default Routes in Cisco Routers

When would you need this: When you have a stub network or you want to forward all traffic through a single specific path.

Special Requirements: None.

In concept, the default route is configured just like a regular static route that we configured in the previous section. The main difference is by telling the router to use that path for all destinations instead of a specific destination.

```
Router(config)#ip route 0.0.0.0 0.0.0.0 next-hop-address
```

Or

```
Router(config)#ip route 0.0.0.0 0.0.0.0 exit-interface
```

where

next-hop-address is the IP address of the neighboring interface that would help the router to deliver packets to the destination address.

exit-interface is the type and number of the interface on which the packets should be forwarded to be delivered to the destination.

Another way of configuring a default route is through the configuration of gateway of last resort:

```
Router(config)#ip default-network default-network-address
```

where the *default-network-address* is the network address of the default network that you want to forward all the traffic the router does not know a path to deliver the traffic to. This command must be combined with another static router telling the router the exit interface or the next-hop address that would deliver packets to the default network. The logical steps are to write a static route telling the router how to reach the default network, and then selecting the network as a default network to catch all the unknown-path traffic.

In IPv6, the default route is configured using the following commands:

```
Router(config)#ipv6 route::/0 next-hop-address
```

Or

```
Router(config)#ipv6 route::/0 exit-interface
```

where

next-hop-address is the IP address of the neighboring interface that would help the router to deliver packets to the destination address.

exit-interface is the type and number of the interface on which the packets should be forwarded to be delivered to the destination.

3.1.3 Administrative Distance of Static Routes

Administrative distance is basically how much you trust the routing information. Table 3.1 shows the default administrative distance values for different routing protocols.

The less the number in the table the more trust this source of routing information is. From the table you can see that Cisco routers trust EIGRP more than OSPF, for example. This means that if the two protocols are running on a router and both of the protocols bring a ‘best path’ to a certain destination networks, the router will select EIGRP’s path as the best path and include it in the routing table.

Table 3.1 Default administrative distance values

Source of routing information	Default administrative distance
Directly connected network	0
Static route	1
Internal EIGRP	90
IGRP	100
OSPF	110
Intermediate system-to-intermediate system (IS-IS)	115
Routing information protocol (RIP)	120

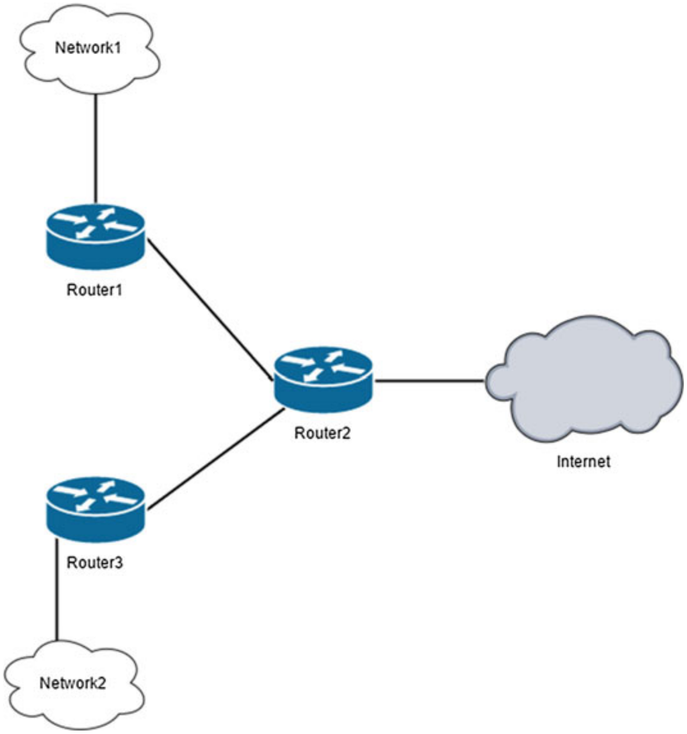


Fig. 3.1 Example of administrative distance usage

You can clearly see that static routes are more trusted than any dynamic routing protocol. In some scenarios, you will need to change the default administrative distance of the static route. For example, if you have a router that is connected to the Internet on one interface and connected to a few other routers that have internal networks connected on LAN interfaces. If you configure the default route as in the previous section, the router will not be able to forward traffic between the internal networks. Instead, all the traffic from one internal network to another will go out to the Internet because the default route is more trusted than any dynamic routing protocols you might be using. Figure 3.1 shows an example of such a network.

If a default route is configured on Router2 and EIGRP is used on the three routers to exchange routing information, data passing from Network1 to Network2 will go out to the Internet because of the default route. In this case, we can change the administrative distance of the default route to a number higher than that of EIGRP so that internal traffic will go internally and all other traffic will pass through the default route.

This is how the configuration is done for default routes:

```
Router(config)#ip route 0.0.0.0 0.0.0.0 next-hop-address
administrative-distance
```

Or

```
Router(config)#ip route 0.0.0.0 0.0.0.0 exit-interface
administrative-distance
```

where

next-hop-address is the IP address of the neighboring interface that would help the router to deliver packets to the destination address.

exit-interface is the type and number of the interface on which the packets should be forwarded to be delivered to the destination.

administrative-distance is a number between 1 and 255. The less the number the more trust the router has in this default route.

For static routes:

```
Router(config)#ip route destination-network-address sub-
netmask next-hop-address administrative-distance
```

Or

```
Router(config)#ip route destination-network-address sub-
netmask exit-interface administrative-distance
```

where

destination-network-address is the network address that you want to deliver the data to.

next-hop-address is the IP address of the neighboring interface that would help the router to deliver packets to the destination address.

subnetmask is the subnetmask of the destination network.

exit-interface is the type and number of the interface on which the packets should be forwarded to be delivered to the destination.

administrative-distance is a number between 1 and 255. The less the number the more trust the router has in this default route.

3.1.4 How to Configure IP Multicast in Cisco Routers

When would you need this: When you have a group of devices that you need to exchange data as a group.

Special Requirements: None.

Multicasts are a useful method to reduce the network traffic that is meant to go to a group of recipients but not to all devices in a network. Multicast configuration is a big topic in terms of breadth and depth. However, in this subsection, we will explain the steps of configuring the Single-RP Sparse configuration, which is the mode recommended by Cisco.

1. Enable IP multicast routing:
   ```
   Router(config)#ip multicast-routing
   ```
2. Select the sparse mode:
   ```
   Router(config)#ip pim sparse-mode
   ```
3. Identify the RP address:
   ```
   Router(config)#ip rp-address rp-ip-address
   ```
   Where the *rp-ip-address* is the IP address which will represent the whole multicast group.
4. On the interface, after identifying the IP address, identify the PIM mode as sparse, sparse-dense, dense, or dense-sparse.
   ```
   Router(config-if)#ip pim sparse-mode
   ```
   Cisco recommends using sparse mode.

3.2 Dynamic Routing

At the dynamic routing section, we will discuss the implementation of RIPv1, RIPv2, EIGRP, and Single-Area OSPF.

3.2.1 How to Configure RIPv1 and RIPv2 in Cisco Routers

When would you need this: When you need to implement a routing protocol for a small network and you need the configuration to be simple. Routing Information Protocol is the simplest that it can get.

Special Requirements: None.

1. The first thing to do is to enable the RIP protocol on the router:
   ```
   Router(config)#router rip
   ```
2. Identify the networks to be advertised using the 'network' command. Using this command, you need to identify only the networks that are directly connected to the router:
   ```
   Router(config-router)#network network-id
   ```
   If the network is subnetted, you will need to write the main network address without the need to write the subnets. For example, if you have the following subnets connected to the router (172.16.0.0/24, 172.16.1.0/24, and 172.16.2.0/24), you can put them all in single 'network' command like this:
   ```
   Router(config-router)#network 172.16.0.0
   ```
   The router is intelligent enough to figure out which subnets are connected to the router.

3. If you need to adjust the timers (update, invalid, holddown, and flush timers), use the ‘`timers basic`’ command. All the four parameters of this command, update, invalid, holddown, and flush timer consecutively, are in seconds:

   ```
   Router(config-router)#timers basic 30 180 180 240
   ```

   The example above is set with the default values of the RIP timers. Remember to keep the relativity of the timer values. Always keep it as (n 6n 6n 8n). If, for example, you set the update timer to 40, you need to make the other timers 240 240 320 consecutively. It is highly recommended that you keep the timers on their default values.
4. You will need to stop the updates from being broadcasted to the Internet, if one of the router interfaces is connected to the Internet. For this purpose, use the ‘`passive interface`’ command. This command prevents the interface from forwarding any RIP broadcasts, but keeps the interface listening to what others are saying in RIP.

   ```
   Router(config-router)#passive-interface interface-type
   interface-number
   ```

   where

   interface-type is the type of the interface, such as Serial, FastEthernet, or Ethernet.

   interface-number is the number of the interface such as 0/0 or 0/1/0.
5. RIP, by nature, sends updates as broadcast. If the router is connected through non-broadcast networks (like Frame Relay), you will need to tell RIP to send the updates on this network as unicast. This is achieved by the ‘`neighbor`’ command:

   ```
   Router(config-router)#neighbor neighbor-address
   ```

   where *neighbor-address* is the IP address of the neighbor.
6. Cisco’s implementation of RIP Version 2 supports authentication, key management, route summarization, classless inter-domain routing (CIDR), and variable-length subnet masks (VLSMs). By default, the router receives RIP Version 1 and Version 2 packets, but sends only Version 1 packets. You can configure the router to receive and send only Version 2 packets. To do so, use the ‘version’ command:

   ```
   Router(config-router)#version 2
   ```

   If you like to stick to version one, just replace the 2 in the command above with 1. Furthermore, you can control the versions of the updates sent and received on each interface to have more flexibility in support of both versions. This is achieved by the ‘`ip rip send version`’ and ‘`ip rip receive version`’ commands:

   ```
   Router(config-if)#ip rip send version 2
   Router(config-if)#ip rip receive version 1
   ```
7. Check the RIP configuration using these commands:

   ```
   Router#show ip route
   Router#show ip protocols
   Router#debug ip rip
   ```

3.2.2 How to Configure RIPng for IPv6

When would you need this: When you want to implement a simple routing protocol for a small-to-medium sized IPv6 network.

Special Requirements: None.

1. Enable IPv6 routing:
   ```
   Router(config)#ipv6 unicast-routing
   ```
2. Enable RIPng process:
   ```
   Router(config)#ipv6 router rip process-name
   ```
   where the *process-name* can be any unique process name you select. The process name is of local significance, i.e., you do not need to use the same process name on all the routers participating in the RIPng process.
3. On the interface you want to participate in the RIPng process, assign an IP address:
   ```
   Router(config-if)#ipv6 address ipv6-address/prefix-length
   ```
   where
   ipv6-address is the IPv6 address you want to assign to this interface.
   prefix-length is the IPv6 prefix length of the network this interface is connected to.
   If you do not wish to assign an IPv6 address to the interface, you can enable the IPv6 operation on the interface and let it create its own link-local address using the following command:
   ```
   Router(config-if)#ipv6 enable
   ```
4. Enable the RIPng process on the interface:
   ```
   Router(config-if)#ipv6 rip process-name
   ```
   where the *process-name* should be the process name that you have selected in step 2.
   Repeat this step on all interface you want to take part in the RIPng routing process.
5. For troubleshooting, use the following commands:
   ```
   Router#show ipv6 route
   Router#show ipv6 route rip
   Router#show ipv6 protocols
   Router#show ipv6 rip
   Router#show ipv6 rip next-hops
   Router#debug ipv6 rip
   ```

3.2.3 How to Configure EIGRP on a Cisco Router

When would you need this: When you are implementing a routing protocol on a large Internetwork and all the networking devices involved are Cisco devices or devices supporting EIGRP.

Special Requirements: EIGRP is a Cisco proprietary protocol. So, either all the routers in the Internetwork must be Cisco routers, or the routers should be EIGRP capable.

Before we start, if you have not set the bandwidth of the interfaces, set them now. For correct routing decisions, you need to set the bandwidth for the serial interfaces depending on the WAN technologies that you are using. This is done using the following command on each serial interface:

```
Router(config-if)#bandwidth bandwidth
```

where *bandwidth* is the bandwidth of the WAN connection in kilobits per second.

Next, you can start configuring EIGRP as in the following steps:

1. Enable EIGRP on the router with the command,
   ```
   Router(config)#router eigrp autonomous-system
   ```
   where *autonomous-system* is the autonomous system number. The same *autonomous-system* number must be used for all the routers that you want to exchange routing information.
2. Instruct the router to advertise the networks that are directly connected to it.
   ```
   Router(config-router)#network network-address
   ```
   where *network-address* is the network address of a network that is directly connected to the router. Repeat this step for each network that is directly connected to the specific router that you are configuring. For subnetted networks, remember that you need only to write the original network address of a group of subnets and the router will automatically identify the subnets.
   For example, if the router is connected to the networks, 172.16.1.0/24, 172.16.2.0/24, and 172.16.3.0/24, you will need to do one '`network`' command with the address 172.16.0.0.
3. By default, EIGRP packets consume a maximum of 50% of the link bandwidth, as configured with the '`bandwidth`' interface configuration command.
   You might want to change that value if a different level of link utilization is required or if the configured bandwidth does not match the actual link bandwidth (it may have been configured to influence route metric calculations). Use the following command to set the percentage of bandwidth to be used on each interface separately:
   ```
   Router(config-if)#ip bandwidth-percent eigrp bandwidth-
   percentage
   ```
   where *bandwidth-percentage* is the percentage of bandwidth to be used (e.g., 70).
4. You can change the intervals of the hello packets and the holddown timer on each interface using command:

```
Router(config-if)#ip hello-interval eigrp autonomous-system timer
```

where *autonomous-system* is the autonomous system number and *time* is the new hello packet interval time in seconds.

```
Router(config-if)#ip hold-time eigrp autonomous-system time
```

where *autonomous-system* is the autonomous system number and *time* is the new holddown time in seconds.

5. Check your configuration on the routers after configuring all the routers in the internetwork using the following commands:

   To display information about interfaces configured for EIGRP.

   ```
   Router#show ip eigrp interfaces interface-type autonomous-system
   ```

   Display the EIGRP discovered neighbors.

   ```
   Router#show ip eigrp neighbors
   ```

   To display the EIGRP topology table for a given process.

   ```
   Router#show ip eigrp topology autonomous-system
   ```

   Or

   ```
   Router#show ip eigrp topology network-address subnetmask
   ```

   To display the number of packets sent and received for all or a specified EIGRP process.

   ```
   Router#show ip eigrp traffic autonomous-system
   ```

   where

   interface-type is the interface type.

   autonomous-system autonomous system number.

   network-address and *subnetmask* are the network address and subnet mask.

3.2.4 How to Configure EIGRP Metrics on a Cisco Router

Although it is not recommended, if you need to change the way the metrics of the routes are calculated, you can set them using the command:

```
Router(config-router)#metric weights type-of-service K1 K2 K3 K4 K5
```

where

type-of-service is the type of service index and the values of k1–k5 are used to calculate the metric using the following equation:

$$\text{metric} = \left(k1 \times \text{bandwidth} + \frac{k2 \times \text{bandwidth}}{256 - \text{load}} + k3 \times \text{delay}\right) \times \frac{k5}{\text{reliability} + k4}$$

the default values are k1 = k3 = 1 and k2 = k4 = k5 = 0 and

if k5 = 0, the formula is reduces to

$$\text{metric} = \left(k1 \times \text{bandwidth} + \frac{k2 \times \text{bandwidth}}{256 - \text{load}} + k3 \times \text{delay}\right)$$

It is highly recommended that you leave the metric in the default values unless you are a highly experienced network designer.

3.2.5 How to Configure EIGRP for IPv6 on a Cisco Router

When would you need this: When you are implementing a routing protocol on a large IPv6 Internetwork and all the networking devices involved are Cisco devices or devices supporting EIGRP.

Special Requirements: EIGRP is a Cisco proprietary protocol. So, either all the routers in the Internetwork must be Cisco routers, or the routers should be EIGRP capable.

1. Enable IPv6 routing on the router:
   ```
   Router(config)#ipv6 unicast-routing
   ```
2. Enable EIGRP on the router:
   ```
   Router(config)#ipv6 router eigrp autonomous-system-number
   ```
   where *autonomous-system-number* is the number of the autonomous system in which this EIGRP process will run. Remember to use the same autonomous system number in all the routers that you want to exchange routing information.
3. Enable IPv6 on the interface you want to participate in the EIGRP process:
   ```
   Router(config-if)#ipv6 enable
   ```
   Using the '`ipv6 enable`' command will inform the router to create a link-local IPv6 address for this interface. If you want to use a different IPv6 address, you can use the following command instead of '`ipv6 enable`':
   ```
   Router(config-if)#ipv6 address ipv6-address/prefix-length
   ```
   where
   ipv6-address is the IPv6 address you want to assign to this interface.
   prefix-length is the prefix length for the IPv6 address.
4. Enable EIGRP on the interface connected to other EIGRP-enabled routers by identifying the autonomous system number this interface will be part of:
   ```
   Router(config-if)#ipv6 eigrp autonomous-system-number
   ```
   where *autonomous-system-number* is the number of the autonomous system in which this EIGRP process will run. Remember to use the same autonomous system number used in steps 2 and 4.
5. If you want to manually set up the RouterID to control the internal process of EIGRP, you can use the following optional steps:
   ```
   Router(config-if)#ipv6 router eigrp autonomous-system-number
   ```

Remember to use the same autonomous system number used in steps 2 and 4.

```
Router(config-router)#eigrp router-id router-id
```

where *router-id* is the RouterID used in the EIGRP process. The RouterID is formatted as an IPv4 address even if you are using EIGRP for IPv6 networks.

```
Router(config-router)#exit
```

6. By default, EIGRP packets consume a maximum of 50% of the link bandwidth, as configured with the 'bandwidth' interface configuration command.

   You might want to change that value if a different level of link utilization is required or if the configured bandwidth does not match the actual link bandwidth (it may have been configured to influence route metric calculations). Use the following command to set the percentage of bandwidth to be used on each interface separately:

   ```
   Router(config-if)#ipv6 bandwidth-percent eigrp autonomous-system-number bandwidth-percentage
   ```

   where

   autonomous-system-number is the number of the autonomous system in which this EIGRP process will run.

   bandwidth-percentage is the percentage of bandwidth to be used (e.g., 70).
7. To troubleshoot, use the following commands:

   ```
   Router#show ipv6 eigrp autonomous-system-number
   Router#show ipv6 eigrp interface interface-type interface-number
   Router#show ipv6 eigrp interface interface-type interface-number autonomous-system-number
   ```

3.2.6 EIGRP Implementation Notes

1. If you are using discontiguous networks, which is mostly the case, you should turn off auto-summarization using the following command:

   ```
   Router(config)#no ip auto-summary
   ```
2. You can set manual summary addresses using the following command:

   ```
   Router(config-if)#ip eigrp summary-address autonomous-system summarized-network summary-subnetmask
   ```

   where *autonomous-system* is the autonomous system number and *summarized-summarized-network* is the network address expressing the summary of multiple networks.

   summary-subnetmask is the subnetmask for the summarized address.
3. When you are using non-broadcast networking technologies such as Frame Relay and SMDS, you will need to turn off split-horizon to let EIGRP perform efficiently and effectively

`Router(config-if)#`**`no ip split-horizon`** *`autonomous-system`*
where *autonomous-system* is the autonomous system number.

4. To clear the neighbour table, use the command:
   `Router#`**`clear ip eigrp neighbors`**

3.2.7 How to Configure Single-Area OSPF on a Cisco Router

When would you need this: When you need to set up dynamic routing with Cisco and non-Cisco routers.

Special Requirements: None.

OSPF is one of the most widely used dynamic routing protocols. Cisco's version of OSPF is compatible with non-Cisco routers. If your network is large, jump into Sect. 4.1. Single-area OSPF is suitable for small-to-medium internetworks. An area is a logical grouping of routers running OSPF. All routers in the same area share the same topology database. Multiple-Area OSPF is used for large networks to prevent their topology databases from becoming out of the capability of the router.

Single-area OSPF configuration is as follows:

1. Since OSPF best route calculations rely solely on bandwidth, you need to set up the bandwidth of the serial interface involved in the routing process using the following command on the interface:
   `Router(config-if)#`**`bandwidth`** *`bandwidth`*
   where *bandwidth* is the bandwidth of the connection in kilobits per second. Remember that this command does not change the actual bandwidth. It only changes the bandwidth value being used by the routing protocol for the purpose of best path calculation.
2. Instruct the router to activate the OSPF routing process:
   `Router(config)#`**`router ospf`** *`process-number`*
   where *process-number* is the process number of OSPF. This process number is of local significance. It does not have to be the same on all routers.
3. Instruct the router to advertise the directly connected networks:
   `Router(config-router)#`**`network`** *`network-address wildcard-mask`* **`area 0`**
   where
   network-address is the network address of a directly connected network.
   wildcard-mask is the wildcard mask of the network address.
   Since we are setting a single-area OSPF, we will always use '`area 0`'.
4. Repeat step 3 for every network that is directly connected to the router.
   If you finished the first four steps on all the routers involved in the process, everything should work just fine.

If you want to do more configurations, there are a few optional advanced steps to go through:

1. To change the selection process of the DR (Designated Router) and BDR (Backup Designated Router), use the following command to change the router's OSPF priority on a certain interface:
   ```
   Router(config)#ip ospf priority priority
   ```
   where *priority* is the priority (0–255). The router with the highest priority becomes the DR. A priority of 0 means that this router will never be elected as DR.
2. To restart the whole process of DR and BDR elections, use the command:
   ```
   Router#clear ip ospf process *
   ```
3. To change the cost of a certain link in the OSPF process, use the following command:
   ```
   Router(config-if)#ip ospf cost suggested-cost
   ```
   where CC is the suggested cost (0–65, 535).

For troubleshooting, you can use the following commands:

1. To show the OSPF processes information:
   ```
   Router#show ip ospf
   ```
2. To show the OSPF database of the topology:
   ```
   Router#show ip ospf database
   ```
3. To show the OSPF operation on the interfaces:
   ```
   Router#show ip ospf interface
   ```
4. To show the OSPF neighbors table:
   ```
   Router#show ip ospf neighbor
   ```
5. To debug all the OSPF process events:
   ```
   Router#debug ip ospf events
   ```

3.2.8 How to Configure Single-Area OSPFv3 for IPv6 on a Cisco Router

When would you need this: When you need to set up dynamic routing with Cisco and non-Cisco routers.

Special Requirements: None.

1. Enable IPv6 routing on the router:
   ```
   Router(config)#ipv6 unicast-routing
   ```
2. Since OSPF best route calculations rely solely on bandwidth, you need to set up the bandwidth of the serial interface involved in the routing process using the following command on the interface:
   ```
   Router(config-if)#bandwidth bandwidth
   ```

where *bandwidth* is the bandwidth of the connection in kilobits per second. Remember that this command does not change the actual bandwidth. It only changes the bandwidth value being used by the routing protocol for the purpose of best path calculation.

3. Instruct the router to activate the OSPF routing process:

   ```
   Router(config)#ipv6 router ospf process-number
   ```

   where *process-number* is the process number of OSPF. This process number is of local significance. It does not have to be the same on all routers.
4. Enable OSFP process on each interface you want to participate in the OSPF process:

   ```
   Router(config)#interface interface-type interface-number
   Router(config-if)#ipv6 enable
   Router(config-if)#ipv6 ospf process-number Area 0
   ```

   where

   interface-type and *interface-number* are the type and number of the interface.

   process-number is the process number of OSPF identified in step 3.

   Since we are setting a single-area OSPF, we will always use '`area 0`'.

   Using the '`ipv6 enable`' command will inform the router to create a link-local IPv6 address for this interface. If you want to use a different IPv6 address, you can use the following command instead of '`ipv6 enable`':

   ```
   Router(config-if)#ipv6 address ipv6-address/prefix-length
   ```

   where

   ipv6-address is the IPv6 address you want to assign to this interface.

   prefix-length is the prefix length for the IPv6 address.
5. Repeat step 4 for every network that is directly connected to the router.

   If you finished the first four steps on all the routers involved in the process, everything should work just fine.

If you want to do more configurations, there are a few optional advanced steps to go through:

1. To change the selection process of the DR (Designated Router) and BDR (Backup Designated Router), use the following command to change the router's OSPF priority on a certain interface:

   ```
   Router(config)#ipv6 ospf priority priority
   ```

   where *priority* is the priority (0–255). The router with the highest priority becomes the DR. A priority of 0 means that this router will never be elected as DR.
2. To restart the whole process of DR and BDR elections, use the command:

   ```
   Router#clear ipv6 ospf process *
   ```

3. To change the cost of a certain link in the OSPF process, use the following command:

   ```
   Router(config-if)#ipv6 ospf cost suggested-cost
   ```

where CC is the suggested cost (0–65,535)

For troubleshooting, you can use the following commands:

1. To show the OSPF processes information:
   `Router#`**`show ipv6 ospf`**
2. To show the OSPF database of the topology:
   `Router#`**`show ipv6 ospf database`**
3. To show the OSPF operation on the interfaces:
   `Router#`**`show ipv6 ospf interface`**
4. To show the OSPF neighbors table:
   `Router#`**`show ipv6 ospf neighbor`**
5. To debug all the OSPF process events:
   `Router#`**`debug ipv6 ospf events`**

3.3 How to Configure HSRP on a Cisco Router

When would you need this: When your network design requires redundancy and high availability.

Special Requirements: None.

To understand why and how HSRP protocol works, you need to look into an example. For each local network, there is a default-gateway. This default-gateway is usually the LAN interface of the router. If your network design requires high availability and redundancy, you can use HSRP to set up a different interface in a different router to operate as a standby interface such that whenever the main default-gateway fails, the standby becomes active and the network operation is not interrupted.

HSRP operates by setting a main IP address and a standby IP address for the routers' interfaces that are taking part in the HSRP operation. Most of the time, the IP address used as the default-gateway is called the *virtual IP*. Let us jump into the configuration:

1. On the first router, configure the main IP address:
   `Router1(config-if)#`**`ip address`** *`ip-address subnetmask`*
2. On the first router, configure the virtual IP address:
   `Router1(config-if)#`**`standby`** *`hsrp-group-number`* **`ip`** *`virtual-ip-address`*
3. On the first router, set up the priority of the virtual IP
   `Router1(config-if)#`**`standby`** *`hsrp-group-number`* **`priority`** *`standby-priority`*
   where
   ip-address and *subnetmask* are the IP main address and subnetmask of the interface.

hsrp-group-number is the HSRP group number. You have to use the same number in all the routers that you want to participate in the same HSRP process.

virtual-ip-address is the virtual IP address to be used when the standby router becomes active.

standby-priority is the priority of the standby IP address. The number can be between 0 and 255. The router with the highest priority will be the active one.

4. On the second router, configure the main IP address (which is different from the one used in Router1):

   `Router2(config-if)#`**`ip address`** *`ip-address subnetmask`*

5. On the second router, configure the virtual IP address:

   `Router2(config-if)#`**`standby`** *`hsrp-group-number`* **`ip`** *`virtual-ip-address`*

6. On the first router, set up the priority of the virtual IP

   `Router2(config-if)#`**`standby`** *`hsrp-group-number`* **`priority`** *`standby-priority`*

   where

   ip-address and *subnetmask* are the IP main address and subnetmask of the interface.

   hsrp-group-number is the HSRP group number. You have to use the same number in all the routers that you want to participate in the same HSRP process.

   virtual-ip-address is the virtual IP address to be used when the standby router becomes active.

   standby-priority is the priority of the standby IP address. The number can be between 0 and 255. The router with the highest priority will be the active one.

7. You can troubleshoot using the commands:

   ```
   Router#show standby
   Router#show standby brief
   Router#show standby all
   ```

3.4 How to Configure GLBP on a Cisco Router

When would you need this: When your network design requires redundancy and high availability.

Special Requirements: None.

To understand why and how GLBP protocol works, you need to look into an example. For each local network, there is a default-gateway. This default-gateway is usually the LAN interface of the router. If your network design requires high availability and redundancy, you can use GLBP to set up a different interface in a different router to operate as a standby interface such that whenever the main default-gateway fails, the standby becomes active and the network operation is not interrupted.

GLBP operates by setting a main IP address and a standby IP address for the routers' interfaces that are taking part in the GLBP operation. Most of the time, the IP address used as the default-gateway is called the *virtual IP*. Let us jump into the configuration:

1. On the first router, configure the main IP address:
   ```
   Router1(config-if)#ip address ip-address subnetmask
   ```
2. On the first router, configure the virtual IP address:
   ```
   Router1(config-if)#glbp glbp-group-number ip virtual-ip-address
   ```
3. On the first router, set up the priority of the virtual IP:
   ```
   Router1(config-if)#glbp glbp-group-number priority standby-priority
   ```
   where

   ip-address and *subnetmask* are the IP main address and subnetmask of the interface.

   glbp-group-number is the GLBP group number. You have to use the same number in all the routers that you want to participate in the same GLBP process.

   virtual-ip-address is the virtual IP address to be used when the standby router becomes active.

   standby-priority is the priority of the standby IP address. The number can be between 0 and 255. The router with the highest priority will be the active one.
4. On the second router, configure the main IP address (which is different from the one used in Router1):
   ```
   Router2(config-if)#ip address ip-address subnetmask
   ```
5. On the second router, configure the virtual IP address:
   ```
   Router2(config-if)#glbp glbp-group-number ip virtual-ip-address
   ```
6. On the first router, set up the priority of the virtual IP
   ```
   Router2(config-if)#glbp glbp-group-number priority standby-priority
   ```
   where

   ip-address and *subnetmask* are the IP main address and subnetmask of the interface.

   glbp-group-number is the GLBP group number. You have to use the same number in all the routers that you want to participate in the same GLBP process.

   virtual-ip-address is the virtual IP address to be used when the standby router becomes active.

   standby-priority is the priority of the standby IP address. The number can be between 0 and 255. The router with the highest priority will be the active one.
7. You can troubleshoot using the commands:
   ```
   Router#show glbp
   Router#show glbp brief
   ```

3.5 How to Configure VRRP on a Cisco Router

When would you need this: When your network design requires redundancy and high availability.

Special Requirements: None.

VRRP operates by setting a main IP address and a standby IP address for the routers' interfaces that are taking part in the VRRP operation. Most of the time, the IP address used as the default-gateway is called the *virtual IP*. Let us jump into the configuration:

1. On the first router, configure the main IP address:
   ```
   Router1(config-if)#ip address ip-address subnetmask
   ```
2. On the first router, configure the virtual IP address:
   ```
   Router1(config-if)#vrrp group-number ip virtual-ip-address
   ```
3. On the first router, set up the priority of the virtual IP
   ```
   Router1(config-if)#vrrp group-number priority standby-priority
   ```
   where

   ip-address and *subnetmask* are the IP main address and subnetmask of the interface.

   group-number is the VRRP group number. You have to use the same number in all the routers that you want to participate in the same VRRP process.

   virtual-ip-address is the virtual IP address to be used when the standby router becomes active.

   standby-priority is the priority of the standby IP address. The number can be between 0 and 255. The router with the highest priority will be the active one.
4. On the second router, configure the main IP address (which is different from the one used in Router1 and from the virtual IP):
   ```
   Router2(config-if)#ip address ip-address subnetmask
   ```
5. On the second router, configure the virtual IP address:
   ```
   Router2(config-if)#vrrp group-number ip virtual-ip-address
   ```
6. On the first router, set up the priority of the virtual IP
   ```
   Router2(config-if)#vrrp group-number priority standby-priority
   ```
   where

   ip-address and *subnetmask* are the IP main address and subnetmask of the interface.

   group-number is the GLBP group number. You have to use the same number in all the routers that you want to participate in the same GLBP process.

   virtual-ip-address is the virtual IP address to be used when the standby router becomes active.

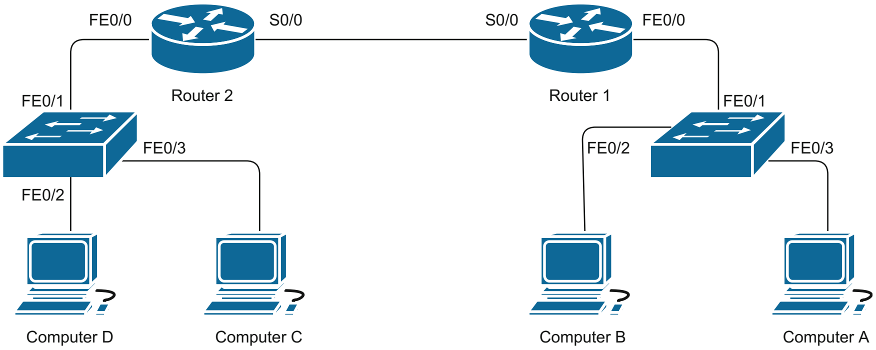
standby-priority is the priority of the standby IP address. The number can be between 0 and 255. The router with the highest priority will be the active one.

7. You can troubleshoot using the commands:

```
Router#show vrrp
Router#show vrrp brief
```

3.6 Training Scenarios

Scenario 3.1



Connect the network shown in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30

2. On Router 2, do the basic configuration:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2

(continued)

(continued)

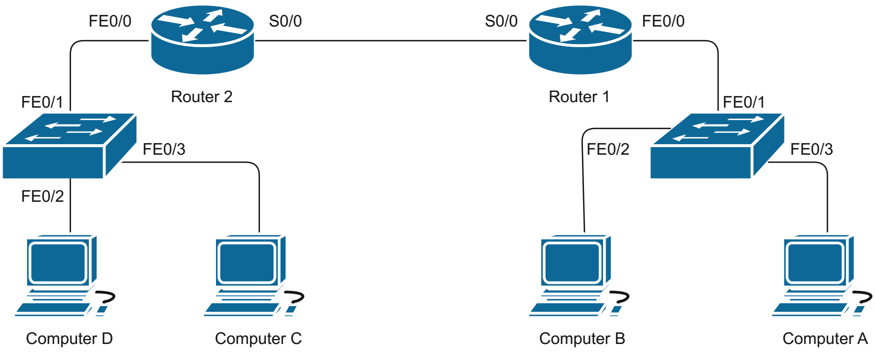
Device	Parameter	Value
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	172.16.1.254/24
	Interface S0/0 IP Address	10.0.0.2/30

3. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IP Address	172.16.0.1
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254
Computer B	IP Address	172.16.0.2
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254
Computer C	IP Address	172.16.1.1
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.1.254
	DNS Server 1	172.16.1.254
	DNS Server 2	172.16.1.254
Computer D	IP Address	172.16.1.2
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.1.254
	DNS Server 1	172.16.1.254
	DNS Server 2	172.16.1.254

4. Configure static routing on Router 1:
   Destination 172.16.1.0/24, next hop 10.0.0.2
5. Configure static routing on Router 2:
   Destination 172.16.0.0/24, next hop 10.0.0.1
6. For testing, PING from Computer A to C and D.

Scenario 3.2



Connect the network shown in the figure above and configure the following settings:

7. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IPv6 Address	2001:2001::1/64
	Interface S0/0 IP Address	10.0.0.1/30

8. On Router 2, do the basic configuration:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IPv6 Address	2001:2002::1/64
	Interface S0/0 IP Address	10.0.0.2/30

9. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IPv6 Address	2001:2001::5
	Prefix Length	64
	Default-Gateway	2001:2001::1
	DNS Server 1	2001:2001::1
	DNS Server 2	2001:2001::1

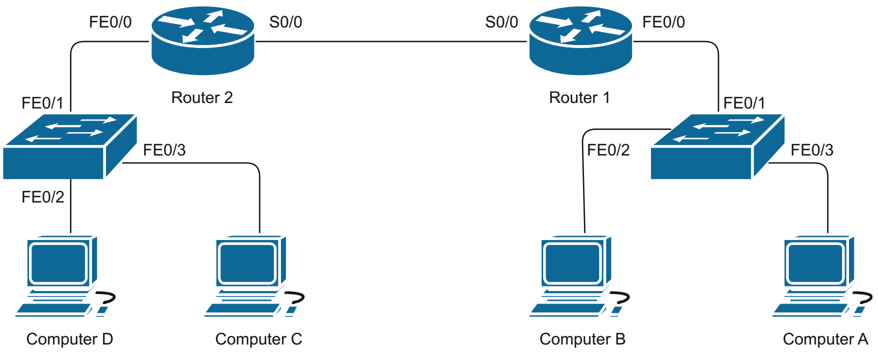
(continued)

(continued)

Device	Parameter	Value
Computer B	IPv6 Address	2001:2001::7
	Prefix Length	64
	Default-Gateway	2001:2001::1
	DNS Server 1	2001:2001::1
	DNS Server 2	2001:2001::1
Computer C	IPv6 Address	2001:2002::8
	Prefix Length	64
	Default-Gateway	2001:2002::1
	DNS Server 1	2001:2002::1
	DNS Server 2	2001:2002::1
Computer D	IPv6 Address	2001:2002::9
	Prefix Length	64
	Default-Gateway	2001:2002::1
	DNS Server 1	2001:2002::1
	DNS Server 2	2001:2002::1

10. Configure static routing on Router 1:
    Destination 2001:2002::/64, next hop 10.0.0.2
11. Configure Static Routing on Router 2:
    Destination 2001:2001::/64, next hop 10.0.0.1
12. For testing, PING from Computer A to C and D.

