



This document describes the documented class exercises . The objective of the document is to present complete and easy to follow detailed procedures to describe the exercises done in class.

Course Procedure Manual

420-634-AB NETWORK INFRASTRUCTURE
AND AUTOMATION (CISCO CCNA III :
Enterprise Networking, Security, and
Automation)

Teacher: Michael Hughes
Student: Monica Perez Mata
Student id : 2498056

Table of Contents

1	Introduction	4
2	General activities	4
2.1	Splashtop and Computer	4
2.2	Verify VMWare Workstation Pro is installed	6
2.3	Updating VMware Workstation Pro.....	6
2.3.1	Post VMware Workstation Pro upgrade activities	13
2.4	Delete VM	15
2.5	Create a snapshot for VM	17
3	CCNA 3 Enterprise Networking, Security and Automation	18
3.1	CISCO ENSA - Commands & Procedures Overview	18
3.1.1	Module 2: Single-Area OSPFv2 Configuration	18
3.1.2	Configure and Apply a Numbered Standard IPv4 ACL Syntax.....	21
3.1.3	Module 6: NAT for IPv4.....	29
3.2	Packet tracer exercises	33
3.2.1	Exercise 2.2.13 - PacketTracer - Point-to-Point Single-Area OSPFv2 Configuration	34
3.2.2	Exercise 2.3.11 - Packet Tracer - Determine the DR and BDR.....	14
3.2.3	Exercise 2.4.11 - Packet Tracer - Modify Single-Area OSPFv2.....	23
3.2.4	Exercise 2.5.3 - Packet Tracer - Propagate a Default Route in OSPFv2	28
3.2.5	Exercise 2.6.6 - Packet Tracer - Verify Single-Area OSPFv2	32
3.2.6	Exercise 2.7.1 - Packet Tracer - Single-Area OSPFv2 Configuration	40
3.2.7	Exercise 2.7.2 - Packet Tracer - Configure Single-Area OSPFv2 - Physical Mode	51
3.2.8	Exercise 4.1.4 - Packet Tracer - Access Control List Demonstration	10
3.2.9	Exercise 5.1.8 - Packet Tracer - Configure Numbered Standard IPv4 ACLs	17
3.2.10	Exercise 5.1.9 - Packet Tracer - Configure Named Standard IPv4 ACLs	28
3.2.11	Exercise 5.2.7 - Packet Tracer - Configure and Modify Standard IPv4 ACLs.....	36
3.2.12	Exercise 5.4.12 - Packet Tracer - Configure Extended ACLs - Scenario 1	51
3.2.13	Exercise 5.4.13 - Packet Tracer - Configure Extended IPv4 ACLs - Scenario 2	61
3.2.14	Exercise 5.5.1 - Packet Tracer - IPv4 ACL Implementation Challenge	73
3.2.15	Exercise 5.5.2 - Packet Tracer - Configure and Verify Extended IPv4 ACLs - Physical Mode	81
3.2.16	Exercise 6.2.7 - Packet Tracer - Investigate NAT Operations	6
3.2.17	Exercise 6.4.5 - Packet Tracer - Configure Static NAT	8
3.2.18	Exercice 6.5.6 - Packet Tracer - Configure Dynamic NAT	14
3.2.19	Exercice 6.6.7 - Packet Tracer - Configure PAT	16
3.2.20	Exercise 6.8.1 -Packet Tracer - Configure NAT for IPv4	21

3.2.21	Exercise 6.8.2 - Lab - Configure NAT for IPv4.....	26
3.3	GNS3 Labs.....	41
3.3.1	OSPF and Packet Capture - DR BDR.....	41
3.3.2	OSPF add Fedora VM with Apache SSH services – Access Control Lists and NAT	78
3.3.3	OSPF - ACLS , Fedora with Apache SSH RSYSLOG , Router DHCP Cloud NAT.....	117
3.3.4	OSPF - ACLS , Fedora with Apache SSH RSYSLOG and TFTP , Router DHCP Cloud NAT	133
3.3.5	Final GNS3 project CCNA 3	145
3.1	Test Automation with Ansible	183
3.1.1	Setup.....	183
3.1.2	Ansible playbooks	213
3.2	Test automation with python and GNS3 Network Automation App	17
3.2.1	Install and configure Network automation APP	17
3.2.2	GNS3 Python excercises.....	32

1 Introduction

This document outlines the procedures learned during the courses

- 420-634-AB NETWORK INFRASTRUCTURE AND AUTOMATION (CISCO CCNA III : Enterprise Networking, Security, and Automation)

The aim of the document is to present complete and easy to follow detailed procedures to describe the necessary steps for all the procedures carried out as class exercises.

The exercises cover some concepts like:

- Designated Router (DR)
- Backup Designated Router (BDR)
- Open Shortest Path First Version 2 (OSPFv2)
- Network Address Translation (NAT)
- Port Address Translation (PAT)
- Access Control Lists (ACL)
- Network Automation with Ansible and Python
- Syslog (System Logging Protocol)
- Rsyslog (Reliable System Logging Protocol)

2 General activities

This section describes the pre-requisites and the setup to work with procedures related to operating systems.

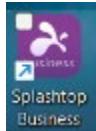
The required pre-requisites:

- 1) The use of Splashtop to securely access John Abbott College Computer Lab is up and running.
- 2) A computer is available to work in John Abbott College Computer Lab
- 3) Vmware Workstation Pro Virtual environment software is installed to create virtual machines on the assigned computer and is up and running.

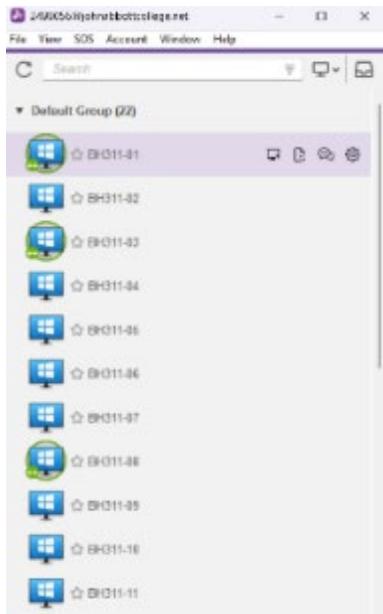
The procedures for installing Splashtop on your home computer are not included in this manual.

2.1 Splashtop and Computer

Splashtop Business is a remote desktop software that allows users to securely access their computers from anywhere. As a pre-requisite for all activities in this document, Splashtop Business is installed, and a PC is assigned, and both are working

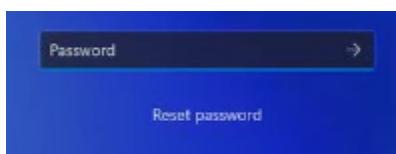


- A) Splashtop Business Application is installed in your home computer, user is logged in and a computer list appears on Splashtop Business, as shown in the image below:



The computer assigned is correctly working when double click computer starts.

- B) User login to computer with appropriate user and password



- C) Windows desktop (like the image below) appears when user logs in.



2.2 Verify VMWare Workstation Pro is installed

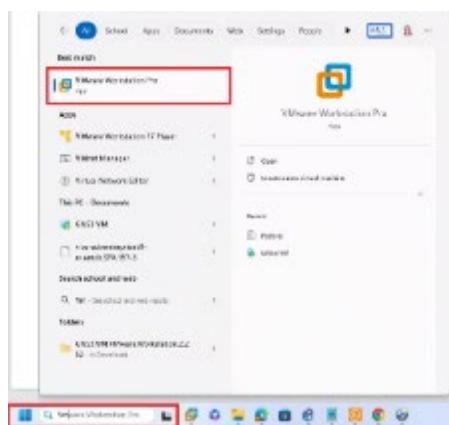
VMware Workstation Pro enables users to set up virtual machines (VMs) on a single physical machine.



As a pre-requisite for all activities in this document

- 1) Make sure VMware Workstation Pro is installed. Check in the search tab at the left down corner of the desktop and look for VMware Workstation Pro.

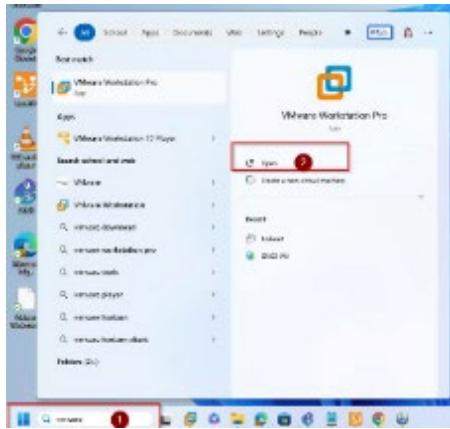
The application appears in the menu.



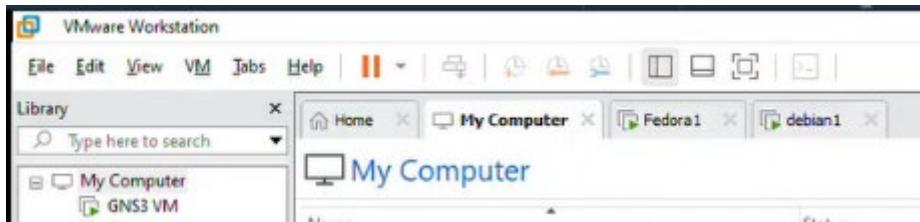
2.3 Updating VMWare Workstation Pro

- A) Open the VMWare Workstation App

- 1 Look for application in windows search
- 2 Once VMWare Workstation Pro appears, open application

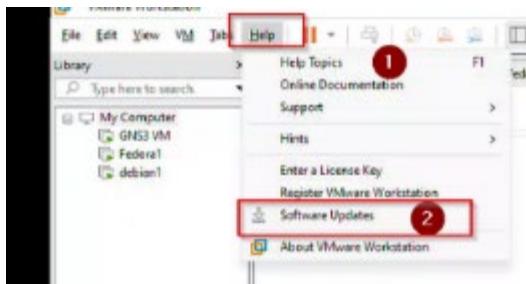


B) VMware workstation opens:

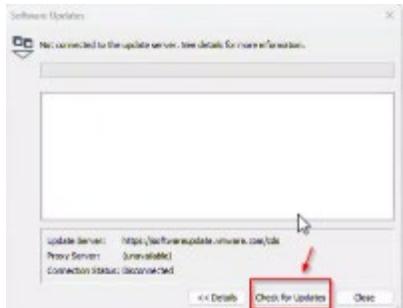


C) Select from top menu and submenu

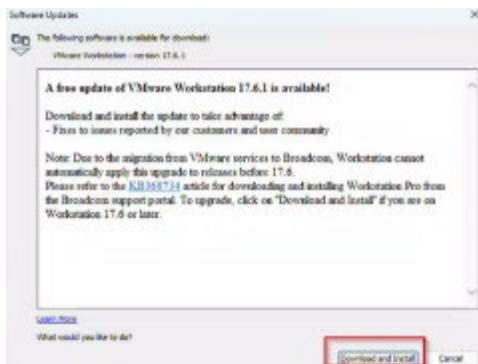
1. Help
2. Select Software Updates



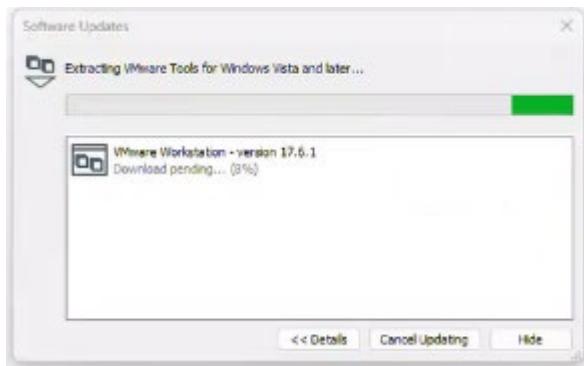
D) A new window will open, select Check for updates



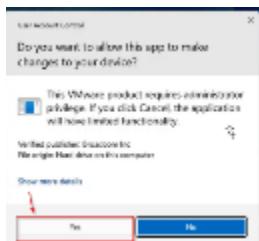
- E) After a couple of second a new window appears indicating upgrades are available, click “Download and Install”



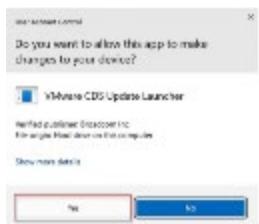
- F) Wait while new VMWare version is extracting



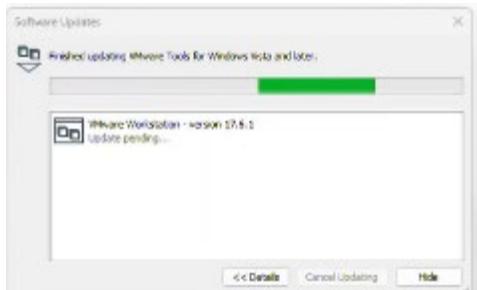
- G) When prompted that “Do you want to allow this app to make changes to your device? This VMware product requires administrator privilege. Click “Yes”



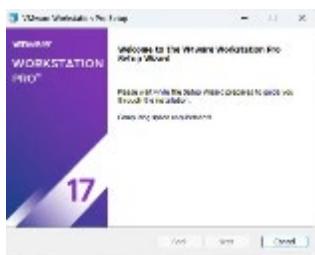
- H) When prompted “Do you want to allow this app to make changes to your device? VMware CDS Update Launcher”, click “Yes”.



- I) Finished uploading VMware tools window appears, wait until finished.



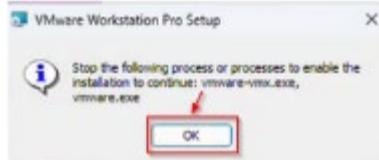
- J) New window with message “Welcome to VMware Workstation Pro Setup Wizard, wait until “Next” is enabled.



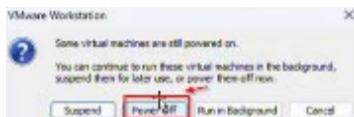
- K) When the in the window “Welcome to the VMware Workstation Pro Setup Wizard” “Next” is enabled, click on it.



- L) If the following window appears is because the VMWare Workstation is running virtual machines. Press “OK” and go ahead to close the VMWare Workstation

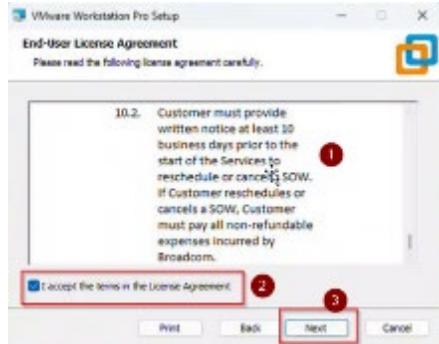


- M) To Stop VMWare Workstation all virtual machines should be powered off. Press “Power Off”

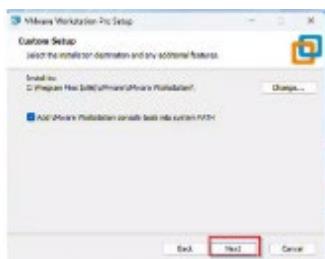


- N) End-User License Agreement,

1. Read the End-User License Agreement
2. Accept the End-User License Agreement by Selecting “I accept the terms in the License Agreement”.
3. Click “Next”

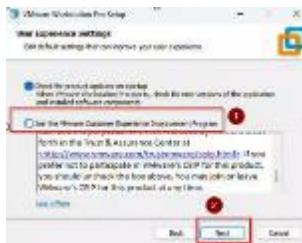


In the “Custom Setup” window, keep the installation destination, select “Add VMWare Workstation console tools into system PATH” and then click “Next”

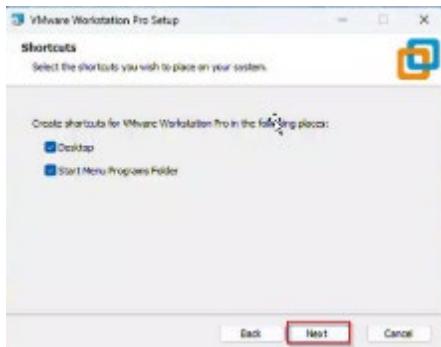


O) When the “User Experience Settings” window pops up,

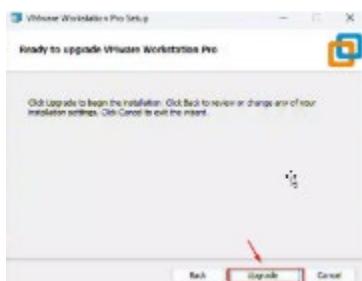
1. Uncheck the checkbox “Join the VMware Customer Experience Improvement Program”
2. Click “Next”



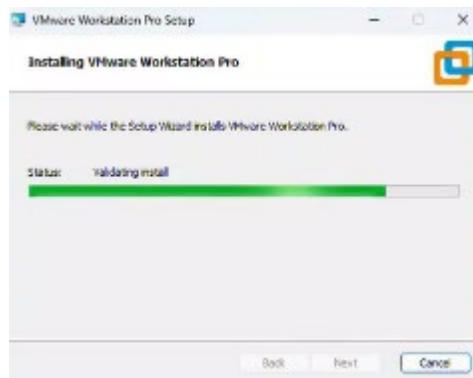
- P) When the “Shortcuts” window pops up, both boxes should be checked, then click “Next”.



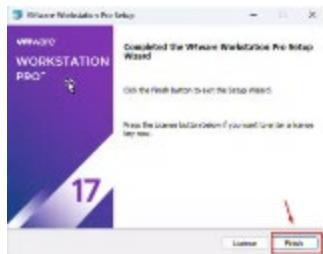
- Q) When the window “Ready to upgrade VMware Workstation Pro” appears click “Upgrade”



- R) The window “Installing VMware Workstation Pro” showing the installation process is initiated, the green bar indicates the process status. Please wait this can take time.



- S) When the “Completed the VMware Workstation Pro Setup Wizard” window pops up, click “Finish”.

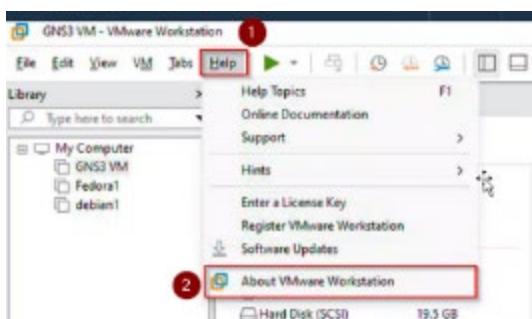


2.3.1 Post VMware Workstation Pro upgrade activities

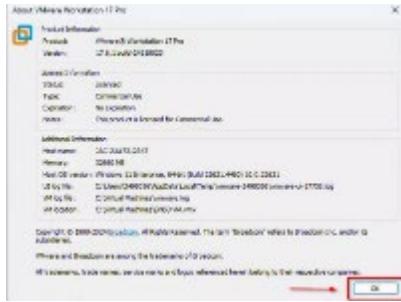
- A) When prompted “You must restart your system for the configuration changes made to VMware Workstation to take effect. Click “Yes” for restart or “No” if you plan to manually restart later.
- B) The virtual machines appear on the screen; virtual machines are not running.



- C) Verify the version of the VMWare workstation Select “Help” from the menu. A submenu will appear, select “About VMware Workstation.”



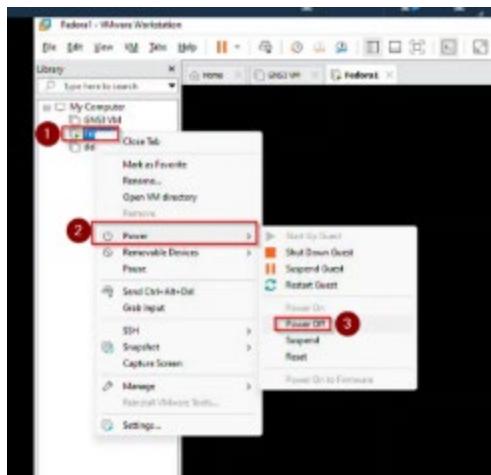
- D) The information about the installed software will pop up. Verify the latest version is installed. Click OK and you are ready to start the VMWare Workstation Pro.



2.4 Delete VM

A) Power off virtual machine if needed.

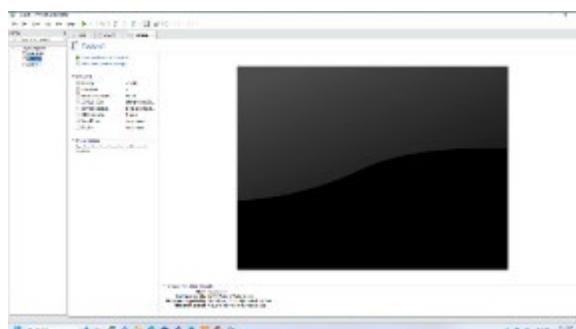
1. Select virtual machine to delete and right click to make submenu appear.
2. In the submenu select “Power>”
3. Submenu will appear select if machine is running “Power off”



4. A confirmation window will appear asking: Are you sure you want to power off the virtual machine <name>? Press “Power off” to continue with the process.



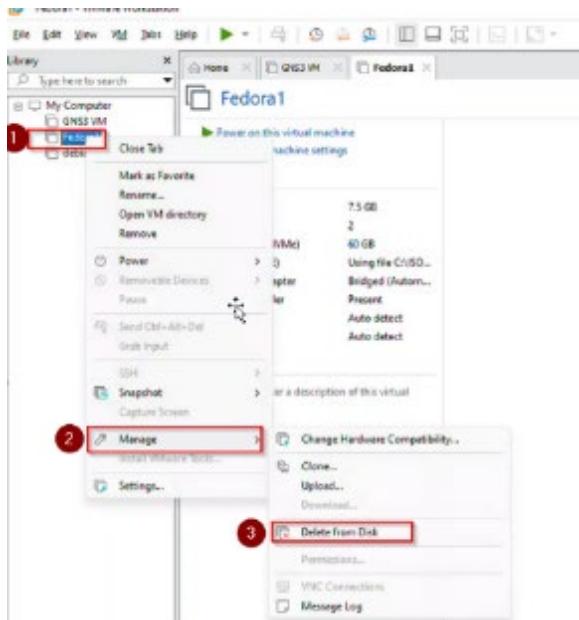
B) Verify virtual machine to be removed is turned off



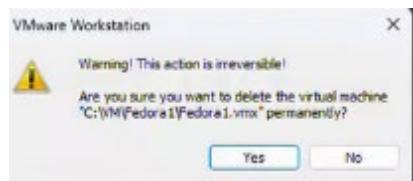
C) Delete form disk

1. Select virtual machine to delete and right click to make submenu appear.
2. In the submenu select “Manage”

3. Submenu will appear select if machine is running “Delete from Disk”

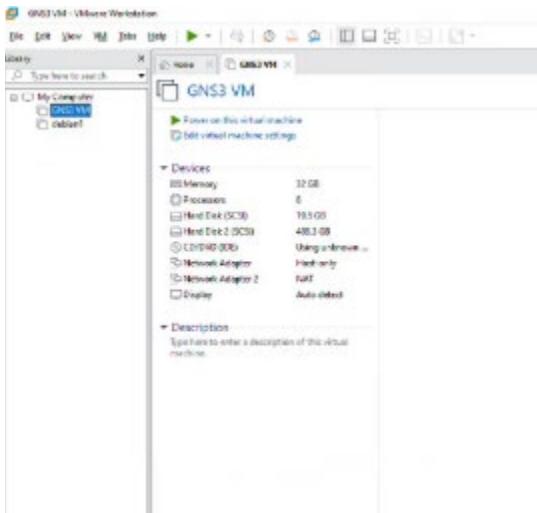


D) Confirm you want to delete VM Press “Yes”



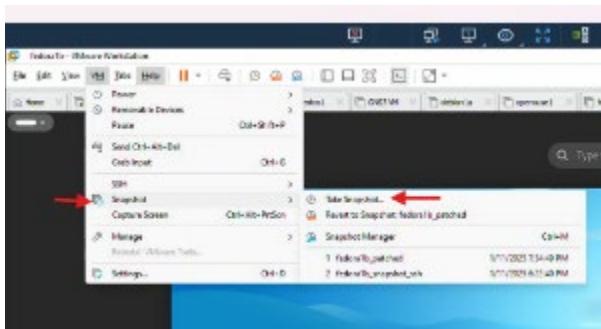
E) Immediately Deleted VM is removed from VMware Workstation Pro

Verify machine is removed from VMware Workstation Pro, VM does not appear on VMware Workstation Pro window.



2.5 Create a snapshot for VM

- A) From the VM you want tot take a snapshot select from Main menu “Snapshot” From Submenu Select “Take Snapshot”.



A window opens , give a name and a description to the snapshot. Press “Take Snapshot”



The process will start , It can take some time. Wait until; the snapshot finished to use the VM

The process per centage is seen at the bottom left of the VM



3 CCNA 3 Enterprise Networking, Security and Automation

3.1 CISCO ENSA - Commands & Procedures Overview

3.1.1 Module 2: Single-Area OSPFv2 Configuration

3.1.1.1 Configure a Router ID

```
R1(config)# router ospf 10
R1(config-router)# router-id n.n.n.n
R1(config-router)# end
R1# show ip protocols | include Router ID
```

3.1.1.2 Configure and verify a Loopback Interface as the Router ID

```
R1(config-if)# interface Loopback 1
R1(config-if)# ip address 1.1.1.1 255.255.255.255
R1(config-if)# end
R1# show ip protocols | include Router ID
    Router ID 1.1.1.1
R1#
```

3.1.1.3 Modify a Router ID

```
R1(config)# router ospf 10
R1(config-router)# router-id n.n.n.n
% OSPF: Reload or use "clear ip ospf process" command, for this to take effect
R1(config-router)# end
R1# clear ip ospf process
Reset ALL OSPF processes? [no]: y
R1# show ip protocols | include Router ID
```

3.1.1.4 Configure OSPF Using the network Command

```
Router(config-router)# network network-address wildcard-mask area
area-id
```

Example:

```
R1(config)# router ospf 10
R1(config-router)# network 10.10.1.0 0.0.0.255 area 0
R1(config-router)# network 10.1.1.4 0.0.0.3 area 0
R1(config-router)# network 10.1.1.12 0.0.0.3 area 0
R1(config-router)#
R1#
```

3.1.1.5 Configure OSPF Using the *ip ospf* Command

```
R1(config-router)# interface GigabitEthernet 0/0/0
R1(config-if)# ip ospf 10 area 0
R1(config-if)# interface GigabitEthernet 0/0/1
R1(config-if)# ip ospf 10 area 0
R1(config-if)# interface Loopback 0
R1(config-if)# ip ospf 10 area 0
R1(config-if)#

```

3.1.1.6 Configure and Verify Passive Interfaces

```
R1(config)# router ospf 10
R1(config-router)# passive-interface loopback 0
R1(config-router)# end
R1# show ip protocols
```

3.1.1.7 OSPF Point-to-Point Networks

Use the interface configuration command **ip ospf network point-to-point** on all interfaces where you want to disable the DR/BDR election process.

```
R1(config)# interface GigabitEthernet 0/0/0
R1(config-if)# ip ospf network point-to-point
```

Loopbacks and Point-to-Point Networks

Using loopbacks to simulate more networks than the equipment can support. By default, loopback interfaces are advertised as /32 host routes.

```
R1(config-if)# interface Loopback 0  
R1(config-if)# ip ospf network point-to-point
```

3.1.1.8 Verify the roles of the OSPFv2 router

```
R1# show ip ospf interface GigabitEthernet 0/0/0
```

3.1.1.9 Verify DR/BDR Adjacencies

```
R1# show ip ospf neighbor
```

3.1.1.10 Configure and verify OSPF Priority

```
R1(config)# interface GigabitEthernet 0/0/0
R1(config-if)# ip ospf priority [1-255]
R1(config-if)# end
R1#
R1# clear ip ospf process
Reset ALL OSPF processes? [no]: y
R1#
R1# show ip ospf interface GigabitEthernet 0/0/0
```

3.1.1.11 Remove OSPF

```
R1(config)# router ospf 10
R1(config-router)# no network 10.10.1.1 0.0.0.0 area 0
R1(config-router)# no network 10.1.1.5 0.0.0.0 area 0
R1(config-router)# no network 10.1.1.14 0.0.0.0 area 0
```

3.1.1.12 Adjust the Reference Bandwidth

```
R1# show ip ospf interface gigabitethernet0/0/0
```

```
R1(config) # router ospf 10  
R1(config-router) # auto-cost reference-bandwidth 10000
```

```
R1# show ip ospf interface gigabitethernet0/0/0
```

This command must be configured on every router in the OSPF domain. Notice that the value is expressed in Mbps; therefore, to adjust the costs for Gigabit Ethernet, use the command **auto-cost reference-bandwidth 1000**. For 10 Gigabit Ethernet, use the command **auto-cost reference-bandwidth 10000**.

To return to the default reference bandwidth, use the **auto-cost reference-bandwidth 100** command.

Note: The **auto-cost reference-bandwidth** command must be configured consistently on all routers in the OSPF domain to ensure accurate route calculations.

3.1.1.13 Verify Hello and Dead Intervals

3.1.1.14 Verify the currently configured OSPFv2 interface intervals:

```
R1# show ip ospf interface g0/0/0
```

3.1.1.15 To see the Dead Time counting down:

```
R1# show ip ospf neighbor
```

3.1.1.16 Modify and verify OSPFv2 Intervals

```
R1(config)# interface g0/0/0
Router(config-if)# ip ospf hello-interval seconds
Router(config-if)# ip ospf dead-interval seconds
R1# show ip ospf neighbor
```

Use the **no ip ospf hello-interval** and **no ip ospf dead-interval** commands to reset the intervals to their default.

Propagate and verify a Default Static Route in OSPFv2

```
R2(config)# interface lo1
R2(config-if)# ip address 64.100.0.1 255.255.255.252
R2(config-if)# exit
R2(config)# ip route 0.0.0.0 0.0.0.0 loopback 1
%Default route without gateway, if not a point-to-point interface, may impact performance
R2(config)# router ospf 10
R2(config-router)# default-information originate
R2(config-router)# end
R2#
R2# show ip route | begin Gateway
```

3.1.1.17 Verify OSPF Neighbors

Use the **show ip ospf neighbor** command to verify that the router has formed an adjacency with its neighboring routers. If the router ID of the neighboring router is not displayed, or if it does not show as being in

a state of FULL, the two routers have not formed an OSPFv2 adjacency.

```
R1# show ip ospf neighbor
```

3.1.1.18 Verify OSPF Protocol Settings

The **show ip protocols** command is a quick way to verify vital OSPF configuration information, which includes the OSPFv2 process ID, the router ID, interfaces explicitly configured to advertise OSPF routes, the neighbors the router is receiving updates from, and the default administrative distance, which is 110 for OSPF.

```
R1# show ip protocols
```

3.1.1.19 Verify OSPF Process Information

The **show ip ospf** command can also be used to examine the OSPFv2 process ID and router ID, as shown in the following command output. This command displays the OSPFv2 area information and the last time the SPF algorithm was executed.

```
R1# show ip ospf
```

3.1.1.20 Verify OSPF Interface Settings

The **show ip ospf interface** command provides a detailed list for every OSPFv2-enabled interface. Specify an interface to display the settings of just that interface, as shown in the following output for Gigabit Ethernet 0/0/0. This command shows the process ID, the local router ID, the type of network, OSPF cost, DR and BDR information on multiaccess links {not shown}, and adjacent neighbors.

```
R1# show ip ospf interface GigabitEthernet 0/0/0
```

Module 5: ACLs for IPv4 Configuration

3.1.2 Configure and Apply a Numbered Standard IPv4 ACL Syntax

```
R1(config)# access-list 10 remark ACE permits ONLY host 192.168.10.10 to the internet
R1(config)# access-list 10 permit host 192.168.10.10
R1(config)# do show access-lists
Standard IP access list 10
10 permit 192.168.10.10
```

```
R1(config)# access-list 10 remark ACE permits all host in LAN 2
R1(config)# access-list 10 permit 192.168.20.0 0.0.0.255
R1(config)# do show access-lists
```

```
Standard IP access list 10
10 permit 192.168.10.10
20 permit 192.168.20.0, wildcard bits 0.0.0.255
R1(config)#
```

Apply ACL 10 outbound on the Serial 0/1/0 interface.

```
R1(config)# interface Serial 0/1/0
R1(config-if)# ip access-group 10 out
R1(config-if)# end
R1#
```

The resulting policy of ACL 10 will only permit host 192.168.10.10 and all host from LAN 2 to exit the Serial 0/1/0 interface. All other hosts in the 192.168.10.0 network will not be permitted to the internet. Use the

show running-config command to review the ACL in the configuration, as shown in the output.

```
R1# show run | section access-list
access-list 10 remark ACE permits host 192.168.10.10 access-list 10 permit 192.168.10.10
access-list 10 remark ACE permits all host in LAN 2 access-list 10 permit 192.168.20.0 0.0.0.255
R1#
```

Finally, use the **show ip interface** command to verify if an interface has an ACL applied to it. In the example output, the output is specifically looking at the Serial 0/1/0 interface for lines that include "access list" text.

```
R1# show ip int Serial 0/1/0 | include access list
Outgoing Common access list is not set Outgoing access list is 10
Inbound Common access list is not set Inbound access list is not set
R1#
```

3.1.2.1 Configure and Apply a Named Standard IPv4 ACL Syntax

```
R1(config)# ip access-list standard PERMIT-ACCESS
R1(config-std-nacl)# remark ACE permits host 192.168.10.10
R1(config-std-nacl)# permit host 192.168.10.10
R1(config-std-nacl) #
```

Now add an ACE permitting only host 192.168.10.10 and another ACE permitting all LAN 2 hosts to the internet.

```
R1(config-std-nacl) # remark ACE permits host 192.168.10.10
R1(config-std-nacl) # permit host 192.168.10.10
R1(config-std-nacl) # remark ACE permits all hosts in LAN 2
R1(config-std-nacl) # permit 192.168.20.0 0.0.0.255
R1(config-std-nacl) # exit
R1(config) #
```

Apply the new named ACL outbound to the Serial 0/1/0 interface.

```
R1(config)# interface Serial 0/1/0
R1(config-if)# ip access-group PERMIT-ACCESS out
R1(config-if)# end
R1#
```

Use the **show access-lists** and **show running-config** command to review the ACL in the configuration, as shown in the output.

```
R1# show access-lists
Standard IP access list PERMIT-ACCESS
10 permit 192.168.10.10
20 permit 192.168.20.0, wildcard bits 0.0.0.255

R1# show run | section ip access-list
ip access-list standard PERMIT-ACCESS remark ACE permits host 192.168.10.10 permit 192.168.10.10
remark ACE permits all hosts in LAN 2 permit 192.168.20.0 0.0.0.255
R1#
```

Finally, use the **show ip interface** command to verify if an interface has an ACL applied to it. In the example output, the output is specifically looking at the Serial 0/1/0 interface for lines that include "access list" text.

```
R1# show ip int Serial 0/1/0 | include access list
Outgoing Common access list is not set Outgoing access list is PERMIT-ACCESS Inbound Common access list is not
set Inbound access list is not set
R1#
```

3.1.2.2 Modify an Numbered ACL

Text Editor Method

ACLs with multiple ACEs should be created in a text editor. This allows you to plan the required ACEs, create the ACL, and then paste it into the router interface. It also simplifies the tasks to edit and fix an ACL.

For example, assume ACL 1 was entered incorrectly using **19** instead of **192** for the first octet, as shown in the running configuration.

```
R1# show run | section access-list
access-list 1 deny 19.168.10.10
access-list 1 permit 192.168.10.0 0.0.0.255
R1#
```

In the example, the first ACE should have been to deny the host at 192.168.10.10. However, the ACE was incorrectly entered.

To correct the error:

- Copy the ACL from the running configuration and paste it into the text editor.
- Make the necessary edits changes.
- Remove the previously configured ACL on the router otherwise, pasting the edited ACL commands will only append {i.e., add} to the existing ACL ACEs on the router.
- Copy and paste the edited ACL back to the router.

Assume that ACL 1 has now been corrected. Therefore, the incorrect ACL must be deleted, and the corrected ACL 1 statements must be pasted in global configuration mode, as shown in the output.

```
R1 (config) # no access-list 1
R1 (config) #
R1 (config) # access-list 1 deny 192.168.10.10
R1 (config) # access-list 1 permit 192.168.10.0 0.0.0.255
R1 (config) #
```

3.1.2.3 Sequence Numbers Method

An ACL ACE can also be deleted or added using the ACL sequence numbers. Sequence numbers are automatically assigned when an ACE is entered. These numbers are listed in the show access-lists command. The show running-config command does not display sequence numbers.

In the previous example, the incorrect ACE for ACL 1 is using sequence number 10, as shown in the example.

```
R1# show access-lists
Standard IP access list 1
10 deny 19.168.10.10
20 permit 192.168.10.0, wildcard bits 0.0.0.255
R1#
```

Use the ip access-list standard command to edit an ACL. Statements cannot be overwritten using the same sequence number as an existing statement. Therefore, the current statement must be deleted first with the no 10 command. Then the correct ACE can be added using sequence number 10 is configured. Verify the changes using the show access-lists command, as shown in the example.

```
R1# conf t
R1 (config) # ip access-list standard 1
R1 (config-std-nacl) # no 10
R1 (config-std-nacl) # 10 deny host 192.168.10.10
R1 (config-std-nacl) # end

R1# show access-lists Standard IP access list 1
10 deny      192.168.10.10
20 permit 192.168.10.0, wildcard bits 0.0.0.255
R1#
```

3.1.2.4 Modify an Numbered ACL

```
R1# show access-lists
Standard IP access list NO-ACCESS
10 deny      192.168.10.10
20 permit 192.168.10.0, wildcard bits 0.0.0.255
R1# configure terminal
R1 (config) # ip access-list standard NO-ACCESS
R1 (config-std-nacl) # 15 deny 192.168.10.5
R1 (config-std-nacl) # end
R1#
R1# show access-lists
Standard IP access list NO-ACCESS
15 deny      192.168.10.5
10 deny      192.168.10.10
20 permit 192.168.10.0, wildcard bits 0.0.0.255
R1#
```

3.1.2.5 Clear ACL Statistics

```
R1# show access-lists
Standard IP access list NO-ACCESS
10 deny      192.168.10.10 (20 matches)
20 permit 192.168.10.0, wildcard bits 0.0.0.255 (64 matches)
R1# clear access-list counters NO-ACCESS

R1# show access-lists
Standard IP access list NO-ACCESS
10 deny      192.168.10.10
20 permit 192.168.10.0, wildcard bits 0.0.0.255
R1#
```

Secure VTY Ports with a Standard IPv4 ACL with the `access-class` command

ACLs typically filter incoming or outgoing traffic on an interface. However, an ACL can also be used to secure remote administrative access to a device using the vty lines.

Use the following two steps to secure remote administrative access to the vty lines:

- Create an ACL to identify which administrative hosts should be allowed remote access.
- Apply the ACL to incoming traffic on the vty lines.

Use the following command to apply an ACL to the vty lines:

```
R1 (config-line)# access-class {access-list-number | access-list-name}
{ in | out }
```

The **in** keyword is the most commonly used option to filter incoming vty traffic. The **out** parameter filters outgoing vty traffic and is rarely applied.

Secure VTY Access Example

```
R1(config)# username ADMIN secret class
R1(config)# ip access-list standard ADMIN-HOST
R1(config-std-nacl)# remark This ACL secures incoming vty lines
R1(config-std-nacl)# permit 192.168.10.10
R1(config-std-nacl)# deny any
R1(config-std-nacl)# exit
R1(config)# line vty 0 4
R1(config-line)# login local
R1(config-line)# transport input telnet
R1(config-line)# access-class ADMIN-HOST in
R1(config-line)# end
R1#
```

In a production environment, you would set the vty lines to only allow SSH, as shown in the example.

```
R1(config)# line vty 0 4
R1(config-line)# login local
R1(config-line)# transport input ssh
R1(config-line)# access-class ADMIN-HOST in
R1(config-line)# end
R1#
```

3.1.2.6 Numbered Extended IPv4 ACL Syntax

To create a numbered extended ACL, use the following global configuration command:

```
Router(config)# access-list access-list-number {deny | permit | remark text} protocol source source-wildcard [operator {port}] destination destination-wildcard [operator {port}] [established] [log]
```

Parameter	Description
<i>access-list-number</i>	<ul style="list-style-type: none"> This is the decimal number of the ACL. Extended ACL number range is 100 to 199 and 2000 to 2699.
deny	This denies access if the condition is matched.
permit	This permits access if the condition is matched.
remark <i>text</i>	<ul style="list-style-type: none"> (Optional) Adds a text entry for documentation purposes. Each remark is limited to 100 characters.
protocol	<ul style="list-style-type: none"> Name or number of an internet protocol. Common keywords include ip, tcp, udp, and icmp. The ip keyword matches all IP protocols.
source	<ul style="list-style-type: none"> This identifies the source network or host address to filter. Use the any keyword to specify all networks. Use the host <i>ip-address</i> keyword or simply enter an <i>ip-address</i> (without the host keyword) to identify a specific IP address.
source-wildcard	(Optional) A 32-bit wildcard mask that is applied to the source.
destination	<ul style="list-style-type: none"> This identifies the destination network or host address to filter. Use the any keyword to specify all networks. Use the host <i>ip-address</i> keyword or <i>ip-address</i>.

<i>destination-wildcard</i>	(Optional) This is a 32-bit wildcard mask that is applied to the destination.
<i>operator</i>	<ul style="list-style-type: none"> • (Optional) This compares source or destination ports. • Some operators include lt (less than), gt (greater than), eq (equal), and neq (not equal).
<i>port</i>	(Optional) The decimal number or name of a TCP or UDP port.
<i>established</i>	<ul style="list-style-type: none"> • (Optional) For the TCP protocol only. • This is a 1st generation firewall feature.
<i>log</i>	<ul style="list-style-type: none"> • (Optional) This keyword generates and sends an informational message whenever the ACE is matched. • This message includes ACL number, matched condition (i.e., permitted or denied), source address, and number of packets. • This message is generated for the first matched packet. • This keyword should only be implemented for troubleshooting or security reasons.

3.1.2.7 Apply an extended IPv4 ACL from an interface

The command to apply an extended IPv4 ACL to an interface is the same as the command used for standard IPv4 ACLs.

```
Router(config-if)# ip access-group {access-list-number | access-list-name} {in | out}
Remove an ACL from an interface
```

To remove an ACL from an interface, first enter the **no ip access-group** interface configuration command. To remove the ACL from the router, use the **no access-list** global configuration command.

3.1.2.8 Apply a Numbered Extended IPv4 ACL

```
R1(config)# access-list 110 permit tcp 192.168.10.0 0.0.0.255 any eq www
R1(config)# access-list 110 permit tcp 192.168.10.0 0.0.0.255 any eq
443
R1(config)# interface g0/0/0
R1(config-if)# ip access-group 110 in
R1(config-if)# exit
R1(config)#
```

TCP Established Extended ACL

The **established** keyword can be used to permit only the return HTTP traffic from requested websites, while denying all other traffic.

```
R1(config)# access-list 120 permit tcp any 192.168.10.0 0.0.0.255 established
R1(config)# interface g0/0/0
R1(config-if)# ip access-group 120 out
R1(config-if)# end
R1# show access-lists
Extended IP access list 110
10 permit tcp 192.168.10.0 0.0.0.255 any eq www
20 permit tcp 192.168.10.0 0.0.0.255 any eq 443 (657 matches)
Extended IP access list 120
10 permit tcp any 192.168.10.0 0.0.0.255 established (1166 matches)
R1#
```

3.1.2.9 Named Extended IPv4 ACL

```
R1(config)# ip access-list extended SURFING
R1(config-ext-nacl)# Remark Permits inside HTTP and HTTPS traffic
R1(config-ext-nacl)# permit tcp 192.168.10.0 0.0.0.255 any eq 80
R1(config-ext-nacl)# permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1(config-ext-nacl)# exit
R1(config)#
R1(config)# ip access-list extended BROWSING
R1(config-ext-nacl)# Remark Only permit returning HTTP and HTTPS traffic
R1(config-ext-nacl)# permit tcp any 192.168.10.0 0.0.0.255 established
R1(config-ext-nacl)# exit

R1(config)# interface g0/0/0
R1(config-if)# ip access-group SURFING in R1(config-if)# ip access-group BROWSING out R1(config-if)# end

R1# show access-lists
Extended IP access list SURFING
10 permit tcp 192.168.10.0 0.0.0.255 any eq www
20 permit tcp 192.168.10.0 0.0.0.255 any eq 443 (124 matches)
Extended IP access list BROWSING
10 permit tcp any 192.168.10.0 0.0.0.255 established (369 matches)
R1#
```

3.1.2.10 Edit Named Extended ACLs

```
R1# show access-lists
Extended IP access list BROWSING
10 permit tcp any 192.168.10.0 0.0.0.255 established
Extended IP access list SURFING
10 permit tcp 19.168.10.0 0.0.0.255 any eq www
20 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1#
R1# configure terminal
R1(config)# ip access-list extended SURFING
R1(config-ext-nacl)# no 10
R1(config-ext-nacl)# 10 permit tcp 192.168.10.0 0.0.0.255 any eq www
R1(config-ext-nacl)# end
```

The output verifies the configuration change using the **show access-lists** command.

```
R1# show access-lists
Extended IP access list BROWSING
10 permit tcp any 192.168.10.0 0.0.0.255 established
Extended IP access list SURFING
10 permit tcp 192.168.10.0 0.0.0.255 any eq www
20 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1#
```

3.1.2.11 Verify Extended ACLs

show ip interface

The **show ip interface** command is used to verify the ACL on the interface and the direction in which it was applied, as shown in the output.

```
R1# show ip interface g0/0/0 | include access list
```

show access-lists

The **show access-lists** command can be used to confirm that the ACLs work as expected. The command displays statistic counters that increase whenever an ACE is matched.

```
R1# show access-lists
```

show running-confg

The **show running-config** command can be used to validate what was configured. The command also displays configured remarks.

The command can be filtered to display only pertinent information, as shown in the following.

```
R1# show running-config | begin ip access-list
```

3.1.3 Module 6: NAT for IPv4

3.1.3.1 Configure Static NAT

Step 1. The first task is to create a mapping between the inside local address and the inside global addresses.

```
R2(config)# ip nat inside source static inside-local-address inside- global-address
R2(config)# ip nat inside source static 192.168.10.254 209.165.201.5
```

Step 2. Configure interfaces participating in the translation as inside or outside relative to NAT.

```
R2(config)# interface serial 0/1/0
R2(config-if)# ip address 192.168.1.2 255.255.255.252
R2(config-if)# ip nat inside
R2(config-if)# exit
R2(config)# interface serial 0/1/1
R2(config-if)# ip address 209.165.200.1 255.255.255.252
R2(config-if)# ip nat outside
```

3.1.3.2 Verify Static NAT

show ip nat translations

To verify NAT operation, issue the **show ip nat translations** command. This command shows active NAT translations.

R2# **show ip nat translations**

show ip nat statistics

The **show ip nat statistics** command displays information about the total number of active translations, NAT configuration parameters, the number of addresses in the pool, and the number of addresses that have been allocated.

To verify that the NAT translation is working, it is best to clear statistics from any past translations using the **clear ip nat statistics** command before testing.

```
R2# clear ip nat statistics
R2# show ip nat statistics
```

3.1.3.3 Configure Dynamic NAT

Step 1

Define the pool of addresses that will be used for translation using the **ip nat pool** command. This pool of addresses is typically a group of public addresses. The addresses are defined by indicating the starting IPv4 address and the ending IPv4 address of the pool. The **netmask** or **prefix-length** keyword indicates which address bits belong to the network and which bits belong to the host for that range of addresses.

In the scenario, define a pool of public IPv4 addresses under the pool name NAT-POOL1.

```
R2(config)# ip nat pool NAT-POOL1 209.165.200.226 209.165.200.240
netmask 255.255.255.224
```

Step 2

Configure a standard ACL to identify {permit} only those addresses that are to be translated. An ACL that is too permissive can lead to unpredictable results. Remember there is an implicit **deny all** statement at the end of each ACL.

In the scenario, define which addresses are eligible to be translated.

```
R2(config)# access-list 1 permit 192.168.0.0 0.0.255.255
```

Step 3

Bind the ACL to the pool, using the following command syntax:

```
Router{config)# ip nat inside source list {access-list-number | access-list-name} pool pool-name
```

This configuration is used by the router to identify which devices **{list}** receive which addresses **{pool}**. In the scenario, bind NAT-POOL1 with ACL 1.

```
R2(config)# ip nat inside source list 1 pool NAT-POOL1
```

Step 4

Identify which interfaces are inside, in relation to NAT; this will be any interface that connects to the inside network.

In the scenario, identify interface serial 0/1/0 as an inside NAT interface.

```
R2(config)# interface serial 0/1/0
R2(config-if)# ip nat inside
```

Step 5

Identify which interfaces are outside, in relation to NAT; this will be any interface that connects to the outside network.

In the scenario, identify interface serial 0/1/1 as the outside NAT interface.

```
R2(config)# interface serial 0/1/1
R2(config-if)# ip nat outside
```

3.1.3.4 Verify Dynamic NAT

show ip nat translations

The output of the **show ip nat translations** command displays all static translations that have been configured and any dynamic translations that have been created by traffic.

```
R2# show ip nat translations
```

Adding the **verbose** keyword displays additional information about each translation, including how long ago the entry was created and used.

```
R2# show ip nat translation verbose
```

By default, translation entries time out after 24 hours, unless the timers have been reconfigured with the **ip nat translation timeout *timeout-seconds*** command in global configuration mode.

