Career Path Series

On the Command Line

The top **100** commands and configurations used in network engineering

```
DHCP-VoIP-Router(config-subif)#ip address 10.10.99.1
DHCP-VoIP-Router(config-subif)#description VoIP-vlan
DHCP-VoIP-Router(config-subif)#exit
DHCP-VoIP-Router(config)#telephony-service
DHCP-VoIP-Router(config-telephony)#max-dn 2
DHCP-VoIP-Router(config-telephony)#max-ephones 2
DHCP-VoIP-Router(config-telephony)#auto-reg-ephone
DHCP-VoIP-Router(config-telephony)#ip source 10.10.9
```

J. Diamond

On the Command Line

The top 100 commands and configurations used in network engineering

First Edition August 2021

Published through Amazon KDP Store

10 9 8 7 6 5 4 3 2 1

Contents (each configuration numbered)

17. Security: Show privilege level
18. Security: Turn on Encryption to encrypt <u>all</u> existing and future passwords
19. Security: Configure a *Secret* Enable password using MD5 hashing
20. Security: Configure a Username and Password using the Authentication / AAA new model
21. Security: Configure a Telnet password for remote login between local devices
22. Security: SSH for remote login using AAA and custom crypto keys
23. Security: AAA, TACACS, and RADIUS
24. Banner: Display a Message of the Day (MotD)
25. Banner: Display Login Message
26. Set device clock: local device time, day, month, and year
27. NTP (Network Time Protocol)
28. NTP (Network Time Protocol): Stratum
29. NTP (Network Time Protocol): Show NTP Associations
30. NTP (Network Time Protocol): Show NTP Status
31. Assign a device to an IP Domain
32. Configure a Loopback address on a device
33. DHCP: Configure a DHCP server on a router or switch
34. DHCP: Configure a DHCP Relay Agent using the IP-helper address command

35. DHCP: Show IP DHCP Binding
36. DHCP: Show IP DHCP Pool
37. DHCP: Security / Snooping and Trusted Ports
38. Configure an IP default-gateway address on a device
39. Configure a static route to a specific network
40. Configure a gateway of last resort on a router
41. VLANs: Configure a VLAN on a switch
42. VLANs: Configure a trunk port on a switch using the Switchport command for RoaS
43. VLANs: Configure a sub-interface on a router to use RoaS
44. VLANs: Configure an access port on a switch using the Switchport command
45. VLANs: Configure switches to use VTP
46. VLANs: Configuring a Native VLAN
47. VLANs: Moving unused ports out of Default VLAN1 for added security
48. VLANs: Show VLAN brief to verify which switch ports are assigned to which VLANs
49. VLANs: Show MAC-Address Table
50. VLANs: Show VLAN ID / Show VLAN Name
51. VLANs: Show Interface Trunk and Show Interface X Status
52. VLANs: Show Interface X Switchport

53. Logging: Configure automated report logging for a sys-log server
54. Logging: Configure timestamps to be included with sys-log automated report logging
55. Logging: Configure timestamps to be included with real time on-screen debug notifications
56. Routing protocol: RIPv2
57. Routing protocol: OSPF (Open Shortest Path First)
58. Routing protocol: OSPF / Show IP OSPF Neighbors
59. Routing protocol: OSPF / Show IP OSPF Database
60. Routing protocol: OSPF / Stub areas and NSSA Explained
61. Routing protocol: OSPF / Simple Fastest Route Formula
62. Routing protocol: EIGRP (Enhanced Interior Gateway Routing Protocol)
63. Routing protocol: BGP (Border Gateway Protocol)
64. Routing protocol: BGP (Border Gateway Protocol) / Ship IP BGP
65. Routing protocol: BGP / Single Homed, Double Homed, and Multi-Homed Topologies
66. Routing protocol: GRE (Generic Routing Encapsulation) Tunnel
67. Routing protocol: GRE / Tunnel Security / ISAKMP and IPSEC
68. Routing protocol: HSRP (Hot Standby Router Protocol)
69. Routing protocol: CEF (Cisco Express Forwarding)
70. Routing protocol: Route Redistribution

71. Routing protocol: MPLS (Multi-Protocol Label Switching)
72. IPv6 routing: Basic setup
73. IPv6 routing: Show IPv6 Interface Brief
74. IPv6 routing: Static route configuration
75. SNMP: Simple Network Management Protocol
76. Local SPAN
77. Remote SPAN
78. QoS: (Quality of Service) part 1 of 2: Matching values with NBAR
79. QoS: (Quality of Service) part 2 of 2: Matching values with DSCP
80. QoS: DSCP (Differentiated Source Code Point)
81. QoS: (Quality of Service / Show Policy-Map
82. PPP (Point-to-Point Protocol)
83. PPP with PAP (Password Authentication Protocol)
84. PPP with CHAP (Challenge Handshake Authentication Protocol)
85. PPP (Point-to-Point Protocol) Multi-Link
86. Ether-Channel
87. Ether-channel: Show Etherchannel Port-Channel
88. Ether-channel: Show Etherchannel Summary

89. Ether-channel: Show Etherchannel and Show Etherchannel Load-Balancing
90. ACL (Access List) Standard
91. ACL Best Practices
92. ACL (Access List) Extended
93. NAT (Network Address Translation) Static NAT
94. NAT (Network Address Translation) Dynamic NAT
95. NAT (Network Address Translation) Overload a.k.a. PAT (Port Address Translation)
96. VOIP (Voice Over IP)
97. VOIP (Voice Over IP) Show Ephone
98. Wireless: Basic setup of a wireless network and WLC
99. Wireless: WLC setup verification on the Monitor tab
100. Wireless: WLAN and Group configuration

Introduction

Welcome. As the title says, this book contains the 100 most used commands and configurations in network engineering. All configurations are shown complete, accompanied by real screenshots, and diagrams. This book is based on Cisco devices. However, many manufacturers today share previously proprietary commands and configurations.

Who this book is for

This book is for you: the student or current network admin or engineer.

How to use this book

Contents number commands and configurations (instead of pages). Simply locate the command or configurations you want to practice, refer to the easy-to-read network diagrams, and follow the examples as shown. It is proven that reading, note taking, and hands-on practice is the best way to learn. Many commands are used in their abbreviated form. Learn these as well because it will save you time on your exams and in your day-to-day work. Among these are: "**sh**" for "show" and "**conf t**" for "configure terminal".

Thank you for your purchase, and I hope you find this book useful in your career. JD

1. Show Running-Config

This basic command is often the first one learned. Abbreviated as “sh run” this command returns a long report of the device’s current running configuration. **Important note for students**: this command returns a lot of general information and is sometimes disabled on exams to force students to use more specific show commands.

```
Router1>en
Router1#sh run
```

2. Write running-config

This command is used to save configuration changes to the device’s running-config.

Router1#wr

Building configuration…

[ok]

Note: Many configurations can be deleted with the “no” command.

3. Show History / Line, VTY, and Aux.

This show command returns a list of all commands recently issued on the device (as shown below). When enabled, a history buffer will save exec commands issued via each of the interfaces: the console (Line), remotely (VTY), and through the auxiliary (Aux) port.

```
Router>en
Router#sh history
```

```
Router#sh history
  en
  sh clock
  conf t
  sh prot
  sh run
  sh ip int br
  sh history
Router#
```

Note: The history buffer resets for each new login. It can record a maximum of 256 events.

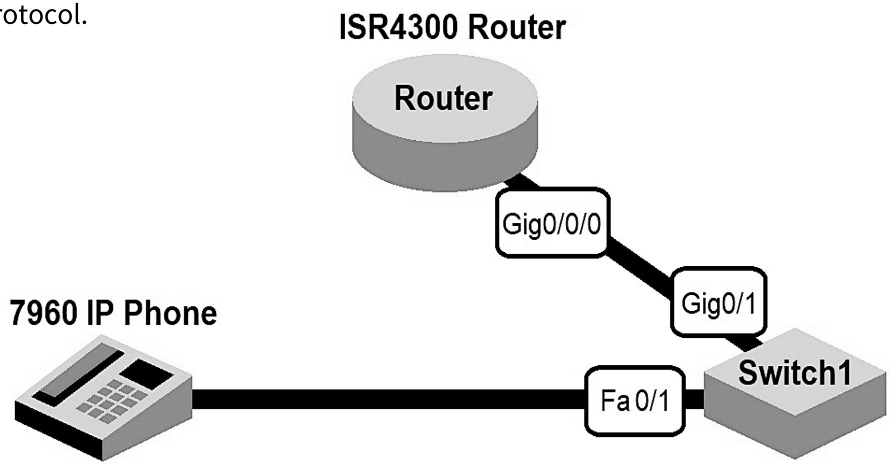
4. Configure device History buffer / Line, VTY, and Aux.

The device history buffer can be reconfigured to save 0 thru 256 events. **Note:** setting it to zero means that no command history will be stored. This does not technically disable terminal history. To do that, configure using the “no” command to disable it. History is saved per session, so it clears upon logging off.

```
Router1>en
Router1#conf t
Router1(config)#line con 0
Router1(config-line)#history size ?
<0-256> Size of history buffer
Router1(config-line)#history size 100
Router1(config-line)#exit
Router1(config)#exit
Router1#
```

5. Show CDP Neighbors

The **Cisco Discovery Protocol** (CDP) is used to discover and identify other connected devices. CDP is a Cisco proprietary protocol. **LLDP** (Link Layer Discovery Protocol) is another device discovery protocol. LLDP is a vendor-neutral protocol.



Note: some devices require CDP (or LLDP) be manually enabled before it can be used. On Cisco devices this is accomplished in global configuration mode with the command “cdp run” or “LLDP run”.

Using the **Show CDP Neighbors** command on the switch in the above sample topology reveals the router and the IP phone. Notice the codes which indicate the type of device. Local interface is the port on the device, and Port ID is the port on the neighbor. Capability tells what the device's basic capabilities are. The IP phone is both, a phone and a host device because vlan traffic may pass through it to a PC on the same access link.

```
Switch1>en
Switch1#sh cdp ne
```

```
Switch1>en
Switch1#sh cdp ne
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone
Device ID     Local Intrfce    Holdtme     Capability    Platform     Port ID
Router        Gig 0/1           152             R        ISR4300      Gig 0/0/0
IP Phone      Fas 0/1           171             H P      7960
Switch1#
```

6. Configure an IPv4 address on an interface

```
Router1#conf t
Router1(config)#int g0/0/0
Router1(config-if)#ip address 192.168.1.1 255.255.255.0
Router1(config-if)#no shut
Router1(config-if)#exit
Router1(config)#exit
```

7. Show IP Interface Brief

This command returns a brief report showing the types of interfaces, IP addresses, method, and current operational and protocol status.

```
Router1#sh ip int br
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0/0   192.168.1.1     YES manual up                    up
GigabitEthernet0/0/1   unassigned      YES unset  administratively down down
Vlan1                  unassigned      YES unset  administratively down down
Router1#
```

8. Configure a new host name for a device

```
Router> en
Router# conf t
Router(config)# hostname Router1
Router1(config)# exit
Router1#
```

9. Configure console (Line) Exec-Timeout of 0

This sets device login time (time-out) via the console port to infinite. Meaning that the user will not be logged off automatically after idle. Useful when tasks require prolonged periods of login time.

```
Router1>en
Router1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router1(config)#line con 0
Router1(config-line)#exec-timeout 0
Router1(config-line)#exit
Router1(config)#exit
Router1#
```

10. Disable DNS Lookup

Another useful configuration. **Disable DNS Lookup** does <u>not</u> disable a DNS server, but instead disables a device's automatic domain name lookup of any unrecognized names *beyond that* of any DNS server available or that already exists on the network. This prevents a device from ceasing all activity to lookup a mistyped command as if it were a non-existent domain name, which can take seconds or minutes to complete depending on the size of the network.

```
Router1> en
Router1# conf t
Router1(config)# no ip domain lookup
Router1(config)# exit
Router1#
```

11. Security: Port security / Shut unused ports using the Int Range command

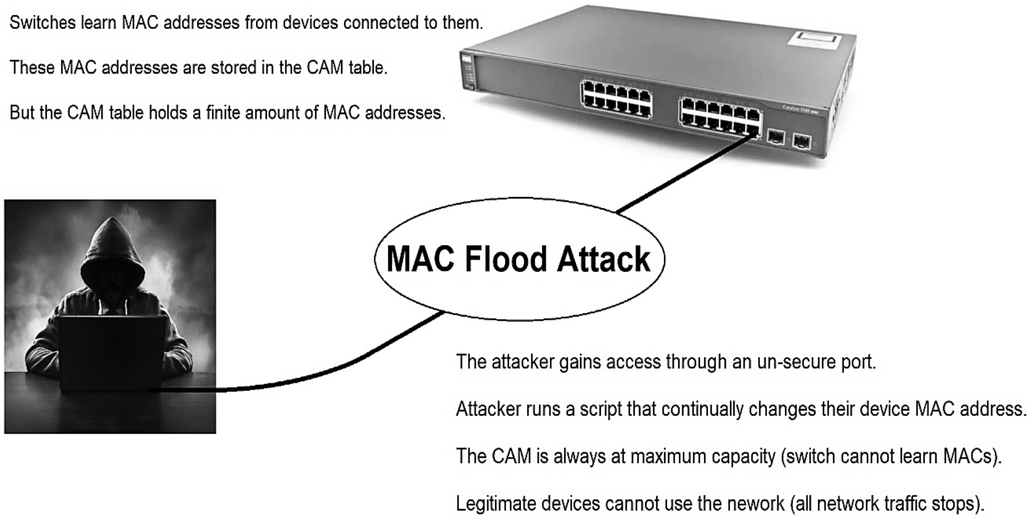
This command is used for hardening networks against unauthorized access by rogue devices by shutting down unused interfaces. In this example, let's say we are using Fast Ethernet ports f0/1 and f0/2. But ports 0/3 thru 0/24 are not being used. We should shut those unused ports. But instead of shutting them one at a time, we can use the Int Range command as shown below. Note: this is especially important for switches because by default all their ports are in the no shut position.

```
Switch1#conf t
Switch1(config)#int range f0/3-24
Switch1(config-if-range)#shut
Switch1(config-if-range)#exit
Switch1(config)#exit
Switch1#
```

Note: Although any network device's ports can be shut, switches are of particular concern because all their ports are by default in the no-shut position (open).

12. Security: Port security / Port Security Settings

To further protect a network against malicious activities, **Port Security** is recommended as it restricts a port to recognizing and allowing access to only specific, predetermined MAC addresses. This is of particular use in public access areas and for protecting wi-fi networks vs. cyberattacks. One such example is MAC flooding (shown here).



To combat MAC flooding (and other cyberattacks) port security configurations are critically important. Thankfully, there is a nice variety pf options that can be combined to meet the needs of your network.

Enables port-security on the device. Some manufacturers require manually turning on access to port security features.

```
Switch1>en
Switch1#conf t
Switch1(config)#int f0/1
Switch1(config-if)#switchport mode access
Switch1(config-if)#switchport port-security
Switch1(config-if)#switchport protected
Switch1(config-if)#switchport port-security maximum 2
Switch1(config-if)#switchport port-security mac-address sticky
Switch1(config-if)#switchport port-security mac-address aging time 30
Switch1(config-if)#switchport port-security violation restrict
Switch1(config)#exit
Switch1#
```

Port Security Options

Maximum	MAC address	Violation	Aging
1 - 132 The maximum number of MAC addresses that can be assigned to a port	H.H.H. Static MAC address assignment to a port	Protect This option disallows any unauthorized MAC from using the port	1 - 1440 Idle or Absolute port timer for authorized MACs (in minutes)
	Sticky Port learns the MAC address on first use and saves it as authorized to use the port	Restrict Similar to Protect with the addition of starting a violation reset timer	
		Shutdown Unauthorized MAC initiates port shutdown and err-disabled state	

13. Security: Port security / Port Security Settings / Show port-security and **Show port-security interface**

Port security setting relating to maximum MAC addresses is verified after connecting a PC to Fa0/1. We can see that 1 of 2 MACs (maximum) are now authorized to use this port.

```
Switch1#sh port-security
Secure Port MaxSecureAddr CurrentAddr SecurityViolation Security Action
               (Count)       (Count)          (Count)
--------------------------------------------------------------------
        Fa0/1        2           0                 0         Restrict
----------------------------------------------------------------------
Switch1#sh port-security int f0/1
Port Security              : Enabled
Port Status                : Secure-down
Violation Mode             : Restrict
Aging Time                 : 5 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 2
Total MAC Addresses        : 0
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0000.0000.0000:0
Security Violation Count   : 0

Switch1#
```

We can further define the parameters of port security by choosing specific conditions to initiate an interval during which time the port will wait between attempts to reset (if it can) after a rule violation.

```
Switch1>en
Switch1#conf t
Switch1(config)#errdisable recover cause psecure-violation
Switch1(config)#errdisable recover interval ?
  <30 - 86400> timer-interval (sec)
Switch1(config)#errdisable recover interval 45
Switch1(config)#exit
Switch1#
```

14. Security: Port security / Errdisable Recovery

This command verifies the configuration.

```
Switch1#sh errdisable recovery
ErrDisable Reason              Timer Status
-----------------              --------------
arp-inspection                 Disabled
bpduguard                      Disabled
channel-misconfig (STP)        Disabled
dhcp-rate-limit                Disabled
dtp-flap                       Disabled
gbic-invalid                   Disabled
inline-power                   Disabled
l2ptguard                      Disabled
link-flap                      Disabled
mac-limit                      Disabled
loopback                       Disabled
pagp-flap                      Disabled
port-mode-failure              Disabled
pppoe-ia-rate-limit            Disabled
psecure-violation              Enabled
security-violation             Disabled
sfp-config-mismatch            Disabled
small-frame                    Disabled
storm-control                  Disabled
udld                           Disabled
vmps                           Disabled
psp                            Disabled

Timer interval: 30 seconds

Interfaces that will be enabled at the next timeout:

Switch1#
```

15. Security: Configure a simple Console Port login password (in plain text)

In this example the word "ADMIN" is configured as the password to be used whenever a user wants to log into this switch locally via the console port. This password is stored in plain text. More secure console port password configurations follow on the next page. Note: this password is case sensitive.

```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#line con 0
Switch(config-line)#password ADMIN
Switch(config-line)#login
Switch(config-line)#exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#wr
Building configuration...
[OK]
Switch#
```

16. Security: Configure an Enable password to elevate privilege level (in plain text)

An Enable password is used to securely elevate a user from the default level, level-1 privileges, to the highest level, administrative level-15. In other words, “enabling” a higher level of privilege. Note: this password is case sensitive. **Note:** there *are* privilege levels between 1 and 15. However, levels 1 and 15 are the most often used.

```
Router>en
Router1#conf t
Router1(config)#enable password cisco123
Router1(config)#exit
Router1#
```

17. Security: Show privilege level

```
Router1>en
Router1#sh priv
```

```
Router1>sh priv
Current privilege level is 1
Router1>en
Router1#show priv
Current privilege level is 15
Router1#
```

18. Turn on Encryption to encrypt all existing and future passwords

Encryption produces a cypher text of the password. However, this is not considered the best method of safeguarding passwords. **Note:** once configured and saved, stronger passwords replace weaker ones.

```
Router1>en
Router1#conf t
Router1(config)#service password-encryption
Router1(config)#exit
Router1#
```

19. Enable Configure a *Secret* Enable password using MD5 hashing

```
Router1#conf t
Router1(config)#enable secret ADMIN
Router1(config)#exit
Router1#
```

Security Comparison **MD7 Encryption** vs. **MD5 Hash**

Enable password is: **admin123**

MD7 Encryption: 7 0820484300175445 41

MD5 Hash: 5 1mERr$AFX/pZT1Lh7NP3Dp3P/qq/

How is Hashing Different from Encryption?

Encryption is a two-way function. Meaning that the data is meant to be encrypted and then decrypted at some later time. Thus, it is good for storing and transporting all kinds of data.

Hashing on the other hand, differs from encryption because hashing is a <u>one</u>-way function. Meaning that it is <u>not</u> meant to be decrypted (de-hashed) at some later time. Thus, it is <u>not</u> used for storage or transportation of data. But is perfect for concealing passwords.

Further, hashing (depending on the hashing algorithm) always produces a fixed-length value.

For example: **MD5** hashing <u>always</u> produces a 32-bit value no matter how big the data is that is hashed. While encryption produces a cipher that is directly proportionate in size to the data being encrypted. This makes encrypted data easier to decode by cybercriminals than hashed data.

The “MD” stands for “Message Digest” while the number indicates the algorithm being used.

MD5 produces 128-bit hashes expressed as 32 hexadecimal characters.

However, the most commonly used algorithms today are SHA (Secure Hashing Algorithm). Specifically: SHA-1 and SHA-256.

SHA-1 produces a 160-bit hashes comprised of 40 hexadecimal characters.

While **SHA-256** produces 256-bit hashes comprised of 64 hexadecimal characters. There are many other methods of encryption and hashing. The choice depends on network security goals.

20. Security: Configure a Username and Login Password using Authentication / AAA new model

This configuration creates an **authentication** environment via the AAA new model for a more secure login, as it requires both a specific username and corresponding password. **Note:** the term “local” means that the login information is stored on the device itself, not on a server.

```
Router1>en
Router1#conf t
Router1(config)#username ADMIN secret cisco123
Router1(config)#aaa new-model
Router1(config)#aaa authentication login default local
Router1(config)#line con 0
Router1(config-line)#login authentication default
Router1(config-line)exit
Router1(config)#exit
Router1#wr
```

21. Security: Configure a Telnet password for remote login between local devices

This configuration allows you to set a password for remote login to network or other host devices from inside or outside a network. However, Telnet transmits all data in plain text. So, it is not secure and should not be used to remote login from *outside* a network. Though it may be used to remote log into local devices from within the network. Note: the "transport" command defines which protocol will be used. Transport input/output can be "none", "all", "telnet", or "SSH".

```
Router1#conf t
Router1(config)#line vty 0 4
Router1(config-line)#password ADMIN
Router1(config-line)#transport telnet
Router1(config-line)#login
Router1(config-line)#exit
Router1(config)# exit
Router1#
```

Test by using the command **telnet** followed by the destination IP address of the device you want to log into.

22. Security: SSH for remote login using AAA and custom crypto keys

Because SSH is a secure protocol using AAA (Authentication, Authorization, and Accounting) it requires several linked configurations. **Note:** If level-15 privileges are to be accessed via SSH, an Enable password will also be needed (configure an Enable password per **#14** instructions).

```
Router1>en
Router1#conf t
Router1(config)#ip domain-name MYDOMAIN.COM
Router1(config)#crypto key generate rsa
The name for the keys will be Router1.MYDOMAIN.COM
Choose the size of the key modulus in the range of 360 to 2048 for your
   General Purpose Keys. Choosing a key modulus greater than 512 may take
   a few minutes.
How many bits in the modulus [512]: 1024
% Generating 1024 bit RSA keys, keys will be non-exportable…[OK]
Router1(config)#username R-ADMIN privilege 15 secret cisco123
```

```
Router1(config)#aaa new-model
Router1(config)#line vty 0 4
Router1(config-line)#transport input ssh
Router1(config-line)#exit
Router1(config)#exit
Router1#wr
```

SSH correct configuration may now be tested by attempting to remote-log in into the network by entering the command string shown (below) on a laptop using free wi-fi at a café.

Note: the **lowercase "-L"** indicates a login with the username followed by the network edge router's IP address.

```
C:\>
C:\>ssh -l Remote-Admin 200.110.55.1

Password:
Router1>en
Password:
Router1#
```

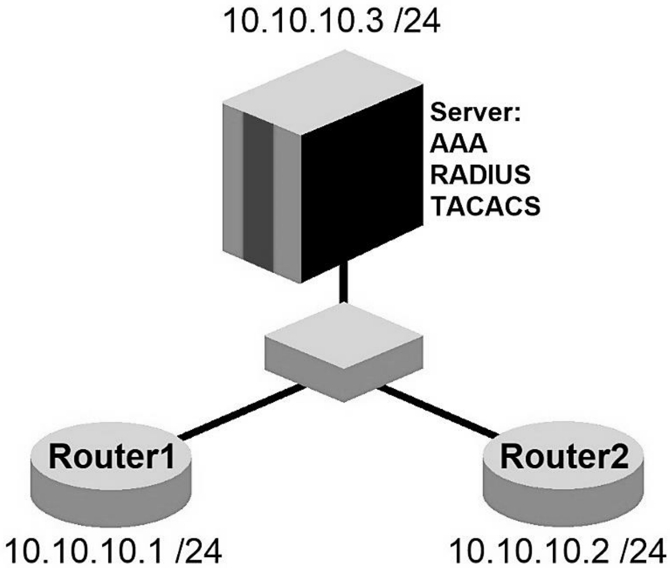
Once logged in and with Enable level-15 privileges, all administrative activities can be remotely and securely executed via SSH. We can further verify the connection by logging into the router locally and using the **Show SSH** command to see if there are any live SSH sessions in progress.

```
User Access Verification

Username: Admin1
Password:
Router1>en
Password:
Router1#sh ssh
Connection      Version Mode Encryption     Hmac          State            Username
3               1.99    IN   aes128-cbc   hmac-sha1   Session Started    Remote-Admin
3               1.99    OUT  aes128-cbc   hmac-sha1   Session Started    Remote-Admin
%No SSHv1 server connections running.
Router1#
Router1#sh ssh
%No SSHv2 server connections running.
%No SSHv1 server connections running.
Router1#
```

After logging off (on the laptop) when we run the Show SSH command again, we see the SSH session has ended and no other SSH sessions are in progress. **Note:** SSHv2 is used by default.

23. AAA, TACACS, and RADIUS

AAA stands for Authentication, Authorization, and Accounting. **Terminal Access Controller Access-Control System (TACACS)** is actually a group of security protocols for remote authentication via a central server. However, a more modern protocol is **RADIUS**. The **Remote Authentication Dial-In User Service** that provides **AAA** management for users whenever they connect to and log into a network.



Example configuration:

Router1 to use **TACACS**

Router2 to use **RADIUS**

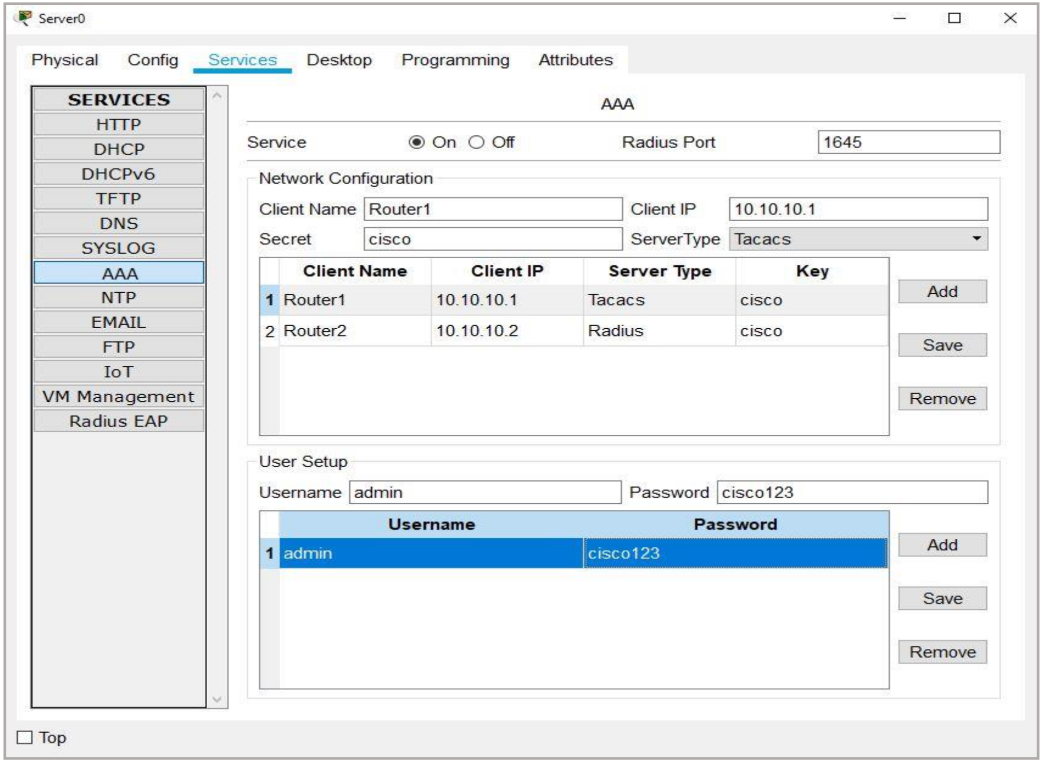
User Setup:

Username **admin**

Password **cisco123**

See virtual UI next page. Many servers use a UI of some kind. Menus and entry-field options will be similar.

Client Name is the device hostname; **Client IP** is the device IP address; **Secret** is the **Key**. The key is <u>not</u> a password. It is an authentication key pointing to the device **Username** and **Password**.



Router1 configured to use **TACACS**. Followed by **Show Running-Config** to verify.

```
Router1#conf t
Router1(config)#aaa new-model
Router1(config)username admin secret cisco123
Router1(config)#aaa authentication login default group tacacs+ local
Router1(config)#aaa authentication enable default group tacacs+ local
Router1(config)#tacacs host 10.10.10.3 key cisco
Router1(config)#exit
```

“secret” uses MD5 hash

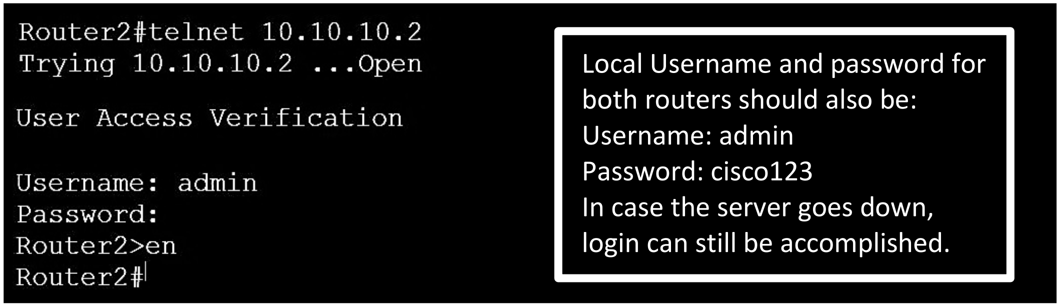
The “**local**“ at the end means a matching local username and password can be used if the AAA server is down. Though of course, you must also configure a local username and password.

```
aaa new-model
!
aaa authentication login default group tacacs+ local
aaa authentication enable default group tacacs+ local
```

Router2 configured to use **RADIUS**. Followed by Show Running-Config to verify.

```
Router2#conf t
Router2(config)#AAA new-model
Router1(config)username admin secret cisco123
Router2(config)#aaa authentication login default group radius local
Router2(config)#aaa authentication enable default group radius local
Router2(config)#radius-server host 10.10.10.3 key cisco
Router2(config)#exit
```

The configuration can be tested by **telnetting** to itself. Username: **admin** Password: **cisco123**



24. Banner: Display Message of the Day (MotD)

This is a generic message displayed at the network login screen. **Note:** Remember also to use a delimiting character not one used in the body of the message itself to mark the beginning and end of the message. You can verify your configuration by exiting all the way out, and then logging in again.

```
Router1>en
Router1#conf t
Router1(config)#banner motd ?
  LINE c banner-text c, where 'c' is a delimiting character
Router1(config)#banner login $Welcome to ABC Company. Our network will
be down for maintenance at 4pm today.$
Router1(config)#exit
Router1#wr
```

25. Banner: Display Login Message

The Login banner message is configured in much the same way as the MotD. Though it is considered a network best practice that the login message include a legal disclaimer. **Note:** when using ‘x’ as the delimiting character, additional lines and ASCII art can be added to put space between MotD and Login banners.

```
Router1>en
Router1#conf t
Router1(config)#banner login x


**************************
  Authorized Personnel Only
**************************x
Router1(config)#exit
Router1#wr
```

Tap the Enter key after the first delimiting ‘x’ to add blank line spaces and/or ASCII art

Remember to add the final delimiting character to finish the configuration

Log out and then log back in to verify both banner messages are displaying properly:

```
Welcome to ABC Company. Our network will be down for maintenance at 4pm today.




*******************************
 Authorized Personnel Only
*******************************


User Access Verification

Username:
```

26. Time and date: Set local device time, day, month, and year / Parts 1 and 2

Part 1 of 2: Look up the UTC (Universal Time Coordinate) and set the current time and date shown by UTC. UTC is the successor to GMT and uses an atomic clock to keep perfect time. It also uses a military-time format of 0-hundred thru 23-hundred hours.

```
Switch>en
Switch#clock set ?
  hh:mm:ss  Current Time
Switch#clock set 13:15:02 ?
  <1-31>   Day of the month
  MONTH    Month of the year
Switch#clock set 13:15:02 1 ?
  MONTH  Month of the year
Switch#clock set 13:15:02 1 JANUARY ?
  <1993-2035>  Year
Switch#clock set 13:15:02 1 JANUARY 2021 ?
  <cr>
Switch#clock set 13:15:02 1 JANUARY 2021
Switch#wr
Building configuration...
[OK]
Switch#
```

Part 2 of 2: Adjust UTC to match your local area date/time. This is done using UTC (+) or (-) the number of hours and minutes to match your local time.

```
Switch#conf t
Switch(config)#clock timezone Pacific ?
  <-23 - 23>  Hours offset from UTC
Switch(config)#clock timezone Pacific -7
Switch(config)#clock timezone Pacific -7 ?
  <0-59>  Minutes offset from UTC
  <cr>
Switch(config)#clock timezone Pacific -7 0
Switch(config)#exit
Switch#
```

Go online to find the UTC and then add or subtract hours to match your local time.