Scenario 3.3



Connect the network shown in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.10.254/24
	Interface S0/0 IP Address	10.0.0.1/30

2. On Router 2, do the basic configuration:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IPv6 Address	2001:2001::1/64
	Interface S0/0 IP Address	10.0.0.2/30

3. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.168.10.1
	Subnet Mask	255.255.255.0
	Default-Gateway	192.168.10.254
	DNS Server 1	192.168.10.254
	DNS Server 2	192.168.10.254
Computer B	IP Address	192.168.10.2
	Subnet Mask	255.255.255.0
	Default-Gateway	192.168.10.254
	DNS Server 1	192.168.10.254
	DNS Server 2	192.168.10.254
Computer C	IP Address	2001:2001::5
	Prefix Length	64
	Default-Gateway	2001:2001::1
	DNS Server 1	2001:2001::1
	DNS Server 2	2001:2001::1

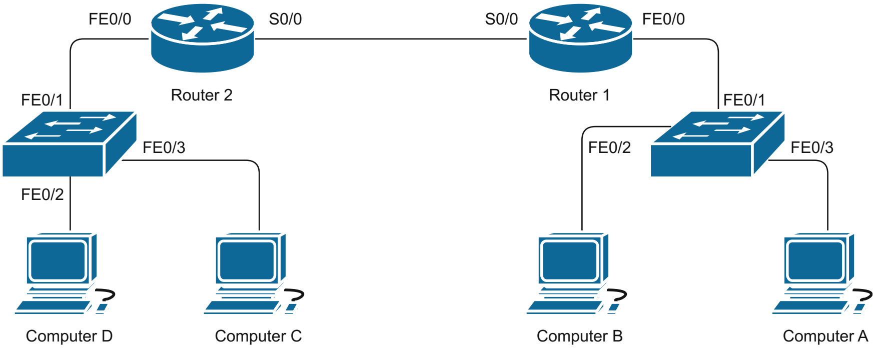
(continued)

(continued)

Device	Parameter	Value
Computer D	IP Address	2001:2001::6
	Prefix Length	64
	Default-Gateway	2001:2001::1
	DNS Server 1	2001:2001::1
	DNS Server 2	2001:2001::1

4. Configure a default routing on Router 1:
   Destination 0.0.0.0/0, next hop 10.0.0.2
5. Configure static routing on Router 2:
   Destination ::/0, next hop 10.0.0.1
6. For testing, PING from Computer A to C and D.

Scenario 3.4



Connect the network shown in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.16.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30

2. On Router 2, do the basic configuration:

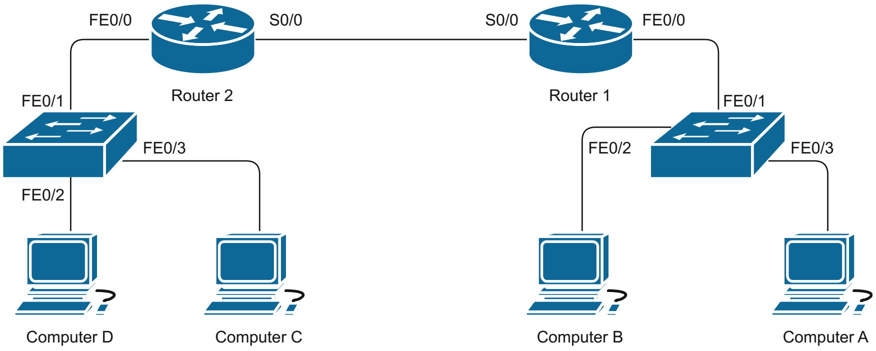
Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	192.16.1.254/24
	Interface S0/0 IP Address	10.0.0.2/30

3. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.16.0.1
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.0.254
	DNS Server 1	192.16.0.254
	DNS Server 2	192.16.0.254
Computer B	IP Address	192.16.0.2
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.0.254
	DNS Server 1	192.16.0.254
	DNS Server 2	192.16.0.254
Computer C	IP Address	192.16.1.1
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254
Computer D	IP Address	192.16.1.2
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254

4. Configure RIPv2 dynamic routing on Router 1 with the advertised networks 10.0.0.0 and 192.16.0.0.
5. Configure RIPv2 dynamic routing on Router 2 with the advertised networks 10.0.0.0 and 192.16.1.0.
6. For testing, PING from Computer A to C and D and show the routing tables in both routers.

Scenario 3.5



Connect the network shown in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.16.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30

2. On Router 2, do the basic configuration:

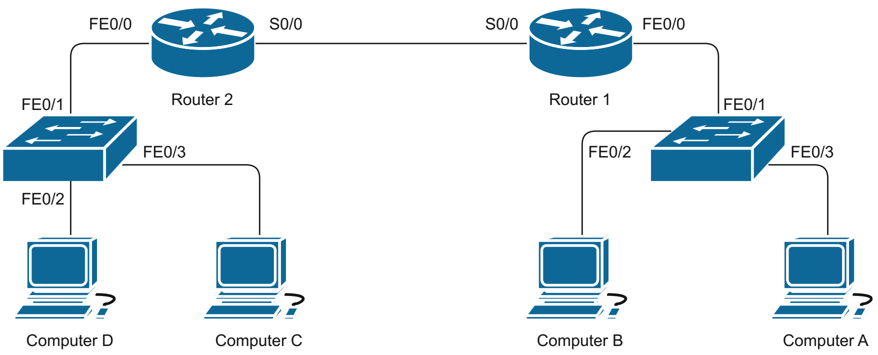
Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	192.16.1.254/24
	Interface S0/0 IP Address	10.0.0.2/30

3. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.16.0.1
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.0.254
	DNS Server 1	192.16.0.254
	DNS Server 2	192.16.0.254
Computer B	IP Address	192.16.0.2
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.0.254
	DNS Server 1	192.16.0.254
	DNS Server 2	192.16.0.254
Computer C	IP Address	192.16.1.1
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254
Computer D	IP Address	192.16.1.2
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254

4. Configure EIGRP dynamic routing on Router 1 with the advertised networks 10.0.0.0 and 192.16.0.0 in autonomous system number 10.
5. Configure EIGRP dynamic routing on Router 2 with the advertised networks 10.0.0.0 and 192.16.1.0 in autonomous system number 10.
6. For testing, PING from Computer A to C and D and show the routing tables in both routers.
7. To see the effect of having a different autonomous system number, remove EIGRP from Router 2. Reconfigure EIGRP on Router 2 with an autonomous systems number of 20. Check the connectivity and routing tables now.

Scenario 3.6



Connect the network shown in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30

2. On Router 2, do the basic configuration:

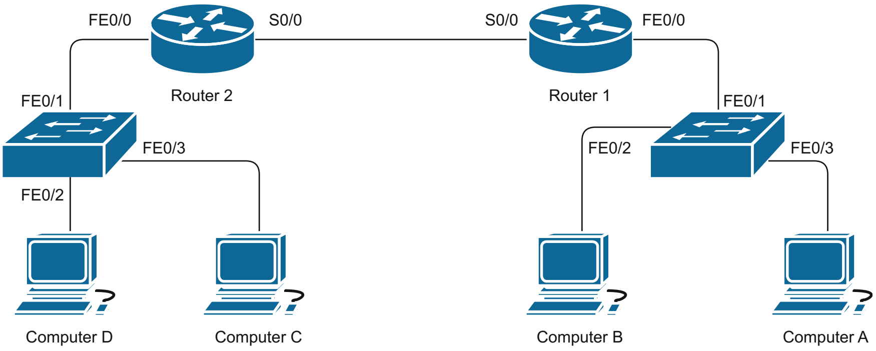
Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	172.16.1.254/24
	Interface S0/0 IP Address	10.0.0.2/30

3. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IP Address	172.16.0.1
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254
Computer B	IP Address	172.16.0.2
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254
Computer C	IP Address	172.16.1.1
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.1.254
	DNS Server 1	172.16.1.254
	DNS Server 2	172.16.1.254
Computer D	IP Address	172.16.1.2
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.1.254
	DNS Server 1	172.16.1.254
	DNS Server 2	172.16.1.254

4. Configure EIGRP dynamic routing on Router 1 with the advertised networks 10.0.0.0 and 172.16.0.0 in autonomous system number 10.
5. Configure EIGRP dynamic routing on Router 2 with the advertised networks 10.0.0.0 and 172.16.1.0 in autonomous system number 10.
6. For testing, PING from Computer A to C and D and show the routing tables in both routers.
7. With everything else left to default, routers with the previous configuration will not operate properly. The reason is that the local networks connected to the routers are discontiguous networks. In EIGRP, addresses are summarized by default. Hence, both routers will say to each other ‘I have 172.16.0.0/16 directly connected to me.’
8. Turn off auto-summary in EIGRP in both routers.
9. PING again from Computer A to C and D and show the routing tables in both routers. Both routers should operate properly.

Scenario 3.7



Connect the network shown in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.16.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30

2. On Router 2, do the basic configuration:

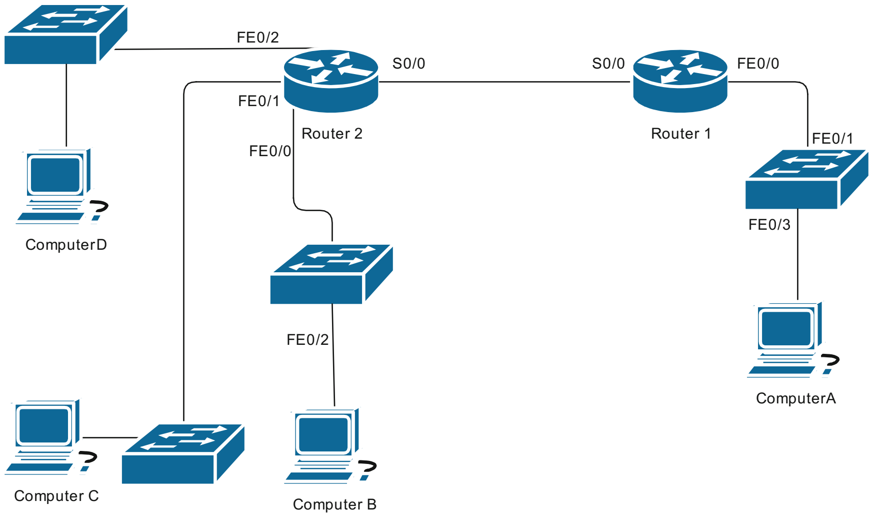
Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	192.16.1.254/24
	Interface S0/0 IP Address	10.0.0.2/30

3. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.16.0.1
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.0.254
	DNS Server 1	192.16.0.254
	DNS Server 2	192.16.0.254
Computer B	IP Address	192.16.0.2
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.0.254
	DNS Server 1	192.16.0.254
	DNS Server 2	192.16.0.254
Computer C	IP Address	192.16.1.1
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254
Computer D	IP Address	192.16.1.2
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254

4. Configure OSPF dynamic routing on Router 1 with the advertised networks 10.0.0.0 and 192.16.0.0 in area 0.
5. Configure OSPF dynamic routing on Router 2 with the advertised networks 10.0.0.0 and 192.16.1.0 in area 0.
6. For testing, PING from Computer A to C and D and show the routing tables in both routers.
7. To see the effect of having a different process number, remove OSPF from Router 2. Reconfigure OSPF on Router 2 with a process number different from Router 1. Check the connectivity and routing tables now.

Scenario 3.8



Connect the network in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.19.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30

2. On Router 2, do the basic configuration:

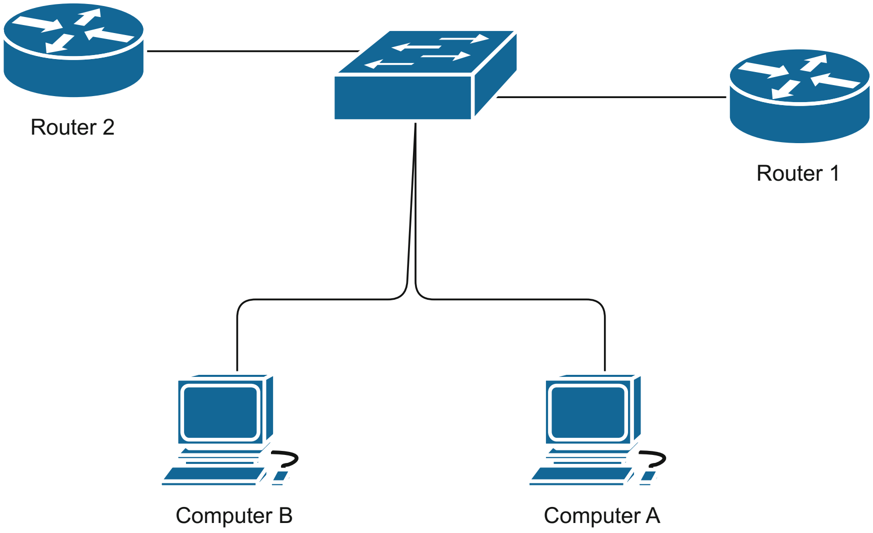
Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	192.16.1.254/24
	Interface FE0/1 IP Address	192.16.2.254/24
	Interface FE0/2 IP Address	192.16.3.254/24
	Interface S0/0 IP Address	10.0.0.2/30

3. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.19.0.1
	Subnet Mask	255.255.255.0
	Default-Gateway	192.19.0.254
	DNS Server 1	192.19.0.254
	DNS Server 2	192.19.0.254
Computer B	IP Address	192.16.1.1
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254
Computer C	IP Address	192.16.2.1
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.2.254
	DNS Server 1	192.16.2.254
	DNS Server 2	192.16.2.254
Computer D	IP Address	192.16.3.1
	Subnet Mask	255.255.255.0
	Default-Gateway	192.16.3.254
	DNS Server 1	192.16.3.254
	DNS Server 2	192.16.3.254

4. Configure EIGRP dynamic routing on Router 1 with the advertised networks 10.0.0.0 and 192.16.0.0 in autonomous system number 10.
5. Configure EIGRP dynamic routing on Router 2 with the advertised networks 10.0.0.0, 192.16.1.0, 192.16.2.0, and 192.16.3.0 in autonomous system number 10.
6. On the interface advertising the EIGRP routers on Router 2 (interface S0/0), configure EIGRP summary IP address of 192.16.0.0 with subnet mask of 255.255.0.0 which summarizes all the three internal networks connected to Router 2. This will reduce the advertisement size sent from the router.
7. For testing, PING from Computer A to C and D and show the routing tables in both routers.

Scenario 3.9



Connect the network shown in the figure above. Configure the following settings:

1. On Router 1:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.254/24

2. On Router 2:

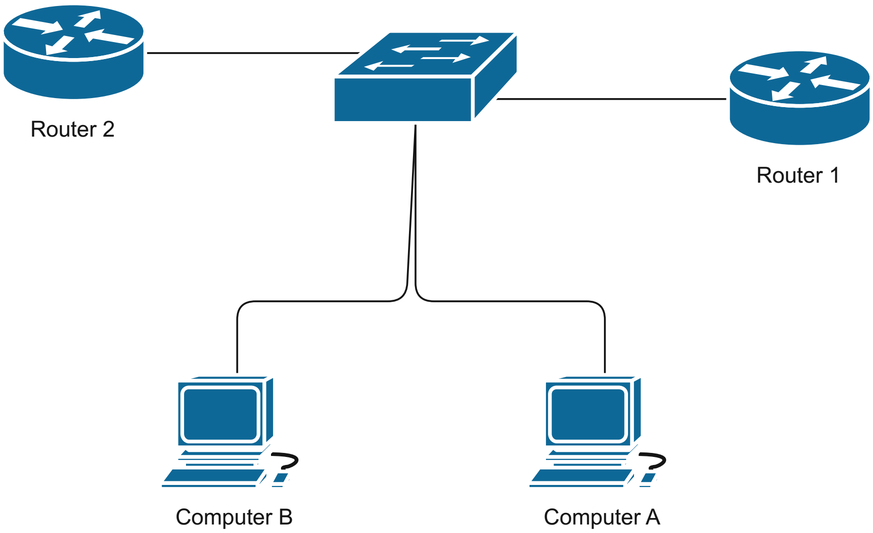
Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	CiscoConsole2
	Secret Password	Cisco2
	VTY Password	CiscoVTY2
	Interface FE0/0 IP Address	172.16.0.253/24

3. Set up HSRP virtual IP address on Router 1 and Router 2 to be 172.16.0.250. Give Router 1 a priority of 200 and Router 2 a priority of 100.
4. Configure the computers with the following settings:

Device	Parameter	Value
Computer A	IP Address	172.16.0.1/24
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.250
	DNS Server 1	172.16.0.250
	DNS Server 2	172.16.0.250
Computer B	IP Address	172.16.0.2
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.250
	DNS Server 1	172.16.0.250
	DNS Server 2	172.16.0.250

5. Do a continuous PING (use –t parameter in Windows, or just the PING command in Linux) from Computer A to the default-gateway.
6. While the PING is continuously showing results, switch off Router 1.
7. Count the downtime until Router 2 receives the flag from the switched-off Router 1.

Scenario 3.10



Connect the network shown in the figure above. Configure the following settings:

1. On Router 1:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.254/24

2. On Router 2:

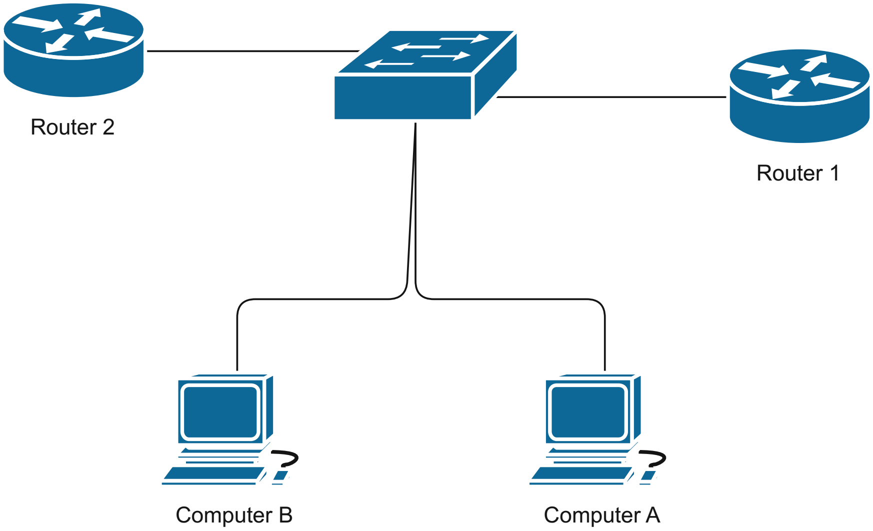
Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	CiscoConsole2
	Secret Password	Cisco2
	VTY Password	CiscoVTY2
	Interface FE0/0 IP Address	172.16.0.253/24

3. Set up GLBP virtual IP address on Router 1 and Router 2 to be 172.16.0.250. Give Router 1 a priority of 200 and Router 2 a priority of 100.
4. Configure the computers with the following settings:

Device	Parameter	Value
Computer A	IP Address	172.16.0.1/24
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.250
	DNS Server 1	172.16.0.250
	DNS Server 2	172.16.0.250
Computer B	IP Address	172.16.0.2
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.250
	DNS Server 1	172.16.0.250
	DNS Server 2	172.16.0.250

5. Do a continuous PING (use –t parameter in Windows, or just the PING command in Linux) from Computer A to the default-gateway.
6. While the PING is continuously showing results, switch off Router 1.
7. Count the downtime until Router 2 receives the flag from the switched-off Router 1.

Scenario 3.11



Connect the network shown in the figure above. Configure the following settings:

1. On Router 1:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.254/24

2. On Router 2:

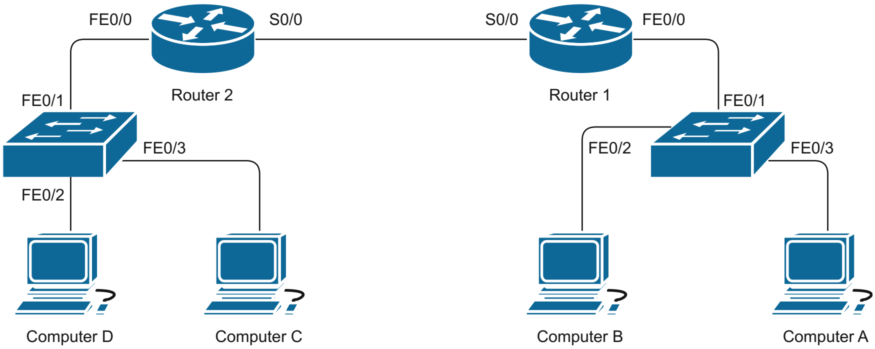
Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	CiscoConsole2
	Secret Password	Cisco2
	VTY Password	CiscoVTY2
	Interface FE0/0 IP Address	172.16.0.253/24

3. Set up VRRP virtual IP address on Router 1 and Router 2 to be 172.16.0.250. Give Router 1 a priority of 200 and Router 2 a priority of 100.
4. Configure the computers with the following settings:

Device	Parameter	Value
Computer A	IP Address	172.16.0.1/24
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.250
	DNS Server 1	172.16.0.250
	DNS Server 2	172.16.0.250
Computer B	IP Address	172.16.0.2
	Subnet Mask	255.255.255.0
	Default-Gateway	172.16.0.250
	DNS Server 1	172.16.0.250
	DNS Server 2	172.16.0.250

5. Do a continuous PING (use –t parameter in Windows, or just the PING command in Linux) from Computer A to the default-gateway.
6. While the PING is continuously showing results, switch off Router 1.
7. Count the downtime until Router 2 receives the flag from the switched-off Router 1.

Scenario 3.12



Connect the network shown in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IPv6 Address	2001::1/64
	Interface S0/0 IPv6 Address	2002::1/64

2. On Router 2, do the basic configuration:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	2003::1/64
	Interface S0/0 IP Address	2002::2/64

3. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IPv6 Address	2001::10
	Prefix Length	64
	Default-Gateway	2001::1
	DNS Server 1	2001::1
	DNS Server 2	2001::1
Computer B	IPv6 Address	2001::11
	Prefix Length	64
	Default-Gateway	2001::1
	DNS Server 1	2001::1
	DNS Server 2	2001::1
Computer C	IPv6 Address	2003::10
	Prefix Length	64
	Default-Gateway	2003::1
	DNS Server 1	2003::1
	DNS Server 2	2003::1
Computer D	IPv6 Address	2003::11
	Prefix Length	64
	Default-Gateway	2001::1
	DNS Server 1	2001::1
	DNS Server 2	2001::1

4. Configure OSPFv3 dynamic routing on Router 1 with the advertised networks 2001::/64 and 2002::/64 in area 0.
5. Configure OSPF Dynamic Routing on Router 2 with the advertised networks 2002::/64 and 2003::/64 in area 0.
6. For testing, PING from Computer A to C and D and show the routing tables in both routers.
7. To see the effect of having a different process number, remove OSPF from Router 2. Reconfigure OSPF on Router 2 with a process number different from Router 1. Check the connectivity and routing tables now.

Chapter 4
Advanced Routing Protocols Configuration

Keywords Cisco · Router · Multiple-area OSPF · Virtual link · IS–IS · Load balancing · BGP · MPLS

4.1 How to Configure Multiple-Area OSPF on a Cisco Router

When would you need this: When you need to set up dynamic routing for a large network and not all your routers are Cisco routers.

Special Requirements: None.

OSPF is one of the most widely used dynamic routing protocols. Cisco's version of OSPF is compatible with non-Cisco routers. If your network is not too large, Sect. 2.3 describes the steps of configuring single-area OSPF.

As defined in the previous section, an area is a logical grouping of routers running OSPF. All routers in the same area share the same topology database.

Multiple-area OSPF is used for large networks to prevent their topology databases from becoming out of the capability of the router.

When you design the areas and assign them IP addresses, remember to assign IP addresses that can be summarized. Make sure that all IP addresses in an area can be summarized into a single address with a different subnet mask. This is very important in reducing the amount of routing information exchanged between routers. (This, basically, is the idea behind creating multiple areas instead of one.)

Area 0, or sometimes referred to as backbone area, will act as the center of the universe for all the other areas. All areas must be connected to Area 0. On the edge of each area, the router connected to another area is called Area Border Router (ABR). If not all areas can be connected to Area 0, there is a solution called virtual links that is discussed later in Sect. 2.5.

Let us move on to the configuration. For the sake of clearness, the ABR router that you plan to put into Area 0 will be called 'Area 0 ABR.' The ABR at the border of other areas will be called 'Area X ABR.'

M.M. Alani, *Guide to Cisco Routers Configuration*,
DOI 10.1007/978-3-319-54630-8_4

4.1.1 Configuration of Area 0 ABR

1. OSPF best route calculations rely solely on bandwidth. Hence, you need to set up the bandwidth of the serial interface involved in the routing process using the following command on the interface:

   ```
   Router(config-if)#bandwidth bandwidth
   ```

   where *bandwidth* is the bandwidth of the connection in kilobits per second.

   Remember that this command does not change the actual bandwidth. It only changes the bandwidth being seen by the routing protocol for the purpose of best path calculation.
2. Instruct the router to activate the OSPF routing process:

   ```
   Router(config)#router ospf process-number
   ```

   where *process-number* is the process number of OSPF. This process number is of local significance. It does not have to be the same on all routers.
3. Instruct the router to advertise the directly connected networks of Area 0

   ```
   Router(config-router)#network network-address wildcard-mask Area 0
   ```

   where

   network-address is the network address of a directly connected network.

   wildcard-mask is the wildcard mask of the network address.
4. Repeat step 3 for every network that is directly connected to the router and is part of Area 0
5. Instruct the router to advertise the directly connected network of Area X

   ```
   Router(config-router)#network network-address wildcard-mask Area X
   ```

   where

   network-address is the network address of the directly connected network that connects Area 0 to Area X.

   wildcard-mask is the wildcard mask of the network address.

   X is the area number.
6. Configure the range of IP addresses for the whole area (similar to a summary address):

   ```
   Router(config-router)#area 0 range network-address subnetmask
   ```

   where

   network-address is the network address that summarizes all the networks in the Area 0.

   subnetmask is the subnet mask for the summarized address.

   This command reduces the size of the topology database, which is important in the backbone area.

4.1.2 Configuration of Area X ABR

1. As mentioned earlier, OSPF best route calculations rely solely on bandwidth. Hence, you need to set up the bandwidth of the serial interface involved in the routing process using the following command on the interface:

   ```
   Router(config-if)#bandwidth bandwidth
   ```

   where bandwidth is the bandwidth of the connection in kilobits per second. Remember that this command does not change the actual bandwidth. It only changes the bandwidth being seen by the routing protocol for the purpose of best path calculation.
2. Instruct the router to activate the OSPF routing process:

   ```
   Router(config)#router ospf process-number
   ```

   where *process-number* is the process number of OSPF. This process number is of local significance. It does not have to be the same on all routers.
3. Instruct the router to advertise the directly connected networks of Area X

   ```
   Router(config-router)#network network-address wildcard-mask Area X
   ```

   where

   network-address is the network address of the directly connected network that is in Area X.

   wildcard-mask is the wildcard mask of the network address.

   X is the area number.
4. Repeat step 3 for every network that is directly connected to the router and is a part of Area X
5. Instruct the router to advertise the directly connected network of Area 0

   ```
   Router(config-router)#network network-address wildcard-mask Area 0
   ```

   where

   network-address is the network address of a directly connected network that connects Area 0 and Area X.

   wildcard-mask is the wildcard mask of the network address.

   0 is the area number.
6. Configure the range of IP addresses for all the networks in Area X (similar to a summary address):

   ```
   Router(config-router)#Area X range network-address subnetmask
   ```

   where

   network-address is the network address that summarizes all the networks in the Area X.

   subnetmask is the subnet mask for the summarized address

4.1.3 Other Commands

You can use the following commands in troubleshooting:

1. To show the OSPF processes information:
   `Router#`**`show ip ospf`**
2. To show the OSPF database of the topology:
   `Router#`**`show ip ospf database`**
3. To show the OSPF operation on the interfaces:
   `Router#`**`show ip ospf interface`**
4. To show the OSPF neighbors table:
   `Router#`**`show ip ospf neighbor`**
5. To debug all the OSPF process events:
   `Router#`**`debug ip ospf events`**
   If you want more OSPF to configure, jump on to the next Sect. 2.5.

4.1.4 More About Multiple-Area OSPF Configuration

This subsection is a continuation of the previous section on configuration of multiple-area OSPF on Cisco routers. Refer to the previous section for the essential configuration of multiple-area OSPF.

1. ASBR
   A router is called Autonomous System Boundary Router (ASBR) when it connects the OSPF area to a different autonomous system. This router should be configured with a summary-address to summarize routes received into OSPF via redistribution:
   `Router(config-router)#`**`summary-address`** *`summary-address subnetmask`*
   where
   summary-address is the summary address for the summarized subnets.
   subnetmask is the subnet mask of the summarized address.
   This command is usually issued at the router connecting the internetwork to the Internet.
2. Stub Areas
   Stub areas in OSPF are areas where only one ABR is there, or where colocated ABRs exist. For this kind of areas, the following configuration can be made to reduce the routing traffic between the ABRs:
   `Router(config-router)#`**`Area`** *`X`* **`stub`**
   where *X* is the area number.
   This command should be issued on both ABRs: the stub area ABR and the Area 0 ABR that is connected to the stub area. If more than one router exists in the

stub area, the previous command needs to be issued on all routers of the stub area.

Usually, if there is only one router in the stub area, instead of defining all directly connected networks, the following command is used:

```
Router(config-router)#network 0.0.0.0 255.255.255.255 Area X
```

where *X* is the area number.

3. Virtual links

By design, all OSPF areas must be connected to Area 0. If there is an area that cannot be directly connected to Area 0, you will have to use a virtual link.

Remember that despite the fact that configuration of the virtual link is simple, many things can go wrong in a virtual link, and it is not a recommended solution.

To implement a virtual link between Area 0 ABR and Area X ABR (where Area X is not directly connected to Area 0), we need to create logical loopback interfaces on both routers and link them together:

On Area 0 ABR:

```
Router(config)#int loopback 0
Router(config-if)#ip address Area0-loopback-address subnetmask
Router(config-if)#exit
Router(config)#router ospf Area0-process-number
Router(config)#Area X virtual-link AreaX-loopback-address
```

where

Area0-loopback-address is an IP address that you assign to the logical interface of Area 0 ABR. This address will be used by the Area X ABR to connect virtually.

subnetmask is the subnet mask that you assign to the logical interface.

Area0-process-number is the process-id of OSPF on the Area 0 ABR.

AreaX-loopback-address is the IP address that you assign to the logical interface of the Area X ABR.

On Area X ABR:

```
Router(config)#int loopback 0
Router(config-if)#ip address AreaX-loopback-address subnetmask
Router(config-if)#exit
Router(config)#router ospf AreaX-process-number
Router(config)#Area X virtual-link Area0-loopback-address
```

where

AreaX-loopback-address is an IP address that you assign to the logical interface of Area X ABR. This address will be used by the Area 0 ABR to connect virtually.

subnetmask is the subnet mask that you assign to the logical interface.
AreaX-process-number is the process-id of OSPF on the Area X ABR.
Area0-loopback-address is the IP address that you assigned to the logical interface of the Area 0 ABR.

4. Some additional show commands to show the virtual links currently configured the ABRs information, respectively:

```
Router#show ip ospf virtual-links
Router#show ip ospf border-routers
```

4.2 How to Configure Integrated IS–IS on a Cisco Router

When would you need this: When you need to set up dynamic routing for a large network and not all your routers are Cisco routers. It is being currently used to support MPLS routing and IPv6 routing.

Special Requirements: None.

IS–IS is an old interior gateway protocol. It is a routing protocol that was aimed to replace TCP/IP, but failed to. Anyway, why are we discussing this old protocol? This is because new interest in it has appeared over the past few years. This interest is caused by the fact that IS–IS protocol is independent, ToS-supporting, and really scalable.

The cornerstone of IS–IS operation is having a properly addressed internetwork for IS–IS. This means that your subnets must be addressed in a summarizable way such that you can express the LANs connected to the router as a summarized address.

Let us move on to the configuration:

1. Create a loopback interface (logical interface) on the router and give it an IP address of your choice. Remember that this IP address will be part of the Network Entity Title (NET) address.

```
Router(config)#int loopback 0
Router(config-if)#ip address loopback-address subnetmask
```

   where
   loopback-address is the IP address you want to assign to the loopback interface.
   subnetmask is the subnet mask that you want to assign to the loopback interface.

2. Write down the NET address that you will assign to the router. There are many ways of creating NET address. We will not discuss them now. We will use the router's loopback address as the system ID as follows:
   AA.BBBB.CCCC.DDDD.EEEE.FF
   where
   AA is the AFI. We will use '49' here as the AFI. This '49' means that we are making up our own NET address.
   BBBB is the area number (ex: 0001, or 0002). Remember that no more than three routers can operate in a single IS–IS area. You should start from 1.

CCCC.DDDD.EEEE is the loopback IP address of the router. Previously defined as 123.456.789.123, here it should be rewritten as 1234.5678.9123. (ex: 172.16.0.254 becomes 172.016.000.254 which becomes 1720.1600.0254).
FF is the SEL. We will use '00' here. This '00' value means that this whole identifier is the NET of the device.
A quick example: For a router with loopback address of 192.168.0.1, and the area of 1, the NET can be written as 49.0001.1921.6800.0001.00

3. Enable the IS–IS routing process on the router:
   ```
   Router(config)#router isis
   ```
4. Configure the NET that you have written down earlier:
   ```
   Router(config-router)#net AA.BBBB.CCCC.DDDD.EEEE.FF
   ```
   where *AA.BBBB.CCCC.DDDD.EEEE.FF* is the NET you have assigned to the router.
5. Configure route summarization only on the routers connected to other areas:
   ```
   Router(config-router)#summary-address summary-address
   subnetmask
   ```
   where
   summary-address is the address summarizing all the networks of the area.
   subnetmask is the subnet mask used for the summarization.
6. Enable IS–IS for IP on the serial interfaces that will be involved in the routing process:
   ```
   Router(config-if)#ip router isis
   ```
7. Repeat this configuration on all the routers that you want to involve in the IS–IS routing process. Remember that each router must have its own NET address and a maximum of three routers in each area.
   What we have introduced here is a very simplified introduction to the configuration of IS–IS. There are many other configuration scenarios that need to be tackled in order to use IS–IS in a large network such as configure IS–IS for NBMA networks, or configuring different IS–IS levels.
   For troubleshooting, you can use the following commands:
   ```
   Router#show clns neighbor
   Router#show clns interface
   Router#show isis database
   Router#show isis database detail
   ```

4.3 How to Configure Load Balancing on a Cisco Router

When would you need this: When you are using a dynamic routing protocol and have more than one path to destination networks.

Special Requirements: None.

The first fact to be set is that all router platforms support load balancing. In a short description, load balancing is the operation in which the router forwards packets in different routes to the same destination. This happens when there is more than one path available for the same destination network.

There are two types of load balancing:

1. Multiple entries to the same destination with equal metrics.
   In this situation, protocols such as RIP, RIPv2, IGRP, EIGRP, and OSPF automatically do the operation and no configuration is needed.
2. Multiple entries to the same destination with different metrics.
   With a complex metric calculation method, like the ones used in IGRP and EIGRP, it is rare to get equal metrics for different paths to the same destination. In this case, configuration is needed.
   You can configure something called *variance*. The variance value determines the percentage that you are willing tolerate in choosing a secondary path. If the value of the variance is chosen to be 1, this means that only the paths with equal best metric will be used. And a value of 1.2, for example, means that the best path as well as the paths with a metric up to 1.2 times best path's metric can be used.
   Here is a numerical example: For a variance of 1.3, if the best path's metric is 1000, paths of metric in the range of 1000–1300 will also be used. Keep in mind that we are talking about multiple paths to the same destination.
   One more important note is that we are talking about paths derived from the same routing protocol, i.e., paths with the same administrative distance.
   The configuration of unequal path load balancing for IGRP and EIGRP is done with a single command:
   ```
   Router(config-router)#variance X
   ```
   where *X* represents the value of the variance that you want to use.

4.4 Per-Packet and Per-Destination Load Balancing

There are two types of load balancing: Per-Packet and Per-Destination. In Per-Packet load balancing, packets going to the same destination are sent over different paths. This way you will guarantee that all paths to the destination network are being used. But using this method causes increased load on the routers' resources and low-end routers may crash. Also, the packets may arrive out of order because of different network latencies in different paths.

Using the Per-Destination load balancing, packets going to one destination pass through one path. This way you will lower the load on the router. But the different paths will not be utilized to the best.

To activate Per-Destination load balancing issue the following command on the interface that you want to use this method of load balancing,

```
Router(config-if)#ip route-cache
```

And to activate Per-Packet load balancing use,

```
Router(config-if)#no ip route-cache
```

Newer switching schemes such as Cisco Express Forwarding (CEF) allow you to do Per-Packet and Per-Destination load balancing more quickly. However, this method requires some extra resources to deal with maintaining CEF entries and adjacencies.

4.5 How to Configure BGP on a Cisco Router

When would you need this: When you need to exchange routing updates between different autonomous systems.

Special Requirements: None.

BGP is categorized as an Exterior Gateway Protocol (EGP) that helps in exchanging routing information between different autonomous systems.

1. Enable BGP process for a specific autonomous system number:
   ```
   Router1(config)#router bgp autonomous-system-number1
   ```
   where *autonomous-system-number1* is the number of the autonomous system of the first router to be included in the BGP process.
2. Identify the networks that you want to consider local in the autonomous system identified in step 1.
   ```
   Router1(config-router)#network network-id mask subnetmask
   ```
   where *network-id* is the network address of the local network and *subnetmask* is its subnet mask. The *subnetmask* part is optional.

   Repeat this step for all the networks you want to consider local in the given *autonomous-system-number1*.
3. Identify the neighbors. A neighbor is a router that is running BGP operation on another autonomous system.
   ```
   Router1(config-router)#neighbor ip-address2 remote-as neighbor-autonomous-system-number2
   ```
   where

   neighbor-ip-address2 is the IP address of the second (or neighbor) router.

   autonomous-system-number2 is the autonomous system number of the second router.
4. On Router 2, we set up the autonomous system number:
   ```
   Router2(config)#router bgp autonomous-system-number2
   ```
   where *autonomous-system-number2* is the number of the autonomous system of the second router to be included in the BGP process.
5. Identify the networks that you want to consider local in the autonomous system identified in step 4.
   ```
   Router2(config-router)#network network-id mask subnetmask
   ```
   where*network-id* is the network address of the local network and *subnetmask* is its subnet mask. The *subnetmask* part is optional.

Repeat this step for all the networks you want to consider local in the given *autonomous-system-number2*.

6. Identify the neighbors.

   ```
   Router2(config-router)#neighbor ip-address1 remote-as
   neighbor-autonomous-system-number1
   ```

   where

   neighbor-ip-address1 is the IP address of the first (or neighbor) router.

   autonomous-system-number1 is the autonomous system number of the first router.

7. Troubleshooting can be done using the following commands:

   ```
   Router#show ip bgp
   Router#show ip bgp neighbors
   Router#show ip bgp neighbors neighbor-ip-address
   ```

4.5.1 About BGP and IGP Synchronization

By default, BGP synchronizes with IGPs. The idea behind synchronization is that BGP does not forward updates that are not forwarded by the IGP first. In certain scenarios this can be useful. For example, if your BGP process is passing updates between two different autonomous systems other than your autonomous system, synchronization is necessary to avoid any routing misinformation.