To clear dynamic entries before the timeout has expired, use the **clear ip nat translation** privileged EXEC mode command as shown.

```
R2# clear ip nat translation *
R2# show ip nat translation
```

Command	Description
clear ip nat translation *	Clears all dynamic address translation entries from the NAT translation table.
clear ip nat translation insideglobal-ip local-ip [outside local-ip global-ip]	Clears a simple dynamic translation entry containing an inside translation or both inside and outside translation.
clear ip nat translation protocolinsideglobal-ip global-port local-ip local-port [outsidelocal-ip local-port global-ip global-port]	Clears an extended dynamic translation entry.

show ip nat statistics

The **show ip nat statistics** command displays information about the total number of active translations, NAT configuration parameters, the number of addresses in the pool, and how many of the addresses have been allocated.

```
R2# show ip nat statistics
```

show running-confg

show running-config command and look for NAT, ACL, interface, or pool commands with the required values. Examine these carefully and correct any errors discovered. The example shows the NAT pool configuration.

```
R2# show running-config | include NAT
```

3.1.3.5 Configure PAT to Use a Single IPv4 Address

```
R2(config)# ip nat inside source list 1 interface serial 0/1/1 overload
R2(config)# access-list 1 permit 192.168.0.0 0.0.255.255
R2(config)# interface serial0/1/0 R2(config-if)# ip nat inside R2(config-if)#
exit
R2(config)# interface Serial0/1/1
R2(config-if)# ip nat outside
```

3.1.3.6 Configure PAT to Use an Address Pool

```
R2(config)# ip nat pool NAT-POOL2 209.165.200.226 209.165.200.240
netmask 255.255.255.224
R2(config)# access-list 1 permit 192.168.0.0 0.0.255.255
R2(config)# ip nat inside source list 1 pool NAT-POOL2 overload
R2(config)#
R2(config)# interface serial0/1/0
```

3.1.3.7 Configure PAT to Use an Address Pool

```
R2(config)# ip nat pool NAT-POOL2 209.165.200.226 209.165.200.240
netmask 255.255.255.224
R2(config)# access-list 1 permit 192.168.0.0 0.0.255.255
R2(config)# ip nat inside source list 1 pool NAT-POOL2 overload
R2(config)#
R2(config)# interface serial0/1/0
1341
R2(config-if)# ip nat inside
R2(config-if)# exit
R2(config)# interface serial0/1/1
R2(config-if)# ip nat outside
R2(config-if)# end
R2#
6.6 - Verify PAT
show ip nat translations
R2# show ip nat translations
show ip nat statistics
R2# show ip nat statistics
```

3.2 Packet tracer exercises

3.2.1 Exercise 2.2.13 - Packet Tracer - Point-to-Point Single-Area OSPFv2 Configuration

3.2.1.1 Addressing Table

Device	Interface	IP Address	Subnet Mask
R1	G0/0/0	192.168.10.1	/24
	S0/1/0	10.1.1.1	/30
	S0/1/1	10.1.1.5	/30
R2	G0/0/0	192.168.20.1	/24
	S0/1/0	10.1.1.2	/30
	S0/1/1	10.1.1.9	/30
R3	G0/0/0	192.168.30.1	/24
	S0/1/0	10.1.1.10	/30
	S0/1/1	10.1.1.6	/30
PC1	NIC	192.168.10.10	/24
PC2	NIC	192.168.20.10	/24
PC3	NIC	192.168.30.10	/24

3.2.1.2 Objectives

Part 1: Configure Router IDs.

Part 2: Configure Networks for OSPF Routing. Part 3:

Configure Passive Interfaces.

Part 4: Verify OSPF configuration.

3.2.1.3 Background

In this activity, you will activate OSPF routing using network statements and wildcard masks, configuring OSPF routing on interfaces, and by using network statements quad-zero masks. In addition, you will configure explicit router IDs and passive interfaces.

3.2.1.4 Instructions

Part 1: Configure router IDs.

- Start the OSPF routing process on all three routers. Use process ID **10**.

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

- Use the **router-id** command to set the OSPF IDs of the three routers as follows

- R1: **1.1.1.1**

- R2: **2.2.2.2**
- R3: **3.3.3.3**

Use the following command:

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

```
!! R1
enable
config t
router ospf 10
router-id 1.1.1.1
```

```
!! R2
enable
config t
router ospf 10
router-id 2.2.2.2
```

```
!! R3
enable
config t
router ospf 10
router-id 3.3.3.3
```

Part 2: Configure Networks for OSPF Routing

Step 1: Configure networks for OSPF routing using network commands and wildcard masks.

How many statements are required to configure OSPF to route all the networks attached to router R1? **Answer 3**

The LAN attached to router R1 has a /24 mask. What is the equivalent of this mask in dotted decimal representation? **Answer 255.255.255.0**

Subtract the dotted decimal subnet mask from 255.255.255.255. What is the result? **Answer 0.0.0.255** (wildcard mask)

What is the dotted decimal equivalent of the /30 subnet mask? **Answer 255.255.255.252**

Subtract the dotted decimal representation of the /30 mask from 255.255.255.255. What is the result? **Answer 0.0.0.253**

- Configure the routing process on **R1** with the network statements and wildcard masks that are required to activate OSPF routing for all the attached networks. The network statement values should be the network or subnet addresses of the configured networks.

```
Router(config-router) # network network-address wildcard-mask area area-id
```

```
!! R1
network 192.168.10.0 0.0.0.255 area 0
```

```
network 10.1.1.0 0.0.0.3 area 0  
network 10.1.1.4 0.0.0.3 area 0
```

- b. Verify that OSPF has been configured properly by displaying the running configuration. If you find an error, delete the network statement using the **no** command and reconfigure it.

```
show running-configuration | section rip
```

Step 2: Configure networks for OSPF routing using interface IP addresses and quad-zero masks.

On router R2, configure OSPF using network commands with the IP addresses of the interfaces and quad-zero masks. The syntax of the network command is the same as was used above.

```
!! R2  
network 192.168.20.1 0.0.0.0 area 0  
network 10.1.1.2 0.0.0.0 area 0  
network 10.1.1.9 0.0.0.0 area 0
```

Step 3: Configure OSPF routing on router interfaces

On router R3, configure the required interfaces with OSPF.

Which interfaces on R3 should be configured with OSPF?

Answer G0/0/0, S0/1/0, S0/1/1

Configure each interface using the command syntax shown below:

```
Router(config-if)# ip ospf process-id area area-id
```

```
!!! R3  
interface GigabitEthernet0/0/0  
ip ospf 10 area 0  
interface Serial0/1/0  
ip ospf 10 area 0  
interface Serial0/1/1  
ip ospf 10 area 0
```

Part 3: Configure Passive Interfaces

OSPF will send its protocol traffic out of all interfaces that are participating in the OSPF process. On links that are not configured to other networks, such as LANs, this unnecessary traffic consumes resources. The passive-interface command will prevent the OSPF process from sending unnecessary routing protocol traffic out LAN interfaces.

Which interfaces on R1, R2, and R3 are LAN interfaces? **Answer** G0/0/0 on all three routers.

Configure the OSPF process on each of the three routers with the **passive-interface** command.

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

!! R1

```
R1(config)# router ospf 10  
R1(config-router)# passive-interface GigabitEthernet0/0/0
```

!! R2

```
router ospf 10  
passive-interface GigabitEthernet0/0/0
```

!! R3

```
router ospf 10  
passive-interface GigabitEthernet0/0/0
```

Part 4: Verify OSPF Configuration

Use **show** commands to verify the network and passive interface configuration of the OSPF process on each router.

!R1

```
show ip route  
show ip route ospf  
show ip ospf neighbor  
show ip ospf database  
show ip ospf interface  
show ip ospf  
! Verify passive interface  
show ip protocols
```

!R2

```
show ip route  
show ip route ospf  
show ip ospf neighbor  
show ip ospf database  
show ip ospf interface  
show ip ospf  
! Verify passive interface  
show ip protocols
```

!R3

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

```
show ip ospf interface  
show ip ospf  
! Verify passive interface  
show ip protocols
```

3.2.1.5 Summary solution

3.2.1.5.1 Scripts

```
!! Router R1  
enable  
configure terminal  
router ospf 10  
router-id 1.1.1.1  
network 192.168.10.1 0.0.0.255 area 0  
network 10.1.1.0 0.0.0.3 area 0  
network 10.1.1.4 0.0.0.3 area 0  
passive-interface g0/0/0  
end
```

```
!! Router R2  
enable  
configure terminal  
router ospf 10  
router-id 2.2.2.2  
network 192.168.20.1 0.0.0.0 area 0  
network 10.1.1.2 0.0.0.0 area 0  
network 10.1.1.9 0.0.0.0 area 0  
passive-interface g0/0/0  
end
```

```
!! Router R3  
enable  
configure terminal  
router ospf 10  
router-id 3.3.3.3  
interface GigabitEthernet0/0/0  
ip ospf 10 area 0  
interface Serial0/1/0  
ip ospf 10 area 0  
interface Serial0/1/1  
ip ospf 10 area 0  
router ospf 10  
passive-interface g0/0/0  
end
```

R1

```

R1>
R1>enable
R1>config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#router ospf 10
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 192.168.10.0 0.0.0.255 area 0
R1(config-router)#network 10.1.1.0 0.0.0.3 area 0
R1(config-router)#network 10.1.1.4 0.0.0.3 area 0
R1(config-router)#end
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1copy ru
R1copy running-config st
R1copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
00:08:18: %OSPF-5-ADJCHG: Process 10, Nbr 2.2.2.2 on Serial0/1/0 from LOADING to FULL, Loading Done
00:11:00: %OSPF-5-ADJCHG: Process 10, Nbr 3.3.3.3 on Serial0/1/1 from LOADING to FULL, Loading Done
R1#

```

R2

```

R2(config)#router ospf 10
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 192.168.20.1 0.0.0.0 area 0
R2(config-router)#network 10.1.1.2 0.0.0.0 area 0
R2(config-router)#network 10.1.1.9 0.0.0.0 area 0
R2(config-router)#passive-interface g0/0/0
^
% Invalid input detected at '^' marker.

R2(config-router)#passive-interface g0/0/0
R2(config-router)#end
R2#
%SYS-5-CONFIG_I: Configured from console by console

```

R3

```

R3(config)#router ospf 10
R3(config-router)#router-id 3.3.3.3
R3(config-router)]interface
R3(config-router)interface GigabitEthernet0/0/0
R3(config-if)#ip ospf 10 area 0
R3(config-if)]interface Serial0/1/0
R3(config-if)ip ospf 10 area 0
R3(config-if)#
R3(config-if)#
00:10:00: %OSPF-5-ADJCHG: Process 10, Nbr 2.2.2.2 on Serial0/1/0 from LOADING to FULL, Loading Done
R3(config-if)interface Serial0/1/1
R3(config-if)ip ospf 10 area 0
R3(config-if)#
00:11:00: %OSPF-5-ADJCHG: Process 10, Nbr 1.1.1.1 on serial0/1/1 from LOADING to FULL, Loading Done
router ospf 10
R3(config-router)#passive-interface g0/0/0
R3(config-router)#end
R3#
%SYS-5-CONFIG_I: Configured from console by console
R3copy run

```

3.2.1.5.2 Printouts

Configuration printouts

!R1

```

show ip route
show ip route ospf
show ip ospf neighbor
show ip ospf database

```

```

show ip ospf interface
show ip ospf
! Verify passive interface
show ip protocols

```

```

!R2
show ip route
show ip route ospf
show ip ospf neighbor
show ip ospf database
show ip ospf interface
show ip ospf
! Verify passive interface
show ip protocols

```

```

!R3
show ip route
show ip route ospf
show ip ospf neighbor
show ip ospf database
show ip ospf interface
show ip ospf
! Verify passive interface
show ip protocols

```

R1

```

R1#show ip interface brief
Interface          IP-Address      OK? Method Status      Protocol
GigabitEthernet0/0/0 192.168.10.1   YES manual up       up
GigabitEthernet0/0/1 unassigned      YES unset administratively down down
Serial0/1/0         10.1.1.1       YES manual up       up
Serial0/1/1         10.1.1.5       YES manual up       up
Serial0/2/0         unassigned     YES unset down     down
Serial0/2/1         unassigned     YES unset administratively down down
Vlan1              unassigned     YES unset administratively down down
R1#
R1#show ip route
Codes: L - Local, 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, # - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

 10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
C    10.1.1.0/16 is directly connected, Serial0/1/0
L    10.1.1.1/32 is directly connected, Serial0/1/0
C    10.1.1.4/30 is directly connected, Serial0/1/1
E    10.1.1.5/32 is directly connected, Serial0/1/1
O    10.1.1.8/30 [110/128] via 10.1.1.2, 00:01:44, Serial0/1/0
      [110/126] via 10.1.1.6, 00:01:44, Serial0/1/1
 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    192.168.10.1/32 is directly connected, GigabitEthernet0/0/0
O    192.168.30.0/24 [110/65] via 10.1.1.6, 00:01:44, Serial0/1/1

```

```

R1#show ip route ospf
  10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
  0    10.1.1.0 [110/120] via 10.1.1.2, 00:02:12, Serial10/1/3
        [110/138] via 10.1.1.6, 00:02:12, Serial10/1/1
  0    192.168.10.0 [110/65] via 10.1.1.6, 00:02:12, Serial10/1/1

R1#show ospf neighbor
R1#show ospf neighbor
  *
  * Invalid input detected at '^' marker.

R1#show ip ospf neighbor

Neighbor ID      Pri   State          Dead Time     Address           Interface
3.3.3.3          0      FULLY-STATEFUL 00:00:30    10.1.1.6       Serial10/1/1
2.2.2.2          0      FULLY-STATEFUL 00:00:34    10.1.1.2       Serial10/1/0
R1#show ip ospf database
  OSPF Router with ID (1.1.1.1) (Process ID 10)
    Router Link States (Area 0)

Link ID          ADV Router      Age      Seq#      Checksum Link count
2.2.2.2          2.2.2.2        193      0x00003004 0x000700 4
1.1.1.1          1.1.1.1        178      0x80003005 0x00a648 5
3.3.3.3          3.3.3.3        179      0x00003005 0x000839 5
R1#

```

```

R1#show ip ospf interface
  GigabitEthernet0/0/0 is up, line protocol is up
    Internet address is 192.168.10.1/24, Area 0
    Process ID 10, Router ID 1.1.1.1, Network Type BROADCAST, Cost: 1
    Transmit Delay is 1 sec, Slave DR, Priority 1
    Designated Router (ID) 1.1.1.1, Interface address 192.168.10.1
    No backup designated router on this network
    Timer intervals configured, Hello 10, Dead 40, Retransmit 5
      Hello due in 00:00:06
    Index 1/1, flood queue length 0
    Next 0x0(0)/0x0(0)
    Last flood scan length is 1, maximum is 1
    Last flood scan time is 0 msec, maximum is 0 msec
    Neighbor Count is 0, Adjacent neighbor count is 0
    Suppress hello for 0 neighbor(s)
R1#show ip ospf
  Routing Process "ospf 10" with ID 1.1.1.1
  Supports only single TOS(TOS0) routes
  Supports opaque LSA
  SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
  Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
  Number of external LSA 0. Checksum Sum 0x00000000
  Number of opaque AS LSA 0. Checksum Sum 0x00000000
  Number of DCbitless external and opaque AS LSA 0
  Number of DoNotAge external and opaque AS LSA 0
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  External flood list length 0
    Area BACKBONE(0)
      Number of interfaces in this area is 3
      Area has no authentication
      SPF algorithm executed 7 times
      Area ranges are
        Number of LSA 3. Checksum Sum 0x01b97d
        Number of opaque link LSA 0. Checksum Sum 0x00000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
      Flood list length 0

```

```

%SYS-5-CONFIG_I: Configured from console by console

R1#show ip protocols

Routing Protocol is "ospf 10"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 1.1.1.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    192.168.10.0 0.0.0.255 area 0
    10.1.1.0 0.0.0.3 area 0
    10.1.1.4 0.0.0.3 area 0
  Passive Interface(s):
    GigabitEthernet0/0/0
  Routing Information Sources:
    Gateway          Distance      Last Update
    1.1.1.1           110          00:14:03
    2.2.2.2           110          00:14:24
    3.3.3.3           110          00:14:03
  Distance: (default is 110)

R1#

```

R2

```

R2#show ip interface brief
Interface          IP-Address      OK? Method Status      Proto
GigabitEthernet0/0/0 192.168.20.1  YES manual up       up
GigabitEthernet0/0/1 unassigned     YES unset administratively down down
Serial0/1/0         10.1.1.2       YES manual up       up
Serial0/1/1         10.1.1.9       YES manual up       up
Vlan1              unassigned     YES unset administratively down down
R2#

```

```

R2#
R2#show ip route
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

      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
C        10.1.1.0/30 is directly connected, Serial0/1/0
L        10.1.1.2/32 is directly connected, Serial0/1/0
O        10.1.1.4/30 [110/128] via 10.1.1.1, 00:25:15, Serial0/1/0
                  [110/128] via 10.1.1.10, 00:25:15, Serial0/1/1
C        10.1.1.8/30 is directly connected, Serial0/1/1
L        10.1.1.9/32 is directly connected, Serial0/1/1
O        192.168.10.0/24 [110/65] via 10.1.1.1, 00:27:55, Serial0/1/0
          192.168.20.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.20.0/24 is directly connected, GigabitEthernet0/0/0
L        192.168.20.1/32 is directly connected, GigabitEthernet0/0/0
O        192.168.30.0/24 [110/65] via 10.1.1.10, 00:25:37, Serial0/1/1

R2#show ip route opf
Translating "opf"...domain server (255.255.255.255)
% Invalid input detected

R2#show ip route ospf
      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
O        10.1.1.4 [110/128] via 10.1.1.1, 00:25:26, Serial0/1/0
                  [110/128] via 10.1.1.10, 00:25:26, Serial0/1/1
O        192.168.10.0 [110/65] via 10.1.1.1, 00:28:06, Serial0/1/0
O        192.168.30.0 [110/65] via 10.1.1.10, 00:25:48, Serial0/1/1

```

IOS Command Line Interface

R2#show ip ospf neighbor						
Neighbor ID	Pri	State	Dead Time	Address	Interface	
1.1.1.1	0	FULL/ -	00:00:36	10.1.1.1	Serial0/1/1	
3.3.3.3	0	FULL/ -	00:00:36	10.1.1.10	Serial0/1/1	

R2#show ip ospf data
R2#show ip ospf database
 OSPF Router with ID (2.2.2.2) (Process ID 10)

Router Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum	Link count
1.1.1.1	1.1.1.1	1629	0x80000005	0x00a6a8	5
3.3.3.3	3.3.3.3	1629	0x80000005	0x008b95	5
2.2.2.2	2.2.2.2	579	0x80000005	0x006acd	5

R2#show ip ospf interface

```
R2#show ip ospf interface
```

```
Serial0/1/0 is up, line protocol is up
  Internet address is 10.1.1.2/30, Area 0
  Process ID 10, Router ID 2.2.2.2, Network Type POINT-TO-POINT, Cost: 64
  Transmit Delay is 1 sec, State POINT-TO-POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:07
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1 , Adjacent neighbor count is 1
    Adjacent with neighbor 1.1.1.1
  Suppress hello for 0 neighbor(s)
Serial0/1/1 is up, line protocol is up
  Internet address is 10.1.1.9/30, Area 0
  Process ID 10, Router ID 2.2.2.2, Network Type POINT-TO-POINT, Cost: 64
  Transmit Delay is 1 sec, State POINT-TO-POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:05
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1 , Adjacent neighbor count is 1
    Adjacent with neighbor 3.3.3.3
  Suppress hello for 0 neighbor(s)
GigabitEthernet0/0/0 is up, line protocol is up
  Internet address is 192.168.20.1/24, Area 0
  Process ID 10, Router ID 2.2.2.2, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 2.2.2.2, Interface address 192.168.20.1
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    No Hellos (Passive interface)
  Index 3/3, flood queue length 0
  Next 0x0(0)/0x0(0)
```

```
R2#show ip ospf
  Routing Process "ospf 10" with ID 2.2.2.2
    Supports only single TOS(TOS0) routes
    Supports opaque LSA
    SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
    Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
    Number of external LSA 0. Checksum Sum 0x000000
    Number of opaque AS LSA 0. Checksum Sum 0x000000
    Number of DCbitless external and opaque AS LSA 0
    Number of DoNotAge external and opaque AS LSA 0
    Number of areas in this router is 1. 1 normal 0 stub 0 nssa
    External flood list length 0
      Area BACKBONE(0)
        Number of interfaces in this area is 3
        Area has no authentication
        SPF algorithm executed 5 times
        Area ranges are
        Number of LSA 3. Checksum Sum 0x019d0a
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

R2#

```
R2#show ip protocols
  Routing Protocol is "ospf 10"
    Outgoing update filter list for all interfaces is not set
    Incoming update filter list for all interfaces is not set
    Router ID 2.2.2.2
    Number of areas in this router is 1. 1 normal 0 stub 0 nssa
    Maximum path: 4
    Routing for Networks:
      10.1.1.2 0.0.0.0 area 0
      10.1.1.9 0.0.0.0 area 0
      192.168.20.1 0.0.0.0 area 0
    Passive Interface(s):
      GigabitEthernet0/0/0
    Routing Information Sources:
      Gateway          Distance      Last Update
      1.1.1.1           110          00:00:43
      2.2.2.2           110          00:13:15
      3.3.3.3           110          00:00:44
    Distance: (default is 110)
```

R2#

R3

```

R3#show ip interface brief
Interface          IP-Address      OK? Method Status      Protocol
GigabitEthernet0/0/0 192.168.30.1   YES manual up       up
GigabitEthernet0/0/1 unassigned     YES unset administratively down down
Serial0/1/0         10.1.1.10     YES manual up       up
Serial0/1/1         10.1.1.6      YES manual up       up
Vlan1              unassigned     YES unset administratively down down
R3#
R3#

```

```

R3>
R3>show ip route
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 a
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
O        10.1.1.0/30 [110/128] via 10.1.1.9, 00:34:08, Serial0/1/0
                  [110/128] via 10.1.1.5, 00:34:08, Serial0/1/1
C        10.1.1.4/30 is directly connected, Serial0/1/1
L        10.1.1.6/32 is directly connected, Serial0/1/1
C        10.1.1.8/30 is directly connected, Serial0/1/0
L        10.1.1.10/32 is directly connected, Serial0/1/0
O        192.168.10.0/24 [110/65] via 10.1.1.5, 00:34:08, Serial0/1/1
O        192.168.20.0/24 [110/65] via 10.1.1.9, 00:16:33, Serial0/1/0
                  192.168.30.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.30.0/24 is directly connected, GigabitEthernet0/0/0
L        192.168.30.1/32 is directly connected, GigabitEthernet0/0/0

R3>show ip route ospf
      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
O        10.1.1.0 [110/128] via 10.1.1.9, 00:34:15, Serial0/1/0
                  [110/128] via 10.1.1.5, 00:34:15, Serial0/1/1
O        192.168.10.0 [110/65] via 10.1.1.5, 00:34:15, Serial0/1/1
O        192.168.20.0 [110/65] via 10.1.1.9, 00:16:40, Serial0/1/0

```

```
R3>show ip ospf nei  
R3>show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
1.1.1.1	0	FULL/ -	00:00:32	10.1.1.5	Serial0/1/1
2.2.2.2	0	FULL/ -	00:00:30	10.1.1.9	Serial0/1/0

```
R3>show ip ospf database  
      OSPF Router with ID (3.3.3.3) (Process ID 10)  
  
      Router Link States (Area 0)  
  
      Link ID          ADV Router      Age           Seq#      Checksum Link count  
3.3.3.3            3.3.3.3        273          0x80000006 0x008996 5  
2.2.2.2            2.2.2.2        1025         0x80000005 0x006acd 5  
1.1.1.1            1.1.1.1        273          0x80000006 0x00a4a9 5  
no show in ospf interface
```

```
1.1.1.1          1.1.1.1        273          0x80000006 0x00a4a9 5  
R3>show ip ospf interface
```

```
GigabitEthernet0/0/0 is up, line protocol is up  
  Internet address is 192.168.30.1/24, Area 0  
  Process ID 10, Router ID 3.3.3.3, Network Type BROADCAST, Cost: 1  
  Transmit Delay is 1 sec, State WAITING, Priority 1  
  No designated router on this network  
  No backup designated router on this network  
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5  
    No Hellos (Passive interface)  
  Index 1/1, flood queue length 0  
  Next 0x0(0)/0x0(0)  
  Last flood scan length is 1, maximum is 1  
  Last flood scan time is 0 msec, maximum is 0 msec  
  Neighbor Count is 0, Adjacent neighbor count is 0  
  Suppress hello for 0 neighbor(s)  
Serial0/1/0 is up, line protocol is up  
  Internet address is 10.1.1.10/30, Area 0  
  Process ID 10, Router ID 3.3.3.3, Network Type POINT-TO-POINT, Cost: 64  
  Transmit Delay is 1 sec, State POINT-TO-POINT,  
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5  
    Hello due in 00:00:07  
  Index 2/2, flood queue length 0  
  Next 0x0(0)/0x0(0)  
  Last flood scan length is 1, maximum is 1  
  Last flood scan time is 0 msec, maximum is 0 msec  
  Neighbor Count is 1, Adjacent neighbor count is 1  
    Adjacent with neighbor 2.2.2.2  
  Suppress hello for 0 neighbor(s)  
Serial0/1/1 is up, line protocol is up  
  Internet address is 10.1.1.6/30, Area 0  
  Process ID 10, Router ID 3.3.3.3, Network Type POINT-TO-POINT, Cost: 64  
  Transmit Delay is 1 sec, State POINT-TO-POINT,  
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5  
    Hello due in 00:00:06  
  Index 3/3, flood queue length 0  
  Next 0x0(0)/0x0(0)
```

```
    suppress neigbor for v neighbor(s)
R3>show ip ospf
  Routing Process "ospf 10" with ID 3.3.3.3
  Supports only single TOS(TOS0) routes
  Supports opaque LSA
  SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
  Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
  Number of external LSA 0. Checksum Sum 0x000000
  Number of opaque AS LSA 0. Checksum Sum 0x000000
  Number of DCbitless external and opaque AS LSA 0
  Number of DoNotAge external and opaque AS LSA 0
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  External flood list length 0

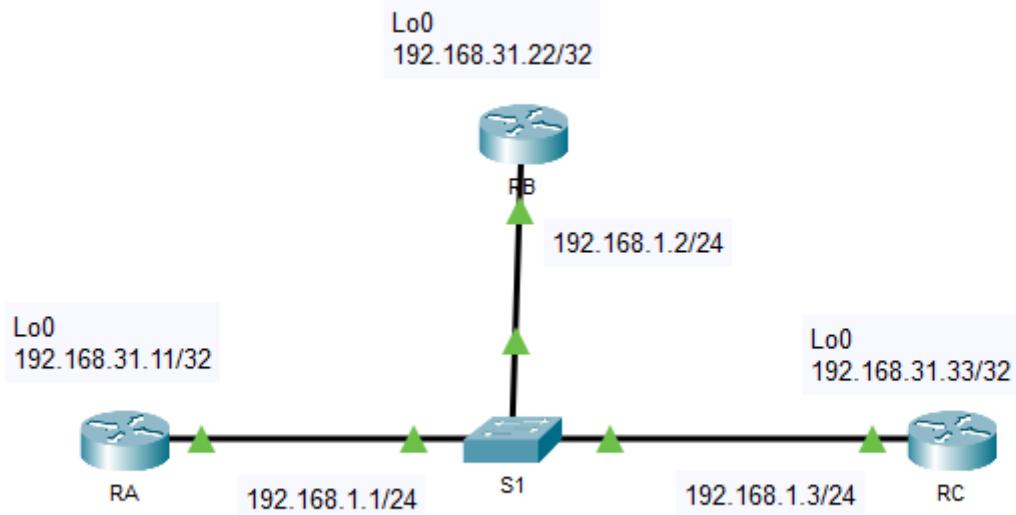
R3>show ip protocols

  Routing Protocol is "ospf 10"
    Outgoing update filter list for all interfaces is not set
    Incoming update filter list for all interfaces is not set
    Router ID 3.3.3.3
    Number of areas in this router is 1. 1 normal 0 stub 0 nssa
    Maximum path: 4
    Routing for Networks:
      Passive Interface(s):
        GigabitEthernet0/0/0
    Routing Information Sources:
      Gateway          Distance      Last Update
      1.1.1.1          110           00:04:54
      2.2.2.2          110           00:17:26
      3.3.3.3          110           00:04:54
    Distance: (default is 110)

R3>
```

3.2.2 Exercise 2.3.11 - Packet Tracer - Determine the DR and BDR

3.2.2.1 *Topology*



3.2.2.2 Addressing Table

Device	Interface	IP Address	Subnet Mask
RA	G0/0	192.168.1.1	255.255.255.0
	Lo0	192.168.31.11	255.255.255.255
RB	G0/0	192.168.1.2	255.255.255.0
	Lo0	192.168.31.22	255.255.255.255
RC	G0/0	192.168.1.3	255.255.255.0
	Lo0	192.168.31.33	255.255.255.255

3.2.2.3 Objectives

Part 1: Examine DR and BDR Changing Roles Part 2:

Modify OSPF Priority and Force Elections

3.2.2.4 Scenario

In this activity, you will examine DR and BDR roles and watch the roles change when there is a change in the network. You will then modify the priority to control the roles and force a new election. Finally, you will verify routers are filling the desired roles.

3.2.2.5 Instructions

Part 1: Examine DR and BDR Changing Roles

Step 1: Wait until the amber link lights turn green.

When you first open the file in Packet Tracer, you may notice that the link lights for the switch are amber. These link lights will stay amber for 50 seconds while the STP protocol on the switch makes sure that one of the routers is not another switch. Alternatively, you can click **Fast Forward Time** to bypass this process.

Step 2: Verify the current OSPF neighbor states.

Use the appropriate command on each router to examine the current DR and BDR. If a router shows FULL/DROTHER it means that the router is not a DR or a BDR.

RA# **show ip ospf neighbor**

Neighbor ID	Pri	State	Dead Time	Address	Interface
192.168.31.33	2	FULL/DR	00:00:35	192.168.1.3	GigabitEthernet0/0
192.168.31.22	1	FULL/BDR	00:00:35	192.168.1.2	GigabitEthernet0/0

RB# **show ip ospf neighbor**

Neighbor ID	Pri	State	Dead Time	Address	Interface
192.168.31.11	1	FULL/DROTHER	00:00:36	192.168.1.1	GigabitEthernet0/0
192.168.31.33	2	FULL/DR	00:00:36	192.168.1.3	GigabitEthernet0/0

RC# **show ip ospf neighbor**

Neighbor ID	Pri	State	Dead Time	Address	Interface
192.168.31.11	1	FULL/DROTHER	00:00:39	192.168.1.1	GigabitEthernet0/0
192.168.31.22	1	FULL/BDR	00:00:38	192.168.1.2	GigabitEthernet0/0

Which router is the DR? RC

Which router is the BDR? RB

What is the OSPF state of router RA? DROTHER

Step 3: Turn on IP OSPF adjacency debugging.

You can monitor the DR and BDR election process with a **debug** command. On **RA** and **RB**, enter the following command.

RA# **debug ip ospf adj**

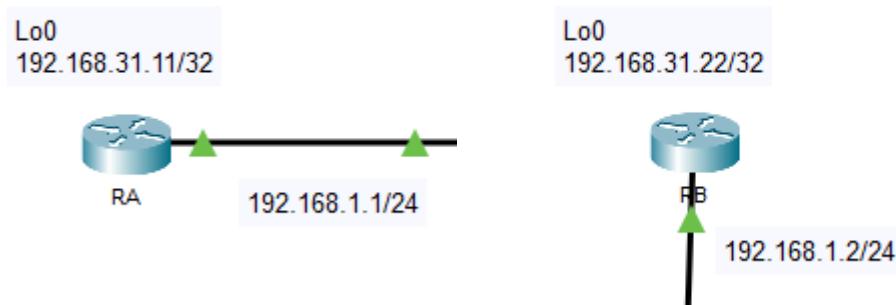
RB# **debug ip ospf adj**

Step 4: Disable the Gigabit Ethernet 0/0 interface on RC.

- Use the **shutdown** command to disable the link between **RC** and the switch to cause roles to change.
- Wait about 30 seconds for the dead timers to expire on **RA** and **RB**.

According to the debug output, which router was elected DR and which router was elected BDR?

RA is now BDR and RB is now DR. RA was the BDR, when the DR failed (RB) it became the DR.



```
RA#show ip ospf neighbor
Neighbor ID      Pri      State        Dead Time    Address
Interface
192.168.31.11    1      FULL/DR      00:00:35   192.168.1.1
192.168.1.1      GigabitEthernet0/0
RA#
```

```
RA#show ip ospf neighbor
Neighbor ID      Pri      State        Dead Time    Address
Interface
192.168.31.22    1      FULL/DR      00:00:30   192.168.1.2
192.168.1.2      GigabitEthernet0/0
RA#
```

Step 5: Restore the Gigabit Ethernet 0/0 interface on RC.

- Re-enable the link between **RC** and the switch.

```
RC#conf t
Enter configuration commands, one per line. End with
CTRL/Z.
RC(config)#int g0/0
RC(config-if)#no shutdown

RC(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed
state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0, changed state to up

RC(config-if)#end
RC#
%SYS-5-CONFIG_I: Configured from console by console

RC#
00:19:21: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.22 on
GigabitEthernet0/0 from LOADING to FULL, Loading Done

00:19:21: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.11 on
GigabitEthernet0/0 from LOADING to FULL, Loading Done
```

- Wait for the new DR/BDR elections to occur.

Did DR and BDR roles change? Explain.

No, roles did not change because the current DR and BDR are still active. A router that comes online with a higher router ID will not assume the BDR role until the BDR fails.

- c. Verify the DR and BDR assignments using the **show ip ospf neighbor** command on router RC.

```
RC# show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
192.168.31.22	1	FULL/DR	00:00:34	192.168.1.2	GigabitEthernet0/0
192.168.31.11	1	FULL/BDR	00:00:34	192.168.1.1	GigabitEthernet0/0

```
RC#show ip ospf neighbor

Neighbor ID      Pri   State          Dead Time    Address      Interface
192.168.31.22    1     FULL/DROTHER  00:00:32     192.168.1.2  GigabitEthernet0/0
192.168.31.11    1     FULL/BDR       00:00:32     192.168.1.1  GigabitEthernet0/0
RC#
```

Note: if the **show ip ospf neighbor** command does not return RB as the DR and RA as the BDR, turn off debugging on RA and RB with the **undebbug all** command and retry steps 4 and 5.

```
RA>
RA>
RA>enable
RA#undebbug all
All possible debugging has been turned off
RA#
00:25:56: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.33 on GigabitEthernet0/0 from FULL to DOWN, Neighbor Down: Dead timer expired
00:25:56: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.33 on GigabitEthernet0/0 from FULL to DOWN, Neighbor Down: Interface down or detached
```

```
RB>
RB>
RB>enable
RB#undebbug all
All possible debugging has been turned off
RB#
RB#
00:25:56: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.33 on GigabitEthernet0/0 from FULL to DOWN, Neighbor Down: Dead timer expired
00:25:56: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.33 on GigabitEthernet0/0 from FULL to DOWN, Neighbor Down: Interface down or detached
RB#
```

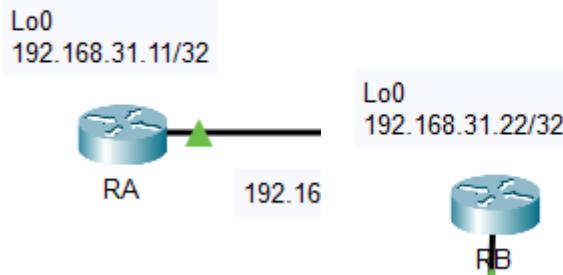
```

RA# show ip ospf neighbor
RA# show ip ospf neighbor
RA# show ip ospf neighbor

Neighbor ID      Pri   State          Dead Time    Address
Address          Interface
192.168.31.11    1     FULL/DR        00:00:31
192.168.1.1      GigabitEthernet0/0
RA#
RA# show ip ospf neighbor
RA# show ip ospf neighbor
RA# show ip ospf neighbor

Neighbor ID      Pri   State          Dead Time    Address
Interface
192.168.31.22    1     FULL/BDR       00:00:33
192.168.1.2      GigabitEthernet0/0
RA#

```



RA is DR

RB is BDR

Step 6: Disable the GigabitEthernet0/0 interface on RB.

- Disable the link between **RB** and the switch to cause roles to change.

```

RB(config)# int g0/0
RB(config-if)# shut
RB(config-if)# shutdown

RB(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed
state to administratively down

%LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0, changed state to down

00:32:48: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.11
on GigabitEthernet0/0 from FULL to DOWN, Neighbor Down:
Interface down or detached

00:32:48: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.33
on GigabitEthernet0/0 from FULL to DOWN, Neighbor Down:
Interface down or detached

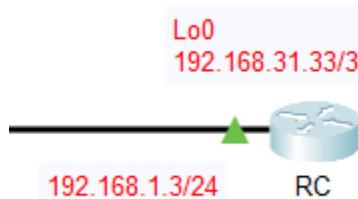
RB(config-if)#end
RB#
%SYS-5-CONFIG_I: Configured from console by console

RB#show ip ospf neighbor
RB#

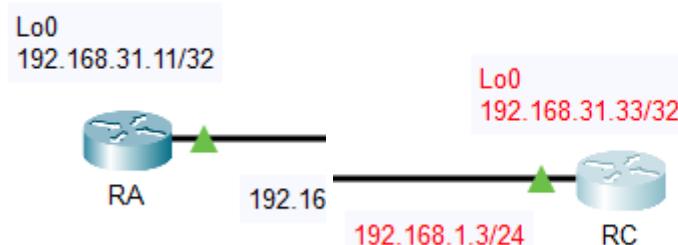
```

- b. Wait about 30 seconds for the holddown timers to expire on **RA** and **RC**.

According to the debug output on **RA**, which router was elected DR and which router was elected BDR?



Router RA (Output 1)					Router RC (Output 2)				
Router ID 192.168.31.33					Router ID 192.168.1.4				
Priority 2					Priority 1				
State FULL/BDR					State FULL/DR				
Interface GigabitEthernet0/0					Interface GigabitEthernet0/0				
RA#					RC#				
show ip ospf neighbor					show ip ospf neighbor				
Neighbor ID 192.168.31.33 Pri 2 State FULL/BDR Dead Time 00:00:37 Address 192.168.1.3 GigabitEthernet0/0					Neighbor ID 192.168.31.11 Pri 1 State FULL/DR Dead Time 00:00:39 Address 192.168.1.1 GigabitEthernet0/0				



RC is now BDR and RA is now DR. RA was the BDR, when the DR failed (RB) it became the DR.

Step 7: Restore the GigabitEthernet0/0 interface on RB.

- a. Re-enable the link between **RB** and the switch.

Wait for the new DR/BDR elections to occur. Did DR and BDR roles change? Explain.

No, roles did not change because the current DR and BDR are still active. A router that comes online with a higher router ID will not assume the BDR role until the BDR fails.

- b. Use the **show ip ospf interface** command on router RC.

What is the status of router RC now? BDR

```

RA# show ip ospf neighbor
* 00:38:32: OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.22 on GigabitEthernet0/0 from LOADING to FULL, Loading Done
* 00:38:37: OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.11 on GigabitEthernet0/0 from LOADING to FULL, Loading Done
* 00:30:22: OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.22 on GigabitEthernet0/0 from FULL to DOWN, Neighbor Down: Dead timer expired
* 00:30:22: OSPF-5-ADJCHG: Process 1, Nbr 192.168.31.22 on GigabitEthernet0/0 from FULL to DOWN, Neighbor Down: Interface down or detached
RA# show ip ospf neighbor
* Neighbor ID      Pri   State          Dead Time    Address
* Interface
* 192.168.31.11    1     FULL/DR        00:00:30    192.168.1.1
* GigabitEthernet0/0
RA# show ip ospf neighbor
* Neighbor ID      Pri   State          Dead Time    Address
* Interface
* 192.168.31.22    2     FULL/DR        00:00:30    192.168.1.3
* GigabitEthernet0/0
* 192.168.31.11    1     FULL/DR        00:00:35    192.168.1.2
* GigabitEthernet0/0
RA#

```

RC (33) is now BDR and RA (.11) is now DR. RA was the BDR, when the DR failed (RB) it became the DR.

Step 8: Turn off Debugging.

Enter the command **undebbug all** on **RA** and **RB** to disable debugging.

Part 2: Modify OSPF Priority and Force Elections

Step 1: Configure OSPF priorities on each router.

- To change the DR and BDR, use the **ip ospf priority** command to configure the GigabitEthernet 0/0 port of each router with the following OSPF interface priorities:
 - RA:** 200
 - RB:** 100
 - RC:** 1 (This is the default priority)

```

RA(config)# interface g0/0
RA(config-if)# ip ospf priority 200

```

- Set the priority on router **RB** and **RC**.

```

Router RA Configuration:
RA#enable
RA(config)#interface g0/0
RA(config-if)#ip ospf priority 200
RA(config-if)#end
RA#
%SYS-5-CONFIG_I: Configured from console by console

RA#show ip ospf neighbor
Neighbor ID      Pri  State        Dead Time   Address
Interface
192.168.31.33    2   FULL/DR      00:00:30   192.168.1.3
GigabitEthernet0/0
192.168.31.22    1   FULL/BROTHER  00:00:30   192.168.1.2
GigabitEthernet0/0
RA#
RA>enable
RA(config)t
Enter configuration commands, one per line. End with CNTL/Z.
RA(config)#interface g0/0
RA(config-if)#ip ospf priority 200
RA(config-if)#end
RA#
%SYS-5-CONFIG_I: Configured from console by console

RA#show ip ospf neighbor
Neighbor ID      Pri  State        Dead Time   Address
Interface
192.168.31.33    2   FULL/DR      00:00:30   192.168.1.3
GigabitEthernet0/0
192.168.31.22    1   FULL/BROTHER  00:00:30   192.168.1.2
GigabitEthernet0/0
RA#

```

Router RB Configuration:

```

RB#enable
RB(config)#interface g0/0
RB(config-if)#ip ospf priority 100
RB(config-if)#end
RB#
%SYS-5-CONFIG_I: Configured from console by console

RB#show ip ospf neighbor
Neighbor ID      Pri  State        Dead Time   Address
Interface
192.168.31.11    1   FULL/DR      00:00:30   192.168.1.1
GigabitEthernet0/0
RB#
RB>enable
RB(config)t
Enter configuration commands, one per line. End with CNTL/Z.
RB(config)#interface g0/0
RB(config-if)#ip ospf priority 100
RB(config-if)#end
RB#
%SYS-5-CONFIG_I: Configured from console by console

RB#show ip ospf neighbor
Neighbor ID      Pri  State        Dead Time   Address
Interface
192.168.31.11    1   FULL/DR      00:00:30   192.168.1.1
GigabitEthernet0/0
RB#

```

Step 2: Force an election by resetting the OSPF process on the routers.

Starting with router **RA**, issue the **clear ip ospf process** on each router to reset the OSPF process.

Step 3: Verify DR and BDR elections were successful.

Wait long enough for OSPF to converge and for the DR/BDR election to occur. This should take a few minutes. You can click **Fast Forward Time** to speed up the process.

```

Router RA Configuration:
RA#enable
RA(config)#interface g0/0
RA(config-if)#ip ospf priority 200
RA(config-if)#end
RA#
%SYS-5-CONFIG_I: Configured from console by console

RA#show ip ospf neighbor
Neighbor ID      Pri  State        Dead Time   Address
Interface
192.168.31.33    1   FULL/BROTHER  00:00:30   192.168.1.3
GigabitEthernet0/0
192.168.31.22    200  FULL/DR      00:00:35   192.168.1.2
GigabitEthernet0/0
RA#
RA>enable
RA(config)c
Press RETURN to get started.

Router RB Configuration:
RB#enable
RB(config)#interface g0/0
RB(config-if)#ip ospf priority 100
RB(config-if)#end
RB#
%SYS-5-CONFIG_I: Configured from console by console

RB#show ip ospf neighbor
Neighbor ID      Pri  State        Dead Time   Address
Interface
192.168.31.11    1   FULL/DR      00:00:30   192.168.1.1
GigabitEthernet0/0
RB#
RB>enable
RB(config)t
Enter configuration commands, one per line. End with CNTL/Z.
RB(config)#interface g0/0
RB(config-if)#ip ospf priority 100
RB(config-if)#end
RB#
%SYS-5-CONFIG_I: Configured from console by console

RB#show ip ospf neighbor
Neighbor ID      Pri  State        Dead Time   Address
Interface
192.168.31.11    1   FULL/DR      00:00:30   192.168.1.1
GigabitEthernet0/0
RB#

```

Router RC Configuration:

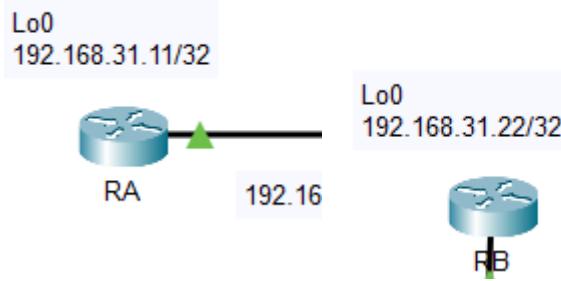
```

RC#enable
RC(config)#interface g0/0
RC(config-if)#ip ospf priority 1
RC(config-if)#end
RC#
%SYS-5-CONFIG_I: Configured from console by console

RC#show ip ospf neighbor
Neighbor ID      Pri  State        Dead Time   Address
Interface
192.168.31.11    1   FULL/DR      00:00:37   192.168.1.1
GigabitEthernet0/0
192.168.31.33    200  FULL/BROTHER  00:00:32   192.168.1.3
GigabitEthernet0/0
RC#

```

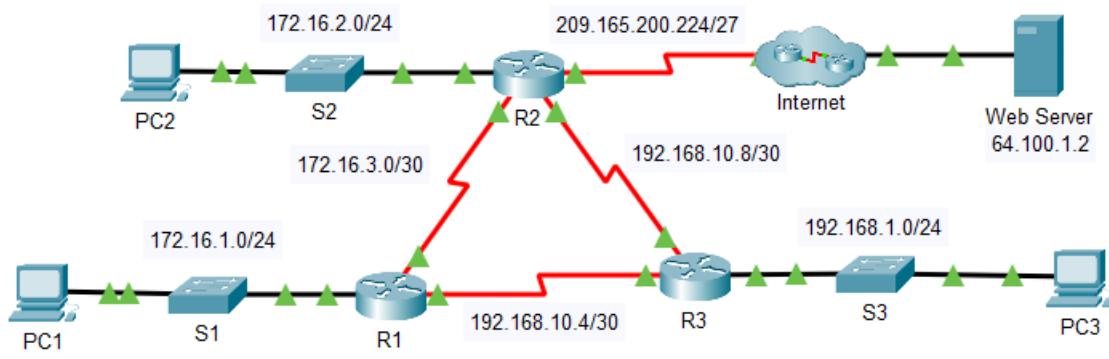
According to output from the **show ip ospf neighbor** command on the routers, which router is now DR and which router is now BDR? RA is now DR and RB is now BDR.



Note: If the routers do not elect the correct DR and BDR after setting the OSPF priorities try restarting Packet Tracer.

3.2.3 Exercise 2.4.11 - Packet Tracer - Modify Single-Area OSPFv2

3.2.3.1 Topology



3.2.3.2 Addressing Table

Device	Interface	IPv4 Address	Subnet Mask	Default Gateway
R1	G0/0	172.16.1.1	255.255.255.0	N/A
	S0/0/0	172.16.3.1	255.255.255.252	
	S0/0/1	192.168.10.5	255.255.255.252	
R2	G0/0	172.16.2.1	255.255.255.0	N/A
	S0/0/0	172.16.3.2	255.255.255.252	
	S0/0/1	192.168.10.9	255.255.255.252	

	S0/1/0	209.165.200.225	255.255.255.224	
R3	G0/0	192.168.1.1	255.255.255.0	N/A
	S0/0/0	192.168.10.6	255.255.255.252	
	S0/0/1	192.168.10.10	255.255.255.252	
PC1	NIC	172.16.1.2	255.255.255.0	172.16.1.1
PC2	NIC	172.16.2.2	255.255.255.0	172.16.2.1
PC3	NIC	192.168.1.2	255.255.255.0	192.168.1.1
Web Server	NIC	64.100.1.2	255.255.255.0	64.100.1.1

3.2.3.3 Objectives

Part 1: Modify OSPF Default Settings Part 2:

Verify Connectivity

3.2.3.4 Scenario

In this activity, OSPF is already configured and all end devices currently have full connectivity. You will modify the default OSPF routing configurations by changing the hello and dead timers and adjusting the bandwidth of a link. Then you will verify that full connectivity is restored for all end devices.

3.2.3.5 Instructions

Part 1: Modify OSPF Default Settings

Step 1: Test connectivity between all end devices.

Before modifying the OSPF settings, verify that all PCs can ping the web server and each other.

Step 2: Adjust the hello and dead timers between R1 and R2.

- Enter the following commands on **R1**.

```
R1(config)# interface s0/0/0
R1(config-if)# ip ospf hello-interval 15
R1(config-if)# ip ospf dead-interval 60
```

- After a short period of time, the OSPF connection with **R2** will fail, as shown in the router output.

```
00:02:50: %OSPF-5-ADJCHG: Process 1, Nbr 209.165.200.225 on Serial0/0/0 from FULL to
DOWN, Neighbor Down: Dead timer expired
```

```
00:02:50: %OSPF-5-ADJCHG: Process 1, Nbr 209.165.200.225 on Serial0/0/0 from FULL to
DOWN, Neighbor Down: Interface down or detached
```

Both sides of the connection need to have the same timer values in order for the adjacency to be maintained. Identify the interface on R2 that is connected to R1. Adjust the timers on the R2 interface to match the settings on **R1**.

```
R2(config)# interface s0/0/0
```

```
R2(config-if)# ip ospf hello-interval 15
R2(config-if)# ip ospf dead-interval 60
```

After a brief period of time you should see a status message that indicates that the OSPF adjacency has been reestablished.

```
00:21:52: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.10.5 on Serial0/0/0 from
LOADING to FULL, Loading Done
```

Step 3: Adjust the bandwidth setting on R1.

- a. Trace the path between **PC1** and the web server located at 64.100.1.2. Notice that the path from **PC1** to 64.100.1.2 is routed through **R2**. OSPF prefers the lower cost path.

```
C:\> tracert 64.100.1.2
```

Tracing route to 64.100.1.2 over a maximum of 30 hops:

```
1 1 ms 0 ms 8 ms 172.16.1.1
2 0 ms 1 ms 0 ms 172.16.3.2
3 1 ms 9 ms 2 ms 209.165.200.226
4 * 1 ms 0 ms 64.100.1.2
```

Trace complete.

- b. On the **R1** Serial 0/0/0 interface, set the bandwidth to 64 Kb/s. This does not change the actual port speed, only the metric that the OSPF process on **R1** will use to calculate best routes.

```
R1(config-if)# bandwidth 64
```

- c. Trace the path between **PC1** and the web server located at 64.100.1.2. Notice that the path from **PC1** to 64.100.1.2 is redirected through **R3**. OSPF prefers the lower cost path.

```
C:\> tracert 64.100.1.2
```

Tracing route to 64.100.1.2 over a maximum of 30 hops:

```
1 1 ms 0 ms 3 ms 172.16.1.1
2 8 ms 1 ms 1 ms 192.168.10.6
3 2 ms 0 ms 2 ms 172.16.3.2
4 2 ms 3 ms 1 ms 209.165.200.226
5 2 ms 11 ms 11 ms 64.100.1.2
```

Trace complete.

Part 2: Verify Connectivity

Verify that all PCs can ping the web server and each other.