In this example the ISP is located on the west coast of North America. This is in the Pacific Standard Time zone.

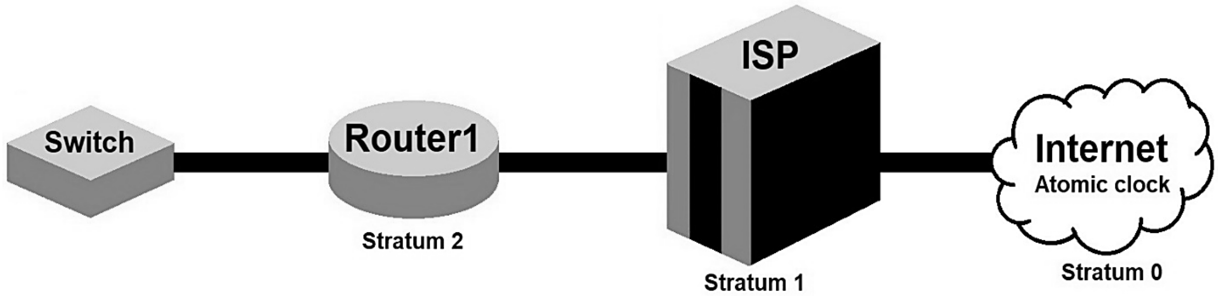
Pacific Standard Time is minus 7 hours and zero minutes behind UTC.

Finish by repeating the “show clock” command to verify that the clock is running and keeping time correctly. Notice the fourth value is displayed in milliseconds.

```
Switch#sh clock
13:15:45.915 UTC Fri Jan 1 2021
Switch#sh clock
13:15:52.10 UTC Fri Jan 1 2021
Switch#sh clock
13:15:55.932 UTC Fri Jan 1 2021
Switch#
```

27. NTP (Network Time Protocol)

The **Network Time Protocol** is used to synchronize all network device clocks. This is important for many reasons besides time. For example: date/time-stamps for sys-log reports, debugs, troubleshooting, and updates. Also, security via time-managed login, DHCP lease times, CAM table resets, and digital certificates which often have a limited time use. For these many important reasons, in the real world of network engineering and administration we do not configure device clocks separately, and we do not configure our devices as authoritative time sources to synchronize other device clocks. Instead, we *source* our time from our ISP or directly from an atomic clock server.



28. NTP (Network Time Protocol) Stratum

The best way for a network to get accurate time is from an atomic clock via the network's ISP. The accuracy or trustworthiness of a time source is measured in **Stratum.** The lower the number the better. Atomic clocks have a Stratum of zero (the most trustworthy). ISPs that get time from atomic clocks have a Stratum of 1, and a network device that gets its time from their ISP has a Stratum of 2, and so on. Stratum ranges from 0 thru 15.

In this example a simulated ISP is configured as the authoritative time source for the network. First, look up the UTC (Universal Time Coordinate) online and set the ISP clock to that time and date.

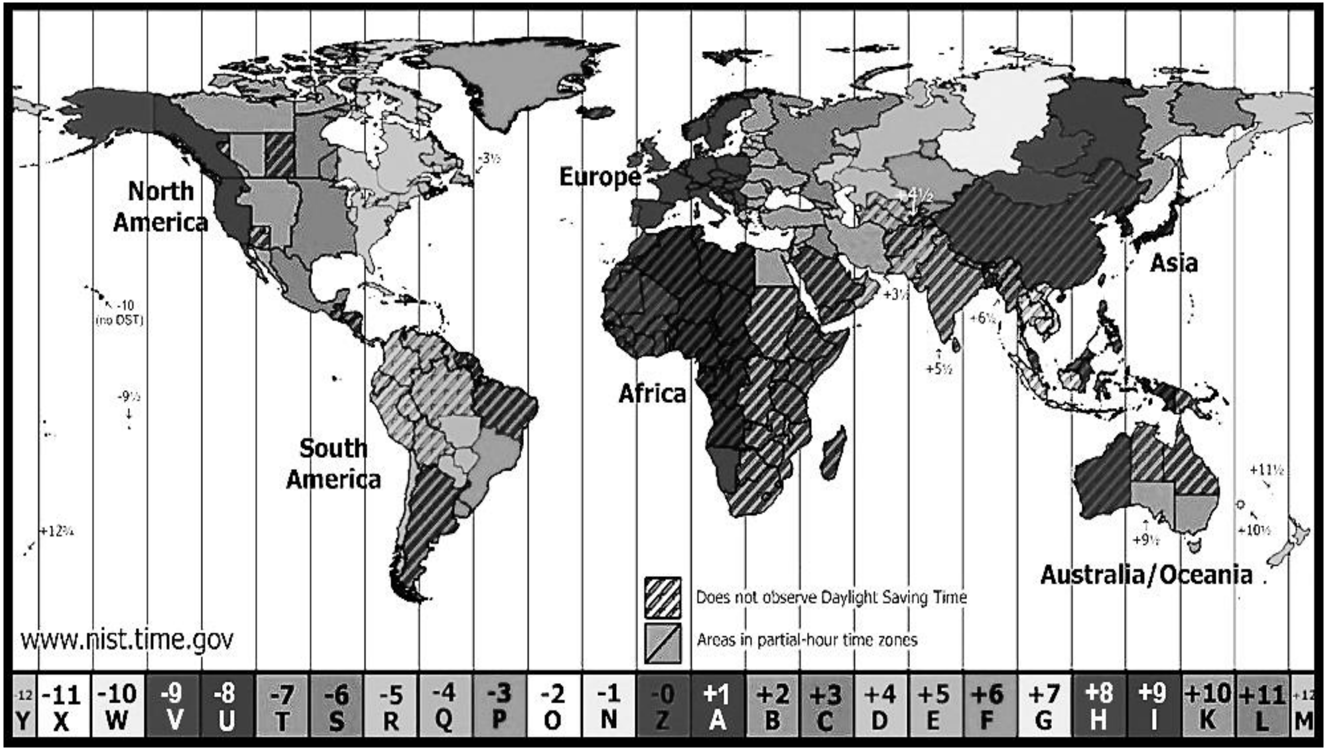
```
ISP>en
ISP#clock set 13:15:02 1 January 2021
ISP#wr
```

Now, adjust the settings to match the ISP's <u>local</u> time. Here it is 8 hours and zero minutes behind UTC.

```
ISP#conf t
ISP(config)#clock timezone Pacific -8 0
ISP(config)#exit
ISP#wr
```

Repeating the Show Clock command verifies correct configuration.

```
ISP>en
ISP#sh clock
6:45:1.581 Pacific Fri Jan 1 2021
ISP#sh clock
6:45:4.582 Pacific Fri Jan 1 2021
ISP#sh clock
6:45:5.761 Pacific Fri Jan 1 2021
ISP#
```


North America
South America
Europe
Africa
Asia
Australia/Oceania
www.nist.time.gov
Does not observe Daylight Saving Time
Areas in partial-hour time zones
-12 Y
-11 X
-10 W
-9 V
-8 U
-7 T
-6 S
-5 R
-4 Q
-3 P
-2 O
-1 N
-0 Z
+1 A
+2 B
+3 C
+4 D
+5 E
+6 F
+7 G
+8 H
+9 I
+10 K
+11 L
+12 M

Next, set the ISP stratum to **1** to indicate that it gets its time from an atomic source (Stratum 0).

```
ISP#conf t
ISP(config)#ntp ?
  authenticate         Authenticate time sources
  authentication-key   Authentication key for trusted time sources
  master               Act as NTP master clock
  server               Configure NTP server
  trusted-key          Key numbers for trusted time sources
  update-calendar      Configure NTP to update the calendar.
ISP(config)#ntp master ?
  <1-15>  Act as NTP master clock
  <cr>
ISP(config)#ntp master 1 ?
  <cr>
ISP(config)#ntp master 1
ISP(config)#exit
```

Now, configure Router1 as the **NTP server** with the source IP address of its authoritative time provider (the ISP). **Note:** the NTP server's timezone name and UTC adjustment does not have to match the ISP.

```
Router1#
Router1#conf t
Router1(config)#ntp server 10.10.10.1
Router1(config)#clock timezone Pacific -8 0
Router1(config)#exit
```

Verify by running the Show Clock command on the ISP and Router1.

```
Router1#sh clock
8:13:23.729 Pacific Fri Jan 1 2021
Router1#
```

```
ISP#sh clock
8:13:20.476 Pacific Fri Jan 1 2021
ISP#
```

Optional: this same configuration can be applied to the switch with Router1's internal interface as Switch1's source IP address.

29. NTP (Network Time Protocol): Show NTP Associations

```
Router1#sh ntp associations

address         ref clock       st   when     poll    reach  delay    offset       disp
*~10.10.10.1    127.127.1.1     1    5        16      377    4.00     0.00         0.12
 * sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
```

30. NTP (Network Time Protocol): Show NTP Status

Notice Router1's stratum is automatically set to stratum 2 because it is one step from the simulated ISP.

```
Router1#sh ntp status
Clock is synchronized, stratum 2, reference is 10.10.10.1
nominal freq is 250.0000 Hz, actual freq is 249.9990 Hz, precision is 2**24
reference time is E37236DB.000003D7 (15:52:59.983 UTC Fri Jan 1 2021)
clock offset is 0.00 msec, root delay is 5.00  msec
root dispersion is 10.65 msec, peer dispersion is 0.12 msec.
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is - 0.000001193
s/s system poll interval is 4, last update was 16 sec ago.
Router1#
```

Note: A network can have more than one NTP server. Simply configure as previously shown and the downstream device will automatically be given a higher value stratum making it the back-up NTP server.

31. Assign a device to an IP Domain

Statically assigns a device to a specific domain. Verifiable by using either the **show run** or **show hosts** commands. **Note:** a domain is necessary for AAA and other security related configurations.

```
Router1>en
Router1#conf t
Router1(config)#ip domain name MYDOMAIN.com
Router1(config)#exit
Router1#
```

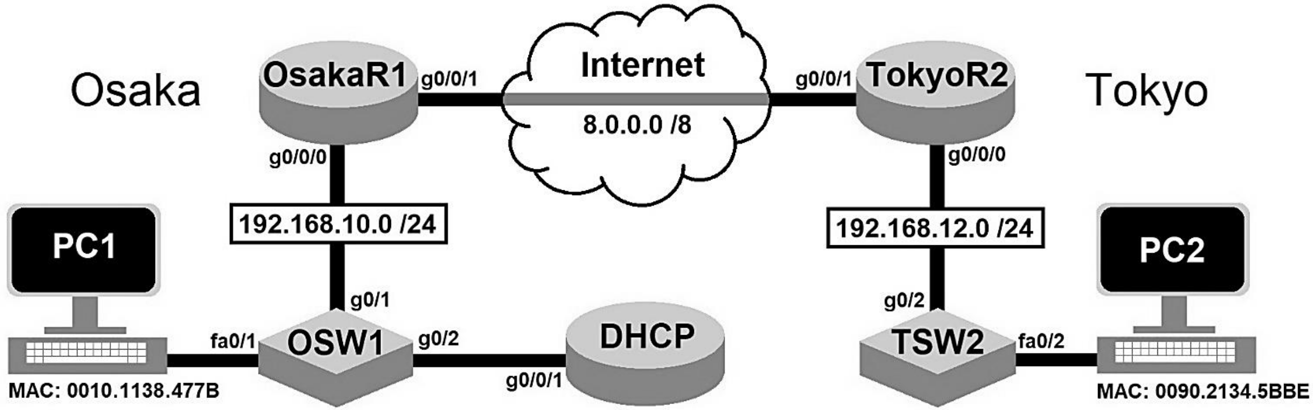
32. Configure a Loopback address on a device

A **loopback** address functions as a virtual interface IP address. Loopback addresses also serve as device IDs and play a role in device elections in various protocols. Devices can have more than one loopback address, and they can number from 0 thru 2147483647.

```
Router1>en
Router1#conf t
Router1(config)#ip loopback 1
Router1(config-if)# ip address 1.1.1.1 255.255.255.0
Router1(config-if)#exit
Router1(config)#exit
Router1#
```

33. DHCP: Configure a DHCP server on a router or switch

A **Dynamic Host Control Protocol** server can be configured on a traditional server form factor or configured on most network devices (routers and switches).



A DHCP server can provide IP addresses for multiple network's devices. The **default-router** is the default gateway for the LAN the DHCP server is on. The **DNS server** address is that of the preferred Domain Name Services provider.

```
Router1#conf t
Router1(config)#ip dhcp excluded-address 192.168.10.1 192.168.10.9
Router1(dhcp-config)#ip dhcp pool OSAKA
Router1(dhcp-config)#network 192.168.10.0 255.255.255.0
Router1(dhcp-config)#default-router 192.168.10.1
Router1(dhcp-config)#dns-server 8.8.8.8
Router1(dhcp-config)#exit
Router1(config)#exit
Router1#
```

> **IMPORTANT:** When configuring a single DHCP server for two or more LANs, a separate Exclusion list, Pool, Network, and Default Router, will be also needed. Along with a DHCP Relay Agent.

34. DHCP: Configure a DHCP Relay using the IP-helper address command

DHCP discover messages are not routable because they are Broadcasts. So, if we have only one DHCP server for a network made of two or more connected LANs, or a WAN, we need a way to route those discovery messages to the other network where the DHCP server is located. To do this we create a **DHCP Relay Agent** by using the IP helper-address configuration. This instructs the gateway router on the remote network to route discovery messages across the WAN to the network where the DHCP server is located.

```
Router2#conf t
Router2(config)#int g0/0/0
Router2(config-if)#ip helper-address 192.168.10.3
Router2(config-if)#exit
Router2(config)#exit
Router2#
```

The address we use is the IP address of the DHCP server on the <u>other</u> LAN. We configure it as shown on the default-gateway of the LAN that *needs* its discover messages relayed to the DHCP server.

35. DHCP: Show IP DHCP Binding

This command shows which IP addresses have been leased to which device IDs according to their MAC address.

```
DHCP#sh ip dhcp bi
IP address       Client-ID/              Lease expiration        Type
                 Hardware address
192.168.10.11    0010.1138.477B           --                     Automatic
192.168.12.11    0090.2134.5BBE           --                     Automatic
DHCP#
```

By default, DHCP servers lease IP addresses for 24-hours. However, if a user is logged in for a continuous 10 hours, for security, at that time, the lease will automatically end, and the user will be logged off. The device will then begin DORA to acquire a new IP address.

36. DHCP: Show IP DHCP Pool

```
DHCP#sh ip dhcp pool

Pool OSAKA :
 Utilization mark (high/low)    : 100 / 0
 Subnet size (first/next)       : 0 / 0
 Total addresses                : 254
 Leased addresses               : 1
 Excluded addresses             : 2
 Pending event                  : none

 1 subnet is currently in the pool
 Current index        IP address range                    Leased/Excluded/Total
 192.168.10.1         192.168.10.1     - 192.168.10.254    1     / 2      / 254

Pool TOKYO :
 Utilization mark (high/low)    : 100 / 0
 Subnet size (first/next)       : 0 / 0
 Total addresses                : 254
 Leased addresses               : 1
 Excluded addresses             : 2
 Pending event                  : none

 1 subnet is currently in the pool
 Current index        IP address range                    Leased/Excluded/Total
 192.168.12.1         192.168.12.1     - 192.168.12.254    1     / 2      / 254
DHCP#
```

37. DHCP: Security / Snooping and Trusted Ports

A security measure of particular use on switches related to trunk ports and DHCP, this command instructs the device to trust only the designated port(s). However, this configuration should be followed by disabling option 82 to prevent all other ports from being shut down.

```
Switch1#conf t
Switch1(config)#int vlan X
Switch1(config-if)#ip dhcp snooping
Switch1(config-if)#exit
Switch1(config)#int X
Switch1(config-if)#no ip dhcp snooping information option
Switch1(config-if)#exit
Switch1(config)#exit
Switch1#
```

Where "X" is the VLAN to be snooped.

Where "X" is the interface to be trusted.

The "no" command for info option disables option 82. Option 82 disables all ports *without* a trusted status whenever the trusted port command is used. On switches it is usually disabled because otherwise all ports would need a separate trusted port configuration.

38. Configure an IP default-gateway address on a device

Configuring an IP default-gateway on a device tells that device where the default entrance and exit is on its network and directs it to forward data to that address for routing outside of its network.

Note: depending on the configurations being used, this configuration is optional for layer-2 switches since they do not perform routing. But it is necessary for host devices like layer-3 core switches, and PCs and IoT. It also bears mentioning that a default gateway and a gateway of last resort are not the same thing.

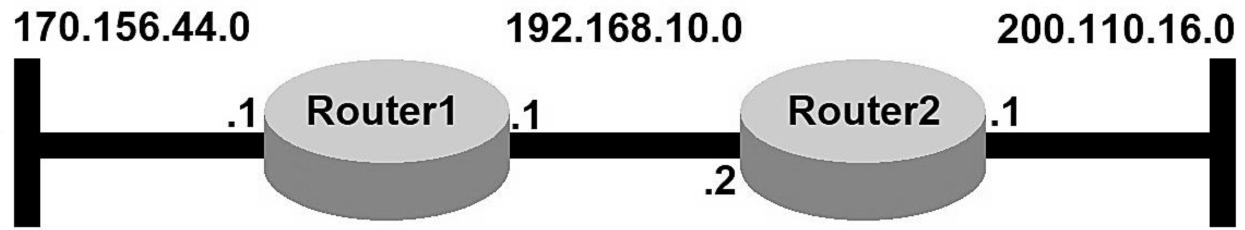
```
Switch1>en
Switch1#conf t
Switch1(config)#ip default-gateway 170.156.44.1
Switch1(config)#exit
Switch1#
```

Devices on a network will only use one default-gateway. See the network diagram on next page.

39. Configure a static route to a specific network

A router needs directions to reach a specific network that it is not directly connected to.

```
Router1>en
Router1#conf t
Router1(config)#ip route 200.110.16.0 255.255.255.0 192.168.10.2
Router1(config)#exit
Router1#
```



Devices on network **170.156.44.0** may now communicate with devices on network 200.110.16.0 because now Router1 knows the route to that network goes **via** Router2’s port (.2) on the 192.168.10.0 network.

Verify any route configuration by using the command **Show IP Route** and consult the codes to identify routing protocols. Below the “S” indicates that our static route is correctly configured and entered.

```
Router1#sh ip ro
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     170.156.0.0/16 is variably subnetted, 5 subnets, 3 masks
C       170.156.44.0/25 is directly connected, GigabitEthernet0/0/0
L       170.156.44.1/32 is directly connected, GigabitEthernet0/0/0
     192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0/1
L       192.168.10.1/32 is directly connected, GigabitEthernet0/0/1
S    200.116.10.0/24 [1/0] via 192.168.10.2

Router1#
```

40. Configure a gateway of last resort on a router

A **gateway of last resort** is a static route that functions as a wildcard route that directs a router to route packets to the nearest available router to finish the routing process. In this context, the zeros indicate "any network". Though of course, hopefully that other router knows a route to the destination network.

```
Router1>en
Router1#conf t
Router1(config)#ip route 0.0.0.0. 0.0.0.0 192.168.10.2
Router1(config)#exit
Router1#
```

In this example the previous topology is used with the addition of two new networks: 192.168.13.0 /24 and 192.168.14.0 /24. Router2 is also configured with a static route to the 192.168.14.0 /24 network.

Note: a gateway of last resort is generally not used for routing between LANs because it is inefficient. Instead, proper routing protocols (such as RIPv2, OSPF, EIGRP, etc.) are used.

Again, the **Show IP Route** lets us verify our IP route configuration. Note: the Asterix after the "S" indicates this static route is also the gateway of last resort.

```
Router1#sh ip ro
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 192.168.10.2 to network 0.0.0.0

     170.156.0.0/16 is variably subnetted, 5 subnets, 3 masks
C       170.156.44.0/25 is directly connected, GigabitEthernet0/0/0
L       170.156.44.1/32 is directly connected, GigabitEthernet0/0/0
     192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0/1
L       192.168.10.1/32 is directly connected, GigabitEthernet0/0/1
S    200.116.10.0/24 [1/0] via 192.168.10.2
S*   0.0.0.0/0 [1/0] via 192.168.10.2

Router1#
```

41. VLANs: Configure a VLAN on a switch

```
Switch1>en
Switch1#conf t
Switch1(config)#vlan 22
Switch1(config)#exit
```

42. VLANs: Configure a trunk port on a switch using the Switchport command for ROS

A trunk port is needed between a switch and a router to use ROS, the Router-On-a-Stick topology.

```
Switch1>en
Switch1#conf t
Switch1(config)#int g0/0/1
Switch1(config-if)#switchport mode trunk
Switch1(config-if)#switchport trunk allowed vlan 22
Switch1(config-if)#exit
Switch1#(config)#exit
```

More VLANs can be allowed simply by adding their ID numbers separated by commas.

43. VLANs: Configure a sub-interface on a router to use RoaS

A trunk port is needed between a switch and a router to use the **Router-On-a-Stick (RoaS)** topology. Trunk ports use logical sub-interfaces on the router end which makes it possible for inter-VLAN routing on a single cable link. **Note:** Encapsulation* and VLAN numbers must match. But the sub-interface and VLAN numbers do not need to match, though we often number them this way to make it easier to remember which sub-interface is assigned to which VLAN. Also, the IP address of the virtual network (the VLAN) can be <u>any</u> class or sub-class. Example: VLAN 22 could have been on network 10.10.10.0 255.0.0.0 if we had wanted. Because it is the ROaS topology that makes inter-VLAN routing possible.

```
Router1#conf t
Router1(config)#int g0/0/1.22
Router1(config-subif)#encapsulation dot1q 22
Router1(config-subif)#ip address 192.168.22.1 255.255.255.0
Router1(config-subif)#exit
Router1(config)#exit
Router1#
```

*Encapsulation is how the data is tagged (by number) according to which VLAN it belongs to.

44. VLANs: Configure an access port on a switch using the Switchport command

An **Access** port is one that will be used by a device. Technically* an access port can only be assigned to one VLAN. If we have many devices connected to the switch and they are all in the same VLAN, we can use the "int range" command. In the example below, Fast Ethernet port /1 is configured using the Switchport command. This time, choosing the "access" mode.

```
Switch1#conf t
Switch1(config)#int f0/1
Switch1(config-if-range)#switchport mode access
Switch1(config-if-range)#switchport access vlan 22
Switch1(config-if-range)#exit
Switch1(config)#exit
Switch1#
```

*The only exception is when an access port is shared by two VLANS, one for data and one for VoIP.

45. VLANs: Configure switches to use VTP

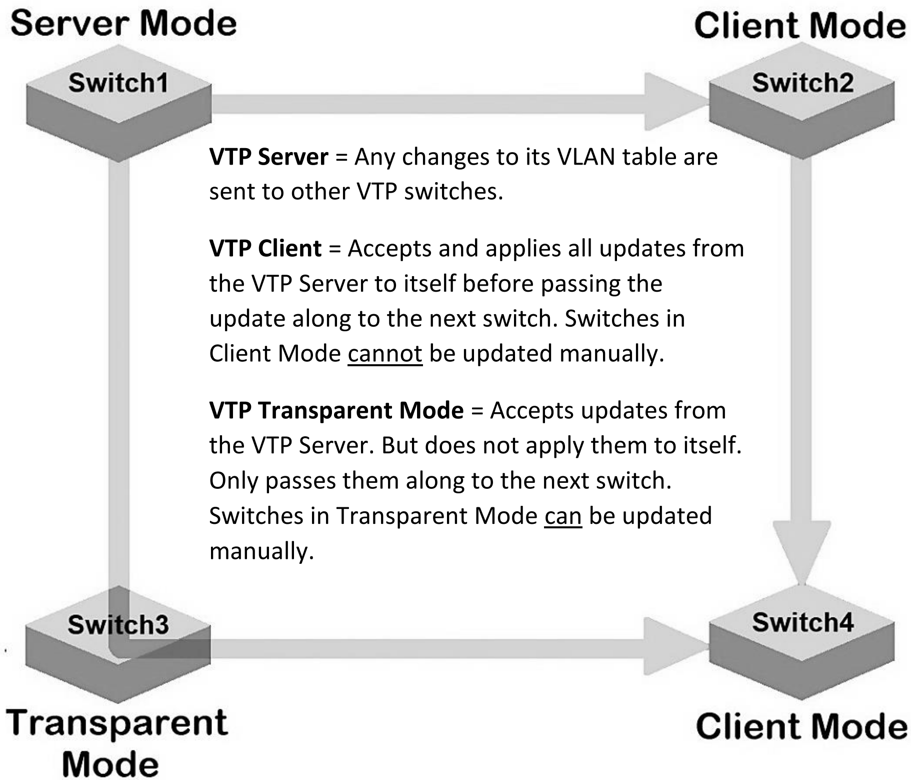
VLAN Trunking Protocol is not actually a trunking protocol. It is more accurately defined as a VLAN database synchronizing protocol, in that one switch acts as a server that automatically propagates any such changes by sending those updates to all the other switches participating in the same VTP environment. Configuring switches to use VTP is a simple process:

Step 1. All participating switches must also be configured with Trunk ports between them.

Step 2: Assign a VTP domain name. This is the VTP domain that all participating switches will belong to. Optionally, you may also configure an administrative password and a version (v1, v2 or v3).

Step 2: Choose a VTP mode. By default, all switches are in server mode. As such this has no effect on other switches until their roles are changed to either client or transparent mode. Note: VTPv3 also includes an “Off” mode.

VTP Modes



VTP Server = Any changes to its VLAN table are sent to other VTP switches.

VTP Client = Accepts and applies all updates from the VTP Server to itself before passing the update along to the next switch. Switches in Client Mode <u>cannot</u> be updated manually.

VTP Transparent Mode = Accepts updates from the VTP Server. But does not apply them to itself. Only passes them along to the next switch. Switches in Transparent Mode <u>can</u> be updated manually.

Here, we configure Switch2 of 4 imagining a topology similar to the diagram on the previous page.

```
Switch2>en
Switch2#conf t
Switch2(config)#vtp ?
  domain    Set the name of the VTP administrative domain.
  mode      Configure VTP device mode
  password  Set the password for the VTP administrative domain
  version   Set the adminstrative domain to VTP version
Switch2(config)#vtp domain MY-VTP-DOMAIN
Changing VTP domain name from NULL to MY-VTP-DOMAIN
Switch2(config)#vtp password VTP-ADMIN
Setting device VLAN database password to VTP-ADMIN
Switch2(config)#vtp version 2
Switch2(config)#vtp mode ?
  client       Set the device to client mode.
  server       Set the device to server mode.
  transparent  Set the device to transparent mode.
Switch2(config)#vtp mode client
Setting device to VTP CLIENT mode.
Switch2(config)#exit
Switch2#
```

VTP Client Switch2's VLAN table <u>before</u> creating VLANs on VTP Server Switch1.

```
Switch2>en
Switch2#sh vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                                Fa0/21, Fa0/22, Fa0/23, Fa0/24
                                                Gig0/2
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
Switch2#
```

VTP Client Switch2's VLAN table <u>after</u> configuring the VTP environment. Now, VLANs created on VTP Server Switch1 will automatically propagated on VTP Client Switch2's (and Switch4's) VLAN tables.

```
Switch2#sh vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                                Fa0/21, Fa0/22, Fa0/23, Fa0/24
                                                Gig0/2
11   VLAN0011                         active
22   VLAN0022                         active
33   VLAN0033                         active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
Switch2#
```

Features	VTPv1	VTPv2	VTPv3
Modes	1. Server 2. Client 3. Transparent	1. Server 2. Client 3. Transparent	1. Server 2. Client 3. Transparent 4. Off
Token Ring	Not supported	Supported	Supported
Private VLANs	Not supported	Not supported	Supported
VLAN range	1 thru 1000	1 thru 1000	1 thu 4094
Authentication	Plain text	Plain text	Encryption

46. VLANs: Configuring a Native VLAN

The term "Native VLAN" is Cisco proprietary, though other network device manufacturers also use that term, or simply call it an untagged VLAN. Because that's all it is. Native VLANs transmit untagged traffic across our trunk ports. So, for example: if PC1 is not a member of any VLAN and wants to send an email to PC2 (which is a member of a VLAN) the Native VLAN makes that possible because it will transmit PC1's email to PC2. Native VLANs are also useful for transmitting data for legacy devices that do not use VLAN tagging. **Note:** since everyone knows the Default VLAN is numbered 1, as a security best practice we should never number our Native VLAN as 1.

```
Switch1>en
Switch1#conf t
Switch1(config)#vlan 44
Switch1(config-vlan)#exit
Switch1(config)#int g0/2
Switch1(config-if)#switchport mode trunk
Switch1(config-if)#switchport trunk native vlan 44
Switch1(config-if)#exit
Switch1(config)#exit
Switch1#
```

47. VLANs: Moving unused ports out of Default VLAN 1 for added security

The Default VLAN is always numbered "1" and since we cannot shut or delete VLAN 1, it is important for security purposes to never number our Native VLAN as 1. That said, after shutting the unused ports, we can improve network security by moving all those unused ports out of the Default VLAN and into the Native VLAN or an unused VLAN with another number of our own choosing (again, by using the int range command). Example: We are using the first three Fast Ethernet ports on Switch1. Here is how to move all remaining ports out of the Default VLAN 1 and into our Native VLAN 44.

```
Switch1#conf t
Switch1(config)#int range f0/4-24
Switch1(config-if)#switchport mode access
Switch1(config-if)#switchport access vlan 44
Switch1(config-if)#exit
Switch1(config)#exit
Switch1#
```

48. VLANs: Show VLAN brief to verify which switch ports are assigned to which VLANs

Following a previous command example, we can see that all unused ports were successfully moved from Default VLAN 1 to Native VLAN 44.

```
Switch1#sh vlan br

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active
11   SALES                            active    Fa0/1
22   ACCTG                            active    Fa0/2
33   WAREHOUSE                        active    Fa0/3
44   NATIVE                           active    Fa0/4, Fa0/5, Fa0/6, Fa0/7
                                                Fa0/8, Fa0/9, Fa0/10, Fa0/11
                                                Fa0/12, Fa0/13, Fa0/14, Fa0/15
                                                Fa0/16, Fa0/17, Fa0/18, Fa0/19
                                                Fa0/20, Fa0/21, Fa0/22, Fa0/23
                                                Fa0/24
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
Switch1#
```

49. VLANs: Show Mac-Address-Table

This MAC address table shows all connected devices to the switch. **Note:** The ports are the same because these devices are in VLANs using g0/1 as their **trunk** port. To see the **access ports** of each connected device, use the **Show VLAN ID** command on the next page.

```
Switch1>en
Switch1#sh mac-address-table
          Mac Address Table
-------------------------------------------


Vlan    Mac Address       Type        Ports
----    -----------       --------    -----

   1    0002.4ad4.6001    DYNAMIC     Gig0/1
  11    0040.0b25.1ea3    DYNAMIC     Gig0/1
  22    00d0.bc40.d8e8    DYNAMIC     Gig0/1
  33    0006.2a0d.61c8    DYNAMIC     Gig0/1
Switch1#
```

50. VLANs: Show VLAN ID / Show VLAN name

This command shows which VLAN (by ID number and/or name) is using which access port. **Note:** you may also use the command "sh vlan" (followed by the name instead of the ID number) to view a similar report.

```
Switch1>en
Switch1#sh vlan ID 22

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
22   ACCTG                            active    Fa0/2

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
22   enet  100022     1500  -      -      -        -    -        0      0

Switch1#
```

51. VLANs: Show Interface Trunk and **Show Interface X Status**

These two commands are of particular use on multi-switched VLANs.

```
Switch1#sh int trunk
Port        Mode         Encapsulation  Status        Native vlan
Gig0/2      on           802.1q         trunking      1

Port        Vlans allowed on trunk
Gig0/2      1,22

Port        Vlans allowed and active in management domain
Gig0/2      1,22

Port        Vlans in spanning tree forwarding state and not pruned
Gig0/2      1,22

Switch1#
```

```
Switch1#sh int g0/2 status
Port      Name               Status       Vlan       Duplex  Speed Type
Gig0/2                       connected    1          auto    auto  10/100BaseTX

Switch1#
```

52. VLANs: Show Interface X Switchport

To see the status of all device interfaces, simply omit the interface ID.

```
Switch1#sh int g0/2 switchport
Name: Gig0/2
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan: none
Trunking VLANs Enabled: 1,22
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL
Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none
```

53. Logging: Configure automated report logging for a sys-log server

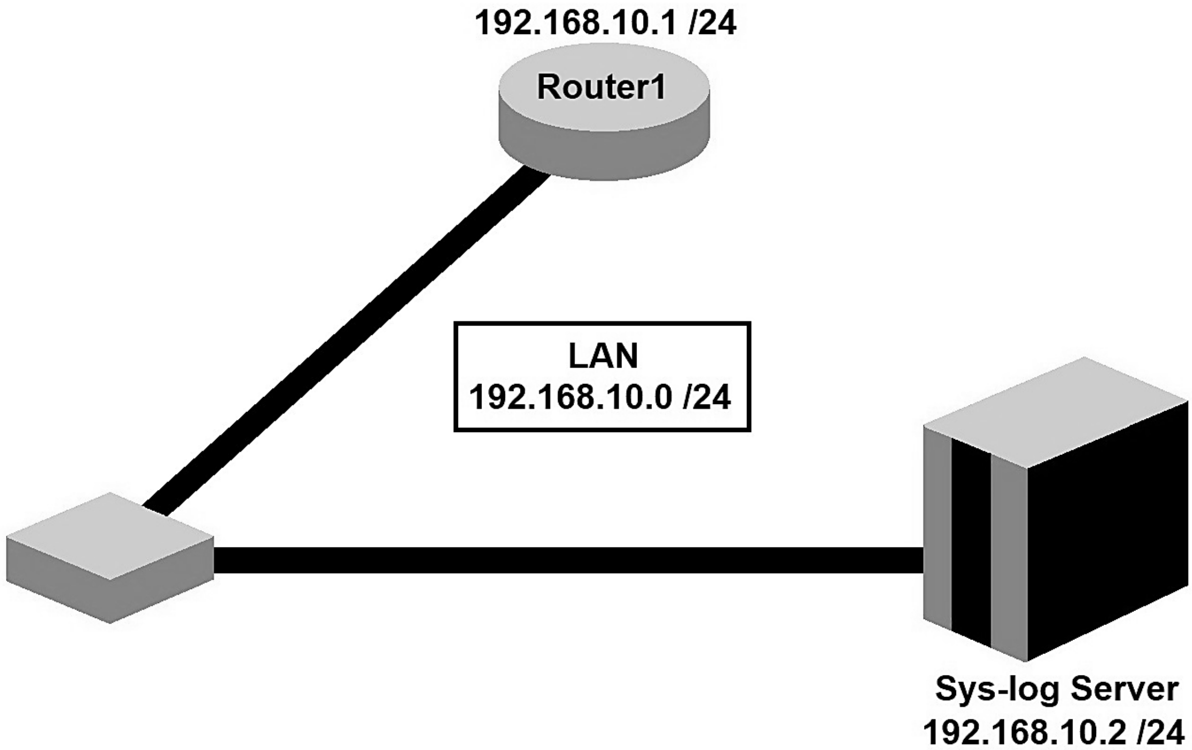
Automated logging detects any changes made to a device and reports them to a systems log server.