However, if the autonomous system on your router is not being used as a path to pass routing information between two other autonomous systems, you can switch off the synchronization using the following command:

```
Router(config-router)#no synchronization
```

The purpose of turning off the synchronization is to reduce the processing load on the router.

4.6 How to Configure BGP for IPv6 on a Cisco Router

When would you need this: When you need to exchange routing updates between different autonomous systems with IPv6.

Special Requirements: None.

BGP is categorized as an Exterior Gateway Protocol (EGP) that helps in exchanging routing information between different autonomous systems.

1. Enable BGP process for a specific autonomous system number:

   ```
   Router1(config)#router bgp autonomous-system-number1
   ```

   where *autonomous-system-number1* is the number of the autonomous system on Router 1 to be included in the BGP process.

2. Identify the networks that you want to consider local in the autonomous system identified in step 1.

   ```
   Router1(config-router)#network ipv6-network-id/prefix-length
   ```

   where *ipv6-network-id* is the network address of the local network and *prefix-length* is prefix length of the IPv6 address.

   Repeat this step for all the networks you want to consider local in the given *autonomous-system-number*.

3. Identify the router ID for the BGP process:

   ```
   Router1(config-router)#router-id router-id
   ```

   where the *router-id* is the router ID to be used in the BGP process. Despite the fact that we are configuring IPv6 BGP, the *router-id* is in IPv4 address format.

4. We identify the neighbors. A neighbor is a router that is running BGP operation on another autonomous system.

   ```
   Router1(config-router)#neighbor ipv6-address2 remote-as autonomous-system-number2
   ```

   where

   ipv6-address2 is the IPv6 address of the second router.

   neighbor-autonomous-system-number is the autonomous system number of the neighbor router.

5. Enable BGP process for a specific autonomous system number on Router 2:

   ```
   Router2(config)#router bgp autonomous-system-number2
   ```

   where *autonomous-system-number2* is the number of the autonomous system on Router 2 to be included in the BGP process.

6. Identify the networks that you want to consider local in the autonomous system identified in step 5.

   ```
   Router2(config-router)#network ipv6-network-id/prefix-length
   ```

   where *ipv6-network-id* is the network address of the local network and *prefix-length* is prefix length of the IPv6 address.

   Repeat this step for all the networks you want to consider local in the given *autonomous-system-number*.

7. Identify the router ID for the BGP process on Router 2:

   ```
   Router2(config-router)#router-id router-id
   ```

   where the *router-id* is the router ID to be used in the BGP process. Despite the fact that we are configuring IPv6 BGP, the *router-id* is in IPv4 address format.

8. We identify the neighbors. A neighbor is a router that is running BGP operation on another autonomous system.

   ```
   Router2(config-router)#neighbor ipv6-address1 remote-as autonomous-system-number1
   ```

   where

   ipv6-address1 is the IPv6 address of the first router.

   neighbor-autonomous-system-number is the autonomous system number of the first router.

9. Troubleshooting can be done using the following commands:
   ```
   Router#show ip bgp
   Router#show ip bgp neighbors
   Router#show ip bgp neighbors neighbor-ip-address
   ```

4.7 How to Configure MPLS on a Cisco Router

When would you need this: When you have a large network and you need to keep the performance high. MPLS also provides very scalable support of VPNs.

Special Requirements: None.

MPLS configuration can come in different topologies, and it should be done differently depending on the job of the router in that particular network.

4.7.1 Configuring the Router for MPLS Switching

To activate MPLS switching on a Cisco router, you'll need to enable Cisco Express Forwarding.

```
Router(config)#ip cef distributed
```

4.7.2 Configuring the Router for MPLS Forwarding

1. Enable MPLS forwarding on the interface level:
   ```
   Router(config-if)#mpls ip
   ```
   This would enable MPLS forwarding for IPv4 packet.
2. Configure MPLS static prefix label bindings:
   ```
   Router(config)#mpls label range min-label max-label [static min-static-label max-static-label]
   Router(config)#mpls static binding ipv4 network mask static-label
   ```
   where

   min-label is the start of the label range.

   max-label is the end of the label range.

   min-static-label is the start of the static label range.

   max-static-label is the end of the static label range.

   network and *mask* are the prefix of the network that you want to bind and its subnet mask.

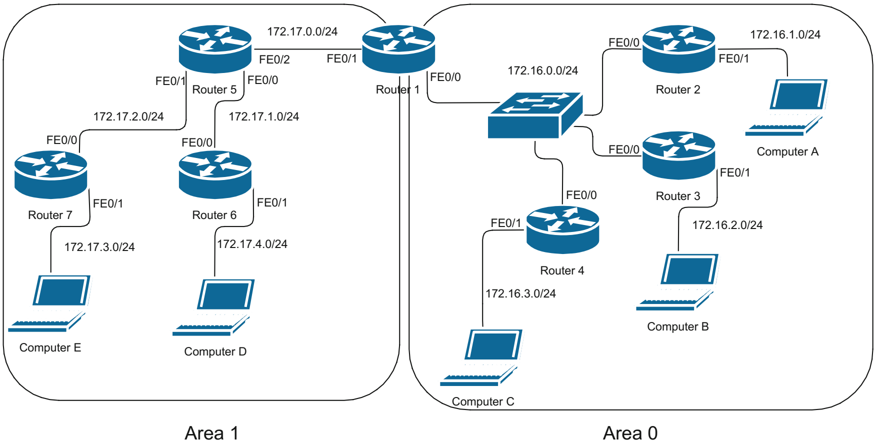
static-label is the specific static label (within the range *min-static-label* to *max-static-label*) that will be bound to the network specified.

3. Verify your configuration using the following commands:

```
Router#show mpls static binding ipv4
Router#show mpls forwarding-table
Router#show mpls label range
```

4.8 Training Scenarios

Scenario 4.1



Connect the network in the diagram above and configure the following settings:

1. Basic configuration for routers:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.1/24
	Interface FE0/1 IP Address	172.17.0.1/24

(continued)

(continued)

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.2/24
	Interface FE0/1 IP Address	172.17.1.1/24
Router 3	Hostname	Router3
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.3/24
	Interface FE0/1 IP Address	172.16.2.1/24
Router 4	Hostname	Router4
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.4/24
	Interface FE0/1 IP Address	172.16.3.1/24
Router 5	Hostname	Router5
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.17.1.1/24
	Interface FE0/1 IP Address	172.17.2.1/24
	Interface FE0/2 IP Address	172.17.0.2/24
Router 6	Hostname	Router6
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.17.1.2/24
	Interface FE0/1 IP Address	172.17.4.1/24
Router 7	Hostname	Router7
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.17.2.2/24
	Interface FE0/1 IP Address	172.17.3.1/24

2. Computers' settings:

Device	Parameter	Value
Computer A	IP Address	172.16.1.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.1.1
	DNS Server 1	172.16.1.1
	DNS Server 2	172.16.1.1
Computer B	IP Address	172.16.2.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.2.1
	DNS Server 1	172.16.2.1
	DNS Server 2	172.16.2.1
Computer C	IP Address	172.16.3.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.3.1
	DNS Server 1	172.16.3.1
	DNS Server 2	172.16.3.1
Computer D	IP Address	172.17.4.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.17.4.1
	DNS Server 1	172.17.4.1
	DNS Server 2	172.17.4.1
Computer E	IP Address	172.17.3.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.17.3.1
	DNS Server 1	172.17.3.1
	DNS Server 2	172.17.3.1

3. OSPF settings:

Router	Network	Area
1	172.16.0.0	0
	172.17.0.0	1
2	172.16.0.0	0
	172.16.1.0	0
3	172.16.0.0	0
	172.16.2.0	0
4	172.16.0.0	0
	172.16.3.0	0
5	172.17.0.0	1
	172.17.1.0	1
	172.17.2.0	1

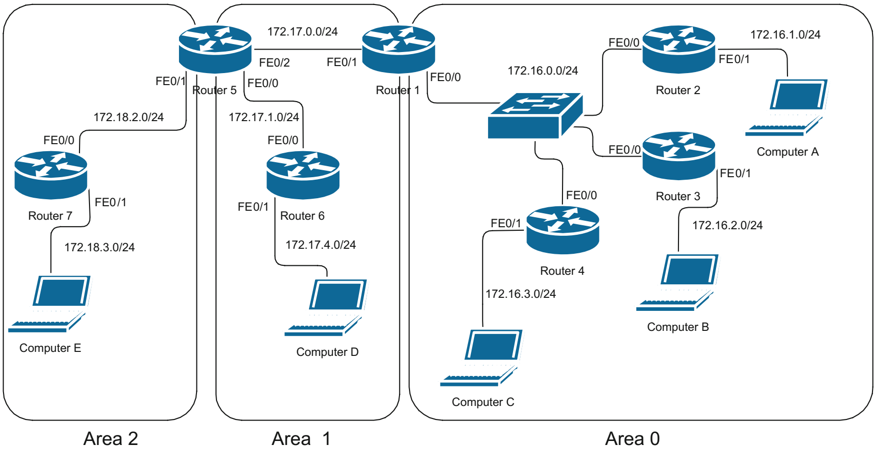
(continued)

(continued)

Router	Network	Area
6	172.17.1.0	1
	172.17.4.0	1
7	172.17.2.0	1
	172.17.3.0	1

4. For internal testing, PING from Computer A to B and C, and from Computer D to Computer E.
5. For inter-area OSPF configuration, instruct the Router 1 to advertise a summarized address for each area.
   Area 0: 172.16.0.0/16
   Area 1: 172.17.0.0/16
6. After convergence, try PING from Computer A to Computer E.

Scenario 4.2



Connect the network in the diagram above and configure the following settings:

7. Basic configuration for routers:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.1/24
	Interface FE0/1 IP Address	172.17.0.1/24

(continued)

(continued)

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.2/24
	Interface FE0/1 IP Address	172.17.1.1/24
Router 3	Hostname	Router3
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.3/24
	Interface FE0/1 IP Address	172.16.2.1/24
Router 4	Hostname	Router4
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.4/24
	Interface FE0/1 IP Address	172.16.3.1/24
Router 5	Hostname	Router5
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.17.1.1/24
	Interface FE0/1 IP Address	172.18.2.1/24
	Interface FE0/2 IP Address	172.17.0.2/24
Router 6	Hostname	Router6
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.17.1.2/24
	Interface FE0/1 IP Address	172.17.4.1/24
Router 7	Hostname	Router7
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.18.2.2/24
	Interface FE0/1 IP Address	172.18.3.1/24

8. Computers' settings:

Device	Parameter	Value
Computer A	IP Address	172.16.1.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.1.1
	DNS Server 1	172.16.1.1
	DNS Server 2	172.16.1.1
Computer B	IP Address	172.16.2.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.2.1
	DNS Server 1	172.16.2.1
	DNS Server 2	172.16.2.1
Computer C	IP Address	172.16.3.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.3.1
	DNS Server 1	172.16.3.1
	DNS Server 2	172.16.3.1
Computer D	IP Address	172.17.4.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.17.4.1
	DNS Server 1	172.17.4.1
	DNS Server 2	172.17.4.1
Computer E	IP Address	172.18.3.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.18.3.1
	DNS Server 1	172.18.3.1
	DNS Server 2	172.18.3.1

9. OSPF settings:

Router	Network	Area
1	172.16.0.0	0
	172.17.0.0	1
2	172.16.0.0	0
	172.16.1.0	0
3	172.16.0.0	0
	172.16.2.0	0
4	172.16.0.0	0
	172.16.3.0	0
5	172.17.0.0	1
	172.17.1.0	1
	172.18.2.0	2

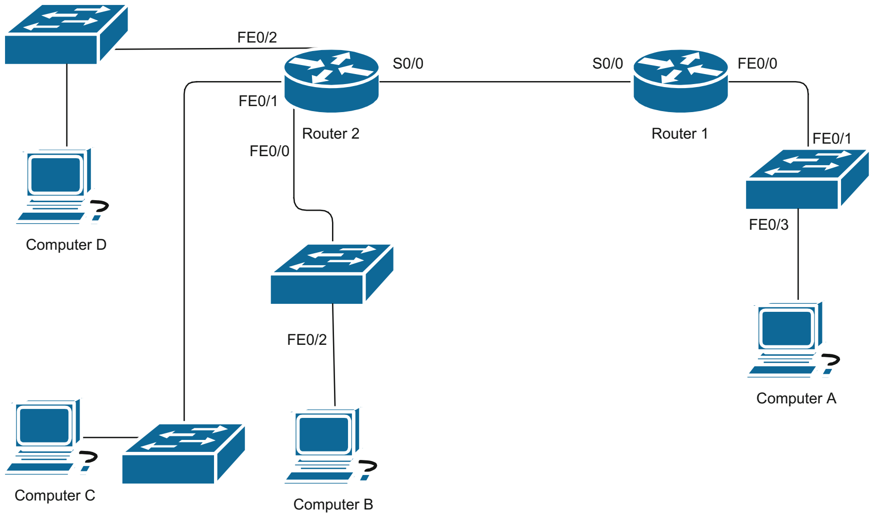
(continued)

(continued)

Router	Network	Area
6	172.17.1.0	1
	172.17.4.0	1
7	172.18.2.0	2
	172.18.3.0	2

10. For internal testing, PING from Computer A to B and C, and from Computer D to Computer E.
11. For inter-area OSPF configuration, instruct the routers to advertise a summarized address for each area.
    Area 0: 172.16.0.0/16
    Area 1: 172.17.0.0/16
    Area 2: 172.18.0.0/16
12. Since Areas 2 and 0 are not directly connected, set up a virtual link between them.
13. After convergence, try PING from Computer A to Computer E.

Scenario 4.3



Connect the network in the figure above and configure the following settings:

1. On Router 1 do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.19.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30

2. On Router 2 do the basic configuration:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	192.16.1.254/24
	Interface FE0/1 IP Address	192.16.2.254/24
	Interface FE0/2 IP Address	192.16.3.254/24
	Interface S0/0 IP Address	10.0.0.2/30

3. On the computers change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.19.0.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.19.0.254
	DNS Server 1	192.19.0.254
	DNS Server 2	192.19.0.254
Computer B	IP Address	192.16.1.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254
Computer C	IP Address	192.16.2.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.2.254
	DNS Server 1	192.16.2.254
	DNS Server 2	192.16.2.254

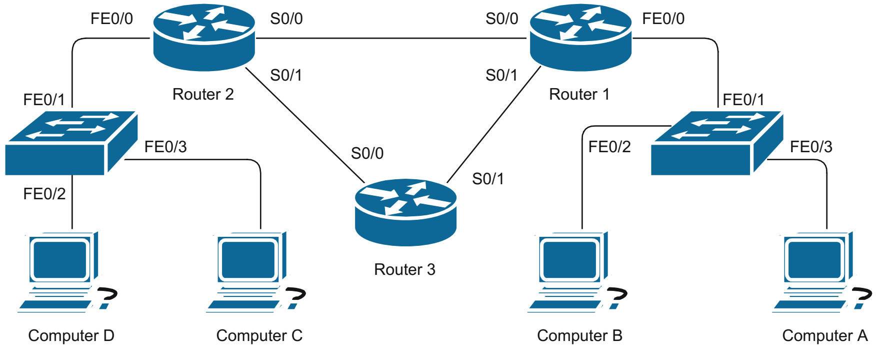
(continued)

(continued)

Device	Parameter	Value
Computer D	IP Address	192.16.3.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.3.254
	DNS Server 1	192.16.3.254
	DNS Server 2	192.16.3.254

4. Create a loopback interface on Router 2 and give it the IP address 192.16.16.1/24.
5. Create a loopback interface on Router 1 and give it the IP address 192.19.17.1/24.
6. Using the loopback addresses configured in the previous steps, configure IS–IS in both routers using the NET creation rules indicated in Sect. 4.2.
7. The summary address for Router 1 should be 192.19.0.0/16 and for Router 2, 192.16.0.0/16.
8. For testing, PING from Computer A to C and D and show the routing tables in both routers.

Scenario 4.4



Connect the network shown in the figure above and configure the following settings:

1. On Router 1 do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.16.0.254/24
	Interface S0/0 IP Address	192.0.0.1/30
	Interface S0/1 IP Address	192.1.0.1/30

2. On Router 2 do the basic configuration:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	192.16.1.254/24
	Interface S0/0 IP Address	192.0.0.2/30
	Interface S0/1 IP Address	192.2.0.1/30

3. On Router 3 do the basic configuration:

Device	Parameter	Value
Router 3	Hostname	Router3
	Console Password	Cisco3Console
	Secret Password	Cisco3
	VTY Password	Cisco3VTY
	Interface S0/0 IP Address	192.2.0.2/30
	Interface S0/1 IP Address	192.1.0.2/30

4. On the computers change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.16.0.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.0.254
	DNS Server 1	192.16.0.254
	DNS Server 2	192.16.0.254

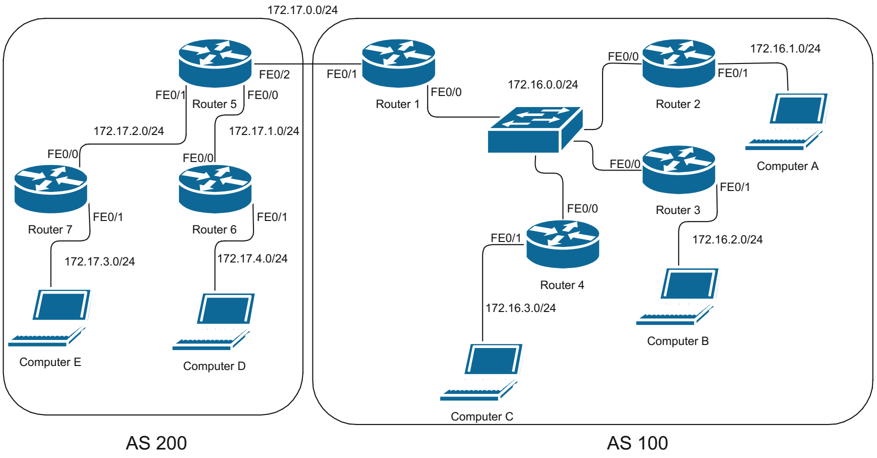
(continued)

(continued)

Device	Parameter	Value
Computer B	IP Address	192.16.0.2
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.0.254
	DNS Server 1	192.16.0.254
	DNS Server 2	192.16.0.254
Computer C	IP Address	192.16.1.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254
Computer D	IP Address	192.16.1.2
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254

5. Configure EIGRP dynamic routing on Router 1 with the advertised networks 192.0.0.0, 192.16.0.0, and 192.1.0.0 in autonomous system number 10.
6. Configure EIGRP dynamic routing on Router 2 with the advertised networks 192.0.0.0, 192.2.0.0, and 192.16.1.0 in autonomous system number 10.
7. Configure EIGRP dynamic routing on Router 3 with the advertised networks 192.1.0.0 and 192.1.0.0
8. For testing, PING from Computer A to C and D and show the routing tables in both routers.
9. Configure load balancing on Routers 1 and 2 and try different variance.
10. Try changing the 'bandwidth' value on the different serial interfaces on the three routers and see how that affects load balancing.

Scenario 4.5



Connect the network in the diagram above and configure the following settings:

1. Basic configuration for routers:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.1/24
	Interface FE0/1 IP Address	172.17.0.1/24
Router 2	Hostname	Router2
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.2/24
	Interface FE0/1 IP Address	172.17.1.1/24
Router 3	Hostname	Router3
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.3/24
	Interface FE0/1 IP Address	172.16.2.1/24
Router 4	Hostname	Router4
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.4/24
	Interface FE0/1 IP Address	172.16.3.1/24
Router 5	Hostname	Router5
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.17.1.1/24
	Interface FE0/1 IP Address	172.17.2.1/24
	Interface FE0/2 IP Address	172.17.0.2/24
Router 6	Hostname	Router6
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.17.1.2/24
	Interface FE0/1 IP Address	172.17.4.1/24

(continued)

(continued)

Device	Parameter	Value
Router 7	Hostname	Router7
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.17.2.2/24
	Interface FE0/1 IP Address	172.17.3.1/24

2. Computers' settings:

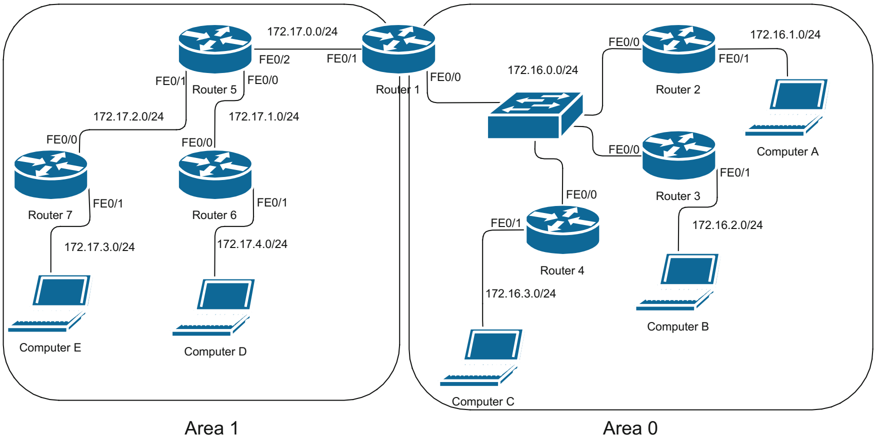
Device	Parameter	Value
Computer A	IP Address	172.16.1.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.1.1
	DNS Server 1	172.16.1.1
	DNS Server 2	172.16.1.1
Computer B	IP Address	172.16.2.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.2.1
	DNS Server 1	172.16.2.1
	DNS Server 2	172.16.2.1
Computer C	IP Address	172.16.3.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.3.1
	DNS Server 1	172.16.3.1
	DNS Server 2	172.16.3.1
Computer D	IP Address	172.17.4.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.17.4.1
	DNS Server 1	172.17.4.1
	DNS Server 2	172.17.4.1
Computer E	IP Address	172.17.3.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.17.3.1
	DNS Server 1	172.17.3.1
	DNS Server 2	172.17.3.1

3. EIGRP settings:

Router	Network	Autonomous system
1	172.16.0.0	100
	172.17.0.0	100
2	172.16.0.0	100
	172.16.1.0	100
3	172.16.0.0	100
	172.16.2.0	100
4	172.16.0.0	100
	172.16.3.0	100
5	172.17.0.0	200
	172.17.1.0	200
	172.17.2.0	200
6	172.17.1.0	200
	172.17.4.0	200
7	172.17.2.0	200
	172.17.3.0	200

4. For internal testing, PING from Computer A to B and C, and from Computer D to Computer E.
5. Configure BGP protocols on Routers 1 and 5 such that there is route redistribution between AS100 and AS200.
6. After convergence, try PING from Computer A to Computer E.

Scenario 4.6



Connect the network in the diagram above and configure the following settings:

1. Basic configuration for routers:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	2001::1/64
	Interface FE0/1 IP Address	3001::1/64
Router 2	Hostname	Router2
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	2001::2/64
	Interface FE0/1 IP Address	2002::1/64
Router 3	Hostname	Router3
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	2001::3/64
	Interface FE0/1 IP Address	2003::1/64
Router 4	Hostname	Router4
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	2001::4/64
	Interface FE0/1 IP Address	2004::1/64
Router 5	Hostname	Router5
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	3002::1/64
	Interface FE0/1 IP Address	3003::1/64
	Interface FE0/2 IP Address	3001::2/64
Router 6	Hostname	Router6
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	3002::2/64
	Interface FE0/1 IP Address	3004::1/64

(continued)

(continued)

Device	Parameter	Value
Router 7	Hostname	Router7
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	3003::2/64
	Interface FE0/1 IP Address	3005::1/64

2. Computers' settings:

Device	Parameter	Value
Computer A	IP Address	2002::10
	Prefix Length	64
	Default Gateway	2002::1
	DNS Server 1	2002::1
	DNS Server 2	2002::1
Computer B	IP Address	2003::10
	Prefix Length	64
	Default Gateway	2003::1
	DNS Server 1	2003::1
	DNS Server 2	2003::1
Computer C	IP Address	2004::10
	Prefix Length	64
	Default Gateway	2004::1
	DNS Server 1	2004::1
	DNS Server 2	2004::1
Computer D	IP Address	3004::10
	Prefix Length	64
	Default Gateway	3004::1
	DNS Server 1	3004::1
	DNS Server 2	3004::1
Computer E	IP Address	3005::10
	Prefix Length	64
	Default Gateway	3005::1
	DNS Server 1	3005::1
	DNS Server 2	3005::1

3. OSPF settings:

Router	Network	Area
1	2001::	0
	3001::	1
2	2001::	0
	2002::	0
3	2001::	0
	2003::	0
4	2001::	0
	2004::	0
5	3001::	1
	3002::	1
	3003::	1
6	3002::	1
	3004::	1
7	3003::	1
	3005::	1

4. For internal testing, PING from Computer A to B and C, and from Computer D to Computer E.
5. For inter-area OSPF configuration, instruct the Router 1 to advertise a summarized address for each area.
   Area 0: 2000::/12
   Area 1: 3000::/12
6. After convergence, try PING from Computer A to Computer E.

Chapter 5
WAN Technologies

Keywords Cisco · Router · ADSL · Frame-relay · ATM · PPP · CHAP · PAP · HDLC · ISDN · BRI · PRI

5.1 How to Configure ADSL on a Cisco Router

When would you need this: When you need to configure your router to operate on ADSL line provided by an ISP.

Special Requirements: ADSL WAN interface on the router.

Before starting, make a list of the information you will need from your service provider. This list includes the following:

Your account's username and password, MTU size (usually 1492), and PVC value (usually 0/35 or 8/35). You will also need to know whether the IP address assigned to you by the ISP is a static public IP address, or your address is going to be dynamically assigned.

The configuration described here moves the PPPoE operation to the router itself, so you will not have to set up PPPoE on the PC in order to dial in the Internet connection. The router will do that for you.

Now, let us jump into the configuration:

1. Enable the PPPoE operation in the global configuration as follows:
   ```
   Router(config)#vpdn enable
   Router(config)#no vpdn-logging
   Router(config)#vpdn-group pppoe
   Router(config-vpdn)#request dialin
   Router(config-vpdn-req-in)#protocol pppoe
   ```
2. Set up the FastEthernet interface that will be connected to the local network, or any other type of interface you want to have:
   ```
   Router(config)#int fa interface-number
   Router(config-if)#ip address local-ip-address subnetmask
   Router(config-if)#ip tcp adjust-mss 1452
   ```
   where

M.M. Alani, *Guide to Cisco Routers Configuration*,
DOI 10.1007/978-3-319-54630-8_5

interface-number is the number of your fast-ethernet interface.
local-ip-address is the local IP address of your fast-ethernet interface.
subnetmask is the subnet mask of the local interface.
If ‘`ip tcp adjust-mss`’ does not work, try out ‘`ip adjust-mss`’ instead. If both do not work, you will need to upgrade the IOS of the router.

3. Set up the physical ADSL interface:
   ```
   Router(config)#int atm 0
   Router(config-if)#no ip address
   Router(config-if)#pvc pvc-number
   Router(config-if-atm-vc)#pppoe-client dial-pool-number
   1
   Router(config-if-atm-vc)#no shut
   ```
   where
   pvc-number is the values of PVC you took from the ISP. (Usually 0/35 or 8/35).
4. Set up the dialer interface:
   ```
   Router(config)#int dialer 1
   ```
   If the IP address the ISP is giving you is a static public IP address, use this command:
   ```
   Router(config-if)#ip address isp-given-address
   subnetmask
   ```
   where
   isp-given-address and *subnetmask* are the static public IP address and subnet mask given to you by the ISP.
   If the IP address is being assigned by the ISP dynamically, use this command instead:
   ```
   Router(config-if)#ip address negotiated
   ```
   After setting the IP address, continue the rest of the dialer configuration:
   ```
   Router(config-if)#mtu mtu-size
   Router(config-if)#no ip directed-broadcast
   Router(config-if)#encapsulation ppp
   Router(config-if)#dialer pool 1
   Router(config-if)#ppp authentication chap pap callin
   Router(config-if)#ppp chap hostname username
   Router(config-if)#ppp chap password password
   Router(config-if)#ppp pap sent-username username password
   password
   ```
   where
   mtu-size is the size of the MTU given to you by the ISP (usually 1492).
   username is the username of your account given by the ISP.
   password is the password of your account given by the ISP.
5. The last step is to configure a default route to forward the traffic to the Internet through the dialer interface:
   ```
   Router(config)#ip classless
   Router(config)#ip route 0.0.0.0 0.0.0.0 interface dialer 1
   ```

It is a common practice to use NAT with the ADSL connection to facilitate the use of the ADSL connection by more than one host. Use the procedure outlined in Sect. 3.3 to configure NAT. In the aforementioned procedure, use the dialer 1 interface instead of the outside interface. For example, the `ip nat outside` command must be given in the dialer 1 interface with the group of commands displayed here in step 4. Another example is in the case of the use of a single public IP address, the NAT command must become

```
ip nat inside source list Access-list-number interface dialer 1 overload
```

5.2 How to Configure PPP on a Cisco Router

When would you need this: When you are creating a WAN link. This procedure might also be required when the other end of a WAN link is not a Cisco router. Point-to-Point Protocol can be used in synchronous, asynchronous, HSSI, and ISDN links.

Special Requirements: None.

1. Get to the interface configuration mode of the router's serial interface and issue the following command,
   ```
   Router(config-if)#encapsulation ppp
   ```
2. If you want to configure authentication (which is almost always the case), go through the following steps:

   a. Choose the authentication type: Password Authentication Protocol (PAP) or Challenge Handshake Authentication Protocol (CHAP)
   ```
   Router(config-if)#ppp authentication authentication-type
   ```
   where *authentication type* is the authentication type which can be: PAP, CHAP, PAP CHAP, or CHAP PAP. The last two choices are to use the second authentication type when the first one fails.
   CHAP is strongly recommended over PAP for two reasons. First, PAP sends the username and password in plaintext, while CHAP sends hashed challenges only. Second is that CHAP does an operation similar to periodic re-authentication in the middle of the communication session, such that it provides more security than PAP.

   b. Set a username and a password that the remote router would use to connect to your local router. You can define many username/password pairs for many PPP connections to the same router.
   ```
   Router(config)#username remote-username password remote-password
   ```
   where *remote-username* is username sent from the remote router, and *remote-password* is its password. If the remote router was not configured with a username to send, it will send its hostname instead.

Issue this command once for each PPP connection. For example, if you are connecting RouterA to RouterB and RouterC, on RouterA issue this command once for each remote router.

c. Now, you can set the username and password that your local router would send to access the remote router. For PAP authentication, you can specify the username and password that the local router will send to the remote router for authentication using the following command,

   ```
   Router(config-if)#ppp pap sent-username sent-username
   password sent-password
   ```

   For CHAP, two commands are used,

   ```
   Router(config-if)#ppp chap hostname sent-username
   Router(config-if)#ppp chap password sent-password
   ```

   The usernames and passwords are case sensitive, so be careful when writing them. This way, you will have to write the username and password of the remote router in your local router and write the username and password of your local router into your remote using the '`username`' command.

   If you do not set the username and password that will be sent from the local router to the remote router for authentication, the router will use its hostname and secret password instead.

3. You can monitor the quality of the serial link that is using PPP with the following command,

   ```
   Router(config-if)#ppp quality percentage
   ```

   where *percentage* is the minimum accepted link quality. If the link quality drops below the *percentage*, the link will be shutdown and considered bad.
4. If the available bandwidth is small, you might consider compressing the data being transmitted using the following command,

   ```
   Router(config-if)#ppp compress compression-type
   ```

   where *compression type* is the compression type which can be `predictor` or `stacker`.
5. To troubleshoot PPP, you can use the following commands,

   ```
   Router#debug ppp negotioations
   Router#debug ppp packets
   Router#debug ppp errors
   Router#debug ppp authentication
   ```

5.3 How to Configure HDLC on a Cisco Router

When would you need this: When you connect two Cisco routers through a WAN connection, or in a back-to-back router setup.

Special Requirements: Both routers need to be Cisco routers.
Cisco HDLC is not compatible with other vendors' HDLC, so you need to make sure that routers on both ends of the communication are Cisco routers.

There is no long procedure to do this. The configuration is actually a single command:

```
Router(config-if)#encapsulation hdlc
```

Usually, the default encapsulation on the out-of-the-box Cisco routers is HDLC.

For troubleshooting, you can use the following commands to check that it has been configured properly:

```
Router#show interface serial interface-number
```

where *interface number* is the number of the interface to check.

You can also check the status and IP address of the interfaces using the following command:

```
Router#show ip interface brief
```

5.4 How to Configure BRI ISDN in a Cisco Router

When would you need this: When you have ISDN WAN link and you want the router to use it.

Special Requirements: The router should have BRI interface(s).

There are two ways to configure ISDN in a Cisco router. The first one is to set up the ISDN connection to be always on. This method will be costly because most ISDN service providers charge not only by monthly subscription, but also by the amount of data that you transfer. Having the connection always alive will cause extra expenses, because all kinds of traffic will pass through the ISDN link.

The second method is Dial on Demand Routing (DDR). DDR employs a mechanism that filters the traffic into interesting (worth connecting for) and non-interesting (not worth it). Using the DDR, the call scenario will be that the router does not set up the connection until interesting traffic needs to be routed to the other side. Once the connection is set up, all kinds of traffic (interesting and non-interesting) will pass unless you filter the passing traffic with an Access-list. Then, the router sets a down counter (idle-timer), and if no interesting traffic comes in and the timer goes to zero, the connection is terminated. If interesting traffic comes in before the idle timer is finished, the traffic is passed and the idle timer is reset. What made this function possible is the very small call setup time in ISDN.

If you are connecting two nodes using ISDN, keep reading. However, if you are connecting more than two nodes, you will need to go to Sect. 5.5 about configuring DDR Dialer Profiles.

To configure DDR on the router's BRI interface, perform the following steps:

1. The first thing to do is to set up routing. Static routing is usually preferred with DDR. Setting dynamic routing protocol will cause the link to be on all (or most) of the time. Thus, static routing is a better solution. You can set up dynamic routing and tune it a bit for the DDR. This tuning might include changing the timers of routing updates.

   The following is an example of static routing;

   ```
   Router(config)#ip route 192.168.1.0 255.255.255.0 192.168.2.1
   Router(config)#ip route 192.168.2.1 255.255.255.255 bri0
   ```

   Or, a default route,

   ```
   Router(config)#ip route 0.0.0.0 0.0.0.0 bri0
   ```

   Keep in mind that you will need to set up routing on both ends of the WAN link.

2. Specify the type of the ISDN switch. This piece of information should be provided to you by the ISDN service provider. You can issue this command,

   ```
   Router(config)#isdn switch-type switch-type
   ```

   where *switch-type* is the ISDN switch type.

   Issuing this command in the global configuration mode will cause all the router's ISDN interfaces to be set to use this type of switch. You can set different types of switches for different interfaces if you issue the same command in the interface configuration mode like the following example:

   ```
   Router(config)#int bri0
   Router(config-if)#isdn switch-type switch1-type
   Router(config-if)#int bri1
   Router(config-if)#isdn switch-type switch2-type
   ```

   After defining the switch type, identify the SPIDs in the BRI interface configuration mode,

   ```
   Router(config-if)#isdn spid1 first-spid ldn1
   Router(config-if)#isdn spid2 second-spid ldn2
   ```

   The SPID and LDN numbers used here should be provided to you by the ISDN service provider.

   Most providers in Europe do not use SPIDs in their ISDN networks. So, unless you are supplied with SPID numbers from the provider, just neglect all the commands of setting SPIDs in this procedure.

3. Specify interesting traffic to the router is next. This traffic is defined as the traffic permitted by a command named '`dialer-list`' which is similar to '`Access-list`'.

   This can be done in two ways; the first is to use the following command,

   ```
   Router(config)#dialer-list list-number protocol protocol-type permit
   ```

   where *list-number* is the dialer list number and *protocol-type* is the protocol you want to permit.

   In addition, you can use '`deny`' instead of the '`permit`' part to prevent a certain protocol from becoming categorized as interesting. However, this is not a very powerful way of defining the interesting traffic.

   The second way is more recommended. The second way is to do a complete Access-list permitting the traffic that you want the router to consider interesting, and then, attach it to a dialer list.

   Create the Access-list the exact same way you create any other Access-list, but do not apply it to an interface. Instead, associate it with a dialer list. All the traffic permitted by this Access-list will be considered interesting. An example is the following:

```
Router(config)#Access-list 110 deny tcp any any telnet
Router(config)#Access-list 110 deny icmp any any
Router(config)#Access-list 110 permit ip any any
```

And the step that will associate the Access-list to the dialer list is:

```
Router(config)#dialer-list 5 protocol ip list 110
```

where 5 is the dialer list number and 110 is the Access-list number. These two numbers do not need to be the same.

Keep in mind that these dialer list and Access-list do not filter the traffic outgoing through the ISDN interface, it just chooses which traffic is entitled to initiate a call. Once the call is set up, all traffic willing to pass through the ISDN link will pass. If you want to filter the traffic that is passing through the ISDN interface, create another Access-list for that with the filters that you find appropriate and apply it to the BRI interface as you do to any other type of interface.

4. Set up the encapsulation protocol, PPP. Using PAP authentication does not provide that much of security, so it is suggested that you use CHAP for authentication.

   The first thing to do to configure PPP to use CHAP is to set a username and a password.

   ```
   Router(config)#username username password password
   ```

   where *username* is the username and *password* is the password. The username should be the hostname of the other end and the password is the secret password of the other end. If you like to use different usernames and passwords, please refer to the PPP configuration procedure in Sect. 5.2.

   Then, move into the interface configuration mode of the ISDN interface,

   ```
   Router(config)#int bri interface-number
   ```

   Now, set an IP address and a subnet mask for the interface,

   ```
   Router(config-if)#ip address ip-address subnetmask
   ```

   Set the encapsulation and authentication types;

   ```
   Router(config-if)#encapsulation ppp
   Router(config-if)#ppp authentication chap
   ```

5. Apply the dialer list to the interface,

   ```
   Router(config-if)#dialer-group dialer-list-number
   ```

   where *dialer list number* is the dialer list that was set up in step 3.

6. Define the idle timeout that you find appropriate for each call,

   ```
   Router(config-if)#dialer idle-timeout call-duration
   ```

   where *call-duration* is the duration of the call in seconds (the default is usually 120 s). The idle timeout is the period of time in which the call will remain alive waiting for more interesting traffic. If more interesting traffic comes in before the timer is over, the timer will be reset. If no interesting traffic comes in, the call will be terminated even if there was non-interesting traffic being transferred.