```
Reply from 172.16.2.2. bytes=32 time=20ms TTL=125

Ping statistics for 172.16.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 20ms, Average = 14ms

C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.1.2: bytes=32 time=1ms TTL=126
Reply from 192.168.1.2: bytes=32 time=6ms TTL=126
Reply from 192.168.1.2: bytes=32 time=9ms TTL=126

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 9ms, Average = 5ms

C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time=16ms TTL=126
Reply from 192.168.1.2: bytes=32 time=1ms TTL=126
Reply from 192.168.1.2: bytes=32 time=14ms TTL=126
Reply from 192.168.1.2: bytes=32 time=5ms TTL=126

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 16ms, Average = 9ms
```

```
C:\>ping 64.100.1.2

Pinging 64.100.1.2 with 32 bytes of data:

Reply from 64.100.1.2: bytes=32 time=43ms TTL=124
Reply from 64.100.1.2: bytes=32 time=24ms TTL=124
Reply from 64.100.1.2: bytes=32 time=27ms TTL=124
Reply from 64.100.1.2: bytes=32 time=22ms TTL=124

Ping statistics for 64.100.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 22ms, Maximum = 43ms, Average = 29ms

c:\>
```

PC2

PC2

Device Programming

Serial Port

```
C:\>ping 172.16.1.1

Pinging 172.16.1.1 with 32 bytes of data:

Reply from 172.16.1.1: bytes=32 time=22ms TTL=253
Reply from 172.16.1.1: bytes=32 time=2ms TTL=253
Reply from 172.16.1.1: bytes=32 time=29ms TTL=253
Reply from 172.16.1.1: bytes=32 time=16ms TTL=253

Ping statistics for 172.16.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 29ms, Average = 16ms

C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time=8ms TTL=126
Reply from 192.168.1.2: bytes=32 time=4ms TTL=126
Reply from 192.168.1.2: bytes=32 time=4ms TTL=126
Reply from 192.168.1.2: bytes=32 time=13ms TTL=126

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 4ms, Maximum = 13ms, Average = 7ms
```

```
C:\>ping 64.100.1.2

Pinging 64.100.1.2 with 32 bytes of data:

Reply from 64.100.1.2: bytes=32 time=21ms TTL=126
Reply from 64.100.1.2: bytes=32 time=17ms TTL=126
Reply from 64.100.1.2: bytes=32 time=12ms TTL=126
Reply from 64.100.1.2: bytes=32 time=1ms TTL=126

Ping statistics for 64.100.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 21ms, Average = 12ms

C:\>
```

PC3

```
PC3
Desktop Programming
Command Prompt
C:\>ping 172.16.1.1

Pinging 172.16.1.1 with 32 bytes of data:

Reply from 172.16.1.1: bytes=32 time=16ms TTL=254
Reply from 172.16.1.1: bytes=32 time=10ms TTL=254
Reply from 172.16.1.1: bytes=32 time=4ms TTL=254
Reply from 172.16.1.1: bytes=32 time=5ms TTL=254

Ping statistics for 172.16.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 16ms, Average = 8ms

C:\>ping 172.16.2.2

Pinging 172.16.2.2 with 32 bytes of data:

Reply from 172.16.2.2: bytes=32 time=20ms TTL=126
Reply from 172.16.2.2: bytes=32 time=6ms TTL=126
Reply from 172.16.2.2: bytes=32 time=17ms TTL=126
Reply from 172.16.2.2: bytes=32 time=8ms TTL=126

Ping statistics for 172.16.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 6ms, Maximum = 20ms, Average = 12ms
```

```
C:\>ping 64.100.1.2

Pinging 64.100.1.2 with 32 bytes of data:

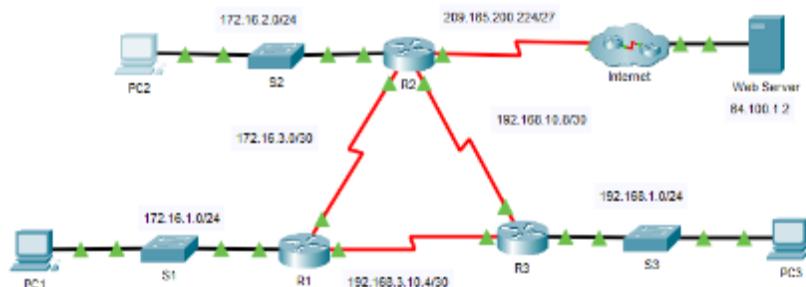
Reply from 64.100.1.2: bytes=32 time=4ms TTL=125
Reply from 64.100.1.2: bytes=32 time=11ms TTL=125
Reply from 64.100.1.2: bytes=32 time=11ms TTL=125
Reply from 64.100.1.2: bytes=32 time=24ms TTL=125

Ping statistics for 64.100.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 24ms, Average = 12ms

C:\>
```

3.2.4 Exercise 2.5.3 - Packet Tracer - Propagate a Default Route in OSPFv2

3.2.4.1 Topology



3.2.4.2 Addressing Table

Device	Interface	IPv4 Address	Subnet Mask	Default Gateway
R1	G0/0	172.16.1.1	255.255.255.0	N/A
	S0/0/0	172.16.3.1	255.255.255.252	
	S0/0/1	192.168.10.5	255.255.255.252	
R2	G0/0	172.16.2.1	255.255.255.0	N/A
	S0/0/0	172.16.3.2	255.255.255.252	
	S0/0/1	192.168.10.9	255.255.255.252	
	S0/1/0	209.165.200.225	255.255.255.224	
R3	G0/0	192.168.1.1	255.255.255.0	N/A
	S0/0/0	192.168.10.6	255.255.255.252	
	S0/0/1	192.168.10.10	255.255.255.252	
PC1	NIC	172.16.1.2	255.255.255.0	172.16.1.1
PC2	NIC	172.16.2.2	255.255.255.0	172.16.2.1
PC3	NIC	192.168.1.2	255.255.255.0	192.168.1.1
Web Server	NIC	64.100.1.2	255.255.255.0	64.100.1.1

3.2.4.3 Objectives

Part 1: Propagate a Default Route Part

2: Verify Connectivity

3.2.4.4 Background

In this activity, you will configure an IPv4 default route to the Internet and propagate that default route to other OSPF routers. You will then verify the default route is in downstream routing tables and that hosts can now access a web server on the Internet.

3.2.4.5 Instructions

Part 1: Propagate a Default Route

Step 1: Test connectivity to the Web Server

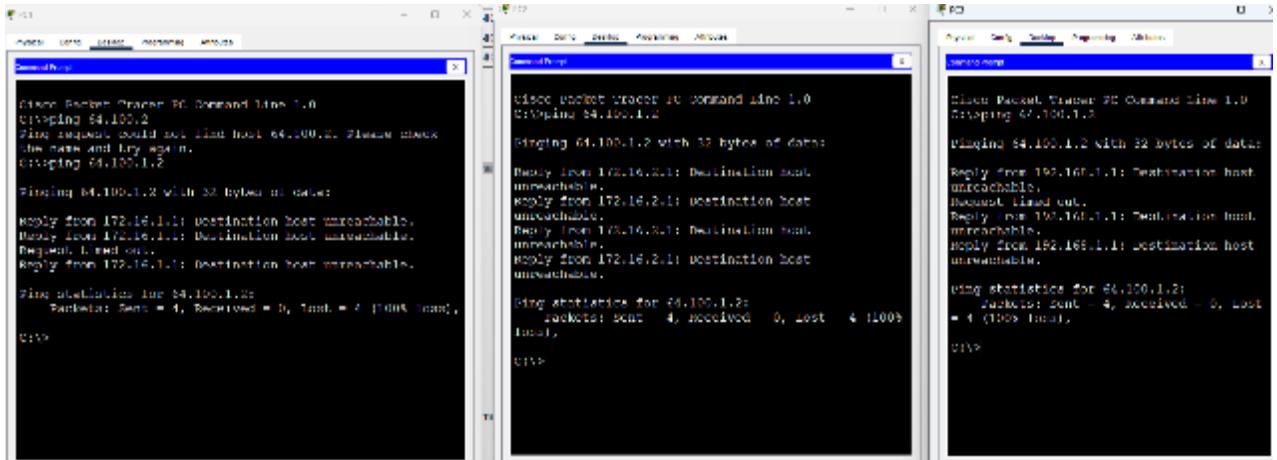
- From PC1, PC2, and PC3, attempt to ping the Web Server IP address, 64.100.1.2.

Were any of the pings successful? **Answer** No

What message did you receive, and which device issued the message?

Answer Destination unreachable, Default gateway (Routers)

176.16.1.1 , 172.16.2.1 and 172.16.1.2



b. Examine the routing tables on routers R1, R2, and R3.

```

R1#show ip route
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

 172.16.0.0/16 is variably subnetted, 5 subnets, 3 masks
C       172.16.1.0/24 is directly connected, GigabitEthernet0/0
L       172.16.1.1/32 is directly connected, GigabitEthernet0/0
O       172.16.2.0/24 [110/65] via 172.16.3.2, Serial0/0/0
C       172.16.3.0/30 is directly connected, Serial0/0/0
L       172.16.3.1/32 is directly connected, Serial0/0/0
C       192.168.1.0/24 [110/65], via 192.168.10.6, GigabitEthernet0/0
C       192.168.1.1/32 is variably subnetted, 3 subnets, 2 masks
C         192.168.1.0/24 is directly connected, Serial0/0/1
L         192.168.1.1/32 is directly connected, Serial0/0/1
O       192.168.1.0/30 [110/128] via 192.168.10.6, 00:08:01, Serial0/0/1
          [110/128] via 192.168.10.5, 00:10:51, Serial0/0/0
 209.165.200.0/27 is subnetted, 1 subnets
O       209.165.200.224/27 [110/128] via 192.168.10.9, 00:11:01, Serial0/0/1

R2#
R3#

```

```

R3#show ip route
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

```

```

 172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
O       172.16.1.0/24 [110/65] via 172.16.3.1, 00:08:01, serial0/0/0
O       172.16.2.0/24 [110/65] via 192.168.10.9, 00:11:01, serial0/0/1
O       172.16.3.0/30 [110/128] via 192.168.10.5, 00:10:51, serial0/0/0
 192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.1.0/24 is directly connected, GigabitEthernet0/0
L       192.168.1.1/32 is directly connected, GigabitEthernet0/0
 192.168.10.0/24 is variably subnetted, 4 subnets, 2 masks
C       192.168.10.4/30 is directly connected, Serial0/0/0
L       192.168.10.6/32 is directly connected, Serial0/0/0
C       192.168.10.8/30 is directly connected, Serial0/0/1
L       192.168.10.10/32 is directly connected, Serial0/0/1
 209.165.200.0/27 is subnetted, 1 subnets
O       209.165.200.224/27 [110/128] via 192.168.10.9, 00:11:01, Serial0/0/1

```

What statement is present in the routing tables that indicates that the pings to the Web Server will fail?

Answer Gateway of last resort is not set

Step 2: Configure a default route on R2.

Configure **R2** with a directly attached default route to the Internet.

```
R2(config)# ip route 0.0.0.0 0.0.0.0 Serial0/1/0
```

Note: Router will give a warning that if this interface is not a point-to-point connection, it may impact performance. You can ignore this warning because it is a point-to-point connection.

```
R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#ip route 0.0.0.0 0.0.0.0 Serial0/1/0
%Default route without gateway, if not a point-to-point interface, may
impact performance
R2(config)#end
R2#
%SYS-5-CONFIG_I: Configured from console by console

R2#
```

Step 3: Propagate the route in OSPF.

Configure OSPF to propagate the default route in OSPF routing updates.

```
R2(config)# router ospf 1
R2(config-router)# default-information originate
```

```
R2#enable
R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#router ospf 1
R2(config-router)#default-information originate
R2(config-router)#end
R2#
%SYS-5-CONFIG_I: Configured from console by console
```

Step 4: Examine the routing tables on R1 and R3.

Examine the routing tables of **R1** and **R3** to verify that the route has been propagated.

```
R1> show ip route
<output omitted>
Gateway of last resort is 172.16.3.2 to network 0.0.0.0
<output omitted>
O*E2 0.0.0.0/0 [110/1] via 172.16.3.2, 00:00:08, Serial0/0/0
!-----
```

R3> show ip route

<output omitted>

Gateway of last resort is 192.168.10.9 to network 0.0.0.0

<output omitted>

0*E2 0.0.0.0/0 [110/1] via 192.168.10.9, 00:08:15, Serial0/0/1

Part 2: Verify Connectivity

Verify that **PC1**, **PC2**, and **PC3** can ping the web server.

```

ping to 64.100.1.2
RTT min/avg/max = 0.000/0.000/0.000 ms
Approximate round trip times in milli-seconds:
    Minimum = 0.000, Maximum = 0.000, Average = 0.000 ms

```

```
Brk>
C:\>ping 64.100.1.2

Pinging 64.100.1.2 with 32 bytes of data:
Request timed out.
Reply from 64.100.1.2: bytes=32 time=1ms ttl=128
Reply from 64.100.1.2: bytes=32 time=1ms ttl=128
Reply from 64.100.1.2: bytes=32 time=1ms ttl=128

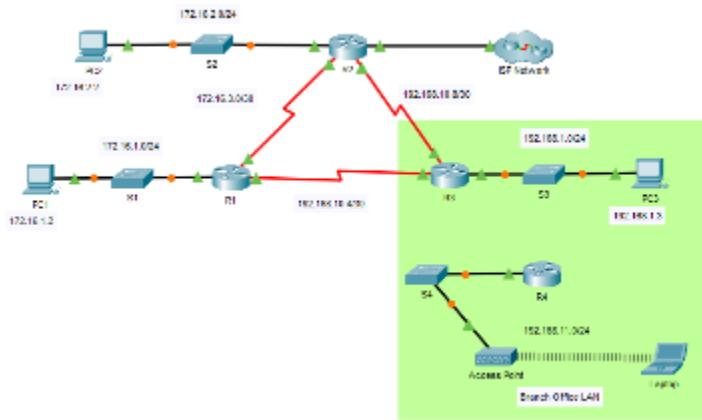
Ping statistics for 64.100.1.2:
    packets sent = 4, received = 3, loss = 1 (25%)
Approximate round trip times in milli seconds:
    Minimum = 1ms, Maximum = 12ms, Average = 1ms

Brk>
C:\>
C:\>
C:\>
C:\>
```

```
Testing 64.100.1.2 with 32 bytes of data:  
Reply from 64.100.1.2 bytes=32 time=1ms  
TTL=128  
  
Ping statistics for 64.100.1.2:  
    Packets: Sent = 4, Received = 4, Lost =  
    0 (0.0% loss),  
    approximate round trip times in milli-  
    seconds:  
        Minimum 14ms, Maximum 18ms, average  
    = 16ms
```

3.2.5 Exercise 2.6.6 - Packet Tracer - Verify Single-Area OSPFv2

3.2.5.1 Topology



3.2.5.2 Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0	172.16.1.1	255.255.255.0	N/A
	G0/1	64.100.54.6	255.255.255.252	
	S0/0/0	172.16.3.1	255.255.255.252	
	S0/0/1	192.168.10.5	255.255.255.252	
R2	G0/0	172.16.2.1	255.255.255.0	N/A
	S0/0/0	172.16.3.2	255.255.255.252	
	S0/0/1	192.168.10.9	255.255.255.252	
R3	G0/0	192.168.1.1	255.255.255.0	N/A
	G0/1	192.168.11.1	255.255.255.0	
	S0/0/0	192.168.10.6	255.255.255.252	
	S0/0/1	192.168.10.10	255.255.255.252	
R4	G0/0/0	192.168.1.2	255.255.255.0	N/A
	G0/0/1	192.168.11.1	255.255.255.0	
ISP Router	NIC	64.100.54.5	255.255.255.252	N/A
PC1	NIC	172.16.1.2	255.255.255.0	172.16.1.1
PC2	NIC	172.16.2.2	255.255.255.0	172.16.2.1
PC3	NIC	192.168.1.2	255.255.255.0	192.168.1.1
Laptop	NIC	DHCP	DHCP	DHCP

3.2.5.3 Objectives

In this lab, you will use the CLI commands to verify the operation of an existing OSPFv2 network. In Part 2, you will add a new LAN to the configuration and verify connectivity.

- Identify and verify the status of OSPF neighbors.
- Determine how the routes are being learned in the network.
- Explain how the neighbor state is determined.

- Examine the settings for the OSPF process ID.
- Add a new LAN into an existing OSPF network and verify connectivity.

3.2.5.4 Background / Scenario

You are the network administrator for a branch office of a larger organization. Your branch is adding a new wireless network into an existing branch office LAN. The existing network is configured to exchange routes using OSPFv2 in a single-area configuration. Your task is to verify the operation of the existing OSPFv2 network, before adding in the new LAN. When you are sure that the current OSPFv2 LAN is operating correctly, you will connect the new LAN and verify that OSPF routes are being propagated for the new LAN. As branch office network administrator, you have full access to the IOS on routers R3 and R4. You only have read access to the enterprise LAN routers R1 and R2, using the username **BranchAdmin**, and the password **Branch1234**.

3.2.5.5 Instructions

Part 1: Verify the existing OSPFv2 network operation.

The following commands will help you find the information needed to answer the questions:

```
show ip interface brief show ip
route
show ip route ospf show ip
ospf neighbor show ip
protocols show ip ospf
show ip ospf interface
```

Step 1: Verify OSPFv2 operation.

Wait until STP has converged on the network. You can click the Packet Tracer Fast Forward Time button to speed up the process. Continue only when all link lights are green.

- a. Log into router **R1** using the username **BranchAdmin** and the password **Branch1234**. Execute the **show ip route** command.

```
R1# show ip route
--- output omitted ---
```

Gateway of last resort is 172.16.3.2 to network 0.0.0.0

```
      172.16.0.0/16 is variably subnetted, 5 subnets, 3 masks
C       172.16.1.0/24 is directly connected, GigabitEthernet0/0
L       172.16.1.1/32 is directly connected, GigabitEthernet0/0
O       172.16.2.0/24 [110/65] via 172.16.3.2, 00:02:18, Serial0/0/0
C       172.16.3.0/30 is directly connected, Serial0/0/0
L       172.16.3.1/32 is directly connected, Serial0/0/0
O       192.168.1.0/24 [110/65] via 192.168.10.6, 00:02:18, Serial0/0/1
          192.168.10.0/24 is variably subnetted, 3 subnets, 2 masks
C           192.168.10.4/30 is directly connected, Serial0/0/1
L           192.168.10.5/32 is directly connected, Serial0/0/1
O           192.168.10.8/30 [110/128] via 172.16.3.2, 00:02:18, Serial0/0/0
                           [110/128] via 192.168.10.6, 00:02:18, Serial0/0/1
O*E2 0.0.0.0/0 [110/1] via 172.16.3.2, 00:02:18, Serial0/0/0
```

How did router **R1** receive the default route? **Answer** - The default route was learned through OSPF.

From which router did **R1** receive the default route? **Answer** – R2

How can you filter the output of **show ip route** to show only the routes learned through OSPF? **Answer** - show ip route ospf

- b. Execute the **show ip ospf neighbor** command on **R1**.

Which routers have formed adjacencies with router **R1**?

Answer R2 and R3

What are the router IDs and state of the routers shown in the command output? **Answer** 2.2.2.2 and 3.3.3.3

Are all of the adjacent routers shown in the output? **Answer** Yes

```
R1#show ip ospf neighbor

Neighbor ID      Pri  State        Dead Time   Address          Interface
2.2.2.2           0    FULL/        00:00:33   172.16.3.2      Serial0/0/0
3.3.3.3           0    FULL/        00:00:33   192.168.10.6    Serial0/0/1
R1#
```

- c. Using the command prompt on **PC1**, ping the address of the **ISP Router** shown in the Address Table. Is it successful? If not, do a **clear ospf process** command on the routers and repeat the ping command.

```
PC1
Physical Config Desktop Programming
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 64.100.54.5

Pinging 64.100.54.5 with 32 bytes of data:
Request timed out.
Reply from 64.100.54.5: bytes=32 time=12ms TTL=253
Reply from 64.100.54.5: bytes=32 time=1ms TTL=253
Reply from 64.100.54.5: bytes=32 time=13ms TTL=253

Ping statistics for 64.100.54.5:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 13ms, Average = 8ms

C:\>
```

Step 2: Verify OSPFv2 operation on R2.

- a. Log into router **R2** using the username **BranchAdmin** and the password **Branch1234**. Execute the **show ip route** command. Verify that routes to all the networks in the topology are shown in the routing table.

```

R2#show ip route
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 - ECP
      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 64.100.54.5 to network 0.0.0.0

  64.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C    64.100.54.4/30 is directly connected, GigabitEthernet0/1
L    64.100.54.6/32 is directly connected, GigabitEthernet0/1
  172.16.0.0/16 is variably subnetted, 5 subnets, 3 masks
C    172.16.1.0/24 [110/65] via 172.16.3.1, 00:42:55, Serial0/0/0
C    172.16.2.0/24 is directly connected, GigabitEthernet0/0
L    172.16.2.1/32 is directly connected, GigabitEthernet0/0
C    172.16.3.0/30 is directly connected, Serial0/0/0
L    172.16.3.2/32 is directly connected, Serial0/0/0
C    192.160.1.0/24 [110/65] via 192.160.10.10, 00:42:55, Serial0/0/1
  192.160.10.0/24 is variably subnetted, 3 subnets, 2 masks
C    192.160.10.4/30 [110/120] via 192.160.10.10, 00:42:55, Serial0/0/1
C    192.160.10.8/32 is directly connected, Serial0/0/1
L    192.160.10.9/32 is directly connected, Serial0/0/1
S*   0.0.0.0/0 [1/0] via 64.100.54.5

R2#

```

How did router R2 learn the default route to the ISP? **Answer** - It was statically configured by the administrator.

b. Enter the **show ip ospf interface g0/0** on router R2.

```

R2#show ip ospf interface g0/0

GigabitEthernet0/0 is up, line protocol is up
  Internet address is 172.16.2.1/24, Area 0
  Process ID 10, Router ID 2.2.2.2, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 2.2.2.2, Interface address 172.16.2.1
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    No Hellos (Passive interface)
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)

R2#

```

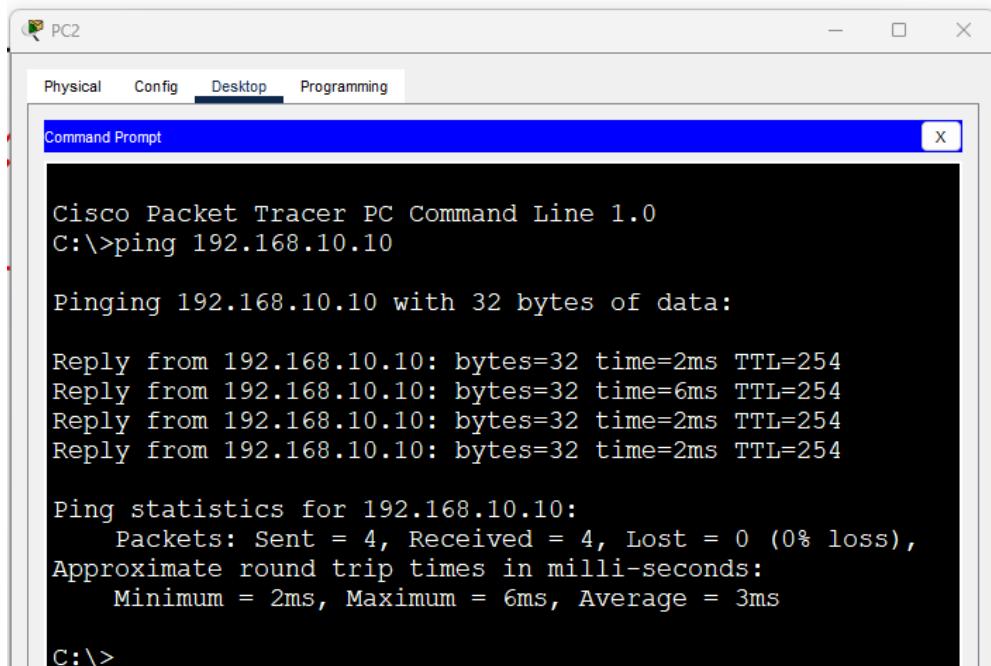
What type of OSPF network is attached to this interface? **Answer** -

BROADCAST

Are OSPF hello packets being sent out this interface? Explain. **Answer** No. The interface is configured as a passive interface in OSPF.

c. Using the command prompt on **PC2**, ping the S0/0/1 address on router **R3**.

Is it successful? **Answer** Yes



PC2

Physical Config Desktop Programming

Command Prompt X

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.10.10

Pinging 192.168.10.10 with 32 bytes of data:

Reply from 192.168.10.10: bytes=32 time=2ms TTL=254
Reply from 192.168.10.10: bytes=32 time=6ms TTL=254
Reply from 192.168.10.10: bytes=32 time=2ms TTL=254
Reply from 192.168.10.10: bytes=32 time=2ms TTL=254

Ping statistics for 192.168.10.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 6ms, Average = 3ms

C:\>
```

Step 3: Verify OSPFv2 operation on R3.

- Execute the **show ip protocols** command on router R3.

```
R3#show ip protocols

Routing Protocol is "ospf 10"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 3.3.3.3
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    192.168.1.0 0.0.0.255 area 0
    192.168.10.4 0.0.0.3 area 0
    192.168.10.8 0.0.0.3 area 0
  Routing Information Sources:
    Gateway          Distance      Last Update
    1.1.1.1           110          00:22:14
    2.2.2.2           110          00:22:15
    3.3.3.3           110          00:22:14
  Distance: (default is 110)

R3#
```

Router R3 is routing for which networks?

Answer-192.168.1.0/24,192.168.10.4/30 and 192.168.10.8/30

- Execute the **show ip ospf neighbor detail** command on router R3.

```
R3#show ip ospf neighbor detail
Neighbor 2.2.2.2, interface address 192.168.10.9
  In the area 0 via interface Serial0/0/1
  Neighbor priority is 0, State is FULL, 7 state changes
  DR is 0.0.0.0 BDR is 0.0.0.0
  Options is 0x00
  Dead Timer due in 00:00:30
  Neighbor is up for 00:54:24
  Index 1/1, retransmission queue length 0, number of retransmission 0
  First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
  Last retransmission scan length is 0, maximum is 0
  Last retransmission scan time is 0 msec, maximum is 0 msec
Neighbor 1.1.1.1, interface address 192.168.10.5
  In the area 0 via interface Serial0/0/0
  Neighbor priority is 0, State is FULL, 6 state changes
  DR is 0.0.0.0 BDR is 0.0.0.0
  Options is 0x00
  Dead timer due in 00:00:38
  Neighbor is up for 00:54:25
  TmrMax 2/2, retransmission queue length 0, number of retransmission 0
  First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
  Last retransmission scan length is 0, maximum is 1
  Last retransmission scan time is 0 msec, maximum is 0 msec
```

R3#

What is the neighbor priority shown for the OSPF neighbor routers? This value is the default. **Answer- 0**

c. Using the command prompt on **PC3**, ping the address of the **ISP Router** shown in the Address Table.

The screenshot shows a Windows-style window titled "Command Prompt" with the title bar "PC3". The menu bar includes "Physical", "Config", "Desktop", and "Programming". The main window displays the output of a ping command:

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.10.10

Pinging 192.168.10.10 with 32 bytes of data:

Reply from 192.168.10.10: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.10.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

Is it successful? **Answer Yes**

Part 2: Add the new Branch Office LAN to the OSPFv2 network.

You will now add the pre-configured Branch Office LAN to the OSPFv2 network.

Step 1: Verify the OSPFv2 configuration on router R4.

Execute a **show run | begin router ospf** command on router **R4**. Verify that the network statements are present for the networks that are configured on the router.

Which interface is configured to not send OSPF update packets? **Answer** Interface GigabitEthernet0/0/1

Step 2: Connect the Branch Office router R4 to the OSPFv2 network.

- a. Using the correct Ethernet cable, connect the G0/0/0 interface on router **R4** to the G0/1 interface on switch **S3**. Use the **show ip ospf neighbor** command to verify that router **R4** is now adjacent with router **R3**.

```
R4#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
3.3.3.3	1	FULL/DR	00:00:37	192.168.1.1	GigabitEthernet0/0/0

```
R4#
```

What state is displayed for router **R3**? Answer - FULL/DR

```
R4#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
3.3.3.3	1	FULL/DR	00:00:39	192.168.1.1	GigabitEthernet0/0/0

```
R4#
```

- b. Using the **show ip ospf neighbor** command on **R3**, determine the state of router **R4**. There may be a delay while OSPF converges.

Switch 1: Router-Forwarder (Process 10, Router 4.4.4.4) on gigabitethernet0/0 from interface to null, loading none, authentication source none						
Information		Statistics				
Neighbor ID	Port	Status	Dead Time	Address	Interface	Priority
0.0.0.0	1	DOWN	00:00:00	192.168.10.2	GigabitEthernet0/0	0
0.0.0.2	0	DOWN	00:00:01	192.168.10.1	Ethernet1/0/0/1	0
0.0.0.1	0	DOWN	00:00:01	192.168.10.1	Ethernet1/0/0/0	0
0.0.0.0						

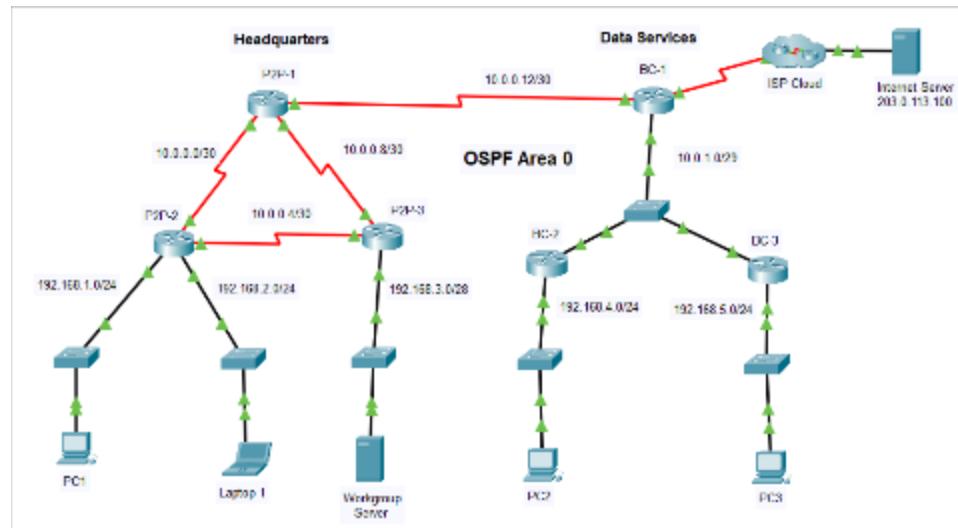
Why is the state of router R4 different than the state of R1 and R2?

Answer - Because the OSPF network type between R1 and R2 is Point-to-Point, there is no OSPF election. R4 is on the same Ethernet network segment as router R3, so the OSPF network type is Broadcast and there is an OSPF election. When more than one router is located on a multiaccess network segment, only one router, the DR, sends OSPFv2 updates. A second router, in this case R4, becomes the Backup Designated Router and can take over if the DR router fails.

- c. Using the command prompt on Laptop, ping the address of PC2.
Is it successful? Yes

3.2.6 Exercise 2.7.1 - Packet Tracer - Single-Area OSPFv2 Configuration

3.2.6.1 Topology



3.2.6.2 Addressing Table

Device	Interface	IP Address / Prefix
P2P-1	S0/1/0	10.0.0.1/30
	S0/1/1	10.0.0.9/30
	S0/2/0	10.0.0.13/30
P2P-2	S0/1/0	10.0.0.2/30
	S0/1/1	10.0.0.5/30
	G0/0/0	192.168.1.1/24
	G0/0/1	192.168.2.1/24
P2P-3	S0/1/0	10.0.0.6/30
	S0/1/1	10.0.0.10/30
	G0/0/0	192.168.3.1/28
BC-1	S0/1/0	10.0.0.14/30
	S0/1/1	64.0.100.2/30

	G0/0/0	10.0.1.1/29
BC-2	G0/0/0	192.168.4.1/30
	G0/0/1	10.0.1.2/29
BC-3	G0/0/0	192.168.5.1/24
	G0/0/1	10.0.1.3/29
Internet Server	NIC	203.0.113.100/24
PC 1	NIC	192.168.1.10/24
Laptop 1	NIC	192.168.2.20/24
Workgroup Server	NIC	192.168.3.14/28
PC 2	NIC	192.168.4.40/24
PC 3	NIC	192.168.5.50/24

3.2.6.3 Objectives

Implement single-area OSPFv2 in both point-to-point and broadcast multiaccess networks.

3.2.6.4 Background

You are helping a network engineer test an OSPF set up by building the network in the lab where you work. You have interconnected the devices and configured the interfaces and have connectivity within the local LANs. Your job is to complete the OSPF configuration according to the requirements left by the engineer.

Use the information provided and the list of requirements to configure the test network. When the task has been successfully completed, all hosts should be able to ping the internet server.

3.2.6.5 Instructions

Configure the network to meet the requirements.

3.2.6.6 Requirements

Use process ID 10 for OSPF activation on all routers.

- Activate OSPF using network statements and inverse masks on the **routers** in the Headquarters network.

```
P2P-1(config)#router ospf 10
P2P-1(config-router)#network 10.0.0.0 0.0.0.3 area 0
P2P-1(config-router)#network 10.0.0.8 0.0.0.3 area 0
P2P-1(config-router)#network 10.0.0.12 0.0.0.3 area 0
```

```
P2P-2(config)#router ospf 10
P2P-2(config-router)#network 10.0.0.0 0.0.0.3 area 0
P2P-2(config-router)#network 10.0.0.4 0.0.0.3 area 0
P2P-2(config-router)#network 192.168.1.0 0.0.0.255 area 0
P2P-2(config-router)#network 192.168.2.0 0.0.0.255 area 0
```

```
P2P-3(config)#router ospf 10
P2P-3(config-router)#network 10.0.0.4 0.0.0.3 area 0
P2P-3(config-router)#network 10.0.0.8 0.0.0.3 area 0
P2P-3(config-router)#network 192.168.3.0 0.0.0.15 area 0
```

- Activate OSPF by configuring the interfaces of the network devices in the Data Service network, where required.

```
BC-1(config)#interface GigabitEthernet0/0/0
BC-1(config-if)#ip ospf 10 area 0
```

```
BC-1(config-if)#interface Serial0/1/0
BC-1(config-if)#ip ospf 10 area 0
```

```
BC-2(config)#interface GigabitEthernet0/0/0
BC-2(config-if)#ip ospf 10 area 0
```

```
BC-2(config-if)#interface GigabitEthernet0/0/1
BC-2(config-if)#ip ospf 10 area 0
```

```
BC-3(config)#interface GigabitEthernet0/0/0
BC-3(config-if)#ip ospf 10 area 0
```

```
BC-3(config-if)#interface GigabitEthernet0/0/1
BC-3(config-if)#ip ospf 10 area 0
```

- Configure router IDs on the multiaccess network routers as follows:

- o BC-1: 6.6.6.6
- o BC-2: 5.5.5.5
- o BC-3: 4.4.4.4

```
BC-1(config)#router ospf 10
```

```
BC-1(config-router)#router-id 6.6.6.6
```

```
BC-2(config)#router ospf 10  
BC-2(config-router)#router-id 5.5.5.5
```

```
BC-3(config)#router ospf 10  
BC-3(config-router)#router-id 4.4.4.4
```

- Configure OSPF so that routing updates are not sent into networks where they are not required.

```
P2P-2(config)#router ospf 10  
P2P-2(config-router)#passive-interface GigabitEthernet0/0/0  
P2P-2(config-router)#passive-interface GigabitEthernet0/0/1
```

```
P2P-3(config)#router ospf 10  
P2P-3(config-router)#passive-interface GigabitEthernet0/0/0
```

```
BC-1(config)#router ospf 10  
BC-1(config-router)#passive-interface Serial0/1/1
```

```
BC-2(config)#router ospf 10  
BC-2(config-router)#passive-interface GigabitEthernet0/0/0
```

```
BC-3(config)#router ospf 10  
BC-3(config-router)#passive-interface GigabitEthernet0/0/0  
Configure router BC-1 with the highest OSPF interface priority so that it will always be the designated router of the multiaccess network.
```

```
BC-1(config)#interface GigabitEthernet0/0/0
```

```
BC-1(config-if)#ip ospf priority 255
```

```
Configure a default route to the ISP cloud using the exit interface command argument.
```

```
BC-1(config)#ip route 0.0.0.0 0.0.0.0 Serial0/1/1
```

- Automatically distribute the default route to all routers in the network.

```
BC-1(config)#router ospf 10  
BC-1(config-router)#default-information originate
```

- Configure the OSPF routers so that the Gigabit Ethernet interface cost will be 10 and the Fast Ethernet cost will be 100.

```
P2P-1(config)#router ospf 10  
P2P-1(config-router)#auto-cost reference-bandwidth 1000
```

```
P2P-2(config)#router ospf 10  
P2P-2(config-router)#auto-cost reference-bandwidth 1000  
P2P-3(config)#router ospf 10  
P2P-3(config-router)#auto-cost reference-bandwidth 1000
```

```
BC-1(config)#router ospf 10
BC-1(config-router)#auto-cost reference-bandwidth 1000
```

```
BC-2(config)#router ospf 10
BC-2(config-router)#auto-cost reference-bandwidth 1000
BC-3(config)#router ospf 10
BC-3(config-router)#auto-cost reference-bandwidth 1000
```

- Configure the OSPF cost value of P2P-1 interface Serial0/1/1 to 50.

```
P2P-1(config)#interface Serial0/1/1
P2P-1(config-if)#ip ospf cost 50
```

- Configure the hello and dead timer values on the interfaces that connect P2P-1 and BC-1 to be twice the default values.

```
P2P-1(config)#interface Serial0/2/0
P2P-1(config-if)#ip ospf hello-interval 20
P2P-1(config-if)#ip ospf dead-interval 80
```

```
BC-1(config)#interface Serial0/1/0
BC-1(config-if)#ip ospf hello-interval 20
BC-1(config-if)#ip ospf dead-interval 80
```

Scripts

```
!!!P2P-1 Configuration
enable
configure terminal
! Interface Configuration
interface Serial0/1/0
  ip address 10.0.0.1 255.255.255.252
  no shutdown
interface Serial0/1/1
  ip address 10.0.0.9 255.255.255.252
  ip ospf cost 50
  no shutdown
interface Serial0/2/0
  ip address 10.0.0.13 255.255.255.252
  ip ospf hello-interval 20
  ip ospf dead-interval 80
  no shutdown
! OSPF Configuration
router ospf 10
  network 10.0.0.0 0.0.0.3 area 0
  network 10.0.0.8 0.0.0.3 area 0
  network 10.0.0.12 0.0.0.3 area 0
  auto-cost reference-bandwidth 1000
  log-adjacency-changes
exit
end
```

```
!!!! P2P-2 Configuration
enable
configure terminal
! Interface Configuration
interface Serial0/1/0
ip address 10.0.0.2 255.255.255.252
no shutdown
interface Serial0/1/1
ip address 10.0.0.5 255.255.255.252
no shutdown
interface GigabitEthernet0/0/0
ip address 192.168.1.1 255.255.255.0
no shutdown
interface GigabitEthernet0/0/1
ip address 192.168.2.1 255.255.255.0
no shutdown
! OSPF Configuration
router ospf 10
network 10.0.0.0 0.0.0.3 area 0
network 10.0.0.4 0.0.0.3 area 0
network 192.168.1.0 0.0.0.255 area 0
network 192.168.2.0 0.0.0.255 area 0
passive-interface GigabitEthernet0/0/0
passive-interface GigabitEthernet0/0/1
auto-cost reference-bandwidth 1000
log-adjacency-changes
exit
end
```

```
!!!!P2P-3 Configuration
enable
configure terminal
! Interface Configuration
interface Serial0/1/0
ip address 10.0.0.6 255.255.255.252
no shutdown
interface Serial0/1/1
ip address 10.0.0.10 255.255.255.252
no shutdown
interface GigabitEthernet0/0/0
ip address 192.168.3.1 255.255.255.240
no shutdown
! OSPF Configuration
router ospf 10
network 10.0.0.4 0.0.0.3 area 0
network 10.0.0.8 0.0.0.3 area 0
network 192.168.3.0 0.0.0.15 area 0
passive-interface GigabitEthernet0/0/0
auto-cost reference-bandwidth 1000
log-adjacency-changes
exit
end
```

```
!!! BC-1 Configuration
enable
configure terminal
! Interface Configuration
interface Serial0/1/0
ip address 10.0.0.14 255.255.255.252
ip ospf hello-interval 20
ip ospf dead-interval 80
ip ospf 10 area 0
no shutdown
interface Serial0/1/1
ip address 64.0.100.2 255.255.255.252
no shutdown
interface GigabitEthernet0/0/0
ip address 10.0.1.1 255.255.255.248
ip ospf 10 area 0
ip ospf priority 255
no shutdown
! OSPF and Default Route Configuration
router ospf 10
router-id 6.6.6.6
passive-interface Serial0/1/1
auto-cost reference-bandwidth 1000
default-information originate
log-adjacency-changes
exit
ip route 0.0.0.0 0.0.0.0 Serial0/1/1
end
```

```
!!!BC-2 Configuration
enable
configure terminal
! Interface Configuration
interface GigabitEthernet0/0/0
ip address 192.168.4.1 255.255.255.252
ip ospf 10 area 0
no shutdown
interface GigabitEthernet0/0/1
ip address 10.0.1.2 255.255.255.248
ip ospf 10 area 0
no shutdown
! OSPF Configuration
router ospf 10
router-id 5.5.5.5
passive-interface GigabitEthernet0/0/0
auto-cost reference-bandwidth 1000
log-adjacency-changes
exit
end
```

```
!!!!BC-3 Configuration
```

```

enable
configure terminal
! Interface Configuration
interface GigabitEthernet0/0/0
ip address 192.168.5.1 255.255.255.0
ip ospf 10 area 0
no shutdown
interface GigabitEthernet0/0/1
ip address 10.0.1.3 255.255.255.248
ip ospf 10 area 0
no shutdown
! OSPF Configuration
router ospf 10
router-id 4.4.4.4
passive-interface GigabitEthernet0/0/0
auto-cost reference-bandwidth 1000
log-adjacency-changes
exit
end

```

Test

- **OSPF Neighbors:** Use show ip ospf neighbor to ensure all routers have established adjacencies.

```

P2P-1#show ip ospf neighbor

Neighbor ID      Pri   State        Dead Time     Address          Interface
192.168.2.1      0     FULL/       -              00:00:37    10.0.0.2        Serial0/1/0
192.168.3.1      0     FULL/       -              00:00:37    10.0.0.10       Serial0/1/1
6.6.6.6          0     FULL/       -              00:01:17    10.0.0.14       Serial0/2/0
P2P-1#

```

```

P2P-2#show ip ospf neighbor

Neighbor ID      Pri   State        Dead Time     Address          Interface
10.0.0.13         0     FULL/       -              00:00:33    10.0.0.1        Serial0/1/0
192.168.3.1      0     FULL/       -              00:00:33    10.0.0.6        Serial0/1/1
P2P-2#

```

```

P2P-3#show ip ospf neighbor

Neighbor ID      Pri   State        Dead Time     Address          Interface
192.168.2.1      0     FULL/       -              00:00:38    10.0.0.5        Serial0/1/0
10.0.0.13         0     FULL/       -              00:00:38    10.0.0.9        Serial0/1/1
P2P-3#

```

```
BC-1#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
5.5.5.5	1	FULL/BDR	00:00:38	10.0.1.2	GigabitEthernet0/0/0
4.4.4.4	1	FULL/DROTHER	00:00:38	10.0.1.3	GigabitEthernet0/0/0
10.0.0.13	0	FULL/-	00:01:18	10.0.0.13	Serial0/1/0
BC-1#					

```
BC-2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
4.4.4.4	1	FULL/DROTHER	00:00:36	10.0.1.3	GigabitEthernet0/0/1
6.6.6.6	255	FULL/DR	00:00:36	10.0.1.1	GigabitEthernet0/0/1
BC-2#					

```
DC-3#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
5.5.5.5	1	FULL/BDR	00:00:31	10.0.1.2	GigabitEthernet0/0/1
6.6.6.6	255	FULL/DR	00:00:31	10.0.1.1	GigabitEthernet0/0/1
DC-3#					

- **Routing Table:** Use show ip route to verify that OSPF routes are being propagated correctly.