Note: a participating device must have a valid IP address to report to the Sys-log server.

For this example, we will use the simple topology shown on the next page.

```
Router1>en
Router1#conf t
Router1(config)#logging host 192.168.10.2
Router1(config)#logging on
Router1(config)#exit
Router1#
```

The sys-log server IP address


192.168.10.1 /24
Router1
LAN
192.168.10.0 /24
Sys-log Server
192.168.10.2 /24

54. Logging: Configure timestamps to be included with sys-log automated report logging

To add date and time for each log reported to the sys-log server, use this command.

```
Router1#conf t
Router1(config)#service timestamps log datetime msec
Router1(config)#exit
Router1#
```

55. Logging: Configure timestamps to be included with real time on-screen debug notifications

This command adds date and timestamps to on-screen change notifications.

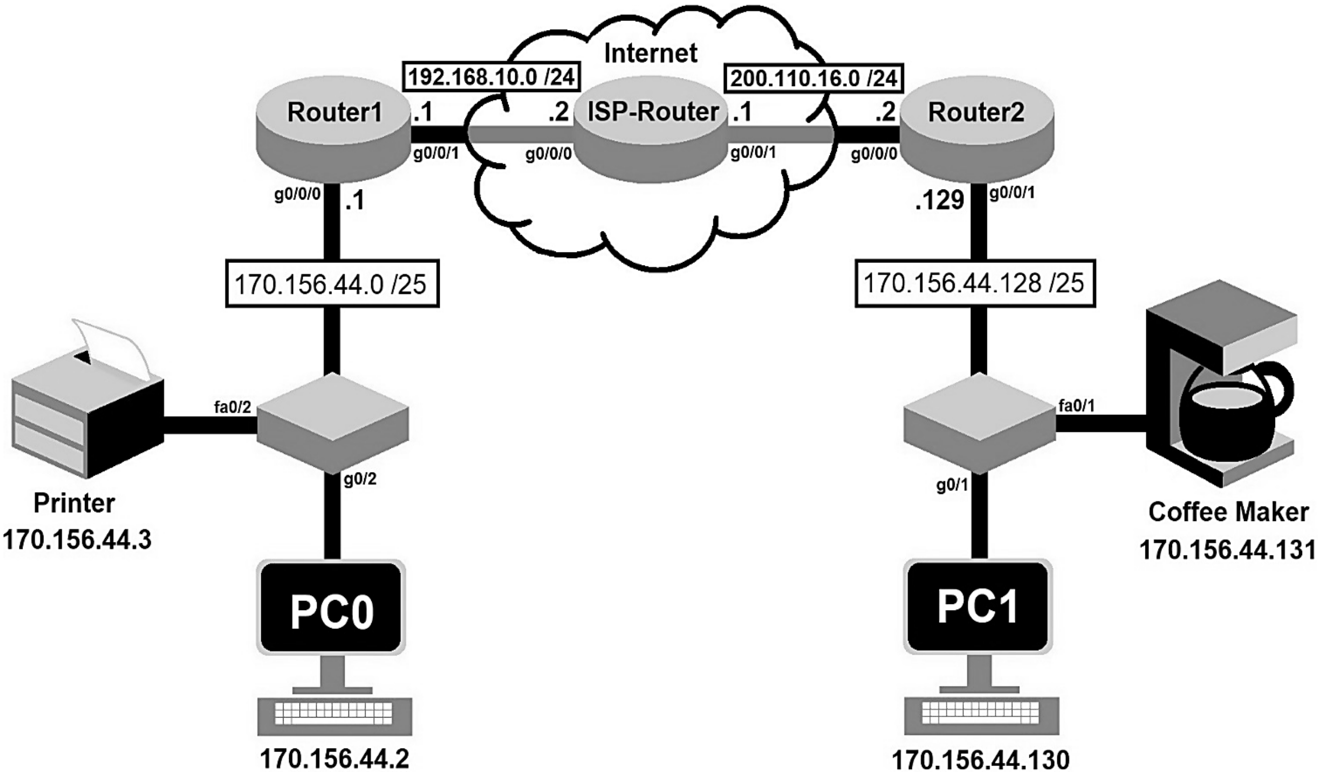
```
Router1#conf t
Router1(config)#service timestamps debug datetime msec
Router1(config)#exit
Router1#
```

56. Routing protocol: RIPv2

RIPv2 supports both classful and classless inter-domain routing (sub-networks). Here we will use Router1 as the example of how to configure each of the routers in our example topology.

```
Router1#conf t
Router1(config)#router rip
Router1(config-router)#version 2
Router1(config-router)#network 170.156.44.0 255.255.255.128
Router1(config-router)#network 192.168.10.0 255.255.255.0
Router1(config-router)#no auto-summary
Router1(config-router)#exit
Router1#(config#)exit
```

Note: the “no auto-summary” command is used to prevent RIPv2 from summarizing sub-network addresses back to their classful boundaries which can cause misrouted traffic and dropped packets. By default, auto-summary is disabled. But as a best practice is often manually disabled to be certain.


Internet
192.168.10.0 /24
200.110.16.0 /24
Router1
.1
.2
ISP-Router
.1
.2
Router2
g0/0/1
g0/0/0
g0/0/1
g0/0/0
g0/0/0
.1
.129
g0/0/1
170.156.44.0 /25
170.156.44.128 /25
fa0/2
fa0/1
g0/2
g0/1
Printer
170.156.44.3
Coffee Maker
170.156.44.131
PC0
PC1
170.156.44.2
170.156.44.130

Each participating router must be configured to include each of the networks it is directly connected to so it can share those routes with the other participating routers. Here is Router1's (abridged) routing table before and after configuration. Network routes marked with an "R" were learned via Router3, which in-turn learned of the two 170.156.44.0 sub-networks from Routers 1 and 2.

```
     170.156.0.0/16 is variably subnetted, 2 subnets, 2 masks
C       170.156.44.0/25 is directly connected, GigabitEthernet0/0/0
L       170.156.44.1/32 is directly connected, GigabitEthernet0/0/0
     192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0/1
L       192.168.10.1/32 is directly connected, GigabitEthernet0/0/1
```

```
     170.156.0.0/16 is variably subnetted, 4 subnets, 3 masks
R       170.156.0.0/16 [120/2] via 192.168.10.2, 00:00:10, GigabitEthernet0/0/1
C       170.156.44.0/25 is directly connected, GigabitEthernet0/0/0
L       170.156.44.1/32 is directly connected, GigabitEthernet0/0/0
R       170.156.44.128/25 [120/2] via 192.168.10.2, 00:00:10, GigabitEthernet0/0/1
     192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0/1
L       192.168.10.1/32 is directly connected, GigabitEthernet0/0/1
R    200.110.16.0/24 [120/1] via 192.168.10.2, 00:00:10, GigabitEthernet0/0/1
```

RIPv2 continued…

As shown in the first of the two screenshots, prior to being configured with a routing protocol Router1 only knows about the networks it is directly connected to. It has <u>no</u> way of communicating with the 200.110.16.0 network or the 170.156.44.128 sub-network because it has no route to reach them. But after each of the three routers are configured with RIPv2, Router1 (and other two routers) now share what routes they know with each other, and this is shown on their respective routing tables (as we see in the "after" screenshot of Router1's routing table).

RIPv2 Operating Requirements

RIPv2 AD is **120**

RIPv2 uses hop-counts to determine the best route to a network. A "hop" is literally going from one LAN to as if hopping over the routers to the next network.

In a **RIPv2** environment the routers share their entire routing tables with each other (instead of just updates like more modern routing protocols do). This makes RIPv2 simple to configure but not very efficient, which can lead to some problems.

57. Routing protocol: OSPF (Open Shortest Path First)

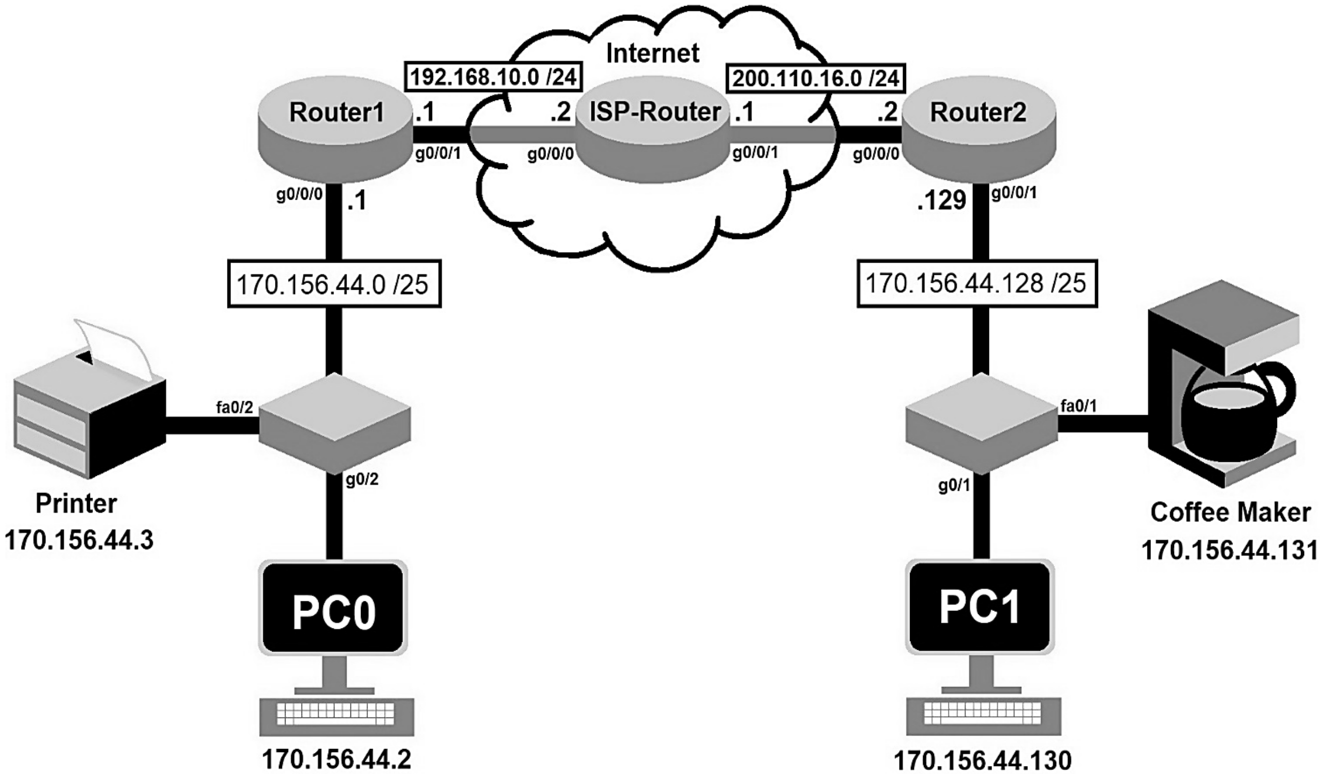
OSPF is the immediate successor to the RIP family of routing protocols. OSPF stands for Open Shortest Path First where "shortest" means the fastest route to a network based on link speed. OSPF has many additional features beyond that of RIPv2 and is more efficient. Here is how to configure OSPF (again, using the network topology on page 57).

This is an ID process number and can be any number from 1 thru 65535. OSPF routers do <u>not</u> have to have matching process ID numbers.

```
ISP-Router>
ISP-Router#conf t
ISP-Router(config)#router OSPF 3
ISP-Router(config-router)#network 170.156.44.0 0.0.0.127 area 0
ISP-Router(config-router)#network 192.168.10.0 0.0.0.127 area 0
ISP-Router(config-router)#exit
ISP-Router(config)#exit
ISP-Router#
```

OSPF networks are configured with an inverse mask.

The first area in an OSPF environment must be Area Zero. This is also known as the backbone area.


Internet
192.168.10.0 /24
200.110.16.0 /24
Router1
.1
.2
ISP-Router
.1
.2
Router2
g0/0/1
g0/0/0
g0/0/1
g0/0/0
g0/0/0
.1
.129
g0/0/1
170.156.44.0 /25
170.156.44.128 /25
fa0/2
fa0/1
g0/2
g0/1
Printer
170.156.44.3
Coffee Maker
170.156.44.131
PC0
PC1
170.156.44.2
170.156.44.130

After configuring OSPF verify correct configuration and proper function with the Show IP Route command. Here is ISP-Router's routing table. The ISP-Router has learned routes to the two non-contiguous sub-networks (indicated by code "O" for OSPF). The configuration is verified correct.

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

Gateway of last resort is not set

   170.156.0.0/25 is subnetted, 2 subnets
O     170.156.44.0/25 [110/2] via 192.168.10.1, 00:16:38, GigabitEthernet0/0/0
O     170.156.44.128/25 [110/2] via 200.110.16.2, 00:10:17, GigabitEthernet0/0/1
   192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.10.0/24 is directly connected, GigabitEthernet0/0/0
L     192.168.10.2/32 is directly connected, GigabitEthernet0/0/0
   200.110.16.0/24 is variably subnetted, 2 subnets, 2 masks
C     200.110.16.0/24 is directly connected, GigabitEthernet0/0/1
L     200.110.16.1/32 is directly connected, GigabitEthernet0/0/1

ISP-Router#
```

58. Show IP OSPF Neighbors for the ISP-Router

```
ISP-Router#sh ip ospf ne

Neighbor ID     Pri   State           Dead Time   Address         Interface
200.110.16.2      1   FULL/DR         00:00:31    200.110.16.2    GigabitEthernet0/0/1
192.168.10.1      1   FULL/BDR        00:00:31    192.168.10.1    GigabitEthernet0/0/0

ISP-Router#
```

Notice under "**State"** the categories of **FULL/DR** and **FULL/BDR.** A "DR" is the designated router for its multi-link. The Backup Designated Router is a backup for that multi-link. Explained further on page-82. This is accomplished via election.

59. Show IP OSPF Database.

Note: Although there are **7** types of LSAs, only LSA-1 and LSA-2 are generated because the topology has only the one area (Area 0).

```
ISP-Router#sh ip ospf da
            OSPF Router with ID (200.110.16.1) (Process ID 3)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
200.110.16.1    200.110.16.1    137         0x80000025 0x00df34 2
192.168.10.1    192.168.10.1    137         0x80000021 0x007fdd 2
200.110.16.2    200.110.16.2    136         0x80000020 0x006427 2

                Net Link States (Area 0)
Link ID         ADV Router      Age         Seq#       Checksum
192.168.10.2    200.110.16.1    1034        0x80000018 0x0071f8
200.110.16.2    200.110.16.2    1329        0x8000001a 0x004ad4
ISP-Router#
```

States of Routers in an OSPF Environment

Important: DRs and BDRs (and DROTHERs) are elected by shared multi-access network segment and not by area. Routers that are not elected to be either DRs or BDRs become DROTHERs.

DR	The **Designated Router** is the central distributor of routing updates on its multi-access network segment. DRs are elected in order according to Router ID, or highest loopback, or highest interface address.
BDR	**Backup Designated Router** is the failover distributor of routing updates on its multi-access network segment if the DR becomes unavailable.
DROTHER	A **Designated Router Other** is a router that is neither a DR nor BDR. DROTHER routers exchange Hello messages only.

On the 192.168.10.0 network the ISP-Router is the DR because it has the highest interface IP address. But on network 200.110.16.0 Router2 is the DR because its IP address 200.110.16.**2** is greater than ISP-Router's IP address 200.110.16.**1**

OSPF Operating Requirements

OSPF AD is **110**

All **OSPF routers** must have the same Area number and Hello and Dead timers must match for the routers to become neighbors. The default setting for Hello and Dead timers are 10 and 40 seconds, respectively.

Options = If authentication is being used on a router it must also be the same on all OSPF routers.

Area = All OSPF environments must begin with an Area 0. This is also referred to as the “backbone” area.

OSPF is a Link State routing protocol. Because it uses **Link State Advertisements (LSAs)** to share routes.

Neighbor ID = Is the ID of an OSPF neighbor router. A router’s ID is decided in one of three ways: First = if it is statically assigned. Second = the highest loopback address. Third = the highest interface IP address.

State = A description of the role of routers and routes in an OSPF environment.

Inverse mask = OSPF network configurations use an inverse mask to *instruct the routing protocol* (not the devices) where the address boundaries are. Optionally and for a bit more security, inverse masks can be configured with 1 bit less than the actual network boundary to restrict the range of address lookup.

Types of Link State Advertisements (LSAs)

Link State Advertisements are the routing updates routers exchange with each other

Type 1 LSA = A Router LSA generated by each router for the area it is located in. In the link-state ID you will find the originating router's ID.

Type 2 LSA = A Network LSA generated by the DR. The link-state ID will be the interface IP address of the DR.

Type 3 LSA = The summary LSA is created by an ABR and flooded into other areas.

Type 4 LSA = This is an LSA generated by the ABR to let the other routers where the ASBR is located. Type 4 LSAs also include the router ID of the ASBR in the link-state ID field.

Type 5 LSA = This LSA is generated for advertising route redistribution between an OSPF network area and another network using a different routing protocol. Route redistribution allows networks to share routing information with each other when their routing protocols are different.

Type 6 LSA = This is a multicast LSA that is not supported and not used.

Type 7 LSA = An external LSA used by NSSAs when participating in route redistribution. Type 5 LSAs are converted to type 7 LSAs by the ASBR.

60. Routing protocol: OSPF / Stub areas and NSSA Explained

OSPF Stub networks operate slightly differently than standard OSPF networks. A shown here.

Area type	Features
Standard	All LSAs accepted. All standard OSPF features.
Stub	No LSA type-5.
Totally stubby	No LSA types 3 and 5.
NSSA	No LSA type 5. Router in the NSSA becomes an ASBR (Autonomous System Border Router) and redistributed routes become LSA type 7. The ABR (Area Border Router) converts type 7 LSAs to type 5 LSAs.
Totally NSSA	No LSA types 3 and 5. Routers and route redistribution treated the same as NSSA.

OSPF Network Areas

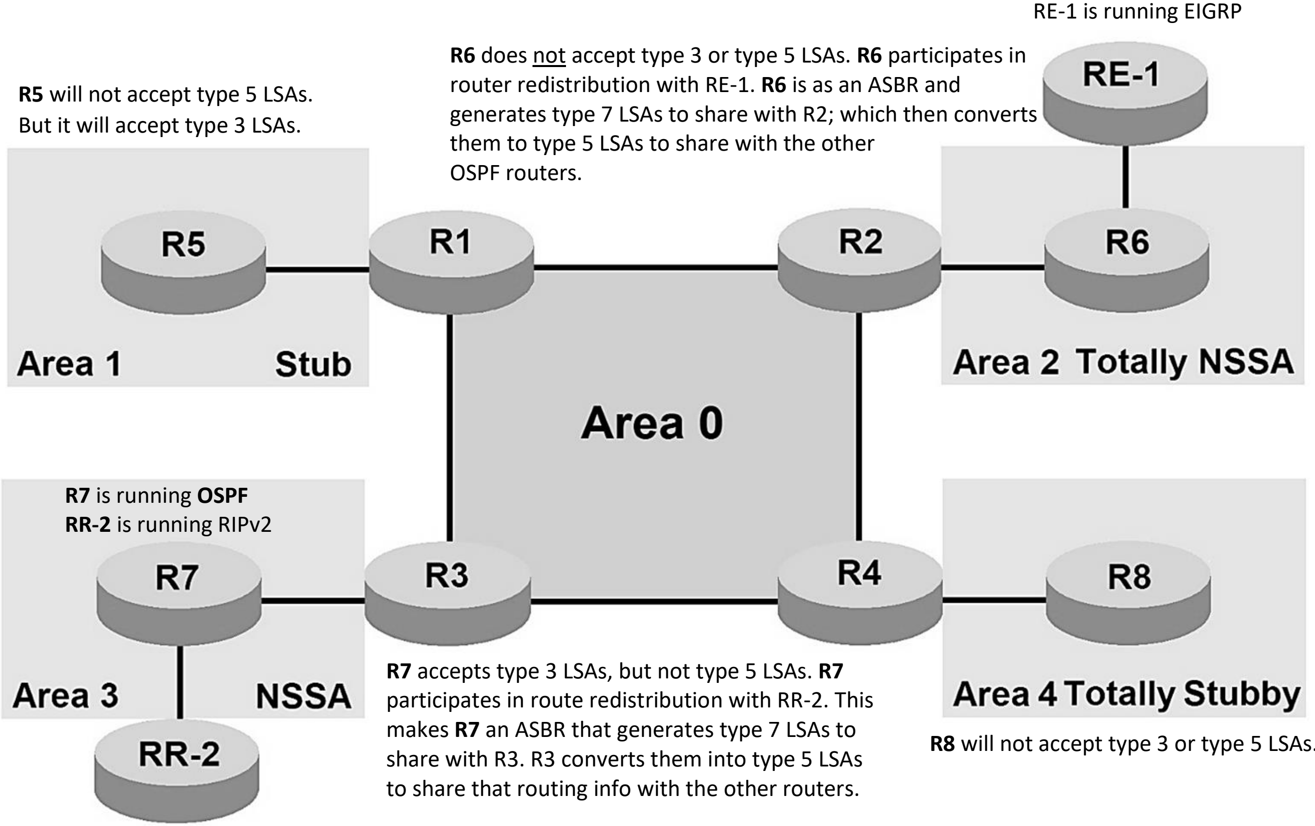
Stub areas are OSPF specific networks usually connected to the OSPF backbone Area. Stub networks have only one way in and out between themselves and another connected OSPF area via an ABR (Area Border Router). Stub areas accept type 3 LSAs, but not type 5 LSAs (no route redistribution).

Totally Stubby areas are more isolated than Stub areas, accepting neither type 3 LSAs nor type 5 LSAs.

NSSAs (Not So Stubby Areas) do accept type 3 LSAs. But they do <u>not</u> accept type 5 LSAs, instead accepting route redistribution by converting type 5 LSAs to type 7 LSAs. Routers in NSSAs become an Autonomous System Border Router (ASBRs). While type 7 LSAs forwarded to an ABR are converted back to type 5 LSAs for advertising to the rest of the OSPF environment.

Totally NSSA networks behave like Totally Stubby networks with one exception: while they also do <u>not</u> accept type 3 LSAs, they <u>do</u> accept route redistribution like NSSA networks do.

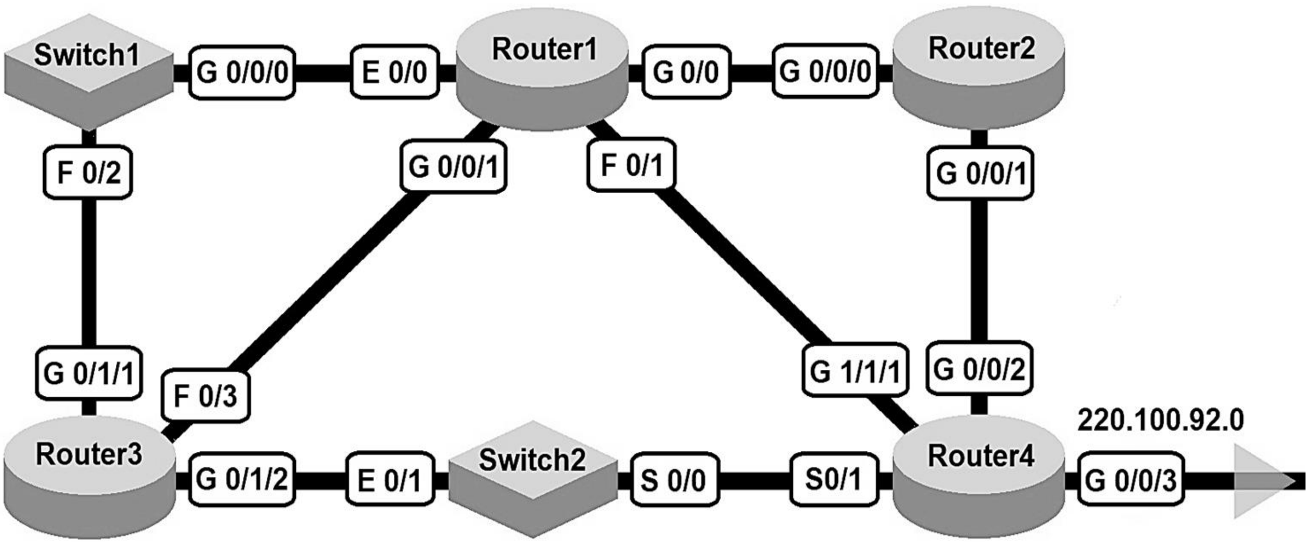
Understanding OSPF Area Types



61. Routing protocol: OSPF / Simple Fastest Route Formula

OSPF uses Dijkstra's algorithm. But there is an easier formula we can use to determine the fastest (lowest cost) route: Serial interface=Cost **64** / Ethernet interface=Cost **10** / Fast Ethernet and above=Cost **1**

Question: Which route will OSPF use from **Switch1** to the **220.100.92.0** network? **Answer:** next page



62. Routing protocol: EIGRP (Enhanced Interior Gateway Routing Protocol)

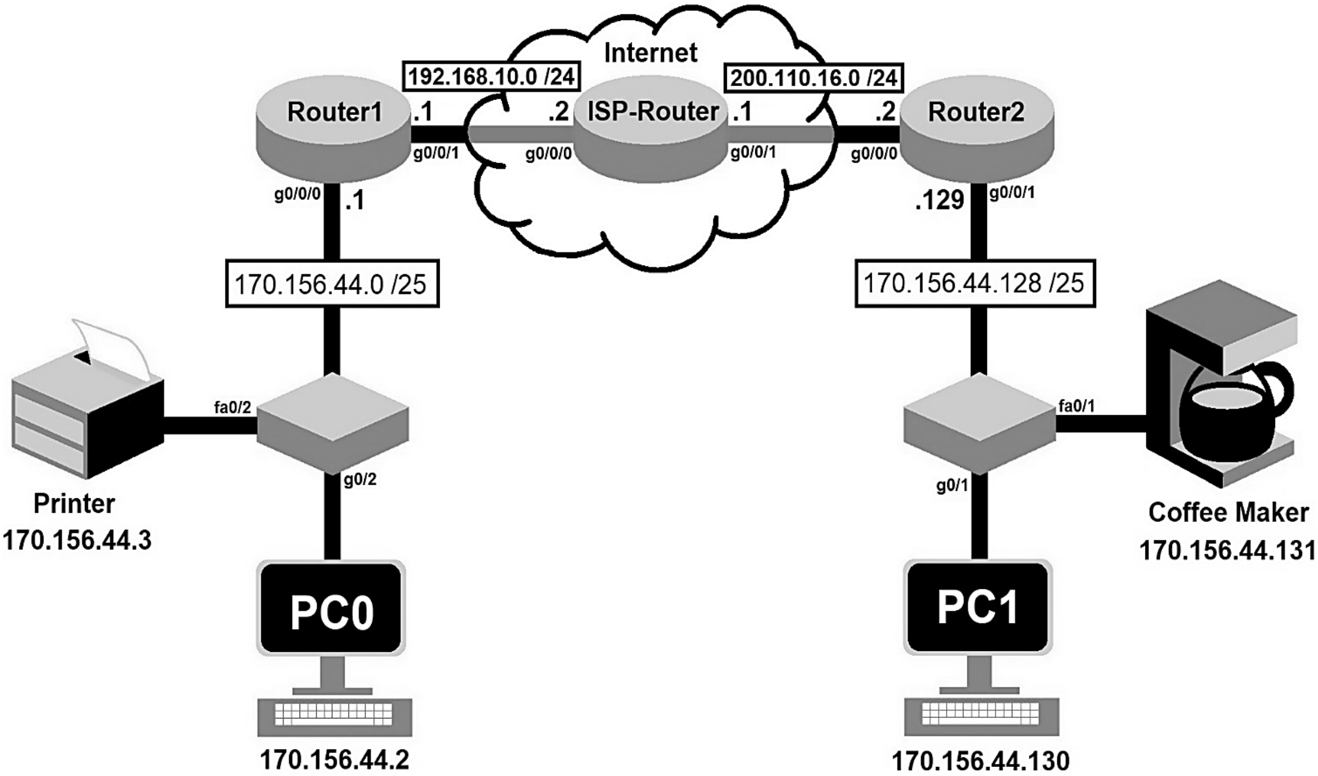
EIGRP succeeds OSPF and is much more efficient. In this section we will use Router1 to show how to configure EIGRP on all the routers in our example network topology (page 57).

```
Router1#conf t
Router(config)#router eigrp 1
Router1(config-router)#network 170.156.44.0 0.0.0.127
Router1(config-router)#network 192.168.10.0 0.0.0.255
Router1(config-router)#no auto-summary
Router1(config-router)#exit
Router1(config)#exit
```

> All EIGRP routers in the same environment must have the same AS (autonomous system) number.

Note: When verifying the configuration with the **Show Running-Configuration** command, the “no auto-summary” option will usually not show up in the report. This is because auto-summary is disabled by default. It is entered here only as a best practice to make sure it *is* disabled.

Answer to previous question:Switch1 >Router3 > Router1 > Router2 > Router4 >Network 220.100.92.0


Internet
192.168.10.0 /24
200.110.16.0 /24
Router1
.1
.2
ISP-Router
.1
.2
Router2
g0/0/1
g0/0/0
g0/0/1
g0/0/0
g0/0/0
.1
.129
g0/0/1
170.156.44.0 /25
170.156.44.128 /25
fa0/2
fa0/1
g0/2
g0/1
Printer
170.156.44.3
Coffee Maker
170.156.44.131
PC0
PC1
170.156.44.2
170.156.44.130

When all three routers are configured, verify proper configuration by checking Router1's routing table. Here we see by code "D" that Router1 it has learned routes to the other two networks via EIGRP. The numbers in **[brackets]** are the AD and the computed cost to reach that network.