7. If you are using this link between two points only and your router will be dialing only for one destination using the ISDN network, use the following command to set the dialer string:

   ```
   Router(config-if)#dialer string dialer-string
   ```

   where *dialer string* is the dialer string that is provided to you by the service provider. This dialer string is similar to the phone number that you dial in the regular PSTN. So, you command the router to dial the string of the other side. For further security, you can use a different command that associates the dialing to a destination IP address with a username and a dialer string,

   ```
   Router(config-if)#dialer map ip destination-address name
   username dialer string
   ```

   where

   destination-address is the IP address of the other end of the ISDN link.

   username is the same username that you have set up to use with PPP.

   dialer-string is the dialer string of the other end of the ISDN link.

8. You can optionally use the following command to set up a threshold of load on which the second channel (in a BRI link) becomes active.

   ```
   Router(config-if)#dialer load-threshold threshold either
   ```

   where *threshold* is a number between 1 and 255, 1 being the minimum load and 255 being 100% load on the first channel. This means that this command tells the router to activate the second channel once the first one is *threshold*/255 loaded.

9. You can check the operation of the ISDN using the following commands;

   ```
   Router#show isdn active
   Router#show isdn status
   Router#show dialer
   ```

   and

   ```
   Router#debug isdn q921
   Router#debug isdn q931
   Router#debug dialer
   ```

5.5 How to Configure ISDN Dialer Profiles in a Cisco Router

When would you need this: When you are using ISDN links among more than two nodes.

Special Requirements: The router should have ISDN interface(s).

If you are implementing ISDN between two nodes only, you can refer to the BRI ISDN configuration procedure in Sect. 5.4.

What the dialer profiles do is the mapping of a dial string along with username to a certain destination. This way, the router knows what number to dial for different ISDN destinations using the same link. The main problem you may face without the use of dialer profiles is that the configuration is applied directly to the physical interface. Thus, different logical links will need to use the same IP address and other configuration settings. The dialer profile applies the settings to the interface on on-call basis.

Multiple dialer interfaces may be configured on a router. Each dialer interface is the complete configuration for a destination. The '`interface dialer`' command creates a dialer interface and enters interface configuration mode.

I will assume that you already set the switch type and SPIDs. If you have not done that yet, refer to Sect. 5.4. Let us start the configuration as the following:

1. Create a dialer interface that contains the configuration of the interface to be used with a certain destination.

   ```
   Router(config)#interface dialer interface-number
   ```

   where *interface-number* is the dialer interface number that you may choose.
2. Configure the dialer interface as if you are configuring the regular DDR in Sect. 5.4. This configuration can be IP address, encapsulation and authentication types, idle timer, and dialer group for interesting traffic. You can configure the encapsulation and authentication types on the physical interface later, if all of your connections use the same encapsulation and authentication types.
3. Configure a dialer string to this interface, along with a '`dialer remote-name`'

   ```
   Router(config-if)#dialer string dialer-string
   Router(config-if)#dialer remote-name destination-name
   ```

   where *dialer-string* is the dial string for the destination and *destination-name* is the name of the destination. Usually, you are supplied with two dial strings for each ISDN end. Just repeat the '`dial string`' command once for each dial string.
4. Associate the dialer interface to a dialer pool. This pool will be associated to one or a group of physical interfaces such that the physical interfaces use these dialer settings on on-call basis.

   ```
   Router(config-if)#dialer pool pool-number
   ```

   where *pool-number* is the dialer pool number.

   Now, repeat steps 1–4 for as many destinations as you have that can be contacted by this router (using the ISDN network). For the destinations you want to use the same physical interface, give the same dialer pool number; *pool-number*.
5. After you finish setting the dialer interfaces, one step is left; associating the physical interfaces to the dialer pool. This is done using the following command:

   ```
   Router(config-if)#dialer pool-member pool-number
   ```

   where *pool-number* is the dialer pool number that you want this physical interface to be associated with.

   Please note that this command must be issued on the physical interface not the dialer interface. You can make the same physical interface a member of more than one dialer pool.

As an optional parameter, you can set priority of the physical interface in the dialer pool if the pool contains more than one physical member. An example is the following,

```
Router(config-if)#dialer pool-member 1 priority 100
```

where *100* is the priority you chose for this physical interface.

If multiple calls need to be placed and only one interface is available, then the dialer pool with the highest priority is the one that dials out.

In general, the dialer pool can be used with any combination of synchronous, asynchronous, BRI, and PRI ISDN interfaces.

5.6 How to Configure Frame-Relay in a Cisco Router

When would you need this: When you are setting up a Frame-relay WAN connection rented from a service provider.

Special Requirements: None.

Frame-relay configuration mainly depends on the topology you are using. Frame-relay may be used as a point-to-point link between two ends, point-to-multipoint between one central station and a few terminal stations, multiple point-to-point links between a central station and a few terminal stations.

5.6.1 Point-to-Point Connection of Two Sites Using Physical Interfaces

1. On the serial interface, change the encapsulation type to Frame-relay:
   ```
   Router(config)#interface serial interface-number
   Router(config-if)#encapsulation Frame-relay
   ```
   where *interface number* is the number of the serial interface connected to the frame-relay equipment.
2. Configure the LMI type:
   ```
   Router(config-if)#Frame-relay lmi-type lmi-type
   ```
   where *lmi-type* is the type of LMI standard used. The supported types are `Cisco`, `ansi` and `q933a`. This information should be given to you by the Frame-relay service provider.
3. Assign an IP address to the interface
   ```
   Router(config-if)#ip address ip-address1 subnetmask1
   ```
   where the *ip address1* and *subnetmask1* are the IP address and subnetmask assigned to the Frame-relay interface on the first side of the link.
4. Map the Frame-relay DLCI number to a destination IP address:

`Router(config-if)#`**`Frame-relay map`** *`ip-address2 dlci-number encapsulation-type`*

where

ip-address2 is the IP address of the other side of the link.

dlci-number is the virtual circuit number given to you by the Frame-relay service provider.

encapsulation-type is the type of encapsulation standard used. The value is usually either `Cisco` or `ietf`. This information should also be given to you by the Frame-relay service provider.

5. On the other end, the serial interface encapsulation type is changed to Frame-relay:

   `Router(config)#`**`interface serial`** *`interface-number`*

   `Router(config-if)#`**`encapsulation Frame-relay`**

   where *interface number* is the number of the serial interface connected to the Frame-relay equipment.

6. Configure the LMI type:

   `Router(config-if)#`**`Frame-relay lmi-type`** *`lmi-type`*

   where *lmi-type* is the type of LMI standard used. The supported types are `Cisco`, `ansi` and `q933a`. This information should be given to you by the Frame-relay service provider. Usually, it is the same type used in step 2.

7. Assign an IP address to the interface

   `Router(config-if)#`**`ip address`** *`ip-address2 subnetmask2`*

   where the *ip address2* and *subnetmask2* are the IP address and subnetmask assigned to the Frame-relay interface on the second side of the link.

8. Map the Frame-relay DLCI number to a destination IP address:

   `Router(config-if)#`**`Frame-relay map`** *`ip-address1 dlci-number encapsulation-type`*

   where

   ip address1 is the IP address of the first side of the link.

   dlci-number is the virtual circuit number given to you by the Frame-relay service provider.

   encapsulation-type is the type of encapsulation standard used. The value is usually either `Cisco` or `ietf`. This information should also be given to you by the Frame-relay service provider.

9. Use the following commands for troubleshooting:

   `Router#`**`show Frame-relay lmi`**

   `Router#`**`show Frame-relay pvc`**

5.6.2 Point-to-Multipoint Using Physical Interfaces

In a point-to-multipoint Frame-relay connection, a central node is connected to a group of nodes using a single physical line. The Frame-relay network will recognize the different destinations through the use of different DLCI numbers on the same link.

The configuration is similar to the previous subsection except that at the central node multiple mappings are configured on the Frame-relay interface while a single mapping is configured on each terminal interface.

1. At the central node, on the serial interface, change the encapsulation type to Frame-relay:
   ```
   Router(config)#interface serial interface-number
   Router(config-if)#encapsulation Frame-relay
   ```
   where *interface number* is the number of the serial interface connected to the Frame-relay equipment.
2. Configure the LMI type:
   ```
   Router(config-if)#Frame-relay lmi-type lmi-type
   ```
   where *lmi-type* is the type of LMI standard used. The supported types are `Cisco`, `ansi` and `q933a`. This information should be given to you by the Frame-relay service provider.
3. Assign an IP address to the interface
   ```
   Router(config-if)#ip address central-ip-address subnetmask1
   ```
   where the *central ip address* and *subnetmask1* are the IP address and subnetmask assigned to the Frame-relay interface on the central side of the link.
4. Map the Frame-relay DLCI number to a destination IP address:
   ```
   Router(config-if)#Frame-relay map ip-address2 dlci-number encapsulation-type
   ```
   where

   ip address2 is the IP address of the other side of the link.

   dlci-number is the virtual circuit number given to you by the Frame-relay service provider.

   encapsulation-type is the type of encapsulation standard used. The value is usually either `Cisco` or `ietf`. This information should also be given to you by the Frame-relay service provider.

   This command is repeated once for every terminal node. Each terminal node would have a different DLCI number.
5. On the terminal end, the serial interface encapsulation type is changed to Frame-relay:
   ```
   Router(config)#interface serial interface-number
   Router(config-if)#encapsulation Frame-relay
   ```
   where *interface number* is the number of the serial interface connected to the Frame-relay equipment.
6. Configure the LMI type:
   ```
   Router(config-if)#Frame-relay lmi-type lmi-type
   ```
   where *lmi-type* is the type of LMI standard used. The supported types are `Cisco`, `ansi` and `q933a`. This information should be given to you by the Frame-relay service provider. Usually, it is the same type used in step 2.
7. Assign an IP address to the interface
   ```
   Router(config-if)#ip address ip-address2 subnetmask2
   ```

where the *ip address2* and *subnetmask2* are the IP address and subnetmask assigned to the Frame-relay interface on the second side of the link.

8. Map the Frame-relay DLCI number to a destination IP address:

   ```
   Router(config-if)#Frame-relay map central-ip-address dlci-number encapsulation-type
   ```

   where

   central-ip-address is the IP address of the central side of the link.

   dlci-number is the virtual circuit number given to you by the Frame-relay service provider.

   encapsulation-type is the type of encapsulation standard used. The value is usually either `Cisco` or `ietf`. This information should also be given to you by the Frame-relay service provider.

5.6.3 Point-to-Multipoint Using Logical Interfaces

In what we call multiple-point-to-point scenario, a single central station is connected through a single physical link the Frame-relay network. Through that Frame-relay network, the central node is also connected to multiple terminal nodes. However, these connections are done by creating a single logical point-to-multipoint link carried over the single physical link.

1. At the central node, on the serial interface, change the encapsulation type to Frame-relay:

   ```
   Router(config)#interface serial interface-number
   Router(config-if)#encapsulation Frame-relay
   ```

   where *interface number* is the number of the serial interface connected to the Frame-relay equipment.

2. Configure the LMI type:

   ```
   Router(config-if)#Frame-relay lmi-type lmi-type
   ```

   where *lmi-type* is the type of LMI standard used. The supported types are `Cisco`, `ansi` and `q933a`. This information should be given to you by the Frame-relay service provider.

3. Assure that there is no IP address assigned to the interface

   ```
   Router(config-if)#no ip address
   ```

4. Create logical interface:

   ```
   Router(config-if)#interface serial interface-number.logical-interface-number point-to-multipoint
   ```

5. On the logical interface, assign an IP address:

   ```
   Router(config-if)#ip address ip-address1 subnetmask1
   ```

   where the *ip-address1* and *subnetmask1* are the IP address and subnetmask assigned to the Frame-relay logical interface on the central side of the link.

6. Map the interface to a specific DLCI number:

   ```
   Router(config-subif)#Frame-relay interface-dlci dlci-number
   ```

   where

   dlci-number is the virtual circuit number given to you by the Frame-relay service provider. This DLCI number resembles the virtual circuit leading to a specific remote node.
7. Repeat steps 6 for as many remote nodes as you need.
8. On the remote node, the serial interface encapsulation type is changed to Frame-relay:

   ```
   Router(config)#interface serial interface-number
   Router(config-if)#encapsulation Frame-relay
   ```

   where *interface number* is the number of the serial interface connected to the Frame-relay equipment.
9. Configure the LMI type:

   ```
   Router(config-if)#Frame-relay lmi-type lmi-type
   ```

   where *lmi-type* is the type of LMI standard used. The supported types are `Cisco`, `ansi` and `q933a`. This information should be given to you by the Frame-relay service provider. Usually, it is the same type used in step 2.
10. Assign an IP address to the interface

    ```
    Router(config-if)#ip address ip-address2 subnetmask2
    ```

    where the *ip-address2* and *subnetmask2* are the IP address and subnetmask assigned to the Frame-relay interface on the remote side of the link.
11. Map the Frame-relay DLCI number to a destination IP address:

    ```
    Router(config-if)#Frame-relay map ip-address1 dlci-number encapsulation-type
    ```

    where

    ip-address1 is the IP address of the first side of the link.

    dlci-number is the virtual circuit number given to you by the Frame-relay service provider.

    encapsulation-type is the type of encapsulation standard used. The value is usually either `Cisco` or `ietf`. This information should also be given to you by the Frame-relay service provider.
12. Repeat steps 8, 9, 10, and 11 on each remote node using different IP addresses and DLCI numbers.

5.6.4 Multiple Point-to-Point Using Logical Interfaces

In what we call multiple-point-to-point scenario, a single central station is connected through a single physical link the Frame-relay network. Through that Frame-relay network, the central node is also connected to multiple terminal nodes. However, these connections are done by creating multiple logical point-to-point links carried over the single physical link. This way, the separation of traffic handled from one node to the other is clearer and the remote nodes cannot communicate unless the traffic passes through the central node.

1. At the central node, on the serial interface change the encapsulation type to Frame-relay:
   ```
   Router(config)#interface serial interface-number
   Router(config-if)#encapsulation Frame-relay
   ```
   where *interface number* is the number of the serial interface connected to the Frame-relay equipment.
2. Configure the LMI type:
   ```
   Router(config-if)#Frame-relay lmi-type lmi-type
   ```
   where *lmi-type* is the type of LMI standard used. The supported types are `Cisco`, `ansi` and `q933a`. This information should be given to you by the Frame-relay service provider.
3. Assure that there is no IP address assigned to the interface
   ```
   Router(config-if)#no ip address
   ```
4. Create logical interface:
   ```
   Router(config-if)#interface serial interface-number.logical-interface-number point-to-point
   ```
5. On the logical interface, assign an IP address:
   ```
   Router(config-if)#ip address ip-address1 subnetmask1
   ```
   where the *ip-address1* and *subnetmask1* are the IP address and subnetmask assigned to the Frame-relay logical interface on the central side of the link.
6. Map the interface to a specific DLCI number:
   ```
   Router(config-subif)#Frame-relay interface-dlci dlci-number
   ```
   where

   dlci-number is the virtual circuit number given to you by the Frame-relay service provider. This DLCI number resembles the virtual circuit leading to a specific remote node.
7. Repeat steps 4, 5 and 6 for as many remote nodes as you need.
8. On the remote node, the serial interface encapsulation type is changed to Frame-relay:
   ```
   Router(config)#interface serial interface-number
   Router(config-if)#encapsulation Frame-relay
   ```
   where *interface number* is the number of the serial interface connected to the Frame-relay equipment.
9. Configure the LMI type:
   ```
   Router(config-if)#Frame-relay lmi-type lmi-type
   ```
   where *lmi-type* is the type of LMI standard used. The supported types are `Cisco`, `ansi` and `q933a`. This information should be given to you by the Frame-relay service provider. Usually, it is the same type used in step 2.
10. Assign an IP address to the interface
    ```
    Router(config-if)#ip address ip-address2 subnetmask2
    ```

where the *ip-address2* and *subnetmask2* are the IP address and subnetmask assigned to the Frame-relay interface on the remote side of the link.

11. Map the Frame-relay DLCI number to a destination IP address:
    `Router(config-if)#`**`Frame-relay map`** *`ip-address1 dlci-number encapsulation-type`*
    where
    ip-address1 is the IP address of the first side of the link.
    dlci-number is the virtual circuit number given to you by the Frame-relay service provider.
    encapsulation-type is the type of encapsulation standard used. The value is usually either `Cisco` or `ietf`. This information should also be given to you by the Frame-relay service provider.
12. Repeat steps 8, 9, 10, and 11 on each remote node using different IP addresses and DLCI numbers.

5.6.5 Frame-Relay and Routing Issues

Cisco routers employ a technique called split-horizon. This technique is used to eliminate routing loops by which a routing update cannot be forwarded to the same interface it came from.

Building on that logic, split-horizon can cause issues when using Frame-relay point-to-multipoint topologies. Now think of a scenario where a routing update is coming from one of the remote points connected on the other end of a point-to-multipoint link. The routing update, due to split-horizon, will not be forwarded on the same physical link over to the other points connected to the point-to-multipoint topology, because it will be considered coming from one interface and cannot be forwarded over to the same interface. This way, the other points will not be able to exchange routing updates.

Split-horizon can be disabled using the following command on the interface level:

```
Router(config-if)#no ip split-horizon
```

On OSPF, you can use the following command:

```
Router(config-if)#ip ospf network point-to-multipoint
```

5.7 How to Configure a Cisco Router as a Frame-Relay Switch

When would you need this: When you are setting up your own Frame-Relay network. This set up is frequently used for laboratory setup.

Special Requirements: A Cisco router with at least two serial interfaces.

This setup is mainly done for laboratory experiments because operating a Cisco router as an actual Frame-Relay Switch requires a high number of Serial interfaces.

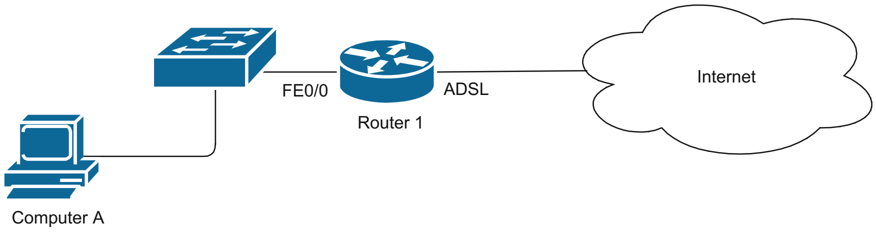
As a start, you need to keep in mind that when a Cisco router operates as a Frame-relay switch, it will stop operating as an IP router. No IP routing processes will occur during the Frame-Relay operation. The router will become, exclusively, a Frame-Relay Switch.

Before you start the configuration, draw the network topology and mark on it the numbers of DLCIs that will be used. What the Frame-relay switch does is receiving a frame with a certain DLCI number from one interface and forwarding it to a different interface after assigning it a different DLCI number. With that said, now we move on to the configuration:

1. Enable Frame-Relay Switching operation on the router's global configuration:
   ```
   Router(config)#Frame-relay switching
   ```
2. Configure the two (or more) serial interfaces that will participate in the Frame-relay switching process
   ```
   Router(config-if)#no ip address
   Router(config-if)#encapsulation Frame-relay
   Router(config-if)#logging event subif-link-status
   Router(config-if)#logging event dlci-status-change
   Router(config-if)#clock rate clock-rate
   Router(config-if)#no Frame-relay inverse-arp
   Router(config-if)#Frame-realy intf-type dce
   ```
   where *clock rate* is the clock rate of your choice (64,000 is a good choice).
   Now, compare the Frame-relay routing configuration on the same interface,
   ```
   Router(config-if)#Frame-relay route incoming-dlci interface serial interface-number outgoing-dlci
   ```
   where
   incoming-dlci is the DLCI number of the incoming frame.
   outgoing-dlci is the DLCI number that will be assigned to the outgoing frame.
   interface-number is the serial interface number to which the frame will be forwarded to be sent out of the router.
   Repeat the Frame-relay routing command for as much DLCIs as you plan to be passing through this interface. Keep in mind that this command is given at the interface that is receiving the Frame-relay frames.
3. After completing the steps of configuration for one of the interfaces in step 2, repeat step 2 on each serial interface you want to be part of the Frame-relay switching process.
4. For verification and troubleshooting, use the following command to find out the status of each route you have configured on the Frame-relay switch:
   ```
   Router#show Frame-realy route
   ```

5.8 Training Scenarios

Scenario 5.1



Connect the network shown in the figure above and configure the following settings:

1. On Router 1:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.0.254/24

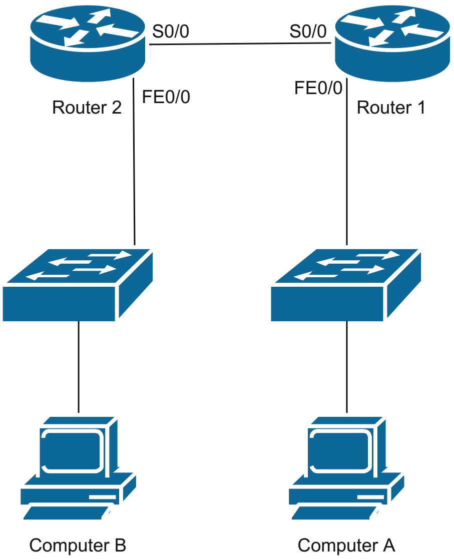
2. On Computer A:

Device	Parameter	Value
Computer A	IP Address	192.168.0.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254

3. Get the following information from your ADSL service provider:
   Your account's username and password, MTU size (usually 1492), and PVC value (usually 0/35 or 8/35). You will also need to know whether the IP address assigned to you by the ISP is a static public IP address, or your address is going to be dynamically assigned.
4. Enable the PPPoE service in global configuration of the router.
5. Set up the physical ADSL interface with no IP address and the PVC number given to you by the service provider.
6. Set up the dialer to get the IP address automatically from the service provider.

7. Set up a default route to the ADSL interface with AD of 121.
8. Test the connection by PING from Computer A to springer.com

Scenario 5.2



Connect the network shown in the figure above and configure the following settings:

1. On the routers:

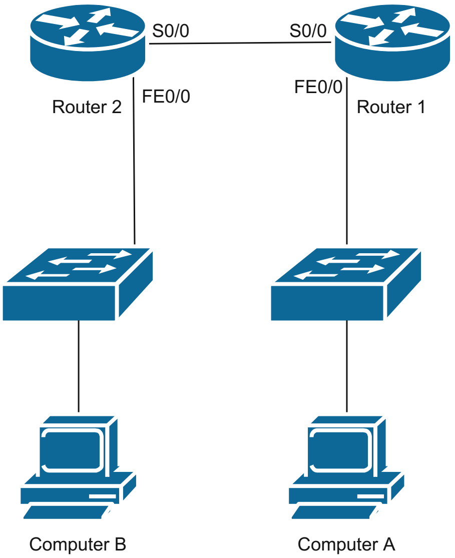
Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	192.168.1.254/24
	Interface S0/0 IP Address	10.0.0.2/30

2. On the computers:

Device	Parameter	Value
Computer A	IP Address	192.168.0.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254
Computer B	IP Address	192.168.1.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.1.254
	DNS Server 1	192.168.1.254
	DNS Server 2	192.168.1.254

3. On Router 1, configure a default route to route all traffic through the interface S0/0.
4. On Router 2, configure a default route to route all traffic through the interface S0/0.
5. On Router 1, set the encapsulation type on S0/0 to be PPP. The interface will be down now.
6. Set up the authentication to PAP and identify the sent-username and password along with the other side's username and password.
7. On Router 2, set the encapsulation type on S0/0 to be PPP.
8. Set up the authentication to PAP and identify the sent-username and password along with the other side's username and password.
9. Both serial interfaces should be up now.
10. Test the connectivity by PING from Computer A to Computer B.

Scenario 5.3



Connect the network shown in the figure above and configure the following settings:

1. On the routers:

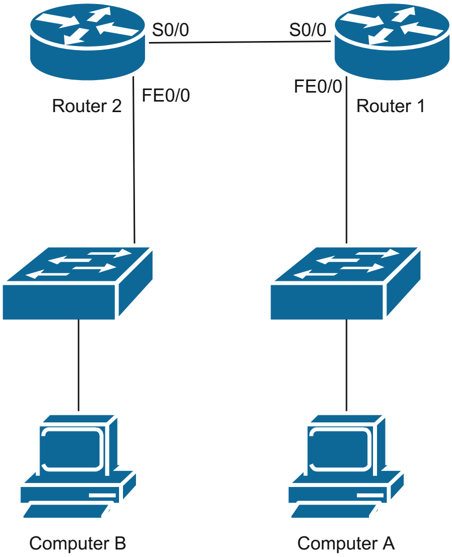
Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	192.168.1.254/24
	Interface S0/0 IP Address	10.0.0.2/30

2. On the computers:

Device	Parameter	Value
Computer A	IP Address	192.168.0.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254
Computer B	IP Address	192.168.1.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.1.254
	DNS Server 1	192.168.1.254
	DNS Server 2	192.168.1.254

3. On Router 1, configure a default route to route all traffic through the interface S0/0.
4. On Router 2, configure a default route to route all traffic through the interface S0/0.
5. On Router 1, set the encapsulation type on S0/0 to be PPP. The interface will be down now.
6. Set up the authentication to CHAP and the other side's username and password.
7. On Router 2, set the encapsulation type on S0/0 to be PPP.
8. Set up the authentication to CHAP and the other side's username and password.
9. Both serial interfaces should be up now.
10. Test the connectivity by PING from Computer A to Computer B.

Scenario 5.4



Connect the network shown in the figure above and configure the following settings:

1. On the routers:

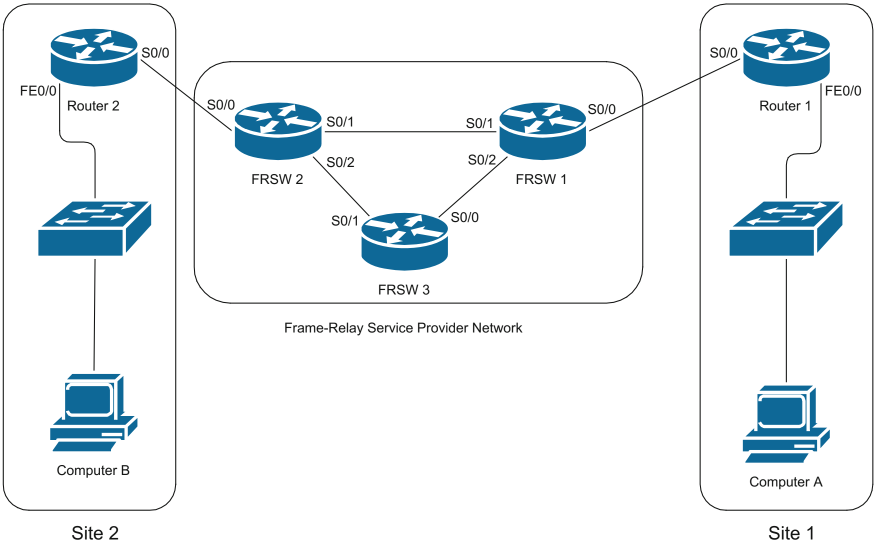
Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	192.168.1.254/24
	Interface S0/0 IP Address	10.0.0.2/30

2. On the computers:

Device	Parameter	Value
Computer A	IP Address	192.168.0.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254
Computer B	IP Address	192.168.1.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.1.254
	DNS Server 1	192.168.1.254
	DNS Server 2	192.168.1.254

3. On Router 1, configure a default route to route all traffic through the interface S0/0.
4. On Router 2, configure a default route to route all traffic through the interface S0/0.
5. On Router 1, set the encapsulation type on S0/0 to be HDLC.
6. On Router 2, set the encapsulation type on S0/0 to be HDLC.
7. Both serial interfaces should be up now.
8. Test the connectivity by PING from Computer A to Computer B.

Scenario 5.5



Connect the network shown in the figure above and configure the following settings:

1. Client routers:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.0.254/24
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	192.168.1.254/24

2. Client computers:

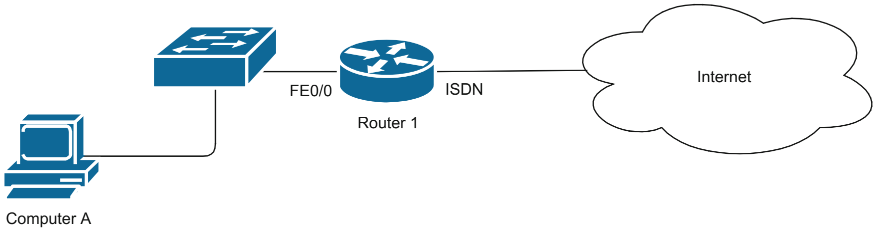
Device	Parameter	Value
Computer A	IP Address	192.168.0.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254
Computer B	IP Address	192.168.1.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.1.254
	DNS Server 1	192.168.1.254
	DNS Server 2	192.168.1.254

3. Frame-relay service provider settings:
   Set up the three routers FRSW 1, 2, and 3 to work as Frame-Relay Switches with the following mapping:

Router	Interface$_{\text{in}}$	DLCI$_{\text{in}}$	Interface$_{\text{out}}$	DLCI$_{\text{out}}$
FRSW 1	S0/0	*100*	S0/2	120
FRSW 1	S0/1	180	S0/0	100
FRSW 2	S0/0	*120*	S0/1	170
FRSW 2	S0/2	150	S0/2	110
FRSW 3	S0/0	120	S0/1	150

4. Set up Router 1 to operate on Frame-relay encapsulation with the mapping of data going to Router 2's LAN to be sent out through S0/0 with DLCI 100.
5. Set up Router 2 to operate on Frame-relay encapsulation with the mapping of data going to Router 1's LAN to be sent out through S0/0 with DLCI 120.
6. Test the setup by PING from Computer A to Computer B.

Scenario 5.6



Connect the network shown in the figure above and configure the following settings:

1. Router 1's basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.168.0.254/24

2. Computer A's configuration:

Device	Parameter	Value
Computer A	IP Address	192.168.0.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.168.0.254
	DNS Server 1	192.168.0.254
	DNS Server 2	192.168.0.254

3. Set up a default route on Router 1 to forward all outgoing traffic through the BRI interface `bri0`.
4. Configure the ISDN switch type, SPIDs, and LDNs. This information is usually received from the service provider.
5. Using an Access-list, select the 'interesting traffic' that is allowed to set up a call to the service provider.

6. Configure the encapsulation protocol as PPP. Set up CHAP authentication. The username and password used in authentication are obtained from the service provider.
7. Apply the dialer list created in step 5 and set up the connection time-out.
8. Set up the dialer map and link it to the username.
9. For testing, PING from Computer A to springer.com

Chapter 6
Security Configuration

Keywords Cisco · Router · Password-encryption · VPN · Site-to-site VPN · PPTP · PPTP VPN · ACL · Access-list · Access-control list · Route authentication · GRE · IPSEC · AAA · RADIUS · TACACS+

6.1 How to Secure Passwords on a Cisco Router

Encrypt the passwords such that they become non-comprehendible to anyone viewing them in the running-configuration.

Secret password is already encrypted. All other passwords (vty, console, and auxiliary) are not. The command to encrypt them is as follows:

```
Router(config)#service password-encryption
```

There are two recommended methods to use this command. Because it is a service, it is not advised to keep it running all the time because it would consume processing power and memory. Thus, it can be used and turned off and the passwords will remain encrypted. One way to do this is to turn this command before setting up any passwords and turning it off after finishing the password setup commands using the following command:

```
Router(config)#no service password-encryption
```

The second way to do it is after finishing the setup of all passwords, turn on password encryption, issue a '`show running-config`' at the privilege mode, and then turn the password encryption off.

The encryption used here is very weak. The only purpose of it is to prevent people looking at the configuration from knowing the password.

M.M. Alani, *Guide to Cisco Routers Configuration*,
DOI 10.1007/978-3-319-54630-8_6

6.2 How to Configure Access-Control Lists on a Cisco Router

When would you need this: When you need to filter traffic to deny and allow packets based on specific features.

Special Requirements: None.

An access-control list (ACL) is a filter configured on a Cisco router to control which packets are allowed to pass through a specific interface in a specific direction.

The ACL anatomy is simple, a list of commands executed sequentially. The commands are basically a condition and effect. If the packet meets a specific conditions, it can either be allowed to pass or denied (discarded). As we said, this list of commands is executed sequentially, and when a packet meets one condition, the effect is applied to this packet, whether it is allow or deny, and then the rest of the conditions are ignored.

To understand this better, let us go through an example. Let us think of an ACL like the following:

1. Condition 1—deny;
2. Condition 2—deny;
3. Condition 3—allow; and
4. Condition 4—allow.

Now, if the packet meets condition 1, it will be denied and no further comparison with other conditions is done. If the packet does not meet condition 1, the packet will be compared against condition 2. If it meets condition 2, it will be denied as well. If it does not meet condition 2, the packet will be compared to condition 3. If it meets condition 3, it will be allowed to pass. If it does not meet condition 3, the packet will be compared to condition 4. If it meets condition 4, it will be allowed to pass. This means that if a packet meets more than one condition, only the effect of the first condition met will be executed. If the packet does not meet condition 4, the packet will be denied.

Generally, for better security, if the packet does not meet any condition, it will be denied. This is usually referred to as ‘Implicit Deny’ at the end of the any ACL. This means that if all the effects in your ACL is ‘deny,’ all traffic will be denied. You need at least one ‘permit’ in and ACL.

ACLs can be categorized into two types: standard and extended. Depending on the type of the ACL, the condition can be either detailed or coarse. Standard ACLs can filter based on the source IP address only. Although this makes the applications of standard ACL limited, it can be useful in some scenarios. An extended ACL can filter based on source and destination IP addresses, source and destination port numbers, protocol type, protocol-specific features (like allowing echo-request but not echo-reply in ICMP), and even on time of receiving the packet.

6.2.1 Standard Access-List Configuration for IPv4

A standard Access-list is identified by having a number between 1 and 99. As explained earlier, it can only filter based on source IP address.

1. Create the ACL with the first condition and effect:
   ```
   Router(config)#Access-list acl-number {permit|deny} source-address
   ```
   where
   acl-number is the ACL number (must be between 1 and 99 for standard ACL).
   permit|deny is the effect of filter. You should select either `permit` or `deny`.
   source-ip-address is the source IPv4 address. This address can be in one of the following formats:

   a. **host** *ipv4-host-address* to identify a single address.
   b. *network-address wildcard-mask* to identify an IPv4 network.
   c. **any** to identify all host addresses.

2. Repeat the command from step 1 for as many filters you need to add to the same ACL as you see necessary. Use the same *acl-number* to add more conditions and effects to the same ACL. Keep in mind that you need to have at least one '`permit`' and you can have one ACL applied per interface per direction so you need to put all of the filters you need in that direction on that interface in a single ACL.
3. Now, you need to apply the ACL you configured in steps 1 and 2 on a specific interface in a specific direction.
   ```
   Router(config)#interface interface-type interface-number
   Router(config-if)#ip access-group acl-number {in|out}
   ```
   where
   interface-type and *interface-number* are the type and number of the interface.
   acl-number is the ACL number that we used in steps 1 and 2.
   in|out is the direction of traffic: '`in`' for incoming traffic (going into the router through the interface) and '`out`' for outgoing from the router to outside through the interface.
4. For troubleshooting, use the following commands:
   ```
   Router#show Access-list
   Router#show Access-list acl-number
   Router#show ip interface interface-type interface-number
   ```

6.2.2 Extended Access-List Configuration for IPv4

Extended ACLs have the ability to screen more packet features as compared to the standard ACL. Extended ACLs can filter based on protocol type, source IP address and port number, and destination IP address and port number. Extended ACLs have the number range of 100–199.

1. Create the ACL with the first condition and effect:
   `Router(config)#`**`Access-list`** *`acl-number`* `{`**`permit|deny`**`}` *`protocol-name source-ip-address operator1 source-port destination-ip-address operator2 destination-port`*
   where
   permit|deny is the effect that you want to have if the packet meets the condition. You can use permit to allow traffic or deny to block traffic.
   protocol-name is the name of the protocol you want to filter. This can be `ip`, `tcp`, `udp`, etc.
   source-ip-address is the source IPv4 address. This address can be in one of the following formats:

   d. **`host`** *`ipv4-host-address`* to identify a single address.
   e. *`network-address wildcard-mask`* to identify an IPv4 network.
   f. **`any`** to identify all host addresses.

   Operator1 is an operator used to identify the port. It can be the following:

   a. `eq` equal to a specific port number identified in the following parameter.
   b. `gt` all ports greater than the following port number.
   c. `lt` all ports less than the following port number.

   source-port is the source port number. The *operator1 source-port* parameters are optional.
   destination-ip-address is the destination IPv6 address. This address can be in one of the following formats:

   a. **`host`** *`ipv6-host-address`* to identify a single address
   b. *`network-address wildcard-mask`* to identify an IPv4 network.
   c. **`any`** to identify all host addresses.

   operator2 is an operator used to identify the port. It can be the following:

   a. `eq` equal to a specific port number identified in the following parameter.
   b. `gt` all ports greater than the following port number.
   c. `lt` all ports less than the following port number.

   destination-port is the destination port number. The *operator2 destination-port* parameters are optional.

2. Repeat step 1 for as many filters as you feel necessary. Remember that you must have at least one command of `permit`. Otherwise, all traffic will be blocked on the interface in that direction.
3. Apply the Access-list to a specific interface in a specific direction:
   ```
   Router(config)#interface interface-type interface-number
   Router(config-if)#ip access-group acl-number {in|out}
   ```
   where
   interface-type and *interface-number* are the type and number of the interface.
   acl-number is the ACL number that we identified in steps 1 and 2.
   in|out is the direction of traffic: '`in`' for incoming traffic (going into the router through the interface) and '`out`' for outgoing from the router to outside through the interface.
4. For troubleshooting, use the following commands:
   ```
   Router#show Access-list
   Router#show Access-list acl-number
   Router#show ip interface interface-type interface-number
   ```

6.2.3 Removing Access-Lists

Removing an Access-list must be done in the exact reverse order of creating and apply the ACL. First, you will need to remove the application of the Access-list.

```
Router(config-if)#no ip access-group acl-number {in|out}
```

where
acl-number is the number of the Access-list to be removed.
The next step is to delete the Access-list. This can be done using a single command:
in|out is the direction in which the Access-list was applied.

```
Router(config)#no Access-list acl-number
```

where *acl-number* is the number of the Access-list to be removed.
If you delete the Access-list before removing it from the interface, all traffic will be blocked in the ACL's direction until you remove the ACL application.