```
--- ---  
P2P-1#show ip route  
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 10.0.0.14 to network 0.0.0.0
```

```
    10.0.0.0/8 is variably subnetted, 8 subnets, 3 masks  
C      10.0.0.0/30 is directly connected, Serial0/1/0  
L      10.0.0.1/32 is directly connected, Serial0/1/0  
O      10.0.0.4/30 [110/697] via 10.0.0.10, 00:08:12, Serial0/1/1  
C      10.0.0.8/30 is directly connected, Serial0/1/1  
L      10.0.0.9/32 is directly connected, Serial0/1/1  
C      10.0.0.12/30 is directly connected, Serial0/2/0  
L      10.0.0.13/32 is directly connected, Serial0/2/0  
O      10.0.1.0/29 [110/648] via 10.0.0.14, 00:08:12, Serial0/2/0  
O      192.168.1.0/24 [110/648] via 10.0.0.2, 00:08:12, Serial0/1/0  
O      192.168.2.0/24 [110/648] via 10.0.0.2, 00:08:12, Serial0/1/0  
      192.168.3.0/28 is subnetted, 1 subnets  
O      192.168.3.0/28 [110/51] via 10.0.0.10, 00:08:12, Serial0/1/1  
      192.168.4.0/30 is subnetted, 1 subnets  
O      192.168.4.0/30 [110/649] via 10.0.0.14, 00:08:12, Serial0/2/0  
O      192.168.5.0/24 [110/649] via 10.0.0.14, 00:08:12, Serial0/2/0  
O*E2 0.0.0.0/0 [110/1] via 10.0.0.14, 00:08:12, Serial0/2/0
```

```
P2P-1#
```

```

P2P-2#show ip route
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 10.0.0.1 to network 0.0.0.0

  10.0.0.0/8 is variably subnetted, 7 subnets, 3 masks
C    10.0.0.0/30 is directly connected, Serial0/1/0
L    10.0.0.2/32 is directly connected, Serial0/1/0
C    10.0.0.4/30 is directly connected, Serial0/1/1
L    10.0.0.5/32 is directly connected, Serial0/1/1
O    10.0.0.8/30 [110/697] via 10.0.0.1, 00:09:02, Serial0/1/0
O    10.0.0.12/30 [110/1294] via 10.0.0.1, 00:09:02, Serial0/1/0
O    10.0.1.0/28 [110/1295] via 10.0.0.1, 00:08:52, Serial0/1/0
  192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C    192.168.1.0/24 is directly connected, GigabitEthernet0/0/0
L    192.168.1.1/32 is directly connected, GigabitEthernet0/0/0
  192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C    192.168.2.0/24 is directly connected, GigabitEthernet0/0/1
L    192.168.2.1/32 is directly connected, GigabitEthernet0/0/1
  192.168.3.0/28 is subnetted, 1 subnets
O    192.168.3.0/28 [110/648] via 10.0.0.6, 00:09:02, Serial0/1/1
  192.168.4.0/30 is subnetted, 1 subnets
O    192.168.4.0/30 [110/1296] via 10.0.0.1, 00:08:52, Serial0/1/0
O    192.168.5.0/24 [110/1296] via 10.0.0.1, 00:08:52, Serial0/1/0
O*E2 0.0.0.0/0 [110/1] via 10.0.0.1, 00:08:52, Serial0/1/0

```

P2P-2#

```

P2P-3#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter areas
      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 areas
      * - candidate default, U - per user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is 10.0.0.9 to network 0.0.0.0

  10.0.0.0/8 is variably subnetted, 7 subnets, 3 masks
O    10.0.0.0/30 [110/1294] via 10.0.0.9, 00:09:39, Serial0/1/1
      [110/1294] via 10.0.0.5, 00:09:39, Serial0/1/0
C    10.0.0.4/30 is directly connected, Serial0/1/0
L    10.0.0.6/32 is directly connected, Serial0/1/0
C    10.0.0.8/30 is directly connected, Serial0/1/1
L    10.0.0.10/32 is directly connected, Serial0/1/1
O    10.0.0.12/30 [110/1294] via 10.0.0.9, 00:09:39, Serial0/1/1
O    10.0.1.0/28 [110/1295] via 10.0.0.9, 00:09:29, Serial0/1/1
O    192.168.1.0/24 [110/648] via 10.0.0.5, 00:09:39, Serial0/1/0
O    192.168.2.0/24 [110/648] via 10.0.0.5, 00:09:39, Serial0/1/0
  192.168.3.0/24 is variably subnetted, 2 subnets, 2 masks
C    192.168.3.0/28 is directly connected, GigabitEthernet0/0/0
L    192.168.3.1/32 is directly connected, GigabitEthernet0/0/0
  192.168.4.0/30 is subnetted, 1 subnets
O    192.168.4.0/30 [110/1296] via 10.0.0.9, 00:09:29, Serial0/1/1
O    192.168.5.0/24 [110/1296] via 10.0.0.9, 00:09:29, Serial0/1/1
O*E2 0.0.0.0/0 [110/1] via 10.0.0.9, 00:09:29, Serial0/1/1

```

P2P-3#

```

BC-1#show ip route
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
      1  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 0.0.0.0 to network 0.0.0.0

  10.0.0.0/8 is variably subnetted, 7 subnets, 3 masks
O   10.0.0.0/30 [110/1294] via 10.0.0.13, 00:10:15, Serial0/1/0
O   10.0.0.4/30 [110/1344] via 10.0.0.13, 00:10:15, Serial0/1/0
O   10.0.0.8/30 [110/697] via 10.0.0.13, 00:10:15, Serial0/1/0
C   10.0.0.12/30 is directly connected, Serial0/1/0
L   10.0.0.14/32 is directly connected, Serial0/1/0
C   10.0.1.0/29 is directly connected, GigabitEthernet0/0/0
L   10.0.1.1/32 is directly connected, GigabitEthernet0/0/0
  64.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C     64.0.100.0/30 is directly connected, Serial0/1/1
L     64.0.100.2/32 is directly connected, Serial0/1/1
O   192.168.1.0/24 [110/1295] via 10.0.0.13, 00:10:15, Serial0/1/0
O   192.168.2.0/24 [110/1295] via 10.0.0.13, 00:10:15, Serial0/1/0
    192.168.3.0/28 is subnetted, 1 subnets
O     192.168.3.0/28 [110/698] via 10.0.0.13, 00:10:15, Serial0/1/0
  192.168.4.0/30 is subnetted, 1 subnets
O     192.168.4.0/30 [110/2] via 10.0.1.2, 00:10:25, GigabitEthernet0/0/0
O   192.168.5.0/24 [110/2] via 10.0.1.3, 00:10:25, GigabitEthernet0/0/0
S*  0.0.0.0/0 is directly connected, Serial0/1/1

```

```

BC-2#show ip route
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
      1  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 10.0.1.1 to network 0.0.0.0

  10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
O   10.0.0.0/30 [110/1295] via 10.0.1.1, 00:10:44, GigabitEthernet0/0/1
O   10.0.0.4/30 [110/1345] via 10.0.1.1, 00:10:44, GigabitEthernet0/0/1
O   10.0.0.8/30 [110/698] via 10.0.1.1, 00:10:44, GigabitEthernet0/0/1
O   10.0.0.12/30 [110/640] via 10.0.1.1, 00:11:19, GigabitEthernet0/0/1
C   10.0.1.0/29 is directly connected, GigabitEthernet0/0/1
L   10.0.1.2/32 is directly connected, GigabitEthernet0/0/1
O   192.168.1.0/24 [110/1296] via 10.0.1.1, 00:10:44, GigabitEthernet0/0/1
O   192.168.2.0/24 [110/1296] via 10.0.1.1, 00:10:44, GigabitEthernet0/0/1
    192.168.3.0/28 is subnetted, 1 subnets
O     192.168.3.0/28 [110/699] via 10.0.1.1, 00:10:44, GigabitEthernet0/0/1
  192.168.4.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.4.0/30 is directly connected, GigabitEthernet0/0/0
L     192.168.4.1/32 is directly connected, GigabitEthernet0/0/0
O   192.168.5.0/24 [110/2] via 10.0.1.3, 00:11:19, GigabitEthernet0/0/1
O*E2 0.0.0.0/0 [110/1] via 10.0.1.1, 00:11:19, GigabitEthernet0/0/1

```

```

BC-3#show ip route
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 10.0.1.1 to network 0.0.0.0

  10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
O   10.0.0.0/30 [110/1295] via 10.0.1.1, 00:11:09, GigabitEthernet0/0/1
O   10.0.0.4/30 [110/1345] via 10.0.1.1, 00:11:09, GigabitEthernet0/0/1
O   10.0.0.8/30 [110/698] via 10.0.1.1, 00:11:09, GigabitEthernet0/0/1
O   10.0.0.12/30 [110/648] via 10.0.1.1, 00:11:19, GigabitEthernet0/0/1
C   10.0.1.0/29 is directly connected, GigabitEthernet0/0/1
L   10.0.1.3/32 is directly connected, GigabitEthernet0/0/1
O   192.168.1.0/24 [110/1296] via 10.0.1.1, 00:11:09, GigabitEthernet0/0/1
O   192.160.2.0/24 [110/1296] via 10.0.1.1, 00:11:09, GigabitEthernet0/0/1
  192.160.3.0/28 is subnetted, 1 subnets
O     192.168.3.0/28 [110/699] via 10.0.1.1, 00:11:09, GigabitEthernet0/0/1
  192.168.4.0/30 is subnetted, 1 subnets
O     192.160.4.0/30 [110/2] via 10.0.1.2, 00:11:19, GigabitEthernet0/0/1
  192.160.5.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.5.0/24 is directly connected, GigabitEthernet0/0/0
L     192.168.5.1/32 is directly connected, GigabitEthernet0/0/0
O*E2 0.0.0.0/0 [110/1] via 10.0.1.1, 00:11:19, GigabitEthernet0/0/1

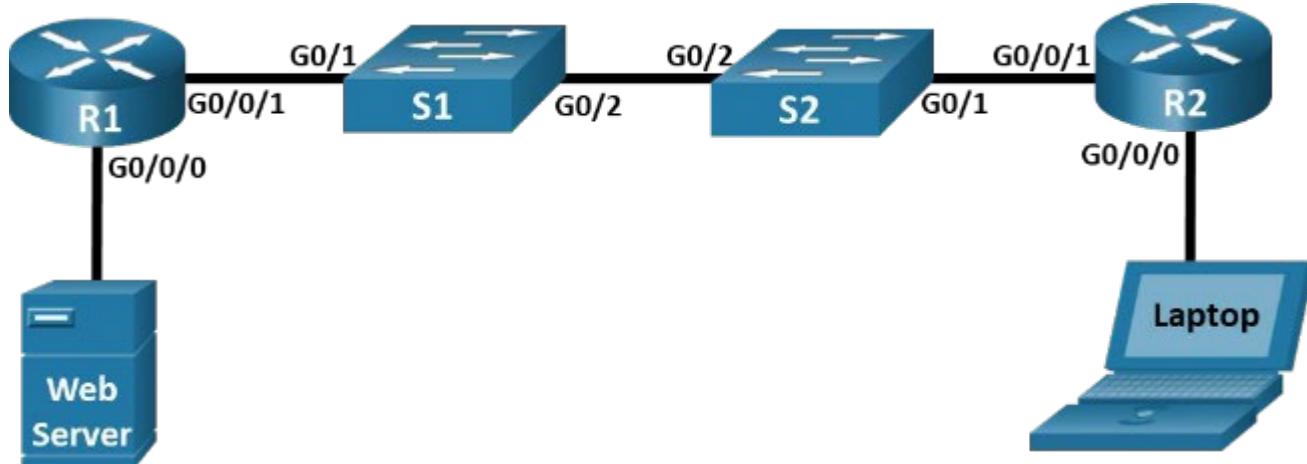
```

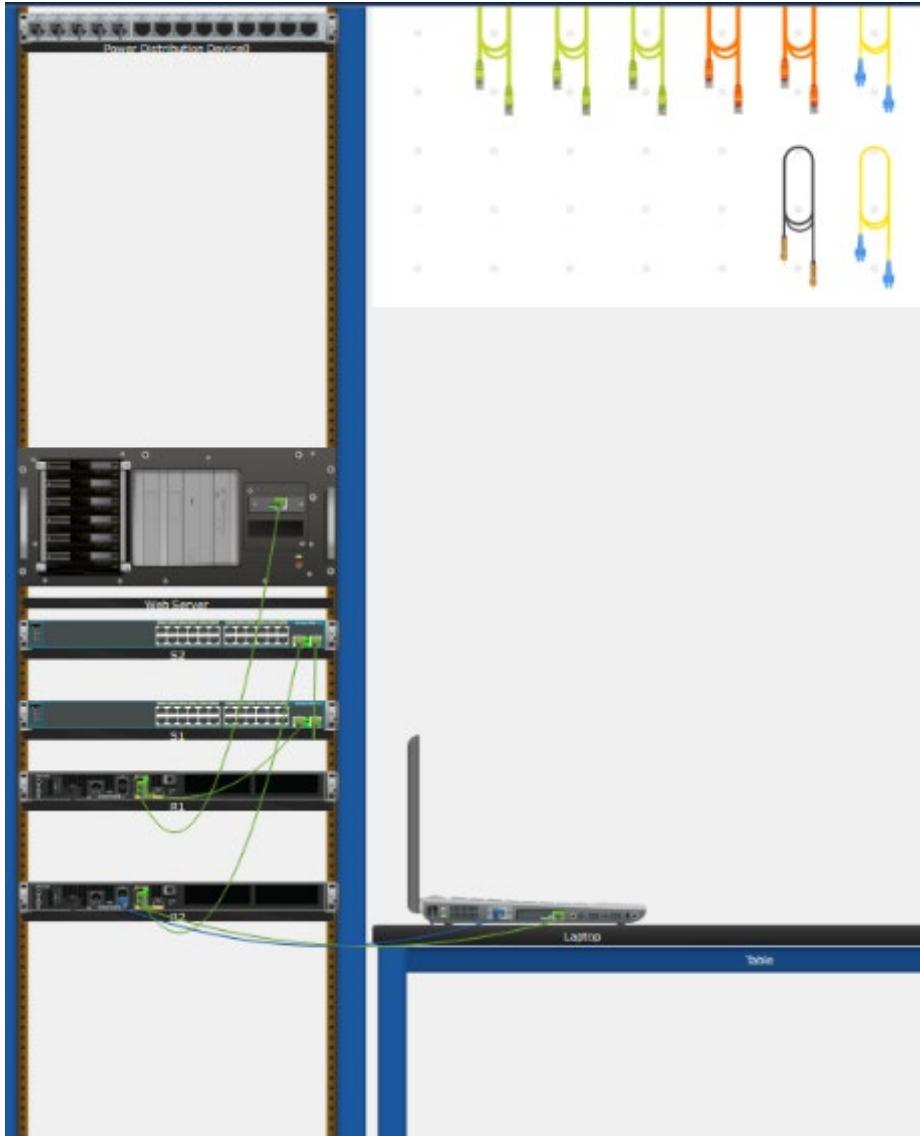
- Default Route:** Ensure the default route is being advertised by **BC-1** and received by other routers.

Gateway of last resort is 0.0.0.0 to network 0.0.0.0

3.2.7 Exercise 2.7.2 - Packet Tracer - Configure Single-Area OSPFv2 - Physical Mode

3.2.7.1 Topology





3.2.7.2 Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0/1	10.53.0.1	255.255.255.0	n/a
	G0/0/0	172.16.1.1	255.255.255.0	n/a
R2	G0/0/1	10.53.0.2	255.255.255.0	n/a
	G0/0/0	192.168.1.1	255.255.255.0	n/a
Web Server	F0	172.16.1.10	255.255.255.0	172.16.1.1
Laptop	F0	192.168.1.10	255.255.255.0	192.168.1.1

3.2.7.3 Objectives

Part 1: Build the Network and Configure Basic Device Settings

Part 2: Configure and Verify Single-Area OSPFv2 for Basic Operation Part

Part 3: Optimize and Verify the Single-Area OSPFv2 Configuration

3.2.7.4 Background/Scenario

You have been tasked with configuring a small company's network using OSPFv2. R1 will share the default route information to R2. After the initial configuration, the organization has asked for the configuration to be optimized to reduce protocol traffic and ensure that R1 remains in control of routing.

Note: The equipment required for this activity is located in the wiring closet on the utility shelf.

3.2.7.5 Instructions

Part 1: Build the Network and Configure Basic Device Settings

Step 1: Cable the network as shown in the topology.

Place the required devices on the rack and the table. Power on the PCs and cable the devices according to the topology. To select the correct port on a switch, right click and select **Inspect Front**. Use the Zoom tool, if necessary. Float your mouse over the ports to see the port numbers. Packet Tracer will score the correct cable and port connections.

- a. There are several switches, routers, and other devices on the **Shelf**. Click and drag the routers **R1** and **R2** and the switches **S1** and **S2** to the **Rack**. Click and drag the **Web Server** to the **Rack**. Click and drag the **Laptop** to the **Table**.
- b. Power on the routers and the laptop.
- c. On the **Cable Pegboard**, click a **Copper Straight-Through** cable. Click the **GigabitEthernet0/1** port on **S1** and then click the **GigabitEthernet0/0/1** port on **R1** to connect them.
- d. On the **Cable Pegboard**, click a **Copper Straight-Through** cable. Click the **GigabitEthernet0/1** port on **S2** and then click the **GigabitEthernet0/0/1** port on **R2** to connect them.
- e. On the **Cable Pegboard**, click a **Copper Cross-Over** cable. Click the **GigabitEthernet0/2** port on **S1** and then click the **GigabitEthernet0/2** port on **S2** to connect them. You should see the cable connecting the two ports.
- f. On the **Cable Pegboard**, click a **Copper Straight-Through** cable. Click the **GigabitEthernet0/0/0** port on **R1** and then click the **FastEthernet0** port on the **Web Server** to connect them.
- g. On the **Cable Pegboard**, click a **Copper Straight-Through** cable. Click the **GigabitEthernet0/0/0** port on **R2** and then click the **FastEthernet0** port on the **Laptop** to connect them.

Visually inspect network connections. Initially, when you connect devices to a switch port, the link lights will be amber. After a minute or so, the link lights will turn green.

Step 2: Configure basic settings for the two routers and two switches.

- a. On the **Cable Pegboard**, click a **Console** cable.
- b. Connect the console cable between the device and the **Laptop**. For the switches, **Inspect Rear** to locate the **Console** port.
- c. Assign a name to the device according to the **Topology**.
- d. Disable DNS lookup to prevent the router from attempting to translate incorrectly entered commands as though they were host names.
- e. Assign **class** as the privileged EXEC encrypted password.
- f. Assign **cisco** as the console password and enable login.
- g. Assign **cisco** as the vty password and enable login.
- h. Encrypt the plaintext passwords.
- i. Create a banner that warns anyone accessing the device that unauthorized access is prohibited.
- j. Save the running configuration to the startup configuration file.
- k. Click one end of the **Console cable** and drag it back to the **Cable Pegboard**.
- l. Repeat Step 2 for each device until **R2**, **S1**, and **S2** are also each configured with basic settings.

Step 3: Configure settings for the server and the laptop.

Configure static IP address information on the **Web Server** and **Laptop** according to the **Addressing Table**.

- a. Click **Web Server > Desktop > IP Configuration**. Enter the IPv4 address, subnet mask, and default gateway information for the **Web Server** according to the **Addressing Table**.
- b. Close or minimize the **Web Server** window.
- c. Repeat the previous steps to assign the IPv4 address information for the **Laptop**, as listed in the **Addressing Table**.

Part 2: Configure and Verify Single-Area OSPFv2 for Basic Operation

Step 1: Configure interface addresses and basic OSPFv2 on each router.

- a. Connect a **Console** cable between **R1** and the **Laptop**.
- b. Configure interface addresses on each router as shown in the **Addressing Table**.
- c. Enter OSPF router configuration mode using process ID 56.
- d. Configure a static router ID for each router (1.1.1.1 for R1, 2.2.2.2 for R2).
- e. Configure a network statement for the network between R1 and R2, placing it in area 0.
- f. Configure a network statement for the other networks connected to R1 and R2 and place them in area 0. Note that the network command for the LAN connected to R1 will not be graded as this network is removed later in the activity.
- g. Switch the console cable to **R2** and repeat substeps b through f for **R2**. After configuring R1 and R2, you can simply use Telnet between them, if you wish, instead of moving the console cable each time.
- h. Verify that OSPFv2 is operational between the routers. Issue the command to verify that R1 and R2 have formed an adjacency.

Which router is identified as the DR? Which is the BDR? What was the selection criteria?

R1 was configured first and was speaking OSPF before R2. So during the OSPF election only R1 was configured for OSPF and became the DR. After R2 was configured for OSPF it became the BDR on the Gigabit segment. The router with the highest router ID is used in the selection of DR and BDR.

```
R1# show ip ospf neighbor

Neighbor ID      Pri  State        Dead Time   Address          Interface
2.2.2.2           1    FULL/BDR     00:01:30   10.53.0.2      GigabitEthernet0/0/1
R1#show ip route ospf
0    192.168.1.0 [110/2] via 10.53.0.2, 00:06:32, GigabitEthernet0/0/1

R1#show ip ospf interface g0/0/1

GigabitEthernet0/0/1 is up, line protocol is up
  Internet address is 10.53.0.1/24, Area 0
  Process ID 56, Router ID 1.1.1.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 10
  Designated Router (DR) 1.1.1.1, Interface address 10.53.0.1
  Backup Designated Router (BDR) 2.2.2.2, Interface address 10.53.0.2
  Timers intervals configured, Hello 30, Dead 120, Retransmit 5
    Hello due in 00:00:29
  Index 1/1, flood queue length 0
  Next Rx0/0/0x0()
  Last flood queue length is 1, maximum is 1
  Last flood span time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 2.2.2.2 (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
R1#
```

R2#show ip ospf neighbor

Neighbor ID	Pri	State	Dead Time	Address	Interface
1.1.1.1	50	FULL/DR	00:01:52	10.63.0.1	GigabitEthernet0/0/1

R2#show ip route ospf

Network	Link	Cost	Protocol	Next-Hop	Interface
0.0.0.0/0	[110/1]	0	OSPF	10.63.0.1	GigabitEthernet0/0/1

R2#

- i. On R1, issue the **show ip route ospf** command to verify that the R2 G0/0/0 network is present in the routing table.
- j. Click **Laptop > Command Prompt**, and then ping the **Web Server** at 172.16.1.10. After one or two timeouts, the ping should be successful. If not, troubleshoot your physical connections and configurations.

```
C:\>ping 172.16.1.10

Pinging 172.16.1.10 with 32 bytes of data:
Request timed out.
Reply from 172.16.1.10: bytes=32 time<1ms TTL=126
Reply from 172.16.1.10: bytes=32 time<1ms TTL=126
Reply from 172.16.1.10: bytes=32 time<1ms TTL=126

Ping statistics for 172.16.1.10:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

Part 3: Optimize the Single-Area OSPFv2 Configuration

Step 1: Implement various optimizations on each router.

- a. On R1, configure the interface G0/0/1 OSPF priority to 50 to ensure that **R1** is the **Designated Router**.
- b. Configure the OSPF timers on interface G0/0/1 of each router for a hello timer of 30 seconds and a dead timer of 120 seconds.
- c. On R1, remove the OSPF network command for 172.16.1.0, and then configure a default static route that uses **interface G0/0/0** as the exit interface. Finally, propagate the default route into OSPF. Note the console message after setting the default route.
- d. Change the reference bandwidth on each router to 1Gbs. After this configuration, restart OSPF using the **clear ip ospf process** command. Note the console message after setting the new reference bandwidth.

Step 2: Verify OSPFv2 optimizations are in place.

- a. Issue the **show ip ospf interface g0/0/1** command on **R1** and verify that the interface priority has been set to 50 and that the time intervals are Hello 30, Dead 120, and the default Network Type is Broadcast.
- b. On **R1**, issue the **show ip route ospf** command to verify that the R2 G0/0/0 network is present in the routing table. Note the difference in the metric between this output and the previous output.
- c. On **R2**, issue the **show ip route ospf** command. The only OSPF route information should be the default route that R1 is propagating.
- d. From the **Laptop**, ping the **Web Server** again. The ping should be successful.

Why is the OSPF cost for the default route different than the OSPF cost at R1 for the 192.168.1.0/24 network?

```
R1#show ip route ospf  
O 192.168.1.0 [110/2] via 10.53.0.2, 00:21:51, GigabitEthernet0/0/1  
R2#show ip route ospf  
O*E2 0.0.0.0/0 [110/1] via 10.53.0.1, 00:20:57, GigabitEthernet0/0/1
```

Why the Costs Are Different

- **192.168.1.0/24:**
 - Internal OSPF route.
 - Cost is cumulative: **2** (1 for each hop).
- **Default Route (0.0.0.0/0):**
 - External OSPF route (Type 2 - E2).
 - Cost is fixed: **1** (set during redistribution and does not change).

1. Internal Route (192.168.1.0/24)

- The **192.168.1.0/24** network is an **internal OSPF route** because it is advertised within the OSPF domain.
- OSPF calculates the cost for internal routes based on the **cumulative cost** of the path from the router to the destination network.
 - The cost is derived from the **bandwidth of the interfaces** along the path.
 - The formula for OSPF cost is:
$$\text{Cost} = \frac{\text{Reference Bandwidth}}{\text{Interface Bandwidth}}$$

Cost=Interface Bandwidth/Reference Bandwidth
(By default, the reference bandwidth is 100 Mbps.)
 - The cost is **cumulative**, meaning each hop adds its own cost to the total.

Example for 192.168.1.0/24:

- R1 learns about the **192.168.1.0/24** network via **R2**.
- The path from **R1** to **192.168.1.0/24** goes through the **10.53.0.0/24** network (GigabitEthernet0/0/1).
- The cost of the GigabitEthernet link is **1** (since the reference bandwidth is 100 Mbps and the interface bandwidth is 1 Gbps).
- Therefore, the total cost for the **192.168.1.0/24** network is **2**:
 - **1** for the link between R1 and R2 (10.53.0.0/24).
 - **1** for the link between R2 and the 192.168.1.0/24 network.

2. External Route (Default Route - 0.0.0.0/0)

- The **default route** (0.0.0.0/0) is an **external route** because it is **redistributed into OSPF** from a static route or another routing protocol.
- OSPF classifies external routes into two types:
 - Type 1 (E1):** The cost is cumulative, including the external cost and the internal OSPF cost to reach the ASBR (Autonomous System Boundary Router).
 - Type 2 (E2):** The cost is **only the external cost**, and it does not change as it is propagated through the OSPF domain.

Example for Default Route (0.0.0.0/0):

- R1 is configured with a **static default route** (ip route 0.0.0.0 0.0.0.0 g0/0/0) and propagates it into OSPF using the default-information originate command.
- By default, OSPF treats redistributed default routes as **Type 2 (E2)**.
- The cost of the default route is set to **1** when it is redistributed into OSPF.
- Since it is an **E2 route**, the cost remains **1** throughout the OSPF domain, regardless of the internal path cost.

Key Points

1. Internal Routes:

- Cost is cumulative.
- Based on the bandwidth of the interfaces along the path.
- Example: O 192.168.1.0 [110/2].

2. External Routes (Type 2 - E2):

- Cost is fixed and does not change as it propagates through the OSPF domain.
- Example: O*E2 0.0.0.0/0 [110/1].

3. Default Route Propagation:

- The default route is redistributed into OSPF as an external route.
- By default, it is treated as **Type 2 (E2)**, so its cost remains **1**.

Printouts

```
R2>enable
Password:
R2#show ip route
Codes: L - Local, C - static, R - RTP, M - mobile, B - BGP
      D - EIGRP, E - 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, # - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is 10.53.0.1 to network 0.0.0.0

  10.0.0.0/0 is variably subnetted, 2 subnets, 2 masks
C    10.53.0.0/24 is directly connected, GigabitEthernet0/0/1
L    10.53.0.2/32 is directly connected, GigabitEthernet0/0/1
  192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C    192.168.1.0/24 is directly connected, GigabitEthernet0/0/0
L    192.168.1.1/32 is directly connected, GigabitEthernet0/0/0
O*E2 0.0.0.0/0 [110/1] via 10.53.0.1, 00:20:00, GigabitEthernet0/0/1

R2#show ip route ospf
O*E2 0.0.0.0/0 [110/1] via 10.53.0.1, 00:20:57, GigabitEthernet0/0/1

R2#
```

```

R1>enable
Password:
R1#show ip route
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 0.0.0.0 to network 0.0.0.0

      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.53.0.0/24 is directly connected, GigabitEthernet0/0/1
L        10.53.0.1/32 is directly connected, GigabitEthernet0/0/1
      172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
C        172.16.1.0/24 is directly connected, GigabitEthernet0/0/0
L        172.16.1.1/32 is directly connected, GigabitEthernet0/0/0
O        192.168.1.0/24 [110/2] via 10.53.0.2, 00:21:49, GigabitEthernet0/0/1
S*      0.0.0.0/0 is directly connected, GigabitEthernet0/0/0

R1#show ip route ospf
O      192.168.1.0 [110/2] via 10.53.0.2, 00:21:51, GigabitEthernet0/0/1

R1#

```

R1#show ip ospf database external

OSPF Router with ID (1.1.1.1) (Process ID 56)

Type-5 AS External Link States

Routing Bit Set on this LSA
 LS age: 1686
 Options: (No TOS-capability, DC)
 LS Type: AS External Link
 Link State ID: 0.0.0.0 (External Network Number)
 Advertising Router: 1.1.1.1
 LS Seq Number: 80000001
 Checksum: 0fecf
 Length: 36
 Network Mask: /0
 Metric Type: 2 (Larger than any link state path)
 TOS: 0
 Metric: 1
 Forward Address: 0.0.0.0
 External Route Tag: 1

R1#

R2#show ip ospf database external

OSPF Router with ID (2.2.2.2) (Process ID 56)

Type-5 AS External Link States

Routing Bit Set on this LSA
LS age: 1676
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 0.0.0.0 (External Network Number)
Advertising Router: 1.1.1.1
LS Seq Number: 80000001
Checksum: 0xfcfc
Length: 36
Network Mask: /0
Metric Type: 2 (Larger than any link state path)
TOS: 0
Metric: 1
Forward Address: 0.0.0.0
External Route Tag: 1
R2#

3.2.7.6 Script

```
!!!! R1 Configuration Script
enable
configure terminal
hostname R1
no ip domain-lookup
enable secret class
line console 0
password cisco
login
exit
line vty 0 4
password cisco
login
exit
service password-encryption
banner motd $ Authorized Users Only! $
interface g0/0/1
ip address 10.53.0.1 255.255.255.0
no shutdown
exit
interface g0/0/0
ip address 172.16.1.1 255.255.255.0
no shutdown
exit
router ospf 56
router-id 1.1.1.1
network 10.53.0.0 0.0.0.255 area 0
network 172.16.1.0 0.0.0.255 area 0
exit
interface g0/0/1
ip ospf priority 50
```

```
ip ospf hello-interval 30
ip ospf dead-interval 120
exit
router ospf 56
  no network 172.16.1.0 0.0.0.255 area 0
exit
ip route 0.0.0.0 0.0.0.0 g0/0/0
router ospf 56
  default-information originate
exit
router ospf 56
  auto-cost reference-bandwidth 1000
exit
end
copy running-config startup-config
```

```
!!!! R2 Configuration Script
enable
configure terminal
hostname R2
no ip domain-lookup
enable secret class
line console 0
  password cisco
  login
exit
line vty 0 4
  password cisco
  login
exit
service password-encryption
banner motd $ Authorized Users Only! $
interface g0/0/1
ip address 10.53.0.2 255.255.255.0
  no shutdown
exit
interface g0/0/0
ip address 192.168.1.1 255.255.255.0
  no shutdown
exit
router ospf 56
  router-id 2.2.2.2
  network 10.53.0.0 0.0.0.255 area 0
  network 192.168.1.0 0.0.0.255 area 0
exit
interface g0/0/1
  ip ospf hello-interval 30
  ip ospf dead-interval 120
exit
router ospf 56
  auto-cost reference-bandwidth 1000
```

```
exit
end
copy running-config startup-config
```

```
!!! S1 Configuration Script
enable
configure terminal
hostname S1
no ip domain-lookup
enable secret class
line console 0
password cisco
login
exit
line vty 0 15
password cisco
login
exit
service password-encryption
banner motd $ Authorized Users Only! $
interface vlan 1
ip address 172.16.1.2 255.255.255.0
no shutdown
exit
ip default-gateway 172.16.1.1
end
copy running-config startup-config
```

```
!!! S2 Configuration Script
enable
configure terminal
hostname S2
no ip domain-lookup
enable secret class
line console 0
password cisco
login
exit
line vty 0 15
password cisco
login
exit
service password-encryption
banner motd $ Authorized Users Only! $
interface vlan 1
ip address 192.168.1.2 255.255.255.0
no shutdown
```

```
exit
ip default-gateway 192.168.1.1
end
copy running-config startup-config
```

Web Server Configuration

```
# Web Server Configuration Script
# Configure the IP address, subnet mask, and default gateway via the GUI or CLI.
# IP Configuration:
# IPv4 Address: 172.16.1.10
# Subnet Mask: 255.255.255.0
# Default Gateway: 172.16.1.1
```

Laptop Configuration

```
# Laptop Configuration Script
# Configure the IP address, subnet mask, and default gateway via the GUI or CLI.
# IP Configuration:
# IPv4 Address: 192.168.1.10
# Subnet Mask: 255.255.255.0
# Default Gateway: 192.168.1.1
```

3.2.7.7 Printouts

```
R1# show ip ospf neighbor

Neighbor ID      Pri  State        Dead Time   Address          Interface
2.2.2.2           1    FULL/BBR     00:01:30   10.53.0.2      GigabitEthernet0/0/1
R1#show ip route ospf
0    192.168.1.0 [110/2] via 10.53.0.2, 00:06:33, GigabitEthernet0/0/1

R1#show ip ospf interface g0/0/1

GigabitEthernet0/0/1 is up, line protocol is up
  Internet address is 10.53.0.1/24, Area 0
  Process ID 56, Router ID 1.1.1.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 10
  Designated Router (DR) 1.1.1.1, Interface address 10.53.0.1
  Backup Designated Router (BDR) 2.2.2.2, Interface address 10.53.0.2
  Timer intervals configured, Hello 30, Dead 120, Retransmit 5
    Hello due in 00:00:29
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood queue length is 1, maximum is 1
  Last flood span time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 2.2.2.2 (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
R1#
```

Laptop

Physical Config Desktop Programming Attributes

Terminal

```
R2#show ip ospf neighbor

Neighbor ID      Pri      State            Read Time    Address          Interface
1.1.1.1           50      FULL/DR          00:01:52    10.63.0.1      GigabitEthernet0/0/1
R2#show ip route ospf
0.0.0.0/0 [110/1] via 10.63.0.1, 00:07:22, GigabitEthernet0/0/1

R2#
```

Laptop

Physical Config Desktop Programming Attributes

Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 172.16.1.10

Pinging 172.16.1.10 with 32 bytes of data:

Request timed out.
Reply from 172.16.1.10: bytes=32 time<1ms TTL=126
Reply from 172.16.1.10: bytes=32 time<1ms TTL=126
Reply from 172.16.1.10: bytes=32 time<1ms TTL=126

Ping statistics for 172.16.1.10:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

Web Server

Physical Config Services Desktop Programming Attributes

Command Prompt

```
172.16.1.1
C:\>ping 192.168.1.10

Pinging 192.168.1.10 with 32 bytes of data:

Reply from 192.168.1.10: bytes=32 time<1ms TTL=126

Ping statistics for 192.168.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
C:\>
C:\>
```

3.2.7.8 Assessment

Activity Results

Congratulations Guest! You completed the activity.

Overall Feedback [Assessment Items](#) [Connectivity Tests](#)

[Expand/Collapse All](#) [Show Incorrect Items](#)

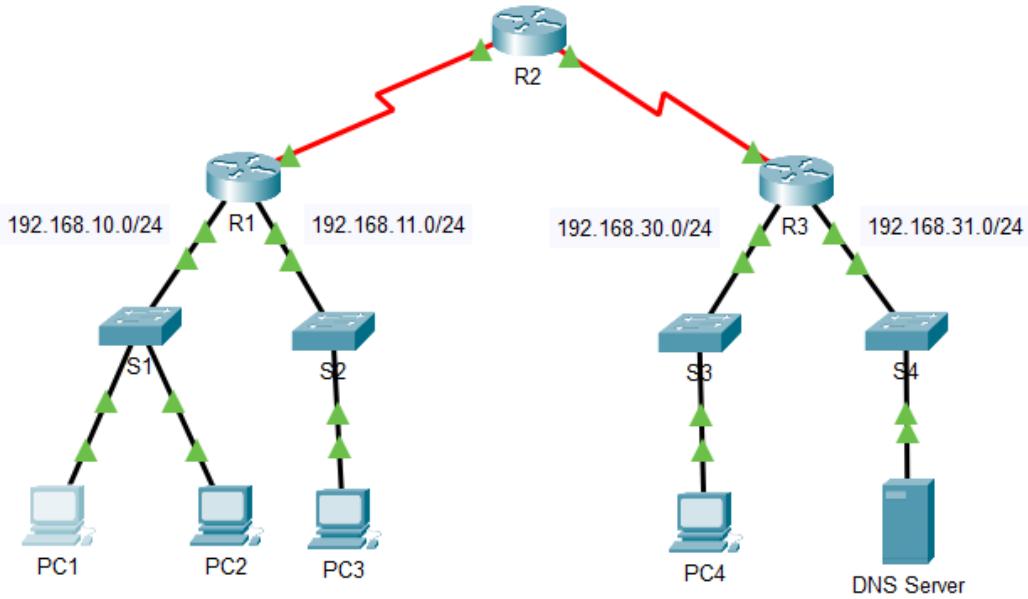
Assessment Item	Status	Points	Component(s)
- Network			
S-Laptop			
✓ Default Gateway	Correct	1	IP
✓ Physical Location	Correct	1	Physical
E- Ports			
E- FastEthernet0/0			
✓ IP Address	Correct	1	IP
E- Link to R2		0	Other
✓ Subnet Mask	Correct	1	IP
✓ Power	Correct	1	Physical
R1			
✓ Banner MOTD	Correct	1	Other
Console Line			
✓ Login	Correct	1	Physical
✓ Password	Correct	1	Other
DNS			
✓ IP Domain-lookup	Correct	1	Other
✓ Enable Secret	Correct	1	Other
✓ Host Name	Correct	1	Other
OSPF			
E- Process ID 10			
E- Area		0	Routing
E- Area 0		1	Routing
✓ Area Status	Correct	1	Routing
✓ Auto Cost	Correct	1	Routing
✓ Default Information	Correct	1	Routing
E- Networks		0	Routing
✓ Router0	Correct	1	Routing
✓ Router ID	Correct	1	Routing
✓ Physical Location	Correct	1	Physical
D- Ports			
E- GigabitEthernet0/0/0			
✓ IP Address	Correct	1	IP
✓ Port Status	Correct	1	Physical
✓ Subnet Mask	Correct	1	IP
E- GigabitEthernet0/0/1			
✓ IP Address	Correct	1	IP
✓ OSPF Dead Interval	Correct	1	Routing
✓ OSPF Hello-Interval	Correct	1	Routing
✓ OSPF Priority	Correct	1	Routing
✓ Port Status	Correct	1	Physical

Assessment Item	Status	Points	Component(s)	Feedback
- Network				
S-GigabitEthernet0/0/1				
✓ IP Address	Correct	1	IP	
✓ OSPF Dead-Interval	Correct	1	Routing	
✓ OSPF Hello-Interval	Correct	1	Routing	
✓ OSPF Priority	Correct	1	Routing	
✓ Port Status	Correct	1	Physical	
✓ Subnet Mask	Correct	1	IP	
✓ Power	Correct	1	Physical	
✓ Service Password Encryption	Correct	1	Other	
✓ Startup Config	Correct	1	Other	
E- VTY Lines				
E- VTY Line 4				
✓ Password	Correct	1	Other	
E- R2				
✓ Banner MOTD	Correct	1	Other	
Console Line				
✓ Login	Correct	1	Physical	
✓ Password	Correct	1	Other	
DNS				
✓ IP Domain-Lookup	Correct	1	Other	
✓ Enable Secret	Correct	1	Other	
✓ Host Name	Correct	1	Other	
OSPF				
E- Process ID 56				
E- Area		0	Routing	
E- Area 0		1	Routing	
✓ Area Status	Correct	1	Routing	
✓ Auto Cost	Correct	1	Routing	
✓ Default Information	Correct	1	Routing	
E- Networks				
✓ Router0	Correct	1	Routing	
✓ Router1	Correct	1	Routing	
✓ Router ID	Correct	1	Routing	
✓ Physical Location	Correct	1	Physical	
D- Ports				
E- GigabitEthernet0/0/0				
✓ IP Address	Correct	1	IP	
✓ Port Status	Correct	1	Physical	
✓ Subnet Mask	Correct	1	IP	

PARAMETER/STATUS		Status	Port(s)	COMPONENT(S)	TYPE/MODE
<input checked="" type="checkbox"/> Physical Location		Correct	1		Physical
<input checked="" type="checkbox"/> Ports					
<input checked="" type="checkbox"/> GigabitEthernet0/0					
<input checked="" type="checkbox"/> IP Address		Correct	1	IP	
<input checked="" type="checkbox"/> Port Status		Correct	1	Physical	
<input checked="" type="checkbox"/> Subnet Mask		Correct	1	IP	
<input checked="" type="checkbox"/> GigabitEthernet0/1					
<input checked="" type="checkbox"/> IP Address		Correct	1	IP	
<input checked="" type="checkbox"/> OSPF Dead Interval		Correct	1	Routing	
<input checked="" type="checkbox"/> OSPF Hello Interval		Correct	1	Routing	
<input checked="" type="checkbox"/> Port Status		Correct	1	Physical	
<input checked="" type="checkbox"/> Subnet Mask		Correct	1	IP	
<input checked="" type="checkbox"/> Power		Correct	1	Physical	
<input checked="" type="checkbox"/> Secure Password Encryption		Correct	1	Other	
<input checked="" type="checkbox"/> Startup Config		Correct	1	Other	
<input checked="" type="checkbox"/> VTY Lines					
<input checked="" type="checkbox"/> VTY Line 4			0	Other	
<input checked="" type="checkbox"/> Password		Correct	1	Other	
<input checked="" type="checkbox"/> S1					
<input checked="" type="checkbox"/> Banner MOTD		Correct	1	Other	
<input checked="" type="checkbox"/> Console Line					
<input checked="" type="checkbox"/> Login		Correct	1	Physical	
<input checked="" type="checkbox"/> Password		Correct	1	Other	
<input checked="" type="checkbox"/> DNS			0	Other	
<input checked="" type="checkbox"/> IP Domain Lookup		Correct	1	Other	
<input checked="" type="checkbox"/> Enable Secret		Correct	1	Other	
<input checked="" type="checkbox"/> Host Name		Correct	1	Other	
<input checked="" type="checkbox"/> Physical Location		Correct	1	Physical	
<input checked="" type="checkbox"/> Ports					
<input checked="" type="checkbox"/> GigabitEthernet0/1			0	Other	
<input checked="" type="checkbox"/> Link to R1			0	Other	
<input checked="" type="checkbox"/> Connects to GigabitEthernet0/1		Correct	1	Physical	
<input checked="" type="checkbox"/> GigabitEthernet0/2			0	Other	
<input checked="" type="checkbox"/> Link to S2			0	Other	
<input checked="" type="checkbox"/> Connects to GigabitEthernet0/2		Correct	1	Physical	
<input checked="" type="checkbox"/> Secure Password Encryption		Correct	1	Other	
<input checked="" type="checkbox"/> Startup Config		Correct	1	Other	
<input checked="" type="checkbox"/> VTY Lines			0	Other	
<input checked="" type="checkbox"/> VTY Line 15			0	Other	
<input checked="" type="checkbox"/> Password		Correct	1	Other	
<input checked="" type="checkbox"/> S2					
<input checked="" type="checkbox"/> Banner MOTD		Correct	1	Other	
<input checked="" type="checkbox"/> Console Line					
<input checked="" type="checkbox"/> Login		Correct	1	Physical	
<input checked="" type="checkbox"/> Password		Correct	1	Other	
<input checked="" type="checkbox"/> DNS			0	Other	
<input checked="" type="checkbox"/> IP Domain Lookup		Correct	1	Other	
<input checked="" type="checkbox"/> Enable Secret		Correct	1	Other	
<input checked="" type="checkbox"/> Host Name		Correct	1	Other	
<input checked="" type="checkbox"/> Physical Location		Correct	1	Physical	
<input checked="" type="checkbox"/> Ports			0	Other	
<input checked="" type="checkbox"/> GigabitEthernet0/1			0	Other	
<input checked="" type="checkbox"/> Link to R2			0	Other	
<input checked="" type="checkbox"/> Connects to GigabitEthernet0/1		Correct	1	Physical	
<input checked="" type="checkbox"/> Secure Password Encryption		Correct	1	Other	
<input checked="" type="checkbox"/> Startup Config		Correct	1	Other	
<input checked="" type="checkbox"/> VTY Lines			0	Other	
<input checked="" type="checkbox"/> VTY Line 15			0	Other	
<input checked="" type="checkbox"/> Password		Correct	1	Other	
<input checked="" type="checkbox"/> Web Server					
<input checked="" type="checkbox"/> Default Gateway		Correct	1	IP	
<input checked="" type="checkbox"/> Physical Location		Correct	1	Physical	
<input checked="" type="checkbox"/> Ports					
<input checked="" type="checkbox"/> GigabitEthernet0/0					
<input checked="" type="checkbox"/> IP Address		Correct	1	IP	
<input checked="" type="checkbox"/> Link to R1			0	Other	
<input checked="" type="checkbox"/> Connects to GigabitEthernet0/0		Correct	1	Physical	
<input checked="" type="checkbox"/> Subnet Mask		Correct	1	IP	
<input checked="" type="checkbox"/> Power		Correct	1	Physical	

3.2.8 Exercise 4.1.4 - Packet Tracer - Access Control List Demonstration

3.2.8.1 Topology



3.2.8.2 Objectives

Part 1: Verify Local Connectivity and Test Access Control List **Part 2: Remove Access Control List and Repeat Test**

3.2.8.3 Background

In this activity, you will observe how an access control list (ACL) can be used to prevent a ping from reaching hosts on remote networks. After removing the ACL from the configuration, the pings will be successful.

3.2.8.4 Addressing Table

Device	Interface	IP Address / Prefix
R1	G0/0	192.168.10.1/24
	G0/1	192.168.11.1/24
	S0/0/0	10.1.1.1/30
R2	S0/0/0	10.10.1.2/30
	S0/0/1	10.10.1.5/30
R3	G0/0	192.168.30.1/24
	G0/1	192.168.31.1/24

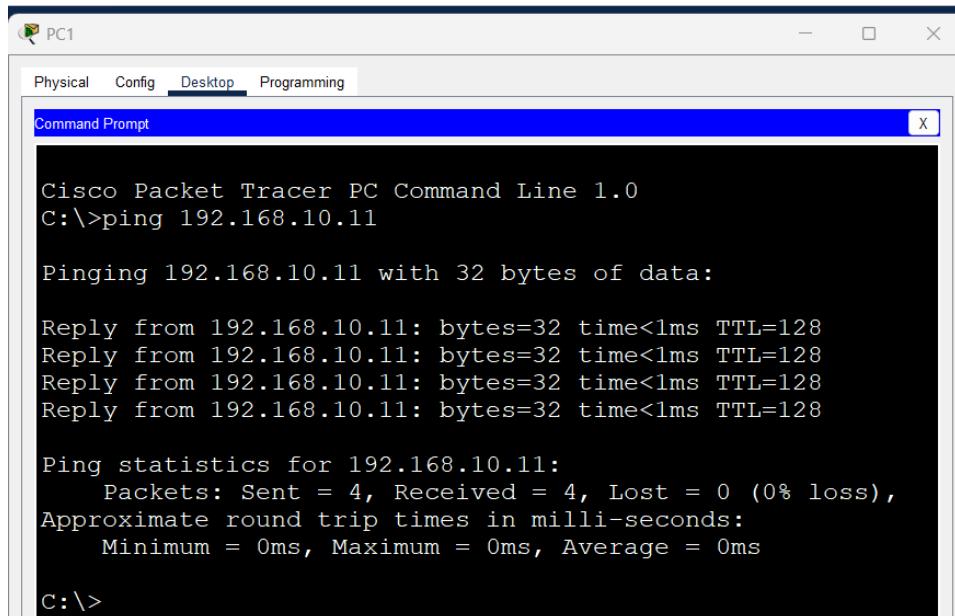
	S0/0/1	10.10.1.6/24
PC1	NIC	192.168.10.10/24
PC2	NIC	192.168.10.11/24
PC3	NIC	192.168.11.10/24
PC4	NIC	192.168.30.12/24
DNS Server	NIC	192.168.31.12/24

3.2.8.5 Instructions

Part 1: Verify Local Connectivity and Test Access Control List

Step 1: Ping devices on the local network to verify connectivity.

- a. From the command prompt of **PC1**, ping **PC2**.



The screenshot shows a window titled "PC1" with a tab bar containing "Physical", "Config", "Desktop", and "Programming". The "Desktop" tab is selected. Below the tabs is a "Command Prompt" window with a blue header bar. The command "ping 192.168.10.11" is entered and executed. The output shows four successful replies from the target IP address, followed by ping statistics indicating 0% loss and 0ms average round trip time. The command prompt ends with "c:\>".

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.10.11

Pinging 192.168.10.11 with 32 bytes of data:

Reply from 192.168.10.11: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.10.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

c:\>
```

- b. From the command prompt of **PC1**, ping **PC3**.

```
C:\>ping 192.168.11.10
Pinging 192.168.11.10 with 32 bytes of data:
Request timed out.
Reply from 192.168.11.10: bytes=32 time=1ms TTL=127
Reply from 192.168.11.10: bytes=32 time=4ms TTL=127
Reply from 192.168.11.10: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.11.10:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 4ms, Average = 1ms

C:\>
```

Why were the pings successful? Because Layers 1 through 3 are fully functional and there is no policy currently filtering ICMP messages between the two local networks.

Step 2: Ping devices on remote networks to test ACL functionality.

- From the command prompt of **PC1**, ping **PC4**.

```
C:\>ping 192.168.30.12
Pinging 192.168.30.12 with 32 bytes of data:
Reply from 192.168.10.1: Destination host unreachable.

Ping statistics for 192.168.30.12:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>
```

- From the command prompt of **PC1**, ping the **DNS Server**.

```
C:\>ping 192.168.31.12
Pinging 192.168.31.12 with 32 bytes of data:
Reply from 192.168.10.1: Destination host unreachable.

Ping statistics for 192.168.31.12:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>
```

Why did the pings fail? (**Hint:** Use simulation mode or view the router configurations to investigate.) The pings fail because R1 is configured with an ACL that denies any ping packets from exiting the Serial 0/0/0 interface.

```
R1>enable
R1#show acc
R1#show access-lists
Standard IP access list 11
  10 deny 192.168.10.0 0.0.0.255 (8 match(es) )
  20 permit any

R1#
```

Part 2: Remove the ACL and Repeat the Test

Step 1: Use show commands to investigate the ACL configuration.

- a. Navigate to R1 CLI. Use the **show run** and **show access-lists** commands to view the currently configured ACLs. To quickly view the current ACLs, use **show access-lists**. Enter the **show access-lists** command, followed by a space and a question mark (?) to view the available options:

```
R1# show access-lists ?
<1-199> ACL number WORD      ACL name
<cr>
```

If you know the ACL number or name, you can filter the **show** output further. However, **R1** only has one ACL; therefore, the **show access-lists** command will suffice.

```
R1#show access-lists
Standard IP access list 11
  10 deny 192.168.10.0 0.0.0.255
  20 permit any
```

The first line of the ACL blocks any packets that originate in the **192.168.10.0/24** network, which includes Internet Control Message Protocol (ICMP) echoes (ping requests). The second line of the ACL allows all other **ip** traffic from **any** source to transverse the router.

- b. For an ACL to impact router operation, it must be applied to an interface in a specific direction. In this scenario, the ACL is used to filter traffic exiting an interface. Therefore, all traffic leaving the specified interface of R1 will be inspected against ACL 11.

Although you can view IP information with the **show ip interface** command, it may be more efficient in some situations to simply use the **show run** command. To obtain a complete list of interfaces that the ACL that may be applied to, and the list of all ACLs that are configured, use the following command:

```
R1# show run | include interface|access
```

```
interface GigabitEthernet0/0 interface GigabitEthernet0/1 interface Serial0/0/0  
ip access-group 11 out interface Serial0/0/1 interface Vlan1  
access-list 11 deny 192.168.10.0 0.0.0.255 access-list 11 permit any
```

```
R1# show run | include interface|access  
interface GigabitEthernet0/0  
interface GigabitEthernet0/1  
interface Serial0/0/0  
  ip access-group 11 out  
interface Serial0/0/1  
interface Vlan1  
access-list 11 deny 192.168.10.0 0.0.0.255  
access-list 11 permit any  
R1#
```

The second pipe symbol ‘|’ creates an OR condition that matches ‘interface’ OR ‘access’. It is important that no spaces are included in the OR condition. Use one or both of these commands to find information about the ACL.

To which interface and in what direction is the ACL applied? Serial 0/0/0, outgoing traffic. Serial 0/0/0, outgoing traffic.

Step 2: Remove access list 11 from the configuration.

You can remove ACLs from the configuration by issuing the **no access list [number of the ACL]** command. The **no access-list** command when used without arguments deletes all ACLs configured on the router. The **no access-list [number of the ACL]** command removes only a specific ACL. Removing an ACL from a router does not remove the ACL from the interface. The command that applies the ACL to the interface must be removed separately.

- a. Under the Serial0/0/0 interface, remove access-list 11, which was previously applied to the interface as an **outgoing** filter:

```
R1(config)# interface s0/0/0
```

```
R1(config-if)# no ip access-group 11 out
```

b. In global configuration mode, remove the ACL by entering the following command:

```
R1(config)# no access-list 11
```

```
R1#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#inter
R1(config)#interface s0/0/0
R1(config-if)#no ip acc
R1(config-if)#no ip access-group 11 out
R1(config-if)#exit
R1(config)#no acc
R1(config)#no access-list 11
R1(config)#end
R1#
%SYS-5-CONFIG_I: Configured from console by console
```

c. Verify that **PC1** can now ping the **DNS Server** and **PC4**.