```
Router1#sh ip ro
 Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
        i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
        * - candidate default, U - per-user static route, o - ODR
        P - periodic downloaded static route

  Gateway of last resort is not set

   170.156.0.0/16 is variably subnetted, 3 subnets, 2 masks
C     170.156.44.0/25 is directly connected, GigabitEthernet0/0/0
L     170.156.44.1/32 is directly connected, GigabitEthernet0/0/0
D     170.156.44.128/25 [90/3328] via 192.168.10.2, 00:01:15,GigabitEthernet0/0/1
   192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.10.0/24 is directly connected, GigabitEthernet0/0/1
L     192.168.10.1/32 is directly connected, GigabitEthernet0/0/1
D  200.110.16.0/24 [90/3072] via 192.168.10.2, 00:03:19, GigabitEthernet0/0/1

Router1#
```

EIGRP Operating Requirements

EIGRP AD is **90**   EIGRP uses the **DUAL** algorithm

AS (Autonomous System) number <u>must</u> match on all routers participating in the same EIGRP environment.

Matching K-Values (turned on or turned off) on each participating router must be the same.

Same sub-net = Routers on the <u>same</u> segment <u>must</u> be within the same network boundary (or sub-net).

Optional = Security settings (if used) must also match on each participating router.

EIGRP is a distance vector <u>and</u> link state routing protocol. EIGRP uses the Diffusing Update Algorithm (DUAL)

K-Values

K-Values are the adjustable interface properties used by EIGRP. They can be modified to change EIGRP routing preferences. **K1 =** Bandwidth   **K2 =** Load   **K3 =** Delay   **K4 =** Reliability   **K5 =** MTU

The **Default K-Value** settings: **K1= 1**   K2= 0   **K3= 1**   K4= 0   K5= 0

The command **Show IP Protocols** allows one to verify the **AS** number and **K-Value** settings. This command is of particular use when troubleshooting

```
Router1#sh ip prot

Routing Protocol is "eigrp  1 "
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  EIGRP maximum hopcount 100
  EIGRP maximum metric variance 1
Redistributing: eigrp 1
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
     170.156.44.0/25
     192.168.10.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.10.2    90            200550439
  Distance: internal 90 external 170

Router1#
```

To verify the status of interfaces on EIGRP routers, use the **Show IP EIGRP Topology** command. Note: **Passive** status means the interface is operating normally. While **Active** indicates there is a problem.

```
Router1#sh ip eigrp topology
IP-EIGRP Topology Table for AS 1/ID(192.168.10.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - Reply status

P 170.156.44.0/25, 1 successors, FD is 5120
         via Connected, GigabitEthernet0/0/0
P 170.156.44.128/25, 1 successors, FD is 3328
         via 192.168.10.2 (3328/3072), GigabitEthernet0/0/1
P 192.168.10.0/24, 1 successors, FD is 2816
         via Connected, GigabitEthernet0/0/1
P 200.110.16.0/24, 1 successors, FD is 3072
         via 192.168.10.2 (3072/2816), GigabitEthernet0/0/1
Router1#
```

Issue the command **Show IP EIGRP Neighbors** to verify neighbor relationships between routers. In this report Gig/0/0/1 is local to Router1 and its neighbor's interface is 192.168.10.2

```
Router1#sh ip eigrp ne
Router1#sh ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address          Interface        Hold Uptime    SRTT   RTO   Q    Seq
                                      (sec)          (ms)         Cnt  Num
0   192.168.10.2     Gig0/0/1         12   07:07:26  40     1000  0    7

Router1#
```

The command **Show IP EIGRP Interfaces 1** allows one to verify the status of participating interfaces. In this report an EIGRP Peer relationship is shown to exist via Router1 interface Gig0/0/1.

```
Router1#sh ip eigrp interfaces 1
IP-EIGRP interfaces for process 1

                         Xmit Queue   Mean   Pacing Time   Multicast    Pending
Interface        Peers   Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Gig0/0/0           0        0/0       1236      0/10           0           0
Gig0/0/1           1        0/0       1236      0/10           0           0
Router1#
```

Finally, the command **Show IP EIGRP Traffic** can be used to show the real-time EIGRP traffic on a router. Simply repeat the command to see the counters changing in real-time.

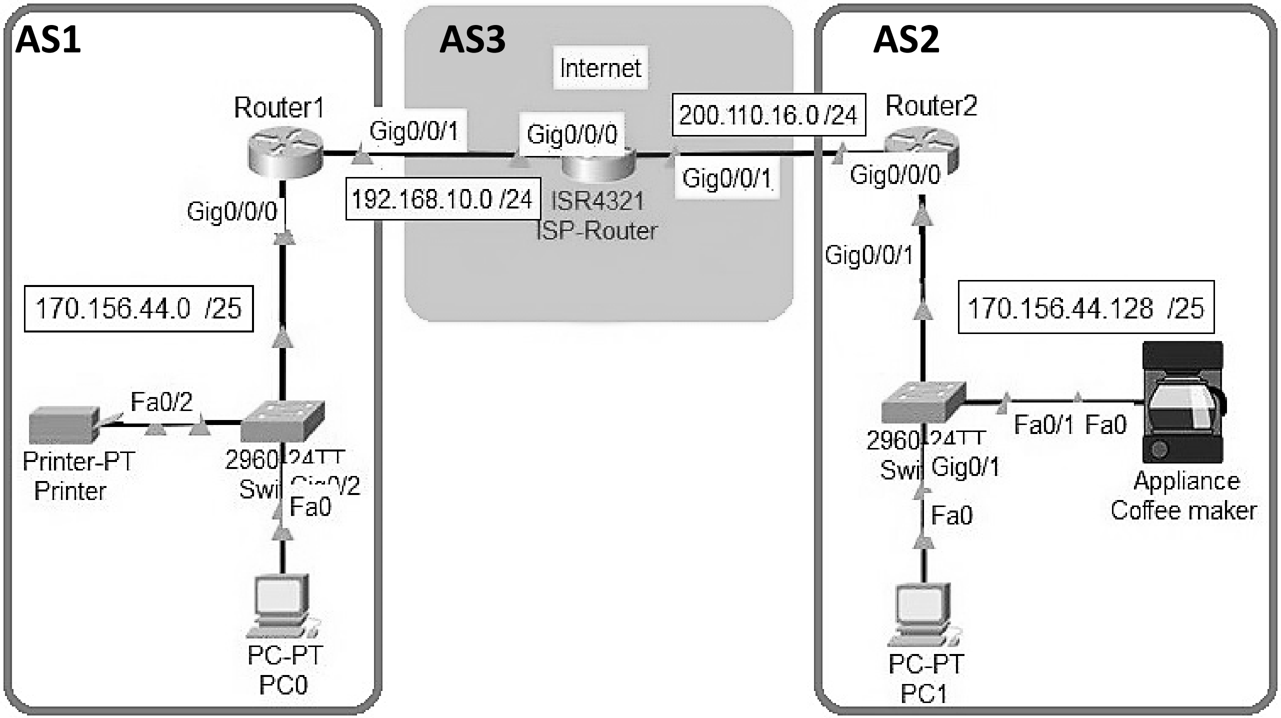
```
Router1#sh ip eigrp traffic 1
IP-EIGRP Traffic Statistics for process 1
  Hellos sent/received: 11160/5566
  Updates sent/received: 4/4
  Queries sent/received: 0/0
  Replies sent/received:  0/0
  Acks sent/received:  4/3
  Input queue high water mark 1, 0 drops
  SIA-Queries sent/received: 0/0
  SIA-Replies sent/received: 0/0
```

63. Routing protocol: BGP (Border Gateway Protocol)

In a **BGP** environment, routers do not form neighbor relationships the same way that they do in OSPF or EIGRP. Instead, BGP establishes BGP pairings to communicate between autonomous systems. However, this does not mean that BGP routers can skip over non-BGP routers on a network segment. To do that, BGP would need route redistribution like any other protocol would. The difference is that in a BGP environment networks are regarded as autonomous areas that communicate with each other. Therefore, when configuring BGP in our topology, Router1 and Router2 only need to advertise the networks in their autonomous systems. While the ISP-Router advertises both networks in its own.

```
Router1#conf t
Router1(config)#router bgp 1
Router1(config-router)#neighbor 192.168.10.2 remote-as 3
Router1(config-router)#network 170.156.44.0 mask 255.255.255.0
Router1(config-router)#exit
Router1(config)#exit
```

Using the BGP configuration as described on the previous page, the WAN is logically divided it into three BGP autonomous systems as shown here:



Proper configuration can be verified with the **Show IP Route** command (shown here abridged). Notice that Router1 has learned the route to AS2's network from its neighbor the ISP-Router in AS3.

```
     170.156.0.0/16 is variably subnetted, 3 subnets, 2 masks
C       170.156.44.0/25 is directly connected, GigabitEthernet0/0/0
L       170.156.44.1/32 is directly connected, GigabitEthernet0/0/0
B       170.156.44.128/25 [20/0] via 192.168.10.2, 00:00:00
     192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0/1
L       192.168.10.1/32 is directly connected, GigabitEthernet0/0/1
B    200.110.16.0/24 [20/0] via 192.168.10.2, 00:00:00

Router1#
```

We can also verify Router1's BGP neighbor by using the "**sh run**" command and scrolling down.

```
!
router bgp 1
 neighbor 192.168.10.2 remote-as 3
 network 170.156.44.0 mask 255.255.255.128
!
```

64. Routing protocol: BGP (Border Gateway Protocol) Show IP BGP

Use the **Show IP BGP** command and its options to get a detailed report on its routes. BGP functions can be seen transiting externally and internally. Notice the hop count resembles a traceroute.

```
Router1#sh ip bgp
BGP table version is 36, local router ID is 192.168.10.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 170.156.44.0/25   0.0.0.0                  0      0 32768 i
*> 170.156.44.128/25 192.168.10.2             0      0     0 3 2 i
*> 200.110.16.0/24   192.168.10.2             0      0     0 3 i

Router1#
```

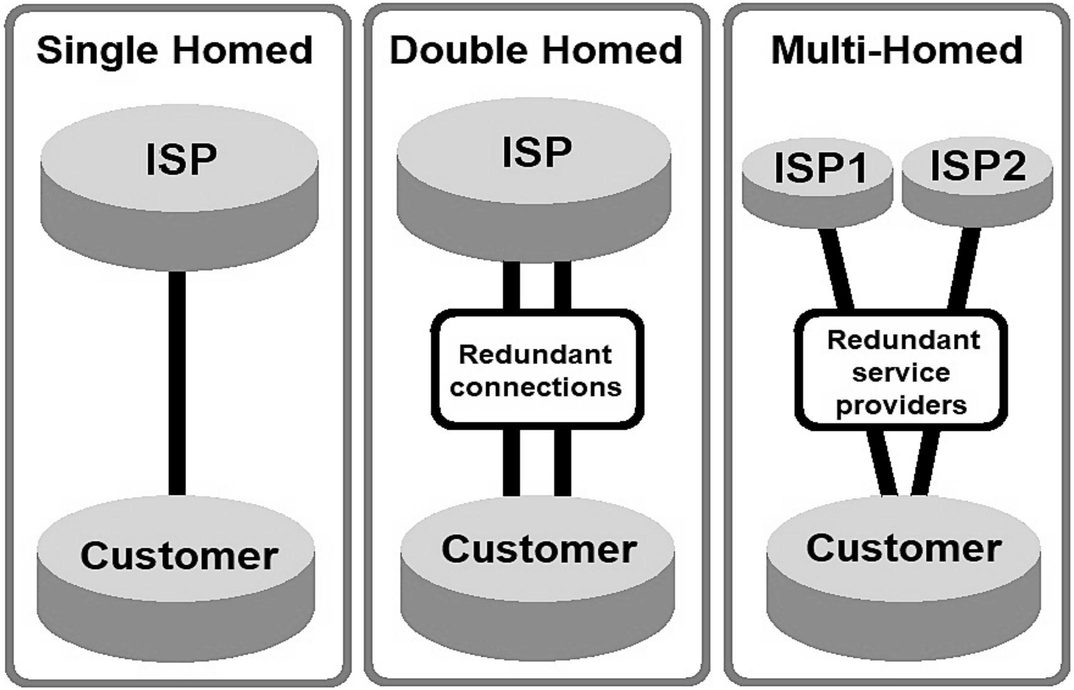
Show IP BGP Neighbors (shown here abridged) gives us more information from Router1's perspective. Notice it also shows the **remote ID** of the ISP-Router (of the *remote* network in AS3, again from Router1's perspective). Router1 is directly connected to 192.168.10.0. in AS3. But 200.110.16.0 (while also in AS3) is remote.

```
Router1#sh ip bgp ne
BGP neighbor is 192.168.10.2,  remote AS 3, external link
  BGP version 4, remote router ID 200.110.16.1
  BGP state = Established, up for 06:59:10
  Last read 06:59:10, last write 06:59:10, hold time is 180, keepalive 60 sec
  Neighbor capabilities:
    Route refresh: advertised and received(new)
    Address family IPv4 Unicast: advertised and received
  Message statistics:
    InQ depth is 0
    OutQ depth is 0
```

Another useful show command for BGP with over a full screen of information is **Show IP BGP Summary**

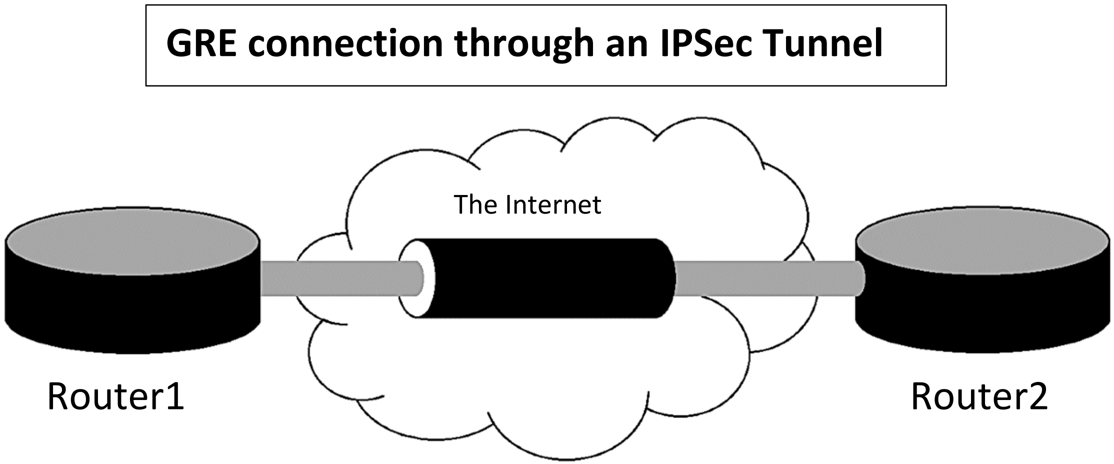
65. Routing protocol: BGP / Single Homed, Double Homed, and Multi-Homed Topologies

In the previous example a **Single Homed** BGP network was configured. This is a common topology for home networks and small business networks. However, in larger networks redundancy is desirable.

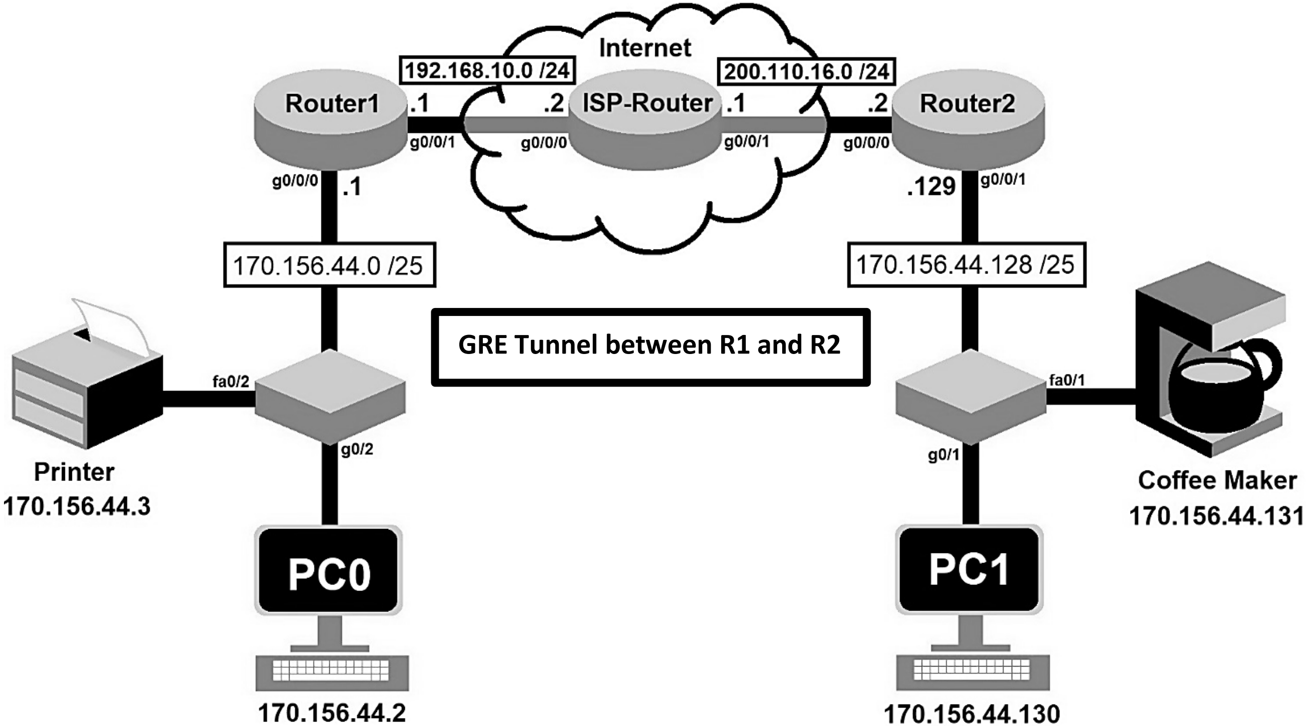


66.Routing protocol: GRE (Generic Routing Encapsulation) Tunnel

GRE allows for point-to-point connectivity via “tunneling”. This also allows the encapsulation of one routing protocol inside of another to transmit data without the need for route redistribution. This is also referred to as “tunneling” and so, to accomplish this, point-to-point GRE tunnels are configured.



For this section we will use the previous **EIGRP** version of our topology, making sure to add our GRE tunnel network IP address to the EIGRP 1 list of network so that EIGRP will work over the tunnel.



GRE Tunnel configuration:

```
Router1>en
Router1#conf t
Router1(config)#int tunnel 1
Router1(config-if)#ip address 170.156.45.1 255.255.255.252
Router1(config-if)#tunnel source g0/0/1
Router1(config-if)#tunnel destination 200.110.16.2
Router1(config-if)#tunnel mode gre ip
Router1(config-if)#exit
Router1(config)#exit
Router1#
```

A custom IP address is configured for the tunnel. For added security a 30-bit mask is used to allow only two IP addresses

Source is the front-facing interface on **Router1**.

Destination is the IP address of the front-facing interface on **Router2**.

GRE Tunnel is completed with a complimentary configuration on Router2, using Router2's front-facing interface as the source (as configured on Router1) and 170.156.45.2 as the IP address for Router2's end of the GRE Tunnel. **Note:** depending on the device manufacturer and iOS, inputting the "?" after "tunnel source" and/or "destination" may provide a menu of additional choices.

First, verify the GRE Tunnel is up with the **Show IP Interface Brief** command.

```
Router1#sh ip int br
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0/0   170.156.44.1    YES manual up                    up
GigabitEthernet0/0/1   192.168.10.1    YES manual up                    up
Tunnel1                170.156.45.1    YES manual up                    up
Vlan1                  unassigned      YES unset  administratively down down
Router1#
```

Next, verify Tunnel 1's route with the **Show IP Route** command (shown here, abridged).

```
   170.156.0.0/16 is variably subnetted, 5 subnets, 3 masks
C     170.156.44.0/25 is directly connected, GigabitEthernet0/0/0
L     170.156.44.1/32 is directly connected, GigabitEthernet0/0/0
D     170.156.44.128/25 [90/3328] via 192.168.10.2, 00:30:56, Gig 0/0/1
C     170.156.45.0/30 is directly connected, Tunnel1
L     170.156.45.1/32 is directly connected, Tunnel1
   192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.10.0/24 is directly connected, GigabitEthernet0/0/1
L     192.168.10.1/32 is directly connected, GigabitEthernet0/0/1
D  200.110.16.0/24 [90/3072] via 192.168.10.2, 00:32:23, Gig 0/0/1
```

The command **Show IP EIGRP Topology** also verifies that Tunnel 1 is operational and participating in the EIGRP environment.

```
Router1#sh ip eigrp top
IP-EIGRP Topology Table for AS 1/ID(192.168.10.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - Reply status

P 170.156.44.0/25, 1 successors, FD is 5120
         via Connected, GigabitEthernet0/0/0
P 170.156.44.128/25, 1 successors, FD is 3328
         via 192.168.10.2 (3328/3072), GigabitEthernet0/0/1
         via 170.156.45.2 (26880256/2816), Tunnel1
P 170.156.45.0/30, 1 successors, FD is 26880000
         via Connected, Tunnel1
P 192.168.10.0/24, 1 successors, FD is 2816
         via Connected, GigabitEthernet0/0/1
P 200.110.16.0/24, 1 successors, FD is 3072
         via 192.168.10.2 (3072/2816), GigabitEthernet0/0/1
         via 170.156.45.2 (26880256/2816), Tunnel1
```

Show IP Interface Tunnel 1

```
interface Tunnel1
 ip address 170.156.45.1 255.255.255.252
 mtu 1476
 tunnel source GigabitEthernet0/0/1
 tunnel destination 170.156.45.2
```

The command **Show Interface Tunnel 1** provides the following (abridged) information.

```
Router1#sh int tunnel 1
Tunnel1 is up, line protocol is up (connected)
  Hardware is Tunnel
  Internet address is 170.156.45.1/30
  MTU 17916 bytes, BW 100 Kbit/sec, DLY 50000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 192.168.10.1 (GigabitEthernet0/0/1), destination 200.110.16.2
  Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
```

We can test the tunnel by pinging Router2's tunnel address. However, a better validation method is to use the **Traceroute** command. If the tunnel is correctly configured the route will show only one "hop" from Router1 directly to Router2. This is because a GRE Tunnel is a point-to-point protocol and should pass through the ISP-Router without stopping.

Success!

```
Router1#traceroute 170.156.45.2
Type escape sequence to abort.
Tracing the route to 170.156.45.2

  1   170.156.45.2    1 msec    0 msec    0 msec
Router1#
```

We can also run tracert from PC2 to Router1's end of the tunnel and the result is the same. PC2 uses its default gateway as normal, and then it is only 1 hop to Router1.

```
C:\>tracert 170.156.45.1

Tracing route to 170.156.45.1 over a maximum of 30 hops:

  1   0 ms      0 ms      0 ms      170.156.44.129
  2   12 ms     0 ms      0 ms      170.156.45.1

Trace complete.
```

Again: Success! Even though Router1 and Router2 are physically separated by two other networks + a third router (the ISP-Router) GRE tunnelling creates a direct connection between Router1 and Router2.

Next, we will configure our tunnel with secure encryption.

67. Routing protocol: GRE / Tunnel Security / ISAKMP and IPSEC

Now that our GRE Tunnel is verified as operating within normal parameters, let's give our tunnel a layer of security via encryption and hashing. We will begin with part 1: Configuring **ISAKMP**. Note: this configuration will need to be mirrored on Router1.

```
Router1>en
Router1#conf t
Router1(config)#crypto isakmp policy 3
Router1(config-isakmp)#encryption aes
Router1(config-isakmp)#authentication pre-share
Router1(config-isakmp)#group 2
Router1(config-isakmp)#exit
Router1(config)#exit
```

The **Advanced Encryption Standard**.
AES's default is 128-bit encryption keys

Diffie-Hellman 2 = 1024-bit modulus

Part 2: Configuring the IPSEC Pre-Share Key and Mapping it all together

Next, we need to configure a **policy map** and a **pre-share key** that Router1 and Router2 will use to encrypt and decrypt data traveling over our tunnel.

```
Router1#conf t
Router1(config)#crypto map MYMAP client authentication list MYKEY
Router1(config)#crypto ipsec transform-set MYSET esp-aes esp-sha-hmac
Router1(config)#crypto isakmp enable
Router1(config)#crypto isakmp policy 10
Router1(config)#encryption aes 128
Router1(config)#authentication pre-share
Router1(config)#group 2
Router1(config)#exit
Router1(config)#crypto isakmp key MYKEY address 200.110.16.2 0.0.0.0
Router1(config)#ip access-list extended MYLIST
Router1(config)#permit gre any any
```

```
Router1(config)#permit esp any any
Router1(config)#exit
Router1(config)#int g0/0/1
Router1(config)#crypto map MYMAP
```

```
Router1(config)#int g0/0/1
Router1(config-if)#crypto map MYMAP
*Jan  3 07:16:26.785: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
Router1(config-if)#exit
Router1(config)#exit
Router1
```

A correct configuration will result in this message (shown above). **Note:** to ensure that your device generates crypto isakmp debug messages issue this command:

```
Router1#debug crypto isakmp
```

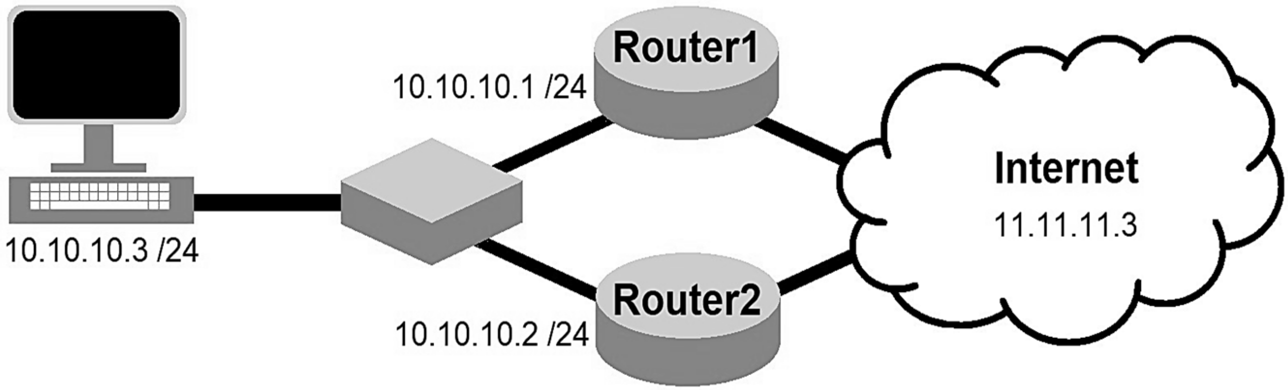
Now let’s verify the security settings with the command: **Show Crypto ISAKMP Policy.**

```
Router1#sh crypto isakmp policy

Global IKE policy
Protection suite of priority 10
        encryption algorithm:   AES - Advanced Encryption Standard (128 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #2 (1024 bit)
        lifetime:               86400 seconds, no volume limit
Router1#
```

Finally, the **Show Crypto ISAKMP SA** command will report the operational status of our secure tunnel. However, for this report to generate information some “interesting” traffic must be sent over the tunnel. Interesting traffic is defined as anything that would activate the encryption process, such remote sessions, DNS queries, email, images, or other data. Whether using live equipment, or a virtual environment such as GNS3 or Packet-Tracer, establishing a remote session via telnet or SSH through the tunnel should create some interesting traffic for Show Crypto ISAKMP SA to report.

68. Routing protocol: HSRP (Hot Standby Router Protocol)

HSRP allows a backup router to become automatically available to the network if the primary router is disabled. Let's say that in this example topology Router1 is the LAN default gateway, and we want Router2 to be the Hot Standby Router so our PC can still access the internet in case Router1 goes down.



Configure **Router1** as the **primary router** and Router2 as the backup.

```
Router1#conf t
Router1(config)#int g0/0/0
Router1(config-if)#standby 10 ip 11.11.11.3
Router1(config-if)#standby 10 preempt
Router1(config-if)#standby 10 priority 120
Router1(config-if)#exit
Router1(config)#exit
```

The **standby group** number is a custom value that you choose. All routers in the same group must have the same group number.

The **priority** value determines which router will be the primary router, with the router having the highest value elected to be the primary router. The router with the lower priority becomes the standby router. The default priority value is 100. Here, it is changed to 120 so Router1 will be elected as the primary router.

```
Router2#conf t
Router2(config)#int g0/0/1
Router2(config-if)#standby 10 ip 11.11.11.3
Router2(config-if)#standby 10 preempt
Router2(config-if)#exit
Router2(config)#exit
Router2#
```

Verify standby configurations by issuing the **Show Standby Brief** command on both routers.

```
Router1#sh standby br
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State    Active     Standby        Virtual IP
Gig0/0/0    10   120 P Active   local      10.10.10.2     11.11.11.3
Router1#
```

```
Router2#sh standby br
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State    Active          Standby      Virtual IP
Gig0/0/1    10   100 P Standby  10.10.10.1      local        11.11.11.3
Router2#
```

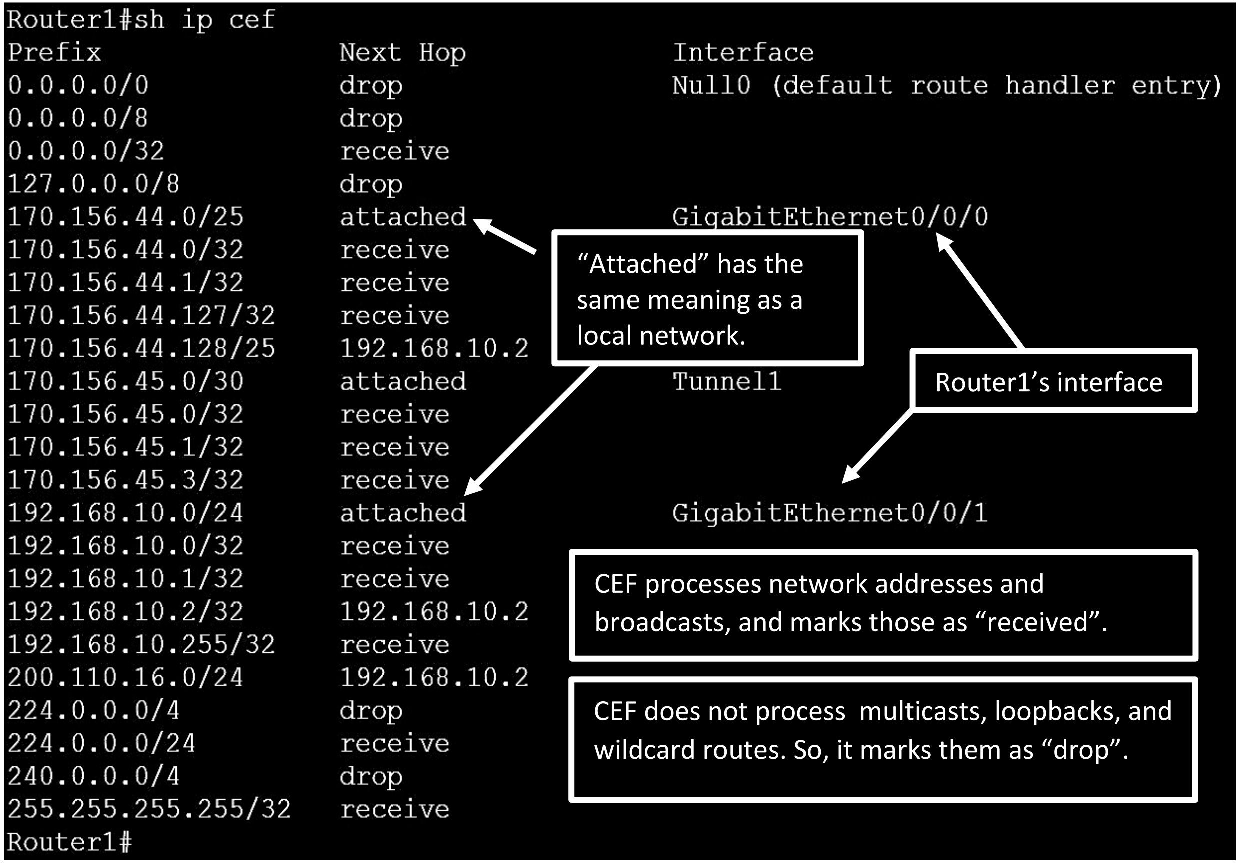
Router1 has the higher priority, so it is the **Active** router. While Router2 is now in a Standby state. Notice also how each router's report lists the IP address of both the active and standby router within their group.

69. Routing protocol: CEF (Cisco Express Forwarding)

Cisco Forwarding Express Forwarding (CEF) is the successor to process switching wherein the processor on a device used to handle all remote connections. CEF encompasses two very useful tables: The FIB and CEF Adjacency tables.

FIB (Forwarding Information Base) is a layer-3 process similar to a routing table as it is updated whenever a router's routing table is updated. The **CEF Adjacency table** is a layer-2 process that contains information about the next hop on a LAN segment by using MAC addresses to identify connections between device interfaces. On the next page is the report returned when using the **Show IP CEF command** on Router1 of our example network. **Note:** On some devices the Show IP CEF command must be enabled with this configuration:

```
Router1#conf t
Router1(config)#sh ip cef
Router1(config)#exit
```




```
Router1#sh ip cef
Prefix              Next Hop            Interface
0.0.0.0/0           drop                Null0 (default route handler entry)
0.0.0.0/8           drop
0.0.0.0/32          receive
127.0.0.0/8         drop
170.156.44.0/25     attached            GigabitEthernet0/0/0
170.156.44.0/32     receive
170.156.44.1/32     receive
170.156.44.127/32   receive
170.156.44.128/25   192.168.10.2
170.156.45.0/30     attached            Tunnel1
170.156.45.0/32     receive
170.156.45.1/32     receive
170.156.45.3/32     receive
192.168.10.0/24     attached            GigabitEthernet0/0/1
192.168.10.0/32     receive
192.168.10.1/32     receive
192.168.10.2/32     192.168.10.2
192.168.10.255/32   receive
200.110.16.0/24     192.168.10.2
224.0.0.0/4         drop
224.0.0.0/24        receive
240.0.0.0/4         drop
255.255.255.255/32  receive
Router1#
```

Next is the **CEF Adjacency Detail** command.

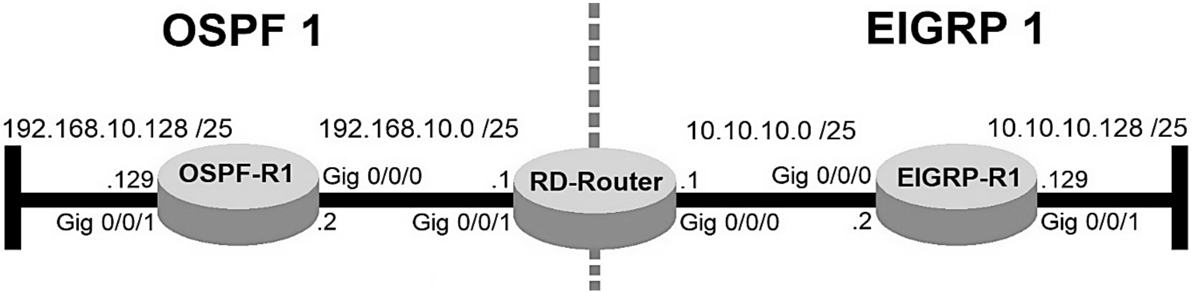
```
Router1>en
Router1#conf t
Router1#sh adj de
Protocol Interface                Address
IP       GigabitEthernet 0/0/0    192.168.10.1(11)
                                  0 packets, 0 bytes
                                  epoch 0
                                  sourced in sev-epoch 0
                                  Encap length 14
                                  0040.0BAA.40010060.472A.B0010800
                                  ARP
```

Source MAC (Router1) followed by destination MAC (Router2)

Bonus info: the 0800 at the end of the MAC string is the process number.

70. Routing protocol: Route Redistribution

Route Redistribution allows for two networks running different routing protocols to exchange routes with each other. So, in a way Route Redistribution works like a translator.