6.3 How to Configure Advanced Access-Control Lists on a Cisco Router

6.3.1 Named Access-Lists

Named ACLs require IOS version 11.2 or higher. Named ACL is not really a special type. A named ACL is either standard or extended ACL with a name instead of a number.

For standard named ACL, the configuration is as follows:

1. Create the named ACL:
   ```
   Router(config)#ip Access-list standard acl-name
   ```
   where *acl-name* is the name of the Access-list.
2. Configure the filtering condition and effect:
   ```
   Router(config-std-nacl)#{permit|deny} source-ip-address
   ```
   where
   permit|deny is the effect of filter. You should select either `permit` or `deny`.
   source-ip-address is the source IPv4 address. This address can be in one of the following formats:

   a. **host** *ipv4-host-address* to identify a single address.
   b. *network-address wildcard-mask* to identify an IPv4 network.
   c. **any** to identify all host addresses.

3. Repeat step 2 for as many filters you see necessary.
4. Apply the Access-list to an interface in a specific direction:
   ```
   Router(config)#interface interface-type interface-number
   Router(config-if)#ip access-group acl-name {in|out}
   ```
   where
   interface-type and *interface-number* are the type and number of the interface
   acl-name is the ACL name that we identified in step 1.
   in|out is the direction of traffic: '`in`' for incoming traffic (going into the router through the interface) and '`out`' for outgoing from the router to outside through the interface.
5. For troubleshooting, use the following commands:
   ```
   Router#show Access-list
   Router#show Access-list acl-name
   Router#show ip interface interface-type interface-number
   ```

For an extended named ACL, the configuration is as follows:

1. Create the named ACL:
   ```
   Router(config)#ip Access-list extended acl-name
   ```
   where *acl-name* is the name of the Access-list.
2. Configure the filtering condition and effect:
   ```
   Router(config-ext-nacl)#{permit|deny} protocol-name source-ip-address operator1 source-port destination-ip-address operator2 destination-port
   ```
   where
   permit|deny is the effect that you want to have if the packet meets the condition. You can use permit to allow traffic or deny to block traffic.
   protocol-name the name of the protocol you want to filter. This can be `ip`, `tcp`, `udp`, etc.

source-ip-address is the source IPv4 address. This address can be in one of the following formats:

g. `host` *`ipv4-host-address`* to identify a single address.
h. *`network-address wildcard-mask`* to identify an IPv4 network.
i. **`any`** to identify all host addresses.

Operator1 is an operator used to identify the port. It can be the following:

d. `eq` equal to a specific port number identified in the following parameter.
e. `gt` all ports greater than the following port number.
f. `lt` all ports less than the following port number.

source-port is the source port number. The *operator1 source-port* parameters are optional.
destination-ip-address is the destination IPv6 address. This address can be in one of the following formats:

d. **`host`** *`ipv6-host-address`* to identify a single address.
e. *`network-address wildcard-mask`* to identify an IPv4 network.
f. **`any`** to identify all host addresses.

operator2 is an operator used to identify the port. It can be the following:

d. `eq` equal to a specific port number identified in the following parameter.
e. `gt` all ports greater than the following port number.
f. `lt` all ports less than the following port number.

destination-port is the destination port number. The *operator2 destination-port* parameters are optional.

3. Repeat step 2 for as many filters you see necessary.
4. Apply the Access-list to an interface in a specific direction:

```
Router(config)#interface interface-type interface-number
Router(config-if)#ip access-group acl-name {in|out}
```

where
interface-type and *interface-number* are the type and number of the interface.
acl-name is the ACL name that we identified in step 1.
in|out is the direction of traffic: '`in`' for incoming traffic (going into the router through the interface) and '`out`' for outgoing from the router to outside through the interface.

4. For troubleshooting, use the following commands:

```
Router#show Access-list
Router#show Access-list acl-name
Router#show ip interface interface-type interface-number
```

6.3.2 About Named Access-Lists

Named Access-lists are preferred among networking professionals because of two reasons; first, the name can express the purpose of the ACL, second, you can edit named Access-lists.

For example, if you wrote the following ACL:

```
Router(config)#ip Access-list extended ServerTraffic
Router(config-ext-nacl)#deny tcp 192.168.0.0 0.0.0.255
host 192.168.1.110 eq ftp
Router(config-ext-nacl)#deny tcp 192.168.0.0 0.0.0.255
host 192.168.1.110 eq 23
Router(config-ext-nacl)#permit ip any any
```

Using this Access-list, all FTP traffic from your local network will be block. In addition, all Telnet traffic will also be blocked, while all other traffic is permitted. After a while, you remembered that you need to allow the administrator's PC from the local network to use Telnet to the server. Instead of erasing the whole Access-list and rewriting it, it is possible to edit it through the following steps.

If you run the following command:

```
Router#show Access-list ServerTraffic
```

You will get an output similar to this:

```
ip Access-list extended ServerTraffic
10 deny tcp 192.168.0.0 0.0.0.255 host 192.168.1.110
20 deny tcp 192.168.0.0 0.0.0.255 host 192.168.1.110 eq 23
30 permit ip any any
```

If you need to add another filter between the first one and the second one, all you need to do is the following:

```
Router(config)#ip Access-list extended ServerTraffic
Router(config-ext-nacl)#15 permit host 192.168.0.118 host
192.168.1.110 eq 23
```

Now, if you run a `show Access-list ServerTraffic` command, you will see the following output:

```
ip Access-list extended ServerTraffic
10 deny tcp 192.168.0.0 0.0.0.255 host 192.168.1.110 eq
ftp
15 permit host 192.168.0.118 host 192.168.1.110 eq 23
20 deny tcp 192.168.0.0 0.0.0.255 host 192.168.1.110 eq 23
30 permit ip any any
```

You can use any number between the two lines, like 11, 12,…,19.

It is also possible to delete specific lines like the following example:

```
Router(config)#ip Access-list extended ServerTraffic
Router(config-ext-nacl)#no 15 permit host 192.168.0.118
host 192.168.1.110 eq 23
```

This flexibility makes it much easier to modify without having to rewrite everything.

6.3.3 Access-Lists for IPv6

IPv6 Access-lists are configured in a way similar to extended named ACLs explained in the previous subsection.

1. Create an IPv6 ACL:
   `Router(config)#`**`ipv6 Access-list`** *`acl-name`*
   where *acl-name* is the name of the Access-list.
2. Write the filtering commands:
   `Router(config-ipv6-acl)#{`**`permit|deny`**`}` *`protocol-name ipv6-source-address operator1 source-port ipv6-destination-address operator2 destination-port`*
   where
   permit|deny is the effect that you want to have if the packet meets the condition. You can use permit to allow traffic or deny to block traffic.
   protocol-name is the name of the protocol you want to filter. This can be `ip`, `tcp`, `udp`, etc.
   ipv6-source-address is the source IPv6 address. This address can be in one of the following formats:

   a. **`host`** *`ipv6-host-address`* to identify a single address
   b. *`ipv6-network-prefix`*`/`*`prefix-length`* to identify an IPv6 network.
   c. **`any`** to identify all host addresses.

   Operator1 is an operator used to identify the port. It can be the following:

   g. `eq` equal to a specific port number identified in the following parameter.
   h. `gt` all ports greater than the following port number.
   i. `lt` all ports less than the following port number.

   source-port is the source port number. The *operator1 source-port* parameters are optional.
   ipv6-destination-address is the destination IPv6 address. This address can be in one of the following formats:

   g. **`host`** *`ipv6-host-address`* to identify a single address.
   h. *`ipv6-network-prefix`*`/`*`prefix-length`* to identify an IPv6 network.
   i. **`any`** to identify all host addresses.

   operator2 is an operator used to identify the port. It can be the following:

   g. `eq` equal to a specific port number identified in the following parameter.
   h. `gt` all ports greater than the following port number.
   i. `lt` all ports less than the following port number.

   destination-port is the destination port number. The *operator2 destination-port* parameters are optional.

3. Repeat step 2 for as many filters as you feel necessary. Remember that you must have at least one command of `permit`. Otherwise, all traffic will be blocked on the interface in that direction.
4. Apply the Access-list to a specific interface in a specific direction:
   ```
   Router(config)#interface interface-type interface-number
   Router(config-if)#ipv6 traffic-filter acl-name {in|out}
   ```
   where
   interface-type and *interface-number* are the type and number of the interface
   acl-name is the ACL name that we identified in step 1.
   in|out is the direction of traffic: '`in`' for incoming traffic (going into the router through the interface) and '`out`' for outgoing from the router to outside through the interface.
5. For troubleshooting, use the following commands:
   ```
   Router#show ipv6 Access-list
   Router#show ipv6 Access-list acl-name
   Router#show ip interface interface-type interface-number
   ```

6.3.4 Reflexive Access-Lists

A reflexive ACL is an Access-list that has the capability to prevent packets of certain features from entering the interface unless these packets were in response to a request that was initiated from inside. For example, you can configure a reflexive ACL in such a way that ICMP traffic does not pass through to your internal network unless it was in response to a request sent from your internal network to the outside.

To configure the reflexive ACL, you will need to setup two ACLs: one for the incoming traffic and one for the outgoing traffic and link them together.

1. Create the first Access-list:
   ```
   Router(config)#ip Access-list extended acl-out-name
   ```
   where *acl-out-name* is the name of the ACL that will be applied to the outgoing traffic.
2. Write the filters and effect that you want to the router to watch their sessions and allow their responses with the '`reflect`' parameter.
   ```
   Router(config-ext-nacl)#{permit|deny} protocol-name source-ip-address operator1 source-port destination-ip-address operator2 destination-port reflect reflection-name
   ```
   where
   permit|deny is the effect that you want to have if the packet meets the condition. You can use permit to allow traffic or deny to block traffic.
   protocol-name the name of the protocol you want to filter. This can be `ip`, `tcp`, `udp`, etc.

source-ip-address is the source IPv4 address. This address can be in one of the following formats:

j. **host** *ipv4-host-address* to identify a single address.
k. *network-address wildcard-mask* to identify an IPv4 network.
l. **any** to identify all host addresses.

Operator1 is an operator used to identify the port. It can be the following:

j. eq equal to a specific port number identified in the following parameter.
k. gt all ports greater than the following port number.
l. lt all ports less than the following port number.

source-port is the source port number. The *operator1 source-port* parameters are optional.
destination-ip-address is the destination IPv6 address. This address can be in one of the following formats:

j. **host** *ipv6-host-address* to identify a single address.
k. *network-address wildcard-mask* to identify an IPv4 network.
l. **any** to identify all host addresses.

operator2 is an operator used to identify the port. It can be the following:

j. eq equal to a specific port number identified in the following parameter.
k. gt all ports greater than the following port number.
l. lt all ports less than the following port number.

destination-port is the destination port number. The *operator2 destination-port* parameters are optional.
reflection-name is the name of the reflection. You can use any name. The purpose of it is to be able to link it to the other Access-lists for the incoming traffic.

3. Remember that you can have on Access-list per interface per direction. Hence, you should configure all the filters that you want, not only the reflexive ones in the same named Access-list. It does not necessarily have to be done after the reflexive part. You can configure the other filters before or after the reflexive part.
4. Create the incoming traffic's ACL:
   ```
   Router(config)#ip Access-list extended acl-in-name
   ```
   where *acl-in-name* is the name of the ACL that will be applied to the incoming traffic.
5. Configure the reflexive filter:
   ```
   Router(config-ext-nacl)#evaluate reflection-name
   ```
   where *reflection-name* is the name of the reflection that you have configured in step 2.

6. Just as we explained in step 3, you can apply one ACL per interface per direction. Hence, you can add after or before step 5 all the filters you need to block and allow traffic as you see appropriate.
7. Apply the Access-lists to an interface:
   ```
   Router(config)#interface interface-type interface-number
   Router(config-if)#ip access-group acl-out-name out
   Router(config-if)#ip access-group acl-in-name in
   ```
   where
   interface-type and *interface-number* are the type and number of the interface
   acl-name is the ACL name that we identified in steps 1 and 4.
8. For troubleshooting, use the following commands:
   ```
   Router#show Access-list
   Router#show Access-list acl-name
   Router#show ip interface interface-type interface-number
   ```

6.3.5 Time-Based Access-Lists

There are some cases in which you need to apply filters in specific times rather than all the time. For example, you need to block traffic to a specific server after working hours or during weekends. This can be done using time-based Access-lists.

This type of Access-lists works by associating a time-range to the Access-list filtering command. So, you do not need to apply the whole Access-list for a specific time. Instead, you have the ability to fine-tune the Access-list to select specific commands to work in specific times.

1. Create a time-range in which we want the filtering command to be active:
   ```
   Router(config)#time-range time-range-name
   ```
   where *time-range-name* is the name of the time-range that we will create.
2. Identify the range of time:
   ```
   Router(config-time-range)#periodic days-of-week starting-time to ending-time
   ```
   where
   days-of-week part can be used to identify specific days of the week in which the range is active. We can write the days with spaces between their names or we can simple write daily.
   starting-time is the time of starting the time-range. It is written in 24-h format like 21:00.
   ending-time is the time in which the time-range ends. It is also written in 24-h format like 18:00.
3. Create a named Access-list (named Access-lists are preferred over numbered Access-lists):

`Router(config)#`**`ip Access-list extended`** *`acl-name`*

4. Configure the time-based filtering commands:
   `Router(config-ext-nacl)#{`**`permit|deny`**`}` *`protocol-name source-ip-address operator1 source-port destination-ip-address operator2 destination-port`* **`time-range`** *`time-range-name`*
   where
   permit|deny is the effect that you want to have if the packet meets the condition. You can use permit to allow traffic or deny to block traffic.
   protocol-name is the name of the protocol you want to filter. This can be `ip`, `tcp`, `udp`, etc.
   source-ip-address is the source IPv4 address. This address can be in one of the following formats:

   a. **`host`** *`ipv4-host-address`* to identify a single address.
   b. *`network-address wildcard-mask`* to identify an IPv4 network.
   c. **`any`** to identify all host addresses.

   Operator1 is an operator used to identify the port. It can be the following:

   a. `eq` equal to a specific port number identified in the following parameter.
   b. `gt` all ports greater than the following port number.
   c. `lt` all ports less than the following port number.

   source-port is the source port number. The *operator1 source-port* parameters are optional.
   destination-ip-address is the destination IPv6 address. This address can be in one of the following formats:

   a. **`host`** *`ipv6-host-address`* to identify a single address.
   b. *`network-address wildcard-mask`* to identify an IPv4 network.
   c. **`any`** to identify all host addresses.

   operator2 is an operator used to identify the port. It can be the following:

   a. `eq` equal to a specific port number identified in the following parameter.
   b. `gt` all ports greater than the following port number.
   c. `lt` all ports less than the following port number.

   destination-port is the destination port number. The *operator2 destination-port* parameters are optional.
   time-range-name is the name of the time-range that was identified in step 1.
5. You can configure other filtering commands; time-based or not within the same Access-list, before or after the command we wrote in step 4.
6. Apply the Access-list to an interface in a specific direction:
   `Router(config)#`**`interface`** *`interface-type interface-number`*
   `Router(config-if)#`**`ip access-group`** *`acl-out-name`* `{`**`in|out`**`}`

where
interface-type and *interface-number* are the type and number of the interface
acl-name is the ACL name that we identified in step 3.
in|out is the direction of traffic: `in` for incoming traffic (going into the router through the interface) and `out` for outgoing from the router to outside through the interface.

7. For troubleshooting, use the following commands:
   ```
   Router#show Access-list
   Router#show Access-list acl-name
   Router#show ip interface interface-type interface-number
   ```

6.4 How to Configure Routing Protocols Authentication on a Cisco Router

6.4.1 Configuring EIGRP Route Authentication

EIGRP route authentication provides MD5 authentication of routing updates from the EIGRP routing protocol. The MD5 keyed digest in each EIGRP packet prevents the introduction of unauthorized or false routing messages from unapproved sources. Before you can enable EIGRP route authentication, you must enable EIGRP.

The steps for setting the EIGRP route authentication are as follows:

1. Identify a keychain to be used in the authentication,
   ```
   Router(config)#key chain key-chain-name
   ```
   where *key-chain-name* is the name of the keychain that will be created.
2. Identify the key number,
   ```
   Router(config-keychain)#key key-number
   ```
   where *key-number* is the number of the key.
3. Identify the key string,
   ```
   Router(config-keychain)#key-string key-string
   ```
   where *key-string* is the key string.
4. You can, optionally, set up a period in which the key will be effective,
   ```
   Router(config-keychain)#accept-lifetime start-time [infitnite | end-time | duration]
   Router(config-keychain)#send-lifetime start-time [infitnite | end-time | duration]
   ```
   You can set a start time and either end time, or duration in seconds, or you can leave the operation infinite.
5. Enable the MD5 authentication on the EIGRP-enabled interface:

`Router(config)#`**`interface`** *`interface-type interface-number`*
`Router(config-if)#`**`ip authentication mode eigrp`** *`autonomous-system-number`* **`md5`**
where
interface-type and *interface-number* are the type and number of the interface.
autonomous-system-number the number of the autonomous system used in the EIGRP process.

6. Associate the interface's EIGRP authentication to the key-chain:
   `Router(config-if)#`**`ipv authentication key-chain eigrp`** *`autonomous-system key-chain-name`*
   where
   key-chain-name is the name of the keychain that was created in step 1.
   autonomous-system-number is the number of the autonomous system used in the EIGRP process.

6.4.2 Configuring EIGRP Route Authentication for IPv6

The steps for setting the EIGRP route authentication are as follows:

1. Identify a keychain to be used in the authentication,
   `Router(config)#`**`key chain`** *`key-chain-name`*
   where *key-chain-name* is the name of the keychain that will be created.
2. Identify the key number,
   `Router(config-keychain)#`**`key`** *`key-number`*
   where *key-number* is the number of the key.
3. Identify the key string,
   `Router(config-keychain)#`**`key-string`** *`key-string`*
   where *key-string* is the key string.
4. You can, optionally, set up a period in which the key will be effective,
   `Router(config-keychain)#`**`accept-lifetime`** *`start-time`* `[`*`infitnite | end-time | duration`*`]`
   `Router(config-keychain)#`**`send-lifetime`** *`start-time`* `[`*`infitnite | end-time | duration`*`]`
   You can set a start time and either end time, or duration in seconds, or you can leave the operation infinite.
5. Enable the MD5 authentication on the EIGRP-enabled interface:
   `Router(config)#`**`interface`** *`interface-type interface-number`*
   `Router(config-if)#`**`ipv6 authentication mode eigrp`** *`autonomous-system-number`* **`md5`**
   where
   interface-type and *interface-number* are the type and number of the interface.

autonomous-system-number is the number of the autonomous system used in the EIGRP process.

6. Associate the interface's EIGRP authentication to the key-chain:

```
Router(config-if)#ipv6 authentication key-chain eigrp
autonomous-system key-chain-name
```

where

key-chain-name is the name of the keychain that was created in step 1.

autonomous-system-number is the number of the autonomous system used in the EIGRP process.

6.4.3 Configuring BGP Peer Authentication

Peer authentication with MD5 in BGP is configured using a single command added to the neighbor BGP configuration.

```
Router(config)#router bgp autonomous-system-number
Router(config-router)#neighbor neighbor-ip-address
password bgp-password
```

where

autonomous-system-number is the number of the autonomous system used in the BGP process.

neighbor-ip-address is the IP address of the BGP neighbor.

bgp-password is the password used for BGP authentication.

The same *bgp-password* must be used on the other BGP peer router.

To check that the MD5 authentication is working, use the following command:

```
Router#show ip bgp neighbors | include Option Flags
```

6.5 How to Configure Site-to-Site VPN in Cisco Routers

When would you need this: When you want to create a secure tunnel to transfer data between two sites without the use of Virtual Private Network (VPN) concentrator or other security devices.

Special Requirements: The routers used must support IPSec (i.e., advsecurity features enabled). Most of Cisco routers do. Another need is that both sides use a static public IP address to connect to the Internet.

We will go through the steps to be done on one side, and the same steps must be repeated on the other side too. The encryption of data will depend on a shared key.

This way we will not need specialized CAs or RSA methodologies. If you have a hub-and-spoke topology refers to the note at the end of this procedure.

1. Create Internet key exchange (IKE) key policy. The policy used for our case is policy number 9, because this policy requires a pre-shared key.
   ```
   Router(config)#crypto isakmp policy 9
   Router(config-isakmp)#hash md5
   Router(config-isakmp)#authentication pre-share
   ```
2. Set up the shared key that would be used in the VPN,
   ```
   Router(config)#crypto isakmp key vpn-key address other-end-address
   ```
   where
   vpn-key is the shared key that you will use for the VPN, and remember to set the same key on the other end.
   other-end-address is the static public IP address of the other end.
3. Now, we set lifetime for the IPSec security associations,
   ```
   Router(config)#crypto ipsec security-association lifetime seconds life-time
   ```
   where *life-time* is the association lifetime in seconds. It is usually used as 86,400, which is one day.
4. Configure an extended Access-list to define the traffic that is allowed to be directed through the VPN-link.
   ```
   Router(config)#Access-list list-number permit ip source-network source-wildcard-mask destination-network destination-wildcard-mask
   ```
   where
   list-number is the Access-list number.
   source-network source-wildcard-mask is the network address and wildcard mask used to identify the source of the data allowed to use the VPN-link.
   destination-network destination-wildcard-mask is the network address and wildcard mask used to identify the destination of the data that need to pass though the VPN-link.
5. Define the transformations set that will be used for this VPN connection,
   ```
   Router(config)#crypto ipsec transform-set set-name transformation-set-1 transformation-set-2
   ```
   where
   set-name is the name of the transformations set. You can choose any name you like.
   transformation-set-1 and *transformation-set-2* is the transformation set. You can use '`esp-3des esp-md5-hmac`'. You can also use '`esp-3des esp-sha-hmac`'. Any one of these two will do the job. In general, I would recommend newer technologies such as AES and SHA-2 or SHA-3.
6. After defining all the previous things, you need to create a cryptomap that associates the Access-list to the other site and the transform set.
   ```
   Router(config)#crypto map map-name priority ipsecisakmp
   Router(config-crypto-map)#set peer other-end-address
   Router(config-crypto-map)#set transform-set set-name
   ```

```
    Router(config-crypto-map)#match address list-number
    ```
    where

    map-name is a name of your choice to the crypto map.

    priority is the priority of this map over other maps to the same destination. If this is your only crypto map give it any number, for example 10.

    other-end-address is the static public IP address of the other end.

    set-name is the name of the transformations set that we configured in step 5.

    list-number is the number of the Access-list that we created to define the traffic in step 4.

7. The last step is to bind the crypto map to the interface that connects the router to the other end.
    ```
    Router(config-if)#crypto map map-name
    ```
    where *map-name* is the name of the crypto map that we defined in step 6.
8. Now, repeat these VPN configuration steps on the other end, and remember to use the same VPN key along with the same authentication and transformation set.
9. For troubleshooting purposes, you can use the following commands:
    ```
    Router(config)#show crypto isakmp sa
    Router(config)#show crypto ipsec sa
    Router(config)#show crypto engine connections active
    Router(config)#show crypto map
    ```

6.6 How to Configure a Cisco Router as a PPTP VPN Server

When would you need this: When you want to create a secure tunnel to transfer data between a user terminals (computer, smartphone, tablet, etc.) to a central site without the use of Virtual Private Network (VPN) concentrator or other security devices.

Special Requirements: The router must be connected to the Internet with a public IP address or using Dynamic-DNS.

1. Enable the router's dial-up VPN capability:
    ```
    Router(config)#vpdn enable
    ```
2. Enable accepting dial-in connections in the VPDN group:
    ```
    Router(config)#vpdn-group group-number
    Router(config-vpdn)#accept-dialin
    Router(config-vpdn-acc-in)#protocol pptp
    ```
    where group-number is the number of the VPDN group. It is a good practice to start from 1.
3. Link the VPDN group to a virtual template:

```
Router(config-vpdn-acc-in)#virtual-template virtual-template-number
Router(config-vpdn-acc-in)#exit
```

where the *virtual-template-number* is the number used to identify the virtual template interface.

4. Identify the local IP address pool for the remote device when they connect to the router:

   ```
   Router(config)#ip local pool pool-name start-address end-address
   ```

   where

   pool-name represent the name of the address pool you will assign to the remote hosts connected to your router.

   start-address and *end-address* identify the first IP address in the pool and the last IP address in the pool. In most cases, network administrators use addresses with the internal range of IP addresses that is used in the organization. This makes the process of filtering and authorization less complicated.

5. Create the virtual template with all of its internal settings (encapsulation, authentication, etc.)

   ```
   Router(config)#interface virtual-template virtual-template-number
   Router(config-if)#encapsulation ppp
   Router(config-if)#peer default ip address pool pool-name
   Router(config-if)#ip unnumbered interface
   Router(config-if)#no keepalive
   Router(config-if)#ppp encrypt mppe encryption-type
   Router(config-if)#ppp authentication authentication-type
   ```

   The encryption-type can be either `40`, `128`, or `auto`. For better encryption, use 128 instead of auto or 40. The automatic selection might cause the use of 40-bit encryption instead of 128-bit encryption.

   In the last command, you can use `pap`, `chap`, `ms-chap`, or `ms-chap-v2` as *authentication-type*. You can combine more than one to allow the remote user to choose any type they want. As a general advice, do not use PAP.

   The *interface* should be the internal interface that is holding an IP address within the range that the remote users are going to be part of. If you want to give the remote users addresses that are not part of your internal network, you can use another interface number.

6. Create the usernames and password for the remote users to use when connecting to your router.

   ```
   Router(config)#username remote-username password remote-password
   ```

   Repeat this step for each username you want to add.

7. Check the configuration using the following commands:
   ```
   Router#show vpdn
   Router#show user
   Router#debug vpdn
   ```

6.7 How to Configure GRE Tunneling in a Cisco Router

When would you need this: When you want to create a tunnel to transfer data between two routers. GRE carries any type of Layer3 traffic through an IP network (like multicast and IPv6 traffic).

Special Requirements: None.

Although GRE tunnel is not considered a secure transfer of data (because it does not involve encryption), it is possible to combine GRE tunneling with other protocols, such as IPSec and CET encryption, to get a secure version of it.

6.7.1 GRE Configuration with no Encryption

1. On Router1, create a tunnel interface, which is a virtual interface, and assign an IP address to it:
   ```
   Router1(config)#int tunnel1
   Router1(config-if)#ip address ip-address-1 subnetmask1
   ```
   where
   ip-address-1 is the IP address of the tunnel interface.
   subnetmask1 is the subnetmask of the tunnel interface.
   Keep in mind that the *ip-address-1* is a local IP address for mapping purposes. This address needs to be a part of the local network that you want to make it able to communicate with the local network on the other side of the tunnel.
2. Set up the public IP address that will be used in the encapsulation process. This IP address is the one that will appear on the outer packet encapsulation header.
   ```
   Router1(config-if)#tunnel source public-ip-1
   ```
   where *public-ip-1* is the public IP address that the tunnel will use in encapsulation on Router1. Usually, as the router connected to the Internet, this needs to be a public IP address.
3. Set up the public IP address of the other end of the tunnel:
   ```
   Router1(config-if)#tunnel destination public-ip-2
   ```
   where *public-ip-2* is the public IP address of the other side.

4. You can also setup the MTU and the MSS:
   ```
   Router1(config-if)#ip mtu mtu-size
   Router1(config-if)#ip adjust-mss mss-size
   ```
   where
   mtu-size is the MTU size and
   mss-size is the MSS size.
5. On Router1, create a tunnel interface, which is a virtual interface, and assign an IP address to it:
   ```
   Router2(config)#int tunnel1
   Router2(config-if)#ip address ip-address-2 subnetmask2
   ```
   ip-address-2 is the IP address of the tunnel interface.
   Subnetmask2 is the subnetmask of the tunnel interface.
   Keep in mind that the *ip-address-2* is a local IP address for mapping purposes. This address needs to be a part of the local network that you want to make it able to communicate with the local network on the other side of the tunnel.
6. Set up the public IP address that will be used in the encapsulation process. This IP address is the one that will appear on the outer packet encapsulation header.
   ```
   Router2(config-if)#tunnel source public-ip-2
   ```
   where *public-ip-2* is the IP address that the tunnel will use in encapsulation. Usually, as the router connected to the Internet, this needs to be a public IP address.
7. Set up the public IP address of the other end of the tunnel:
   ```
   Router2(config-if)#tunnel destination public-ip-1
   ```
   where *public-ip-1* is the public IP address of the other side.
8. You can also set up the MTU and the MSS:
   ```
   Router2(config-if)#ip mtu mtu-size
   Router2(config-if)#ip adjust-mss mss-size
   ```
   mtu-size is the MTU size and
   mss-size is the MSS size.
   Remember to use the same values you used on Router1.
9. Set up routing between the two sides:
   ```
   Router1(config)#ip route local-network-1 subnetmask1 public-ip-2
   Router2(config)#ip route local-netwok-2 subnetmask2 public-ip-1
   ```
   where
   local-network-1 and *subnetmask1* are the network address and subnet mask of the network which Router1's local IP address is part of.
   public-ip-2 is the public IP address used on Router2.
   local-netwok-2 and *subnetmask2* are the network address and subnet mask of the network which Router2's local IP address is part of
   public-ip-1 is the public IP address used on Router1.

6.7.2 GRE Point-to-Point Configuration Over IPSec

IPSec is used here in tunneling mode to encrypt the payload and header of the GRE packets.

1. On Router1, create a tunnel interface, which is a virtual interface, and assign an IP address to it:
   ```
   Router1(config)#int tunnel1
   Router1(config-if)#ip address ip-address-1 subnetmask1
   ```
   where
   ip-address-1 is the IP address of the tunnel interface.
   subnetmask1 is the subnetmask of the tunnel interface.
   Keep in mind that the *ip-address-1* is a local IP address for mapping purposes. This address needs to be a part of the local network that you want to make it able to communicate with the local network on the other side of the tunnel.
2. Set up the public IP address that will be used in the encapsulation process. This IP address is the one that will appear on the outer packet encapsulation header.
   ```
   Router1(config-if)#tunnel source public-ip-1
   ```
   where *public-ip-1* is the public IP address that the tunnel will use in encapsulation on Router1. Usually, as the router connected to the Internet, this needs to be a public IP address.
3. Set up the public IP address of the other end of the tunnel:
   ```
   Router1(config-if)#tunnel destination public-ip-2
   ```
   where *public-ip-2* is the public IP address of the other side.
4. You can also set up the MTU and the MSS:
   ```
   Router1(config-if)#ip mtu mtu-size
   Router1(config-if)#ip adjust-mss mss-size
   ```
   where
   mtu-size is the MTU size and
   mss-size is the MSS size.
5. On Router1, create a tunnel interface, which is a virtual interface, and assign an IP address to it:
   ```
   Router2(config)#int tunnel1
   Router2(config-if)#ip address ip-address-2 subnetmask2
   ```
   ip-address-2 is the IP address of the tunnel interface.
   Subnetmask2 is the subnetmask of the tunnel interface.
   Keep in mind that the *ip-address-2* is a local IP address for mapping purposes. This address needs to be a part of the local network that you want to make it able to communicate with the local network on the other side of the tunnel.
6. Set up the public IP address that will be used in the encapsulation process. This IP address is the one that will appear on the outer packet encapsulation header.
   ```
   Router2(config-if)#tunnel source public-ip-2
   ```

where *public-ip-2* is the IP address that the tunnel will use in encapsulation. Usually, as the router connected to the Internet, this needs to be a public IP address.

7. Set up the public IP address of the other end of the tunnel:
   ```
   Router2(config-if)#tunnel destination public-ip-1
   ```
   where *public-ip-1* is the public IP address of the other side.
8. You can also set up the MTU and the MSS:
   ```
   Router2(config-if)#ip mtu mtu-size
   Router2(config-if)#ip adjust-mss mss-size
   ```
   mtu-size is the MTU size and
   mss-size is the MSS size.
   Remember to use the same values you used on Router1.
9. Set up routing between the two sides:
   ```
   Router1(config)#ip route local-network-1 subnetmask1 public-ip-2
   Router2(config)#ip route local-netwok-2 subnetmask2 public-ip-1
   ```
   where
   local-network-1 and *subnetmask1* are the network address and subnet mask of the network which Router1's local IP address is part of.
   public-ip-2 is the public IP address used on Router2.
   local-netwok-2 and *subnetmask2* are the network address and subnet mask of the network which Router2's local IP address is part of.
   public-ip-1 is the public IP address used on Router1.
10. Create an Access-list to identify which traffic to encrypt:
    ```
    Router1(config)#Access-list acl-number permit gre host public-ip-1 host public-ip-2
    ```
    where
    acl-number is the number of the Access-list which can be any number from 100 to 199.
    public-ip-1 is the public IP address used on Router1.
    public-ip-2 is the public IP address used on Router2.
11. Configure the ISAKMP policy, pre-shared keys, and the transform set:
    ```
    Router1(config)#crypto isakmp policy 1
    Router1(config)#authentication pre-share
    Router1(config)#crypto isakmp key pre-shared-key address public-ip-2
    Router1(config)#crypto ipsec transform-set transform-set-name esp-3des esp-md5-hmac
    ```
    where
    pre-shared-key is the pre-shared key to be used in encrypting the traffic. The same key must be used on both sides.
    public-ip-2 is the public IP address used on Router2.
    transform-set-name is the name of the transform set.

12. Create and bind a cryptomap to the traffic identified by the Access-list from step 10.

    ```
    Router1(config)#crypto map map-name 10 ipsec-isakmp
    Router1(config-map)#set peer public-ip-2
    Router1(config-map)#set transform-set transform-set-name
    Router1(config-map)#match address acl-number
    ```

    where

    map-name is the name of the crypto map.

    public-ip-2 is the public IP address used on Router2.

    transform-set-name is the name of the transform set identified in step 11.

    acl-number is the number of the Access-list identified in step 10.
13. Apply the crypto map to the physical interface on which the tunnel's traffic will be forwarded.

    ```
    Router1(config)#interface interface-type interface-number
    Router1(config-if)#crypto map map-name
    Router1(config-if)#interface Tunnel1
    Router1(config-if)#crypto map map-name
    ```

    where

    interface-type and *interface-number* are the type and number of the physical interface that will carry the GRE tunnel data.

    map-name is the name of the crypto map.
14. Repeat steps 10, 11, 12, and 13 on Router2 but remember to change *public-ip-2* to *public-ip-1* in all of these steps.
15. For verification purposes, use the following command:

    ```
    Router(config)#show crypto session
    ```

6.8 How to Configure AAA Service on a Cisco Router

When would you need this: When you want to authenticate users using a remote TACACS+ or RADIUS server.

Special Requirements: None.

Using a centralized authentication service is generally better than using decentralized (router-level) authentication. I will give you one example that will convince you of the importance of AAA; when you have a group of network administrators but no AAA service, all those administrators must be aware of all of the routers' passwords. If one administrator leaves his/her job, you will have to change all the administrator passwords for all routers. However, if you are using a centralized authentications service, you can simply disable the account that you want to deny access to.

Remember that in this configuration, the router itself will not become an AAA server. You will have to set up the server elsewhere using either RADIUS technology or TACACS+.

6.8.1 RADIUS Configuration

1. Identify the address and key used by the RADIUS server:

   ```
   Router(config)#radius-server host server-ip-address key key-number key-value
   ```

   where

   server-ip-address is the IP address of the RADIUS server. This address can be an IPv4 or IPv6 address.

   key-number is the key number (usually this is a number between 0 and 7).

   key-value is the value of the key.

   At the end of this simple step, you can stop and everything should work properly.
2. If you have more than one server, you can create server groups so that the router can use a group of servers instead of one.

   ```
   Router(config)#aaa group server radius group-name
   Router(config-radius)#server server-ip-address
   ```

   where

   group-name is the name of the server group

   server-ip-address is the IP address of the RADIUS server. This address can be an IPv4 or IPv6 address.

   Repeat the last command for each different RADIUS server you want to add to the group.
3. Configure the router to use a specific user group for VTY:

   ```
   Router(config)#line vty 0 4
   Router(config-line)#login authentication group-name
   ```

   where the *group-name* is the user group name. The same name identified in step 1.
4. For troubleshooting, use the following commands:

   ```
   Router#show radius-server
   Router#show radius-server groups
   ```

6.8.2 TACACS+ Configuration

1. Enable AAA with TACACS+:

   ```
   Router(config)#aaa new-model
   Router(config)#aaa authentication login group-name group tacacs+
   ```

where the *group-name* is the group name of administrators allowed to login as identified at the TACACS+server.

2. Identify the server information:

```
Router(config)#tacacs server server-name
Router(config-tacacs)#address address-type ip-address
Router(config-tacacs)#key server-key
```

where
server-name is a name that you want to assign to the server.
address-type is IPv4 or IPv6.
ip-address is the IP address of the server.
server-key is the secret key configured on the server.

3. Configure the router to use a specific user group for VTY:

```
Router(config)#line vty 0 4
Router(config-line)#login authentication group-name
```

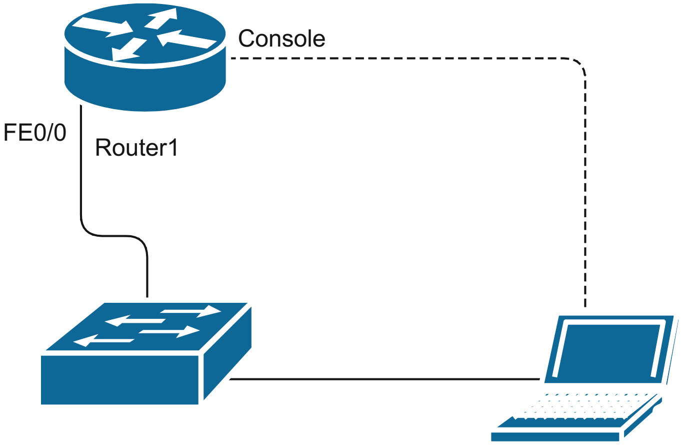
where the *group-name* is the user group name. The same name identified in step 1.

4. For troubleshooting, you can use the following command:

```
Router#show tacacs
```

6.9 Training Scenarios

Scenario 6.1



Connect the simple network shown in the figure above and configure the following settings:

1. Router 1's basic configuration:

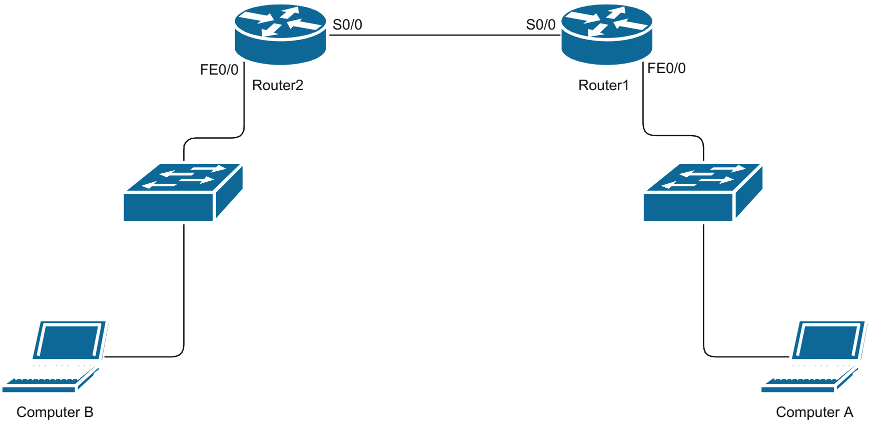
Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.1/24

2. Computer A's settings:

Device	Parameter	Value
Computer A	IP Address	172.16.0.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.0.1
	DNS Server 1	172.16.0.1
	DNS Server 2	172.16.0.1

3. On the router, do a '`show run`' command. You should see the secret password encrypted, but neither the VTY nor the console passwords would be encrypted.
4. Enable the service '`password-encryption`'.
5. Do another '`show run`' command. You should see the console and VTY passwords encrypted.
6. Disable the service '`password-encryption`'.
7. Once more do a '`show run`', and you should see that the passwords are still encrypted.

Scenario 6.2



Connect the network shown in the figure above and configure the following settings:

1. Basic configuration of the routers:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	10.0.0.254/24
	Interface S0/0 IP Address	10.1.1.1/30
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	10.0.1.254/24
	Interface S0/0 IP Address	10.1.1.2/30

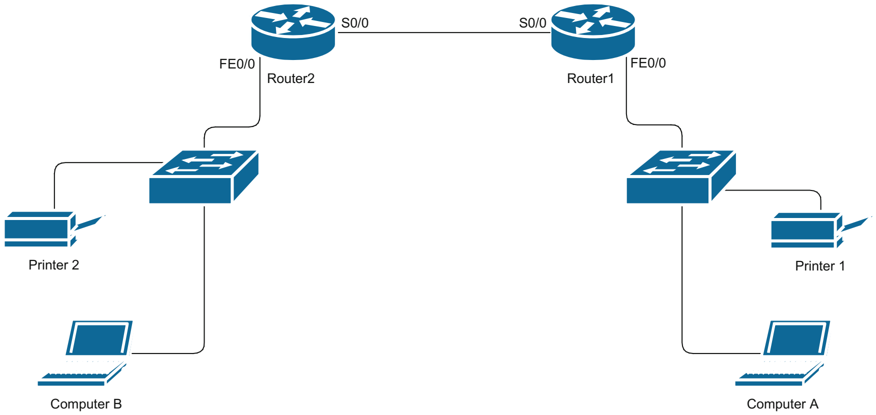
2. Computers' configuration:

Device	Parameter	Value
Computer A	IP Address	10.0.0.1/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.0.254
	DNS Server 1	10.0.0.254
	DNS Server 2	10.0.0.254
Computer B	IP Address	10.0.1.1/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.1.254
	DNS Server 1	10.0.1.254
	DNS Server 2	10.0.1.254

3. On Router 1, create a default route to send all traffic through the exit interface S0/0.
4. On Router 2, also create a default route to send all traffic through the exit interface S0/0.
5. Create a standard Access-list to block traffic from the LAN connected to Router 2 to Computer A. Since standard Access-lists filter traffic based on source IP address only, we need to create the ACL as close as possible to the destination. Thus, we will create the ACL on Router 1.
6. Before applying the Access-list, PING from Computer B to Computer A. The PING should be successful.

7. Apply the ACL on Router 1 on outgoing traffic on interface FE0/0.
8. PING from Computer B to Computer A. The PING will fail.

Scenario 6.3



Connect the network shown in the figure above and configure the following settings:

1. Basic configuration of the routers:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	10.0.0.254/24
	Interface S0/0 IP Address	10.1.1.1/30
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	10.0.1.254/24
	Interface S0/0 IP Address	10.1.1.2/30

2. Computers' configuration:

Device	Parameter	Value
Computer A	IP Address	10.0.0.1/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.0.254
	DNS Server 1	10.0.0.254
	DNS Server 2	10.0.0.254
Computer B	IP Address	10.0.1.1/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.1.254
	DNS Server 1	10.0.1.254
	DNS Server 2	10.0.1.254

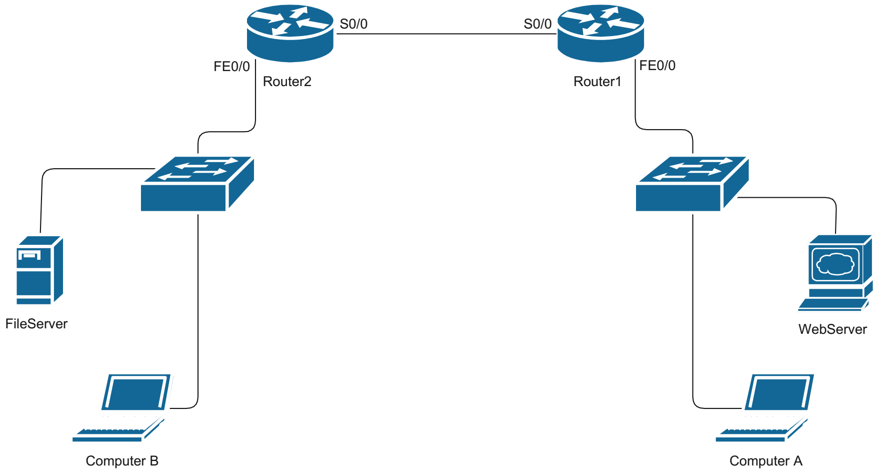
3. Printers' configuration:

Device	Parameter	Value
Printer 1	IP Address	10.0.0.10/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.0.254
	DNS Server 1	10.0.0.254
	DNS Server 2	10.0.0.254
Printer 2	IP Address	10.0.1.10/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.1.254
	DNS Server 1	10.0.1.254
	DNS Server 2	10.0.1.254

4. On Router 1, create a default route to send all traffic through the exit interface S0/0.
5. On Router 2, also create a default route to send all traffic through the exit interface S0/0.
6. Create an extended ACL to block traffic from Router 2's LAN to Printer 1. Since extended ACLs have the capability to filter based on source and destination IP addresses, we can place the list as close as possible to the source, to reduce network traffic. Hence, we will create the ACL on Router 2.
7. For testing, PING from Computer B to Printer 1. The PING should be successful.
8. Apply the ACL to the incoming traffic on interface FE0/0 in Router 2.
9. Redo the PING from Computer B to Printer 1. It should fail now.
10. Create another extended ACL to block traffic from Router 1's LAN to Printer 2.
11. For testing, PING from Computer A to Printer 2. The PING should be successful.

12. Apply the ACL to the incoming traffic on interface FE0/0 in Router 1.
13. Redo the PING from Computer A to Printer 2. It should fail now.

Scenario 6.4



Connect the network shown in the figure above and configure the following settings:

1. Basic configuration of the routers:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	10.0.0.254/24
	Interface S0/0 IP Address	10.1.1.1/30
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	10.0.1.254/24
	Interface S0/0 IP Address	10.1.1.2/30

2. Computers' configuration:

Device	Parameter	Value
Computer A	IP Address	10.0.0.1/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.0.254
	DNS Server 1	10.0.0.254
	DNS Server 2	10.0.0.254
Computer B	IP Address	10.0.1.1/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.1.254
	DNS Server 1	10.0.1.254
	DNS Server 2	10.0.1.254

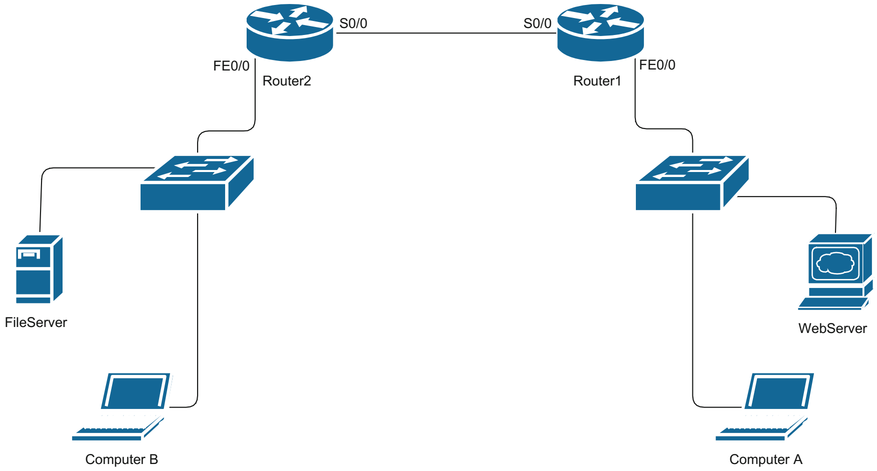
3. Servers' configuration:

Device	Parameter	Value
Web Server	IP Address	10.0.0.10/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.0.254
	DNS Server 1	10.0.0.254
	DNS Server 2	10.0.0.254
File Server	IP Address	10.0.1.10/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.1.254
	DNS Server 1	10.0.1.254
	DNS Server 2	10.0.1.254

4. On Router 1, create a default route to send all traffic through the exit interface S0/0.
5. On Router 2, also create a default route to send all traffic through the exit interface S0/0.
6. Create an extended ACL to block Telnet traffic from Router 2's LAN to the Web Server. Since extended ACLs have the capability to filter based on source and destination IP addresses, we can place the list as close as possible to the source, to reduce network traffic. Hence, we will create the ACL on Router 2.
7. For testing, on Computer B, open a Webpage on the Web Server. The page should be displayed successfully.
8. Apply the ACL to the incoming traffic on interface FE0/0 in Router 2.
9. On Computer B, open a Webpage on the Web Server. It should fail now.
10. Create another extended ACL to block ICMP traffic from Router 1's LAN to the File Server.

11. For testing, PING from Computer A to the File Server. The PING should be successful.
12. Apply the ACL to the incoming traffic on interface FE0/0 in Router 1.
13. Redo the PING from Computer A to the File Server. It should fail now.

Scenario 6.5



Connect the network shown in the figure above and configure the following settings:

1. Basic configuration of the routers:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	10.0.0.254/24
	Interface S0/0 IP Address	10.1.1.1/30
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	10.0.1.254/24
	Interface S0/0 IP Address	10.1.1.2/30

2. Computers' configuration:

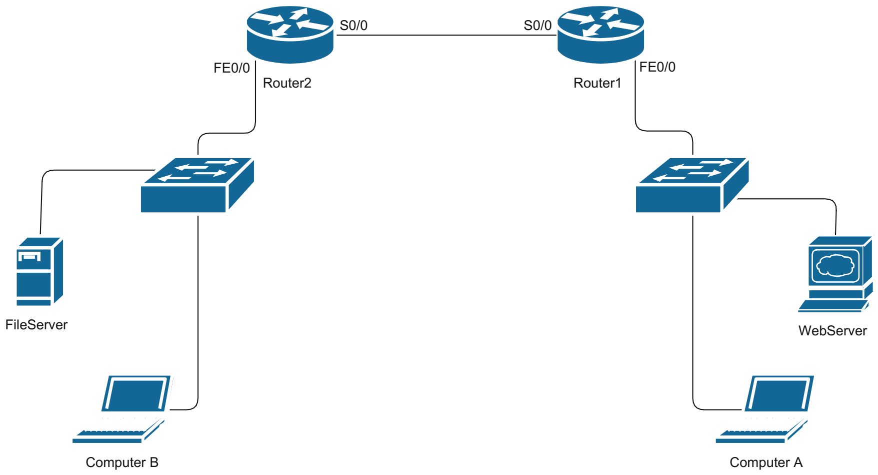
Device	Parameter	Value
Computer A	IP Address	10.0.0.1/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.0.254
	DNS Server 1	10.0.0.254
	DNS Server 2	10.0.0.254
Computer B	IP Address	10.0.1.1/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.1.254
	DNS Server 1	10.0.1.254
	DNS Server 2	10.0.1.254

3. Servers' configuration:

Device	Parameter	Value
Web Server	IP Address	10.0.0.10/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.0.254
	DNS Server 1	10.0.0.254
	DNS Server 2	10.0.0.254
File Server	IP Address	10.0.1.10/24
	Subnet Mask	255.255.255.0
	Default Gateway	10.0.1.254
	DNS Server 1	10.0.1.254
	DNS Server 2	10.0.1.254

4. On Router 1, create a default route to send all traffic through the exit interface S0/0.
5. On Router 2, also create a default route to send all traffic through the exit interface S0/0.
6. Create a named extended ACL to block Telnet traffic from Router 2's LAN to the Web Server. Since extended ACLs have the capability to filter based on source and destination IP addresses, we can place the list as close as possible to the source, to reduce network traffic. Hence, we will create the ACL on Router 2.
7. For testing, on Computer B, open a webpage on the Web Server. The page should be displayed successfully.
8. Apply the ACL to the incoming traffic on interface FE0/0 in Router 2.
9. Redo the PING from Computer B to Printer 1. It should fail now.
10. Add another line to the ACL (before the `permit any any` line) to block all traffic between Computer A and Computer B.
11. For testing, PING from Computer A to Computer B. The PING should be unsuccessful.

Scenario 6.6



Connect the network shown in the figure above and configure the following settings:

1. Basic configuration of the routers:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IPv6 Address	2001::1/64
	Interface S0/0 IPv6 Address	2002::1/64
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IPv6 Address	2003::1/64
	Interface S0/0 IPv6 Address	2002::2/64

2. Computers' configuration:

Device	Parameter	Value
Computer A	IP Address	2001::10
	Prefix Length	64
	Default Gateway	2001::1
	DNS Server 1	2001::1
	DNS Server 2	2001::1
Computer B	IP Address	2003::10
	Prefix Length	64
	Default Gateway	2003::1
	DNS Server 1	2003::1
	DNS Server 2	2003::1

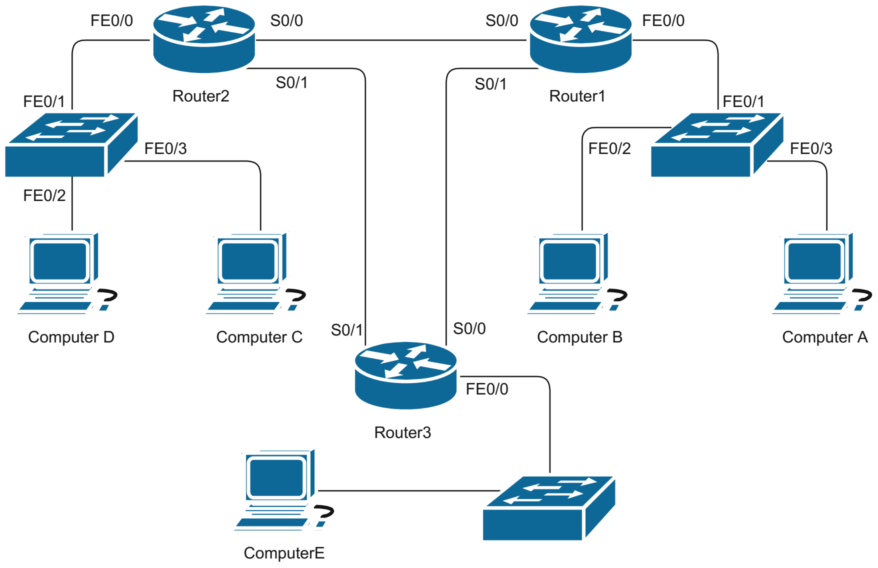
3. Servers' configuration:

Device	Parameter	Value
Web Server	IP Address	2001::100
	Prefix Length	64
	Default Gateway	2001::1
	DNS Server 1	2001::1
	DNS Server 2	2001::1
File Server	IP Address	2003::100
	Prefix Length	64
	Default Gateway	2003::1
	DNS Server 1	2003::1
	DNS Server 2	2003::1

4. On Router 1, create a default route to send all traffic through the exit interface S0/0.
5. On Router 2, also create a default route to send all traffic through the exit interface S0/0.
6. Create an extended ACL to block Telnet traffic from Router 2's LAN to the Web Server. Since extended ACLs have the capability to filter based on source and destination IP addresses, we can place the list as close as possible to the source, to reduce network traffic. Hence, we will create the ACL on Router 2.
7. For testing, on Computer B, open a Webpage on the Web Server. The page should be displayed successfully.
8. Apply the ACL to the incoming traffic on interface FE0/0 in Router 2.
9. On Computer B, open a Webpage on the Web Server. It should fail now.
10. Create another extended ACL to block ICMP traffic from Router 1's LAN to the File Server.

11. For testing, PING from Computer A to the File Server. The PING should be successful.
12. Apply the ACL to the incoming traffic on interface FE0/0 in Router 1.
13. Redo the PING from Computer A to the File Server. It should fail now.

Scenario 6.7



Connect the network shown in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	192.16.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30
	Interface S0/1 IP Address	10.0.1.1/30

2. On Router 2, do the basic configuration:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	192.16.1.254/24
	Interface S0/0 IP Address	10.0.0.2/30
	Interface S0/1 IP Address	10.0.2.1/30

3. On Router 3, do the basic configuration:

Device	Parameter	Value
Router 3	Hostname	Router3
	Console Password	Cisco3Console
	Secret Password	Cisco3
	VTY Password	Cisco3VTY
	Interface FE0/0 IP Address	192.16.2.254/24
	Interface S0/0 IP Address	10.0.1.2/30
	Interface S0/1 IP Address	10.0.2.2/30

4. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IP Address	192.16.0.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.0.254
	DNS Server 1	192.16.0.254
	DNS Server 2	192.16.0.254
Computer B	IP Address	192.16.0.2
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.0.254
	DNS Server 1	192.16.0.254
	DNS Server 2	192.16.0.254
Computer C	IP Address	192.16.1.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254
Computer D	IP Address	192.16.1.2
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.1.254
	DNS Server 1	192.16.1.254
	DNS Server 2	192.16.1.254

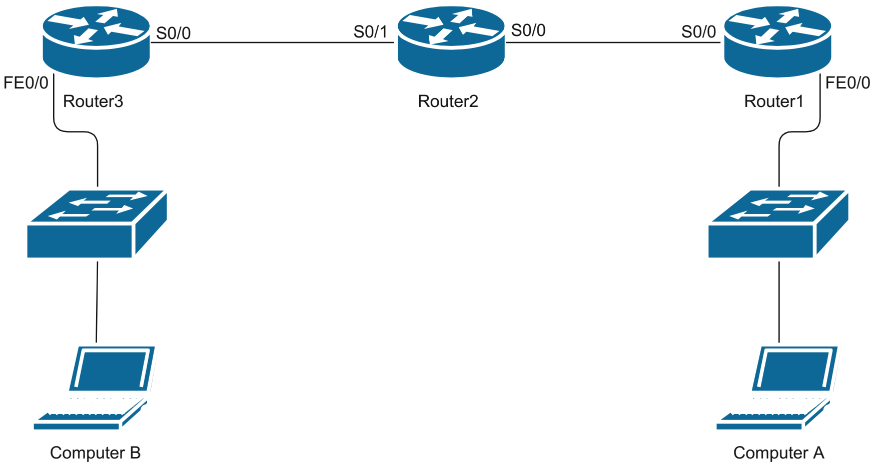
(continued)

(continued)

Device	Parameter	Value
Computer E	IP Address	192.16.2.1
	Subnet Mask	255.255.255.0
	Default Gateway	192.16.2.254
	DNS Server 1	192.16.2.254
	DNS Server 2	192.16.2.254

5. Configure EIGRP dynamic routing on Router 1 with the advertised networks 10.0.0.0 and 192.16.0.0 in autonomous system number 10.
6. Configure EIGRP dynamic routing on Router 2 with the advertised networks 10.0.0.0 and 192.16.1.0 in autonomous system number 10.
7. Configure EIGRP dynamic routing on Router 3 with the advertised networks 10.0.0.0 and 192.16.2.0 in autonomous system number 10.
8. For testing, PING from Computer A to C and D and show the routing tables in both routers.
9. Configure EIGRP route authentication between Routers 1 and 2 using the key string CiscoAuth.
10. Enable the authentication on interfaces S0/0 in Routers 1 and 2.
11. Do not configure authentication on Router 3.
12. After convergence, check to see if the network 192.168.2.0 is shown in the routing table of Routers 1 and 2.

Scenario 6.8



Connect the network shown in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30

2. On Router 2, do the basic configuration:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface S0/0 IP Address	10.0.0.2/30
	Interface S0/1 IP Address	10.0.1.1/30

3. On Router 3, do the basic configuration:

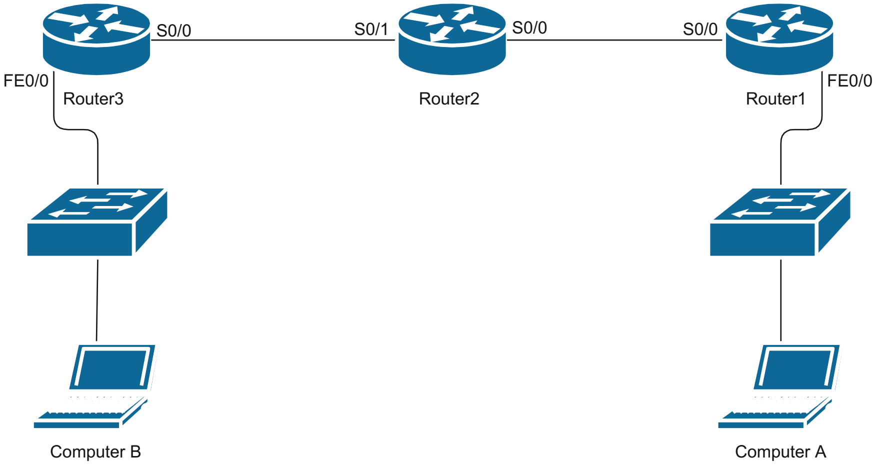
Device	Parameter	Value
Router 3	Hostname	Router3
	Console Password	Cisco3Console
	Secret Password	Cisco3
	VTY Password	Cisco3VTY
	Interface FE0/0 IP Address	172.16.1.254/24
	Interface S0/0 IP Address	10.0.1.2/30

4. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IP Address	172.16.0.1
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254
Computer B	IP Address	172.16.1.2
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.1.254
	DNS Server 1	172.16.1.254
	DNS Server 2	172.16.1.254

5. Configure EIGRP dynamic routing on Router 1 with the advertised networks 10.0.0.0 and 172.16.0.0 in autonomous system number 10.
6. Configure EIGRP dynamic routing on Router 2 with the advertised network 10.0.0.0 in autonomous system number 10.
7. Configure EIGRP dynamic routing on Router 3 with the advertised networks 10.0.0.0 and 172.16.1.0 in autonomous system number 10.
8. For testing, PING from Computer A to C and D and show the routing tables in both routers.
9. Another test would be using Router 2 to capture packets flowing from using the 'Embedded Packet Capture' functionality. This functionality will enable you to capture the packets and save them as a small PCAP file that can be extracted from the router and read using software like WireShark.
   Once you capture the packets and read their content, you can easily see that the content is not encrypted.
10. Set up a site-to-site VPN between Routers 1 and 3 using the following settings:
    VPN key: CiscoKey@123
    Transformation Set name: CiscoTranform
    Transformation Set: esp-3des esp-sha-hmac
    Crypto map name: MyCryptoMap
    Crypto map priority: 7
11. Redo the capturing of packets done in step 9. The data in all packets flowing from Computer A to Computer B should be encrypted.

Scenario 6.9



Connect the network shown in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.254/24
	Interface S0/0 IP Address	10.0.0.1/30

2. On Router 2, do the basic configuration:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface S0/0 IP Address	10.0.0.2/30
	Interface S0/1 IP Address	10.0.1.1/30

3. On Router 3, do the basic configuration:

Device	Parameter	Value
Router 3	Hostname	Router3
	Console Password	Cisco3Console
	Secret Password	Cisco3
	VTY Password	Cisco3VTY
	Interface FE0/0 IP Address	172.16.1.254/24
	Interface S0/0 IP Address	10.0.1.2/30

4. On the computers, change the following settings:

Device	Parameter	Value
Computer A	IP Address	172.16.0.1
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

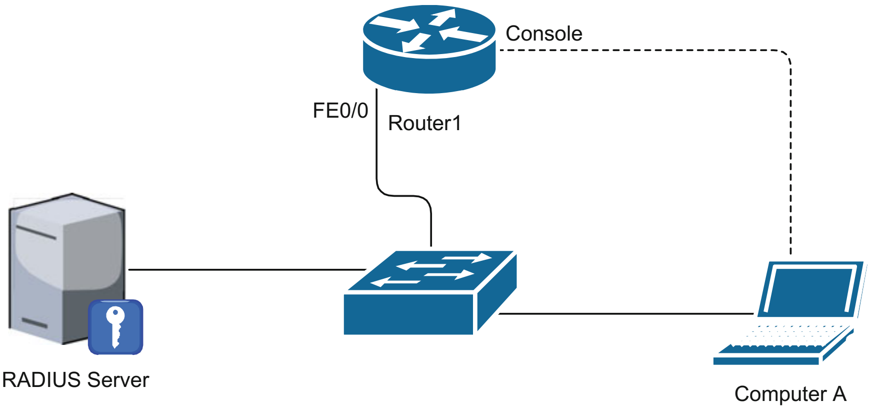
(continued)

(continued)

Device	Parameter	Value
Computer B	IP Address	172.16.1.2
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.1.254
	DNS Server 1	172.16.1.254
	DNS Server 2	172.16.1.254

5. Configure EIGRP dynamic routing on Router 1 with the advertised networks 10.0.0.0 and 172.16.0.0 in autonomous system number 10.
6. Configure EIGRP dynamic routing on Router 2 with the advertised network 10.0.0.0 in autonomous system number 10.
7. Configure EIGRP dynamic routing on Router 3 with the advertised networks 10.0.0.0 and 172.16.1.0 in autonomous system number 10.
8. For testing, PING from Computer A to C and D and show the routing tables in both routers.
9. Another test would be using Router 2 to capture packets flowing from using the 'Embedded Packet Capture' functionality. This functionality will enable you to capture the packets and save them as a small PCAP file that can be extracted from the router and read using software like WireShark.
   Once you capture the packets and read their content, you can easily see that the content is not encrypted.
10. Set up a GRE tunnel with IPSec between Routers 1 and 3.
11. Redo the capturing of packets done in step 9. The data in all packets flowing from Computer A to Computer B should be encrypted.

Scenario 6.10



Connect the simple network shown in the figure above and configure the following settings:

1. Router 1's basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	Interface FE0/0 IP Address	172.16.0.1/24

2. Computer A's settings:

Device	Parameter	Value
Computer A	IP Address	172.16.0.10
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.0.1
	DNS Server 1	172.16.0.1
	DNS Server 2	172.16.0.1

3. RADIUS server settings:

Device	Parameter	Value
RADIUS server	IP Address	172.16.0.100
	Subnet Mask	255.255.255.0
	Default Gateway	172.16.0.1
	DNS Server 1	172.16.0.1
	DNS Server 2	172.16.0.1

4. Configure the router to communicate with the RADIUS server by entering the server's IP address along with the key number and value. The key number and value to be obtained from the RADIUS server settings.
5. On the router, create an AAA group named MyRADIUSGroup and add to it the server's IP address.
6. To test the RADIUS server setup, configure the router to do VTY security authentication using the server group named MyRADIUSGroup. For this task, use the command **`login authentication`** *`group-name`*.

Chapter 7
Cisco Router Management

Keywords Cisco · Router · Cisco IOS · IOS · IOS upgrade · Magic number · TFTP · Flash · Rommon

7.1 Hints and Tips Before Upgrading the IOS of a Cisco Router

Upgrading your router's IOS is a critical operation. You need to be careful and cautious with every command you write. Take a look on these hints and tips before you start upgrading.

1. Before considering upgrading, evaluate the real need to a new IOS. If the router's current IOS covers all the jobs that you need the router to do, no upgrade is needed. Upgrade is usually necessary when you are adding new hardware, the router is not capable of handling what you want, or there is a problem with the old IOS. Sometimes there appear to be some security glitches in the IOS, so you might need to upgrade even if the router is performing smoothly.
2. To see the contents of the flash and check for the available space, use the following command,
   `Router#`**`show flash`**
   if your router has PCMCIA flash, use this command instead,
   `Router#`**`show slot0`**:
   and
   `Router#`**`show slot1`**:
3. If the space is not enough for the old and new copies of IOS together, you will have to erase the old one. Do not do that manually using '`delete flash:` *`ios-file-name.bin`*'. Once you start copying the new IOS, you will be asked to erase or keep the old contents of the flash. If you have enough space for both copies, do not erase the flash.
4. If the flash of your router is class B and have more than one bank, you can partition the flash. Partitioning the flash is useful in any copying operations

M.M. Alani, *Guide to Cisco Routers Configuration*,
DOI 10.1007/978-3-319-54630-8_7

because the router would be able to hold and maintain two different copies of IOS files. Partitioning protects you from the risk of erasing the old copy of IOS accidentally while upgrading. A procedure for flash partitioning is in Sect. 7.9.

5. Do not change the IOS file name; neither the old one nor the new one. You must have full understanding of the IOS file name conventions. You can find a brief description of the meanings of the IOS file name in Sect. 7.2.
6. It is always safer to do the upgrade through TFTP server's and not through XMODEM. TFTP server's software is easy to master and many distributions of TFTP server's are available for free.
7. When upgrading through HyperTerminal (XMODEM), do not reload the router before the whole copying process is complete.
8. If you have more than one IOS file on the flash and you do not know which one is currently loaded, use the '`show version`' command to find the name of the loaded IOS file.

7.2 Understanding the IOS File Name Convention

Before planning an upgrade or install of an IOS file, you will need to understand the meaning of the name of each IOS file.

Although Cisco has switched to a different way of handling IOS, it is still important to understanding the different versions of IOS files because a lot of the devices you will work on will be using the old naming convention for some time.

The old IOS file name is usually similar to this form:

`platform-featureset-format.aaa-bb``.bin`

1. The platform part identifies the series of the platform of the device. For example:
   `c1005`—For 1005 platform
   `c1600`—For 1600 platform
   `c1700`—For 1700, 1720, and 1750 platforms
   `c2500`—For 25xx, 3xxx, 5100, and AO (11.2 and later only) platforms
   `c2600`—For 2600 platform
   `c2800`—For Catalyst 2800 platform
   `c2900`—For 2910 and 2950 platforms
   `c3620`—For 3620 platform
   `c3640`—For 3640 platform
   `c4000`—For 4000 platform (11.2 and later only)
   `c4500`—For 4500 and 4700 platforms
2. The *`featureset`* part identifies the feature set. For example,
   `b` For Apple talk support
   `boot` For boot image
   `c` For CommServer lite (CiscoPro)
   `drag` For IOS-based diagnostic image
   `g` For ISDN subset (SNMP, IP, Bridging, ISDN, PPP, IPX, and AppleTalk)

`i` For IP subset (SNMP, IP, Bridging, WAN, Remote Node, and Terminal Services)
`n` For IPX support
`q` For asynchronous support
`t` For Telco return (12.0)
`y` For reduced IP (SNMP, IP RIP/IGRP/EIGRP, Bridging, ISDN, and PPP) (c1003 or c1004)
`z` For managed modems
`40` For 40 bit encryption
`50` For 50 bit encryption

3. The format part identifies the location where this IOS would operate
   `f` For flash
   `m` For RAM
   `r` For ROM
   `l` For the image will be relocated at run time
   The file might also be compressed. The following letters denote the compression type,
   `z` For zip compression
   `x` For mzip compression
   `w` For 'STAC' compression
4. aaa-bb represent the version of the IOS. It is usually read like this '`Version aa.a(bb)`'. The last part of the IOS file name might contain letters such as T (new feature release identifier), S (individual release number), or XR (modular packages).

Cisco follows a packaging model that provides a wide selection of feature sets for the new IOS files. These feature sets are:

a. `ipbase`—for basic IP features
b. `ipvoice`—VoIP support
c. `advsecurity`—advanced security feature
d. `spservices`—service provider services
e. `entbase`—basic enterprise services
f. `advipservices`—advanced IP services
g. `entservices`—enterprise services
h. `adventerprise`—advanced enterprise services

These feature sets have an inheritance structure such that each feature set contains all the features of previous sets with additions. For example, `advsecurity` feature set contains all features from `ipvoice` and `ipbase` and adds more security features to them.

7.3 How to Back up and Restore the Configuration of a Cisco Router

When would you need this: When you plan to implement something new in the configuration, or when you need to copy the configuration from one router to the other, or for regular backups.

Special Requirements: None.

7.3.1 TFTP

Before starting the procedure of configuration backup or restore, you will need to install TFTP server's software on a PC connected to the router Ethernet interface.

There are many free downloadable TFTP server's software on the Internet; however, our recommendation is TFTPd or Free TFTP server's.

Afterward, you make sure to direct the TFTP server's software to the folder that you want to contain the backups, and that the TFTP server's has enough free space to contain the backups.

1. Create a console connection with the default settings (9600 baud, 8 databits, 0 parity bits, 1 stop bit, no flow control).
2. Check the connectivity between the router and the TFTP server's with the 'ping' command in the privileged mode.
3. Start copying the configuration to the TFTP server's:

   ```
   Router#copy run tftp
   ```

   Or

   ```
   Router#copy start tftp
   ```

   Then you will be asked for the IP address of the TFTP server's

   ```
   Address or name of remote host []? tftp-server-address
   ```

   Afterward, you will be asked for a destination file name to be saved on the TFTP server's.

   ```
   Destination filename [Router-config]? backup_cfg_for_routerX
   ```

   It is better to choose a descriptive name so you would not mix the different configuration files.

   Now you will see the progress of the operation.

   ```
   !!
   nnn bytes copied in t.tt secs (rr bytes/sec)
   ```

   The configuration file is usually copied quickly because it is not more than few kilobytes.

   The backup procedure is now over. You can open the file copied to the TFTP server's with the text editor and view or modify it.
4. The restore procedure is done by replacing step 3 of the previous procedure with the following:

```
Router#copy tftp run
```

Or

```
Router#copy tftp start
```

Now you will be asked to provide the TFTP server's IP address

```
Address or name of remote host []? tftp-server-address
```

Then you will be asked for the source file name

```
Source filename []?backup_cfg_for_routerX
Destination filename [running-config]?
\\\ or [startupconfig]
Accessing tftp://tftp-server-address/backup_cfg_for_routerX...
Loading backup_cfg_for_routerX from tftp-server-address (via FastEthernet0/0): !
[OK - bbbb bytes]
nnnn bytes copied in t.tt secs (rrr bytes/sec)
```

It is advised that you remove any configuration lines containing 'AAA' commands from the backup file before restoring, so you would not have any security problems accessing the router. You can do that with any text editor.

There are two other ways to back up and restore the configuration: FTP and the HyperTerminal.

7.3.2 FTP

You can use FTP to back up and restore the configuration by doing the following:

1. Give the router username and password to use for FTP access:
   ```
   Router(config)#ip ftp username ftp-username
   Router(config)#ip ftp password ftp-password
   ```
2. Use the following commands for copying the configuration to and from the FTP server:
   ```
   Router#copy run ftp or copy start ftp
   ```
   And
   ```
   Router#copy ftp run or copy ftp start
   ```
   And you will have to give the same info given in step 3 of the previous procedure to complete the transfers.

7.3.3 HyperTerminal

If you do not have TFTP or FTP server's around, you can use the good old

HyperTerminal to back up and restore the configuration by doing the following steps:

1. Create a console connection with the default settings (9600 baud, 8 databits, 0 parity bits, 1 stop bit, no flow control).
2. Issue the following command:

   ```
   Router#terminal length 0
   ```

   This command will cause the show commands results to be displayed continuously without pagination.
3. On the HyperTerminal menu, select 'Transfer', and from the transfer menu, select 'Capture Text'. The Capture Text window will appear.
4. Choose a name for the configuration file to be saved (e.g., configuration.txt) and click Start.
5. On the router, issue the command:

   `Router#`**`show run`** or **`show start`**

   depending on the configuration you want to back up
6. After you see the whole configuration displayed, on the HyperTerminal menu, go to the 'Transfer' menu and select the 'Capture Text' submenu and select 'Stop' to end the screen capture.

   This concludes the backup. You may also edit the file that you have saved to erase the lines containing '`AAA`' commands to avoid access and security problems that may be caused by the restore operation.

   The restore procedure goes as the following:

1. Open the configuration backup file with a text editor and select all the text by pressing Ctrl-A key combination. Now choose 'Copy' from the Edit menu or simply press Ctrl-C.
2. Go to the HyperTerminal window that is connecting you to the router you want to perform the restore on. Afterward, go to the privileged mode.
3. From the HyperTerminal menu, open the 'Edit' menu and select 'Paste to Host'.
4. Check the configuration by '`show run`' command. If everything sounds fine, use the '`copy run start`' command to save the restored configuration.

7.4 How to Back up an IOS File from a Cisco Router

When would you need this: When you are planning to upgrade the IOS file or you need to copy it to another router.

Special Requirements: None.

Before starting the procedure of IOS file backup, you will need to install TFTP server's software on a PC connected to the router Ethernet interface. There are many free downloadable TFTP server's software on the Internet; however, our recommendation is TFTPd or Free TFTP server's.

Afterward, you make sure to direct the TFTP server's to the folder that you want to contain the backups, and that the TFTP server's has enough free space to contain the backups.

1. Create a console connection with the default settings (9600 baud, 8 databits, 0 parity bits, 1 stop bit, no flow control).
2. Check the connectivity between the router and the TFTP server's with the '`ping`' command.
3. Start copying the IOS file with one of the following commands:
   `Router#`**`copy flash tftp`**
   Use this command if your router has Internal Flash Memory (e.g., 2600). If your router uses PCMCIA flash cards (e.g., 3600), use the following command:
   `Router#`**`copy slot1: tftp`**
   or `Slot0:` depending on the file you want to copy
4. Now you will be asked for the IP address of the TFTP server's:
   `Address or name of remote host []?` *`tftp-server-address`*
5. Afterward, you will be asked for a destination file name to be saved on the TFTP server's.
   Destination filename [*ios-filename*.bin]?
   It is better to leave the IOS file name as it is and press 'enter' to avoid any possible confusion at the time of restore.
   Now you will see the progress of the files transfer.