```
C:\>ping 192.168.30.12

Pinging 192.168.30.12 with 32 bytes of data:

Request timed out.
Reply from 192.168.30.12: bytes=32 time=16ms TTL=125
Reply from 192.168.30.12: bytes=32 time=2ms TTL=125
Reply from 192.168.30.12: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.30.12:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 16ms, Average = 6ms

C:\>ping 192.168.31.12

Pinging 192.168.31.12 with 32 bytes of data:

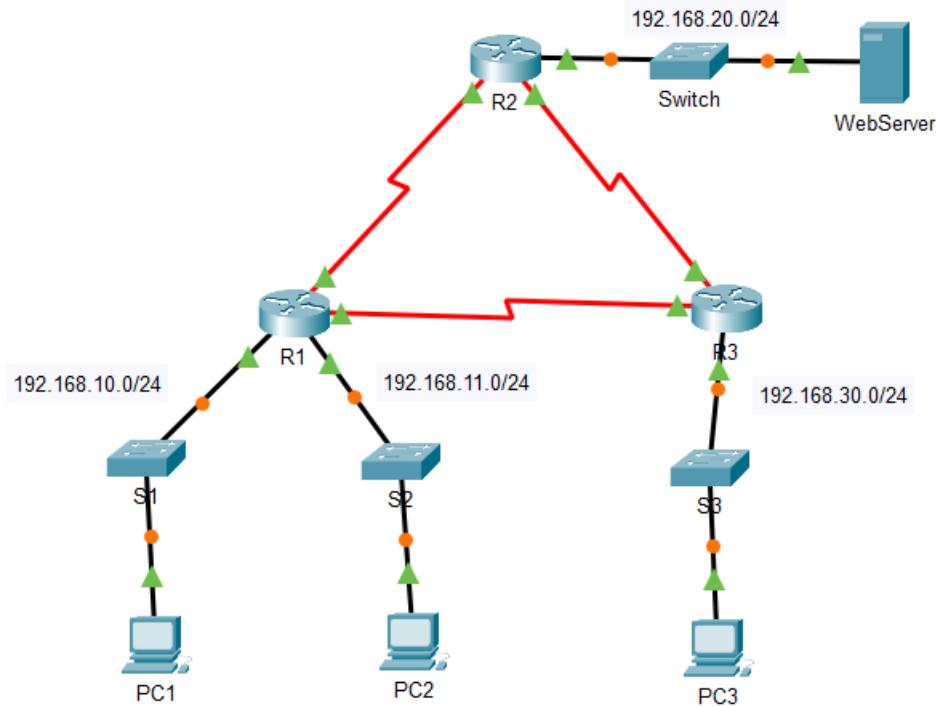
Request timed out.
Reply from 192.168.31.12: bytes=32 time=20ms TTL=125
Reply from 192.168.31.12: bytes=32 time=50ms TTL=125
Reply from 192.168.31.12: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.31.12:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 50ms, Average = 24ms

C:\>|
```

3.2.9 Exercise 5.1.8 - Packet Tracer - Configure Numbered Standard IPv4 ACLs

3.2.9.1 Topology



3.2.9.2 Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0	192.168.10.1	255.255.255.0	N/A
	G0/1	192.168.11.1	255.255.255.0	
	S0/0/0	10.1.1.1	255.255.255.252	
	S0/0/1	10.3.3.1	255.255.255.252	
R2	G0/0	192.168.20.1	255.255.255.0	N/A
	S0/0/0	10.1.1.2	255.255.255.252	
	S0/0/1	10.2.2.1	255.255.255.252	
R3	G0/0	192.168.30.1	255.255.255.0	N/A
	S0/0/0	10.3.3.2	255.255.255.252	
	S0/0/1	10.2.2.2	255.255.255.252	
PC1	NIC	192.168.10.10	255.255.255.0	192.168.10.1
PC2	NIC	192.168.11.10	255.255.255.0	192.168.11.1
PC3	NIC	192.168.30.10	255.255.255.0	192.168.30.1

WebServer	NIC	192.168.20.254	255.255.255.0	192.168.20.1
-----------	-----	----------------	---------------	--------------

3.2.9.3 Objectives

Part 1: Plan an ACL Implementation

Part 2: Configure, Apply, and Verify a Standard ACL

3.2.9.4 Background / Scenario

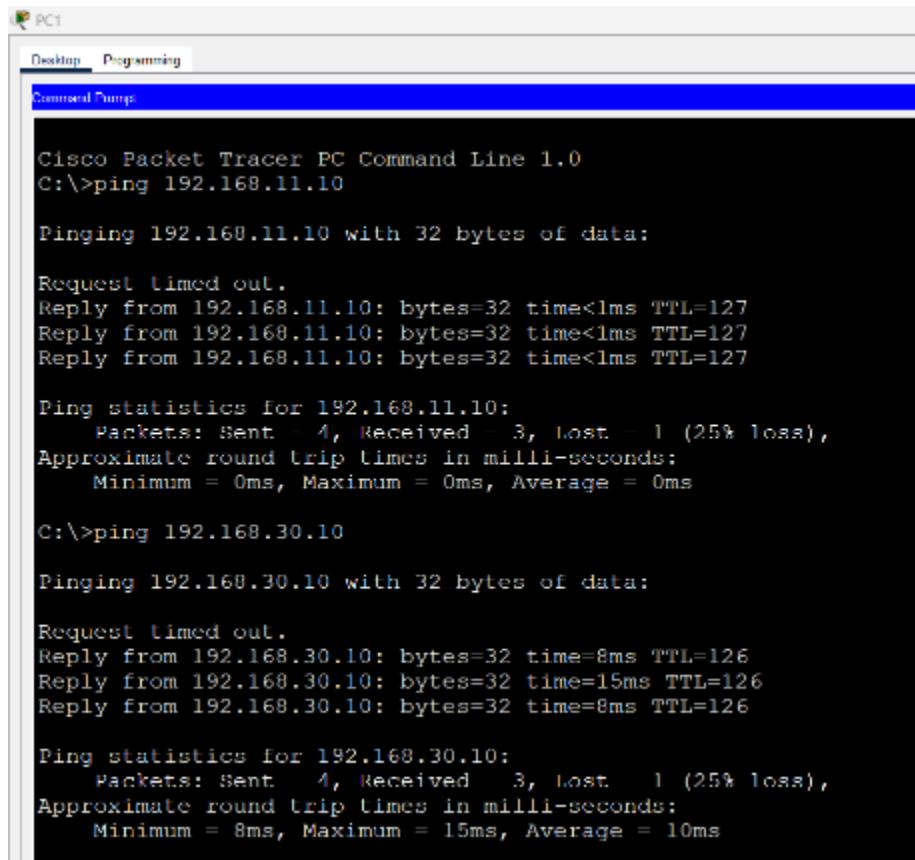
Standard access control lists (ACLs) are router configuration scripts that control whether a router permits or denies packets based on the source address. This activity focuses on defining filtering criteria, configuring standard ACLs, applying ACLs to router interfaces, and verifying and testing the ACL implementation. The routers are already configured, including IP addresses and Enhanced Interior Gateway Routing Protocol (EIGRP) routing.

3.2.9.5 Instructions

Part 1: Plan an ACL Implementation

Step 1: Investigate the current network configuration.

Before applying any ACLs to a network, it is important to confirm that you have full connectivity. Verify that the network has full connectivity by choosing a PC and pinging other devices on the network. You should be able to successfully ping every device.



```

PC1
Desktop Programming
Command Prompt

Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.11.10

Pinging 192.168.11.10 with 32 bytes of data:
Request timed out.
Reply from 192.168.11.10: bytes=32 time<1ms TTL=127
Reply from 192.168.11.10: bytes=32 time<1ms TTL=127
Reply from 192.168.11.10: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.11.10:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 192.168.30.10

Pinging 192.168.30.10 with 32 bytes of data:
Request timed out.
Reply from 192.168.30.10: bytes=32 time=8ms TTL=126
Reply from 192.168.30.10: bytes=32 time=15ms TTL=126
Reply from 192.168.30.10: bytes=32 time=8ms TTL=126

Ping statistics for 192.168.30.10:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 8ms, Maximum = 15ms, Average = 10ms
  
```

```
C:\>ping 192.168.20.254

Pinging 192.168.20.254 with 32 bytes of data:

Request timed out.
Reply from 192.168.20.254: bytes=32 time=9ms TTL=126
Reply from 192.168.20.254: bytes=32 time=9ms TTL=126
Reply from 192.168.20.254: bytes=32 time=1ms TTL=126

Ping statistics for 192.168.20.254:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 9ms, Average = 6ms

C:\>
```

Step 2: Evaluate two network policies and plan ACL implementations.

- a. The following network policies are implemented on **R2**:

- The 192.168.11.0/24 network is not allowed access to the **WebServer** on the 192.168.20.0/24 network.
- All other access is permitted.

To restrict access from the 192.168.11.0/24 network to the **WebServer** at 192.168.20.254 without interfering with other traffic, an ACL must be created on **R2**. The access list must be placed on the outbound interface to the **WebServer**. A second rule must be created on **R2** to permit all other traffic.

- b. The following network policies are implemented on **R3**:

- The 192.168.10.0/24 network is not allowed to communicate with the 192.168.30.0/24 network.
- All other access is permitted.

To restrict access from the 192.168.10.0/24 network to the 192.168.30.0/24 network without interfering with other traffic, an access list will need to be created on **R3**. The ACL must be placed on the outbound interface to **PC3**. A second rule must be created on **R3** to permit all other traffic.

Part 2: Configure, Apply, and Verify a Standard ACL

Step 1: Configure and apply a numbered standard ACL on R2.

- a. Create an ACL using the number **1** on **R2** with a statement that denies access to the 192.168.20.0/24 network from the 192.168.11.0/24 network.

```
R2(config)# access-list 1 deny 192.168.11.0 0.0.0.255
```

- b. By default, an access list denies all traffic that does not match any rules. To permit all other traffic, configure the following statement:

```
R2(config)# access-list 1 permit any
```

- c. Before applying an access list to an interface to filter traffic, it is a best practice to review the contents of the access list, in order to verify that it will filter traffic as expected.

```
R2# show access-lists
```

```
Standard IP access list 1
 10 deny 192.168.11.0 0.0.0.255
 20 permit any
```

- d. For the ACL to actually filter traffic, it must be applied to some router operation. Apply the ACL by placing it for outbound traffic on the GigabitEthernet 0/0 interface. Note: In an actual operational network, it is not a good practice to apply an untested access list to an active interface.

```
R2(config)# interface GigabitEthernet0/0
```

```
R2(config-if)# ip access-group 1 out
```

Step 2: Configure and apply a numbered standard ACL on R3.

- a. Create an ACL using the number **1** on **R3** with a statement that denies access to the 192.168.30.0/24 network from the **PC1** (192.168.10.0/24) network.

```
R3(config)# access-list 1 deny 192.168.10.0 0.0.0.255
```

- b. By default, an ACL denies all traffic that does not match any rules. To permit all other traffic, create a second rule for ACL 1.

```
R3(config)# access-list 1 permit any
```

- c. Verify that the access list is configured correctly.

```
R3# show access-lists
```

```
Standard IP access list 1
 10 deny 192.168.10.0 0.0.0.255
 20 permit any
```

- d. Apply the ACL by placing it for outbound traffic on the GigabitEthernet 0/0 interface.

```
R3(config)# interface GigabitEthernet0/0
```

```
R3(config-if)# ip access-group 1 out
```

Step 3: Verify ACL configuration and functionality.

- a. Enter the **show run** or **show ip interface gigabitethernet0/0** command to verify the ACL placements.

```
R2#show ip interface g0/0
GigabitEthernet0/0 is up, line protocol is up (connected)
  Internet address is 192.168.20.1/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
Outgoing access list is 1
  Inbound access list is not set
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachables are always sent
  ICMP mask replies are never sent
  IP fast switching is disabled
  IP fast switching on the same interface is disabled
  IP Flow switching is disabled
  IP Fast switching turbo vector
  IP multicast fast switching is disabled
  IP multicast distributed fast switching is disabled
  Router Discovery is disabled
  IP output packet accounting is disabled
  IP access violation accounting is disabled
  TCP/IP header compression is disabled
  RTP/IP header compression is disabled
  Probe proxy name replies are disabled
  Policy routing is disabled
  Network address translation is disabled
  BGP Policy Mapping is disabled
  Input features: MCI Check
  WCCP Redirect outbound is disabled
  WCCP Redirect inbound is disabled
  WCCP Redirect exclude is disabled
```

- b. With the two ACLs in place, network traffic is restricted according to the policies detailed in Part 1. Use the following tests to verify the ACL implementations:

- A ping from 192.168.10.10 to 192.168.11.10 succeeds.

PC1

Desktop Programming

Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ipconfig

FastEthernet0 Connection:(default port)

Connection-specific DNS Suffix...:
Link-local IPv6 Address.....: FE80::201:96FF:FE06:A5AB
IPv6 Address.....: ::
IPv4 Address.....: 192.168.10.10
Subnet Mask.....: 255.255.255.0
Default Gateway.....: ::
                           192.168.10.1

Bluetooth Connection:

Connection-specific DNS Suffix...:
Link-local IPv6 Address.....: ::
IPv6 Address.....: ::
IPv4 Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: ::
                           0.0.0.0

C:\>ping 192.168.11.10

Pinging 192.168.11.10 with 32 bytes of data:

Request timed out.
Reply from 192.168.11.10: bytes=32 time<1ms TTL=127
Reply from 192.168.11.10: bytes=32 time<1ms TTL=127
Reply from 192.168.11.10: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.11.10:
   Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
   Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

- A ping from 192.168.10.10 to 192.168.20.254 succeeds.

```
C:\>ping 192.168.20.254

Pinging 192.168.20.254 with 32 bytes of data:

Request timed out.
Reply from 192.168.20.254: bytes=32 time=1ms TTL=126
Reply from 192.168.20.254: bytes=32 time=1ms TTL=126
Reply from 192.168.20.254: bytes=32 time=12ms TTL=126

Ping statistics for 192.168.20.254:
   Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
   Minimum = 1ms, Maximum = 12ms, Average = 4ms
```

- A ping from 192.168.11.10 to 192.168.20.254 fails.

```
PC2
Desktop Programming
Command Prompt

Cisco Packet Tracer PC Command Line 1.0
C:>ipconfig

FastEthernet0 Connection:(default port)

Connection-specific DNS Suffix...:
Link-local IPv6 Address.....: FE80::20B:BEFF:FE5:8C47
IPv6 Address.....: ::
IPv4 Address.....: 192.168.11.10
Subnet Mask.....: 255.255.255.0
Default Gateway.....: ::
                           192.168.11.1

Bluetooth Connection:

Connection-specific DNS Suffix...:
Link-local IPv6 Address.....: ::
IPv6 Address.....: ::
IPv4 Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: ::
                           0.0.0.0

C:>ping 192.168.20.254
Ping request could not find host 192.168.20.254. Please check
C:>ping 192.168.20.254

Pinging 192.168.20.254 with 32 bytes of data:

Reply from 10.1.1.2: Destination host unreachable.

Ping statistics for 192.168.20.254:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- A ping from 192.168.10.10 to 192.168.30.10 fails.

```
C:>ping 192.168.30.10

Pinging 192.168.30.10 with 32 bytes of data:

Reply from 10.3.3.2: Destination host unreachable.

Ping statistics for 192.168.30.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- A ping from 192.168.11.10 to 192.168.30.10 succeeds.

```
^C
C:>ping 192.168.30.10

Pinging 192.168.30.10 with 32 bytes of data:

Request timed out.
Reply from 192.168.30.10: bytes=32 time=14ms TTL=126
Reply from 192.168.30.10: bytes=32 time=14ms TTL=126
Reply from 192.168.30.10: bytes=32 time=9ms TTL=126

Ping statistics for 192.168.30.10:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 9ms, Maximum = 14ms, Average = 12ms

C:>
```

- A ping from 192.168.30.10 to 192.168.20.254 succeeds.

```

PC1
Desktop Programming
Command Prompt

FastEthernet0 Connection: (default port)

Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: FE80::209:7CFF:FE4C:972B
IPv6 Address.....: ::
IPv4 Address.....: 192.168.30.10
Subnet Mask.....: 255.255.255.0
Default Gateway.....: ::
                           192.168.30.1

Bluetooth Connection:

Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: ::
IPv6 Address.....: ::
IPv4 Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: ::
                           0.0.0.0

C:\>\ping 192.168.20.254
Invalid Command.

C:\>ping 192.168.20.254
Pinging 192.168.20.254 with 32 bytes of data:

Reply from 192.168.20.254: bytes=32 time=10ms TTL=126
Reply from 192.168.20.254: bytes=32 time=11ms TTL=126
Reply from 192.168.20.254: bytes=32 time=1ms TTL=126
Reply from 192.168.20.254: bytes=32 time=11ms TTL=126

Ping statistics for 192.168.20.254:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli seconds:
    Minimum = 1ms, Maximum = 11ms, Average = 8ms

```

- c. Issue the **show access-lists** command again on routers R2 and R3. You should see output that indicates the number of packets that have matched each line of the access list. Note: The number of matches shown for your routers may be different, due to the number of pings that are sent and received.

R2# show access-lists

```

Standard IP access list 1
  10 deny 192.168.11.0 0.0.0.255 (4 match(es))
  20 permit any (8 match(es))

```

R3# show access-lists

```

Standard IP access list 1
  10 deny 192.168.10.0 0.0.0.255 (4 match(es))
  20 permit any (8 match(es))

```

3.2.9.6 Scripts

!! R2

```

enable
configure terminal
interface GigabitEthernet0/0
  ip access-group 1 out
access-list 1 deny 192.168.11.0 0.0.0.255
access-list 1 permit any
end

```

!!! R3

```
enable
configure terminal
interface GigabitEthernet0/0
  ip access-group 1 out
access-list 1 deny 192.168.10.0 0.0.0.255
access-list 1 permit any
end
```

3.2.9.7 Printouts

R2

```
R2#show ip interface brief
Interface          IP-Address      OK? Method Status      Protocol
GigabitEthernet0/0 192.168.20.1   YES manual up       up
GigabitEthernet0/1 unassigned     YES unset administratively down down
Serial0/0/0         10.1.1.2      YES manual up       up
Serial0/0/1         10.2.2.1      YES manual up       up
Vlan1              unassigned     YES unset administratively down down
R2#
```

```
R2#! R2
R2#enable
R2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#interface GigabitEthernet0/0
R2(config-if)# ip access-group 1 out
R2(config-if)#access-list 1 deny 192.168.11.0 0.0.0.255
R2(config)#access-list 1 permit any
R2(config)#end
R2#
%SYS-5-CONFIG_I: Configured from console by console
```

```
R2#show acce
R2#show access-lists
Standard IP access list 1
  10 deny 192.168.11.0 0.0.0.255
  20 permit any
```

```

R2#show running-config | section interface
interface GigabitEthernet0/0
 ip address 192.168.20.1 255.255.255.0
 ip access-group 1 out
 duplex auto
 speed auto
interface GigabitEthernet0/1
 no ip address
 duplex auto
 speed auto
 shutdown
interface Serial0/0/0
 description Link to R1
 ip address 10.1.1.2 255.255.255.252
interface Serial0/0/1
 description Link to R3
 ip address 10.2.2.1 255.255.255.252
 clock rate 4000000
interface Vlan1
 no ip address
 shutdown
 passive-interface GigabitEthernet0/0
R2#
R2#show running-config | section zcess
R2#show running-config | section access
 ip access-group 1 out
access-list 1 deny 192.168.11.0 0.0.0.255
access-list 1 permit any
R2#

```

```

R2#show ip interface g0/0
GigabitEthernet0/0 is up, line protocol is up (connected)
 Internet address is 192.168.20.1/24
 Broadcast address is 255.255.255.255
 Address determined by setup command
 MTU is 1500 bytes
 Helper address is not set
 Directed broadcast forwarding is disabled
Outgoing access list is 1
 Inbound access list is not set
 Proxy ARP is enabled
 Security level is default
 Split horizon is enabled
 ICMP redirects are always sent
 ICMP unreachables are always sent
 ICMP mask replies are never sent
 IP fast switching is disabled
 IP fast switching on the same interface is disabled
 IP Flow switching is disabled
 IP Fast switching turbo vector
 IP multicast fast switching is disabled
 IP multicast distributed fast switching is disabled
 Router Discovery is disabled
 IP output packet accounting is disabled
 IP access violation accounting is disabled
 TCP/IP header compression is disabled
 RTP/IP header compression is disabled
 Probe proxy name replies are disabled
 Policy routing is disabled
 Network address translation is disabled
 BGP Policy Mapping is disabled
 Input features: MCI Check
 WCCP Redirect outbound is disabled
 WCCP Redirect inbound is disabled
 WCCP Redirect exclude is disabled

```

R3

```
R3>
R3>show ip interface brief
Interface          IP-Address      OK? Method Status       Protocol
GigabitEthernet0/0  192.168.30.1   YES manual up        up
GigabitEthernet0/1  unassigned     YES unset administratively down down
Serial0/0/0         10.3.3.2      YES manual up        up
Serial0/0/1         10.2.2.2      YES manual up        up
Vlan1              unassigned     YES unset administratively down down
R3>
R3>
R3>
```

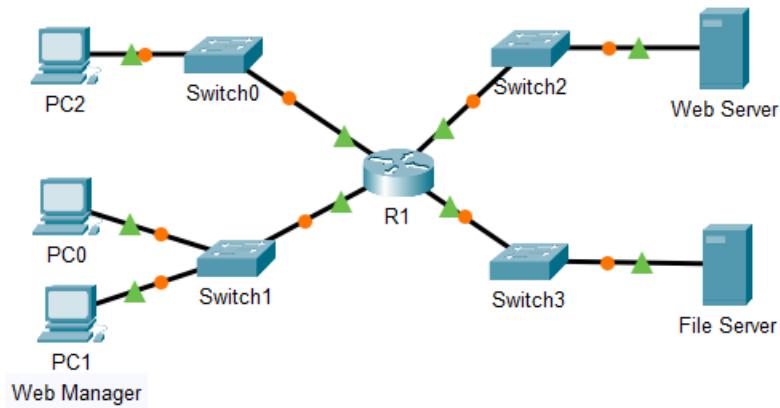
```
R3>!!! R3
R3>enable
R3#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#interface GigabitEthernet0/0
R3(config-if)# ip access-group 1 out
R3(config-if)#access-list 1 deny 192.168.10.0 0.0.0.255
R3(config)#access-list 1 permit any
R3(config)#end
R3#
%SYS-5-CONFIG_I: Configured from console by console
```

```
R3#show access-lists
Standard IP access list 1
  10 deny 192.168.10.0 0.0.0.255
  20 permit any
```

```
R3#show running-config | section interface
interface GigabitEthernet0/0
  description R3 LAN
  ip address 192.168.30.1 255.255.255.0
  ip access-group 1 out
  duplex auto
  speed auto
interface GigabitEthernet0/1
  no ip address
  duplex auto
  speed auto
  shutdown
interface Serial0/0/0
  ip address 10.3.3.2 255.255.255.252
interface Serial0/0/1
  description Link to R2
  ip address 10.2.2.2 255.255.255.252
interface Vlan1
  no ip address
  shutdown
  passive-interface GigabitEthernet0/0
R3#
R3#show running-config | section access
  ip access-group 1 out
  access-list 1 deny 192.168.10.0 0.0.0.255
  access-list 1 permit any
R3#
```

3.2.10 Exercise 5.1.9 - Packet Tracer - Configure Named Standard IPv4 ACLs

3.2.10.1 Topology



3.2.10.2 Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	F0/0	192.168.100.1	255.255.255.0	N/A
	F0/1	192.168.200.1	255.255.255.0	
	E0/0/0	192.168.10.1	255.255.255.0	
	E0/1/0	192.168.20.1	255.255.255.0	
File Server	NIC	192.168.200.100	255.255.255.0	192.168.200.1
Web Server	NIC	192.168.100.100	255.255.255.0	192.168.100.1
PC0	NIC	192.168.20.3	255.255.255.0	192.168.20.1
PC1	NIC	192.168.20.4	255.255.255.0	192.168.20.1
PC2	NIC	192.168.10.3	255.255.255.0	192.168.10.1

3.2.10.3 Objectives

Part 1: Configure and Apply a Named Standard ACL Part 2:

Verify the ACL Implementation

3.2.10.4 Background / Scenario

The senior network administrator has asked you to create a standard named ACL to prevent access to a file server. The file server contains the data base for the web applications. Only the Web Manager workstation PC1 and the Web Server need to access the File Server. All other traffic to the File Server should be denied.

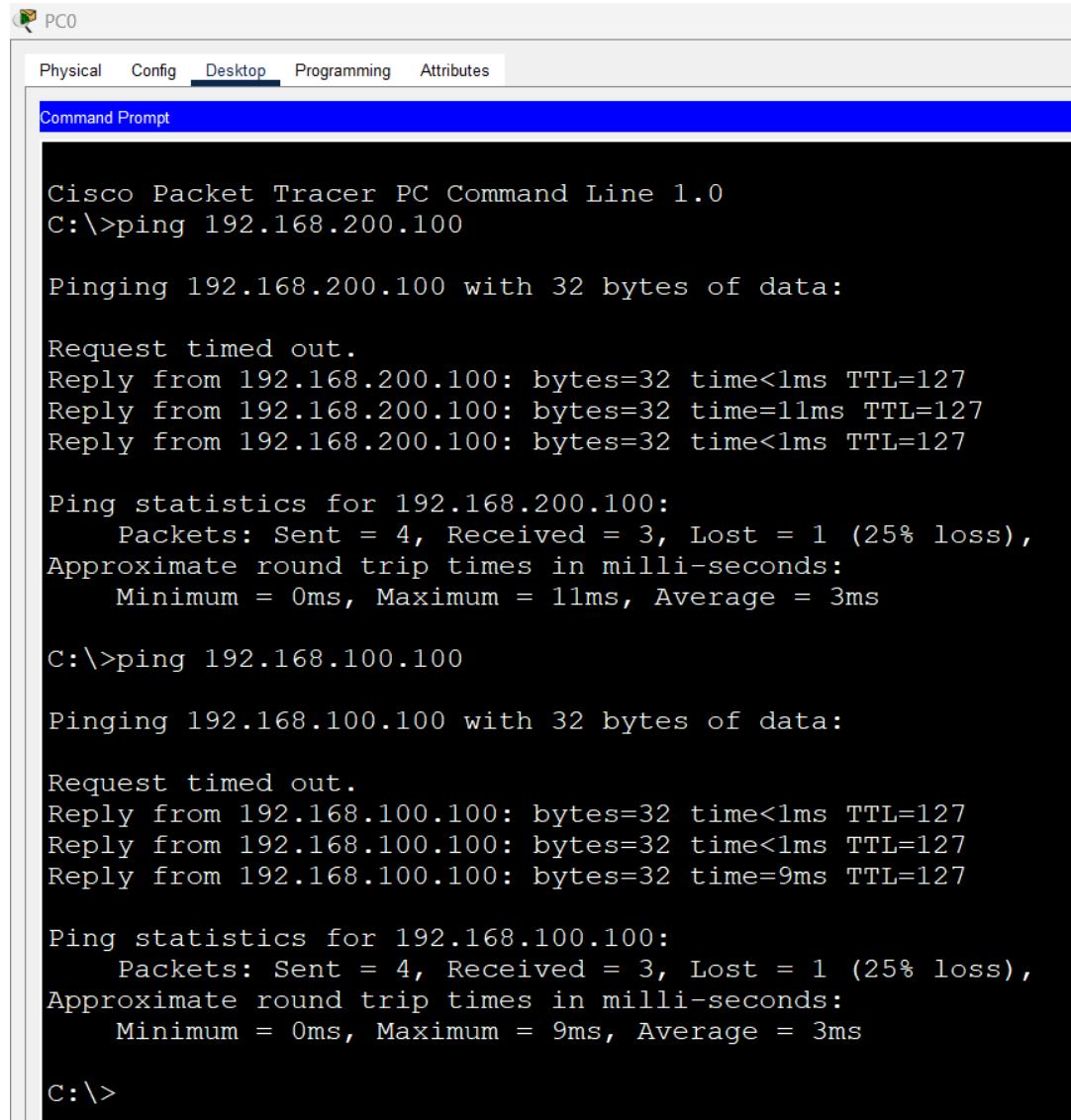
3.2.10.5 Instructions

Part 1: Configure and Apply a Named Standard ACL

Step 1: Verify connectivity before the ACL is configured and applied.

All three workstations should be able to ping both the **Web Server** and **File Server**.

PC0



The screenshot shows the Cisco Packet Tracer PC Command Line interface for PC0. The window title is "PC0". The menu bar includes "Physical", "Config", "Desktop", "Programming", and "Attributes", with "Desktop" being the active tab. A toolbar icon for "Command Prompt" is visible. The main area is a terminal window titled "Command Prompt". The terminal output shows two ping sessions:

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.200.100

Pinging 192.168.200.100 with 32 bytes of data:

Request timed out.
Reply from 192.168.200.100: bytes=32 time<1ms TTL=127
Reply from 192.168.200.100: bytes=32 time=11ms TTL=127
Reply from 192.168.200.100: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.200.100:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 11ms, Average = 3ms

C:\>ping 192.168.100.100

Pinging 192.168.100.100 with 32 bytes of data:

Request timed out.
Reply from 192.168.100.100: bytes=32 time<1ms TTL=127
Reply from 192.168.100.100: bytes=32 time<1ms TTL=127
Reply from 192.168.100.100: bytes=32 time=9ms TTL=127

Ping statistics for 192.168.100.100:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 9ms, Average = 3ms

C:\>
```

PC1

PC1

Physical Config Desktop Programming Attributes

Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.100.100

Pinging 192.168.100.100 with 32 bytes of data:

Reply from 192.168.100.100: bytes=32 time=3ms TTL=127
Reply from 192.168.100.100: bytes=32 time=1ms TTL=127
Reply from 192.168.100.100: bytes=32 time=1ms TTL=127
Reply from 192.168.100.100: bytes=32 time=6ms TTL=127

Ping statistics for 192.168.100.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 6ms, Average = 2ms

C:\>ping 192.168.200.100

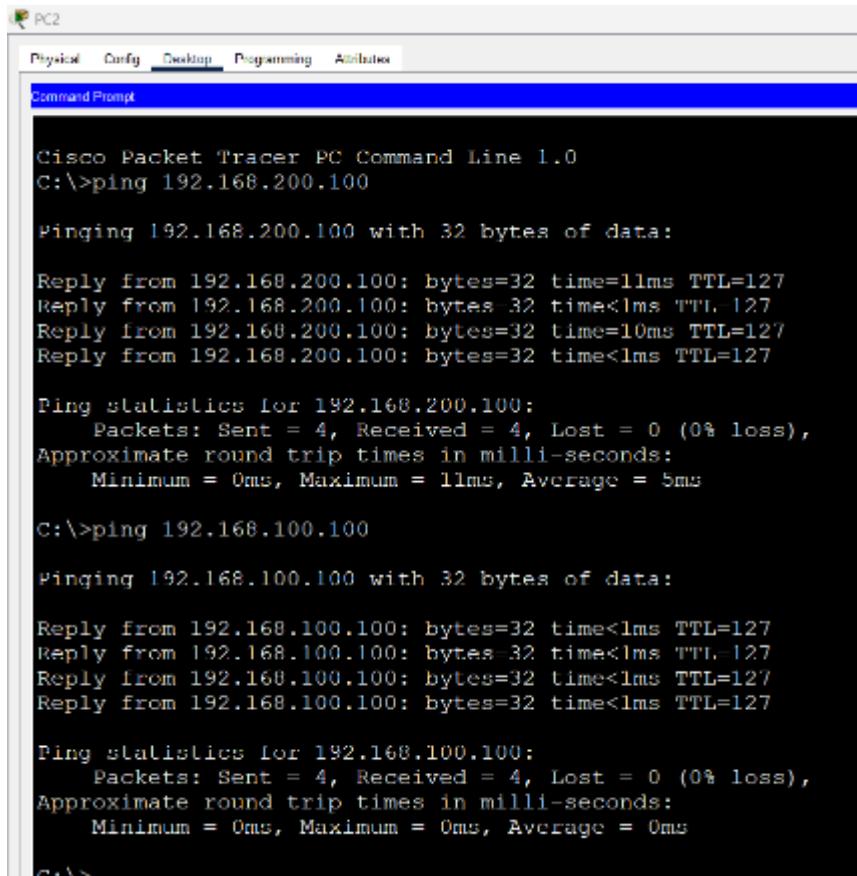
Pinging 192.168.200.100 with 32 bytes of data:

Reply from 192.168.200.100: bytes=32 time<1ms TTL=127
Reply from 192.168.200.100: bytes=32 time=1ms TTL=127
Reply from 192.168.200.100: bytes=32 time=1ms TTL=127
Reply from 192.168.200.100: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.200.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>
```

PC2



The screenshot shows a Cisco Packet Tracer window titled "PC2". The menu bar includes "Physical", "Config", "Desktop", "Programming", and "Attributes". A tab labeled "Command Prompt" is selected. The main area displays the output of a ping command from a Windows command prompt. The first ping to 192.168.200.100 shows variable round trip times (11ms, <1ms, 10ms, <1ms) and a 0% loss rate. The second ping to 192.168.100.100 shows a 0ms round trip time and 0% loss.

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.200.100

Pinging 192.168.200.100 with 32 bytes of data:

Reply from 192.168.200.100: bytes=32 time=11ms TTL=127
Reply from 192.168.200.100: bytes=32 time=<1ms TTL=127
Reply from 192.168.200.100: bytes=32 time=10ms TTL=127
Reply from 192.168.200.100: bytes=32 time=<1ms TTL=127

Ping statistics for 192.168.200.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 11ms, Average = 5ms

C:\>ping 192.168.100.100

Pinging 192.168.100.100 with 32 bytes of data:

Reply from 192.168.100.100: bytes=32 time=<1ms TTL=127

Ping statistics for 192.168.100.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Step 2: Configure a named standard ACL.

- Configure the following named ACL on R1.

```
R1(config)# ip access-list standard File_Server_Restrictions
R1(config-std-nacl)# permit host 192.168.20.4
R1(config-std-nacl)# permit host 192.168.100.100
R1(config-std-nacl)# deny any
```

```
enable
configure terminal
ip access-list standard File_Server_Restrictions
    permit host 192.168.20.4
    permit host 192.168.100.100
    deny any
exit
end
write memory
```

```

-----  

R1#enable  

R1#configure terminal  

Enter configuration commands, one per line. End with CNTL/Z.  

R1(config)#ip access-list standard File_Server_Restrictions  

R1(config-std-nacl)#permit host 192.168.20.4  

R1(config-std-nacl)#permit host 192.168.100.100  

R1(config-std-nacl)#deny any  

R1(config-std-nacl)#exit  

R1(config)#end  

R1#write memory  

%SYS-5-CONFIG_I: Configured from console by console  

Building configuration...  

[OK]  

R1#

```

Note: For scoring purposes, the ACL name is case-sensitive, and the statements must be in the same order as shown.

- b. Use the **show access-lists** command to verify the contents of the access list before applying it to an interface. Make sure you have not mistyped any IP addresses and that the statements are in the correct order.

```

R1# show access-lists  

Standard IP access list File_Server_Restrictions  

  10 permit host 192.168.20.4  

  20 permit host 192.168.100.100  

  30 deny any

```

```

Building configuration...  

[OK]  

R1#show access-lists  

Standard IP access list File_Server_Restrictions  

  10 permit host 192.168.20.4  

  20 permit host 192.168.100.100  

  30 deny any

```

R1#

Step 3: Apply the named ACL.

- a. Apply the ACL outbound on the Fast Ethernet 0/1 interface.

Note: In an actual operational network, applying an access list to an active interface is not a good practice and should be avoided if possible.

```
R1(config-if)# ip access-group File_Server_Restrictions out
```

- b. Save the configuration.

```

enable
configure terminal
interface FastEthernet0/1
  ip access-group File_Server_Restrictions out
exit
end
write memory

```

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#interface FastEthernet0/1
R1(config-if)# ip access-group File_Server_Restrictions out
R1(config-if)#exit
R1(config)#
R1#write memory
Building configuration...
[OK]
R1#
SYS-5-CONFIG_I: Configured from console by console
R1#
```

Part 2: Verify the ACL Implementation

Step 1: Verify the ACL configuration and application to the interface.

Use the **show access-lists** command to verify the ACL configuration. Use the **show run** or **show ip interface fastethernet 0/1** command to verify that the ACL is applied correctly to the interface.

Step 2: Verify that the ACL is working properly.

All three workstations should be able to ping the **Web Server**, but only **PC1** and the **Web Server** should be able to ping the **File Server**. Repeat the **show access-lists** command to see the number of packets that matched each statement.

```
--"
R1#show access-lists
Standard IP access list File_Server_Restrictions
    10 permit host 192.168.20.4
    20 permit host 192.168.100.100
    30 deny any

R1#
R1#show run
R1#show running-config | section interface
interface FastEthernet0/0
    ip address 192.168.100.1 255.255.255.0
    duplex auto
    speed auto
interface FastEthernet0/1
    ip address 192.168.200.1 255.255.255.0
    ip access-group File_Server_Restrictions out
    duplex auto
    speed auto
interface Ethernet0/0/0
    ip address 192.168.10.1 255.255.255.0
    duplex auto
    speed auto
interface Ethernet0/1/0
    ip address 192.168.20.1 255.255.255.0
    duplex auto
    speed auto
interface Vlan1
    no ip address
    shutdown
```

```
R1#show ip interface FastEthernet 0/1
FastEthernet0/1 is up, line protocol is up (connected)
  Internet address is 192.168.200.1/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
Outgoing access list is File_Server_Restrictions
  Inbound access list is not set
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachable messages are always sent
  ICMP mask replies are never sent
  IP fast switching is disabled
  IP fast switching on the same interface is disabled
  IP Flow switching is disabled
  IP Fast switching turbo vector
  IP multicast fast switching is disabled
  IP multicast distributed fast switching is disabled
  Router Discovery is disabled
  IP output packet accounting is disabled
  IP access violation accounting is disabled
  TCP/IP header compression is disabled
  RTP/IP header compression is disabled
  Probe proxy name replies are disabled
  Policy routing is disabled
  Network address translation is disabled
  BGP Policy Mapping is disabled
  Input features: MCI Check
  WCCP Redirect outbound is disabled
  WCCP Redirect inbound is disabled
  WCCP Redirect exclude is disabled
```

```
R1#
```

```
C:\>
C:\>ping 192.168.200.100
```

```
Pinging 192.168.200.100 with 32 bytes of data:
```

```
Reply from 192.168.200.100: bytes=32 time=1ms TTL=127
Reply from 192.168.200.100: bytes=32 time<1ms TTL=127
Reply from 192.168.200.100: bytes=32 time<1ms TTL=127
Reply from 192.168.200.100: bytes=32 time<1ms TTL=127
```

```
Ping statistics for 192.168.200.100:
```

```
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

```
C:\>
```

```

R1#
R1#show access-lists
Standard IP access list File_Server_Restrictions
  10 permit host 192.168.20.4 (4 match(es))
  20 permit host 192.168.100.100
  30 deny any

```

R1#

3.2.10.6 Script

```

!!! R1
enable
configure terminal
ip access-list standard File_Server_Restrictions
permit host 192.168.20.4
permit host 192.168.100.100
deny any
exit
interface FastEthernet0/1
  ip access-group File_Server_Restrictions out
exit
end
write memory

```

3.2.10.7 Assessment

Activity Results

Congratulations Guest! You completed the activity.

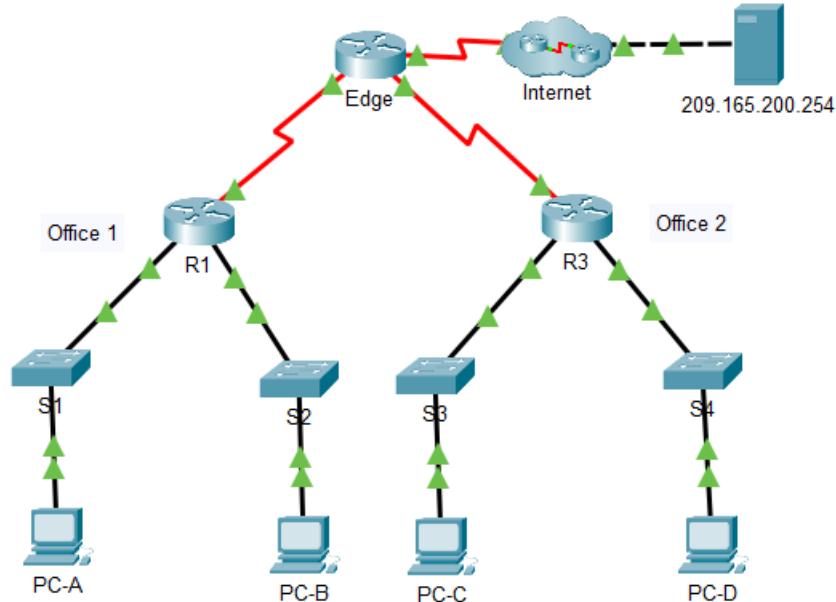
Overall Feedback Assessment Items Connectivity Tests

[Expand/Collapse All](#) [Show Incorrect Items](#)

Assessment Items	/	Status	Points	Component(s)	Feedback
Network	/				
R1	/				
ACL	/	✓	Correct	0	ACL IPv4 Standard AC...
Ports	/			0	Other Other
FastEthernet0/1	/	✓	Access-group Out Correct	20	IPv4 Standard AC...

3.2.11 Exercise 5.2.7 - Packet Tracer - Configure and Modify Standard IPv4 ACLs

3.2.11.1 Topology



3.2.11.2 Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0/0	192.168.10.1	255.255.255.0	N/A
	G0/0/1	192.168.20.1	255.255.255.0	
	S0/1/0 (DCE)	10.1.1.1	255.255.255.252	
Edge	S0/1/0	10.1.1.2	255.255.255.252	N/A
	S0/1/1 (DCE)	10.2.2.2	255.255.255.252	
	S0/2/1	209.165.200.225	255.255.255.224	
R3	G0/0/0	192.168.30.1	255.255.255.0	N/A
	G0/0/1	192.168.40.1	255.255.255.0	
	S0/1/1	10.2.2.1	255.255.255.252	
S1	VLAN 1	192.168.10.11	255.255.255.0	192.168.10.1
S2	VLAN 1	192.168.20.11	255.255.255.0	192.168.20.1
S3	VLAN 1	192.168.30.11	255.255.255.0	192.168.30.1
S4	VLAN 1	192.168.40.11	255.255.255.0	192.168.40.1
PC-A	NIC	192.168.10.3	255.255.255.0	192.168.10.1

PC-B	NIC	192.168.20.3	255.255.255.0	192.168.20.1
PC-C	NIC	192.168.30.3	255.255.255.0	192.168.30.1
PC-D	NIC	192.168.40.3	255.255.255.0	192.168.40.1

3.2.11.3 Objectives

Part 1: Verify Connectivity

Part 2: Configure and Verify Standard Numbered and Named ACLs

Part 3: Modify a Standard ACL

3.2.11.4 Background / Scenario

Network security and traffic flow control are important issues when designing and managing IP networks. The ability to configure proper rules to filter packets, based on established security policies, is a valuable skill.

In this lab, you will set up filtering rules for two business locations that are represented by R1 and R3.

Management has established some access policies between the LANs located at R1 and R3, which you must implement. The Edge router sitting between R1 and R3 has been provided by the ISP will not have any ACLs placed on it. You would not be allowed any administrative access to the Edge router because you can only control and manage your own equipment.

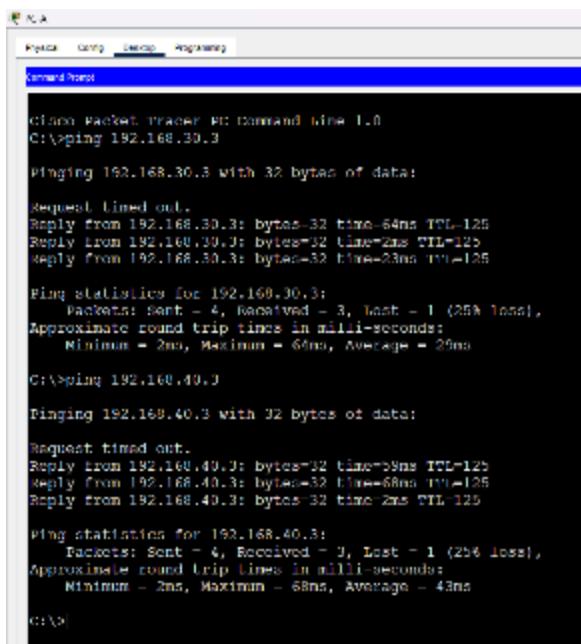
3.2.11.5 Instructions

Part 1: Verify Connectivity

In Part 1, you verify connectivity between devices.

Note: It is very important to test whether connectivity is working **before** you configure and apply access lists. You want to be sure that your network is properly functioning before you start to filter traffic.

From PC-A, ping PC-C and PC-D. Were your pings successful? Yes



```

C:\>ping 192.168.30.3

Pinging 192.168.30.3 with 32 bytes of data:
Request timed out.
Reply from 192.168.30.3: bytes=32 time=64ms TTL=125
Reply from 192.168.30.3: bytes=32 time=2ms TTL=125
Reply from 192.168.30.3: bytes=32 time=23ms TTL=125

Ping statistics for 192.168.30.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 64ms, Average = 29ms

C:\>ping 192.168.40.3

Pinging 192.168.40.3 with 32 bytes of data:
Request timed out.
Reply from 192.168.40.3: bytes=32 time=28ms TTL=125
Reply from 192.168.40.3: bytes=32 time=68ms TTL=125
Reply from 192.168.40.3: bytes=32 time=2ms TTL=125

Ping statistics for 192.168.40.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 68ms, Average = 43ms

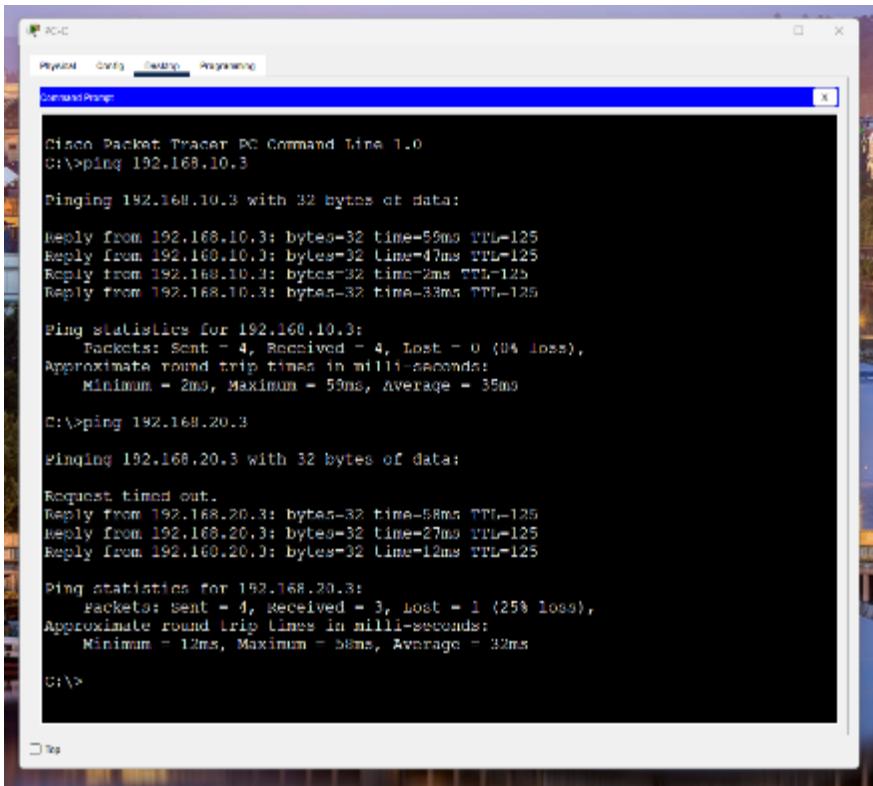
C:\>

```

From R1, ping PC-C and PC-D. Were your pings successful? Yes

```
R1>enable  
R1#ping 192.168.30.3  
  
Type escape sequence to abort.  
Sending 5, 100-byte ICMP Echos to 192.168.30.3, timeout  
is 2 seconds:  
!!!!!  
Success rate is 100 percent (5/5), round-trip min/avg/  
max = 31/47/61 ms  
  
R1#ping 192.168.40.3  
  
Type escape sequence to abort.  
Sending 5, 100-byte ICMP Echos to 192.168.40.3, timeout  
is 2 seconds:  
!!!!!  
Success rate is 100 percent (5/5), round-trip min/avg/  
max = 37/47/64 ms  
  
R1#
```

From PC-C, ping PC-A and PC-B. Were your pings successful? Yes



From R3, ping PC-A and PC-B. Were your pings successful? Yes

```
R3>enable
R3#ping 192.168.10.3

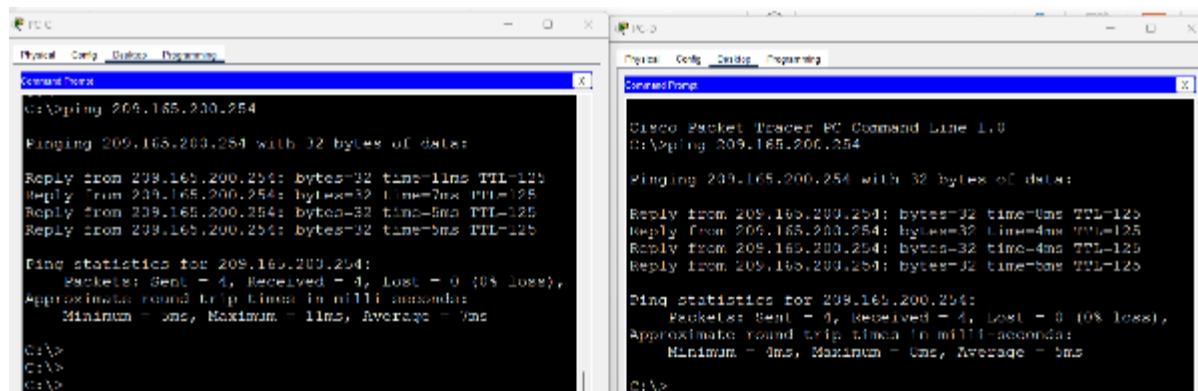
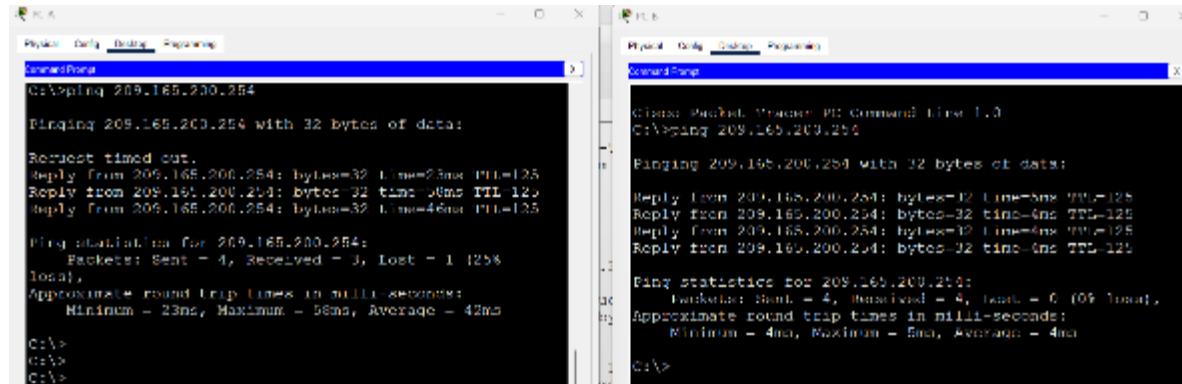
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 22/37/76 ms
```

R3#ping 192.168.20.3

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.20.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/35/58 ms
```

R3#

Can all of the PCs ping the server at 209.165.200.254? Yes



Part 2: Configure and Verify Standard Numbered and Named ACLs

Step 1: Configure a numbered standard ACL.

Standard ACLs filter traffic based on the source IP address only. A typical best practice for standard ACLs is to configure and apply the ACL as close to the destination as possible. For the first access list in this activity, create a standard numbered ACL that allows traffic from all hosts on the 192.168.10.0/24 network and all hosts on the 192.168.20.0/24 network to access all hosts on the 192.168.30.0/24 network. The security policy also states that an explicit **deny any** access control entry (ACE), also referred to as an ACL statement, should be present at the end of all ACLs.

What wildcard mask would you use to allow all hosts on the 192.168.10.0/24 network to access the 192.168.30.0/24 network? **Answer 0.0.0.255**

Following Cisco's recommended best practices, on which router would you place this ACL?

Answer - R3 - A typical best practice for standard ACLs is to configure and apply the ACL as close to the destination as possible.

On which interface would you place this ACL? In what direction would you apply it?

Answer - G0/0/0. The ACL should be applied going out.