```
RD-Router#conf t
RD-Router(config)#router ospf 1
RD-Router(config-router)#redistribute eigrp 1 subnets
RD-Router(config-router)#router eigrp 1
RD-Router(config-router)#redistribute ospf 1
RD-Router(config-router)#exit
```

Note: Some manufacturers include subnets by default, while others require it only be entered once to activate for all following entries.

OSPF-R1 before Route Redistribution.

```
     192.168.10.0/24 is variably subnetted, 4 subnets, 2 masks
C       192.168.10.0/25 is directly connected, GigabitEthernet0/0/0
L       192.168.10.2/32 is directly connected, GigabitEthernet0/0/0
C       192.168.10.128/25 is directly connected, GigabitEthernet0/0/1
L       192.168.10.129/32 is directly connected, GigabitEthernet0/0/1
OSPF-R1#
```

EIGRP-R1 before Route Redistribution.

```
     10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C       10.10.10.0/25 is directly connected, GigabitEthernet0/0/0
L       10.10.10.2/32 is directly connected, GigabitEthernet0/0/0
C       10.10.10.128/25 is directly connected, GigabitEthernet0/0/1
L       10.10.10.129/32 is directly connected, GigabitEthernet0/0/1
EIGRP-R1#
```

OSPF-R1 and **EIGRP-R2** after redistribution is configured on the RD-Router. Notice the *new* network entries appear with codes showing they have been redistributed to the other router's routing protocol.

```
     10.0.0.0/25 is subnetted, 1 subnets
O       10.10.10.0/25 [110/2] via 192.168.10.1, 00:08:48, GigabitEthernet0/0/0
     192.168.10.0/24 is variably subnetted, 4 subnets, 2 masks
C       192.168.10.0/25 is directly connected, GigabitEthernet0/0/0
L       192.168.10.2/32 is directly connected, GigabitEthernet0/0/0
C       192.168.10.128/25 is directly connected, GigabitEthernet0/0/1
L       192.168.10.129/32 is directly connected, GigabitEthernet0/0/1

OSPF-R1#
```

```
     10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C       10.10.10.0/25 is directly connected, GigabitEthernet0/0/0
L       10.10.10.2/32 is directly connected, GigabitEthernet0/0/0
C       10.10.10.128/25 is directly connected, GigabitEthernet0/0/1
L       10.10.10.129/32 is directly connected, GigabitEthernet0/0/1
     192.168.10.0/25 is subnetted, 1 subnets
D       192.168.10.0/25 [90/3072] via 10.10.10.1, 00:10:27, GigabitEthernet0/0/0

EIGRP-R1#
```

Issuing the command Show Running Configuration on RD-Router also verifies proper configuration.

```
!
router eigrp 1
 redistribute ospf 1
 network 10.10.10.0 0.0.0.127

!
router ospf 1
 redistribute eigrp 1 subnets
 network 192.168.10.0 0.0.0.127 area 0
!
```

71. Routing protocol: MPLS (Multi-Protocol Label Switching)

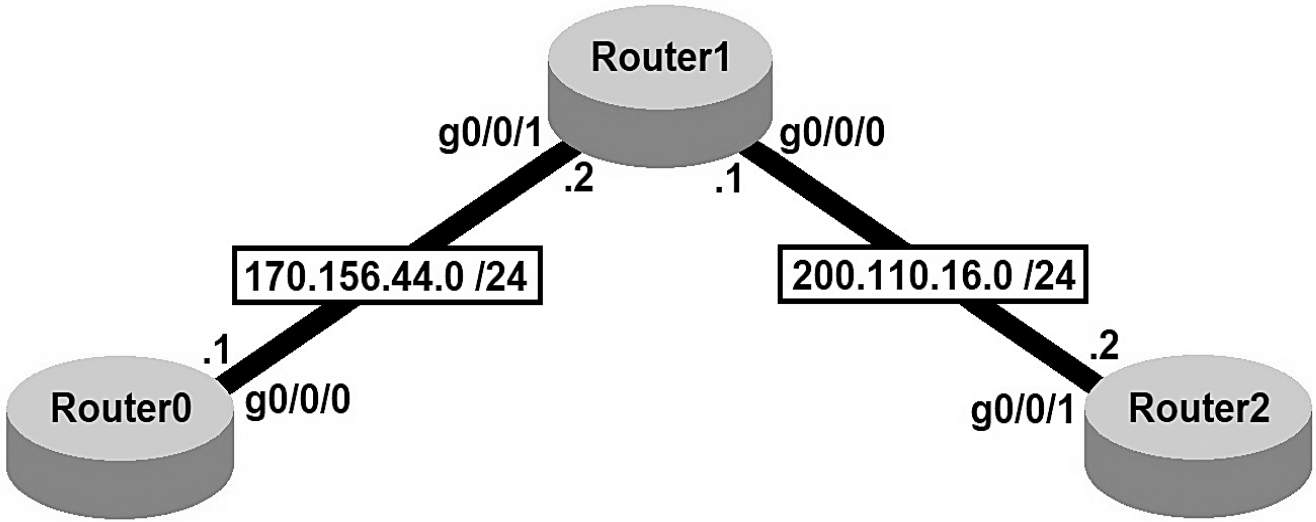
MPLS is a protocol invented for moving data quickly and efficiently between networks. The term "multi-protocol" means that MPLS is not connection dependent. One network might be using serial, another can use fiber, and one could even use T-1, and all could be connected. So, it's a true *one-to-many* solution.

MPLS also includes QoS. This is a big improvement over a standard VPN which does not include QoS due to the extra packet load due to its encryption features. So, MPLS is also better for VoIP.

The term "label" refers to how packets are handled by the ISP. With MPLS the ISP inspects the packet's ToS (or DSCP) and places an MPLS label on it and sends it on its way. This means the packet will not need to be "opened" and inspected at each network junction.

All of this information is stored in the **VRF** (Virtual Routing and Forwarding) table. The table contains Route Distinguishers that give a route a unique identifier.

MPLS has been in use for over two decades. But with the expansion of cloud computing and SaaS (Software as a Service), MPLS is slowly being replaced by SD-WAN. This is because SD-WAN eliminates backhauling of data (external and internal) to a central server for security, or other purposes.



In this example, OSPF is the routing protocol being used with **MPLS** configured on each router to enable LDP (Label Distribution Protocol), which will instruct routers to generate labels for MPLS traffic.

After configuring OSPF on all routers, this configuration is repeated on each router.

```
Router0#conf t
Router0(config)#router ospf 1
Router0(config-router)#mpls ldp autoconfig
Router0(config-router)#exit
Router0(config)#exit
Router0#
```

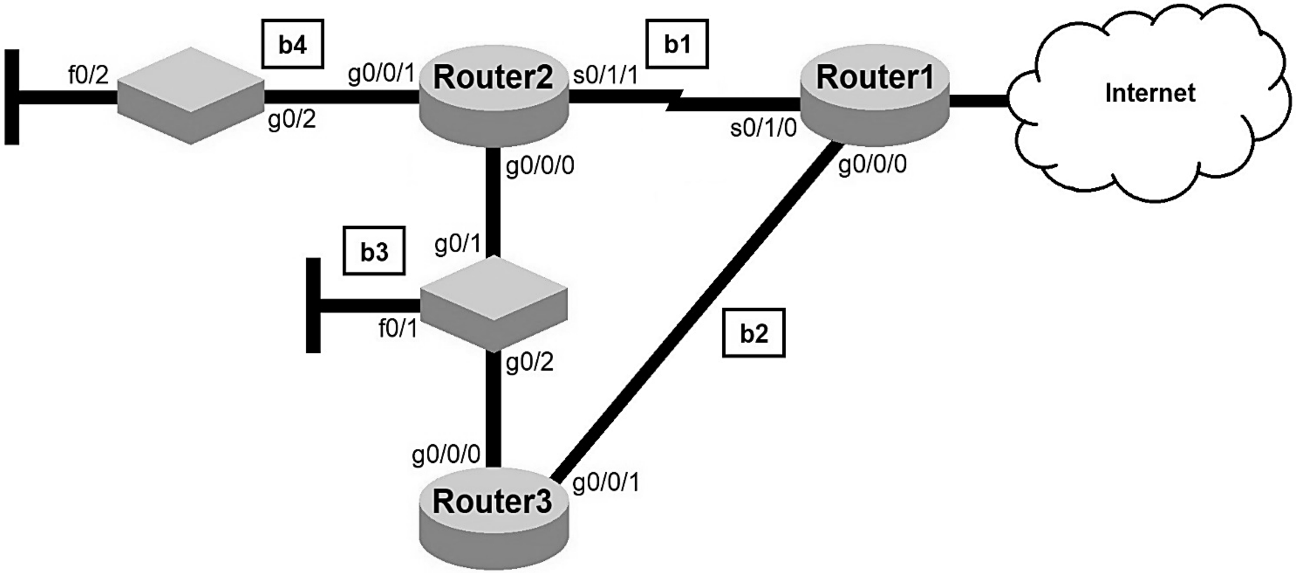
Verify configuration with the **Show MPLS Interface** command.

```
Router1#sh mpls int
Interface              IP           Tunnel   Operational
GigabitEthernet0/0/0   Yes(ldp)     No       Yes
GigabitEthernet0/0/1   Yes(ldp)     No       Yes
Router1#
```

Verify MPLS is functioning properly by running a **Traceroute** from **Router0** to **Router2**. The traceroute report will look the same as any other but with the addition of MPLS labeling.

```
Router0#traceroute 200.110.16.2
Type escape sequence to abort.
Tracing the route to 200.110.16.2

  1   170.156.44.2 [MPLS: Label 17 Exp 0] 80 msec 64 msec 40 msec
  2   200.110.16.2 42 msec 40 msec *
Router0#
```

72. IPv6: Basic setup



Note: Some manufacturers require that IPv6 be manually enabled prior to configuration.

Comparing IPv4 and IPv6

IPv4

32-bit length / decimal

Uses **ARP**: Address Resolution Protocol

Address types:

- Network (representing the whole network)
- Host (unique to each device per LAN)
- Broadcast (local one-to-all on a LAN, not routable without additional application-specific configurations)

IPv6

128-bit length / hexadecimal

Uses **NDP**: Network Discovery Protocol

Address types:

- Multicast (one-to-a-specific-group)
- Link-local (one-to-one non-routable
- Global unicast (one-to-one routable)
- Anycast (one-to-duplicate devices or services as backups)
- Unique local (private routable only)

Basic IPv6 setup: configuring **link-local** and a **global unicast** addresses.

```
Router1#conf t
Router1(config)#int s0/1/0
Router1(config-if)#ipv6 address 2001:a1:1001:b1::1/64
Router1(config-if)#ipv6 address fe80::1 link-local
Router1(config-if)#no shut
Router1(config-if)#exit
Router1(config)#exit
```

This configuration is repeated on the other two routers for networks **b2**, **b3**, and **b4**. Notice the leading zeros are removed from quartets per IPv6 allowed abbreviation rules, and that the network mask can be entered as CIDR notation. **Note:** If no address is configured and there is no DHCP available, IPv6 will generate an address automatically using a built-in application called EUI64 (Extended Unique Identifier). This protocol uses the device's MAC address to generate a unique routable address.

Neighbor Discovery Protocol

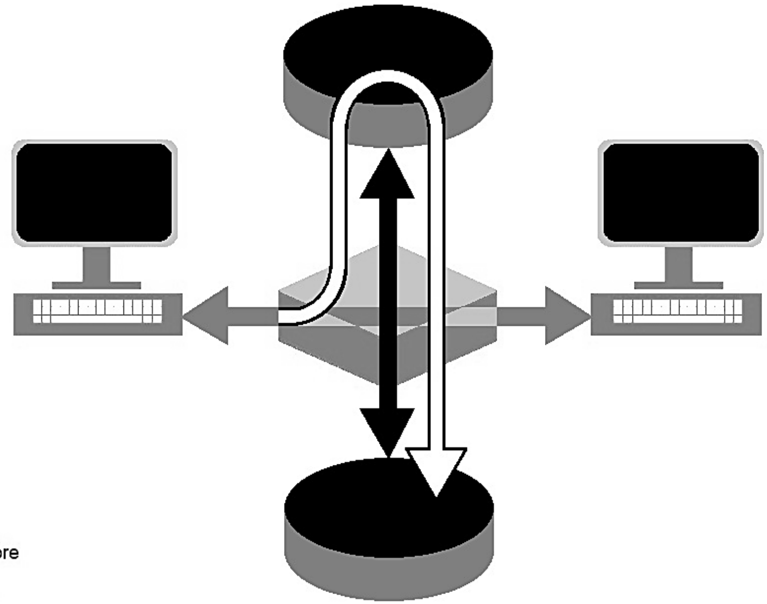
1 **Routers**
Router solicitation sent to all routers on the LAN using multicast ff02::2

2 All routers will answer with a router advertisement using multicast ff02::1

1 **Host devices**
Neighbor solicitation sent to all other hosts on the LAN using multicast ff02::1 contains source host's address information

2 All neighbor hosts will reply with a neighbor advertisement containing their address information

1 **Routers to Host devices**
Redirect is sent by routers to direct hosts to more preferrable routes (if any exist) and to redirect queries to neighbors if a host cannot find them



73. IPv6: Show IPv6 Interface Brief

This command returns a report similar to both the Show IP Interface Brief and Show Running-Config commands. However, it is specific to just interfaces configured for IPv6.

```
Router1#sh ipv6 int br
GigabitEthernet0/0/0       [up/up]
    FE80::1
    2001:A1:1001:B2::1
GigabitEthernet0/0/1       [administratively down/down]
    unassigned
Serial0/1/0                [up/up]
    FE80::1
    2001:A1:1001:B1::1
Serial0/1/1                [down/down]
    unassigned
Vlan1                      [administratively down/down]
    unassigned
Router1#
```

Verify by pinging from Router2 to Router1 on b2. Because link-local addresses <u>can</u> be the <u>same</u> on contiguous sub-networks the *source* interface for the ping must be specified.

```
Router2#ping fe80::1
Output Interface: serial0/1/1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FE80::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
Router2#
```

74. IPv6: Static Unicast route

Next, static **global unicast** routes are configured in a manner similar to IPv4 static routes. Below is the configuration for static routes from Router1 b1 and b2 to b3 via the ::2 interfaces on routers 2 and 3.

```
Router1#conf t
Router1(config)#ipv6 unicast
Router1(config)#ipv6 route 2001:a1:1001:b3::/64 2001:a1:1001:b1::2
Router1(config)#ipv6 route 2001:a1:1001:b3::/64 2001:a1:1001:b2::2
Router1(config)#exit
Router1#
```

Verify with ping and the **Show IPv6 Route** commands.

```
Router1#sh ipv6 ro
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2001:A1:1001:B1::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:A1:1001:B1::1/128 [0/0]
     via Serial0/1/0, receive
C   2001:A1:1001:B2::/64 [0/0]
     via GigabitEthernet0/0/0, directly connected
L   2001:A1:1001:B2::1/128 [0/0]
     via GigabitEthernet0/0/0, receive
S   2001:A1:1001:B3::/64 [1/0]
     via 2001:A1:1001:B1::2
     via 2001:A1:1001:B2::2
L   FF00::/8 [0/0]
     via Null0, receive
Router1#
```

75. SNMP (Simple Network Management Protocol)

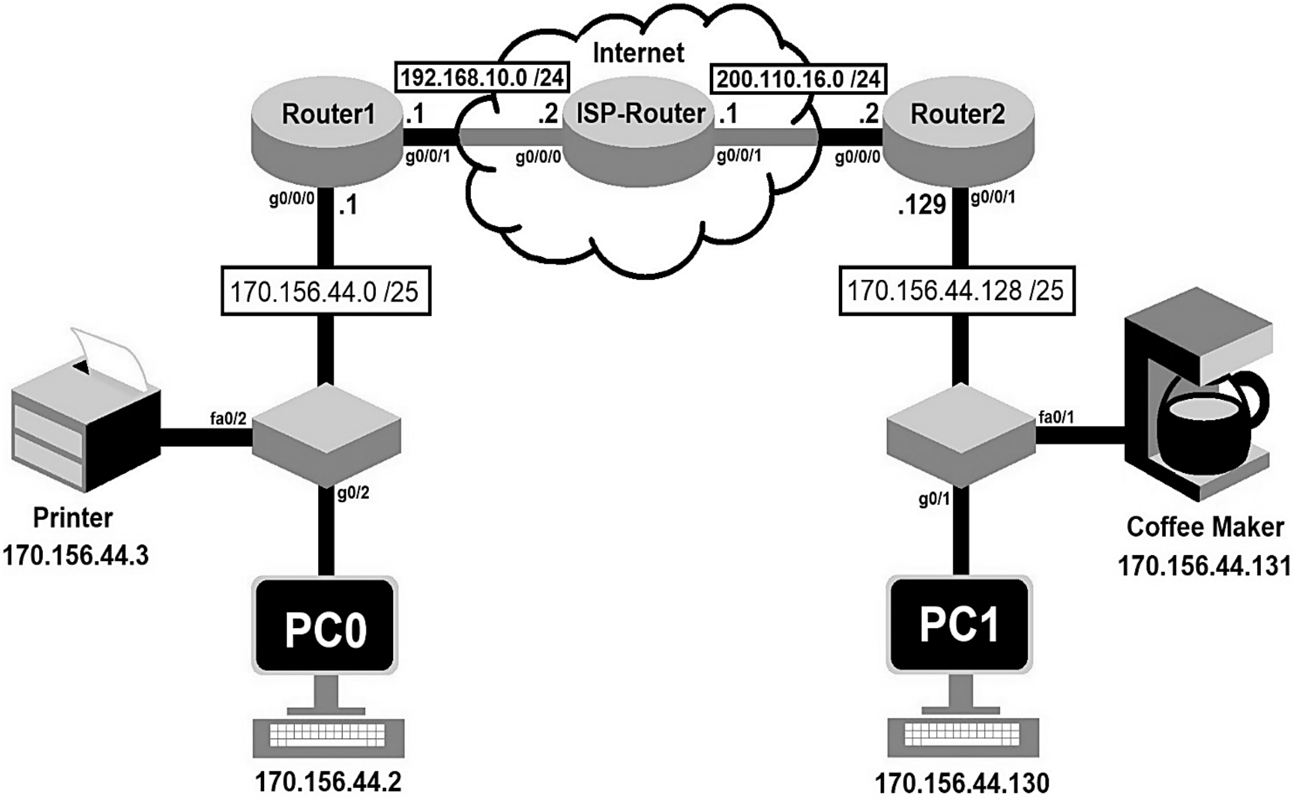
Simple Network Management Protocol allows for monitoring and managing the behavior of network devices via an **MIB** (Management Information Base). In this example it is configured on Router2 only.

```
Router2#conf t
Router2(config)#snmp-server community GUEST ro
Router2(config)#snmp-server community ADMIN rw
Router2(config)#exit
Router2#
```

> Permissions:
>
> ro = Read Only the MIB information
>
> rw = Read Write capability while in MIB

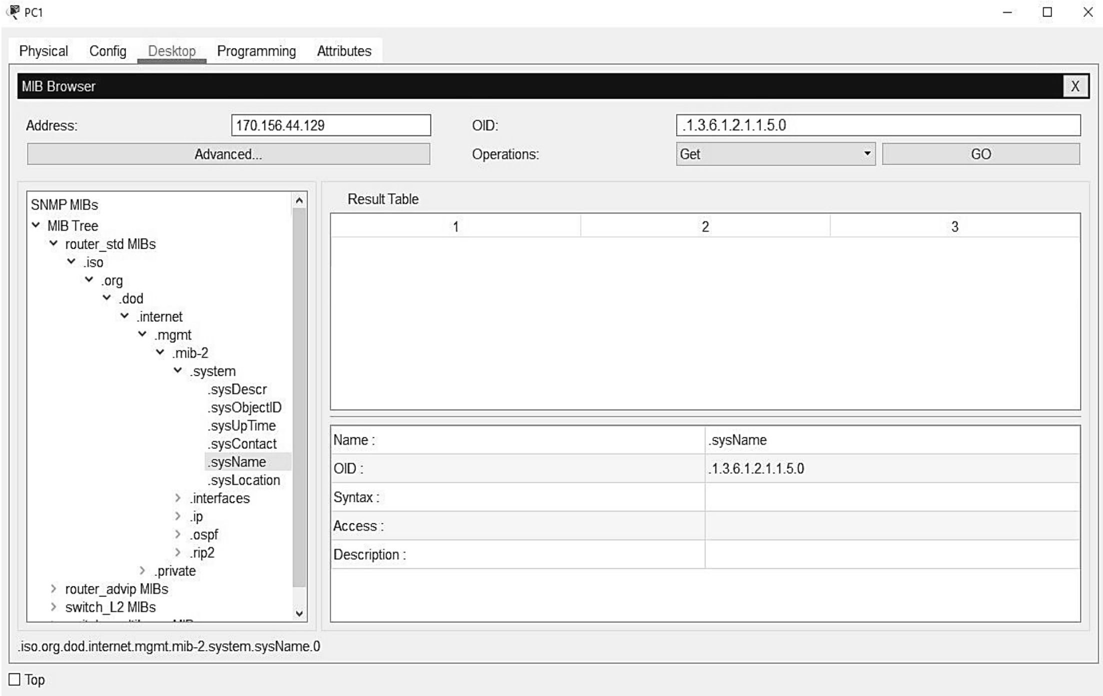
Verify by using the Show Running-Config command on Router2.

```
snmp-server community GUEST RO
snmp-server community ADMIN RW
!

Router2#
```

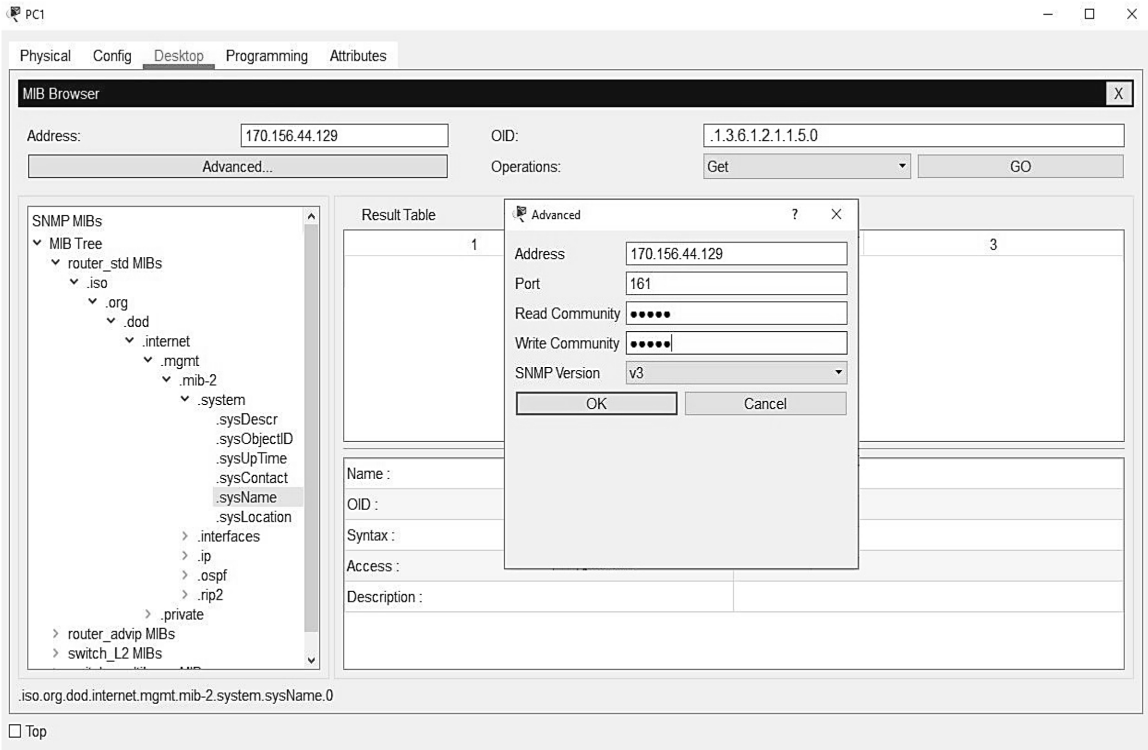


Internet
192.168.10.0 /24
200.110.16.0 /24
Router1
.1
.2
ISP-Router
.1
.2
Router2
g0/0/1
g0/0/0
g0/0/1
g0/0/0
g0/0/0
.1
.129
g0/0/1
170.156.44.0 /25
170.156.44.128 /25
fa0/2
g0/2
fa0/1
g0/1
Printer
170.156.44.3
Coffee Maker
170.156.44.131
PC0
PC1
170.156.44.2
170.156.44.130

Now we can verify SNMP Is functioning by testing it through the PC1 by using the **MIB**. Go to **Desktop** and then **MIB Browser**. Open the **MIB tree** and drill down to system. Choose sysName or enter the OID (Organizational Identifier) manually as shown.



Next, open the **Advanced** menu and enter the SNMP community names.



SNMP Versions

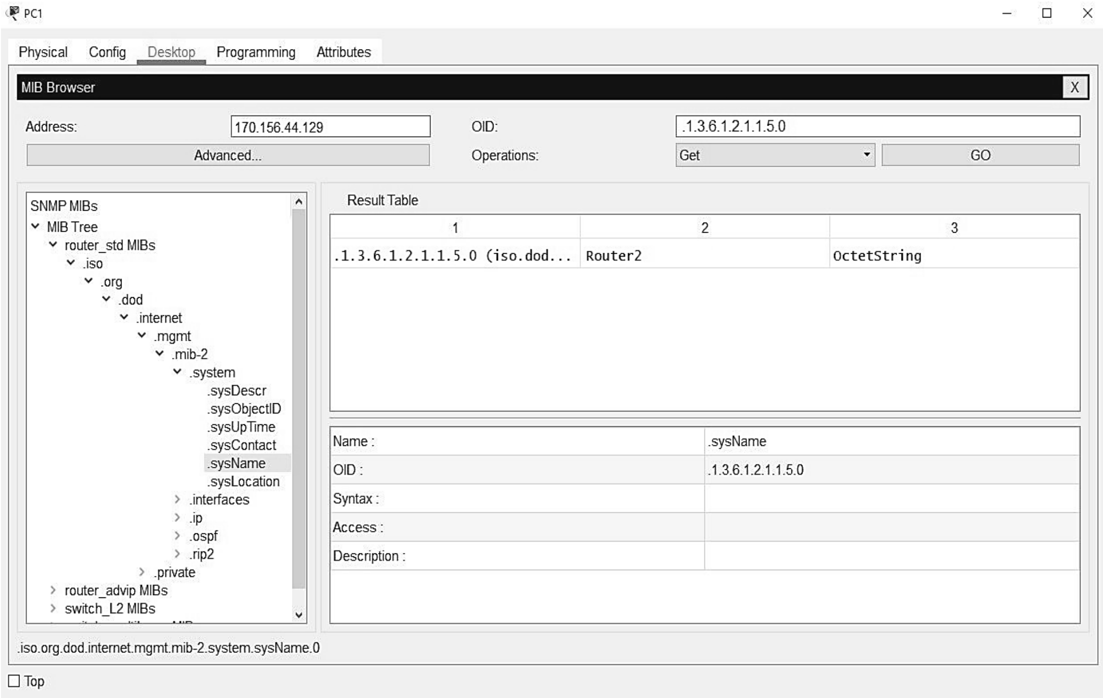
SNMPv1 = In this version the network device and the manager share a password sent in every message. However, it is sent in clear text which is not secure.

SNMPv2 = Added "Get Bulk" to the MIB search, allowing for searches of lists of OIDs. Version 2 also introduced much better security. But its implementation was overly complex, leading to version 2c.

SNMPv2c = Restored the original security.

SNMPv3 = Added a more efficient framework for better performance, along with better and easier to use security with cryptography.

Select “Get” and click “Go” to retrieve the information about Router2. **Bonus:** Configure on Router1 and then try drilling down into other areas and searching for device interfaces; or attempt to edit information.



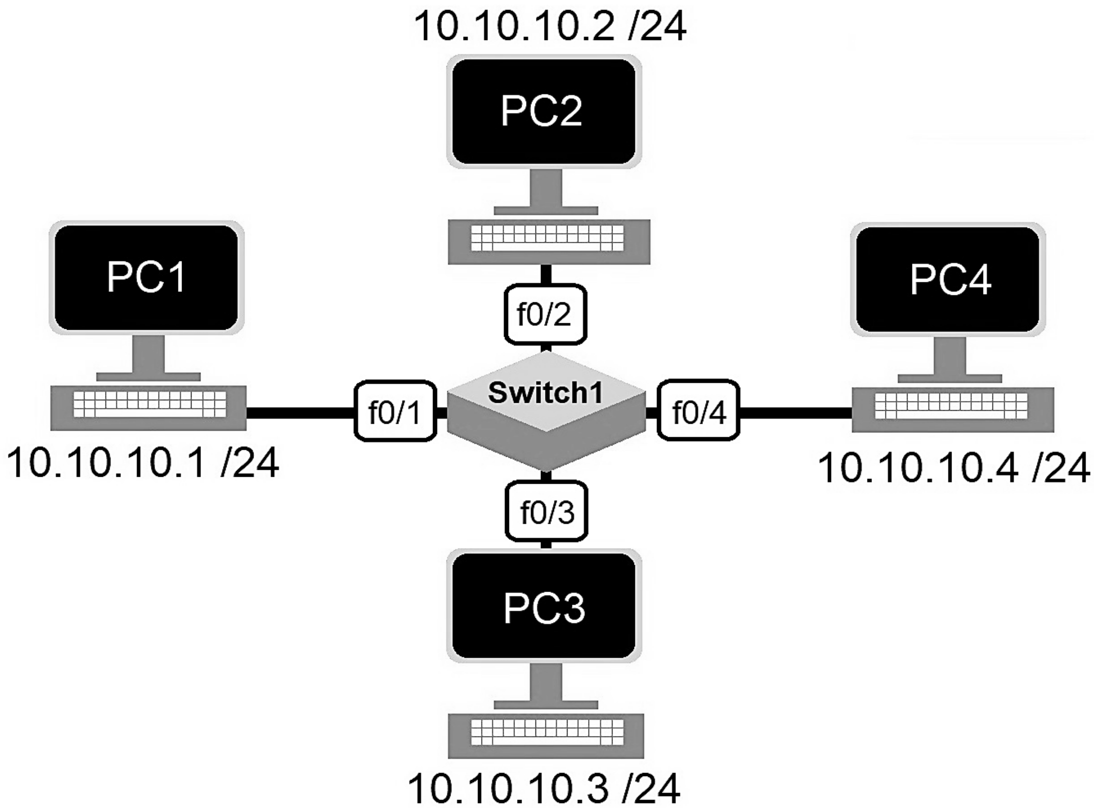
76. Local SPAN

SPAN stands for **Switched Port Analyzer** also referred to as "Local SPAN". This protocol directs a switch to duplicate traffic it receives from a source and then make a copy of it and send the copy to another device (by destination port or VLAN). This is a useful for both security and for creating backups of data sent over a network as configuring SPAN could be used to send copies of files to a NAS or save traffic for later analysis by cybersecurity.

"Both" indicates that all PC1's traffic (sent and received) will be copied and sent to PC4

```
Switch1#conf t
Switch1(config)#monitor session 1 source interface f0/1 both
Switch1(config)#monitor session 1 destination interface f0/4
Switch1(config)#exit
Switch1#
```

Note: More than one source and destination can be configured allowing for the activity of multiple devices to be monitored and multiple locations to save that log.


10.10.10.2 /24
PC2
f0/2
PC1
PC4
f0/1
Switch1
f0/4
10.10.10.1 /24
10.10.10.4 /24
f0/3
PC3
10.10.10.3 /24

Verify configuration with **Show Running-Config** and **Show Monitor** commands. Verify operation by pinging from the source to another other device. This can also be accomplished using Wire Shark or by running simulation mode in Packet-Tracer, GNS3, or any other virtual environment

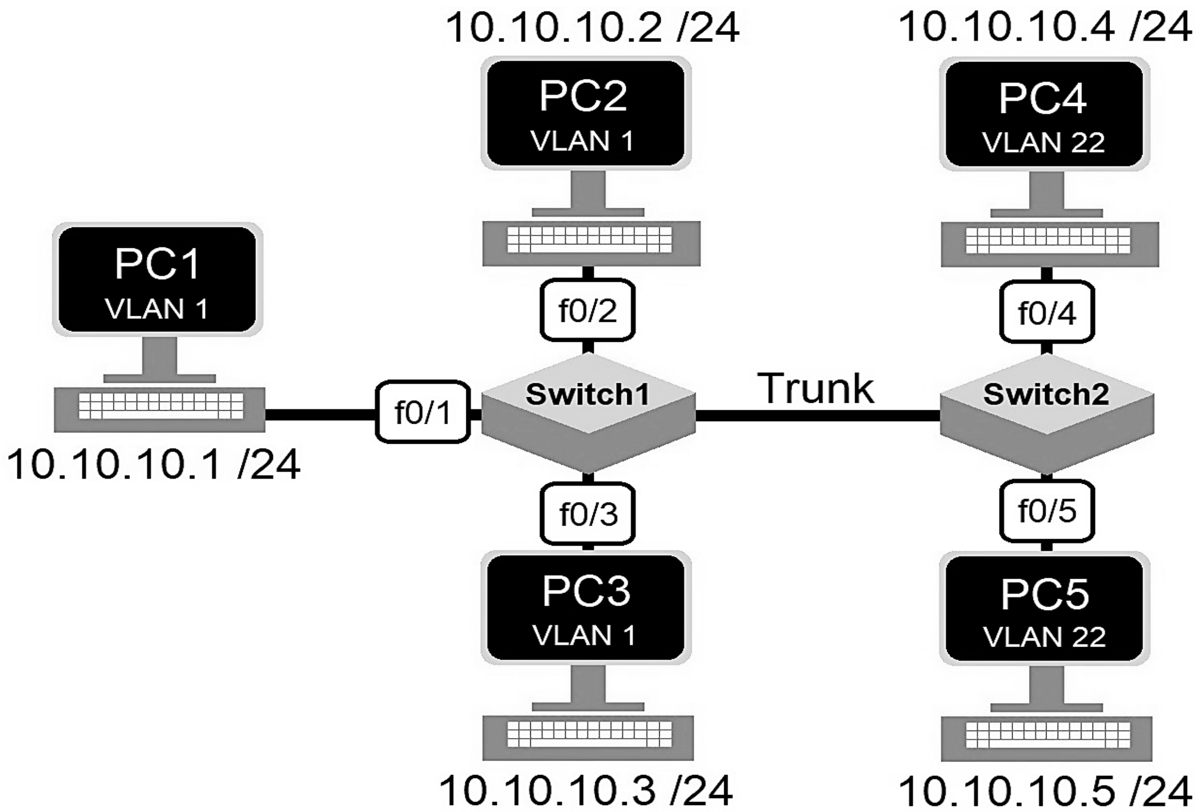
```
!
monitor session 1 source interface Fa0/1
monitor session 1 destination interface Fa0/4
!