```
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !

! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !

! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !

! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !

! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !

! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !

! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! !  ! ! !
[OK - nnnnn bytes]
nnnnnn bytes copied in yy.yyy secs (zzz bytes/sec)
```

For the restore procedure, you can refer to the IOS file upgrade procedure in Sect. 7.5.

7.5 How to Upgrade IOS on a Cisco Router

When would you need this: The upgrade is required when you plan to add more duties to the router or a new hardware module. The installation is also required when the IOS image you have on the router is corrupted.

Special Requirements: The router's flash size should be enough for the new IOS image.

Before starting the procedure of IOS upgrade, you will need to install TFTP server's software on a PC connected to the router Ethernet interface. There are many free downloadable TFTP server's software on the Internet; however, our recommendation is TFTPd or Free TFTP server's.

Afterward, you should make sure to direct the TFTP server's to the folder containing the new IOS image that you have.

We will put down two procedures for two different type of routers; a procedure for routers having Internal Flash (e.g., 2600), and a slightly different procedure for routers with PCMCIA flash cards (e.g., 3600).

7.5.1 Upgrade Procedure for Cisco Routers with Internal Flash

1. Create a console connection with the default settings (9600 baud, 8 databits, 0 parity bits, 1 stop bit, no flow control).
2. Verify the connectivity between the router and the TFTP server's using '`ping`'. Make sure that the router interface and the TFTP server's IP addresses are in the same range and the ping is responding well.
3. Although the upgrade will be happening in the flash and the configuration is saved in the NVRAM, make a backup of the configuration. This is recommended in case something goes wrong in the upgrade. Also, make a backup copy of the IOS you already have on the router. In case the new IOS image is corrupted, you will be on the safe side. For the backup process, please refer to the IOS backup procedure in Sect. 7.4 and configuration backup procedure in Sect. 7.3
4. Start the upgrade by the command:

   ```
   Router#copy tftp flash
   ```

   Now you will be prompted for the IP address of the TFTP server's:

   ```
   Address or name of remote host []? tftp-server-address
   ```

   Afterward, you will be asked for the name of the new IOS file being copied from the TFTP server's:

   ```
   Source filename []? ios-filename.bin
   ```

   Keep in mind that the IOS file name is case sensitive.

   Now you will be asked for the destination file name on your router,

   ```
   Destination filename []?ios-filename.bin
   ```

It is preferred to keep it as the source file name, to be able to easily identify the files on the TFTP server's as compared to the ones on the routers. Also, keeping the name means identifying the features of this IOS easily.

Now you will be asked whether to erase the existing file(s) in the flash or not. If you have enough free space on the flash, do not erase the old IOS image, you might need it.

```
Erase flash: before copying? [confirm]
```

Afterward, the router starts copying the new IOS file to the router, or start erasing the flash and then copying. If you wish to keep the contents of the flash, just write 'n' and hit enter.

```
Erasing the flash filesystem will remove all files! Continue?
[confirm]y
Erasing device... eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeee ...erased
Erase of flash: complete
Loading ios-filename.bin from tftp-server-address (via
Ethernet0/0):
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! !
[OK - nnnnnn/mmmmmm bytes]
Verifying checksum... OK (0xAC8A)
nnnnnn bytes copied in tt.tt secs (yyyyy bytes/sec)
```

The copying process takes several minutes; the time differs from network to network. During the copying process, messages are displayed to indicate which file has been accessed.

The exclamation point '`!`' indicates that the copying process is taking place. Each exclamation point indicates that ten packets have been transferred successfully.

A checksum verification of the image occurs after the image is written to flash memory.

5. Before reloading the router, you need to make sure of two things. The first is that the configuration register value is 0x2102. You can check that with the '`show version`' command. If the configuration register's value is not 0x2102, you will need to set it to that value with the following command:

   ```
   Router(config)#config-register 0x2102
   ```

   If you did not erase the contents of the flash, you will need to set up the router to boot from the new IOS file with the following commands:

   ```
   Router(config)#no boot system
   Router(config)#boot system flash ios-filename.bin
   ```

6. If you type the reload command, the router asks you if you want to save the configuration. You need to pay attention to this situation. If the router is in boot mode for instance, it is a subset of the full Cisco IOS software which is running and there is no routing functionality. Therefore, all the routing configuration is automatically erased from the running-configuration. Thus, if you save the configuration at this time, you will erase the complete startup-configuration that is already there in the NVRAM and replace it by the incomplete running-configuration.

   Save the configuration only if you are sure that you have the full configuration in the output of show run. It is not necessary to save the configuration to take into account the new config-register if this one has been changed previously. That is done automatically.

   ```
   Router#reload
   System configuration has been modified.Save?[yes/no]: y
   Building configuration...[OK]
   Proceed with reload? [confirm]y
   ```

7. To verify that the new image is loaded after the '`reload`', use '`show version`' command.

   ```
   00:22:25: %SYS-5-CONFIG_I: Configured from console by
   console
   Cisco Internetwork Operating System Software
   IOS TM CNNNN Software (CNNNN-N-N), Version NN.N(NN),
   RELEASE SOFTWARE (fc1)
   Copyright (c) 1986-2002 by Cisco Systems, Inc.
   Compiled Mon 25-Mar-02 20:33 by xxxxx
   Image text-base: 0980008088, data-base: 0980828788
   ROM: System Bootstrap, Version nn.n(n)XA4, RELEASE
   SOFTWARE
   ```

```
(fc1)
XXXX uptime is 22 minutes
System returned to ROM by reload
System image file is 'flash: cNNNN-N-NN.NNN-NN.bin ' /
```

Check it here

In step 1 or after the upgrade, if the router boots into rommon mode or boot mode and you have one of the following cases:

```
rommon 1>dir flash:
device does not contain a valid magic number
dir: cannot open device 'flash:'
rommon 2>
```

or

```
router(boot)>dir
device does not contain a valid magic number
boot: cannot open 'flash:'
boot: cannot determine first file name on device 'flash:'
```

This means that the flash is empty or the file system is corrupted. In this case, you have to consider using the procedure of upgrading or installing the IOS from rommon mode in Sect. 7.7.

7.5.2 Upgrade Procedure for Cisco Routers with PCMCIA Flash

1. Create a console connection with the default settings (9600 baud, 8 databits, 0 parity bits, 1 stop bit, no flow control). If your router does not boot regularly, refer to Sect. 6.6.
2. Verify the connectivity between the router and the TFTP server's with the 'ping' command. Make sure that the TFTP server's software is running and the working directory of the TFTP server's contains the new IOS file. It is also advised that you backup the configuration and old IOS file before proceeding. For this purpose, you can refer to IOS backup procedure in Sect. 7.4 and configuration backup procedure in Sect. 7.3.
3. Check if you have enough space in the flash card for the new IOS file:

   ```
   Router#dir slot1:
   ```

   If you find that there is not enough space, you can delete one or more files from the flash:

   ```
   Router#delete slot1: old-ios-file.bin
   ```

   If you delete one or more files from the flash do not reload or power-cycle the router until you finish this procedure. The operating system the router using now is loaded to the router's RAM. Thus, if you power-cycle the router before flashing the new IOS, the router will malfunction.

4. Copy the new IOS file from the TFTP server's to the router:

   ```
   Router#copy tftp slot1:
   Address or name of remote host []? tftp-server-address
   Source filename []? ios-filename.bin
   Destination filename [ios-filename.bin]?
   Accessing tftp://tftp-server-address/ios-filename.bin
   Erase slot1: before copying? [confirm]n
   ```

   You can say 'no' here because you have already emptied space for the new IOS file

   ```
   Loading ios-filename.bin from tftp-server-address (via
   Ethernet1/0):
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
   ! ! ! ! ! ! ! ! ! !  ! ! !
   [OK - nnnnnnn/mmmmmmmm bytes]
   Verifying checksum... OK (0x13F0)
   nnnnnnnn bytes copied in tt.ttt secs (rrrrr bytes/sec)
   Router#
   ```

5. Verify the new IOS file in the flash card:

   ```
   Router#dir slot1:
   ```

6. Direct the router to load the new IOS file in the next boot:

   ```
   Router(config)#no boot system
   Router(config)#boot system flash slot1: ios-filename.bin
   ```

7. Make sure that the configuration register has the value of 0x2102. This is verified by the '`show version`' command. If the configuration register has a value other than 0x2102, use the following command to change it:
   ```
   Router(config)#config-register 0x2102
   ```
8. Save the configuration with one of the two following commands:
   ```
   Router#write memory
   ```
   Or
   ```
   Router#copy run start
   ```
9. Reload the router with '`reload`' command and verify the new IOS version with the '`show version`' command after the reload. This command will also show you the name of the IOS file that has been loaded.

7.6 How to Upgrade IOS of a Cisco Router Using HyperTerminal

When would you need this: When you want to upgrade the IOS file and you do not have TFTP of FTP server's around, or you have a corrupted IOS in the router and the router is not booting up.
Special Requirements: The router flash size should be adequate for the new IOS image, and you should have enough RAM in the router for the operation and temporary storage of the new IOS file.

It is recommended that you backup the old IOS file before the upgrade using the IOS backup procedure in Sect. 7.4, and back up the configuration too using the procedure in Sect. 7.3 if you still have access to the router. Keep in mind that this procedure is not recommended. It is recommended that you upgrade using a TFTP or FTP server from Sect. 7.5. This is because it takes much more time and you do not see an error when it occurs until the copying is finished. To upgrade or install IOS using HyperTerminal or any other terminal emulation software, do the following steps:

1. Create a console connection with the default settings (9600 baud, 8 databits, 0 parity bits, 1 stop bit, no flow control).
2. Change the baud rate of the console port to its maximum 115,200
   ```
   Router#set baud=115200
   ```
3. Reset the console port
   ```
   Router#reset
   ```
   Now you will not get anymore output on the screen until you finish step 4.
4. Change the connection speed of the HyperTerminal by disconnecting and reconnecting with the baud rate of 115200 with all the rest of the settings mentioned in step 1 left the same.
5. Prepare the router for the reception of the new IOS file
   ```
   Router#upload XMODEM
   ```
   Now you will have the following message,

```
Ready for X/Modem upload...
[note: no status bar for XMODEM transfers, abort with
Control-X or break]
```

6. Send the file from your terminal emulation software. This is done by selecting 'Transfer' from the upper menu of HyperTerminal and then 'Send File'. In the Send File dialog box, choose the new IOS file using the 'Browse' button, choose 'XMODEM' as the protocol, and then press 'Send'.

   The sending operation may take a long time, and there is no progress indicator in the router, but you will have a progress indicator in the HyperTerminal.

   After the copying is finished, you will receive a message,

   ```
   upload: succeeded (ttt seconds)
   ```

   Now the new IOS file is in the router's RAM.
7. If you do not have enough space in the router's flash for the old and new files, delete the old IOS file (this is not recommended though),

   ```
   Router#delete flash:old-ios-file.bin
   ```
8. Save the new IOS file to the flash,

   ```
   Router#save file=new-ios-file.bin
   ```

   The *new-ios-file.bin* is a name of your choice for the new IOS file. It is better to use the same name of the original file that was stored on your computer.
9. Direct the router to load the new IOS file at the next startup,

   ```
   Router(config)#no boot system
   Router(config)#boot system flash new-ios-file.bin
   ```
10. You can reload the router now, but remember to change back the settings of the HyperTerminal to the 9600 baud to get output on your screen after the reload. If you type the reload command, the router might ask you whether you want to save the configuration. If the router is in boot mode for instance, it is a subset of the full Cisco IOS software which is running and there is no routing functionality.

    Therefore, all the routing configuration is gone in the running-configuration and if you save the configuration at this time, then you erase the good startup-configuration in NVRAM and replace it by the incomplete running-configuration.

    Save the configuration only if you are sure that you have the full configuration in the output of show run. It is not necessary to save the configuration.

7.7 How to Upgrade or Install IOS on Cisco Router Using rommon Mode

When would you need this: If your router flash or IOS file is corrupted, you can use this procedure to install a new IOS file. Although it is not recommended, this procedure can also be used to upgrade the router IOS.

Special Requirements: The router flash size should be enough for the new IOS file.

Before starting the procedure of IOS upgrade or installation, you will need to install TFTP server's software on a PC connected to the router Ethernet interface.

There are many free downloadable TFTP server's software on the Internet; however, our recommendation is TFTPd and Free TFTP server's.

After installing the TFTP server's software on your computer, make sure to direct the TFTP server's to the folder containing the new IOS image that you have.

If you are using this procedure to upgrade the IOS file and router is operating properly, it is preferred to back up the old IOS file before starting the upgrade procedure. For this purpose, refer to the IOS backup procedure in Sect. 5.4.

1. Create a console connection with the default settings (9600 baud, 8 databits, 0 parity bits, 1 stop bit, no flow control).
2. If your flash or IOS file is corrupted and your router goes directly to router boot mode (`Router(boot)#`), go to step 4. If your router has some problems and boots into the ROMmon mode directly (`rommon 1>` or >), go to step 3. If your router boots normally, interrupt the router boot sequence by pressing Ctrl-Break once the router is powered on. This will take you to ROMmon mode with the prompt:
   ```
   rommon 1>
   ```
   Or
   ```
   >
   ```
3. Change the value of the configuration register to 0x2101 to instruct the router to boot into router boot mode. Afterward, reload the router.
   If you have the '`rommon 1>`' prompt, use the commands:
   ```
   rommon 1>confreg 0x2101
   rommon 2>reset
   ```
   While if you have the '>' prompt, use:
   ```
   >o/r 0x2101
   >i
   ```
4. Now you are in the router boot mode with the prompt (`Router(boot)#`), you will need to give a valid IP address and default-gateway address to the router, so it can communicate with the TFTP server. This IP address will be assigned to the router interface Ethernet 0/0 or fastethernet 0/0. Make sure that this interface is where you connect the TFTP server to the router.
   ```
   Router(boot)>enable
   Router(boot)#configure terminal
   Router(boot)(config)#interface ethernet 0
   Router(boot)(config-if)#ip address interface-address subnetmask
   Router(boot)(config-if)#no shutdown
   Router(boot)(config-if)#exit
   Router(boot)(config)#ip default-gateway default-gateway-address
   ```

You can replace Ethernet with fastethernet if this is the type of the interface your router has. Also, you can use any IP address and subnetmask that you see suitable as long as it fits with the TFTP server. The default gateway IP address does not matter if you have the TFTP server in the same network where the router interface is. You can easily set it up to be the IP address of the TFTP server.

5. Check the connectivity between the router and the TFTP server with the '`ping`' command.
6. Start the copying of the new IOS file from the TFTP server to the flash.

```
Router(boot)#copy tftp flash
```

Now you will be asked for the IP address of the TFTP server,

```
Address or name of remote host [255.255.255.255]? tftp-server-address
```

Then, the source file name,

```
Source file name? ios-filename.bin
```

Please note that the file name is case sensitive and make sure that the TFTP server's working directory is the one containing the new IOS file.

Afterward, you will be asked for a destination file name,

```
Destination file name [ios-filename.bin]?
```

It is advised that you keep the file name unchanged for future reference.

```
Accessing file 'ios-filename.bin' on tftp-server-address....
Loading ios-filename.bin from tftp-server-address (via
Ethernet0):
! [OK] Device needs erasure before copying new file
Erase flash device before writing? [confirm]y
Copy 'ios-filename.bin' from server as 'ios-filename.
bin'into Flash
WITH erase? [yes/no]yes
Erasing device...
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee ...erased
Loading ios-filename.bin from tftp-server-address (via
Ethernet0):

! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !
OK - nnnnnn/yyyyyy bytes]
Verifying checksum... OK (096526)
Flash device copy took 00:yy:yy [hh:mm:ss]
Router(boot)#
```

7. Change back the configuration register value to 0x2102.

```
Router(boot)#configure terminal
Router(boot)(config)#config-register 0x2102
Router(boot)(config)#exit
Router(boot)#
```

8. Reload the router

```
Router(boot)#reload
System configuration has been modified. Save? [yes/no]:no
Building configuration...
[OK]
Proceed with reload? [confirm]
```

9. Everything should look fine now, and you should be getting the regular (`Router>`) prompt. To check the version and file name of the new IOS, use the '`show version`' command.

7.8 How to Copy IOS from One Cisco Router to Another

When would you need this: When you want to copy IOS file from one router to another for the purpose of upgrade or install. This is usually required when you do not have a TFTP server's around.

Special Requirements: The flash size of the destination router should be adequate for the new IOS file size. The models of both routers must be the same.

On the source router that contains the IOS file that you want to copy, issue the following command:

```
Router(config)#tftp-server flash:/source-ios-file-name.bin
```

Where *source-ios-file-name.bin* is the name of the IOS file that you want to copy. If you are using a router that has PCMCIA flash car, replace the '`flash:`' with '`slot0:`' or '`slot1:`' in the previous command, depending on the slot that contains the file that you want to copy.

This command will make the router act as a TFTP server's. The rest of the procedure is done on the target router and can be found in Sect. 7.5.

After you complete the copy operation, issue the command to disable the TFTP server's on the router:

```
Router(config)#no tftp-server flash:/source-ios-file-name.bin
```

7.9 How to Partition Internal Flash Memory of a Cisco Router

When would you need this: When you have enough space in the router's flash and you intend to have two IOS images to load alternatively.

Special Requirements: To partition flash memory, you must have at least two banks of flash memory. A bank is a set of four chips. This requirement includes systems that support a single SIMM that has two banks of flash memory. The minimum partition size is the size of a bank.

On most class B flash file systems, you can partition banks of flash memory into separate, logical devices so that the router can hold and maintain two or more different software images.

This partitioning allows you to write software into flash memory while running software in another bank of flash memory.

This command is an example of how to partition flash memory:

```
Router(config)#partition flash partitions size-of-partition1 size-of-partition2
```

This following command is for Cisco 1600 and 3600 series routers:

```
Router(config)#partition flash-filesystem: number-of-partitions partition-size
```

All sizes mentioned here are in Megabytes. This task succeeds only if the system has at least two banks of flash, and the partitioning does not cause an existing file in flash memory to be split across the partitions.

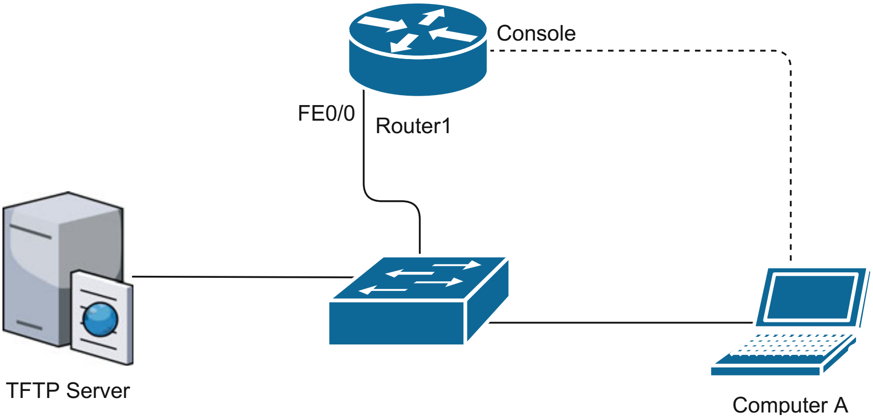
For all platforms except the Cisco 1600 series and 3600 series routers, flash memory can only be partitioned into two partitions.

For the Cisco 1600 and 3600 series routers, the number of partitions that you can create in a flash memory device equals the number of banks in the device.

Issue the `show flash-filesystem: all` command to view the number of banks on the flash memory device. The number of partition size entries you set must be equal to the number of specified partitions. For example, the partition slot0: 2 8 8 command configures two partitions to be 8 MB in size each. The first 8 corresponds to the first partition's size in MBs, while the second 8 corresponds to the second partition's size in MBs.

7.10 Training Scenarios

Scenario 7.1



Connect the network shown in the figure above and configure the following settings:

1. On Router 1 do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.254/24

2. On Computer A change the following settings:

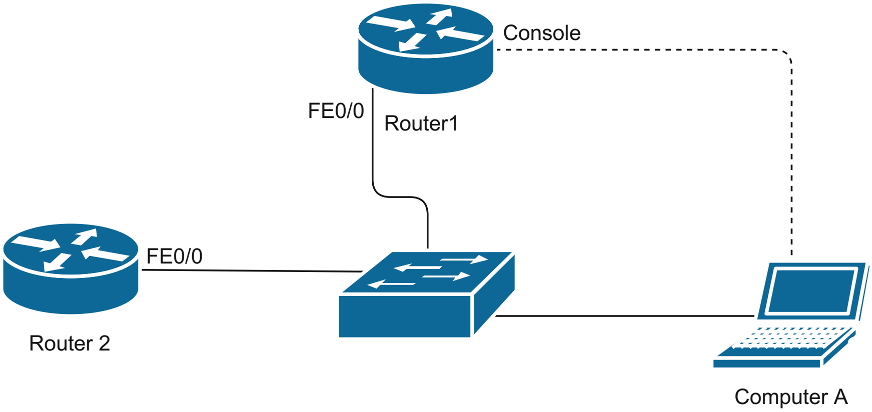
Device	Parameter	Value
Computer A	IP Address	172.16.0.1
	Subnet Mask	255.255.255.0
	default-gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

3. TFTP server's settings:

Device	Parameter	Value
TFTP server's	IP Address	172.16.0.100
	Subnet Mask	255.255.255.0
	default-gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

4. To check the connectivity, ping from Computer A to the TFTP server's and from Router 1 to the TFTP server's.
5. TFTP server's contains, in its working directory, a new version of IOS named `new-ios.bin`.
6. Upgrade the IOS on Router 1 to the new version available on the TFTP server's.
7. Direct the router to boot into the new file when the router reboots.

Scenario 7.2



Connect the network shown in the figure above and configure the following settings:

1. On Router 1 do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.254/24

2. On Router 2 do the basic configuration:

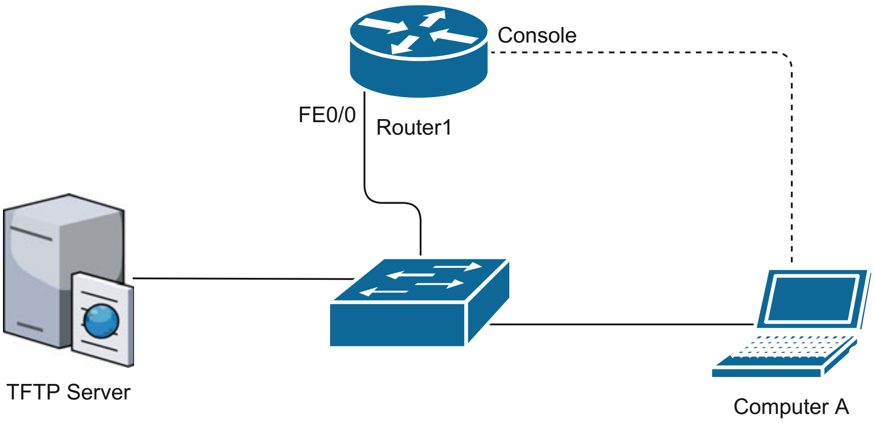
Device	Parameter	Value
Router 2	Hostname	Router2
	Console Password	Cisco2Console
	Secret Password	Cisco2
	VTY Password	Cisco2VTY
	Interface FE0/0 IP Address	172.16.0.250/24

3. On Computer A change the following settings:

Device	Parameter	Value
Computer A	IP Address	172.16.0.1
	Subnet Mask	255.255.255.0
	default-gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

4. To check the connectivity, ping from Computer A to the Router 2 and from Router 1 to the Router 2.
5. Router 2 contains, in its flash, a new version of IOS named `new-ios.bin`.
6. Set up Router 2 to be a TFTP server's and publishing its flash contents as the TFTP working directory.
7. Upgrade the IOS on Router 1 to the new version available on the TFTP server's (Router 2).
8. Direct the Router 1 to boot into the new file when the router reboots.
9. Remove the TFTP settings from Router 2.

Scenario 7.3



Connect the network shown in the figure above and configure the following settings:

1. On Router 1 do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console Password	CiscoConsole
	Secret Password	Cisco
	VTY Password	CiscoVTY
	Interface FE0/0 IP Address	172.16.0.254/24

2. On Computer A change the following settings:

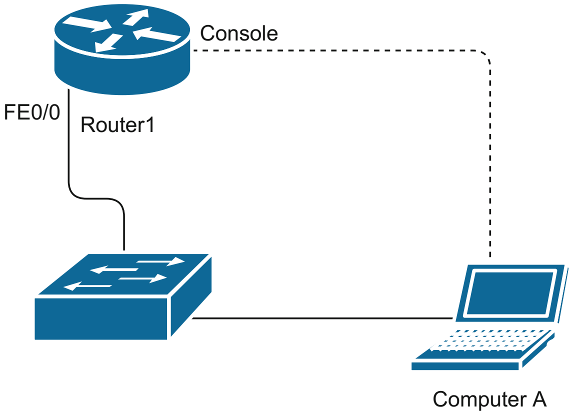
Device	Parameter	Value
Computer A	IP Address	172.16.0.1
	Subnet Mask	255.255.255.0
	default-gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

3. TFTP server's settings:

Device	Parameter	Value
TFTP server's	IP Address	172.16.0.100
	Subnet Mask	255.255.255.0
	default-gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

4. To check the connectivity, ping from Computer A to the TFTP server's and from Router 1 to the TFTP server's.
5. Back up the running-configuration from Router 1 the TFTP server's. Name the destination file 'backup-configuration.conf'
6. Change the hostname of Router 1 to 'NotRouter1'. Do a '`show run`' just to make sure that the configuration has changed.
7. Restore the backup-configuration.conf file from the TFTP server's to the running-configuration.
8. Do a '`show run`' just to make sure that the configuration has changed back to the original.

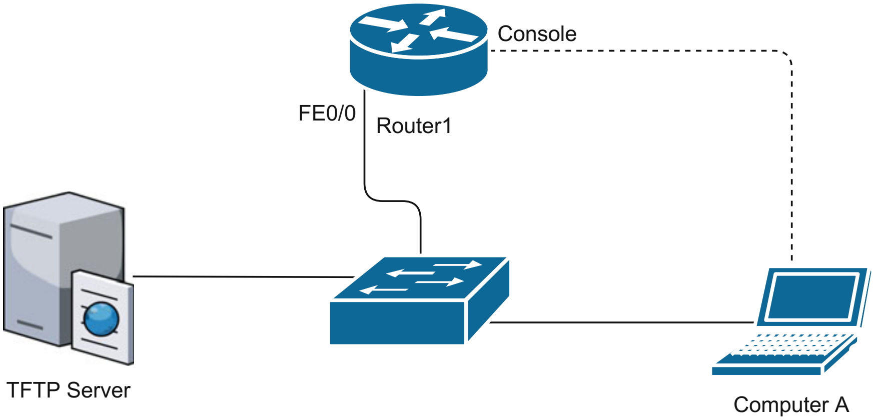
Scenario 7.4



Connect the simple network shown in the figure above. The IOS file on Router 1 has been corrupted and you cannot access the TFTP server's nor set up an IP address.

1. Use the Terminal Emulation software installed on Computer A to restore the IOS file from Computer A to Router 1 using XMODEM protocol.
2. Configure the router to load the new IOS instead of the old one after reboot.

Scenario 7.5



Connect the network shown in the figure above. Router 1 has a corrupted IOS image and it does not boot properly. The router keeps booting to rommon mode. Configure the following settings:

1. On Computer A change the following settings:

Device	Parameter	Value
Computer A	IP Address	172.16.0.1
	Subnet Mask	255.255.255.0
	default-gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

2. TFTP server's settings:

Device	Parameter	Value
TFTP server's	IP Address	172.16.0.100
	Subnet Mask	255.255.255.0
	default-gateway	172.16.0.254
	DNS Server 1	172.16.0.254
	DNS Server 2	172.16.0.254

3. To check the connectivity, ping from Computer A to the TFTP.
4. Using rommon mode, set up the router's FE0/0 interface to use the IP address 172.16.0.254/24.
5. The TFTP server's contains, in its working directory, a new version of IOS named `new-ios.bin`.
6. Upgrade the IOS on Router 1 to the new version available on the TFTP server's in rommon mode.
7. Direct the router to boot into the new file when the router reboots.

Chapter 8
Remote Connectivity to Cisco Router

Keywords Cisco · Router · Telnet · SSH · Username · Access-list · Remote connection

8.1 How to Configure SSH on a Cisco Router

When would you need this: When you need to configure your router remotely through an insecure environment.

Special Requirements: IOS version over 12.1.3.T (with a 'k9' in its feature set).

Using Telnet over the Internet is not a smart choice. This is due to the fact that Telnet transports everything in plaintext without any kind of encryption. The alternative for that is the use of secure shell host (SSH). SSH encrypts the traffic between the router and the terminal to ensure protection of the content. Let us jump into the configuration now:

1. You need to set up a hostname and domain name because they will be used in generating the security keys used in encryption:
   ```
   Router#config t
   Router(config)#hostname router-name
   router-name(config)#ip domain-name your-domain
   ```
   where
   router-name is the hostname of your choice.
   your-domain is the domain name of your network. If you are not using a domain name, just give any name for the sake of SSH.
2. Generate the keys to be used for the RSA encryption:
   ```
   router-name(config)#crypto key generate rsa
   ```
3. Set up the two important parameters of SSH—the connection timeout and the number of authentication retries:
   ```
   router-name(config)#ip ssh time-out time-out
   ```
   where *time-out* is the connection timeout in seconds (e.g., for two minutes put 120 seconds).

M.M. Alani, *Guide to Cisco Routers Configuration*,
DOI 10.1007/978-3-319-54630-8_8

`router-name(config)#`**`ip ssh authentication-retries`** *`number-of-retries`*
where *number-of-retries* is the maximum number of authentication retries allowed.
The settings of these two parameters, or one of them, along with the '`crypto key generate rsa`' enable SSH.

4. Disable Telnet sessions, and set up the router to accept only SSH. Before doing that, it is advised to try out SSH and make sure it is working properly.
   `router-name(config)#`**`line vty 0 4`** (or **`0 15`**, depending on the router type)
   `router-name(config-line)#`**`transport input ssh`**
5. For troubleshooting, use the following command:
   `router-name#`**`sh ip ssh`**
6. As an additional security measure, you can change the port number that SSH uses. By default, SSH uses port number 22. You can change that through the following command:
   `router-name(config)#`**`ip ssh port`** *`new-port-number`*
   where *new-port-number* is a port number of your choice. Remember to set up your SSH client to contact the new port, not the old 22.
7. To disable SSH, you can use the command:
   `router-name(config)#`**`crypto key zeroize rsa`**
   This command deletes the RSA key. Hence, SSH will be disabled. If you want to go back to Telnet afterward, use these commands:
   `router-name(config)#`**`line vty 0 4`** (or **`0 15`**, depending on the router type)
   `router-name(config-line)#`**`transport input telnet`**
8. For further security, it is advised that you configure an Access-list to limit the IP addresses that are allowed to initiate SSH sessions with the router. This can be done using the procedure of securing Telnet sessions with an Access-list shown in Sect. 8.3.

8.2 How to Secure Remote Access Sessions with Password Only

This can be done for both Telnet and SSH.

8.2.1 Telnet

Generally, Telnet is not a secure protocol. All configuration commands, including passwords, are sent in plaintext. It is highly not recommended to use Telnet to connect to a router through the Internet. Some network administrators still use Telnet for local access to the router. If you plan to use Telnet, set up the Telnet password. If you do not intend to use Telnet in the near future, do not set it up.

```
Router(config)#line vty 0 4
Router(config)#transport input telnet
Router(config-line)#password telnet-password
Router(config-line)#login
```

In some routers, `vty 0 15` is used instead of `vty 0 4`, depending on the number of simultaneous Telnet sessions you want to allow. If you need only one, just write

```
Router(config)#line vty 0
```

Instead of the first command written above.

8.2.2 SSH

SSH is much more secure as compared to Telnet because it involves encryption of all data transmitted between the two sides. More detailed explanation of SSH configuration was discussed in Sect. 8.1.

```
Router(config)#line vty 0 4
Router(config-line)#transport input ssh
Router(config-line)#password ssh-password
Router(config-line)#login
```

In some routers, `vty 0 15` is used instead of `vty 0 4`, depending on the number of simultaneous Telnet sessions you want to allow. If you need only one, just write

```
Router(config)#line vty 0
```

Instead of the first command written above.

8.3 How to Secure Remote Access Sessions with Username and Password

When would you need this: When you need to protect your remote sessions with a username and a password instead of a password only.

Special Requirements: None.

8.3.1 Telnet

If you plan to use Telnet, set up Telnet to use a username and a password instead of username only. If you do not intend to use Telnet in the near future, do not set it up.

```
Router(config)#username telnet-username password telnet-password
Router(config)#line vty 0 4
Router(config-line)#transport input telnet
Router(config-line)#login local
```

where

telnet-username is the username to be used when logging in using Telnet.

telnet-password is the password associated with the username.

It is possible to repeat the '`username`' command to allow more than one user to login through Telnet.

In some routers, `vty 0 15` is used instead of `vty 0 4`, depending on the number of simultaneous Telnet sessions you want to allow. If you need only one, just write

```
Router(config)#line vty 0
```

Instead of the second command written above.

8.3.2 SSH

Remember that this is not the complete SSH configuration. The detailed configuration is discussed in Sect. 8.1.

```
Router(config)#username ssh-username password ssh-password
Router(config)#line vty 0 4
Router(config-line)#transport input ssh
Router(config-line)#login local
```

where

ssh-username is the username to be used when logging in using SSH.

ssh-password is the password associated with the username.

It is possible to repeat the '`username`' command to allow more than one user to login through SSH.

In some routers, `vty 0 15` is used instead of `vty 0 4`, depending on the number of simultaneous remote sessions you want to allow. If you need only one, just write

```
Router(config)#line vty 0
```

Instead of the second command written above.

8.3.3 Console

Although the console connection cannot be categorized as a remote connection, we would like to explain protecting it using a username and password instead of password only like explained in Sect. 1.2.