If we place the ACL on the S0/1/1 interface on R3 going in. This would effectively block the LANs on R1 from getting to the 192.168.40.0/24 network as well!

- a. Configure the ACL on R3. Use 1 for the access list number.

```
R3(config)# access-list 1 remark Allow R1 LANs Access
R3(config)# access-list 1 permit 192.168.10.0 0.0.0.255
R3(config)# access-list 1 permit 192.168.20.0 0.0.0.255
R3(config)# access-list 1 deny any
```

- b. Apply the ACL to the appropriate interface in the proper direction.

```
R3(config)# interface g0/0/0
R3(config-if)# ip access-group 1 out
```

```
R3>enable
R3#
R3#config t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#access-list 1 remark Allow R1 LANs Access
R3(config)#access-list 1 permit 192.168.10.0 0.0.0.255
R3(config)#access-list 1 permit 192.168.20.0 0.0.0.255
R3(config)#access-list 1 deny any
R3(config)#
R3(config)#interface g0/0/0
R3(config-if)#ip access-group 1 out
R3(config-if)#end
R3#
%SYS-5-CONFIG_I: Configured from console by console
R3#copy me
R3#copy ru
R3#copy running-config
R3#copy running-config st
R3#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R3#
```

- c. Verify a numbered ACL.

The use of various **show** commands can help you to verify both the syntax and placement of your ACLs in your router.

To see access list 1 in its entirety with all ACEs, which command would you use? **Answer** R3# show access-lists

```
R3#show access-lists
Standard IP access list 1
    10 permit 192.168.10.0 0.0.0.255
    20 permit 192.168.20.0 0.0.0.255
    30 deny any
```

What command would you use to see where the access list was applied and in what direction? **Answer** - show ip interface g0/0/0

- 1) On R3, issue the **show access-lists 1** command.

```
R3# show access-list 1
```

```
Standard IP access list 1
    permit 192.168.10.0, wildcard bits 0.0.0.255
    permit 192.168.20.0, wildcard bits 0.0.0.255
    deny any
```

```
R3#show access-list 1
Standard IP access list 1
    permit 192.168.10.0 0.0.0.255
    permit 192.168.20.0 0.0.0.255
    deny any

R3#
```

- 2) On R3, issue the **show ip interface g0/0/0** command.

```
R3# show ip interface g0/0/0
GigabitEthernet0/0/0 is up, line protocol is up (connected)
    Internet address is 192.168.30.1/24
    Broadcast address is 255.255.255.255
    Address determined by setup command
    MTU is 1500 bytes
    Helper address is not set
    Directed broadcast forwarding is disabled
    Outgoing access list is 1
    Inbound access list is not set
<Output omitted>Questions:
```

```
R3#show ip int br g0/0/0
GigabitEthernet0/0/0 is up, line protocol is up (connected)
    Internet address is 192.168.30.1/24
    Broadcast address is 255.255.255.255
    Address determined by setup command
    MTU is 1500 bytes
    Helper address is not set
    Directed broadcast forwarding is disabled
    Outgoing access list is 1
    Inbound access list is not set
    Proxy ARP is enabled
    Security level is default
    Split horizon is enabled
    ICMP redirects are always sent
    ICMP unreachable messages are always sent
    ICMP mask replies are never sent
    IP fast switching is disabled
    IP fast switching on the same interface is disabled
    IP Flow switching is disabled
    IP fast switching turbo vector
    IP multicast fast switching is disabled
    IP multicast distributed fast switching is disabled
    Router Discovery is disabled
    IP output packet accounting is disabled
    IP access violation accounting is disabled
    TCP/IP header compression is disabled
    RTP/IP header compression is disabled
    Probe proxy name replies are disabled
    Policy routing is disabled
    Network address translation is disabled
    EGP Policy Mapping is disabled
    Input features: MCI Check
    WCCP Redirect outbound is disabled
    WCCP Redirect inbound is disabled
    WCCP Redirect exclude is disabled

R3#
```

- 3) Test the ACL to see if it allows traffic from the 192.168.10.0/24 network to access the 192.168.30.0/24 network.

From the PC-A command prompt, ping the PC-C IP address. Were the pings successful? Yes

```
C:\>ping 192.168.30.3

Pinging 192.168.30.3 with 32 bytes of data:

Reply from 192.168.30.3: bytes=32 time=42ms TTL=125
Reply from 192.168.30.3: bytes=32 time=30ms TTL=125
Reply from 192.168.30.3: bytes=32 time=21ms TTL=125
Reply from 192.168.30.3: bytes=32 time=23ms TTL=125

Ping statistics for 192.168.30.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 21ms, Maximum = 42ms, Average = 29ms

C:\>
```

- 4) Test the ACL to see if it allows traffic from the 192.168.20.0/24 network access to the 192.168.30.0/24 network.

From the PC-B command prompt, ping the PC-C IP address. Were the pings successful? Yes

```
C:\>ping 192.168.30.3

Pinging 192.168.30.3 with 32 bytes of data:

Reply from 192.168.30.3: bytes=32 time=9ms TTL=125
Reply from 192.168.30.3: bytes=32 time=2ms TTL=125
Reply from 192.168.30.3: bytes=32 time=40ms TTL=125
Reply from 192.168.30.3: bytes=32 time=36ms TTL=125

Ping statistics for 192.168.30.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 40ms, Average = 21ms
```

- 5) Should pings from PC-D to PC-C be successful? Ping from PC-D to PC-C to verify your answer. No, ping verifies that the ACL is working as intended.

```
C:\>ping 192.168.30.3

Pinging 192.168.30.3 with 32 bytes of data:

Reply from 192.168.40.1: Destination host unreachable.

Ping statistics for 192.168.30.3:
    Packets: Sent = 4, Received = 0, Lost = 4 (100%
loss),

C:\>
```

- d. From the R1 prompt, ping PC-C's IP address again.

```
R1# ping 192.168.30.3
```

Was the ping successful? Explain.

Answer No ping failed. When you ping from the router, it uses the closest interface to the destination as its source address. The pings had a source address of 10.1.1.1. The access list on R3 only allows the 192.168.10.0/24 and the 192.168.20.0/24 networks access.

```

R1#ping 192.168.30.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.30.3, timeout
is 2 seconds:
UUUUU
Success rate is 0 percent (0/5)

R1#

```

- e. Issue the **show access-lists 1** command again. Note that the command output displays information for the number of times each ACE was matched by traffic that reached interface Gigabit Ethernet 0/0/0.

```

R3# show access-lists 1
Standard IP access list 1
    permit 192.168.10.0 0.0.0.255 (4 match(es))
    permit 192.168.20.0 0.0.0.255 (4 match(es))
    deny any (4 match(es))

```

```

R3#show access-lists 1
Standard IP access list 1
    permit 192.168.10.0 0.0.0.255 (4 match(es))
    permit 192.168.20.0 0.0.0.255 (4 match(es))
    deny any (9 match(es))

R3#

```

Step 2: Configure a named standard ACL.

Create a named standard ACL that conforms to the following policy: allow traffic from all hosts on the 192.168.40.0/24 network access to all hosts on the 192.168.10.0/24 network. Also, only allow host PC-C access to the 192.168.10.0/24 network. The name of this access list should be called BRANCH-OFFICE-POLICY.

Following Cisco's recommended best practices, on which router would you place this ACL? R1

On which interface would you place this ACL? In what direction would you apply it?

Answer - G0/0/0. The ACL should be applied going out.

Placing the ACL on the S0/0/0 interface on R1 going in. This would effectively block all traffic from the LANs on R3 from getting to the 192.168.20.0/24 network.

- a. Create the standard named ACL BRANCH-OFFICE-POLICY on R1.

```

R1 (config)# ip access-list standard BRANCH-OFFICE-POLICY
R1 (config-std-nacl)# permit host 192.168.30.3
R1 (config-std-nacl)# permit 192.168.40.0 0.0.0.255
R1 (config-std-nacl)# end
R1#

```

```
*Feb 15 15:56:55.707: %SYS-5-CONFIG_I: Configured from console by console
```

Look at the first ACE in the access list. What is another way to write this? **Answer** - permit 192.168.30.3 0.0.0.0

- b. Apply the ACL to the appropriate interface in the proper direction.

```
R1# config t  
R1(config)# interface g0/0/0  
R1(config-if)# ip access-group BRANCH-OFFICE-POLICY out
```

```
R1#config t  
Enter configuration commands, one per line. End with CNTL/Z.  
R1(config)#interface g0/0/0  
R1(config-if)#ip access-group BRANCH-OFFICE-POLICY out  
R1(config-if)#end  
R1#  
%SYS-5-CONFIG_I: Configured from console by console  
  
R1#show acc  
R1#show access-lists  
R1#  
R1#  
R1#conf  
R1#configure t  
Enter configuration commands, one per line. End with CNTL/Z.  
R1(config)#  
R1(config)#ip access-list standard BRANCH-OFFICE-POLICY  
R1(config-std-nacl)#permit host 192.168.30.3  
R1(config-std-nacl)#permit 192.168.40.0 0.0.0.255  
R1(config-std-nacl)#end  
R1#  
%SYS-5-CONFIG_I: Configured from console by console
```

- c. Verify a named ACL.

- 1) On R1, issue the show access-lists command.

```
R1# show access-lists  
Standard IP access list BRANCH-OFFICE-POLICY  
    10 permit host 192.168.30.3  
    20 permit 192.168.40.0 0.0.0.255
```

```
R1#show access-lists  
Standard IP access list BRANCH-OFFICE-POLICY  
    10 permit host 192.168.30.3  
    20 permit 192.168.40.0 0.0.0.255  
  
R1#
```

Is there any difference between this ACL on R1 and the ACL on R3? If so, what is it?

Answer - deny any on R1, it is implied.

Explicitly configuring the *deny any* ACE is a good practice and reinforces the concept because it shows up in the output of the show access-lists command.

It is easy to forget the implicit deny any when troubleshooting ACLs.

This could easily result in traffic being denied that should have been allowed. In addition, if the explicit deny any ACE is present, it can be logged, and the number of matches for the ACE condition can be viewed with show access-lists.

- 2) On R1, issue the **show ip interface g0/0/0** command to verify that the ACL is configured on the interface.

```
R1# show ip interface g0/0/0
GigabitEthernet0/0/0 is up, line protocol is up (connected)
  Internet address is 192.168.10.1/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is BRANCH-OFFICE-POLICY
  Inbound access list is not set
<Output omitted>Question:
```

```
R1#show ip interface g0/0/0
GigabitEthernet0/0/0 is up, line protocol is up (connected)
  Internet address is 192.168.10.1/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is BRANCH-OFFICE-POLICY
  Inbound access list is not set
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachables are always sent
  ICMP mask replies are never sent
  IP fast switching is disabled
  IP fast switching on the same interface is disabled
  IP Flow switching is disabled
  IP Fast switching turbo vector
  IP multicast fast switching is disabled
  IP multicast distributed fast switching is disabled
  Router Discovery is disabled
  IP output packet accounting is disabled
  IP access violation accounting is disabled
  TCP/IP header compression is disabled
  RTP/IP header compression is disabled
  Probe proxy name replies are disabled
  Policy routing is disabled
  Network address translation is disabled
  BGP Policy Mapping is disabled
  Input features: MCI Check
  WCCP Redirect outbound is disabled
  WCCP Redirect inbound is disabled
  WCCP Redirect exclude is disabled
```

```
R1#
```

Test the ACL. From the command prompt on PC-C, ping the IP address of PC-A. Were the pings successful? Answer Yes

```
C:\>ping 192.168.10.3

Pinging 192.168.10.3 with 32 bytes of data:

Request timed out.
Reply from 192.168.10.3: bytes=32 time=2ms TTL=125
Reply from 192.168.10.3: bytes=32 time=4ms TTL=125
Reply from 192.168.10.3: bytes=32 time=4ms TTL=125

Ping statistics for 192.168.10.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 4ms, Average = 3ms

C:\>
```

- 3) Test the ACL to ensure that only the PC-C host is allowed access to the 192.168.10.0/24 network. You must do an extended ping and use the G0/0/0 address on R3 as your source. Ping PC-A's IP address.

R3# **ping**

Protocol [ip]:

Target IP address: **192.168.10.3**

Repeat count [5]:

Datagram size [100]:

Timeout in seconds [2]:

Extended commands [n]: y

Source address or interface: 192.168.30.1

Type of service [0]:

Set DF bit in IP header? [no]:

Validate reply data? [no]:

Data pattern [0xABCD]:

Loose, Strict, Record, Timestamp, Verbose[none]:

Sweep range of sizes [n]:

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.10.3, timeout is 2 seconds:

Packet sent with a source address of 192.168.30.1

U.U.U

Were the pings successful? No

```

R3#ping
Protocol [ip]: 192.168.10.3
% Unknown protocol - "192.168.10.3", type "ping ?" for help

R3#ping
Protocol [ip]:
Target IP address: 192.168.10.3
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 192.168.30.1
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.3, timeout is 2 seconds:
Packet sent with a source address of 192.168.30.1
UUUUU
Success rate is 0 percent (0/5)

```

R3#

- 4) Test the ACL to see if it allows traffic from the 192.168.40.0/24 network access to the 192.168.10.0/24 network. From the PC-D command prompt, ping the PC-A IP address.

Were the pings successful? **Answer Yes**

```

C:\>ping 192.168.10.3

Pinging 192.168.10.3 with 32 bytes of data:

Reply from 192.168.10.3: bytes=32 time=10ms TTL=125
Reply from 192.168.10.3: bytes=32 time=2ms TTL=125
Reply from 192.168.10.3: bytes=32 time=2ms TTL=125
Reply from 192.168.10.3: bytes=32 time=5ms TTL=125

Ping statistics for 192.168.10.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli seconds:
        Minimum = 2ms, Maximum = 10ms, Average = 4ms

C:\>

```

Part 3: Modify a Standard ACL

It is common in business for security policies to change. For this reason, ACLs may need to be modified. In Part 3, you will change one of the ACLs you configured previously to match a new management policy that is

being put in place.

Attempt to ping the server at 209.165.200.254 from PC-A. Notice that the ping is not successful. The ACL on R1 is blocking internet traffic from returning to PC-A. This is because the source address in the packets that are returned is not in the range of permitted addresses.

```
C:\>ping 209.165.200.254

Pinging 209.165.200.254 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 209.165.200.254:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Management has decided that [traffic that is returning from the 209.165.200.224/27 network should be allowed full access to the 192.168.10.0/24 network](#). Management also wants [ACLs on all routers to follow consistent rules](#). A **deny any** ACE should be placed at the end of all ACLs. You must modify the BRANCH-OFFICE-POLICY ACL.

You will add two additional lines to this ACL. There are two ways you could do this:

OPTION 1: Issue a **no ip access-list standard BRANCH-OFFICE-POLICY** command in global configuration mode. This would remove the ACL from the router. Depending upon the router IOS, one of the following scenarios would occur: all filtering of packets would be cancelled, and all packets would be allowed through the router; or, because you did not remove the **ip access-group** command from the G0/1 interface, filtering is

still in place. Regardless, when the ACL is gone, you could retype the whole ACL, or cut and paste it in from a text editor.

OPTION 2: You can modify ACLs in place by adding or deleting specific lines within the ACL itself. This can come in handy, especially with ACLs that are long. The retying of the whole ACL or cutting and pasting can easily lead to errors. Modifying specific lines within the ACL is easily accomplished.

For this activity, use Option 2.

Step 1: Modify a named standard ACL.

- From R1, issue the **show access-lists** command.

```
R1# show access-lists

Standard IP access list BRANCH-OFFICE-POLICY

    10 permit 192.168.30.3 (8 matches)

    20 permit 192.168.40.0 0.0.0.255 (5 matches)
```

- Add two additional lines at the end of the ACL. From global config mode, modify the ACL, BRANCH-OFFICE-POLICY.

```
R1#(config)# ip access-list standard BRANCH-OFFICE-POLICY
R1(config-std-nacl)# 30 permit 209.165.200.224 0.0.0.31
R1(config-std-nacl)# 40 deny any
R1(config-std-nacl)# end
```

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#ip access-list standard BRANCH-OFFICE-POLICY
R1(config-std-nacl)#
R1(config-std-nacl)#30 permit 209.165.200.224 0.0.0.31
R1(config-std-nacl)#40 deny any
R1(config-std-nacl)#end
R1#
%SYS-5-CONFIG_I: Configured from console by console
```

- Verify the ACL.

- On R1, issue the **show access-lists** command.

```
R1# show access-lists

Standard IP access list BRANCH-OFFICE-POLICY

    10 permit 192.168.30.3 (8 matches)

    20 permit 192.168.40.0, wildcard bits 0.0.0.255 (5 matches)
    30 permit 209.165.200.224, wildcard bits 0.0.0.31

    40 deny any
```

```
R1#
R1#show access-lists
Standard IP access list BRANCH-OFFICE-POLICY
    10 permit host 192.168.30.3 (4 match(es))
    20 permit 192.168.40.0 0.0.0.255 (4 match(es))
    30 permit 209.165.200.224 0.0.0.31
    40 deny any

R1#
```

Do you have to apply the BRANCH-OFFICE-POLICY to the G0/0/0 interface on R1? **Answer** - No, the ip access-group BRANCH-OFFICE-POLICY out command is still in place on G0/1.

- 2) Test the ACL to see if it allows traffic from the 209.165.200.224/27 network access to return to the 192.168.10.0/24 network. From PC-A, ping the server at 209.165.200.254.

Were the pings successful? Yes

```
C:\>ping 209.165.200.254
Pinging 209.165.200.254 with 32 bytes of data:
Reply from 209.165.200.254: bytes=32 time=10ms TTL=125
Reply from 209.165.200.254: bytes=32 time=7ms TTL=125
Reply from 209.165.200.254: bytes=32 time=5ms TTL=125
Reply from 209.165.200.254: bytes=32 time=3ms TTL=125

Ping statistics for 209.165.200.254:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 3ms, Maximum = 10ms, Average = 6ms

C:\>
```

3.2.11.6 Reflection Questions

1. As you can see, standard ACLs are very powerful and work quite well. Why would you ever have the need for using extended ACLs?

Answer

Standard ACLs can only filter based on the source address. Also, they are not granular. They allow or deny everything (all protocols and services).

standard ACLs must be applied as close to the destination as possible. This allows unnecessary traffic to use network bandwidth

Extended ACLs, while harder to write, are well-suited for complex networks where you may need to allow traffic for only certain Layer 4 ports to have access to networks while denying others.

Extended ACLs can block traffic close to the source. This prevents unnecessary traffic from traveling to the destination where it is blocked.

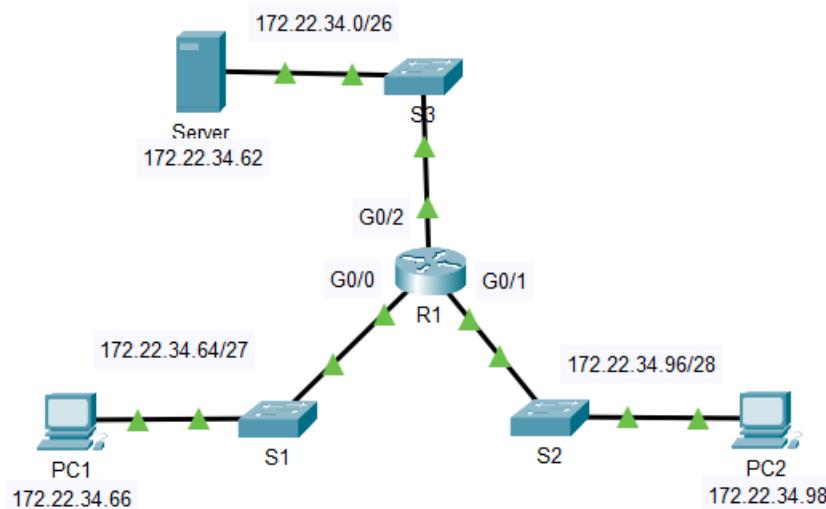
2. More typing is typically required when using a named ACL as opposed to a numbered ACL. Why would you choose named ACLs over numbered?

Answer

- 1- Using named ACLs gives you the ability to modify specific lines within the ACL itself, without retyping the entire list.
- 2- Having a named ACL is a good best practice as it helps to document the purpose of the ACL with a descriptive name.

3.2.12 Exercise 5.4.12 - Packet Tracer - Configure Extended ACLs - Scenario 1

3.2.12.1 Topology



3.2.12.2 Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0	172.22.34.65	255.255.255.224	N/A
	G0/1	172.22.34.97	255.255.255.240	
	G0/2	172.22.34.1	255.255.255.192	
Server	NIC	172.22.34.62	255.255.255.192	172.22.34.1
PC1	NIC	172.22.34.66	255.255.255.224	172.22.34.65
PC2	NIC	172.22.34.98	255.255.255.240	172.22.34.97

3.2.12.3 Objectives

Part 1: Configure, Apply and Verify an Extended Numbered ACL Part 2:

Configure, Apply and Verify an Extended Named ACL

3.2.12.4 Background / Scenario

Two employees need access to services provided by the server. **PC1** only needs FTP access while **PC2** only needs web access. Both computers need to be able to ping the server, but not each other.

3.2.12.5 Instructions

Part 1: Configure, Apply and Verify an Extended Numbered ACL

Step 1: Configure an ACL to permit FTP and ICMP from PC1 LAN.

- From global configuration mode on R1, enter the following command to determine the first valid number for an extended access list.

```
R1(config)# access-list ?
```

```
<1-99>      IP standard access list
```

```
<100-199>  IP extended access list
```

```
R1>ena
R1>enable
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#access-list ?
  <1-99>    IP standard access list
  <100-199>  IP extended access list
R1(config)#access-list
```

- b. Add **100** to the command, followed by a question mark.

```
R1(config)# access-list 100 ?
  deny   Specify packets to reject
  permit  Specify packets to forward
  remark Access list entry comment
```

```
  remark Access list entry comment
R1(config)#access-list 100 ?
  deny   Specify packets to reject
  permit  Specify packets to forward
  remark Access list entry comment
R1(config)#access-list 100
```

- c. To permit FTP traffic, enter **permit**, followed by a question mark.

```
R1(config)# access-list 100 permit ?
  ahp    Authentication Header Protocol
  eigrp  Cisco's EIGRP routing protocol
  esp    Encapsulation Security Payload
  gre    Cisco's GRE tunneling
  icmp   Internet Control Message Protocol
  ip     Any Internet Protocol
  ospf   OSPF routing protocol
  tcp    Transmission Control Protocol
  udp    User Datagram Protocol
```

```
R1(config)#access-list 100 permit ?
  ahp    Authentication Header Protocol
  eigrp  Cisco's EIGRP routing protocol
  esp    Encapsulation Security Payload
  gre    Cisco's GRE tunneling
  icmp   Internet Control Message Protocol
  ip     Any Internet Protocol
  ospf   OSPF routing protocol
  tcp    Transmission Control Protocol
  udp    User Datagram Protocol
```

- d. When configured and applied, this ACL should permit FTP and ICMP. ICMP is listed above, but FTP is not. This is because FTP is an application layer protocol that uses TCP at the transport layer. Enter **TCP** to further refine the ACL help.

```
R1(config)# access-list 100 permit tcp ?
```

```
  A.B.C.D  Source address
  any      Any source host
```

host A single source host

```
|       user password removed
R1(config)#access-list 100 permit tcp ?
  A.B.C.D  Source address
  any      Any source host
  host     A single source host
```

- e. The source address can represent a single device, such as PC1, by using the **host** keyword and then the IP address of PC1. Using the keyword **any** permits any host on any network. Filtering can also be done by a network address. In this case, it is any host that has an address belonging to the 172.22.34.64/27 network. Enter this network address, followed by a question mark.

```
R1(config)# access-list 100 permit tcp 172.22.34.64 ?
```

A.B.C.D Source wildcard bits

```
| R1(config)# access-list 100 permit tcp 172.22.34.64 ?
|   A.B.C.D  Source wildcard bits
```

- f. Calculate the wildcard mask by determining the binary opposite of the /27 subnet mask.

```
1111111.1111111.1111111.1100000 = 255.255.255.224
00000000.00000000.00000000.0001111 = 0.0.0.31
```

- g. Enter the wildcard mask, followed by a question mark.

```
R1(config)# access-list 100 permit tcp 172.22.34.64 0.0.0.31 ?
```

A.B.C.D Destination address

any Any destination host

eq Match only packets on a given port number

gt Match only packets with a greater port number

host A single destination host

lt Match only packets with a lower port number

neq Match only packets not on a given port number

range Match only packets in the range of port numbers

```
|       user password removed
R1(config)# access-list 100 permit tcp 172.22.34.64 0.0.0.31 ?
  A.B.C.D  Destination address
  any      Any destination host
  eq      Match only packets on a given port number
  gt      Match only packets with a greater port number
  host    A single destination host
  lt      Match only packets with a lower port number
  neq    Match only packets not on a given port number
  range   Match only packets in the range of port numbers
```

- h. Configure the destination address. In this scenario, we are filtering traffic for a single destination, which is the server. Enter the **host** keyword followed by the server's IP address.

```
R1(config)# access-list 100 permit tcp 172.22.34.64 0.0.0.31 host
```

```
172.22.34.62 ?
```

```
dscp      Match packets with given dscp value
eq       Match only packets on a given port number
established established

gt        Match only packets with a greater port number
lt        Match only packets with a lower port number
neq      Match only packets not on a given port number
precedence Match packets with given precedence value

range     Match only packets in the range of port numbers

<cr>
```

```
R1(config)# access-list 100 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 ?
dscp      Match packets with given dscp value
eq       Match only packets on a given port number
established established
gt        Match only packets with a greater port number
lt        Match only packets with a lower port number
neq      Match only packets not on a given port number
precedence Match packets with given precedence value
range     Match only packets in the range of port numbers
<cr>
```

- i. Notice that one of the options is <cr> (carriage return). In other words, you can press **Enter** and the statement would permit all TCP traffic. However, we are only permitting FTP traffic; therefore, enter the **eq** keyword, followed by a question mark to display the available options. Then, enter **ftp** and press **Enter**.

```
R1(config)# access-list 100 permit tcp 172.22.34.64 0.0.0.31 host
```

```
172.22.34.62 eq ?
```

```
<0-65535> Port number
ftp      File Transfer Protocol (21)
pop3    Post Office Protocol v3 (110)

smtp    Simple Mail Transport Protocol (25)
telnet   Telnet (23)

www     World Wide Web (HTTP, 80)
```

```
    R1(config)# access-list 100 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 eq ?
    <0-65535> Port number
    ftp      File Transfer Protocol (21)
    pop3    Post Office Protocol v3 (110)
    smtp    Simple Mail Transport Protocol (25)
    telnet   Telnet (23)
    www     World Wide Web (HTTP, 80)
```

```
R1(config)# access-list 100 permit tcp 172.22.34.64 0.0.0.31 host
172.22.34.62 eq ftp
```

- j. Create a second access list statement to permit ICMP (ping, etc.) traffic from PC1 to Server. Note that the access list number remains the same and a specific type of ICMP traffic does not need to be specified.

```
R1(config)# access-list 100 permit icmp 172.22.34.64 0.0.0.31 host  
172.22.34.62
```

- k. All other traffic is denied, by default.

- l. Execute the **show access-list** command and verify that access list 100 contains the correct statements. Notice that the statement **deny any any** does not appear at the end of the access list. The default execution of an access list is that if a packet does not match a statement in the access list, it is not permitted through the interface.

```
R1#show access-lists
```

```
Extended IP access list 100
```

```
    10 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 eq ftp  
    20 permit icmp 172.22.34.64 0.0.0.31 host 172.22.34.62
```

```
R1#show access-lists  
Extended IP access list 100  
    10 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 eq ftp  
    20 permit icmp 172.22.34.64 0.0.0.31 host 172.22.34.62
```

```
R1#
```

Step 2: Apply the ACL on the correct interface to filter traffic.

From **R1**'s perspective, the traffic that ACL 100 applies to is inbound from the network connected to the Gigabit Ethernet 0/0 interface. Enter interface configuration mode and apply the ACL.

Note: On an actual operational network, it is not a good practice to apply an untested access list to an active interface.

```
R1(config)# interface gigabitEthernet 0/0
```

```
R1(config-if)# ip access-group 100 in
```

```
R1#configure t  
R1#configure terminal  
Enter configuration commands, one per line. End with CNTL/Z.  
R1(config)#interface gigabitEthernet 0/0  
R1(config-if)#ip access-group 100 in  
R1(config-if)#end  
R1#  
%SYS-5-CONFIG_I: Configured from console by console  
  
R1#copy running-config startup-config  
Destination filename [startup-config]?  
Building configuration...  
[OK]  
R1#
```

```
R1#show running-config | section interface
interface GigabitEthernet0/0
  ip address 172.22.34.65 255.255.255.224
  ip access-group 100 in
  duplex auto
  speed auto
interface GigabitEthernet0/1
  ip address 172.22.34.97 255.255.255.240
  duplex auto
  speed auto
interface GigabitEthernet0/2
  ip address 172.22.34.1 255.255.255.192
  duplex auto
  speed auto
interface Vlan1
  no ip address
  shutdown
R1#
```

```
R1#show running-config | section access
  ip access-group 100 in
access-list 100 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 eq ftp
access-list 100 permit icmp 172.22.34.64 0.0.0.31 host 172.22.34.62
R1#
```

```
R1#show ip interface g0/0
GigabitEthernet0/0 is up, line protocol is up (connected)
  Internet address is 172.22.34.65/27
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is not set
  Inbound access list is 100
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachables are always sent
  ICMP mask replies are never sent
  IP fast switching is disabled
  IP fast switching on the same interface is disabled
  IP Flow switching is disabled
  IP Fast switching turbo vector
  IP multicast fast switching is disabled
  IP multicast distributed fast switching is disabled
  Router Discovery is disabled
  IP output packet accounting is disabled
  IP access violation accounting is disabled
  TCP/IP header compression is disabled
  RTP/IP header compression is disabled
  Probe proxy name replies are disabled
  Policy routing is disabled
  Network address translation is disabled
  BGP Policy Mapping is disabled
  Input features: MCI Check
  WCCP Redirect outbound is disabled
  WCCP Redirect inbound is disabled
  WCCP Redirect exclude is disabled
```

```
R1#
```

Step 3: Verify the ACL implementation.

- a. Ping from PC1 to Server. If the pings are unsuccessful, verify the IP addresses before continuing.
- b. FTP from PC1 to Server. The username and password are both **cisco**.
PC> **ftp 172.22.34.62**
- c. Exit the FTP service.
ftp> **quit**
- d. Ping from PC1 to PC2. The destination host should be unreachable, because the ACL did not explicitly permit the traffic.

```

C:\>ping 172.22.34.62
Pinging 172.22.34.62 with 32 bytes of data:
Request timed out.
Reply from 172.22.34.62: bytes=32 time<1ms TTL=127
Reply from 172.22.34.62: bytes=32 time<1ms TTL=127
Reply from 172.22.34.62: bytes=32 time<1ms TTL=127

Ping statistics for 172.22.34.62:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ftp 172.22.34.62
Trying to connect...172.22.34.62
Connected to 172.22.34.62
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>quit

221- Service closing control connection.
c:\>ping 172.22.34.98
Pinging 172.22.34.98 with 32 bytes of data:

Reply from 172.22.34.65: Destination host unreachable.

Ping statistics for 172.22.34.98:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

```

Part 2: Configure, Apply and Verify an Extended Named ACL

Step 1: Configure an ACL to permit HTTP access and ICMP from PC2 LAN.

- Named ACLs start with the **ip** keyword. From global configuration mode of R1, enter the following command, followed by a question mark.

```
R1(config)# ip access-list ?
      extended  Extended Access List
      standard   Standard Access
      List
```

- You can configure named standard and extended ACLs. This access list filters both source and destination IP addresses; therefore, it must be extended. Enter **HTTP_ONLY** as the name. (For Packet Tracer scoring, the name is case-sensitive and the access list statements must be the correct order.)

```
R1(config)# ip access-list extended HTTP_ONLY
```

- The prompt changes. You are now in extended named ACL configuration mode. All devices on the **PC2** LAN need TCP access. Enter the network address, followed by a question mark.

```
R1(config-ext-nacl)# permit tcp 172.22.34.96 ?
```

```
      A.B.C.D  Source wildcard bits
```

- An alternative way to calculate a wildcard is to subtract the subnet mask from 255.255.255.255.

```

255.255.255.255
-
- 255.255.255.240
-----
=
0. 0. 0. 15
R1(config-ext-nacl)# permit tcp 172.22.34.96 0.0.0.15

```

- e. Finish the statement by specifying the server address as you did in Part 1 and filtering **www** traffic.

```
R1(config-ext-nacl)# permit tcp 172.22.34.96 0.0.0.15 host 172.22.34.62 eq www
```

- f. Create a second access list statement to permit ICMP (ping, etc.) traffic from **PC2** to **Server**. Note: The prompt remains the same and a specific type of ICMP traffic does not need to be specified.

```
R1(config-ext-nacl)# permit icmp 172.22.34.96 0.0.0.15 host 172.22.34.62
```

- g. All other traffic is denied, by default. Exit extended named ACL configuration mode.

- h. Execute the **show access-list** command and verify that access list **HTTP_ONLY** contains the correct statements.

```
R1# show access-lists
Extended IP access list 100
10 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 eq ftp
20 permit icmp 172.22.34.64 0.0.0.31 host 172.22.34.62
Extended IP access list HTTP_ONLY
10 permit tcp 172.22.34.96 0.0.0.15 host 172.22.34.62 eq www
20 permit icmp 172.22.34.96 0.0.0.15 host 172.22.34.62
```

```

R1#enable
R1(config)
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#ip access-list ?
  extended Extended Access List
  standard Standard Access List
R1(config)#ip access-list ex
R1(config)#ip access-list extended HTTP_ONLY
R1(config-ext-nacl)#permit tcp 172.22.34.96 ?
  A.R.C.D. Source wildcard bits
R1(config-ext-nacl)#permit tcp 172.22.34.96 0.0.0.15
% Incomplete command.
R1(config ext nacl)#permit tcp 172.22.34.96 0.0.0.15 host 172.22.34.62 eq www
R1(config-ext-nacl)#permit icmp 172.22.34.96 0.0.0.15 host 172.22.34.62
R1(config ext nacl)#end
R1#
*SYS-5-CONFIG_T: Configured from console by console

R1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R1#show access-lists
Extended IP access list 100
  10 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 eq ftp (10 match(es))
  20 permit icmp 172.22.34.64 0.0.0.31 host 172.22.34.62 (4 match(es))
Extended IP access list HTTP_ONLY
  10 permit tcp 172.22.34.96 0.0.0.15 host 172.22.34.62 eq www
  20 permit icmp 172.22.34.96 0.0.0.15 host 172.22.34.62

R1#

```

Step 2: Apply the ACL on the correct interface to filter traffic.

From **R1**'s perspective, the traffic that access list **HTTP_ONLY** applies to is inbound from the network connected to the Gigabit Ethernet 0/1 interface. Enter interface configuration mode and apply the ACL.

Note: On an actual operational network, it is not a good practice to apply an untested access list to an active interface. It should be avoided if possible.

```
R1(config)# interface gigabitEthernet 0/1
```

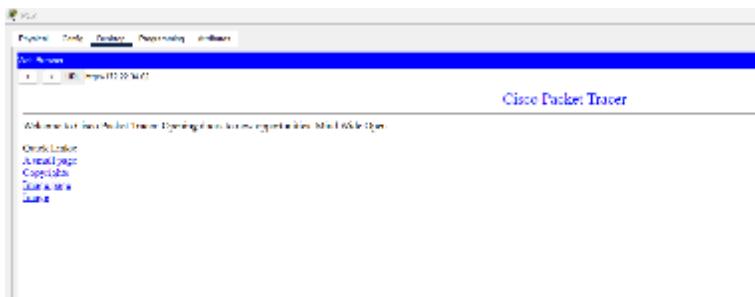
```
R1(config-if)# ip access-group HTTP_ONLY in
```

```
R1#  
R1#  
R1#ena  
R1#enable  
R1#conf  
R1#configure t  
Enter configuration commands, one per line. End with CNTL/Z.  
R1(config)#interface gigabitEthernet 0/1  
R1(config-if)#ip access-group HTTP_ONLY in  
R1(config-if)#end  
R1#  
#SYS-5-CONFIG_I: Configured from console by console  
copy running-config startup-config  
Destination filename [startup-config]?  
Building configuration...  
[OK]  
R1#
```

Step 3: Verify the ACL implementation.

- Ping from **PC2** to **Server**. If the ping is unsuccessful, verify the IP addresses before continuing.
- From PC2 open a web browser and enter the IP address of the Server. The web page of the Server should be displayed.
- FTP from **PC2** to **Server**. The connection should fail. If not, troubleshoot the access list statements and the access-group configurations on the interfaces.

```
Cisco Packet Tracer PC Command Line 1.0  
C:\>ping 172.22.34.62  
  
Pinging 172.22.34.62 with 32 bytes of data:  
  
Reply from 172.22.34.62: bytes=32 time<1ms TTL=127  
  
Ping statistics for 172.22.34.62:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
    Approximate round trip times in milli-seconds:  
        Minimum = 0ms, Maximum = 0ms, Average = 0ms  
  
C:\>ftp 172.22.34.62  
Trying to connect...172.22.34.62  
  
error opening ftp://172.22.34.62/ (Timed out)  
  
.  
  
(Disconnecting from ftp server)  
  
C:\>
```

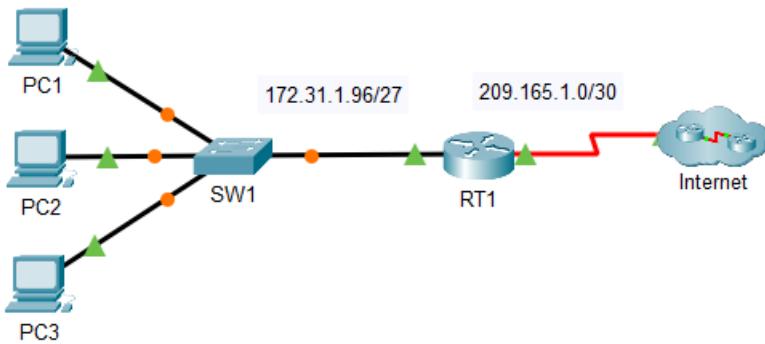


3.2.12.6 Script

```
enable
configure terminal
access-list 100 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 eq ftp
access-list 100 permit icmp 172.22.34.64 0.0.0.31 host 172.22.34.62
interface gigabitEthernet 0/0
  ip access-group 100 in
ip access-list extended HTTP_ONLY
  permit tcp 172.22.34.96 0.0.0.15
  permit tcp 172.22.34.96 0.0.0.15 host 172.22.34.62 eq www
  permit icmp 172.22.34.96 0.0.0.15 host 172.22.34.62
interface gigabitEthernet 0/1
  ip access-group HTTP_ONLY in
end
copy running-config startup-config
```

3.2.13 Exercise 5.4.13 - Packet Tracer - Configure Extended IPv4 ACLs - Scenario 2

3.2.13.1 Topology



3.2.13.2 Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
RT1	G0/0	172.31.1.126	255.255.255.224	N/A

	S0/0/0	209.165.1.2	255.255.255.252	
PC1	NIC	172.31.1.101	255.255.255.224	172.31.1.126
PC2	NIC	172.31.1.102	255.255.255.224	172.31.1.126
PC3	NIC	172.31.1.103	255.255.255.224	172.31.1.126
Server1	NIC	64.101.255.254		
Server2	NIC	64.103.255.254		

3.2.13.3 Objectives

Part 1: Configure a Named Extended ACL

Part 2: Apply and Verify the Extended ACL

3.2.13.4 Background / Scenario

In this scenario, specific devices on the LAN are allowed to various services on servers located on the internet.

3.2.13.5 Instructions

Part 1: Configure a Named Extended ACL

Configure one named ACL to implement the following policy:

- Block HTTP and HTTPS access from **PC1** to **Server1** and **Server2**. The servers are inside the cloud and you only know their IP addresses.
- Block FTP access from **PC2** to **Server1** and **Server2**.
- Block ICMP access from **PC3** to **Server1** and **Server2**.

Note: For scoring purposes, you must configure the statements in the order specified in the following steps.

Step 1: Deny PC1 access to HTTP and HTTPS services on Server1 and Server2.

- a. Create a named extended IP access list on router RT1 which will deny **PC1** access to the HTTP and HTTPS services of **Server1** and **Server2**. Four access control statements are required. Use **LimitedAccess** as the name of the named access list in this activity.

What is the command to begin the configuration of an extended access list with the name LimitedAccess?

Answer ip access-list extended ACL

- b. Begin the ACL configuration with a statement that denies access from **PC1** to **Server1**, only for HTTP (port 80). Refer to the addressing table for the IP address of **PC1** and **Server1**.

RT1(config-ext-nacl)# **deny tcp host 172.31.1.101 host 64.101.255.254 eq 80**

- c. Next, enter the statement that denies access from **PC1** to **Server1**, only for HTTPS (port 443).

RT1(config-ext-nacl)# **deny tcp host 172.31.1.101 host 64.101.255.254 eq 443**

- d. Enter the statement that denies access from **PC1** to **Server2**, only for HTTP. Refer to the addressing table for the IP address of **Server 2**.

RT1(config-ext-nacl)# **deny tcp host 172.31.1.101 host 64.103.255.254 eq 80**

- e. Enter the statement that denies access from **PC1** to **Server2**, only for HTTPS.

RT1(config-ext-nacl)# **deny tcp host 172.31.1.101 host 64.103.255.254 eq 443**

Step 2: Deny PC2 to access FTP services on Server1 and Server2.

Refer to the addressing table for the IP address of **PC2**.

- a. Enter the statement that denies access from **PC2** to **Server1**, only for FTP (port 21 only).

RT1(config-ext-nacl)# **deny tcp host 172.31.1.102 host 64.101.255.254 eq 21**

- b. Enter the statement that denies access from **PC2** to **Server2**, only for FTP (port 21 only).

RT1(config-ext-nacl)# **deny tcp host 172.31.1.102 host 64.103.255.254 eq 21**

Step 3: Deny PC3 to ping Server1 and Server2.

Refer to the addressing table for the IP address of **PC3**.

- a. Enter the statement that denies ICMP access from **PC3** to **Server1**.

RT1(config-ext-nacl)# **deny icmp host 172.31.1.103 host 64.101.255.254**

- b. Enter the statement that denies ICMP access from **PC3** to **Server2**.

RT1(config-ext-nacl)# **deny icmp host 172.31.1.103 host 64.103.255.254**

Step 4: Permit all other IP traffic.

By default, an access list denies all traffic that does not match any rule in the list. Enter the command that permits all traffic that does not match any of the configured access list statements. Answer permit ip any any

Step 5: Verify the access list configuration before applying it to an interface.

Before any access list is applied, the configuration needs to be verified to make sure that there are no typographical errors and that the statements are in the correct order. To view the current configuration of the access list, use either the **show access-lists** or the **show running-config** command.

RT1# **show access-lists**

Extended IP access list LimitedAccess

```
10    deny   tcp    host   172.31.1.101 host   64.101.255.254      eq    www
20    deny   tcp    host   172.31.1.101 host   64.101.255.254      eq    443
30    deny   tcp    host   172.31.1.101 host   64.103.255.254      eq    www
40    deny   tcp    host   172.31.1.101 host   64.103.255.254      eq    443
50    deny   tcp    host   172.31.1.102 host   64.101.255.254      eq    ftp
60    deny   tcp    host   172.31.1.102 host   64.103.255.254      eq    ftp
70    deny icmp host 172.31.1.103 host 64.101.255.254
80    deny icmp host 172.31.1.103 host 64.103.255.254
90    permit ip any any
```

```
RT1#config t
Enter configuration commands, one per line. End with CNTL/D/Y.
RT1(config)#ip access-list extended ACL
RT1(config-ext-nacl)#deny tcp host 172.31.1.101 host 64.101.255.254 eq 80
RT1(config-ext-nacl)#deny tcp host 172.31.1.101 host 64.101.255.254 eq 443
RT1(config ext nacl)#deny tcp host 172.31.1.101 host 64.103.255.254 eq 80
RT1(config ext nacl)#deny tcp host 172.31.1.101 host 64.103.255.254 eq 443
RT1(config ext nacl)#deny tcp host 172.31.1.102 host 64.101.255.254 eq 21
RT1(config-ext-nacl)#deny tcp host 172.31.1.102 host 64.103.255.254 eq 21
RT1(config-ext-nacl)#deny icmp host 172.31.1.103 host 64.101.255.254
RT1(config-ext-nacl)#deny icmp host 172.31.1.103 host 64.103.255.254
RT1(config ext nacl)#permit ip any any
RT1(config ext nacl)#end
RT1#
*SYN-5-CONFIG_I: Configured from console by console

RT1#show access-lists
RT1#show access-lists
Extended IP access list ACL
  10 deny tcp host 172.31.1.101 host 64.101.255.254 eq www
  20 deny tcp host 172.31.1.101 host 64.101.255.254 eq 443
  30 deny tcp host 172.31.1.101 host 64.103.255.254 eq www
  40 deny tcp host 172.31.1.101 host 64.103.255.254 eq 443
  50 deny tcp host 172.31.1.102 host 64.101.255.254 eq ftp
  60 deny tcp host 172.31.1.102 host 64.103.255.254 eq ftp
  70 deny icmp host 172.31.1.103 host 64.101.255.254
  80 deny icmp host 172.31.1.103 host 64.103.255.254
  90 permit ip any any
```

RT1# **show running-config | begin access-list**

ip access-list extended LimitedAccess

```
deny   tcp    host   172.31.1.101 host   64.101.255.254      eq    www
deny   tcp    host   172.31.1.101 host   64.101.255.254      eq    443
deny   tcp    host   172.31.1.101 host   64.103.255.254      eq    www
```

```

deny  tcp    host   172.31.1.101  host   64.103.255.254      eq    443
deny  tcp    host   172.31.1.102  host   64.101.255.254      eq    ftp
deny  tcp    host   172.31.1.102  host   64.103.255.254      eq    ftp
deny icmp host 172.31.1.103 host 64.101.255.254
deny icmp host 172.31.1.103 host 64.103.255.254 permit ip any any

RT1#show running-config | begin access-list
ip access-list extended ACL
deny tcp host 172.31.1.101 host 64.101.255.254 eq www
deny tcp host 172.31.1.101 host 64.101.255.254 eq 443
deny tcp host 172.31.1.101 host 64.103.255.254 eq www
deny tcp host 172.31.1.101 host 64.103.255.254 eq 443
deny tcp host 172.31.1.102 host 64.101.255.254 eq ftp
deny tcp host 172.31.1.102 host 64.103.255.254 eq ftp
deny icmp host 172.31.1.103 host 64.101.255.254
deny icmp host 172.31.1.103 host 64.103.255.254
permit ip any any
!
!
!
!
line con 0
!
line aux 0
!
line vty 0 4
 login
!
--More--

```

Note: The difference between the output of the **show access-lists** command and the output of the **show running-config** command is that the **show access-lists** command includes the sequence numbers assigned to the configuration statements. These sequence numbers enable the editing, deleting, and inserting of single lines within the access list configuration. Sequence numbers also define the processing order of individual access control statements, starting with the lowest sequence number.

Part 2: Apply and Verify the Extended ACL

The traffic to be filtered is coming from the 172.31.1.96/27 network and is destined for remote networks. Appropriate ACL placement depends on the relationship of the traffic with respect to **RT1**. In general, extended access lists should be placed on the interface closest to the source of the traffic.

Step 1: Apply the ACL to the correct interface and in the correct direction.

Note: In an actual operational network, an untested ACL should never be applied to an active interface. This is not a good practice and can disrupt network operation.

On which interface should the named ACL be applied, and in which direction?

Answer - interface g0/0 in

Enter the configuration commands to apply the ACL to the interface.

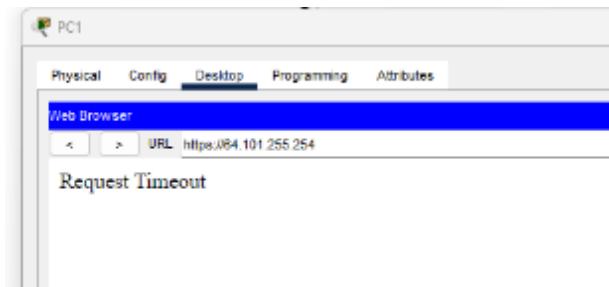
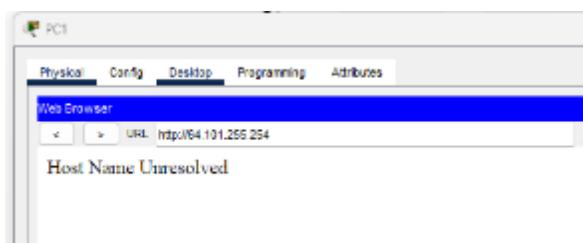
```
RT1(config)#interface g0/0
RT1(config-if)#ip access-group ACL in
RT1(config-if)#end
```

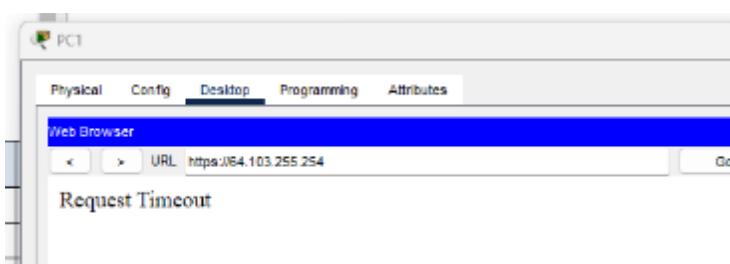
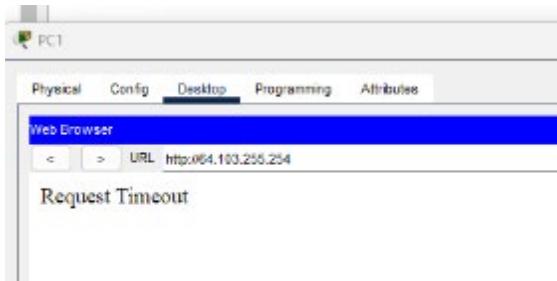
```
RT1#config t
Enter configuration commands, one per line. End with CNTL/Z.
RT1(config)#interface g0/0
RT1(config-if)#ip access-group ACL in
RT1(config-if)#end
RT1#
%SYS-5-CONFIG_I: Configured from console by console
```

Step 2: Test access for each PC.

- Access the websites of **Server1** and **Server2** using the web browser of **PC1**. Use both the HTTP and HTTPS protocols. Use the **show access-lists** command to view which access list statement permitted or denied the traffic. The output of the **show access-lists** command displays the number of packets that match each statement since the last time the counters were cleared, or the router rebooted.