Switch1#sh monitor
Session 1
---------
Type                    : Local Session
Description             : -
Source Ports            :
    Both                : Fa0/1
Destination Ports       : Fa0/4
    Encapsulation       : Native
          Ingress       : Disabled
```

77. Remote SPAN

Remote Span allows traffic monitoring of source ports spread out over multiple switches. The configuration is similar to Local SPAN with a few minor additions. **Note:** for example purposes, VLAN22 with the name REMOTEVLAN has already been configured on both switches, along with a trunk port between them each allowing VLAN22 and VLAN1.

```
Switch1#conf t
Switch1(config)#monitor session 1 destination remote vlan 22 reflector f0/10
Switch1(config)#monitor session 1 source interface f0/1 both
Switch1(config)#exit
```

```
Switch2#conf t
Switch2(config)#monitor session 1 source remote vlan 22
Switch2(config)#monitor session 1 destination interface f0/4
Switch2(config)#exit
```


10.10.10.2 /24
PC2
VLAN 1
10.10.10.4 /24
PC4
VLAN 22
PC1
VLAN 1
f0/2
f0/4
f0/1
Switch1
Trunk
Switch2
10.10.10.1 /24
f0/3
f0/5
PC3
VLAN 1
PC5
VLAN 22
10.10.10.3 /24
10.10.10.5 /24

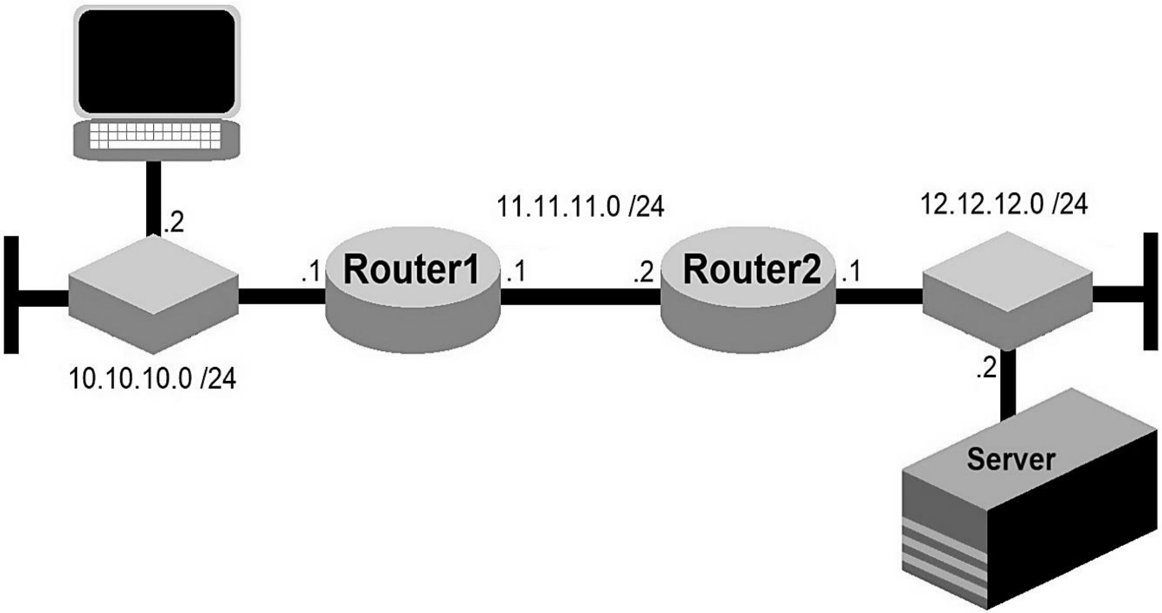
Verify configuration with **Show Running-Config** and **Show Monitor** commands. Verify **RSPAN** operation by pinging from the source to another other device. This can also be accomplished using Wire Shark or by running simulation mode in Packet-Tracer, GNS3, or any other virtual environment

```
!
monitor session 1 source interface Fa0/1
monitor session 1 destination remote vlan 22 reflector-port Fa0/10
!
Switch1#
```

```
Switch2#sh monitor
Session 1
---------
Type                   : Remote Source Session
Description            : -
Source RSPAN VLAN      : 22
Destination Ports      : Fa0/4
    Encapsulation      : Native
          Ingress      : Disabled
```

78. QoS (Quality of Service) part 1 of 2: Matching values with NBAR

QoS is used to affect the flow of different types of data traffic across our networks. The typical way to do this is to prioritize according to data type and then divide the available bandwidth between them. In part 1 of this example **Router1** will use **NBAR** to prioritize data traffic. In part 2 **Router2** will use **DSCP**.



Comparing NBAR to DSCP

Network-Based Application Recognition is a classification engine that recognizes and classifies a wide variety of protocols and applications, including web-based and other applications and protocols that use dynamic TCP/UDP port assignments.

In a way, it functions like a more granular type of ACL, in that it examines packets at each interface to apply the correct policy.

Differentiated Source Code Point is the successor to ToS (Type of Service) and applies QoS policies like NBAR.

The primary difference between DSCP and NBAR however, is that DSCP marks traffic so that it can be identified and prioritized at each hop without requiring a deep packet inspection. This makes DSCP cost less in processor cycles.

```
Router1#conf t
Router1(config)#class-map VOIP
Router1(config-cmap)#match protocol rtp
Router1(config-cmap)#exit
Router1(config)#class-map HTTP
Router1(config-cmap)#match protocol http
Router1(config-cmap)#exit
Router1(config)#class-map ICMP
Router1(config-cmap)#match protocol icmp
Router1(config-cmap)#exit
Router1(config)#policy POLICY1
Router1(config-pmap)#class VOIP
Router1(config-pmap-c)#set ip dscp ef
Router1(config-pmap-c)#priority 100
```

On Router1

QoS via NBAR paramters by class-map name and matching protocol:

1. VOIP = phone traffic (rtp)
2. HTTP = internet access (http)
3. ICMP = network maintenance (icmp)

Combine into one policy:

- POLICY1

Set VOIP (rtp) as the *priority* traffic type.

Set bandwidth for each class of traffic:

- VOIP/rtp = 100
- HTTP/http = 50
- ICMP/icmp = 50

```
Router1(config-pmap)#class HTTP
Router1(config-pmap-c)#set ip dscp af31
Router1(config-pmap-c)#bandwidth 50
Router1(config-pmap)#class ICMP
Router1(config-pmap-c)#set ip dscp af11
Router1(config-pmap-c)#bandwidth 25
Router1(config-pmap)#exit
Router1(config-pmap)#exit
```

Class-maps and policy configurations are completed.

Verify with the **Show Running-Config** command.

```
!
class-map match-all VOIP
 match protocol rtp
class-map match-all HTTP
 match protocol http
class-map match-all ICMP
 match protocol icmp
!
policy-map POLICY1
 class VOIP
  priority 100
  set ip dscp ef
 class HTTP
  bandwidth 50
  set ip dscp af31
 class ICMP
  bandwidth 25
  set ip dscp af11
```

Lastly, the policy containing the class-maps must be bound to an interface.

`Router1(config)#int g0/0` ← the 11.11.11.**1** interface

`Router1(config-if)#service-policy` **`output`** `POLICY1`

`Router1(config-if)#exit`

`Router1(config)#exit`

`Router1#wr`

Router1 configurations are complete. Verify with the **Show Running-Config** command.

```
!
interface GigabitEthernet0/0
 ip address 11.11.11.1 255.255.255.0
 service-policy output POLICY1
 duplex auto
 speed auto
```

79. QoS (Quality of Service) part 2 of 2: Matching values with DSCP

```
Router2#conf t
Router2(config)#class-map VOIP
Router2(config-cmap)#match ip DSCP ef
Router2(config-cmap)#exit
Router2(config)#class-map HTTP
Router2(config-cmap)#match ip dscp af31
Router2(config-cmap)#exit
Router2(config)#class-map ICMP
Router2(config-cmap)#match ip dscp af11
Router2(config-cmap)#exit
```

On Router2

QoS via **DSCP** parameters by class-map name and matching ip protocol:

1. VOIP = phone traffic (ef)
2. HTTP = internet access (af31)
3. ICMP = network maintenance (af11)

Combine into one policy for remarking:

- REMARK1

Set VOIP (ef) as taking precedence over the other traffic types.

Set precedence for each class of traffic:

- VOIP = 5
- HTTP = 3
- ICMP = 0

A policy must now be configured for remarking according to the traffic type.

```
Router2(config)#policy REMARK1
Router2(config-pmap)#class VOIP
Router2(config-pmap-c)#set precedence 5
Router2(config-pmap-c)#exit
Router2(config-pmap)#class HTTP
Router2(config-pmap-c)#set precedence 3
Router2(config-pmap)#class ICMP
Router2(config-pmap-c)#set precedence 0
```

```
!
class-map match-all VOIP
 match ip dscp ef
class-map match-all HTTP
 match ip dscp af31
class-map match-all ICMP
 match ip dscp af11
!
policy-map REMARK1
 class VOIP
  set precedence 5
 class HTTP
  set precedence 3
 class ICMP
  set precedence 0
```

Precedence basics:

5 = Critical. Voice, specifically real-time telephony/IP phones.

3 = Audio and video. Basic online activities.

0 = Best effort, routine network maintenance activities.

Finally, the policy must be bound to an interface.

```
Router2(config)#int g0/1
```
← the 11.11.11.**2** interface

```
Router2(config-if)#service-policy input REMARK1
Router2(config-if)#exit
Router2(config)#exit
Router2#wr
```

Class-maps and policy configurations are completed. Verify with the **Show Running-Config** command.

```
!
interface GigabitEthernet0/1
 ip address 11.11.11.2 255.255.255.0
 service-policy input REMARK1
 duplex auto
 speed auto
```

80. QoS: DSCP (Differentiated Source Code Point) Values

Category	Code	Example
VoIP	EF	Cisco IP phones (G.711, G.729)
Broadcast Video	CS5	Cisco IP Surveillance / Enterprise TV
Real-time Interactive	CS4	Cisco TelePresence
Multi-media Conferencing	AF4	Cisco Jabber, WebEx
Multi-media Streaming	AF3	Cisco Digital Media (VoDs)
Network Control	CS6	EIGRP, OSPF, BGP, HSRP, etc.
Signaling	CS3	SCCP, SIP, H.323
Ops/Admin/Mgmt	CS2	SNMP, SSH, Syslog
Transactional Data	AF2	ERP, CRM, database apps
Bulk Data	AF1	E-mail, FTP, back-up apps
Best Effort	DF	Default class
Scavenger	CS1	YouTube, Netflix, online gaming

81. Show Policy-Map

The **Show Policy-Map** *by name* command returns the following information.

```
Router1#sh policy-map POLICY1
  Policy Map POLICY1
    Class VOIP
      Strict Priority
      Bandwidth 100 (kbps) Burst 2500 (Bytes)
      set ip dscp ef
    Class HTTP
      Bandwidth 50 (kbps) Max Threshold 64 (packets)
      set ip dscp af31
    Class ICMP
      Bandwidth 25 (kbps) Max Threshold 64 (packets)
      set ip dscp af11
Router1#
```

Show Policy-Map *by interface type* on **Router2** (shown here abridged) returns detailed information about all class-maps contained in the policy bound to that interface. **Notice here**, that **no** packets have been marked. Let's test the configuration (next page).

```
    Class-map: ICMP (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: ip dscp af11 (10)
      QoS Set
        precedence 0
          Packets marked 0

    Class-map: class-default (match-any)
      368 packets, 28944 bytes
      5 minute offered rate 176 bps, drop rate 0 bps
      Match: any


Router2#
```

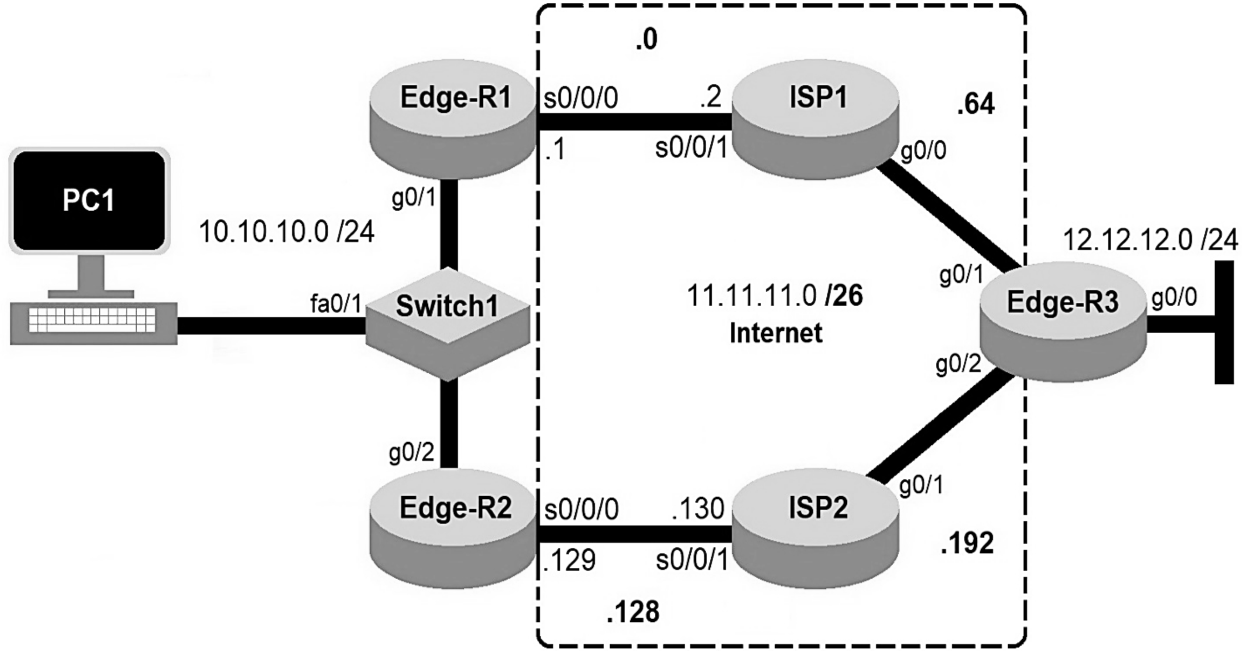
Note: Traffic <u>not</u> assigned a QoS policy (when QoS is configured on a device) is automatically classified as **class-default (match any)** as seen in the report.

A ping is sent from **PC1** to the **server** on the **12.12.12.0** network to test the configuration. Issuing the Show Policy-Map command now, verifies the configuration is correct and is functioning.

```
    Class-map: ICMP (match-all)
      3 packets, 384 bytes
      5 minute offered rate 12 bps, drop rate 0 bps
      Match: ip dscp af11 (10)
      QoS Set
        precedence 0
          Packets marked 3

    Class-map: class-default (match-any)
      461 packets, 36340 bytes
      5 minute offered rate 190 bps, drop rate 0 bps
      Match: any


Router2#
```

82. PPP (Point-to-Point Protocol)

PPP can be used on many VPNs. In this example a PPP route will be configured from PC1's network to the ISP routers.



Edge-R1

Note: The same configuration is issued on **ISP1** serial interface s0/0/1.

```
Edge-R1#conf t
Edge-R1(config)#int s0/0/0
Edge-R1(config-if)#encapsulation PPP
Edge-R1(config-if)exit
Edge-R1(config)#exit
Edge-R1#wr
```

Verify with **Show Running-Config**

```
!
interface Serial0/0/0
 ip address 11.11.11.1 255.255.255.192
 encapsulation ppp

Edge-R1#
```

Verify further with **Show Interface**. Notice that LCP (Link Control Protocol) is now open, allowing other network control protocols IPCP (IPv4) and CDPCP (Cisco Discovery Protocol) to be used over the PPP link.

```
Edge-R1#sh int s0/0/0
Serial0/0/0 is up, line protocol is up (connected)
  Hardware is HD64570
  Internet address is 11.11.11.1/26
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, loopback not set, keepalive set (10 sec)
  LCP Open
  Open: IPCP, CDPCP
```

```
ISP1#sh int s0/0/1
Serial0/0/1 is up, line protocol is up (connected)
  Hardware is HD64570
  Internet address is 11.11.11.2/26
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, loopback not set, keepalive set (10 sec)
  LCP Open
  Open: IPCP, CDPCP
```

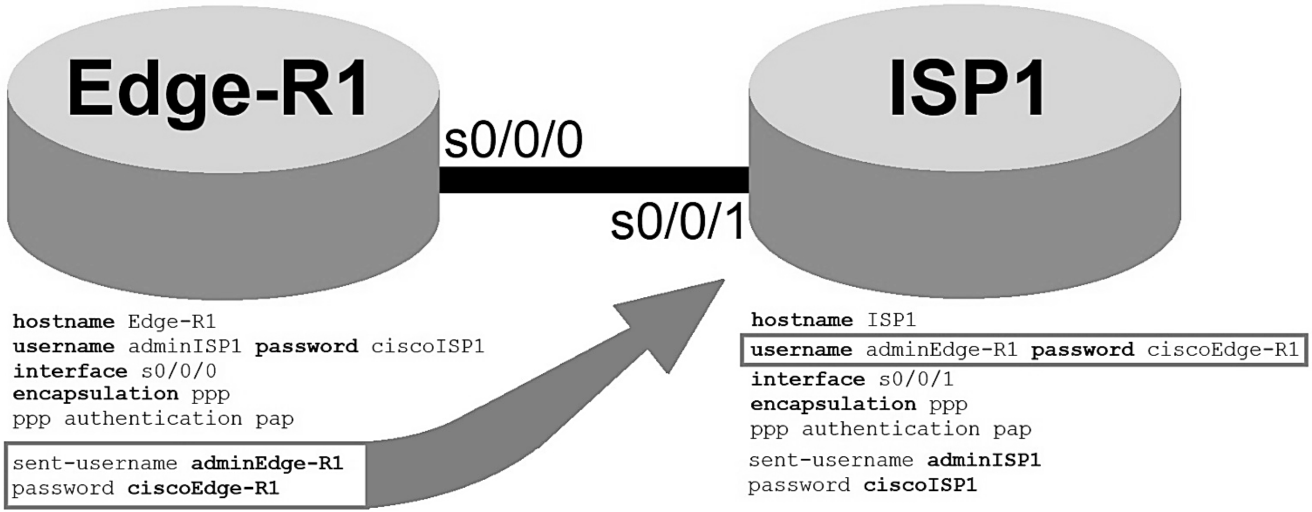
83. PPP with PAP (Password Authentication Protocol)

When configuring either **PAP** or **CHAP** (or when using any other secured point-to-point link protocol) devices must authenticate themselves to each other before a connection is established.

```
Edge-R1#conf t
Edge-R1(config)#username adminISP1 password ciscoISP1
Edge-R1(config)#int s0/0/0
Edge-R1(config-if)#encapsulation ppp
Edge-R1(config-if)#ppp pap sent-username adminEdge-R1 password ciscoEdge-R1
Edge-R1(config-if)#exit
Edge-R1(config)#exit
Edge-R1#wr
```

Configure **ISP1** the same as Edge-R1, but with the username and password stored on Edge-R1 as ISP1's sent-username and password.

When using PAP a **username** and **password** <u>must</u> first be configured locally on the connected devices. The requestor sends their PAP username and password to the responder. The responder checks their locally stored username and password to check that they match. In the example shown here, Edge-R1 **sends** its PAP authentication to **ISP1** which checks to see if the credentials match before startning a PPP session.



Show Running-Config verifies proper configuration. The locally stored username and password is sent by the Edge-R1 router to identify (authenticate) itself to ISP1 when opening a PPP PAP connection. The ISP1 Router compares that username and password to the credentials it has stored in memory. If it matches, it sends a reply to open a PPP PAP connection with the Edge-R1 router.

```
username adminEdge-R1 password 0 ciscoEdge-R1
!
interface Serial0/0/1
 ip address 11.11.11.2 255.255.255.192
 encapsulation ppp
 ppp pap sent-username adminISP1 password 0 ciscoISP1
 clock rate 2000000
!

ISP1#
```

Further verification of connectivity can be accomplished by pings and traceroutes.

84. PPP with CHAP (Challenge Handshake Authentication Protocol)

CHAP works similar to PAP except that the password is the same on both devices.

On Edge-R2

```
Edge-R2#conf t
Edge-R2(config)#username ISP2 password cisco123
Edge-R2(config)#exit
Edge-R2#conf t
Edge-R2(config)#int s0/1/0
Edge-R2(config-if)#encapsulation PPP
Edge-R2(config-if)#ppp authentication chap
Edge-R2(config-if)exit
Edge-R2#wr
```

Configure the same on **ISP2**. But with "Edge-R2" but with the same password of cisco123.

Verifying CHAP on both ends with the **Show Running-Config** command.

```
username ISP2 password 0 cisco123
!
interface Serial0/1/0
 ip address 11.11.11.129 255.255.255.192
 encapsulation ppp
 ppp authentication chap
 clock rate 2000000
!
Edge-R2#
```

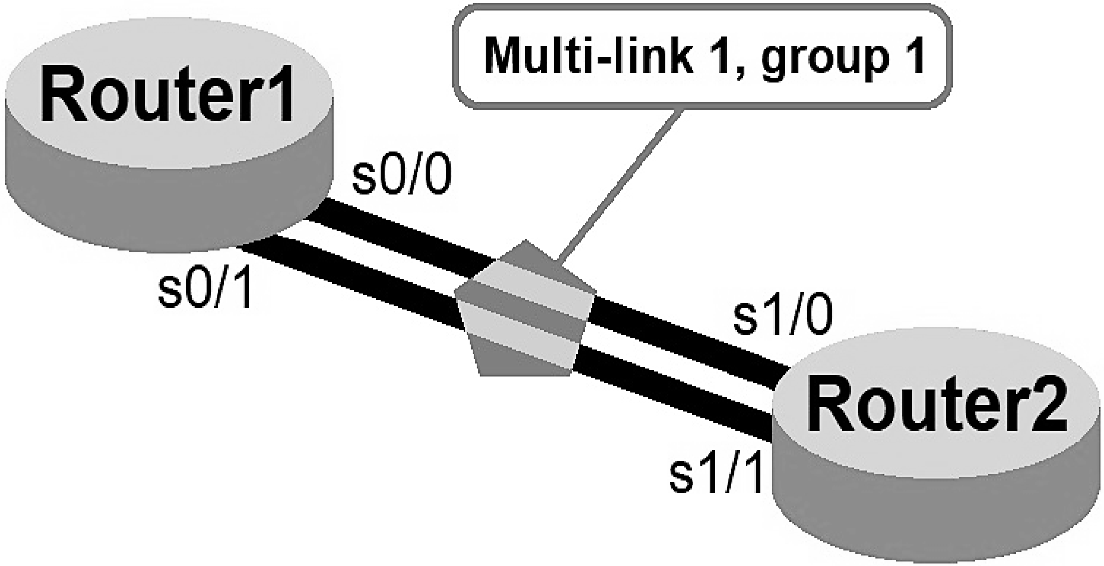
Further verification of connectivity can be accomplished by pings and traceroutes.

Comparing PAP and CHAP

Password Authentication Protocol	Challenge Handshake Authentication Protocol
• Used in PPP connections. • Uses a matching username and password. • Authenticates one time per session. • Not as secure as CHAP	• Used in PPP connections. • Uses a matching password. • Authenticates periodically to ensure only allowed users can participate. • More secure than PAP

85. PPP (Point-to-Point Protocol) Multi-Link

Multi-Link connects two devices via multiple PPP links. **Note:** PAP and CHAP are optional as usual.



Configuring a multilink connection between Router1 and Router2.

```
Router1#conf t
Router1(config)#interface multilink 1
Router1(config-if)#encapsulation ppp
Router1(config-if)#ppp multilink
Router1(config-if)#ip address 14.14.14.1 255.255.255.0
Router1(config-if)#ppp multilink group 1
Router1(config-if)#exit
Router1(config)#int s0/0
Router1(config-if)#encapsulation ppp
Router1(config-if)#ppp multilink group 1
Router1(config-if)#ppp authentication chap
Router1(config-if)#exit
Router1(config)#exit
Router1#
```

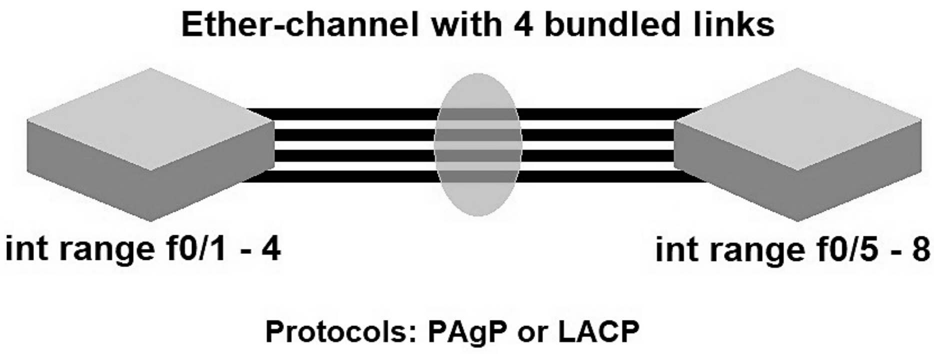
Note: no IP addresses for the physical interfaces. Instead, an IP address is assigned to the multilink interface.

Each interface must be configured separately and at both ends as usual.

Additions such as PAP or CHAP are configured in the normal way per interface.

86. Ether-Channel

Ether-channel configures multiple ethernet cable links to function as a *single link* between devices to move a greater volume of traffic by spreading the load over multiple links. Ether-channel also provides redundancy as a link that fails does not affect the other links in the group.



Up to 8 links can be bundled per ether-channel. But only a maximum of 4 can operate simultaneously. Additional links are placed in standby mode in case one of the other links goes down. Ether-channel also uses one of two protocols: **PAgP** (Port Aggregation Protocol) or **LACP** (Link Aggregation Control Protocol).

Configured on **Switch1** with the option of configuring it as a trunk.

```
Switch1#conf t
Switch1(config)#int f0/1-4
Switch1(config-if-range)#speed auto
Switch1(config-if-range)#duplex auto
Switch1(config-if-range)#mdix auto
Switch1(config-if-range)#channel-group 1 mode LACP active
Switch1(config-if-range)#switchport mode trunk
Switch1(config-if-range)#switchport trunk allowed vlan 1
Switch1(config-if-range)#exit
Switch1(config-if)#exit
Switch1(config)#exit
```

Optional: Mdix auto instructs switches to reverse the transmit-receive circuit when straight-through cables are being used instead of (recommended) cross-over cables.

Note: If prompted, choose option **dot1q** encapsulation, as auto results in the older ISL protocol being used. Remember to include allowed VLANs on the trunk. The default VLAN1 is used here as an example.

Comparing PAgP and LACP

PAgP	LACP
Port Aggregation Protocol • Cisco proprietary. • Bundles 8 links maximum. • Maximum of 4 links operate simultaneously. • Additional links place in standby mode to function as backups. • Config commands: On*, Auto, Desirable.	*Link Aggregation Control Protocol* • Vendor neutral. • Bundles 16 links maximum. • Maximum of 4 links operate simultaneously. • Additional links placed in standby mode to function as backups. • Config commands: On*, Active, Passive.

***Note:** The **"on"** command can also be used with wireless controllers.

A similar configuration is repeated on **Switch2**. LACP is the protocol being used. Active or Passive mode may be selected because Switch1 is already configured for Active.

```
Switch2#conf t
Switch2(config)#int f0/5-8
Switch2(config-if-range)#speed auto
Switch2(config-if-range)#duplex auto
Switch2(config-if-range)#channel-group 1 mode LACP passive
Switch2(config-if-range)#switchport mode trunk
Switch2(config-if-range)#switchport trunk allowed vlan 1
Switch2(config-if-range)#exit
Switch2(config-if)#exit
Switch2(config)#exit
```

Speed and Duplex must match on each end of all participating links.

When using LACP, active and passive states are used.

Verify ether-channel configuration with the **Show Running-Config** command.

```
interface Port-channel1
 switchport mode trunk
!
interface FastEthernet0/1
 switchport trunk allowed vlan 1
 switchport mode trunk
 channel-group 1 mode active
!
interface FastEthernet0/2
 switchport trunk allowed vlan 1
 switchport mode trunk
 channel-group 1 mode active
!
interface FastEthernet0/3
 switchport trunk allowed vlan 1
 switchport mode trunk
 channel-group 1 mode active
!
interface FastEthernet0/4
 switchport trunk allowed vlan 1
 switchport mode trunk
 channel-group 1 mode active
!
Switch1#
```

```
interface Port-channel1
 switchport mode trunk
!
interface FastEthernet0/5
 switchport trunk allowed vlan 1
 switchport mode trunk
 channel-group 1 mode passive
!
interface FastEthernet0/6
 switchport trunk allowed vlan 1
 switchport mode trunk
 channel-group 1 mode passive
!
interface FastEthernet0/7
 switchport trunk allowed vlan 1
 switchport mode trunk
 channel-group 1 mode passive
!
interface FastEthernet0/8
 switchport trunk allowed vlan 1
 switchport mode trunk
 channel-group 1 mode passive
!
Switch2#
```

87. Ether-channel: Show Etherchannel Port-Channel

```
Switch1#sh etherchannel port-channel
                Channel-group listing:
                ----------------------

Group: 1
----------      Port-channels in the group:
                ---------------------------

Port-channel: Po1    (Primary Aggregator)
------------

Age of the Port-channel   = 00d:00h:50m:58s
Logical slot/port   = 2/1       Number of ports = 4
GC                  = 0x00000000      HotStandBy port = null
Port state          = Port-channel
Protocol            =   LACP
Port Security       = Disabled

Ports in the Port-channel:

Index   Load   Port     EC state        No of bits
------+------+------+------------------+-----------
  0     00     Fa0/1    Active             0
  0     00     Fa0/2    Active             0
  0     00     Fa0/3    Active             0
  0     00     Fa0/4    Active             0
Time since last port bundled:    00d:00h:42m:31s    Fa0/4
```

88. Ether-channel: Show Etherchannel Summary

Code "P" indicates that link is active and functioning normally.