```
Router(config)#username console-username password console-password
Router(config)#line console 0
Router(config-line)#login local
```

where

console-username is the username to be used when logging in using console connection.

console-password is the password associated with the username.

It is possible to repeat the '`username`' command to allow more than one user to login through console port.

8.4 How to Secure Telnet Sessions Using Access-Lists on a Cisco Router

When would you need this: When you need to set up Telnet on a Cisco Router to facilitate remote configuration.

Special Requirements: None.

The steps to secure a Telnet session with an Access-list are very simple. However, we will start by creating a password for the Telnet access on the router as a first step of security:

1. If you expect to use no more than one Telnet session simultaneously, enable only one using the following command in the global configuration mode:
   ```
   Router(config)#line vty 0
   ```
   If you need to initiate more than one Telnet session at the same time, which is highly unlikely, you can write '`line vty 0 4`' or '`line vty 0 15`' depending on the type of the router you are using.
2. Set up a password for the Telnet session:
   ```
   Router(config-line)#password telnet-password
   ```
   where *telnet-password* is a password of your choice.
3. Activate the Telnet password:
   ```
   Router(config-line)#login
   ```
4. Configure the Access-list to allow the IP address of your network administrators' computers that will be allowed to Telnet the router:
   ```
   Router(config)#access-list acl-number permit host admin-computer-address
   ```
   where

   acl-number is the access-list number of your choice. Since we are creating a standard Access-list, the number has to be between 1 and 99.

   admin-computer-address is the IP address of the network admin that will be allowed to Telnet the router.

5. If you want to Telnet the router from more than one computer, repeat step 4 for each IP address that you want to allow the Telnet from. Remember to keep the same Access-list number for all the different IP addresses.
6. Apply the Access-list to the Telnet line:

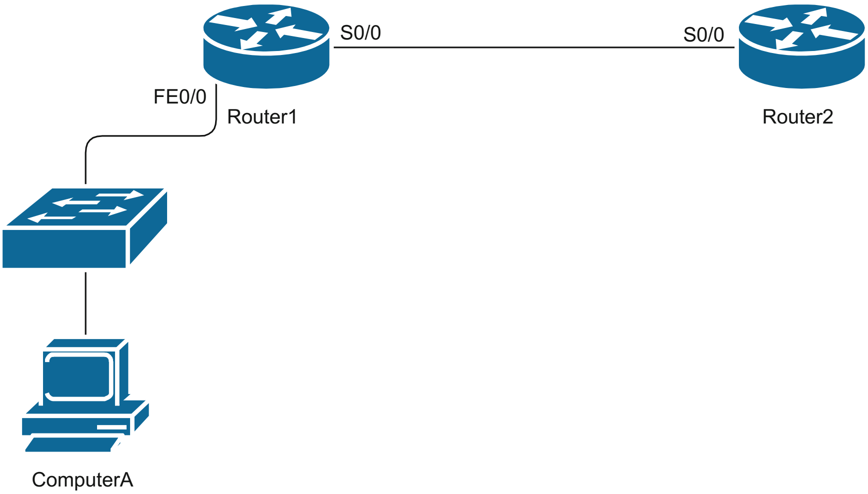
```
Router(config)#line vty 0
Router(config-line)#access-class acl-number in
```

   where *acl-number* is the number of the Access-list that you have configured earlier in step 4.

   This command applies the Access-list to the Telnet line on the incoming traffic.

8.5 Training Scenarios

Scenario 8.1



Connect the network shown in the figure above and configure the following settings:

1. Routers' basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console password	CiscoConsole
	Secret password	Cisco
	VTY password	CiscoVTY
	Interface S0/0 IP address	10.0.0.1/30
	Interface FE0/0 IP address	172.16.0.254/24

(continued)

(continued)

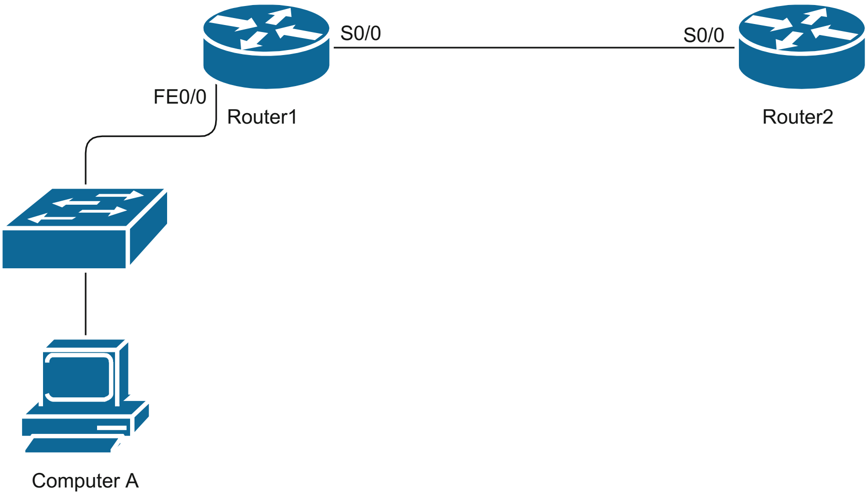
Device	Parameter	Value
Router 2	Hostname	Router2
	Console password	Cisco2Console
	Secret password	Cisco2
	Interface S0/0 IP address	10.0.0.2/30

2. On Computer A, change the following settings:

Device	Parameter	Value
Computer A	IP address	172.16.0.1
	Subnet mask	255.255.255.0
	Default gateway	172.16.0.254
	DNS server 1	172.16.0.254
	DNS server 2	172.16.0.254

3. Configure a default route on Router 1 to forward all traffic through the exit interface S0/0.
4. Configure another default route on Router 2 to forward all traffic though the exit interface S0/0.
5. For testing, PING from Computer A to S0/0 on Router 2.
6. Configure SSH access on Router 2 using the following parameters:
   Domain name: mohammedalani.com
   SSH timeout: 2 min.
   SSH port number: 4005
   Password: ciscoSSH
7. Install a SSH client on Computer A (like PUTTY) and use it to connect to Router 2.

Scenario 8.2



Connect the network shown in the figure above and configure the following settings:

1. Routers' basic configuration:

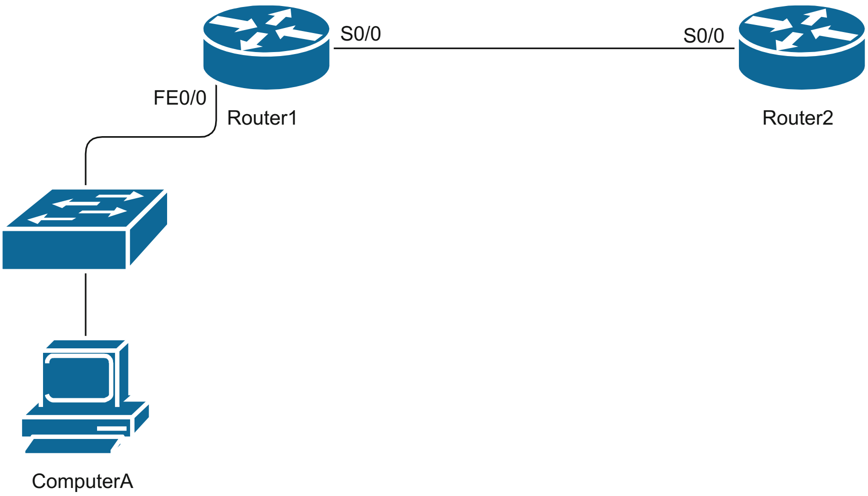
Device	Parameter	Value
Router 1	Hostname	Router1
	Console password	CiscoConsole
	Secret password	Cisco
	VTY password	CiscoVTY
	Interface S0/0 IP address	10.0.0.1/30
	Interface FE0/0 IP address	172.16.0.254/24
Router 2	Hostname	Router2
	Console password	Cisco2Console
	Secret password	Cisco2
	Interface S0/0 IP address	10.0.0.2/30

2. On Computer A, change the following settings:

Device	Parameter	Value
Computer A	IP address	172.16.0.1
	Subnet mask	255.255.255.0
	Default gateway	172.16.0.254
	DNS server 1	172.16.0.254
	DNS server 2	172.16.0.254

3. Configure a default route on Router 1 to forward all traffic through the exit interface S0/0.
4. Configure another default route on Router 2 to forward all traffic though the exit interface S0/0.
5. For testing, PING from Computer A to S0/0 on Router 2.
6. Configure Telnet access on Router 2 using the following parameters:
   Username: TelnetUser1
   Password: ciscoTelnet
7. Install a Telnet client on Computer A (like PUTTY) and use it to connect to Router 2.

Scenario 8.3



Connect the network shown in the figure above and configure the following settings:

1. Routers' basic configuration:

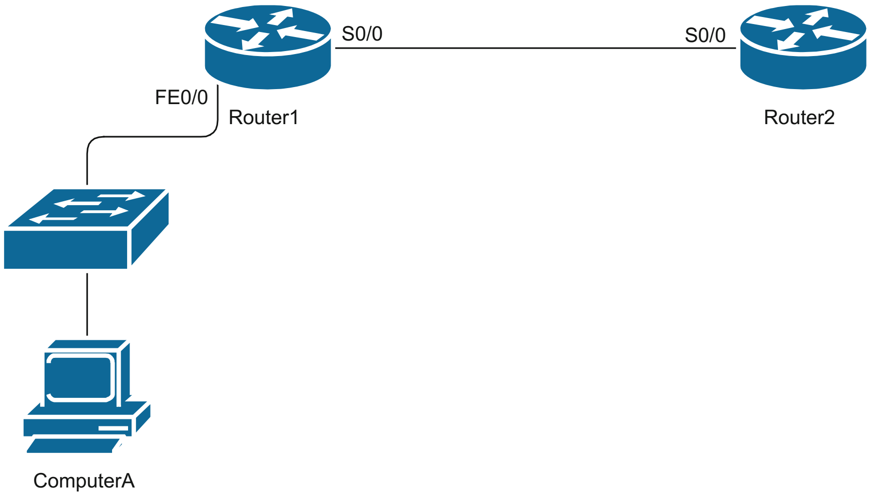
Device	Parameter	Value
Router 1	Hostname	Router1
	Console password	CiscoConsole
	Secret password	Cisco
	VTY password	CiscoVTY
	Interface S0/0 IP address	10.0.0.1/30
	Interface FE0/0 IP address	172.16.0.254/24
Router 2	Hostname	Router2
	Console password	Cisco2Console
	Secret password	Cisco2
	Interface S0/0 IP address	10.0.0.2/30

2. On Computer A, change the following settings:

Device	Parameter	Value
Computer A	IP address	172.16.0.1
	Subnet mask	255.255.255.0
	Default gateway	172.16.0.254
	DNS server 1	172.16.0.254
	DNS server 2	172.16.0.254

3. Configure a default route on Router 1 to forward all traffic through the exit interface S0/0.
4. Configure another default route on Router 2 to forward all traffic though the exit interface S0/0.
5. For testing, PING from Computer A to S0/0 on Router 2.
6. Configure SSH access on Router 2 using the following parameters:
   Domain name: mohammedalani.com
   SSH timeout: 2 min.
   SSH port number: 4055
   Username: SSHUser1
   Password: ciscoSSH
7. Install a SSH client on Computer A (like PUTTY) and use it to connect to Router 2.

Scenario 8.4



Connect the network shown in the figure above and configure the following settings:

1. Routers' basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console password	CiscoConsole
	Secret password	Cisco
	VTY password	CiscoVTY
	Interface S0/0 IP address	10.0.0.1/30
	Interface FE0/0 IP address	172.16.0.254/24

(continued)

(continued)

Device	Parameter	Value
Router 2	Hostname	Router2
	Console password	Cisco2Console
	Secret password	Cisco2
	Interface S0/0 IP address	10.0.0.2/30

2. On Computer A, change the following settings:

Device	Parameter	Value
Computer A	IP address	172.16.0.1
	Subnet mask	255.255.255.0
	Default gateway	172.16.0.254
	DNS server 1	172.16.0.254
	DNS server 2	172.16.0.254

3. Configure a default route on Router 1 to forward all traffic through the exit interface S0/0.
4. Configure another default route on Router 2 to forward all traffic though the exit interface S0/0.
5. For testing, PING from Computer A to S0/0 on Router 2.
6. Configure Telnet access on Router 2 using the following parameters:
   Username: TelnetUser1
   Password: ciscoTelnet
7. Install an Telnet client on Computer A (like PUTTY) and use it to connect to Router 2.
8. Create an ACL to control access to Telnet on Router 2. Allow access to 172.16.0.10 only and deny all other IP addresses. Apply the ACL to the VTY line.
9. Try to Telnet from Computer A. Your trial should fail.
10. Change the IP address of Computer A to 172.16.0.10.
11. Try to Telnet from Computer A to Router 2. Your trial should succeed.

Chapter 9
Hints and Tips

Keywords Cisco • Router • Show command • Password recovery • Break key sequence • ROMMON

9.1 Top 10 Tips for Cisco Routers Configuration

There are few simple things that might help administrators in utilizing their time working with Cisco routers. I gathered the most important 10 things in my point of view:

1. The best sequence of configuring a Cisco router, as I see it, is the following:

   a. Set up the hostname with the '**`hostname`** *`hostname`*' command.
   b. Set up the secret password (better than enable password) with the '**`enable secret`** *`password`*' command.
   c. Set up console and Telnet passwords (use the '`logging synchronous`' command at the console) with the '**`password`** *`password`*' and '**`login`**' commands.
   d. Encrypt the unencrypted passwords with '**`service password-encryption`**' command and do not forget to turn it off after you '**`show run`**'.
   e. Set up the interfaces (IP addresses, description, bandwidth, etc.) with '**`ip address`**', '**`bandwidth`**', and '**`description`**' commands.
   f. Set up the Routing protocols (or static routes).
   g. Test the connectivity with '**`ping`**' and '**`traceroute`**'.
   h. Set up the access-lists.
   i. Test the connectivity (again).

2. Be as descriptive as possible.
   Use the '**`description`**' command on all interfaces. Give useful description in it. Describe the network to which this interface is connected, the bandwidth of the link, the duplex settings, and any other information that you might think useful.

M.M. Alani, *Guide to Cisco Routers Configuration*,
DOI 10.1007/978-3-319-54630-8_9

Use '**remark**' in writing the access-lists so you would identify the access-list according to its function. And if you find it necessary, use banners.
Examples:

```
RouterA(config-if)#description This link is connected to
the Accounting LAN
RouterA(config)#access-list 101 remark This list stops
the Telnet to the Marketing net
RouterA(config)#banner motd #This router is connected to
the marketing and accounting LANS#
```

3. Use hotkeys.
   There are many useful hotkeys in the configuration command line environment. Few of the most important are as follows:
   Ctrl-P: Recalls the previous command in the history buffer (Up Arrow ↑ does the same thing)
   Ctrl-N: Recalls the next command in the history buffer (Down Arrow ↓ does the same thing)
   Ctrl-E: Goes to the end of the line
   Ctrl-A: Goes to the beginning of the line
   Tab Complete the command after writing adequate number of letters
4. Prevent the router from looking up DNS server for wrong commands.
   When you misspell a command and hit the Enter key, the router does not recognize the command and thinks that it might be a hostname. The router, then, tries to contact the DNS server to resolve the name to an IP address so it would Telnet it. This would waste some time, especially when you have not set up a valid DNS server (because the router will broadcast the request and waits for a DNS server to reply). To turn this off, use the '**transport preferred none**' command in the console and vty lines.
   Example:

   ```
   RouterA(config)#line con 0
   RouterA(config-line)#transport preferred none
   ```
5. Set up the Bandwidth of serial interfaces.
   Use the '**bandwidth**' command for setting the bandwidth of all serial interfaces to guarantee the correct calculation of routing table. The bandwidth of a serial link is dependent on the type of WAN connection you are using. Unlike Ethernet or Fast Ethernet, serial interfaces cannot automatically detect the bandwidth of the link. And the bandwidth of the actual link might be different from the small link between the serial interface and the modem or CSU/DSU device you are using. Also remember to write the bandwidth after the '**bandwidth**' command in Kilobits per second.
   Example:

   ```
   RouterA(config)#int serial 0
   RouterA(config-if)#bandwidth 1024
   ```
   This means the link bandwidth is 1 Mbit/s

6. Turn off auto-summarization of routing updates when using subnetted addresses.
   If you are using subnetting, remember use the '**no auto-summary**' command to turn off auto-summarization when using routing protocols that support it, like OSPF.
   Example:
   ```
   RouterA(config)#no auto-summary
   ```
7. Turn off split-horizon in two cases.
   The first is when you are doing inter-VLAN routing. This is because updates from one VLAN cannot pass to other VLANs. The second case is when you are using Frame Relay to connect one site to multiple sites.
   Example:
   ```
   RouterA(config-if)#no ip split-horizon
   ```
8. The '**show**' command is your best friend.
   Whenever you are in trouble, or even if you are not in trouble, yet your best friend comes up: the '**show**' command. The most widely used 'show' commands are in the following lists:

   a. **show version**—Shows a large amount of information such as the IOS version, the configuration register value, and the interfaces available.
   b. **show ip route**—Shows the routing table.
   c. **show ip interface**—Shows the access-lists applied to interfaces.
   d. **show access-list**—Shows the contents of access-lists.
   e. **show ip protocols**—Shows information about the routing protocols currently running.
   f. **show cdp neighbor detail**—Shows detailed information about neighbouring devices.
   g. **show interface**—Shows status information about interfaces.
   h. **show run**—Shows the running configuration, i.e., all the commands now in action.

   More details about these show commands are explained in the next section.
9. Keep the IP addresses of servers and printers out of the DHCP pool.
   When using the router as a DHCP server, do not forget to exclude the addresses of servers, router interfaces, and printers off the DHCP pool.
   Example:
   ```
   RouterA(config)#ip dhcp excluded-address 192.168.0.1
   RouterA(config)#ip dhcp excluded-address 192.168.0.1 192.168.0.10
   ```
   You can use a single IP address in this command or a start-IP and end-IP to define a range of exclusions.
10. Keep a scheduled '**reload**' when configuring a router remotely.
    When you are configuring a router remotely, you might do something wrong and lose the connectivity with the router. In this case, you will need to restart the router physically. There are chances that no one is around the router to

restart it for you. You can solve this by yourself by using the '**reload in** *n-minutes*' command.
This command schedules a reload after *n-minutes* minutes. So, before you start nosing around the router remotely, issue this command and schedule a reload. If something goes wrong and you lose the connectivity with the router, the router will reload and you get back in business. If things go smooth and you do not need to reload after all, you can issue a '**reload cancel**' command to stop the scheduled restart from happening.

9.2 Ten Show Commands Everybody Needs to Know in Cisco Routers

Some commands in the Cisco router configuration are just irreplaceable. The '`show`' commands are the most widely used in Cisco routers. Here is a list of the 10 mostly used of these 'show' commands.

1. `show running-config`
   This command shows the complete configuration that is running currently.
   Using it you can troubleshoot almost all issues regarding routing, filtering, secure access, and many other issues. Using it before you start configuring the router would give you a clear idea of what services and protocols are operating by default and which are turned off by default.
   If you think it is too much information, you can show parts of the running configuration like the following examples:
   ```
   Router#sh run | begin int
   Router#sh run | section rip
   ```
2. `show startup-config`
   This command shows the configuration that is saved on the NVRAM. It is helpful in knowing the configuration that will be applied the next time the router is reloaded. This command also comes handy if you need to know the configuration that was loaded at the startup of the router before you made changes to it.
3. `show interface`
   This command shows status and statistics of interfaces. This command is almost always needed in troubleshooting routing and link issues. Information shown using this command includes interface IP address and subnet mask, interface status, encapsulation type, bandwidth, and many other important indicators about the interface operation.
4. `show ip route`
   This command shows the routing table. This table helps you in finding out the next hop for each and every routable packet. It is the first indicator to point a problem in routing.

5. `show ip protocols`
   This command shows the active routing protocols on the router and what networks are these protocols advertising. It also shows the sources of routing updates received at this router. It is very useful in troubleshooting routing issues.
6. `show access-list`
   This command shows the contents of each access-list. It is very useful in troubleshooting filtering issues. Note that this command does not show you where each access-list is applied.
7. `show ip interface`
   This command displays information about IP protocol and the interface. This command shows which access-lists are applied at the interfaces and in which direction. This kind of information is not shown by the '`show access-list`' command. However, you can find out which access-list is applied where by using '`show run`' also.
8. `show cdp neighbor detail`
   This command displays detailed information about the neighbouring devices such as IP addresses, platforms, and hostnames. This command can be useful in troubleshooting connectivity issues and can also be used in finding out how devices are connected to each other when you have no clearly-drawn network map.
   This command assumes that the CDP protocol is running. Many security procedures recommend that you do not enable CDP protocol. It can be useful to switch it on for a short period of time to gather the required information and then switch it off.
9. `show version`
   This command shows detailed information about the IOS. It shows the file name of the IOS along with the version of the IOS and value of the configuration register. The configuration register is a set of bits that controls the boot sequence of the router. This command is the only command used to show this register's value.
10. `show flash` or `show slot0`
    This command is used to view the contents of the flash, the size of the IOS file (s), and the size of the flash and how much of it is free. It is necessary in upgrading or installing IOS files.

9.3 How to Simulate Break Key Sequence in a Cisco Router

When would you need this: When you are recovering a lost password and the required 'Ctrl-Break' key combination is not working.

Special Requirements: None.

Table 9.1 Boot sequence interruption key combinations

Terminal emulation software	Operating system	Key combination
HyperTerminal	Windows 7, Vista, XP, and Server	Ctrl-Break
Kermit	Unix	Ctrl-\I or Ctrl-\B
Minicom	Linux	Ctr-A F
SecureCRT	Windows	Ctrl-Break
Telnet	–	Ctrl-] then type `send break`
Z-Terminal	Mac	Command-B

First of all, you have to make sure that you are pressing the correct key sequence. There are few, slightly different, keys to press to break the router boot sequence in different routers and different terminal emulation software. Table 9.1 shows a list of different keystrokes to interrupt the router boot sequence.

The auxiliary (AUX) port is not active during the boot sequence of a router. Therefore, it is of no use to send a break through the AUX port. You need to have connection to the console port and have these default settings:

- Baud rate: 9600
- Parity: None
- Data bits: 8
- Stop bits: 1
- Flow control: None

Until here, things are supposed to be going smooth. If you have everything set right, and you press the correct key strokes during router initialization (within the first 60 s of router startup), you will be transferred to the ROM Monitor mode.

- If the above is not working, you might consider the following notes:
- If you are using the HyperTerminal of Windows NT, you might consider upgrading the HyperTerminal. Some versions of Windows NT have Hyper Terminal software that cannot send the correct Break Key signal. My personal recommendation is a software called ‘PuTTY’.
- If you are using a DB9-to-USB converter to connect to the console port, you might need to connect to a DB9 port directly. Not all converters of this type can convey the correct break sequence.
- If you still do not know the exact reason why this is not working, you should consider simulating the Break Key sequence.

To simulate the Break Key sequence, go through the following steps carefully:

1. Connect to the router with these terminal settings:

    - Baud rate: 1200
    - Parity: None

- Data bits: 8
- Stop bits: 1
- Flow control: None

   You will no longer see any output on your screen, and this is normal.

2. Power cycle (switch off and then on) the router and press the Spacebar for 10–15 s in order to generate a signal similar to the break sequence.
3. Disconnect your terminal and reconnect with a 9600 baud rate. You enter the ROM Monitor mode.

   If all of this fails, you should consider trying a different PC or Emulation software.

9.4 How to Recover Cisco 2600 Routers Password

When would you need this: When you forget the secret, enable, or console password of a 2600-series Cisco Router.

Special Requirements: None.

1. Interrupt the router booting operation. This is done by pressing (Ctrl-Break) key simultaneously as soon as you turn on the router. This step will get you to the ROM monitor mode (ROMMON).

   You will see something similar (not necessarily identical) to the following:

   ```
   System Bootstrap, Version 11.3(2)9A4, RELEASE SOFTWARE
   (fc1)
   Copyright (c) 1999 by cisco Systems, Inc.
   TAC:Home:SW:IOS:Specials for info
   PC = 09fff0a530, Vector = 09500, SP = 09680127b0
   C2600 platform with 32768 Kbytes of main memory
   PC = 09fff0a530, Vector = 09500, SP = 0980004374 monitor:
   command "boot" aborted due to user interrupt
   rommon 1>
   ```

   The (`rommon 1>`) prompt is for the ROM monitor mode. If you are having a problem interrupting the boot sequence of the router, you might be interested in the previous procedure to simulate Break Key sequence in Sect. 9.3.

2. Now you should change the value of the configuration register in order to make the router neglect the contents of the NVRAM in the next boot up. This is achieved using the following command:

   ```
   rommon 1>confreg 0x2142
   ```

   This command will change the sixth bit (originally the configuration register is 0x2102) to one. By doing so, the router will ignore the startup configuration in the next boot despite the fact that the startup configuration is not erased.

3. Perform a restart to the router using the following command:

   ```
   rommon 2>reset
   ```

4. The router will now restart and ask you if you want to use the set-up mode; choose no. Now, in order not to lose the configuration that you already have in the router, you should go to the privileged mode and perform:
   ```
   Router#copy start run
   ```
   This will get you back your old configuration but with one exception, you already are in the privileged mode without having to know the password.
   Now you choose a new password or passwords if you may:
   ```
   Router(config) #enable secret your-new-password
   ```
   You can also put new console and Telnet passwords if necessary.
5. To get things going back to normal, change the value of the configuration register to its original form (0x2102) using the following global configuration command:
   ```
   Router(config)#config-register 0x2102
   ```
6. Save the configuration including the new passwords that you know:
   ```
   Router#copy run start
   ```
7. Reload and you are good to go:
   ```
   Router#reload
   ```

9.5 How to Recover Cisco 2500 Routers Password

When would you need this: When you lose the secret, enable, or console password of a 2500 Cisco Router.

Special Requirements: None.

1. Interrupt the router booting operation. This is done by pressing (Ctrl-Break) keys simultaneously as soon as you turn on the router. This step will get you to the ROM monitor mode (ROMMON).
   You will see output similar (but not necessarily identical) to the following:
   ```
   System Bootstrap, Version 11.0(10c), SOFTWARE
   Copyright (c) 1986–1996 by cisco Systems
   2500 processor with 14336 Kbytes of main memory
   Abort at 091098FEC (PC)
   >
   ```
   The (>) prompt is for the ROM monitor mode. If you are having a problem interrupting the boot sequence of the router, take a look into the procedure to simulate Break Key sequence in Sect. 9.3.
2. Change the value of the configuration register in order to make the router neglect the contents of the NVRAM in the next boot up. This is achieved using the following command:
   ```
   >o/r 0x2142
   ```
   This command will change the sixth bit (originally the configuration register is 0x2102) to one. By doing so, the router will act as new in the next boot, i.e., the router will not look for the startup configuration in the NVRAM. The startup configuration will not be erased.

3. Perform a restart to the router using the following command:

   ```
   >i
   ```

   The (i) stands for (initialize).
4. The router now will restart and ask you if you want to use the setup mode; choose no. Now, in order not to lose the configuration that you already have in the router, you should go to the privileged mode and perform:

   ```
   Router#copy start run
   ```

   This will get you back your old configuration but with one exception, you already are in the privileged mode without having to know the password. Now you put a new password:

   ```
   Router(config) #enable secret your-new-password
   ```

   You can also put new console and Telnet passwords.
5. To get things back to normal, change the value of the configuration register to its original form (0x2102) using the following global configuration command:

   ```
   Router(config) #config-register 0x2102
   ```

6. Now you should save the configuration including the new passwords that you know:

   ```
   Router#copy run start
   ```

7. Now reload and you are good to go:

   ```
   Router#reload
   ```

9.6 How to Disable ROMMON Password Recovery in a Cisco Router

When would you need this: When you need to protect your router configuration from being seen by strangers.

Special Requirements: None.

There is no long configuration procedure here. It is a single command:

```
Router(config)#no service password-recovery
```

This command will prevent the user from accessing ROMMON for password recovery purposes like the two procedures we discussed earlier. When this is done, the only way of password recovery will be by erasing the startup configuartion.

If you issue the aforementioned command, save the settings, restart the router, and press Ctrl-Break, you will see a message similar to this:

```
PASSWORD RECOVERY IS DISABLED
Do you want to reset the router to factory default configuration and proceed [y/n]?
```

The command will also prevent you from changing the configuration register value to 0x2142.

To re-enable ROMMON password recovery:

```
Router(config)#service password-recovery
```

9.7 How to Use a Cisco Router as a Packet Sniffer

When would you need this: When you have an unexplainable packet loss when trying to connect to a remote location and you cannot figure out whether it is the LAN or the WAN that has the issue.

Special Requirements: Cisco IOS 12.4(20)T or later.

1. Create an access-list to select the specific type of traffic that you want to capture (for example, ICMP traffic for PING purposes).

   ```
   Router(config)#access-list acl-number permit icmp host host1-address host host2-address
   Router(config)#access-list acl-number permit icmp host host2-address host host1-address
   ```

   where

   acl-number is the number of the access-list. This is an extended access-list so the number must be from 100 to 199.

   host1-address the IP address of the first host

   host2-address the IP address of the second host
2. Enable packet capturing feature in the router and tell the router to capture the traffic that meets the conditions of access-list number *acl-number*.

   ```
   Router#monitor capture buffer buffer-name filter access-list acl-number
   ```

   where *buffer-name* is the name that you choose for the buffer and *acl-number* is the number of the access-list created in the previous step.

   Now you should get a message ‘Filter Association Succeeded’.
3. At this point, the router has not started capturing packet just yet. We need to create a capture point and associate it with the capture buffer that we created in the previous step.

   ```
   Router#monitor capture point ip cef point-name all both
   Router#monitor capture point associate point-name buffer-name
   ```

   where the point-name is the name of the capture point and the buffer-name is the same one created in step 2.
4. Start the capture using the following command:

   ```
   Router#monitor capture point start point-name
   ```
5. Ping from Host1 to Host2:

   ```
   Router#ping host2-address repeat 4 timeout 1
   ```
6. To show the captured packet in a specific buffer:

   ```
   Router#show monitor capture buffer buffer-name dump
   ```
7. To show all the capture buffers:

   ```
   Router#show monitor capture buffer all parameters
   ```

   And all the capture points:

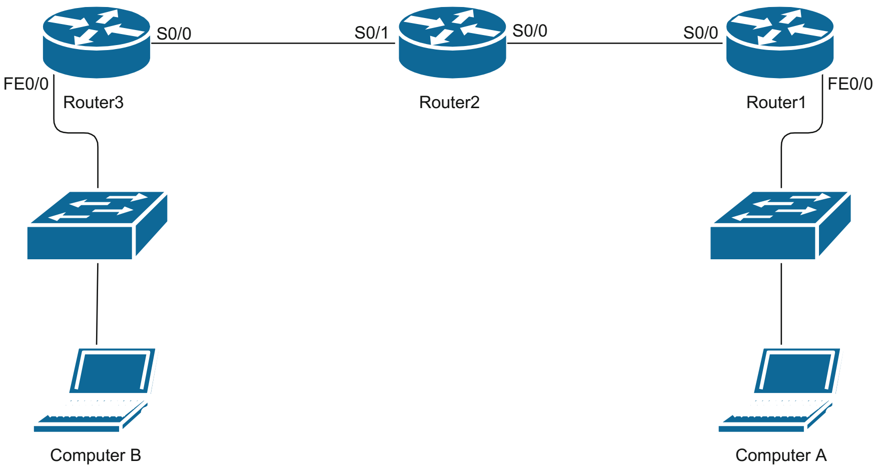
   ```
   Router#show monitor capture point all
   ```

8. To stop capturing:

```
Router#monitor capture point stop point-name
```

9.8 Training Scenarios

Scenario 9.1



Connect the network shown in the figure above and configure the following settings:

1. On Router 1, do the basic configuration:

Device	Parameter	Value
Router 1	Hostname	Router1
	Console password	Cisco Console
	Secret password	Cisco
	VTY password	CiscoVTY
	Interface FE0/0 IP address	172.16.0.254/24
	Interface S0/0 IP address	10.0.0.1/30

2. On Router 2, do the basic configuration:

Device	Parameter	Value
Router 2	Hostname	Router2
	Console password	Cisco2Console
	Secret password	Cisco2
	VTY password	Cisco2VTY
	Interface S0/0 IP address	10.0.0.2/30
	Interface S0/1 IP address	10.0.1.1/30

3. On Router 3, do the basic configuration:

Device	Parameter	Value
Router 3	Hostname	Router3
	Console password	Cisco3Console
	Secret password	Cisco3
	VTY password	Cisco3VTY
	Interface FE0/0 IP address	172.16.1.254/24
	Interface S0/0 IP address	10.0.1.2/30

4. On the computers, change the following settings:

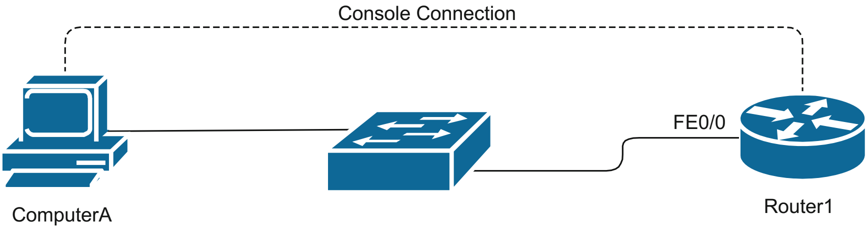
Device	Parameter	Value
Computer A	IP address	172.16.0.1
	Subnet mask	255.255.255.0
	Default-gateway	172.16.0.254
	DNS server 1	172.16.0.254
	DNS server 2	172.16.0.254
Computer B	IP address	172.16.1.2
	Subnet mask	255.255.255.0
	Default-gateway	172.16.1.254
	DNS server 1	172.16.1.254
	DNS server 2	172.16.1.254

5. Configure EIGRP Dynamic Routing on Router 1 with the advertised networks 10.0.0.0 and 172.16.0.0 in autonomous system number 10.
6. Configure EIGRP Dynamic Routing on Router 2 with the advertised network 10.0.0.0 in autonomous system number 10.
7. Configure EIGRP Dynamic Routing on Router 3 with the advertised networks 10.0.0.0 and 172.16.1.0 in autonomous system number 10.
8. For testing, PING from Computer A to C and D and show the routing tables in both routers.
9. Another test would be using Router 2 to capture packets flowing from using the 'Embedded Packet Capture' functionality. This functionality will enable you to capture the packets and save them as a small PCAP file that can be extracted from the router and read using a software like WireShark.
   Once you capture the packets and read their content, you can easily see that the content is not encrypted.
10. Set up a site-to-site VPN between Routers 1 and 3 using the following settings:
    VPN key: CiscoKey@123
    Transformation Set name: CiscoTranform
    Transformation Set: esp-3des esp-sha-hmac

Crypto map name: MyCryptoMap
Crypto map priority: 7

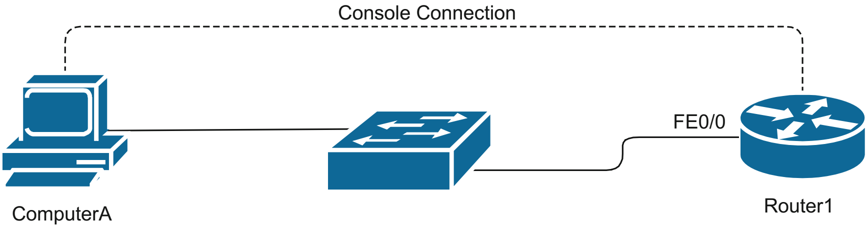
11. Redo the capturing of packets done in step 9. The data in all packets flowing from Computer A to Computer B should be encrypted.

Scenario 9.2



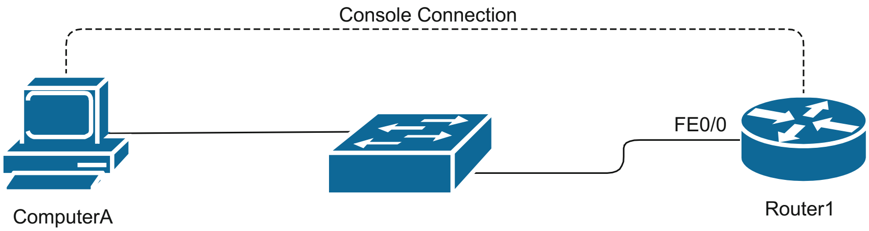
Connect the network shown in the figure above. The model of the router is 2601. The network administrator has forgotten the secret password configured on the router. You are required to invoke the password recovery procedure to regain access to the router's configuration modes.

Scenario 9.3



Connect the network shown in the figure above. The model of the router is 2505. The network administrator has forgotten the secret password configured on the router. You are required to invoke the password recovery procedure to regain access to the router's configuration modes.

Scenario 9.4



Connect the network shown in the figure above. The model of the router is 2505. The network administrator thinks that he will never forget the secret password of the router. The administrator has asked you to disable password recovery procedure so no one can change the settings of the router unless they have proper privilege.

Further Guidance

Configuring RIP. http://www.cisco.com/en/US/docs/ios/12_0/np1/configuration/guide/1crip.html

Configuring EIGRP. http://www.cisco.com/en/US/docs/ios/12_0/np1/configuration/guide/1ceigrp.html

Configuring OSPF. http://www.cisco.com/en/US/docs/ios/12_0/np1/configuration/guide/1cospf.html

Configuring Integrated IS-IS. http://www.cisco.com/en/US/docs/ios/11_3/np1/configuration/guide/1cisis.html

Per-Packet Load Balancing. http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/pplb.html

Cisco IOSDHCP Server. http://www.cisco.com/en/US/docs/ios/12_0t/12_0t1/feature/guide/Easyip2.html

Configurable DHCP Client. http://www.cisco.com/en/US/docs/ios/12_3t/12_3t8/feature/guide/gtdhcpcf.html

Configuring Network Address Translation: Getting Started. http://www.cisco.com/en/US/tech/tk648/tk361/technologies_tech_note09186a0080094e77.shtml

Configuring InterVLAN Routing and ISL/802.1Q Trunking on a Catalyst 2900XL/3500XL/2950

Switch Using an External Router. http://www.cisco.com/en/US/tech/tk389/tk815/technologies_configuration_example09186a00800949fd.shtml

Cisco DSL Router Configuration and Troubleshooting Guide. http://www.cisco.com/en/US/tech/tk175/tk15/technologies_configuration_example09186a008015407f.shtml

Understanding and Configuring PPP CHAP Authentication. http://www.cisco.com/en/US/tech/tk713/tk507/technologies_tech_note09186a00800b4131.shtml

HDLC Back-to-Back Connections. http://www.cisco.com/en/US/tech/tk713/tk317/technologies_configuration_example09186a00800944ff.shtml

Configuring ISDN BRI. http://www.cisco.com/en/US/docs/ios/dial/configuration/guide/dia_cfg_sdn_bri_ps6350_TSD_Products_Configuration_Guide_Chapter.html

Configuring ISDN DDR with Dialer Profiles. http://www.cisco.com/en/US/tech/tk801/tk133/technologies_configuration_example09186a0080093c2e.shtml

Cisco–Configuring Frame-Relay Switching. http://www.cisco.com/warp/public/125/fr_switching.pdf

White Paper: Cisco IOS and NX-OS Software Reference Guide. http://www.cisco.com/web/about/security/intelligence/ios-ref.html

Backup and Restore of Configuration Files. http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_tech_note09186a008020260d.shtml

Software Upgrade Procedure. http://www.cisco.com/en/US/products/ps5855/products_tech_note09186a00801fc986.shtml

How To Copy a System Image from One Device to Another. http://www.cisco.com/en/US/products/hw/routers/ps233/products_tech_note09186a00800a6744.shtml

How to Upgrade from ROMmon Using the Boot Image. http://www.cisco.com/en/US/products/hw/routers/ps214/products_tech_note09186a0080110ed1.shtml

M.M. Alani, *Guide to Cisco Routers Configuration*,
DOI 10.1007/978-3-319-54630-8

How to Partition Internal Flash Memeory. https://supportforums.cisco.com/docs/DOC-4062

Configuring IP Access Lists. http://www.cisco.com/en/US/products/sw/secursw/ps1018/products_tech_note09186a00800a5b9a.shtml

Configuring Secure Shell on Routers and Switches Running Cisco IOS. http://www.cisco.com/en/US/tech/tk583/tk617/technologies_tech_note09186a00800949e2.shtml

Cisco IOS VPN Configuration Guide: Site-to-Site and Extranet VPN Business Scenarios. http://www.cisco.com/en/US/docs/security/vpn_modules/6342/configuration/guide/6342site3.html

Standard Break Key Sequence Combinations During Password Recovery. http://www.cisco.com/en/US/products/hw/routers/ps133/products_tech_note09186a0080174a34.shtml

Password Recovery Procedure for the Cisco 2600 and 2800 Series Routers. http://www.cisco.com/en/US/products/hw/routers/ps259/products_password_recovery09186a0080094675.shtml

Password Recovery Procedure for the Cisco 2000, 2500, 3000, 4000, AccessPro, 7000 (RP),
AGS, IGS, and STS-10x. http://www.cisco.com/en/US/products/hw/routers/ps233/products_password_recovery09186a0080094795.shtml

Configuring Multicast. http://www.cisco.com/c/en/us/td/docs/ios/12_2/ip/configuration/guide/fipr_c/1cfmulti.html

Configuring MPLS. http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/mp_basic/configuration/xe-16/mp-basic-xe-16-book.html

Configuring HSRP. http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst3560/software/release/12-2_52_se/configuration/guide/3560scg/swhsrp.html

Configuring Reflexive ACLs. http://www.cisco.com/c/en/us/td/docs/ios/12_2/security/configuration/guide/fsecur_c/scfreflx.html

Configuring Timed ACLs. http://www.cisco.com/c/en/us/support/docs/security/ios-firewall/23602-confaccesslists.html

Configuring GRE. http://www.cisco.com/c/en/us/tech/ip/ip-tunneling/tech-configuration-examples-list.html

Configuring Multiple Area OSPF. http://www.cisco.com/c/en/us/support/docs/ip/open-shortest-path-first-ospf/118879-configure-ospf-00.html