Note: To clear the counters on an access list, use the **clear access-list counters** command.





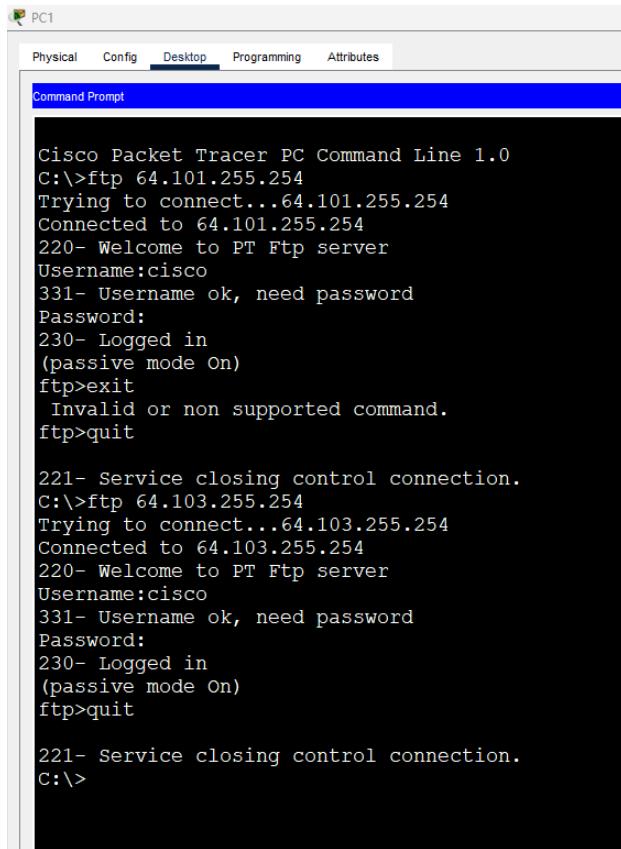
RT1#show ip access-lists

Extended IP access list LimitedAccess

10	deny	tcp	host	172.31.1.101 host 64.101.255.254	eq www (12 match(es))
20	deny	tcp	host	172.31.1.101 host 64.101.255.254	eq 443 (12 match(es))
30	deny	tcp	host	172.31.1.101 host 64.103.255.254	eq www
40	deny	tcp	host	172.31.1.101 host 64.103.255.254	eq 443
50	deny	tcp	host	172.31.1.102 host 64.101.255.254	eq ftp
60	deny	tcp	host	172.31.1.102 host 64.103.255.254	eq ftp
70	deny	icmp	host	172.31.1.103 host 64.101.255.254	
80	deny	icmp	host	172.31.1.103 host 64.103.255.254	
90	permit	ip	any	any	

```
RT1#show ip access-lists
Extended IP access list ACL
10 deny tcp host 172.31.1.101 host 64.101.255.254 eq www (10 match(es))
20 deny tcp host 172.31.1.101 host 64.101.255.254 eq 443 (29 match(es))
30 deny tcp host 172.31.1.101 host 64.103.255.254 eq www (202 match(es))
40 deny tcp host 172.31.1.101 host 64.103.255.254 eq 443 (64 match(es))
50 deny tcp host 172.31.1.102 host 64.101.255.254 eq ftp
60 deny tcp host 172.31.1.102 host 64.103.255.254 eq ftp
70 deny icmp host 172.31.1.103 host 64.101.255.254
80 deny icmp host 172.31.1.103 host 64.103.255.254
90 permit ip any any
```

- b. Access FTP of **Server1** and **Server2** using **PC1**. The username and password is **cisco**.



PC1

Physical Config Desktop Programming Attributes

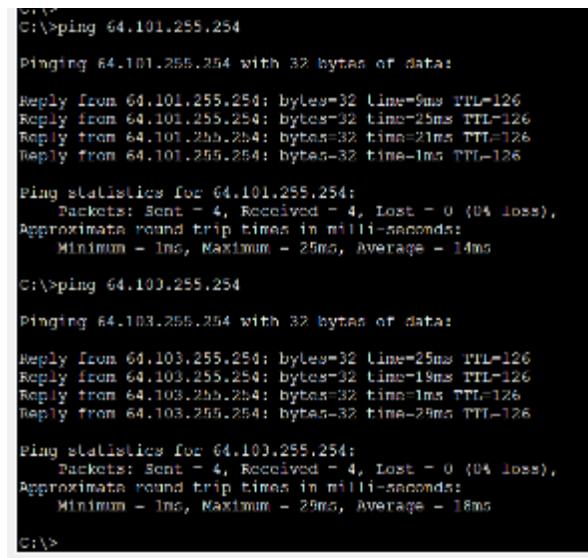
Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:>ftp 64.101.255.254
Trying to connect...64.101.255.254
Connected to 64.101.255.254
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>exit
  Invalid or non supported command.
ftp>quit

221- Service closing control connection.
C:>ftp 64.103.255.254
Trying to connect...64.103.255.254
Connected to 64.103.255.254
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>quit

221- Service closing control connection.
C:>
```

c. Ping **Server1** and **Server2** from PC1.



```
C:\>ping 64.101.255.254
Pinging 64.101.255.254 with 32 bytes of data:
Reply from 64.101.255.254: bytes=32 time=9ms TTL=126
Reply from 64.101.255.254: bytes=32 time=29ms TTL=126
Reply from 64.101.255.254: bytes=32 time=21ms TTL=126
Reply from 64.101.255.254: bytes=32 time=1ms TTL=126

Ping statistics for 64.101.255.254:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 1ms, Maximum = 29ms, Average = 14ms

C:\>ping 64.103.255.254
Pinging 64.103.255.254 with 32 bytes of data:
Reply from 64.103.255.254: bytes=32 time=25ms TTL=126
Reply from 64.103.255.254: bytes=32 time=19ms TTL=126
Reply from 64.103.255.254: bytes=32 time=1ms TTL=126
Reply from 64.103.255.254: bytes=32 time=29ms TTL=126

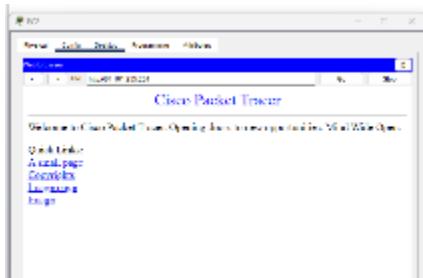
Ping statistics for 64.103.255.254:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 1ms, Maximum = 29ms, Average = 18ms

C:\>
```

d. Repeat Step 2a to Step 2c with **PC2** and **PC3** to verify proper access list operation.

PC2

Web access OK



Ping Ok

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 64.101.255.254

Pinging 64.101.255.254 with 32 bytes of data:
Reply from 64.101.255.254: bytes=32 time=18ms TTL=126
Reply from 64.101.255.254: bytes=32 time=16ms TTL=126
Reply from 64.101.255.254: bytes=32 time=9ms TTL=126
Reply from 64.101.255.254: bytes=32 time=40ms TTL=126

Ping statistics for 64.101.255.254:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 4ms, Maximum = 48ms, Average = 20ms

C:\>ping 64.103.255.254

Pinging 64.103.255.254 with 32 bytes of data:
Reply from 64.103.255.254: bytes=32 time=12ms TTL=126
Reply from 64.103.255.254: bytes=32 time=1ms TTL=126
Reply from 64.103.255.254: bytes=32 time=4ms TTL=126
Reply from 64.103.255.254: bytes=32 time=18ms TTL=126

Ping statistics for 64.103.255.254:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli seconds:
        Minimum = 1ms, Maximum = 18ms, Average = 9ms
```

FTP not allowed

```
C:\>ftp 64.101.255.254
Trying to connect...64.101.255.254

%Error opening ftp://64.101.255.254/ (Timed out)
.

(Disconnecting from ftp server)

C:\>ftp 64.103.255.254
Trying to connect...64.103.255.254

%Error opening ftp://64.103.255.254/ (Timed out)
.

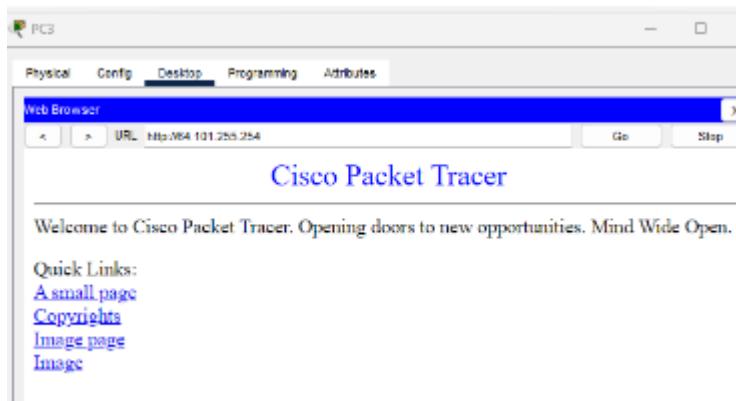
(Disconnecting from ftp server)
```

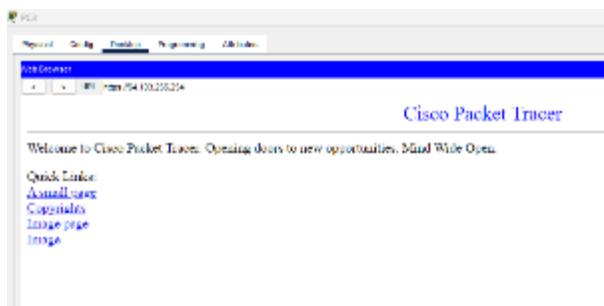
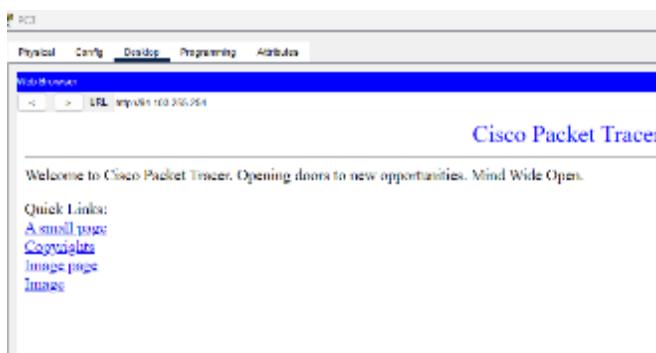
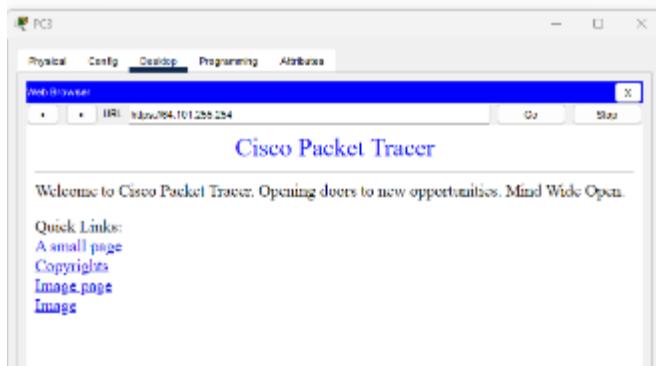
```
* listening on interface br0
* connected to source list o
10:09:49:11.111 host 64.101.255.254 nc 2201 [40] match[host]
10:09:49:11.111 host 64.101.255.254 nc 443 [40] match[host]
10:09:49:11.111 host 64.103.255.254 nc 2201 [40] match[host]
10:09:49:11.111 host 64.103.255.254 nc 443 [40] match[host]
10:09:49:11.111 host 64.101.255.254 nc ftp [40] match[host]
10:09:49:11.111 host 64.103.255.254 nc ftp [40] match[host]
10:09:49:11.111 host 64.101.255.254 nc 2201 [40] match[host]
10:09:49:11.111 host 64.103.255.254 nc 2201 [40] match[host]
10:09:49:11.111 host 64.101.255.254 nc 443 [40] match[host]
10:09:49:11.111 host 64.103.255.254 nc 443 [40] match[host]
```

10:11:49

PC3

Web OK





Ping not allowed

```
File Edit Desktop Reporting Windows
Cisco Packet Tracer PC Command Line 1.6
Tracing 64.101.255.254
Pinging 64.101.255.254 with 32 bytes of data:
Reply from 172.31.1.126: Destination host unreachable.

Ping statistics for 64.101.255.254:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
Tracing 64.103.255.254
Pinging 64.103.255.254 with 32 bytes of data:
Reply from 172.31.1.126: Destination host unreachable.

Ping statistics for 64.103.255.254:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>
```

```
RT1# show ip access-lists
Extended IP access list AOL
90 deny ip host 172.31.1.101 host 64.101.255.254 eq www {40 match(es)}
80 deny ip host 172.31.1.101 host 64.101.255.254 eq 443 {29 match(es)}
30 deny ip host 172.31.1.101 host 64.103.255.254 eq www {202 match(es)}
40 deny ip host 172.31.1.101 host 64.103.255.254 eq 443 {64 match(es)}
50 deny ip host 172.31.1.102 host 64.101.255.254 eq ftp {12 match(es)}
60 deny ip host 172.31.1.102 host 64.103.255.254 eq ftp {12 match(es)}
70 deny icmp host 172.31.1.103 host 64.101.255.254 {4 match(es)}
80 deny icmp host 172.31.1.103 host 64.103.255.254 {4 match(es)}
90 permit ip any any 1103 match(es)
```

RT1#

ftp allowed

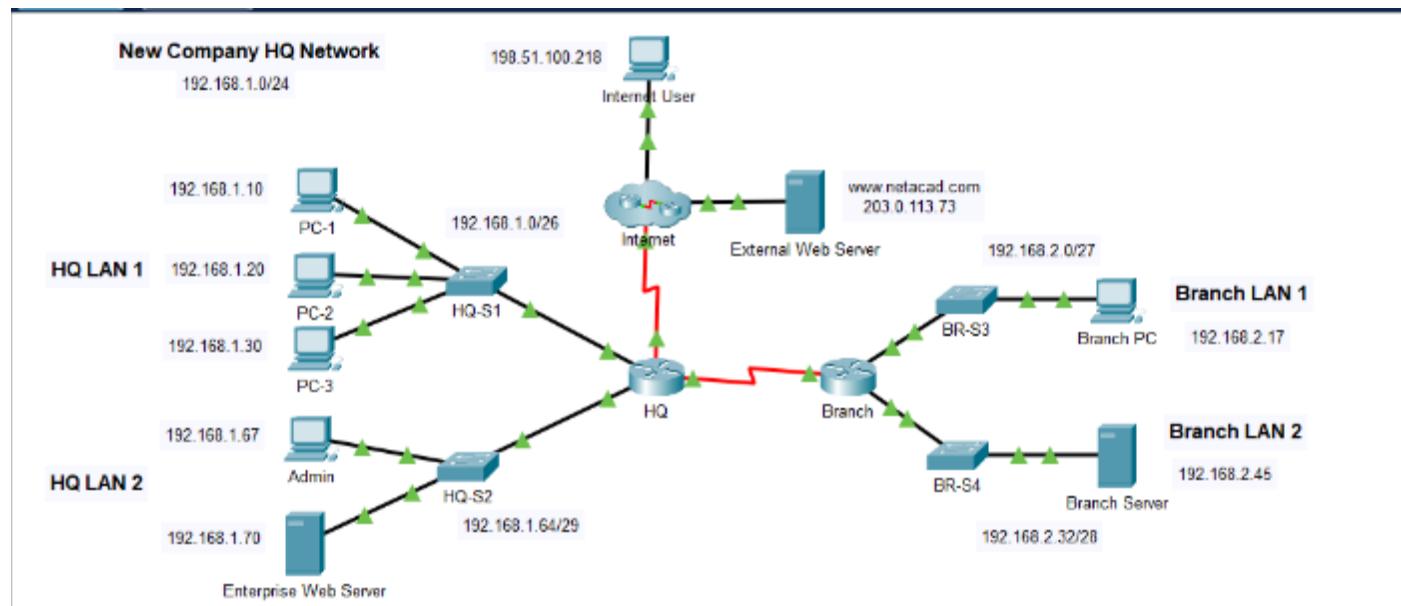
```
C:\>ftp 64.101.255.254
Trying to connect...64.101.255.254
Connected to 64.101.255.254
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>quit

221- Service closing control connection.
C:\>ftp 64.103.255.254
Trying to connect...64.103.255.254
Connected to 64.103.255.254
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>quit

221- Service closing control connection.
C:\>
```

3.2.14 Exercise 5.5.1 - Packet Tracer - IPv4 ACL Implementation Challenge

3.2.14.1 Topology



3.2.14.2 Addressing Table

Device	Interface	IP Address
HQ	G0/0/0	192.168.1.1/26
	G0/0/1	192.168.1.65/29
	S0/1/0	192.0.2.1/30
	S0/1/1	192.168.3.1/30
Branch	G0/0/0	192.168.2.1/27
	G0/0/1	192.168.2.33/28
	S0/1/1	192.168.3.2/30
PC-1	NIC	192.168.1.10/26
PC-2	NIC	192.168.1.20/26
PC-3	NIC	192.168.1.30/26

Admin	NIC	192.168.1.67/29
Enterprise Web Server	NIC	192.168.1.70/29
Branch PC	NIC	192.168.2.17/27
Branch Server	NIC	192.168.2.45/28
Internet User	NIC	198.51.100.218/24
External Web Server	NIC	203.0.113.73/24

3.2.14.3 Objectives

- Configure a router with standard named ACLs.
- Configure a router with extended named ACLs.
- Configure a router with extended ACLs to meet specific communication requirements.
- Configure an ACL to control access to network device terminal lines.
- Configure the appropriate router interfaces with ACLs in the appropriate direction.
- Verify the operation of the configured ACLs.

3.2.14.4 Background / Scenario

In this activity you will configure extended, standard named, and extended named ACLs to meet specified communication requirements.

3.2.14.5 Instructions

Step 1: Verify Connectivity in the New Company Network

First, test connectivity on the network as it is before configuring the ACLs. All hosts should be able to ping all other hosts.

Step 2: Configure Standard and Extended ACLs per Requirements.

Configure ACLs to meet the following requirements:

Important guidelines:

- Do **not** use explicit deny any statements at the end of your ACLs.
- Use shorthand (**host** and **any**) whenever possible.
- Write your ACL statements to address the requirements in the order that they are specified here.
- Place your ACLs in the most efficient location and direction.

ACL 1 Requirements

- Create ACL **101**.
- Explicitly block FTP access to the Enterprise Web Server from the internet.
- No ICMP traffic from the internet should be allowed to any hosts on HQ LAN 1
- Allow all other traffic.

```
HQ(config)#access-list 101 deny tcp any host 192.168.1.70 eq ftp
HQ(config)#access-list 101 deny icmp any 192.168.1.0 0.0.0.63
HQ(config)#access-list 101 permit ip any any
HQ(config)#interface Serial0/1/0
HQ(config-if)#ip access-group 101 in
```

ACL 2 Requirements

- Use ACL number **111**
- No hosts on HQ LAN 1 should be able to access the Branch Server.
- All other traffic should be permitted.

```
HQ(config)#access-list 111 deny ip any host 192.168.2.45
```

```
HQ(config)#access-list 111 permit ip any any  
HQ(config)#interface GigabitEthernet0/0/0  
HQ(config-if)#ip access-group 111 in
```

ACL 3: Requirements

- Create a named standard ACL. Use the name **vty_block**. The name of your ACL must match this name exactly.
- Only addresses from the HQ LAN 2 network should be able to access the VTY lines of the HQ router.

```
HQ(config)#ip access-list standard vty_block  
HQ(config-std-nacl)#permit 192.168.1.64 0.0.0.7  
HQ(config-std-nacl)#line vty 0 4  
HQ(config-line)#access-class vty_block in
```

ACL 4: Requirements

- Create a named extended ACL called **branch_to_hq**. The name of your ACL must match this name exactly.
- No hosts on either of the Branch LANs should be allowed to access HQ LAN 1. Use one access list statement for each of the Branch LANs.
- All other traffic should be allowed.

```
Branch(config)#ip access-list extended branch_to_hq  
Branch(config-ext-nacl)#deny ip 192.168.2.0 0.0.0.31 192.168.1.0 0.0.0.63  
Branch(config-ext-nacl)#deny ip 192.168.2.32 0.0.0.15 192.168.1.0 0.0.0.63  
Branch(config-ext-nacl)#permit ip any any  
Branch(config-ext-nacl)#interface Serial0/1/1  
Branch(config-if)#ip access-group branch_to_hq out
```

Step 3: Verify ACL Operation.

- a. Perform the following connectivity tests between devices in the topology. Note whether or not they are successful.
Note: Use the **show ip access-lists** command to verify ACL operation.

Use the **clear access list counters** command to reset the match counters.

```

HQ#show ip access-lists
Extended IP access list 101
    10 deny tcp any host 192.168.1.70 eq ftp
    20 deny icmp any 192.168.1.0 0.0.0.63
    30 permit ip any any
Standard IP access list vty_block
    10 permit 192.168.1.64 0.0.0.7
Extended IP access list 111
    10 deny ip any host 192.168.2.45
    20 permit ip any any

HQ#

```

```

Branch#show ip access-lists
Extended IP access list branch_to_hq
    10 deny ip 192.168.2.0 0.0.0.31 192.168.1.0 0.0.0.63
    20 deny ip 192.168.2.32 0.0.0.15 192.168.1.0 0.0.0.63
    30 permit ip any any

```

Branch#

Send a ping request from Branch PC to the Enterprise Web Server. Was it successful? Explain.

Answer - The ping was successful because it was permitted by the ACL

```

CISCO Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.70

Pinging 192.168.1.70 with 32 bytes of data:

Request timed out.
Reply from 192.168.1.70: bytes=32 time=22ms TTL=126
Reply from 192.168.1.70: bytes=32 time=3ms TTL=126
Reply from 192.168.1.70: bytes=32 time=19ms TTL=126

Ping statistics for 192.168.1.70:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 3ms, Maximum = 22ms, Average = 14ms
C:\>

```

Which ACL statement permitted or denied the ping between these two devices? List the access list name or number, the router on which it was applied, and the specific line that the traffic matched.

Answer - The last line in the branch_to_hq ACL on the Branch Router is permit ip any

```

Branch#show ip access-lists
Extended IP access list branch_to_hq
    10 deny ip 192.168.2.0 0.0.0.31 192.168.1.0 0.0.0.63
    20 deny ip 192.168.2.32 0.0.0.15 192.168.1.0 0.0.0.63
    30 permit ip any any (4 match(es))

Branch#

```

Attempt to ping from PC-1 on the HQ LAN 1 to the Branch Server. Was it successful? Explain.

Answer - The ping was not successful because the traffic was blocked by an access list.

```
C:\>ping 192.168.2.45
Pinging 192.168.2.45 with 32 bytes of data:
Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 192.168.2.45:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>
```

Which ACL statement permitted or denied the ping between these two devices?

Answer - Statement 10 in access list 111 on the HQ router denies all traffic to the branch server.

```
HQ#show ip access-lists
Extended IP access list 101
  10 deny tcp any host 192.168.1.70 eq ftp
  20 deny icmp any 192.168.1.0 0.0.0.63
  30 permit ip any any
Standard IP access list vty_block
  10 permit 192.168.1.64 0.0.0.7
Extended IP access list 111
  10 deny ip any host 192.168.2.45 (4 match(es))
  20 permit ip any any (4 match(es))

HQ#
```

Open a web browser on the External Server and attempt to bring up a web page stored on the Enterprise Web Server. Is it successful? Explain.

Answer - Yes, the External Server can access a web page on the Enterprise Web Server. HTTP traffic is not blocked to the Enterprise Web Server.



```

HQ#clear access-list counters
HQ#show ip access-lists
Extended IP access list 101
    10 deny tcp any host 192.168.1.70 eq ftp
    20 deny icmp any 192.168.1.0 0.0.0.63
    30 permit ip any any
Standard IP access list vty_block
    10 permit 192.168.1.64 0.0.0.7
Extended IP access list 111
    10 deny ip any host 192.168.2.45
    20 permit ip any any

HQ#show ip access-lists
Extended IP access list 101
    10 deny tcp any host 192.168.1.70 eq ftp
    20 deny icmp any 192.168.1.0 0.0.0.63
    30 permit ip any any (5 match(es))
Standard IP access list vty_block
    10 permit 192.168.1.64 0.0.0.7
Extended IP access list 111
    10 deny ip any host 192.168.2.45
    20 permit ip any any
...

```

Which ACL statement permitted or denied the ping between these two devices?

```

C:\>ping 192.168.2.45

Pinging 192.168.2.45 with 32 bytes of data:

Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 192.168.2.45:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

```

Answer - Line 20 in access list 101 on the HQ router permitted this traffic.

```

HQ#clear access-list counters
HQ#
HQ#
HQ#
HQ#show ip access-lists
Extended IP access list 101
    10 deny tcp any host 192.168.1.70 eq ftp
    20 deny icmp any 192.168.1.0 0.0.0.63
    30 permit ip any any
Standard IP access list vty_block
    10 permit 192.168.1.64 0.0.0.7
Extended IP access list 111
    10 deny ip any host 192.168.2.45 (4 match(es))
    20 permit ip any any

```

- b. Test connections to an internal server from the internet.

From the command line on the Internet User PC, attempt to make an FTP connection to the Branch Server. Is

the FTP connection successful?

Answer - Yes, the FTP connection from the internet User PC to the Branch Server is successful.

Which access list should be modified to prevent users from the Internet to make FTP connections to the Branch Server?

Answer - The access list 101 on the HQ router needs to be modified to deny this traffic.

Which statement(s) should be added to the access list to deny this traffic?

Answer - The statement “deny tcp any host 192.168.2.45 eq 21” or “deny tcp any host 192.168.2.45 range 20 21” needs to be added to the access list 101.

3.2.14.6 Scripts

ACL

ACL Name/Number	Purpose	Applied Interface	Direction
ACL 101	Block FTP and ICMP from the internet	HQ S0/1/0	Inbound
ACL 111	Block HQ LAN 1 from accessing Branch Server	HQ G0/0/1	Inbound
vty_block	Restrict VTY access to HQ LAN 2	VTY lines	Inbound
branch_to_hq	Block Branch LANs from accessing HQ LAN 1	Branch S0/1/1	Inbound

```
!!! Router HQ
enable
conf t
access-list 101 deny tcp any host 192.168.1.70 eq ftp
access-list 101 deny icmp any 192.168.1.0 0.0.0.63
access-list 101 permit ip any any
ip access-list standard vty_block
  permit 192.168.1.64 0.0.0.7
access-list 111 deny ip any host 192.168.2.45
access-list 111 permit ip any any
interface GigabitEthernet0/0/0
  ip access-group 111 in
interface Serial0/1/0
  ip access-group 101 in
line vty 0 4
  access-class vty_block in
end
```

```
!!! Router Branch
enable
conf t
```

```

ip access-list extended branch_to_hq
  deny ip 192.168.2.0 0.0.0.31 192.168.1.0 0.0.0.63
  deny ip 192.168.2.32 0.0.0.15 192.168.1.0 0.0.0.63
  permit ip any any
interface Serial0/1/1
  ip access-group branch_to_hq out
end

```

HQ

```

HQ>enable
HQ#!!! Router HQ
HQ#enable
HQ#conf t
Enter configuration commands, one per line. End with CNTL/Z.
HQ(config)#access-list 101 deny tcp any host 192.168.1.70 eq ftp
HQ(config)#access-list 101 deny icmp any 192.168.1.0 0.0.0.63
HQ(config)#access-list 101 permit ip any any
HQ(config)#ip access-list standard vty_block
HQ(config-std-nacl)# permit 192.168.1.64 0.0.0.7
HQ(config-std-nacl)#access-list 111 deny ip any host 192.168.2.45
HQ(config)#access-list 111 permit ip any any
HQ(config)#interface GigabitEthernet0/0/0
HQ(config-if)# ip access-group 111 in
HQ(config-if)#interface Serial0/1/0
HQ(config-if)# ip access-group 101 in
HQ(config-if)#line vty 0 4
HQ(config-line)# access-class vty_block in
HQ(config-line)#end
HQ#
%SYS-5-CONFIG_I: Configured from console by console

HQ#copy run
HQ#copy running-config s
HQ#copy running-config st
HQ#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
HQ#

```

Branch

```

Branch>!!! Router Branch
Branch>enable
Branch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Branch(config)#ip access-list extended branch_to_hq
Branch(config-ext-nacl)# deny ip 192.168.2.0 0.0.0.31 192.168.1.0 0.0.0.63
Branch(config-ext-nacl)# deny ip 192.168.2.32 0.0.0.15 192.168.1.0 0.0.0.63
Branch(config-ext-nacl)# permit ip any any
Branch(config-ext-nacl)#interface Serial0/1/1
Branch(config-if)# ip access-group branch_to_hq out
Branch(config-if)#end
Branch#
%SYS-5-CONFIG_I: Configured from console by console

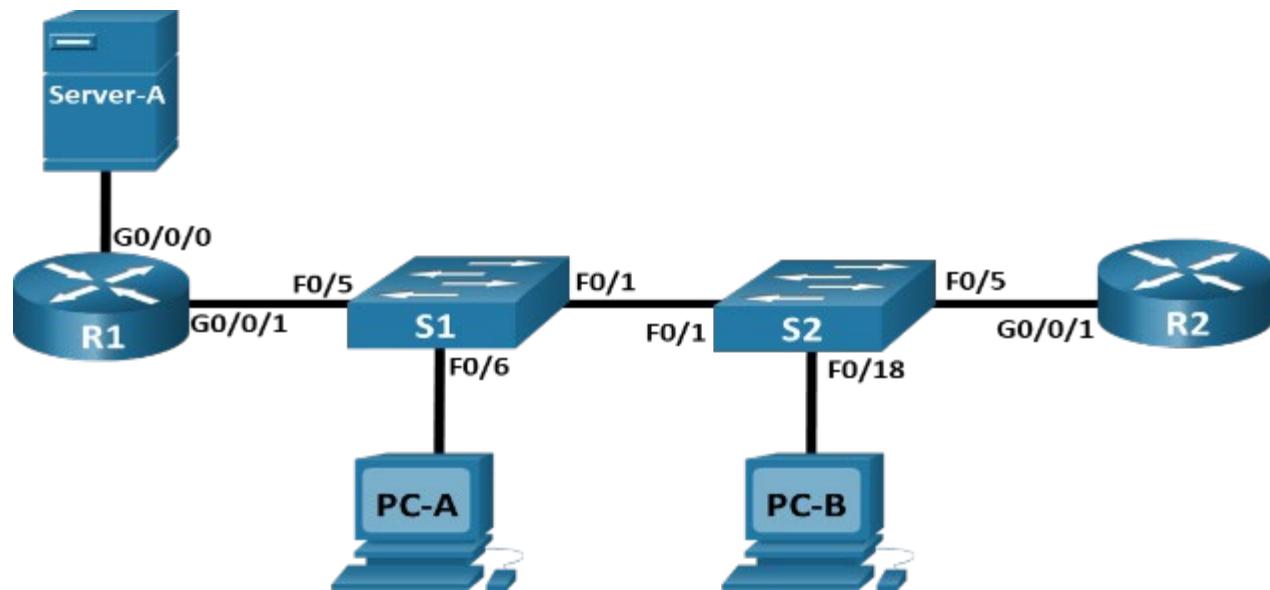
Branch#copy ruy
Branch#copy ru
Branch#copy running-config st
Branch#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Branch#

```

3.2.15 Exercise 5.5.2 - Packet Tracer - Configure and Verify Extended IPv4 ACLs

- Physical Mode

3.2.15.1 Topology



3.2.15.2 Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0/1	N/A	N/A	N/A
	G0/0/1.20	10.20.0.1	255.255.255.0	
	G0/0/1.30	10.30.0.1	255.255.255.0	
	G0/0/1.40	10.40.0.1	255.255.255.0	
	G0/0/1.1000	N/A	N/A	
	G0/0/0	172.16.1.1	255.255.255.0	
R2	G0/0/1	10.20.0.4	255.255.255.0	N/A
S1	VLAN 20	10.20.0.2	255.255.255.0	10.20.0.1
S2	VLAN 20	10.20.0.3	255.255.255.0	10.20.0.1
PC-A	NIC	10.30.0.10	255.255.255.0	10.30.0.1
PC-B	NIC	10.40.0.10	255.255.255.0	10.40.0.1
Server-A	NIC	172.16.1.2	255.255.255.0	172.16.1.1

3.2.15.3 VLAN Table

VLAN	Name	Interface Assigned
20	Management	S2: F0/5
30	Operations	S1: F0/6

40	Sales	S2: F0/18
999	ParkingLot	S1: F0/2-4, F0/7-24, G0/1-2 S2: F0/2-4, F0/6-17, F0/19-24, G0/1-2
1000	Native	N/A

3.2.15.4 Objectives

Part 1: Build the Network and Configure Basic Device Settings

Part 2: Configure VLANs on the Switches

Part 3: Configure Trunking

Part 4: Configure Routing

Part 5: Configure Remote Access

Part 6: Verify Connectivity

Part 7: Configure and Verify Extended Access Control Lists

3.2.15.5 Background / Scenario

In this Packet Tracer Physical Mode (PTPM) activity, you have been tasked with configuring access control lists (ACLs) on a small company's network. ACLs are one of the simplest and most direct means of controlling Layer 3 traffic. R1 will be hosting an internet connection and sharing the default route information to R2. After initial configuration is complete, the company has some specific traffic security requirements that you will be responsible for implementing.

Note: There are over 100 items scored in this activity. Therefore, Packet Tracer will display the number of items currently correct instead of the percentage score.

3.2.15.6 Instructions

Part 1: Build the Network and Configure Basic Device Settings

Step 1: Cable the network as shown in the topology.

- a. Cable and power on the devices as shown in the topology diagram. Use a console cable to connect a **PC** to each switch or router as you configure them. To access a switch or router, you must connect a console cable between the PCs and the device you wish to configure. We recommend connecting **PC-A** to **R1** and **PC-B** to **R2**.
- b. Then, when configuring the switches, connect **PC-A** to **S1** and **PC-B** to **S2**. After you have connected the console cable, click the **PC > Desktop tab > Terminal**, and then click **OK**, to access the command line.

When changing a console cable to a new device, such as between a router and a switch, it is easier to click the end of the console cable and drag it back to the Cable Pegboard than it is to try to connect the cable directly to another device. After attaching a console cable to a different device, you must close and reopen the **Terminal** window to establish a new connection.

Step 2: Configure basic settings for each router.

- a. Assign a device name to the router.
- b. Disable DNS lookup to prevent the router from attempting to translate incorrectly entered commands as though they were host names.
- c. Assign **class** as the privileged EXEC encrypted password.
- d. Assign **cisco** as the console password and enable login.
- e. Assign **cisco** as the vty password. You will enable login later in this activity.
- f. Encrypt the plaintext passwords.
- g. Create a banner that warns anyone accessing the device that unauthorized access is prohibited.
- h. Save the running configuration to the startup configuration file.

Step 3: Configure basic settings for each switch.

- a. Assign a device name to the switch.
- b. Disable DNS lookup to prevent the router from attempting to translate incorrectly entered commands as though they were host names.
- c. Assign **class** as the privileged EXEC encrypted password.
- d. Assign **cisco** as the console password and enable login.
- e. Assign **cisco** as the vty password. You will enable login later in this activity.
- f. Encrypt the plaintext passwords.
- g. Create a banner that warns anyone accessing the device that unauthorized access is prohibited.
- h. Save the running configuration to the startup configuration file.

Part 2: Configure VLANs on the Switches

Step 1: Create VLANs on both switches.

- a. Create and name the required VLANs on each switch from the VLAN table.
- b. Configure the management interface and default gateway on each switch using the IP address information in the Addressing Table.
- c. Assign all unused ports on the switch to the Parking Lot VLAN. Configure them for static access mode, and administratively deactivate them.

Note: The **interface range** command helps to accomplish this task with as few commands as necessary.

Step 2: Assign VLANs to the correct switch interfaces.

- a. Assign used ports to the appropriate VLAN (specified in the VLAN table) and configure them for static access mode.
- b. Issue the **show vlan brief** command and verify that the VLANs are assigned to the correct interfaces.

Part 3: Configure Trunking

Step 1: Manually configure trunk interface F0/1.

- a. Change the switchport mode on interface F0/1 to force trunking. Make sure to do this on both switches.
- b. As a part of the trunk configuration, set the native vlan to 1000 on both switches. You may see error messages temporarily while the two interfaces are configured for different Native VLANs.
- c. As another part of trunk configuration, specify that VLANs 10, 20, 30, and 1000 are allowed to cross the trunk.
- d. Issue the **show interfaces trunk** command to verify trunking ports, the Native VLAN and allowed VLANs across the trunk.

Step 2: Manually configure S1's trunk interface F0/5.

- a. Configure S1's interface F0/5 with the same trunk parameters as F0/1. This is the trunk to R1.
- b. Save the running configuration to the startup configuration file.

Part 4: Configure Routing

Step 1: Configure Inter-VLAN Routing on R1.

- a. Activate interface G0/0/1 on the router.
- b. Configure sub-interfaces for each VLAN as specified in the Addressing Table. All sub-interfaces use 802.1Q encapsulation. Ensure the sub-interface for the Native VLAN does not have an IP address assigned. Include a description for each sub-interface.
- c. Configure interface G0/0/1 on R1 with addressing from the Addressing Table.
- d. Use the **show ip interface brief** command to verify that the sub-interfaces are operational.

Step 2: Configure the R2 interface g0/0/1 using the Addressing table and a default route with the next hop 10.20.0.1

Part 5: Configure Remote Access

Step 1: Configure all network devices for basic SSH support.

- a. Create a local user with the username **SSHadmin** and **\$cisco123!** as the encrypted password.
- b. Use **ccna-lab.com** as the domain name.
- c. Generate crypto keys using a 1024-bit modulus.
- d. Configure the first five vty lines on each device to support SSH connections only and to authenticate to the local user database.

Part 6: Verify Connectivity

Step 1: Configure PC hosts.

Refer to the Addressing Table for PC host address information.

Step 2: Complete the following tests. All should be successful.

Note: If you click **Check Results**, you will see that the five highlighted **Connectivity Tests** show as incorrect. This is because you have not implemented ACLs yet. After ACLs are implemented, these five highlighted **Connectivity Tests** should successfully fail.

From	Protocol	Destination	Result
PC-A	Ping	10.40.0.10	Success
PC-A	Ping	10.20.0.1	Success
PC-B	Ping	10.30.0.10	Success
PC-B	Ping	10.20.0.1	Success
PC-B	Ping	172.16.1.1	Success
PC-B	HTTPS	172.16.1.2	Success
PC-A	HTTPS	172.16.1.2	Success
PC-B	SSH	10.20.0.4	Success
PC-B	SSH	172.16.1.1	Success

Part 7: Configure and Verify Extended Access Control Lists

When basic connectivity is verified, the company requires the following security policies to be implemented:

Policy 1: The Sales Network is not allowed to SSH to the Management Network (but other SSH is allowed).

Policy 2: The Sales Network is not allowed to access server-A using any web protocol (HTTP/HTTPS). All other web traffic is allowed.

Policy 3: The Sales Network is not allowed to send ICMP echo requests to the Operations or Management Networks. ICMP echo requests to other destinations are allowed.

Policy 4: The Operations network is not allowed to send ICMP echo requests to the Sales Network. ICMP echo requests to other destinations are allowed.

Step 1: Develop and apply extended access lists that will meet the security policy statements.

Step 2: Verify that security policies are being enforced by the deployed access lists.

Run the following tests. The expected results are shown in the table:

Note: Click **Check Results** to force Packet Tracer to run all the **Connectivity Tests** again.

From	Protocol	Destination	Result
PC-A	Ping	10.40.0.10	Fail
PC-A	Ping	10.20.0.1	Success
PC-B	Ping	10.30.0.10	Fail
PC-B	Ping	10.20.0.1	Fail
PC-B	Ping	172.16.1.1	Success
PC-B	HTTPS	172.16.1.2	Fail

From	Protocol	Destination	Result
PC-A	HTTPS	172.16.1.2	Success
PC-B	SSH	10.20.0.4	Fail
PC-B	SSH	172.16.1.1	Success

3.2.15.7 Scripts

!!! S1

```
enable
configure terminal
```

```
! Step 2: Configure basic settings
hostname S1
no ip domain-lookup
enable secret class
line console 0
password cisco
login
line vty 0 4
password cisco
login
exit
service password-encryption
banner motd #Unauthorized access is prohibited.#
```

! Step 2: Configure VLANs

```
vlan 20
name Management
exit
vlan 30
name Operations
exit
vlan 40
name Sales
exit
vlan 999
name ParkingLot
exit
vlan 1000
name Native
exit
```

! Assign VLANs to interfaces

```
interface range f0/2-4, f0/7-24, g0/1-2
switchport mode access
switchport access vlan 999
shutdown
exit
```

```
interface f0/6
switchport mode access
switchport access vlan 30
description Operations VLAN
exit
interface vlan 20
ip address 10.20.0.2 255.255.255.0
no shutdown
exit
ip default-gateway 10.20.0.1
```

! Step 3: Configure Trunking

```
interface f0/1
switchport mode trunk
switchport trunk native vlan 1000
switchport trunk allowed vlan 20,30,40,1000
exit
interface f0/5
switchport mode trunk
switchport trunk native vlan 1000
switchport trunk allowed vlan 20,30,40,1000
exit
```

! Step 5: Configure Remote Access

```
username SSHadmin secret $cisco123!
ip domain-name ccna-lab.com
crypto key generate rsa general-keys modulus 1024
line vty 0 15
transport input ssh
login local
exit
end
! Save configuration
write memory
```

!!!!!!!

!!! s2

!!!!!!

```
enable
configure terminal
```

! Step 2: Configure basic settings

```
hostname S2
no ip domain-lookup
enable secret class
line console 0
password cisco
login
exit
service password-encryption
banner motd #Unauthorized access is prohibited.#
```

```
! Step 2: Configure VLANs
```

```
vlan 20
  name Management
  exit
vlan 30
  name Operations
  exit
vlan 40
  name Sales
  exit
vlan 999
  name ParkingLot
  exit
vlan 1000
  name Native
  exit
```

```
! Assign VLANs to interfaces
```

```
interface range f0/2-4, f0/6-17, f0/19-24, g0/1-2
  switchport mode access
  switchport access vlan 999
  shutdown
  exit
interface f0/5
  switchport mode access
  switchport access vlan 20
  description Management VLAN
  exit
interface f0/18
  switchport mode access
  switchport access vlan 40
  description Sales VLAN
  exit
interface vlan 20
  ip address 10.20.0.3 255.255.255.0
  no shutdown
  exit
  ip default-gateway 10.20.0.1
```

```
! Step 3: Configure Trunking
```

```
interface f0/1
  switchport mode trunk
  switchport trunk native vlan 1000
  switchport trunk allowed vlan 20,30, 40,1000
  exit
```

```
! Step 5: Configure Remote Access
```

```
username SSHadmin secret $cisco123!
ip domain-name ccna-lab.com
crypto key generate rsa general-keys modulus 1024
line vty 0 15
```

```
transport input ssh
login local
password cisco
end
! Save configuration
write memory
```

!!!!!!
!!! R1
!!!!!!

```
enable
configure terminal
```

```
! Step 2: Configure basic settings
hostname R1
no ip domain-lookup
enable secret class
line console 0
password cisco
login
line vty 0 15
password cisco
login
exit
service password-encryption
banner motd #Unauthorized access is prohibited.#
```

```
! Step 4: Configure Inter-VLAN Routing
interface g0/0/1
no shutdown
exit
interface g0/0/1.20
encapsulation dot1Q 20
ip address 10.20.0.1 255.255.255.0
description Management VLAN
exit
interface g0/0/1.30
encapsulation dot1Q 30
ip address 10.30.0.1 255.255.255.0
description Operations VLAN
exit
interface g0/0/1.40
encapsulation dot1Q 40
ip address 10.40.0.1 255.255.255.0
description Sales VLAN
exit
interface g0/0/1.1000
encapsulation dot1Q 1000 native
description Native VLAN
```

```
exit
interface g0/0/0
ip address 172.16.1.1 255.255.255.0
no shutdown
exit

! Step 5: Configure Remote Access
username SSHadmin secret $cisco123!
ip domain-name ccna-lab.com
crypto key generate rsa general-keys modulus 1024
line vty 0 15
transport input ssh
login local
exit

! Policy 1: Block Sales Network from SSH to Management Network
access-list 101 deny tcp 10.40.0.0 0.0.0.255 10.20.0.0 0.0.0.255 eq 22
access-list 101 permit tcp any any eq 22

! Policy 2: Block Sales Network from accessing Server-A via HTTP/HTTPS
access-list 102 deny tcp 10.40.0.0 0.0.0.255 host 172.16.1.2 eq 80
access-list 102 deny tcp 10.40.0.0 0.0.0.255 host 172.16.1.2 eq 443
access-list 102 permit tcp any any eq 80
access-list 102 permit tcp any any eq 443

! Policy 3: Block Sales Network from sending ICMP to Operations/Management Networks
access-list 103 deny icmp 10.40.0.0 0.0.0.255 10.30.0.0 0.0.0.255 echo
access-list 103 deny icmp 10.40.0.0 0.0.0.255 10.20.0.0 0.0.0.255 echo
access-list 103 permit icmp any any

! Policy 4: Block Operations Network from sending ICMP to Sales Network
access-list 104 deny icmp 10.30.0.0 0.0.0.255 10.40.0.0 0.0.0.255 echo
access-list 104 permit icmp any any

! Apply ACLs to interfaces
interface g0/0/1.40
ip access-group 101 in
ip access-group 102 in
ip access-group 103 in
exit
interface g0/0/1.30
ip access-group 104 in
exit
end
! Save configuration
write memory
```

!!!!!!
!!! R2
!!!!!!!

```

enable
configure terminal

! Step 2: Configure basic settings
hostname R2
no ip domain-lookup
enable secret class
line console 0
password cisco
login
line vty 0 15
password cisco
login
exit
service password-encryption
banner motd #Unauthorized access is prohibited.#

! Step 4: Configure Interface and Default Route
interface g0/0/1
ip address 10.20.0.4 255.255.255.0
no shutdown
exit
ip route 0.0.0.0 0.0.0.0 10.20.0.1

! Step 5: Configure Remote Access
username SSHadmin secret $cisco123!
ip domain-name ccna-lab.com
crypto key generate rsa general-keys modulus 1024

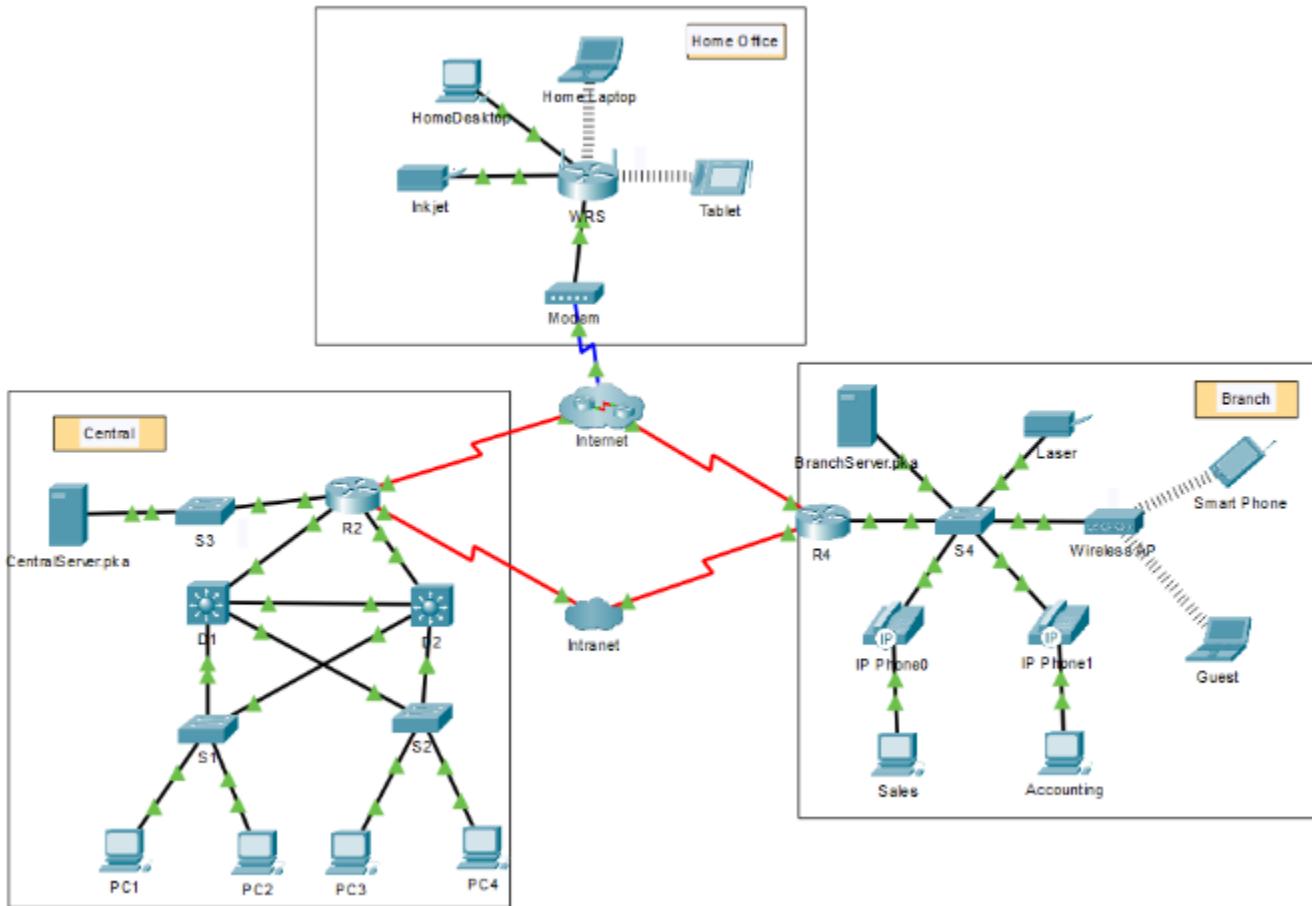
line vty 0 15
transport input ssh
login local
exit

! Save configuration
write memory

```

3.2.16 **Exercise 6.2.7 - Packet Tracer - Investigate NAT Operations**

3.2.16.1 Topology



3.2.16.2 Addressing Table

The following table provides addressing for networking device interfaces only.