```
Switch1#sh etherchannel summary
Flags:  D - down        P - in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------

1      Po1(SU)           LACP   Fa0/1(P) Fa0/2(P) Fa0/3(P) Fa0/4(P)
Switch1#
```

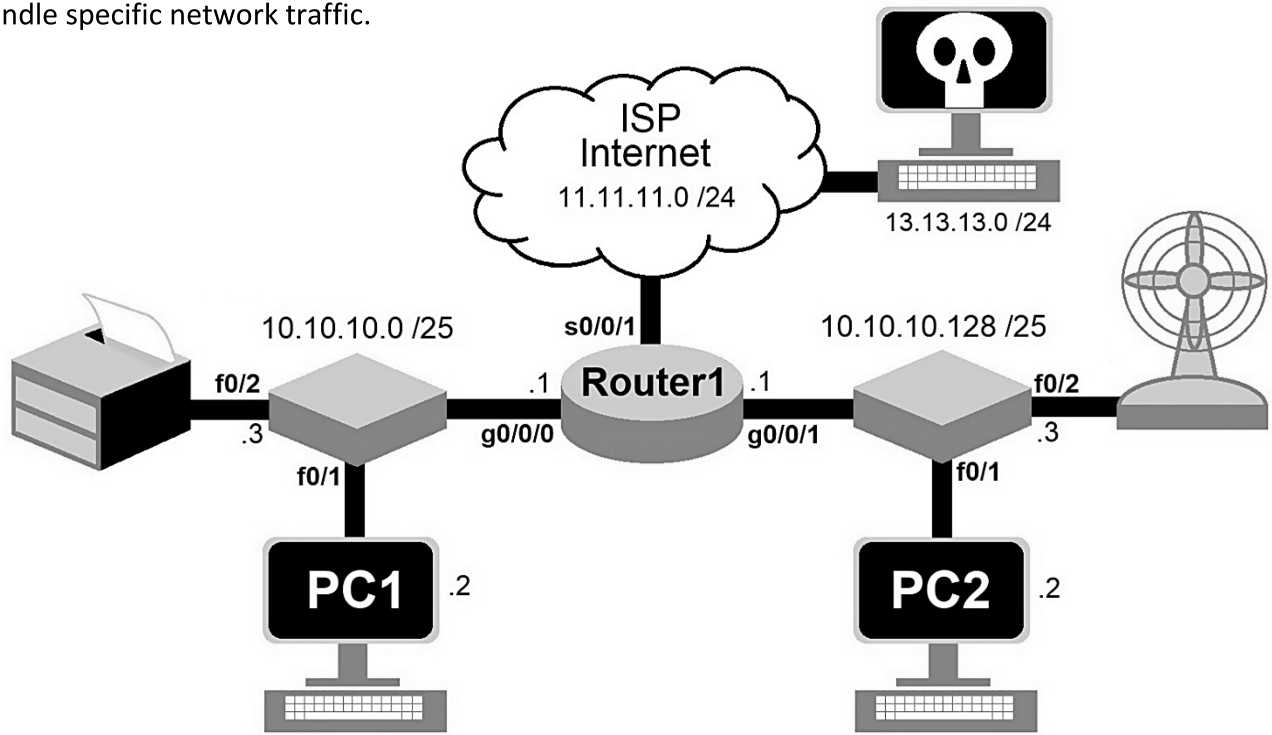
89. Ether-channel: Show Etherchannel and **Show Etherchannel Load-Balancing**

Note: the **L2 group state** indicates this is a layer-2 ether-channel. Switches that also route have a group state of L3 indicating both layer-2 and layer-3 capability. **Load balancing** is (by default) by source-MAC. Meaning, that each sender theoretically has their own private link *while* they are using it.

```
Switch1#sh etherchannel
                Channel-group listing:
                ----------------------
Group: 1
----------
Group state = L2
Ports: 4 Maxports = 16
Port-channels: 1 Max Port-channels = 16
Protocol:   LACP
Switch1#


Switch1#sh etherchannel load-balance
EtherChannel Load-Balancing Operational State (src-mac):
Non-IP: Source MAC address
  IPv4: Source MAC address
  IPv6: Source MAC address

Switch1#
```

90. ACL (Access List) Standard

Access lists can be used to allow or deny access through an interface, or even to give directions on how to handle specific network traffic.



The most basic ACL is the **Standard ACL** which can be numbered from **1 thru 99**. In this example a potentially dangerous website should not be accessible to, or have access to, hosts on the network.

```
Router1#conf t
Router1(config)#access-list 1 deny 13.13.13.0 0.0.0.0
Router1(config)#access-list 1 permit any
Router1(config)#int s0/0/1
Router1(config-if)#ip access-group 1 in
Router1(config-if)#exit
Router1(config)#exit
Router1#
```

The result of this simple ACL is that host devices can still *try* to connect to 13.13.13.0, but that website will not be able to respond to those requests because the access list will deny all packets coming from that source. Meanwhile, the “permit any” command at the bottom of the ACL makes certain that hosts on both 10.10.10.0 LANs can still access each other, the internet, and other networks.

Verify with **Show Running-Config.**

```
!
interface Serial0/0/1
 ip address 11.11.11.1 255.255.255.0
 ip access-group 1 in
 clock rate 2000000
!
access-list 1 deny host 13.13.13.0
access-list 1 permit any
!
Router1#
```

Note: the term “host” is a default that appears on some devices. However, all packets from all hosts on the entire 13.13.13.0 network are denied access to the 10.10.10.0 network via interface s0/0/1.

Comparing Standard ACLs to Extended ACLs

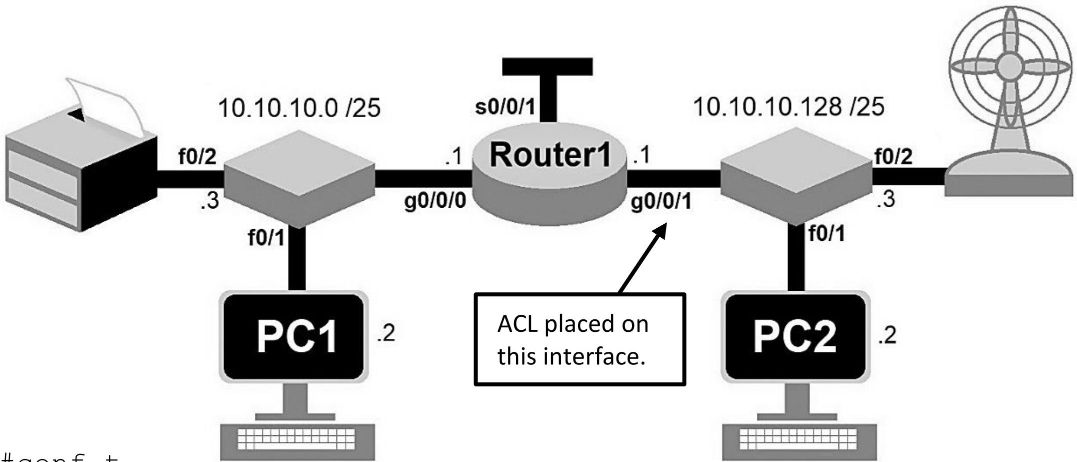
Standard ACL	Extended ACL
• Inspects by source address only. • Number range: 1 - 99. • Permit or deny.	• Inspects by <u>both</u> source and destination address. • Number range: 100 - 199. • Can be named. • Can inspect by port number, port range, and port exclusion range. • Can inspect by protocol type and number. • Permit or deny.

91. ACL Best Practices

- Devices read ACL statements in order the first statement to the last (literally from top to bottom). So, it is important to configure statements in the correct order.
- ACLs should be placed as close to the target of their deny or permit statements as possible. This reduces the amount of unnecessary network traffic.
- All ACLs have a default or implicit “deny any” as their last statement. This deny statement is invisible and cannot be deleted. Thus, a “permit any” should be included as the last entered statement on all ACLs to ensure that other networks and hosts remain available.

92. ACL (Access List) Extended

Extended ACLs have more options than standard ACLs. In the following example an extended ACL is configured to secure access to the network printer. Hosts on the 10.10.10.0 LAN can use it. But hosts on the 10.10.10.128 LAN are not permitted to use it. All other communication between both LANs must still be permitted however.




```
Router1#conf t
Router1(config)#access-list 110 deny tcp 10.10.10.128 0.0.0.0 eq 80 10.10.10.3
0.0.0.0
Router1(config)#access-list 110 permit any
Router1(config)#int g0/0/1
Router1(config)#ip access-group 110 in
Router1(config)#exit
```

Verify proper configuration with the **Show Running-Config** command.

```
!
interface GigabitEthernet0/0/1
 ip address 10.10.10.129 255.255.255.128
 ip access-group 110 in
 duplex auto
 speed auto
!
interface Vlan1
 no ip address
 shutdown
!
ip classless
!
ip flow-export version 9
!
!
access-list 110 deny tcp host 10.10.10.128 eq www host 10.10.10.3

Router1#
```

Breaking it all down

Tcp is the protocol used by http. Its port is 80 (Bonus info: https uses port 144).

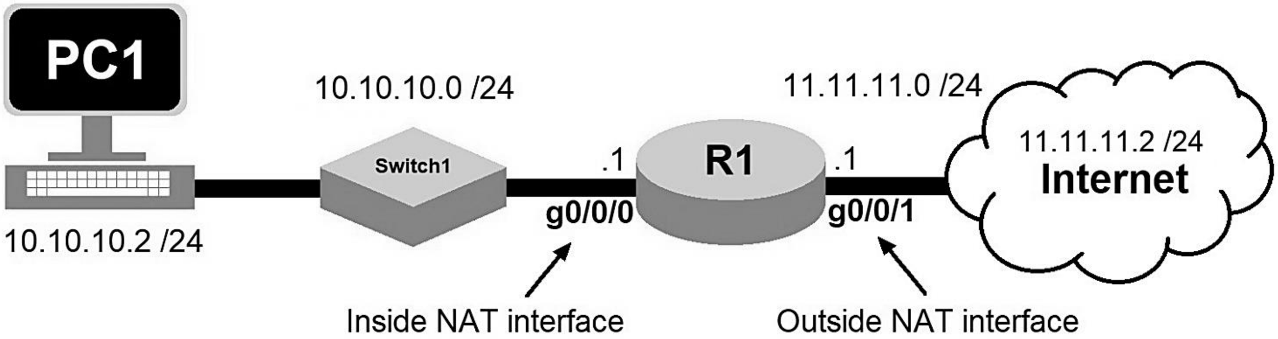
Eq 80 means “equal to port 80” which is the TCP port used by http traffic. Thus, in the show run report “eq www” is displayed.

To make sure that other networks and devices are still available to hosts on LAN 10.10.10.128, a specific destination IP address “**host 10.10.10.3**” is added to the end of the deny command string so that only that device is restricted.

Finally, refer to page-169, the final command string of “**permit any**” is added to account for the implicit deny any statement.

93. NAT (Network Address Translation) Static NAT

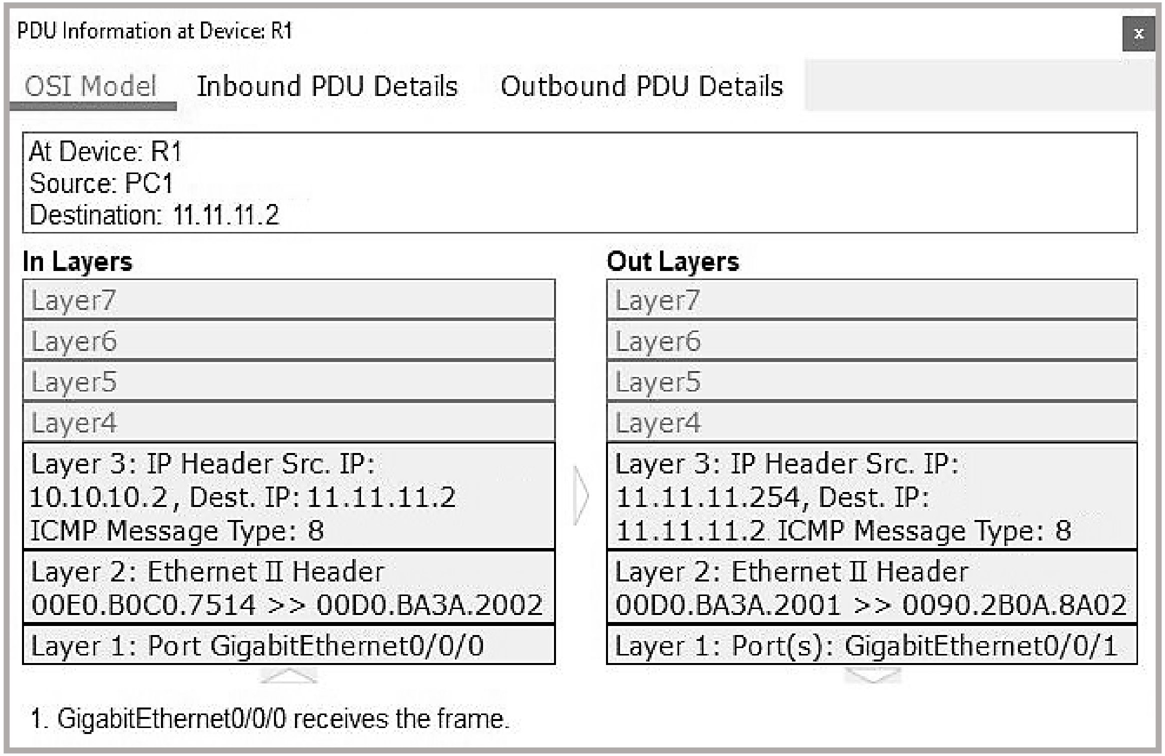
Network Address Translation is a **one-to-one** mapping of a private IP address to a public IP address. This is necessary because private network IP addresses cannot be routed across public networks (such as the internet). In his example static NAT will be used to allow PC1 to access the internet, and likewise for hosts outside the network to communicate with PC1.



```
R1#conf t
R1(config)#int g0/0/0
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#int g0/0/1
R1(config-if)#ip nat outside
R1(config-if)#exit
R1(config)#ip nat inside source static 10.10.10.2 11.11.11.254
R1(config)#exit
R1#
```

PC1's private IP address of 10.10.10.2 will now always be translated to the public IP address 11.11.11.254 whenever PC1 accesses the internet. Likewise, whenever a host or network outside of PC1's network communicates with PC1, they will use the public IP address 11.11.11.254 and the router will translate it back to 10.10.10.2 before routing it onto the LAN. This static assignment is maintained in the router's NAT Table

A **ping** from PC1 to internet address **11.11.11.2** verifies proper configuration. Notice when the packet arrives at g0/0/1 the **source address** is correctly NAT'd to the public routable address.

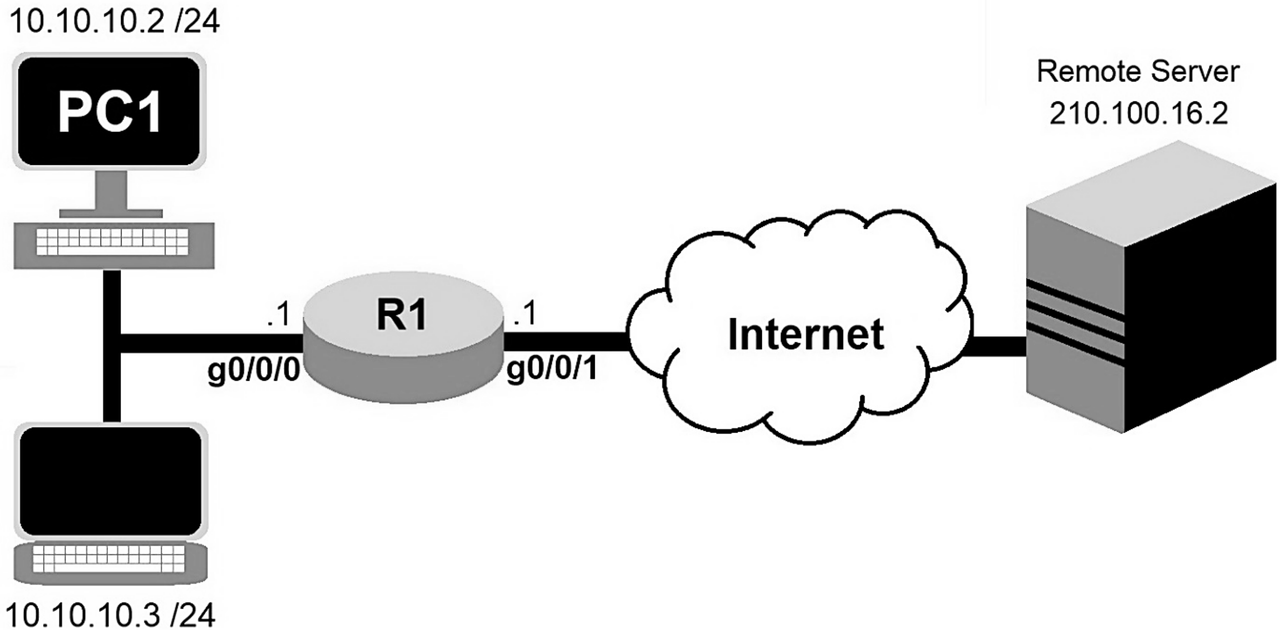


94. NAT (Network Address Translation) Dynamic NAT

With **Dynamic NAT** devices get their public addresses from a pool. These addresses are mapped on a first-come, first-served basis. So, in a way, Dynamic NAT functions somewhat like DHCP. In this example **Dynamic NAT** will be configured to support multiple devices by way of a range of available addresses.

```
R1#conf t
R1(config)#ip nat pool MYPOOL 210.100.16.3 210.100.16.10 netmask
255.255.255.0
R1(config #ip nat inside source list 1 pool MYPOOL
R1(config)#interface g0/0/0
R1(config)#ip nat inside
R1(config)#interface g0/0/1
R1(config-if)#ip nat outside
R1(config-if)#exit
R1(config)#access-list 1 permit 10.10.10.0 0.0.0.255
R1(config)#exit
```

Implicit permission to all devices on the 10.10.10.0 network.


10.10.10.2 /24
PC1
Remote Server
210.100.16.2
.1
R1
.1
Internet
g0/0/0
g0/0/1
10.10.10.3 /24

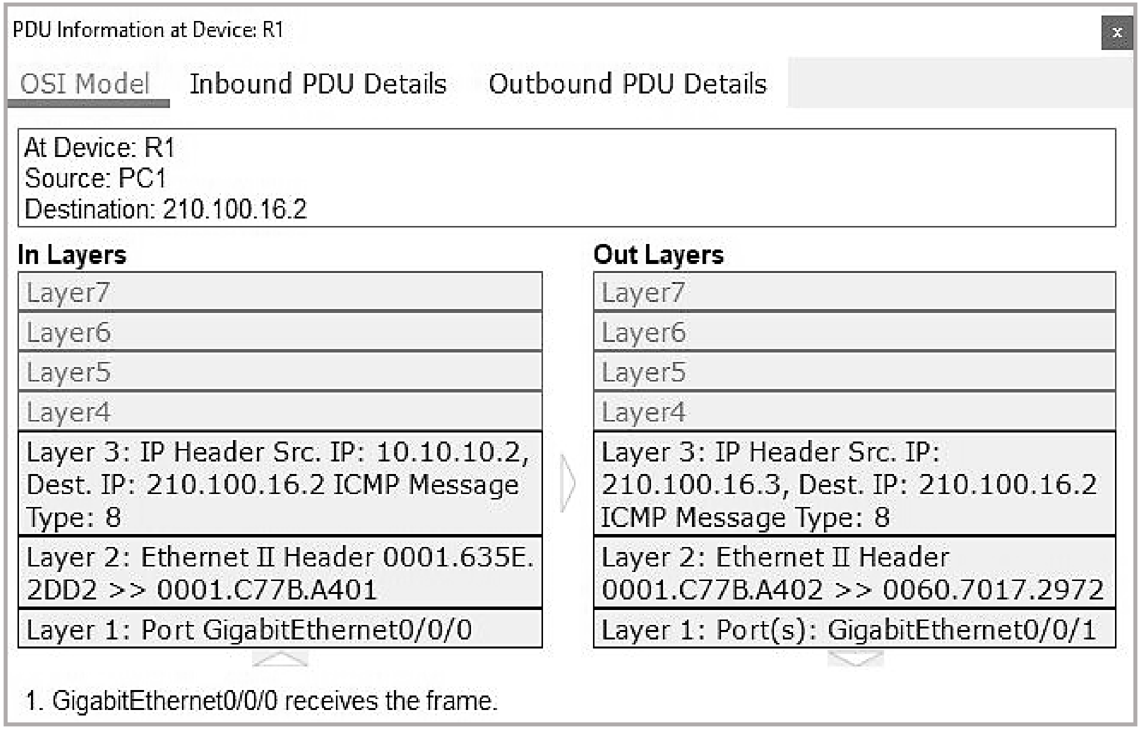
Verify with **Show NAT statistics.**

```
R1#sh ip nat statistics
Total translations: 0 (0 static, 0 dynamic, 0 extended)
Outside Interfaces: GigabitEthernet0/0/1
Inside Interfaces: GigabitEthernet0/0/0
Hits: 11  Misses: 12
Expired translations: 12
Dynamic mappings:
-- Inside Source
access-list 1 pool MYPOOL refCount 0
 pool MYPOOL: netmask 255.255.255.0
       start 210.100.16.3 end 210.100.16.10
       type generic, total addresses 8 , allocated 0 (0%), misses 0
```

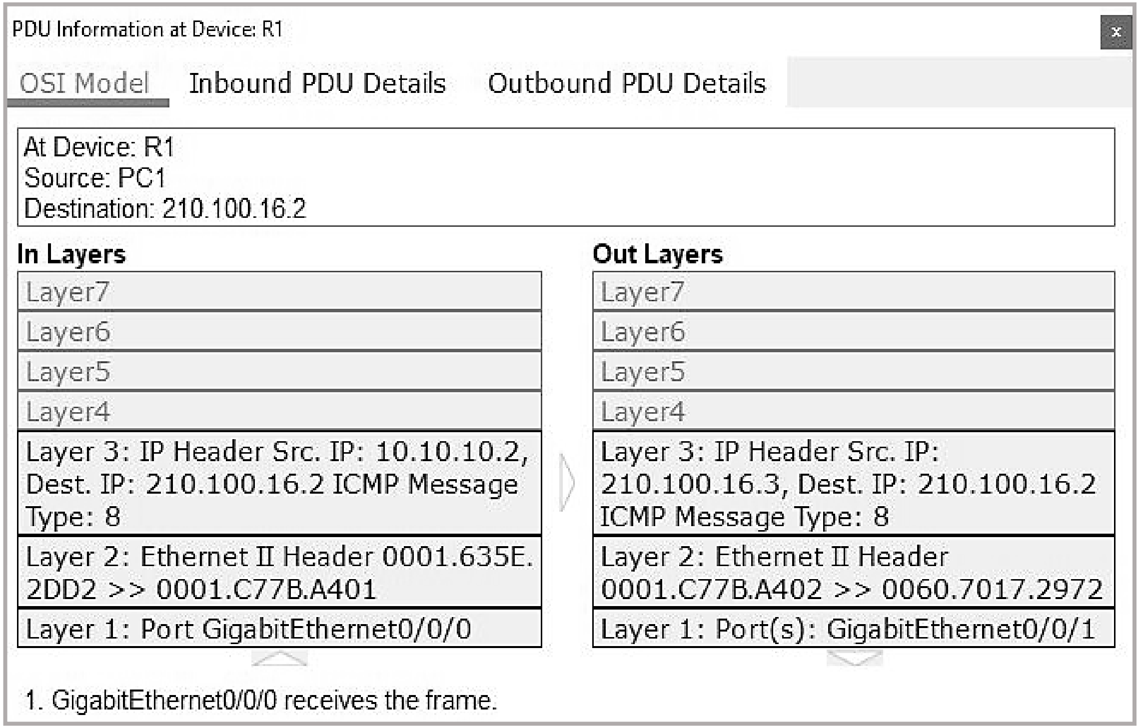
Notice the configured **range** of available addresses in NAT pool **MYPOOL**. Verifiable by pinging the remote server from PC1.

A **ping from PC1** to the remote server shows at g0/0/0 the packet retains PC1's local **source IP address**.

But upon arrival at g0/0/1 the address has been dynamically translated before it is routed to the server.

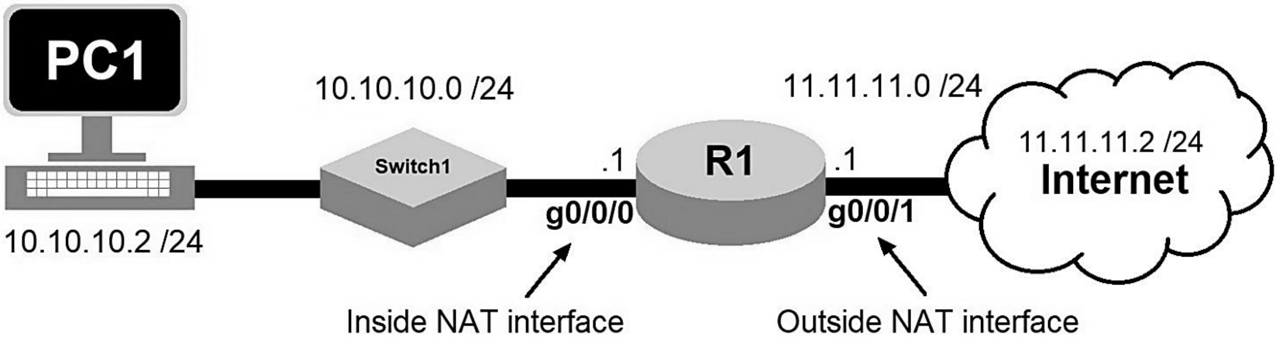


The **remote server's reply** also verifies a correct configuration.



95. NAT (Network Address Translation) Overload a.k.a. PAT (Port Address Translation)

With **NAT Overload (a.k.a. PAT)** we assign an **ACL** to an interface that tells the router to translate (or transform) all internal device IP addresses so they can be routed over the internet. **PAT** is also referred to as “overload” because it can create unique publicly routable address for as many internal devices that need one. This is more efficient and far less expensive than either Static NAT or Dynamic NAT.



Configuring NAT PAT on **R1**.

```
R1(config)#int g0/0/0
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#int g0/0/1
R1(config-if)#ip nat outside
R1(config-if)#exit
R1(config)#ip access-list standard MYACCESSLIST
R1(config-std-nacl)#permit 10.10.10.0 0.0.0.255
R1(config-std-nacl)#exit
R1(config)#ip nat inside source list MYACCESSLIST interface g0/0/1 overload
R1(config)#exit
R1#
```

```
interface GigabitEthernet0/0/0
 ip address 10.10.10.1 255.255.255.0
 ip nat inside
 duplex auto
 speed auto
!
interface GigabitEthernet0/0/1
 ip address 11.11.11.1 255.255.255.0
 ip nat outside
 duplex auto
 speed auto
!
ip nat inside source list MYACCESSLIST interface GigabitEthernet0/0/1 overload

R1#
```

Verify configuration with **the Show Running-Config** command. Notice the last line in the report:

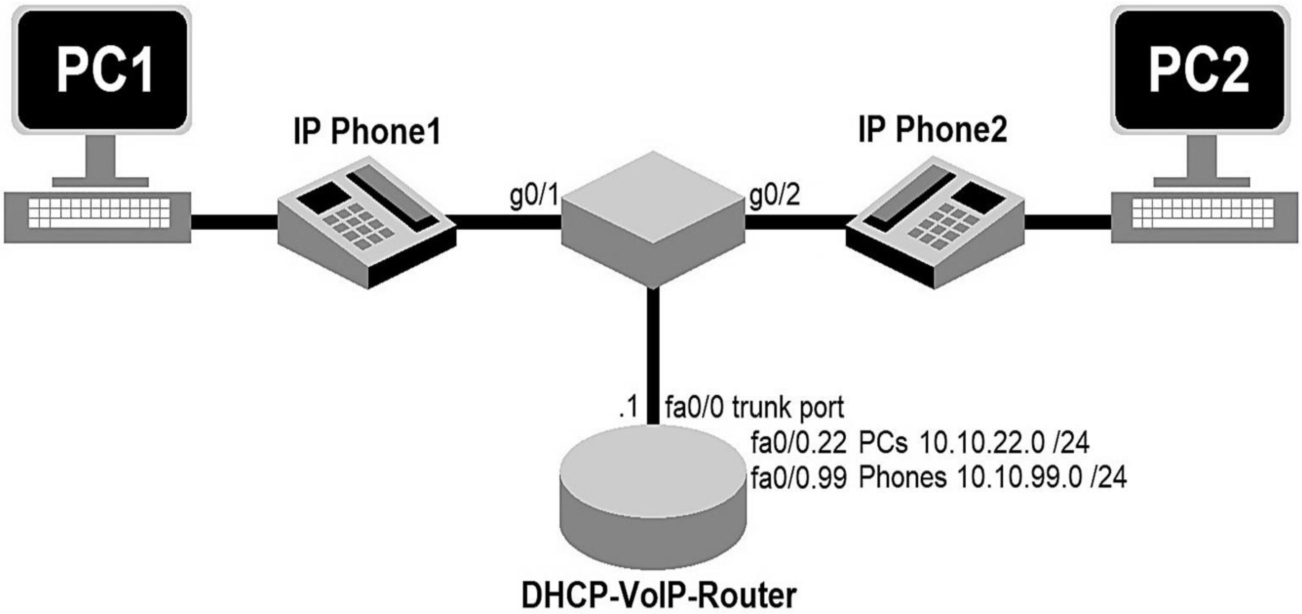
It tells us **NAT** is applied to the **inside** (inside interface g0/0/0) because it is the **source** for packets needing NAT; the list it will use is **MYACCESSLIST** to define which network's packets will be NAT'd; the outside interface is **GigabitEthernet g0/0/1**; the **overload** command instructs the router to PAT all internal IP addresses from that inside source interface.

Further verification with **Show Access-Lists** and **Show IP NAT Statistics** commands.

```
R1#sh acc
Standard IP access list MYACCESSLIST
    10 permit 10.10.10.0 0.0.0.255 (20 match(es))

Gateway-Router1#sh ip nat stat
Total translations: 0 (0 static, 0 dynamic, 0 extended)
Outside Interfaces: GigabitEthernet0/0/1
Inside Interfaces: GigabitEthernet0/0/0
Hits: 10  Misses: 42
Expired translations: 10
Dynamic mappings:
```

96. VoIP (Voice Over IP)

Voice Over IP is a process where we use our existing cables to route and switch voice traffic along with regular data traffic. This is accomplished by way of a separate voice VLAN that can run over the same access port. VOIP requires configurations similar to standard VLANs, but a few additions.



Configure DHCP for VoIP and PC vlans.

```
DHCP-VoIP-Router#
DHCP-VoIP-Router#conf t
DHCP-VoIP-Router(config)#int f0/0.1
DHCP-VoIP-Router(config-subif)#encapsulation dot1q 99
DHCP-VoIP-Router(config-subif)#ip address 10.10.99.1 255.255.255.0
DHCP-VoIP-Router(config-subif)#description VoIP-vlan-99
DHCP-VoIP-Router(config-subif)#exit
DHCP-VoIP-Router(config)#exit
DHCP-VoIP-Router#
DHCP-VoIP-Router(config)#
DHCP-VoIP-Router(config)#int f0/0.2
DHCP-VoIP-Router(config-subif)#encapsulation dot1q 22
DHCP-VoIP-Router(config-subif)#ip address 10.10.22.1 255.255.255.0
DHCP-VoIP-Router(config-subif)#description PCs-vlan-22
```

```
DHCP-VoIP-Router(config-subif)#exit
DHCP-VoIP-Router(config)#exit
DHCP-VoIP-Router#
```

Configure Telephony

The number of phone numbers per system configuration is 1 - 144.

```
DHCP-VoIP-Router#
DHCP-VoIP-Router#conf t
DHCP-VoIP-Router(config)#telephony-service
DHCP-VoIP-Router(config-telephony)#max-dn 2
DHCP-VoIP-Router(config-telephony)#max-ephones 2
DHCP-VoIP-Router(config-telephony)#auto-reg-ephone
DHCP-VoIP-Router(config-telephony)#ip source 10.10.99.1 port 2000
DHCP-VoIP-Router(config-telephony)#exit
DHCP-VoIP-Router(config)#exit
```

E-phones can register with the server in two ways: Generic auto-reg (shown). Or auto-reg a specific number of phones.

Configure Phone 1

```
DHCP-VoIP-Router#conf t
DHCP-VoIP-Router(config)#
DHCP-VoIP-Router(config)#ephone 1
DHCP-VoIP-Router(config-ephone)#button 1:1
DHCP-VoIP-Router(config-ephone)#exit
DHCP-VoIP-Router(config)#ephone-dn 1
DHCP-VoIP-Router(config-ephone-dn)#number 9901
DHCP-VoIP-Router(config-ephone-dn)#exit
DHCP-VoIP-Router(config)#exit
```

Configure Phone 2

```
DHCP-VoIP-Router#conf t
DHCP-VoIP-Router(config)#
DHCP-VoIP-Router(config)#ephone 2
```

```
DHCP-VoIP-Router(config-ephone)#button 1:2
DHCP-VoIP-Router(config-ephone)#exit
DHCP-VoIP-Routerconfig)#ephone-dn 2
DHCP-VoIP-Router(config-ephone-dn)#number 9902
DHCP-VoIP-Router(config-ephone-dn)#exit
DHCP-VoIP-Router(config)#exit
```

Verify configuration starting with the **Show IP DHCP Binding** command to make sure the phones are receiving the correct IP addresses according to their VLAN: either 22 or 99. Which also lets us identify what devices are connected and their MACs.

```
DHCP-VoIP-Router#sh ip dhcp  binding
IP address       Client-ID/              Lease expiration        Type
                 Hardware address
10.10.99.14      00D0.5886.4EB0           --                     Automatic
10.10.99.12      0001.C702.2B48           --                     Automatic
10.10.22.11      00D0.BA40.51E6           --                     Automatic
10.10.22.12      0090.2BA8.4A88           --                     Automatic
DHCP-VoIP-Router#
```

Follow with **Show Running-Config.**

Note: Report reformatted here to fit on one page.

```
DHCP-VoIP-Router#sh run

telephony-service
 max-ephones 2    ephone-dn 1    ephone-dn 2
 max-dn 2         number 9901    number 9902
 ip source-address 10.10.99.1 port 2000
!
ephone 1                              ephone 2
 device-security-mode none             device-security-mode none
 mac-address 0001.C702.2B48            mac-address 00D0.5886.4EB0
 type 7960                             type 7960
 button 1:1                            button 1:2
!

DHCP-VoIP-Router#
```

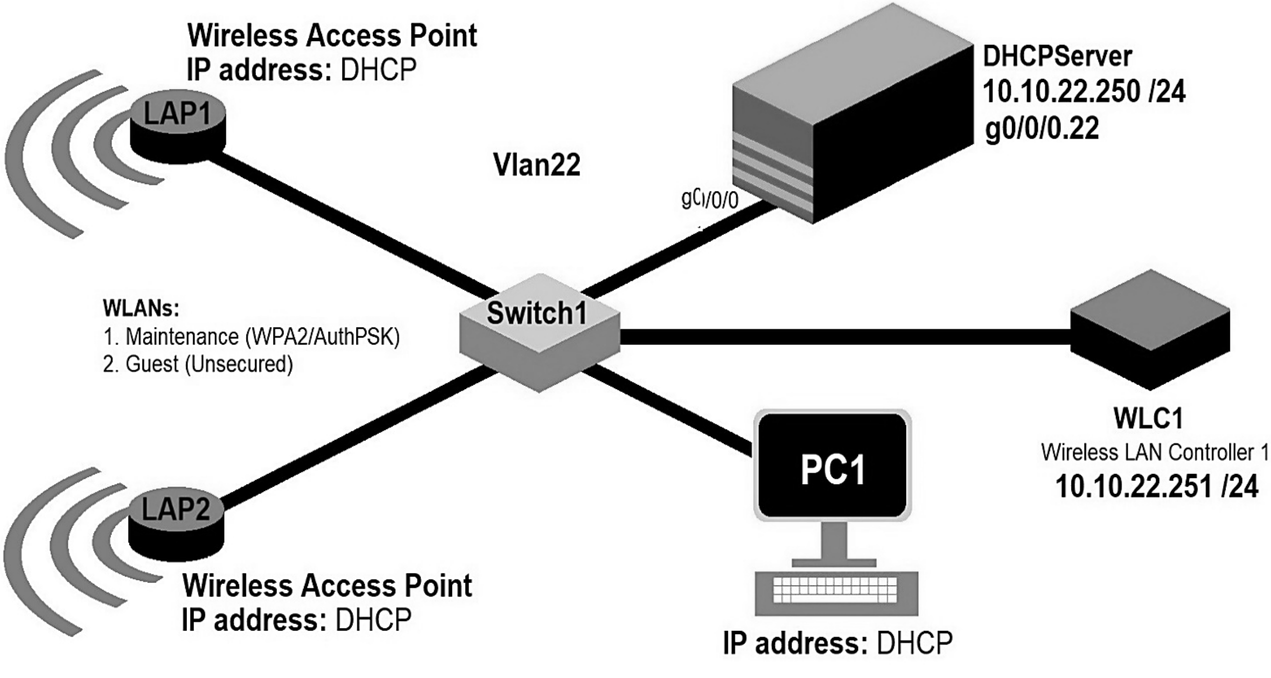
97. VoIP (Voice Over IP): Show Ephone

Verify further with this new show command: **Show Ephone.** Notice this show command also returns the phone model number, dynamically assigned IP addresses, phone number, and MAC address.

```
DHCP-VoIP-Router#sh ephone

ephone-1 Mac:0001.C702.2B48 TCP socket:[1] activeLine:0
REGISTERED in SCCP ver 12 and Server in ver 8 mediaActive:0
offhook:0 ringing:0 reset:0 reset_sent:0 paging 0 debug:0 caps:8
IP:10.10.99.12 1027 7960   keepalive 43 max_line 2
button 1: dn 1  number 9901 CH1   IDLE


ephone-2 Mac:00D0.5886.4EB0 TCP socket:[1] activeLine:0
REGISTERED in SCCP ver 12 and Server in ver 8 mediaActive:0
offhook:0 ringing:0 reset:0 reset_sent:0 paging 0 debug:0 caps:8
IP:10.10.99.14 1027 7960   keepalive 43 max_line 2
button 1: dn 2  number 9902 CH1   IDLE

DHCP-VoIP-Router#
```

98. Wireless: Basic setup of a wireless network and WLC

For this configuration a **vlan 22** and **DHCP server** will be configured to provide IP addresses to the PC and two wireless access points. A **wireless controller** will also be configured along with two **access points**.



WLC configuration will include the following configurations, most already covered in previous sections:

System name: Wireless22

System Username: Admin1

System Password: Cisco123

Group: default

System parameters are strictly for logging into the Wireless Controller itself to make administrative level configurations on the Maintenance network.

WLANs:

- Maintenance (secure login using WPA2+)
- Guest (no security)

The reason we use a **wireless controller** is efficiency. A wireless controller centrally manages all the access points (APs) so we do no not have to manage each separately.

Password: Maintenance22

Note: To test the configuration, wireless devices will be set to “Guest” and will automatically login and receive an IP address from DHCP. This is similar to how a coffee shop employee gives a customer the shop’s wi-fi name so they can access the network without needing a password. Meanwhile, although both WLANs are technically in Vlan22 the guest will not be able to access the Maintenance WLAN without knowing its name and having the login credentials.

DHCP Pool **Wireless22** verified.

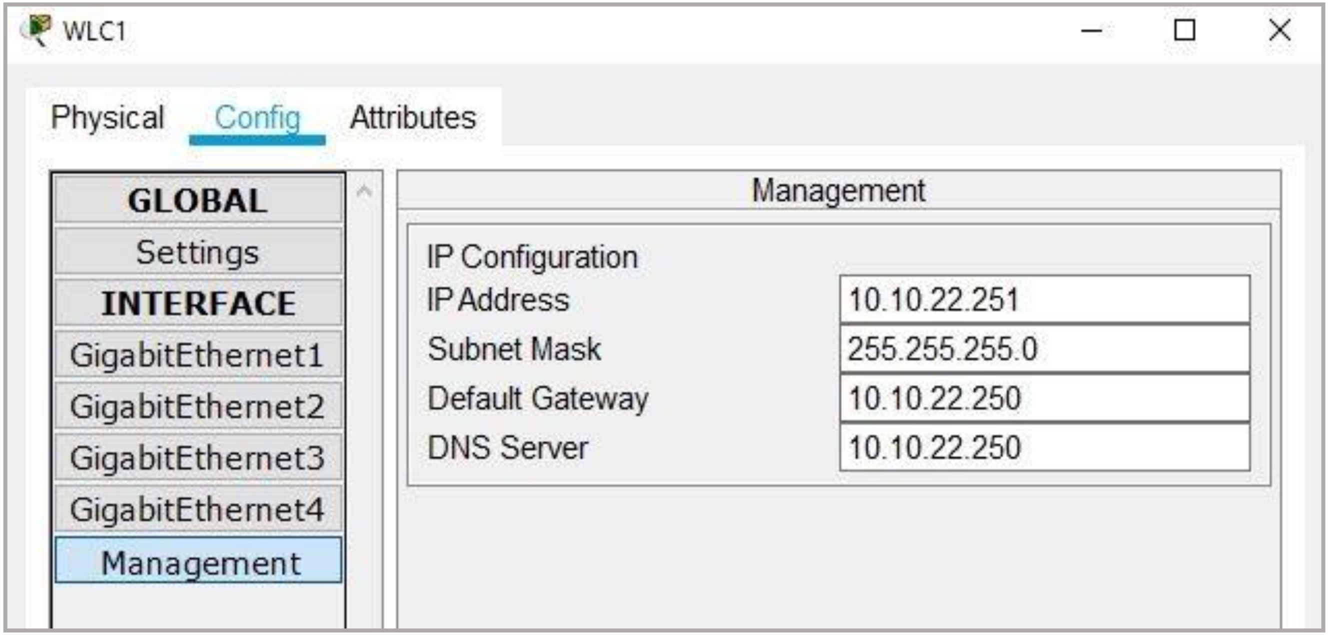
```
DHCPServer#sh run
Building configuration...
!
hostname DHCPServer
!
ip dhcp excluded-address 10.10.22.250 10.10.22.254
!
ip dhcp pool Wireless22
 network 10.10.22.0 255.255.255.0
 default-router 10.10.22.250
 dns-server 10.10.22.250
 domain-name Wireless22
```

Sub-interface verified as up/up for vlan 22.

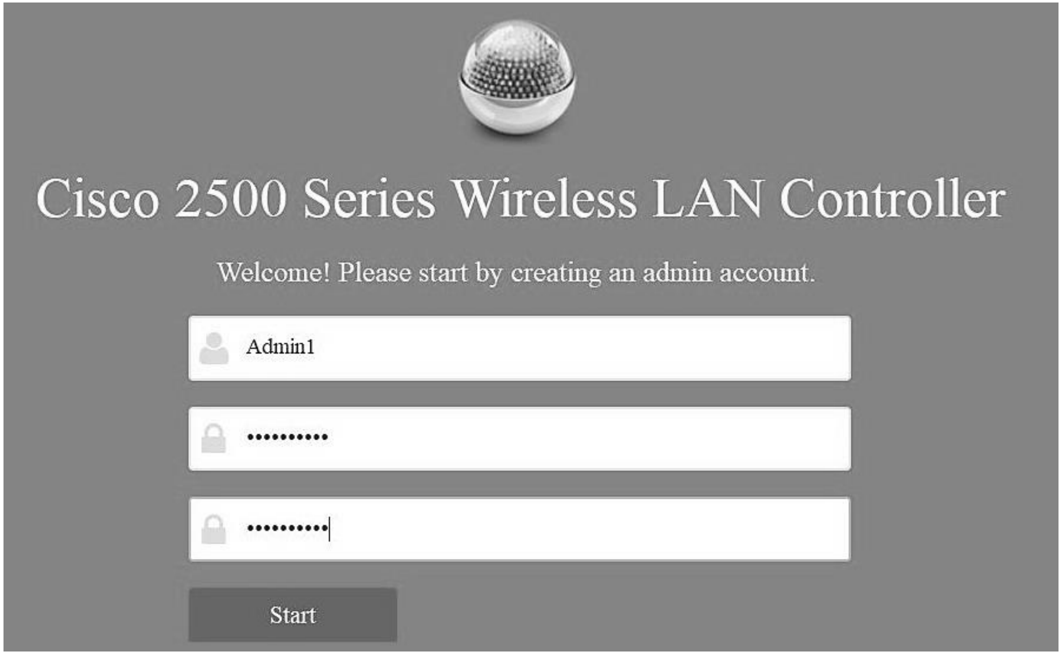
```
DHCPServer#sh ip int br
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0/0   unassigned      YES manual up                    up
GigabitEthernet0/0/0.2210.10.22.250    YES manual up                    up
GigabitEthernet0/0/1   unassigned      YES NVRAM  administratively down down
Vlan1                  unassigned      YES unset  administratively down down
```

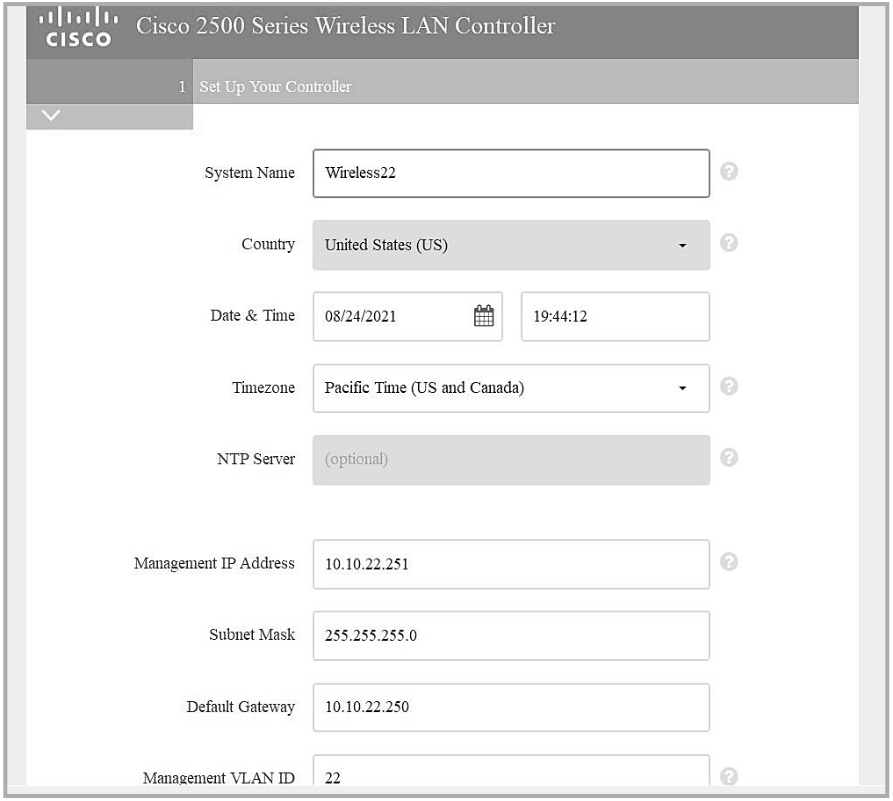
Note: the interface must be in the <u>no</u> shut position for its sub-interfaces to be operational (up/up).

The Wireless Controller **WLC1** is configured via the simple UI. **10.10.22.250** is the IP address of the default-gateway, sub-interface g0/0/0.22 of VLAN22.

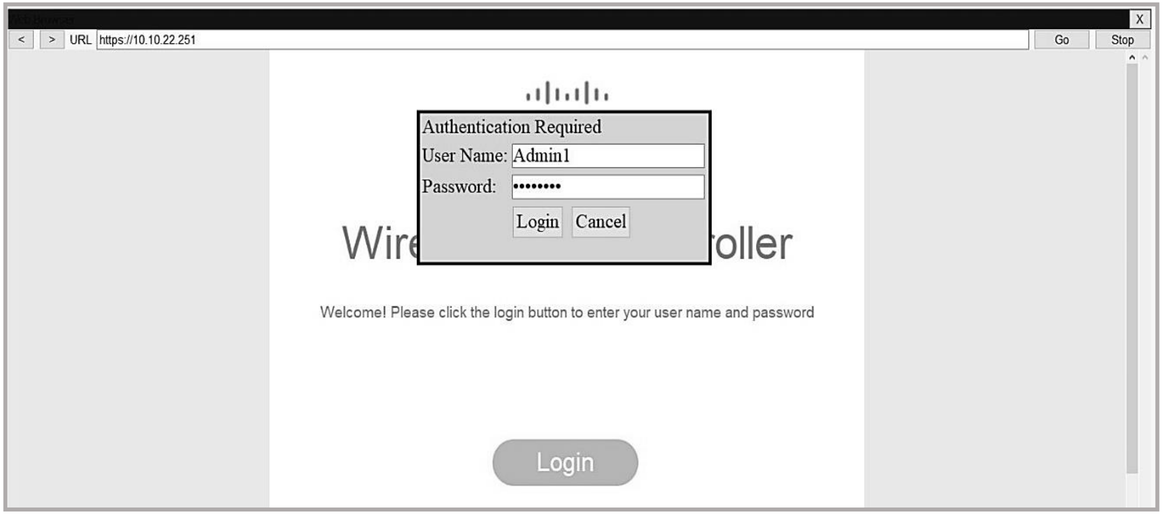


PC1 is used to access the **Wireless Controller** via its IP address http://10.10.22.251 to configure a username **Admin1** and password **Cisco123** for the Maintenance WLAN. An additional WLAN for guests will be added to the group. This will result in a total of 2 WLANS: Maintenance and Guest.

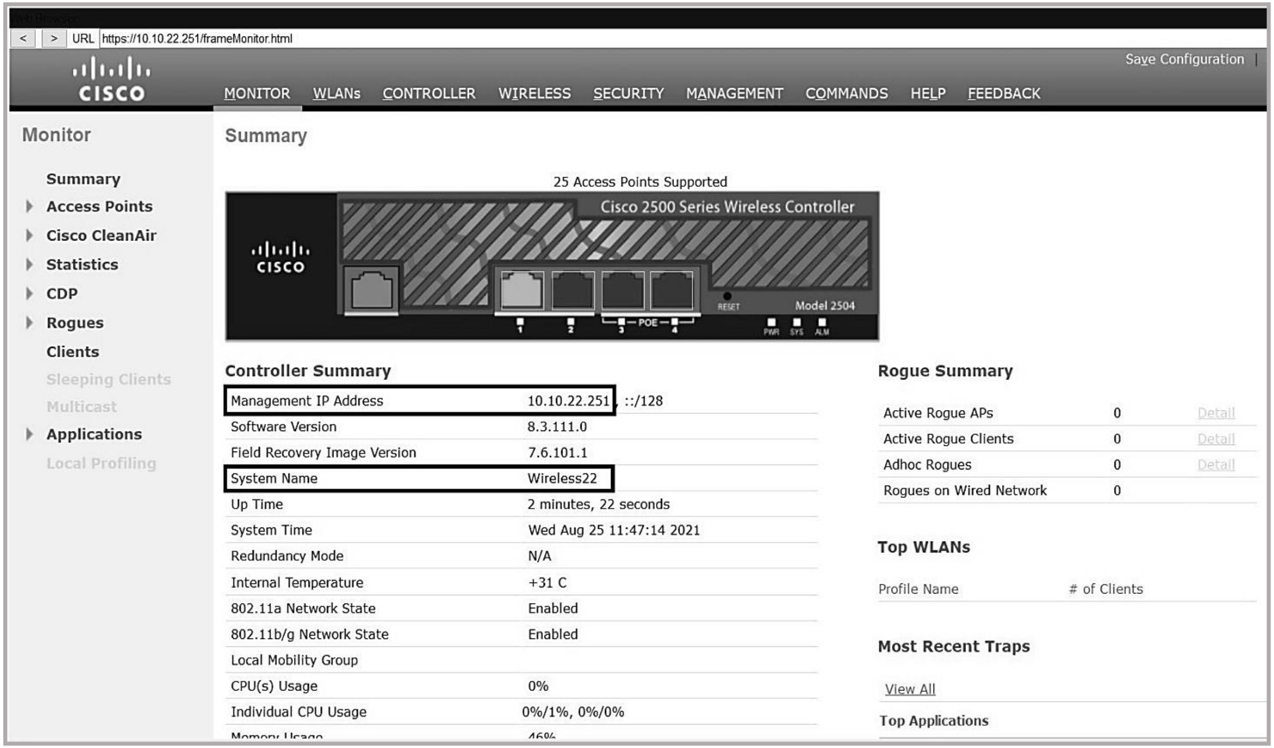



Cisco 2500 Series Wireless LAN Controller
1 Set Up Your Controller
System Name
Wireless22
Country
United States (US)
Date & Time
08/24/2021
19:44:12
Timezone
Pacific Time (US and Canada)
NTP Server
(optional)
Management IP Address
10.10.22.251
Subnet Mask
255.255.255.0
Default Gateway
10.10.22.250
Management VLAN ID
22

After completing initial setup, login is done again but with **https** because a username and password are now needed.



99. Wireless: WLC setup verification on the Monitor tab.

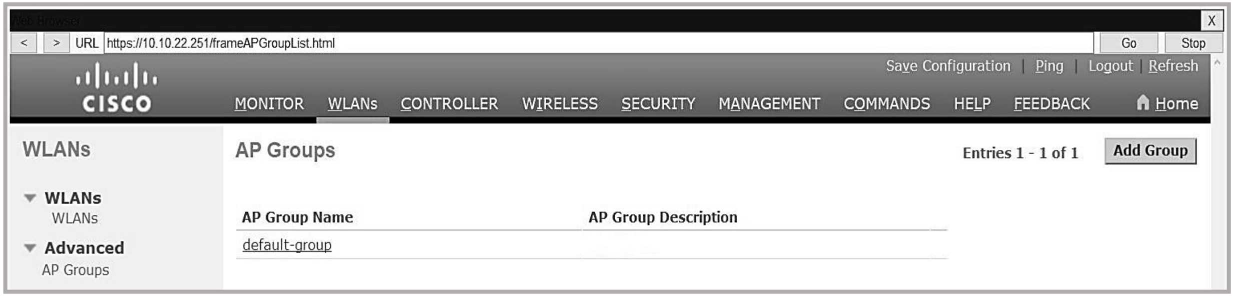


100. Wireless: WLAN and Group configuration / and Worker bridges.

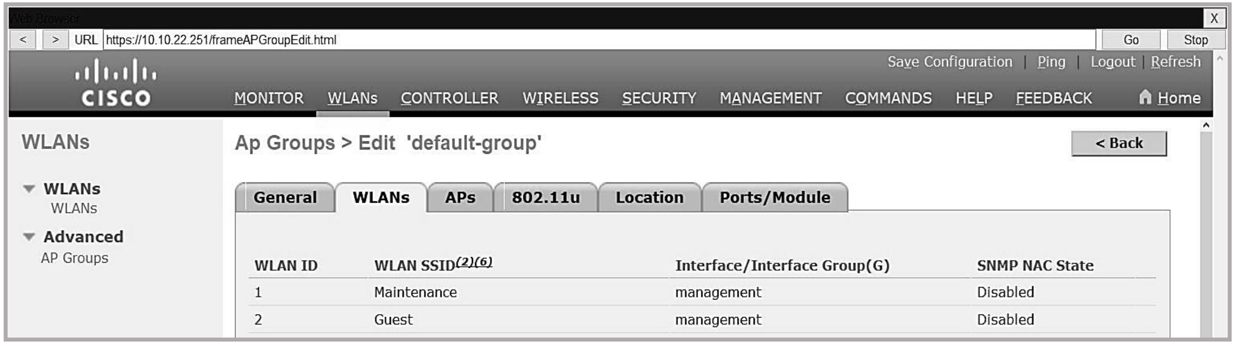
The default WLAN name can be changed. It is changed to **Maintenance** by clicking on the ID number and making the change, and then finishing by clicking the Apply button. Another WLAN named **Guest** is added by using the Create New function. **Notice** the Maintenance WLAN has security protocols while the Guest WLAN does not. Guests *can* access the network but <u>cannot</u> log into any network devices.

Note: User interfaces differ between manufacturers and models of WLC. This is just an example.

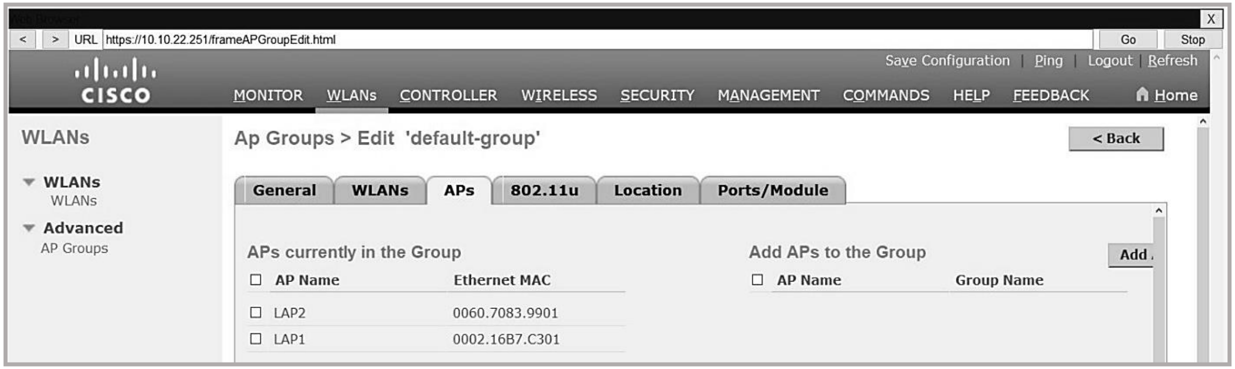
In the **AP Groups tab** the default-group is given a description to match the system name by clicking on default-group and entering a description as shown below. Then clicking the Apply button.



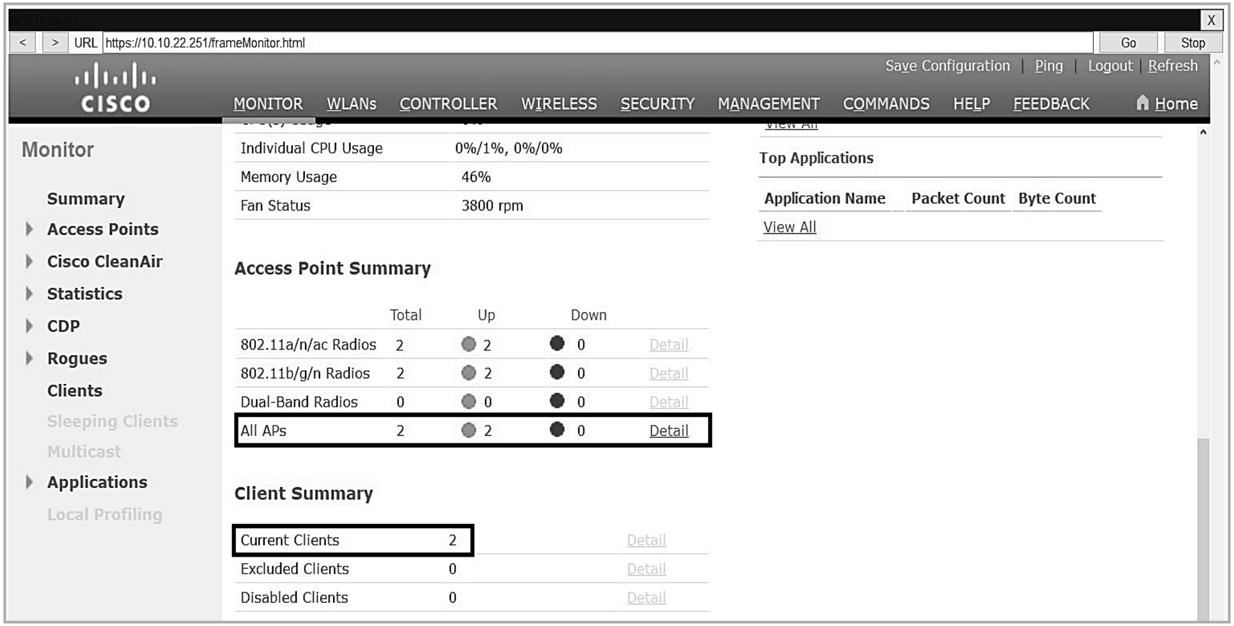
On the **WLANs tab** under default-group the two WLANs now appear as part of that group.



On the **APs tab** the two access points **LAP1** and **LAP2** appear along with their MAC addresses.

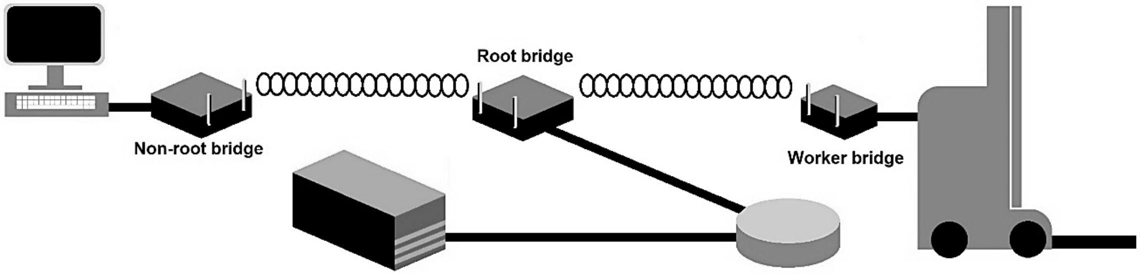


Returning to the Monitor tab and scrolling down, the number of APs and their status, plus current guest devices, can be seen. In this example a nearby phone and a tablet have logged into the Guest WLAN.



Worker bridges

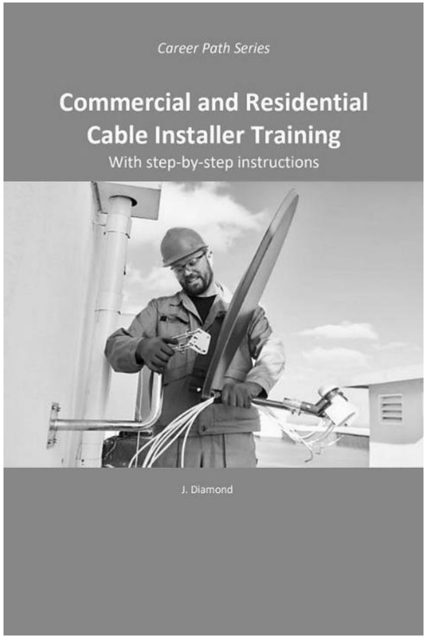
A wireless **worker bridge** is an autonomous device suitable for connecting small network segments. It is ideal for mobility. For example: a forklift with an onboard shopping system but no wireless capability could use a worker bridge to connect it to a warehouse's main network to update inventory levels in real-time.

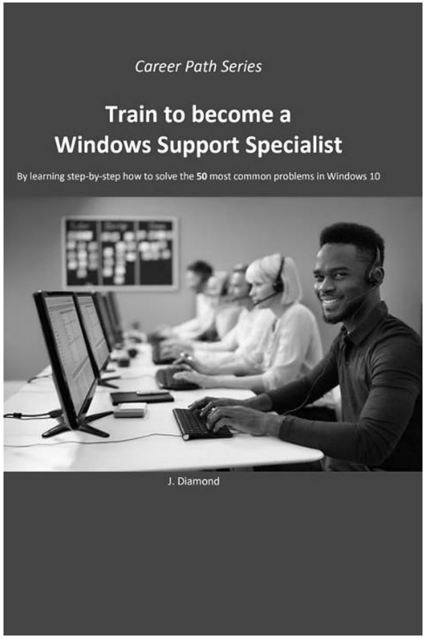


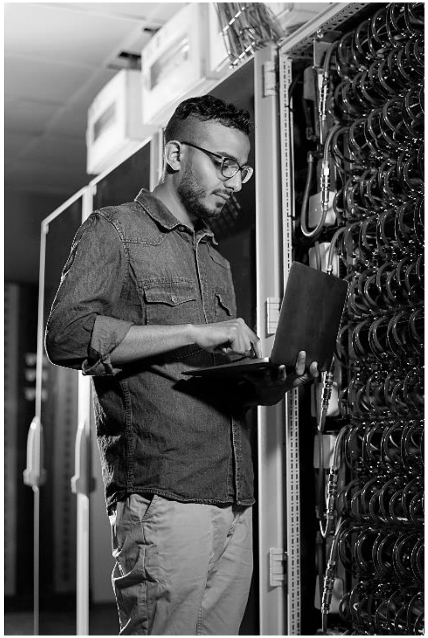
The only limitation is that **worker bridge devices** are smaller than stationary enterprise devices and are therefore not optimized for connecting large network segments with many devices. For large networks, root and non-root wireless bridging devices are recommended. In the previous wireless network topology, were Switch1 *wireless* it would likely be the root bridge for that network. ***Note:** Worker bridges can also be wired like any other AP and used in a stationary topology if desired.

Thank you

For more job training books, online classes, and free training, please visit **CCNAUltimateLabs.com**







Notes