Device	Interface	IP Address and Prefix
R2	G0/0	10.255.255.245/30
	G0/1	10.255.255.249/30
	G0/2	10.10.10.1/24
	S0/0/0	64.100.100.2/27
	S0/0/1.1	64.100.200.2/30
R4	G0/0	172.16.0.1/24
	S0/0/0	64.100.150.1/30
	S0/0/1.1	64.100.200.1/30
WRS	LAN	192.168.0.1/24
	Internet	64.104.223.2/30

3.2.16.3 Objectives

Part 1: Investigate NAT Operation Across the Intranet

Part 2: Investigate NAT Operation Across the Internet

Part 3: Conduct Further Investigations

Scenario

As a frame travels across a network, the MAC addresses may change. IP addresses can also change when a packet is forwarded by a device configured with NAT. In this activity, we will investigate what happens to IP addresses during the NAT process.

3.2.16.4 Instructions

Part 1: Investigate NAT Operation Across the Intranet

Step 1: Wait for the network to converge.

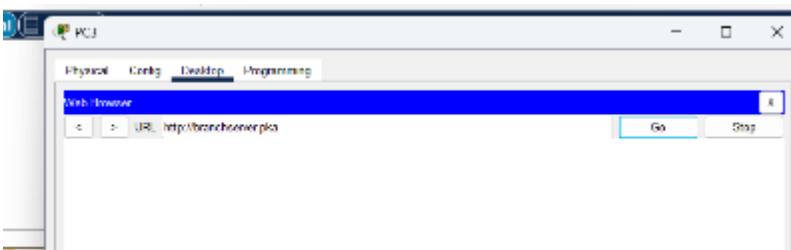
It might take a few minutes for everything in the network to converge. You can speed the process up by clicking Fast Forward Time.

Step 2: Generate an HTTP request from any PC in the Central domain.

- Switch to **Simulation** mode and edit the filters to show only HTTP requests.



- Open the Web Browser of any PC in the **Central** domain and type the URL **http://branchserver.pka** and click **Go**. Minimize the browser window.



- c. Click **Capture / Forward** until the PDU is over **D1** or **D2**. Click on the most recent PDU in the Event List. Record the source and destination IP addresses.

To what devices do those addresses belong? **Answer** for PC3 Source 10.3.0.4 and Destination R4 64.100.200.1

- d. Click **Capture / Forward** until the PDU is over **R2**. Record the source and destination IP addresses in the outbound packet.

To what devices do those addresses belong? **Answer** Source 64.100.100.3 and Destination R4 64.100.200.1
The first address is not assigned to an interface.

When we do next step e and see the address came from the NAT pool **R2Pool**.

R4 is the second address.

- e. Login to R2 from the CLI using the password **class** to enter privileged EXEC and issue the following command:

```
R2# show run | include pool
ip nat pool R2Pool 64.100.100.3 64.100.100.31 netmask 255.255.255.224 ip nat
inside source list 1 pool R2Pool
```

```
R2#show run | include pool
ip nat pool R2Pool 64.100.100.3 64.100.100.31 netmask 255.255.255.224
ip nat inside source list 1 pool R2Pool
R2#
```

The address came from the NAT pool **R2Pool**.

Note – Every IP coming out of R2 will take an ip address in the R2Pool ie in the range of 64.100.100.3 to 64.100.100.31. Not it will not use the private ip from PC3 10.1.0.4

- f. Click **Capture / Forward** until the PDU is over **R4**. Record the source and destination IP addresses in the **outbound** packet.

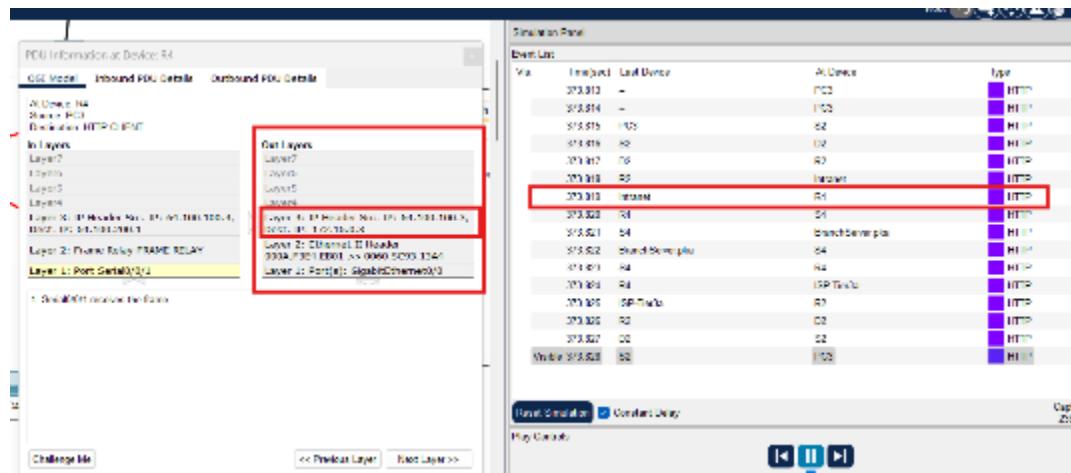
To what devices do those addresses belong? **Answer** – on R4 Source IP is 64.100.100.3 and destination IP is 172.16.0.3.

Source = The first address is from R2Pool on R2.

Destination = Branchserver.pka is the second address.the ip is not in the table but is part of network in R4 G/0/0

R4	G0/0	172.16.0.1/24
		64.100.100.1/30

Note this in **outboud** layers



- g. Click **Capture / Forward** until the PDU is over **Branchserver.pka**. Record the source and destination TCP port addresses in the outbound segment.

Answer - source port 80, destination port 1025

PDU Information at Device (BranchServer) pane:

- CGI Model: Inbound PDU Details
- Outbound PDU Details
- At Device: BranchServer (port 80 - PC)
- Destination: H111.CB1B
- In Layers:
 - Layer 7: HTTP
 - Layer 6: Headers
 - Layer 5: Layer 4: TCP SYN from: 10.10.10.25, TCP (Port: 1025)
 - Layer 4: IP Header Src. IP: 64.100.100.3
 - Dest. IP: 172.16.0.3
 - Layer 3: Ethernet II Header: 000A.F3C4.EB01 to 0060.0CB0.13A4
 - Layer 2: Port: FastEthernet0/0
 - Layer 1: Port: FastEthernet0/0
- Challenge Me...

Simulation Panel:

Idx	Type	Last Device	At Device	Type
373.013		PC3	PC3	H1 TCP
373.014	-	PC3	PC3	H1 TCP
373.015	PC3	R2	R2	H1 TCP
373.016	R2	R2	R2	H1 TCP
373.017	R2	R2	R2	H1 TCP
373.018	R2	Intranet	R4	H1 TCP
373.019	Intranet	R4	R4	H1 TCP
373.020	R4	R4	R4	H1 TCP
373.021	R4	BranchServer (port 80 - PC)	R4	H1 TCP
373.022	BranchServer (port 80 - PC)	R4	R4	H1 TCP
373.023	R4	R4	R4	H1 TCP
373.024	R4	ISP-Router	R2	H1 TCP
373.025	R2	R2	R2	H1 TCP
373.026	R2	R2	R2	H1 TCP
373.027	R2	R2	R2	H1 TCP
373.028	R2	PC3	PC3	H1 TCP

Buttons: Constant Delay

- h. On both **R2** and **R4**, run the following command and match the IP addresses and ports recorded above to the correct line of output:

R2# show ip nat translations

R4# **show ip nat translations**

```
R2>enable
Password:
R2#show ip nat translations
Pro Inside global    Inside local        Outside local        Outside global
tcp 64.100.100.2:25  10.10.10.2:25      ---                ---
tcp 64.100.100.2:443 10.10.10.2:443      ---                ---
tcp 64.100.100.2:0:0 10.10.10.2:0:0       ---                ---
tcp 64.100.100.3:1025 10.3.0.4:1025      64.100.200.1:80   64.100.200.1:80
tcp 64.100.100.3:1026 10.3.0.4:1026      64.100.200.1:80   64.100.200.1:80
tcp 64.100.100.3:1027 10.3.0.4:1027      64.100.200.1:80   64.100.200.1:80
tcp 64.100.100.3:1028 10.3.0.4:1028      64.100.200.1:80   64.100.200.1:80
R2#
```

```
R4#show ip nat translations
Pro Inside global    Inside local        Outside local        Outside global
tcp 64.100.150.1:80  172.16.0.3:80      ---                ---
tcp 64.100.200.1:25  172.16.0.3:25      ---                ---
tcp 64.100.200.1:443 172.16.0.3:443      ---                ---
tcp 64.100.200.1:80  172.16.0.3:80      ---                ---
tcp 64.100.200.1:80  172.16.0.3:80      64.100.100.3:1025 64.100.100.3:1025
tcp 64.100.200.1:80  172.16.0.3:80      64.100.100.3:1026 64.100.100.3:1026
tcp 64.100.200.1:80  172.16.0.3:80      64.100.100.3:1027 64.100.100.3:1027
tcp 64.100.200.1:80  172.16.0.3:80      64.100.100.3:1028 64.100.100.3:1028
R4#
```

What do the inside local IP addresses have in common? **Answer** - They are reserved for private use.

R2 all starts with 10. And for R4 all start with 172.16

Did any private addresses cross the intranet? **Answer** -No

- i. Click the Reset Simulation button and remain in Simulation Model.



Part 2: Investigate NAT Operation Across the Internet

Step 1: Generate an HTTP request from any computer in the home office.

- a. Open the Web Browser of any PC in the **Home Office** domain and type the URL <http://centralserver.pka> and click **Go**.



- b. Click Capture / Forward until the PDU is over WRS. Record the inbound source and destination IP addresses and the outbound source and destination addresses.

No.	Timestamp	Last Device	Ac Device	Type
15.032	-	HomeDesktop	HomeDesktop	HTTP
15.033	-	HomeDesktop	HomeDesktop	HTTP
15.034	15:05:44.000000000 HomeDesktop	WRS	WRS	HTTP
15.035	15:05:44.000000000 WRS	Modem	Modem	HTTP
15.036	15:05:44.000000000 Modem	Cable Provider	Cable Provider	HTTP
15.037	15:05:44.000000000 Cable Provider	R2	R2	HTTP
15.038	15:05:44.000000000 R2	S3	S3	HTTP

To what devices do those addresses belong?

Answer -

Inbound: Source 192.168.0.101 (computer - homedesktop) and Destination 64.100.100.2. (R2)

Outbound: source 64.104.223.2 (WRS) and Destination 64.100.100.2. (R2)

- c. Click **Capture / Forward** until the PDU is over R2. Record the source and destination IP addresses in the outbound packet.

No.	Timestamp	Last Device	Ac Device	Type
15.032	-	HomeDesktop	HomeDesktop	HTTP
15.033	-	HomeDesktop	HomeDesktop	HTTP
15.034	15:05:44.000000000 HomeDesktop	WRS	WRS	HTTP
15.035	15:05:44.000000000 WRS	Modem	Modem	HTTP
15.036	15:05:44.000000000 Modem	Cable Provider	Cable Provider	HTTP
15.037	15:05:44.000000000 Cable Provider	R2	R2	HTTP
15.038	15:05:44.000000000 R2	S3	S3	HTTP

To what devices do those addresses belong?

Answer -

Source 64.104.223.2 WRS and Destination 10.10.10.2 (web server centralserver.pka.)

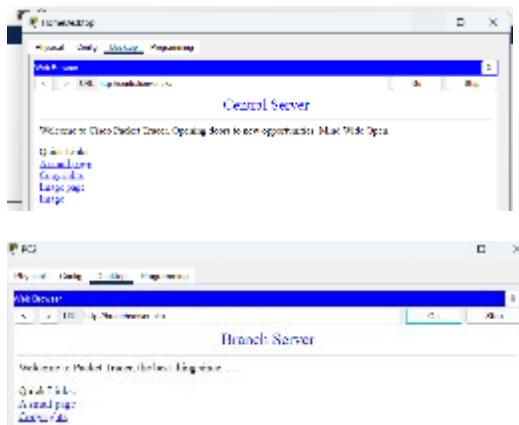
- d. On R2, run the following command and match the IP addresses and ports recorded above to the correct line of output:

R2# show ip nat translations

```
R2#show ip nat translations
Pro Inside global     Inside local        Outside local      Outside global
tcp 64.100.100.2:25   10.10.10.2:25      ---              ---
tcp 64.100.100.2:443  10.10.10.2:443      ---              ---
tcp 64.100.100.2:80   10.10.10.2:80      ---              ---
tcp 64.100.100.2:80   10.10.10.2:80      64.104.223.2:1025 64.104.223.2:1025
tcp 64.100.100.3:1025 10.3.0.4:1025      64.100.200.1:80  64.100.200.1:80
tcp 64.100.100.3:1026 10.3.0.4:1026      64.100.200.1:80  64.100.200.1:80
tcp 64.100.100.3:1027 10.3.0.4:1027      64.100.200.1:80  64.100.200.1:80
tcp 64.100.100.3:1028 10.3.0.4:1028      64.100.200.1:80  64.100.200.1:80
```

e. Return to Realtime mode.

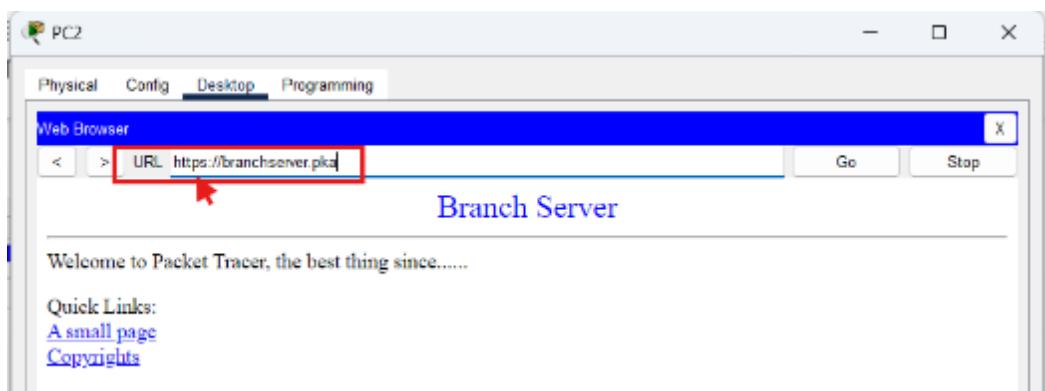
Did all of the web pages appear in the browsers? **Answer - Yes**



Part 3: Conduct Further Investigations

Experiment with more packets, both HTTP and HTTPS and answer the following questions.

HTTPS - <https://branchserver.pka>



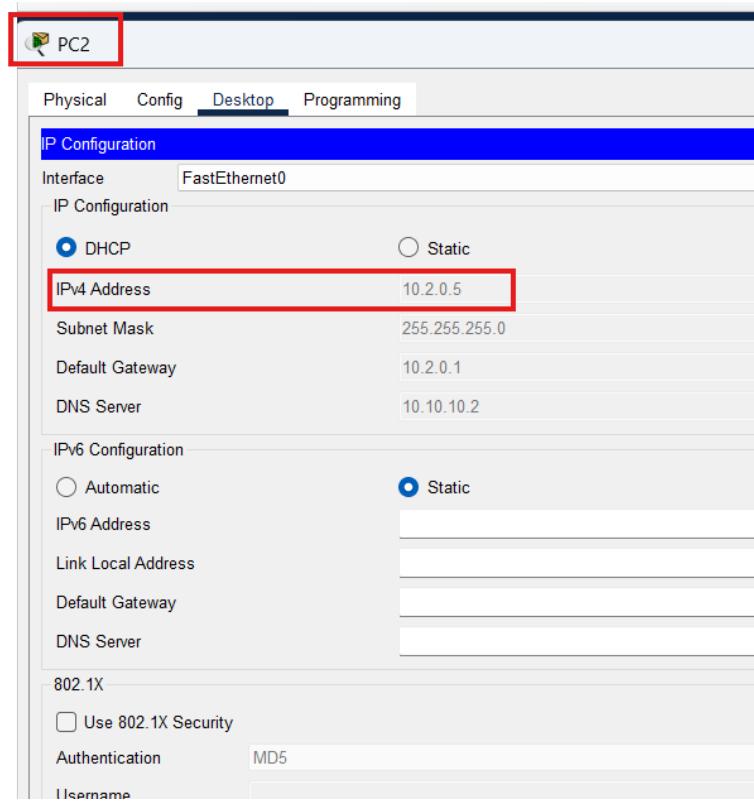
Port 443 is primarily used for secure web traffic. It is the default port for HTTPS (Hypertext Transfer Protocol Secure) communication

```

Password:
R2#show ip nat tr
R2#show ip nat translations
Pro Inside global Inside local Outside local Outside global
udp 64.100.100.2:1027 10.10.10.2:1027 64.100.8.8:53 64.100.8.8:53
tcp 64.100.100.2:25 10.10.10.2:25 ---- ----
tcp 64.100.100.2:443 10.10.10.2:443 ---- ----
tcp 64.100.100.2:80 10.10.10.2:80 ---- ----
tcp 64.100.100.2:800 10.10.10.2:800 64.104.223.2:1025 64.104.223.2:1025
tcp 64.100.100.3:1025 10.3.0.4:1025 64.100.200.1:80 64.100.200.1:80
tcp 64.100.100.3:1027 10.3.0.4:1027 64.100.200.1:80 64.100.200.1:80
tcp 64.100.100.4:1025 10.2.0.5:1025 64.100.200.1:80 64.100.200.1:80
tcp 64.100.100.4:1026 10.2.0.5:1026 64.100.200.1:443 64.100.200.1:443
R2#

```

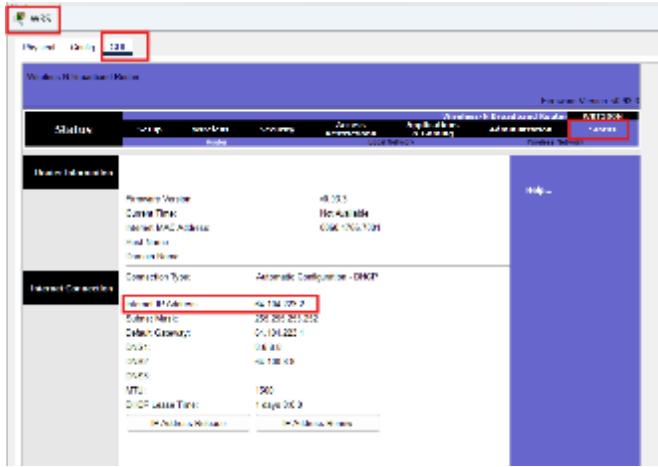
PC2 is 10.2.0.5



Do the NAT translation tables grow?

Answer - Yes. There are additional entries as new conversations are started.

Does WRS have a NAT pool of addresses? **Answer** - No, it uses the same IP address for all devices. IP address used is 64.104.223.3 for home office to go outside.



Is this how the computers in the classroom connect to the internet? **Answer** - It depends on the campus infrastructure. An easy way to check is using something like <https://www.whatismyip.org> to determine if all machines in the classroom are using the same address.

Why does NAT use four columns of addresses and ports? **Answer** - The columns list the inside global, inside local, outside local, and outside global addresses.

1. *Inside Global Address: The IP address assigned to the device when it communicates with external networks. This address is routable on the global internet and is used to represent the inside device to the outside world.*
2. *Inside Local Address: The IP address assigned to a device on the internal (local) network. This address is usually not routable on the global internet.*
3. *Outside Local Address: The IP address of an external device as perceived from the local network. This address may be the same as the outside global address or different depending on the NAT configuration.*
4. *Outside Global Address: The IP address assigned to an external device by the external network. This address is routable on the global internet and is used to represent the outside device to the inside network.*
 - a) *Private IP Address: Internal (private) IP addresses of devices within the local network.*
 - b) *Private Port Number: Internal port numbers associated with the private IP addresses.*
 - c) *Public IP Address: External (public) IP addresses that are used when the private IP addresses communicate with external networks, like the internet.*
 - d) *Public Port Number: External port numbers associated with the public IP addresses.*

Where are the networks are inside global and inside local? **Answer** –
The **inside local** addresses are on the LANs within each domain.

The terms "inside global" and "inside local" are used to describe different perspectives of IP addresses when NAT is configured:

- Inside Local: This is the IP address assigned to a device on the internal (local) network. It is typically a private IP address that is not routable on the global internet.
- Inside Global: This is the IP address assigned to the internal device when it communicates with external networks. It is routable on the global internet and represents the inside device to the outside world.

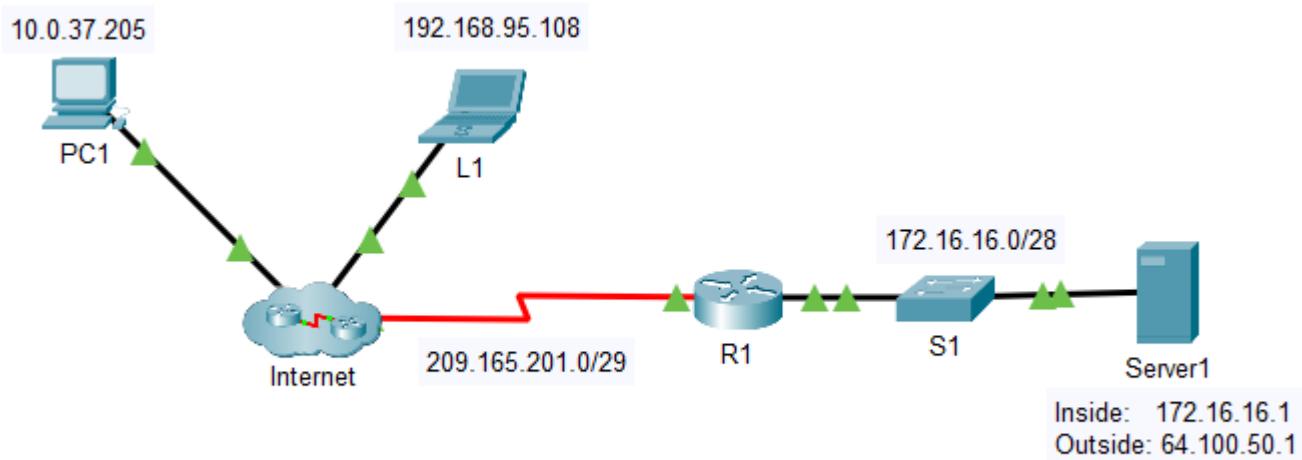
On which devices are NAT services operating? What do they have in common?

Answer WRS, R2, and R4.

They all connect internal LANs to outside networks that require routable IP addresses.

3.2.17 Exercise 6.4.5 - Packet Tracer - Configure Static NAT

3.2.17.1 Topology



3.2.17.2 Objectives

Part 1: Test Access without NAT

Part 2: Configure Static NAT Part

3: Test Access with NAT

3.2.17.3 Scenario

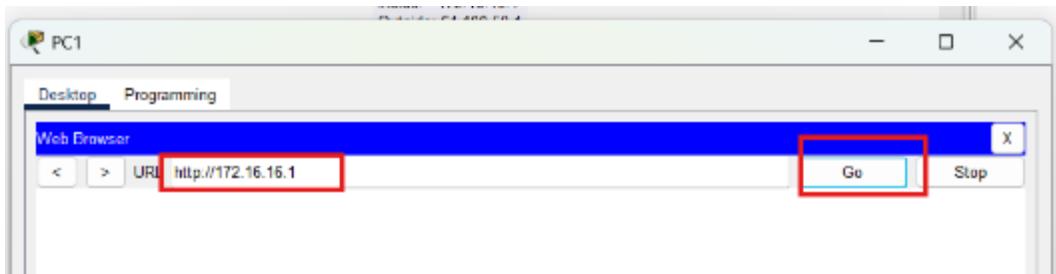
In IPv4 configured networks, clients and servers use private addressing. Before packets with private addressing can cross the internet, they need to be translated to public addressing. Servers that are accessed from outside the organization are usually assigned both a public and a private static IP address. In this activity, you will configure static NAT so that outside devices can access an inside server at its public address.

3.2.17.4 Instructions

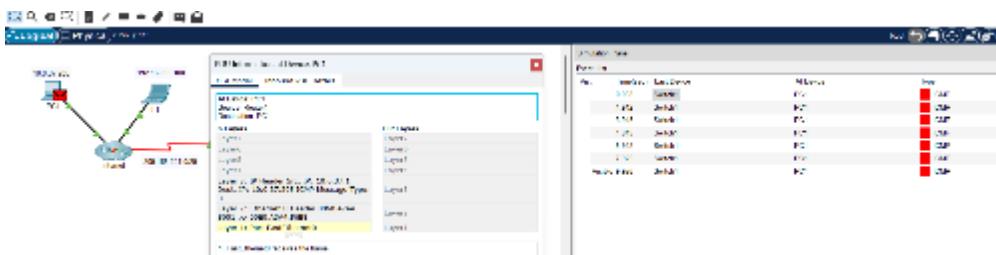
Part 1: Test Access without NAT

Step 1: Attempt to connect to Server1 using Simulation Mode.

- Switch to Simulation mode.
- From **PC1** or **L1**, use the Web Browser to attempt to connect to the **Server1** web page at 172.16.16.1. Continue to click the **Capture Forward** button, notice how the packets never leave the internet cloud. The attempts should fail.



Simulation Panel				
Event List				
Vis.	Time(sec)	Last Device	At Device	Type
0.008	Switch1	PC1		ICMP
1.912	Switch1	PC1		ICMP
3.915	Switch1	PC1		ICMP
4.919	Switch1	PC1		ICMP
5.922	Switch1	PC1		ICMP
7.925	Switch1	PC1		ICMP
Visible 9.926	Switch1	PC1		ICMP



- c. Exit **Simulation mode**.
- d. From **PC1**, ping the **R1** S0/0/0 interface (209.165.201.2). The ping should succeed.

```
Cisco Packet Tracer PC Command Line 1.0
C:>ping 209.165.201.2

Pinging 209.165.201.2 with 32 bytes of data:

Reply from 209.165.201.2: bytes=32 time=1ms TTL=254
Reply from 209.165.201.2: bytes=32 time=7ms TTL=254
Reply from 209.165.201.2: bytes=32 time=12ms TTL=254
Reply from 209.165.201.2: bytes=32 time=4ms TTL=254

Ping statistics for 209.165.201.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 12ms, Average = 6ms

C:>
```

Step 2: View R1 routing table and running-config.

- a. View the running configuration of **R1**. Note that there are no commands referring to NAT. An easy way to confirm this is to issue the following command:

```
R1# show run | include nat
```

Nothing is configured

```
R1>  
R1>enable  
R1# show run | include nat  
R1#
```

- b. Verify that the routing table does not contain entries referring to the IP network addresses for **PC1** and **L1**.

```
R1#show ip route  
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  
  
      172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks  
C        172.16.16.0/28 is directly connected, GigabitEthernet0/0  
L        172.16.16.14/32 is directly connected, GigabitEthernet0/0  
      209.165.201.0/24 is variably subnetted, 2 subnets, 2 masks  
C        209.165.201.0/29 is directly connected, Serial0/0/0  
L        209.165.201.2/32 is directly connected, Serial0/0/0  
  
R1#
```

- c. Verify that NAT is not being used by **R1**.

```
R1# show ip nat translations
```

```
R1#show ip nat translations  
R1#
```

Part 2: Configure Static NAT

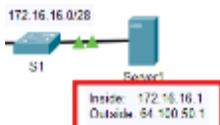
Step 1: Configure static NAT statements.

Refer to the Topology. Create a static NAT translation to map the **Server1** inside address to its outside address.

```
R1(config)# ip nat inside source static 172.16.16.1 64.100.50.1
```

```
Enter configuration commands, one per line. End with Ctrl/Z.  
R1(config)#ip nat inside source static 172.16.16.1 64.100.50.1  
R1(config)#+
```

The command establish translation form one private ip to one public ip in a static way for server 1.



Step 2: Configure interfaces.

- Configure the **G0/0** interface as an inside interface.

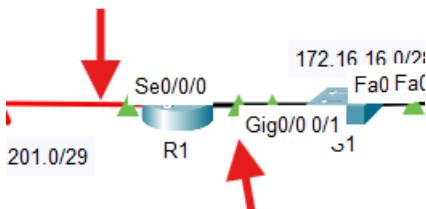
```
R1(config)# interface g0/0
R1(config-if)# ip nat inside
```

- Configure the s0/0/0 public interface as an outside interface.

Answer

```
R1(config)# interface s0/0/0
R1(config-if)# ip nat outside
```

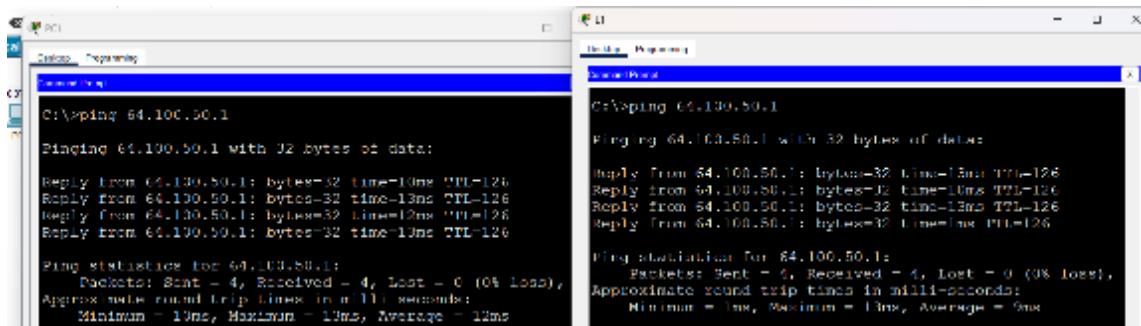
```
R1(config)#interface g0/0
R1(config-if)#ip nat inside
R1(config-if)#interface s0/0/0
R1(config-if)#ip nat outside
R1(config-if) #
```



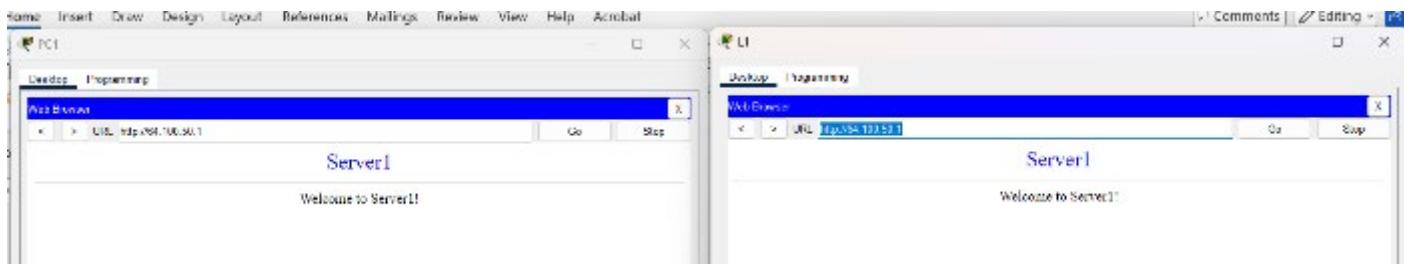
Part 3: Test Access with NAT

Step 1: Verify connectivity to the Server1 web page.

- Open the command prompt on **PC1** or **L1**, attempt to ping the public address for **Server1**. Pings should succeed.



- Verify that both **PC1** and **L1** can now access the **Server1** web page.



Step 2: View NAT translations.

Use the following commands to verify the static NAT configuration on R1:

```
show running-config  
show ip nat translations  
show ip nat statistics
```

```
!  
ip nat inside source static 172.16.16.1 64.100.50.1  
ip classless  
!  
ip flow-export version 9  
!  
!  
!  
!  
!  
!  
Line con 0  
!  
Line aux 0  
!  
Line vty 0 4  
 login  
!  
!  
!  
end  
  
R1#show running-config | section nat  
 ip nat inside  
 ip nat outside  
 ip nat inside source static 172.16.16.1 64.100.50.1  
R1#
```

```
!  
interface GigabitEthernet0/0 ←  
 ip address 172.16.16.14 255.255.255.240  
 ip nat inside  
 duplex auto  
 speed auto  
!  
interface GigabitEthernet0/1  
 no ip address  
 duplex auto  
 speed auto  
 shutdown  
!  
interface GigabitEthernet0/2  
 no ip address  
 duplex auto  
 speed auto  
 shutdown  
!  
interface Serial0/0 ←  
 ip address 209.165.201.2 255.255.255.240  
 ip nat outside  
!  
interface Serial0/0/1
```

```

R1#show ip nat translations
Pro Inside global     Inside local      Outside local      Outside global
tcp 64.100.50.1:80   172.16.16.1:80   209.165.201.1:1026 209.165.201.1:1026
tcp 64.100.50.1:80   172.16.16.1:80   209.165.201.1:1029 209.165.201.1:1029
--- 64.100.50.1       172.16.16.1        ---                  ---
R1#

```

```

R1#show ip nat statistics
Total translations: 3 (1 static, 2 dynamic, 2 extended)
Outside Interfaces: Serial0/0/0
Inside Interfaces: GigabitEthernet0/0
Hits: 25 Misses: 14
Expired translations: 12
Dynamic mappings:
R1#

```

Cisco Packet Tracer - C:\Users\moalp\OneDrive\Documents\NETWORK_ADMINISTRATION_AEC\AU

File Edit Options View Tools Extensions Window Help

Activity Results

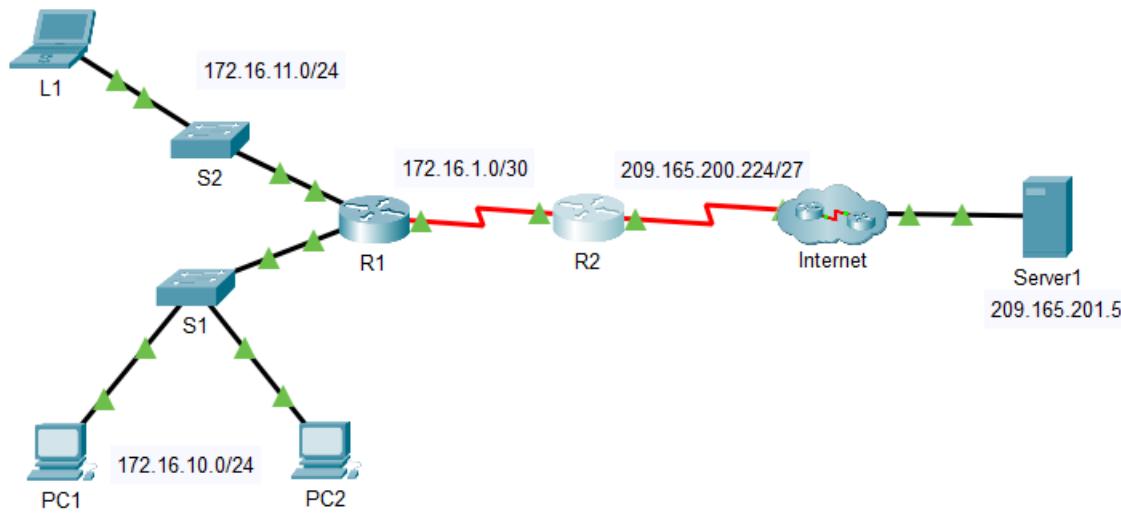
Congratulations Guest! You completed the activity.

Overall Feedback Assessment Items Connectivity Tests

Assessment Items	Status	Points	Component(s)
Network			
R1			
NAT			NAT
Inside Source Static		0	NAT
<input checked="" type="checkbox"/> NAT Source Setting 1 Correct	Correct	50	Static NAT Config...
Ports			Other
GigabitEthernet0/0		0	NAT Interface Con...
<input checked="" type="checkbox"/> NAT Mode	Correct	25	Other
Serial0/0/0		0	NAT Interface Con...
<input checked="" type="checkbox"/> NAT Mode	Correct	25	

3.2.18 Exercice 6.5.6 - Packet Tracer - Configure Dynamic NAT

3.2.18.1 Topology



3.2.18.2 Objectives

Part 1: Configure Dynamic NAT Part 2:

Verify NAT Implementation

3.2.18.3 Instructions

Part 1: Configure Dynamic NAT

Step 1: Configure traffic that will be permitted.

On **R2**, configure one statement for ACL 1 to permit any address belonging to the 172.16.0.0/16 network.

```
R2(config)# access-list 1 permit 172.16.0.0 0.0.255.255
```

Step 2: Configure a pool of address for NAT.

Configure **R2** with a NAT pool that uses two addresses in the 209.165.200.228/30 address space.

```
R2(config)# ip nat pool NAT_POOL_NAME 209.165.200.229 209.165.200.230 netmask 255.255.255.252
```

Notice in the topology there are 3 network addresses that would be translated based on the ACL created. What will happen if more than 2 devices attempt to access the internet?

Answer - The additional devices would be denied access until one of the previous translations timed out freeing up an address to use.

Step 3: Associate ACL 1 with the NAT pool.

Enter the command that associates ACL 1 with the NAT pool that you just created.

```
R2(config)# ip nat inside source list 1 pool NAT_POOL_NAME
```

Step 4: Configure the NAT interfaces.

Configure **R2** interfaces with the appropriate inside and outside NAT commands.

```
R2(config)# interface s0/0/0
R2(config-if)# ip nat outside
R2(config-if)# interface s0/0/1
R2(config-if)# ip nat inside
```

Part 2: Verify NAT Implementation

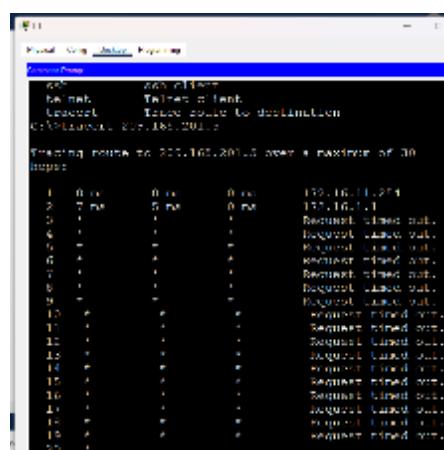
Step 1: Access services across the internet.

From the web browser of **L1**, **PC1**, or **PC2**, access the web page for **Server1**.



A third computer will not be allowed to connect because we only have 2 ip addresses

L1 will not succeed to connect to web server



Step 2: View NAT translations.

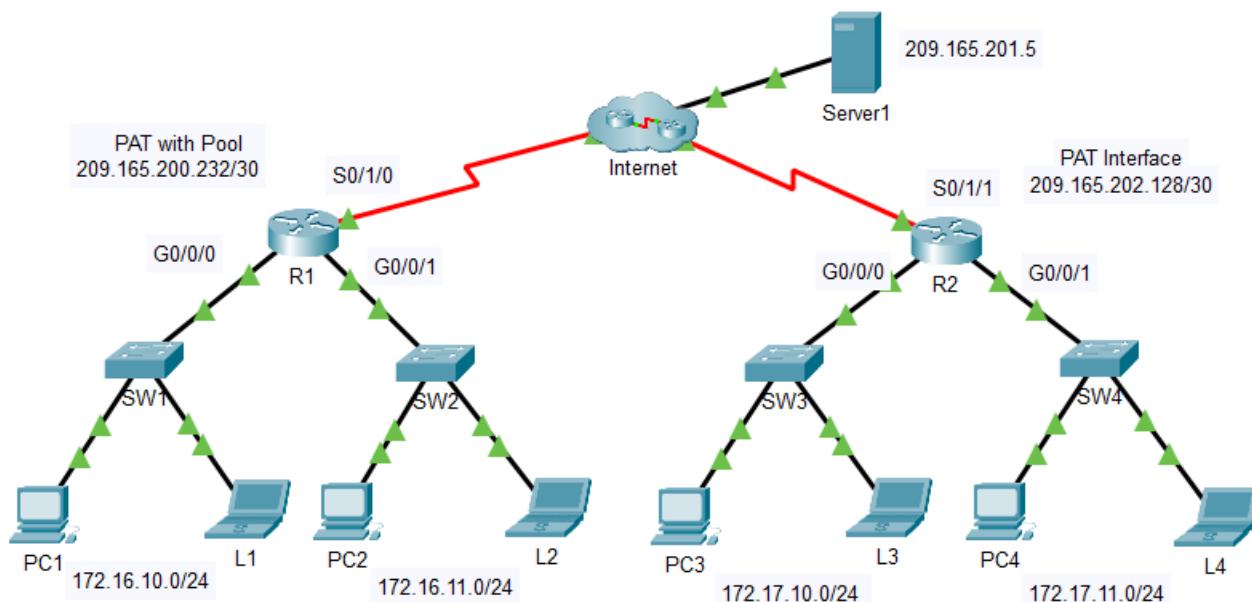
View the NAT translations on R2. Identify the internal source address of the PC and the translated address from the NAT pool in the command output.

```
R2# show ip nat translations
```

```
R2#show ip nat translations
Pro Inside global     Inside local      Outside local      Outside global
tcp 209.165.200.229:1025172.16.10.1:1025  209.15.201.5:80   209.15.201.5:80
tcp 209.165.200.229:1026172.16.10.1:1026  209.15.201.5:80   209.15.201.5:80
tcp 209.165.200.229:1027172.16.10.1:1027  209.165.201.5:80   209.165.201.5:80
tcp 209.165.200.229:1028172.16.10.1:1028  209.165.201.5:80   209.165.201.5:80
tcp 209.165.200.230:1025172.16.10.2:1025  209.165.201.5:80   209.165.201.5:80
tcp 209.165.200.230:1026172.16.10.2:1026  209.165.201.5:80   209.165.201.5:80
R2#
```

3.2.19 Exercice 6.6.7 - Packet Tracer - Configure PAT

3.2.19.1 Topology



3.2.19.2 Objectives

Part 1: Configure Dynamic NAT with Overload

Part 2: Verify Dynamic NAT with Overload Implementation

Part 3: Configure PAT using an Interface

Part 4: Verify PAT Interface Implementation

Part 1: Configure Dynamic NAT with Overload

Step 1: Configure traffic that will be permitted.

On **R1**, configure one statement for ACL 1 to permit any address belonging to 172.16.0.0/16.

```
R1(config)# access-list 1 permit 172.16.0.0 0.0.255.255
```

Step 2: Configure a pool of address for NAT.

Configure **R1** with a NAT pool that uses the two useable addresses in the 209.165.200.232/30 address space.

```
R1(config)# ip nat pool ANY_POOL_NAME 209.165.200.233 209.165.200.234 netmask  
255.255.255.252
```

Step 3: Associate ACL 1 with the NAT pool and allow addresses to be reused.

```
R1(config)# ip nat inside source list 1 pool ANY_POOL_NAME overload
```

Step 4: Configure the NAT interfaces.

Configure **R1** interfaces with the appropriate inside and outside NAT commands.

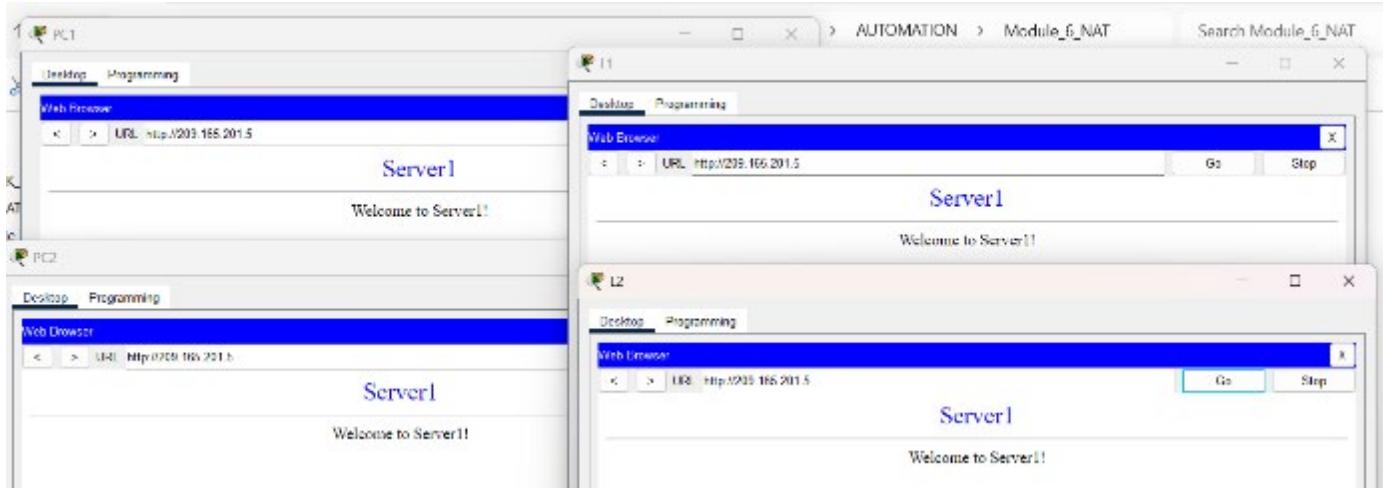
```
R1(config)# interface s0/1/0  
R1(config-if)# ip nat outside  
R1(config-if)# interface g0/0/0  
R1(config-if)# ip nat inside  
R1(config-if)# interface g0/0/1  
R1(config-if)# ip nat inside
```

Part 2: Verify Dynamic NAT with Overload Implementation

Step 1: Access services across the internet.

From the web browser of each of the PCs that use **R1** as their gateway (**PC1**, **L1**, **PC2**, and **L2**), access the web page for **Server1**.

Were all connections successful? Yes



Step 2: View NAT translations

View the NAT translations on R1.

```
R1# show ip nat translations
```

Notice that all four devices were able to communicate, and they are using just one address out of the pool. PAT will continue to use the same address until it runs out of port numbers to associate with the translation. Once that occurs, the next address in the pool will be used. While the theoretical limit would be 65,536 since the port number field is a 16 bit number, the device would likely run out of memory before that limit would be reached.

```
R1 (config)#ena
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#show ip nat translations
Pro Inside global      Inside local        Outside local      Outside global
tcp 209.165.200.233:1024[172.16.10.11:1025] 209.165.201.5:80 209.165.201.5:80
tcp 209.165.200.233:1025[172.16.11.10:1025] 209.165.201.5:80 209.165.201.5:80
tcp 209.165.200.233:1026[172.16.11.11:1025] 209.165.201.5:80 209.165.201.5:80
tcp 209.165.200.233:1027[172.16.10.10:1027] 209.165.201.5:80 209.165.201.5:80

R1 #|
```

Part 3: Configure PAT using an Interface

Step 1: Configure traffic that will be permitted.

On R2, configure one statement for ACL 2 to permit any address belonging to 172.17.0.0/16.

```
R2(config)# access-list 2 permit 172.17.0.0 0.0.255.255
```

Step 2: Associate ACL 2 with the NAT interface and allow addresses to be reused.

Enter the R2 NAT statement to use the interface connected to the internet and provide translations for all internal devices.

```
R2(config)# ip nat inside source list 2 interface s0/1/1 overload
```

Step 3: Configure the NAT interfaces.

Configure **R2** interfaces with the appropriate inside and outside NAT commands.

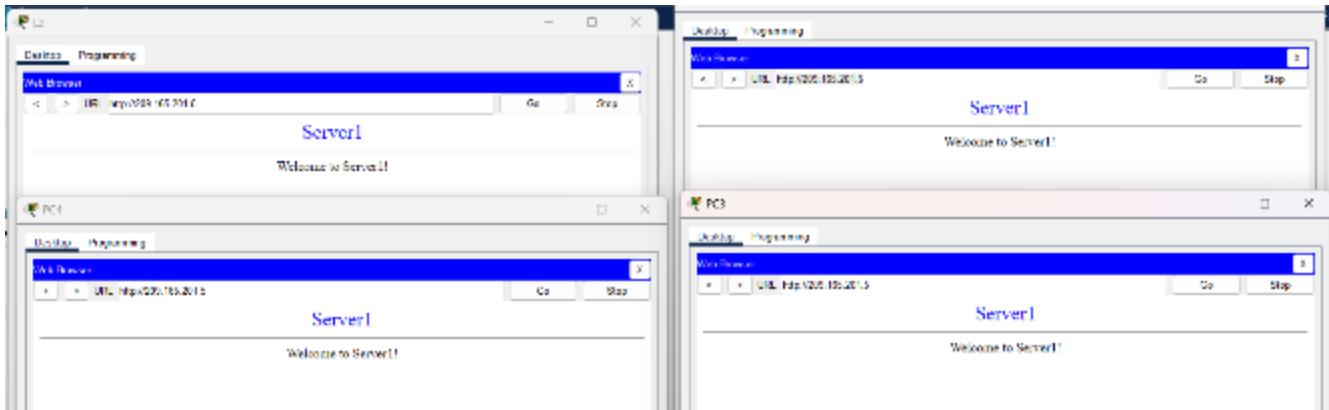
```
R2(config)# interface s0/1/1
R2(config-if)# ip nat outside
R2(config-if)# interface g0/0/0
R2(config-if)# ip nat inside
R2(config-if)# interface g0/0/1
R2(config-if)# ip nat inside
```

Part 4: Verify PAT Interface Implementation

Step 1: Access services across the internet.

From the web browser of each of the PCs that use **R2** as their gateway (**PC3**, **L3**, **PC4**, and **L4**), access the web page for **Server1**.

Were all connections successful? **Answer Yes**



Step 2: View NAT translations.

View the NAT translations on **R2**.

```
R2#show ip nat translations
Pro Inside global     Inside local      Outside local      Outside global
tcp 209.165.202.130:1024172.17.10.10:1025  209.165.201.5:80  209.165.201.5:80
tcp 209.165.202.130:1025172.17.11.11:1025  209.165.201.4:80  209.165.201.4:80
tcp 209.165.202.130:1026172.17.11.11:1026  209.165.201.5:80  209.165.201.5:80
tcp 209.165.202.130:1027172.17.10.11:1025  209.165.201.5:80  209.165.201.5:80
tcp 209.165.202.130:1028172.17.11.10:1025  209.165.201.5:80  209.165.201.5:80
R2#
```

Step 3: Compare NAT statistics on R1 and R2.

Compare the NAT statistics on the two devices. Why

doesn't **R2** list any dynamic mappings?

```

R2#show ip nat statistics
Total translations: 5 (0 static, 5 dynamic, 5 extended)
Outside Interfaces: Serial0/1/1
Inside Interfaces: GigabitEthernet0/0/0 , GigabitEthernet0/0/1
Hits: 43 Misses: 5
Expired translations: 0
Dynamic mappings:
R2#

```

```

R1#show ip nat stat
R1#show ip nat statistics
Total translations: 4 (0 static, 4 dynamic, 4 extended)
Outside Interfaces: Serial0/1/0
Inside Interfaces: GigabitEthernet0/0/0 , GigabitEthernet0/0/1
Hits: 28 Misses: 33
Expired translations: 0
Dynamic mappings:
-- Inside Source
access-list 1 pool ANY_POOL_NAME refCount 4
  pool ANY_POOL_NAME: netmask 255.255.255.252
    start 209.165.200.233 end 209.165.200.234
      type generic, total addresses 2 , allocated 1 (50%), misses 0
R1#
R1#

```

Answer R1 lists dynamic mappings for the pool of addresses that has been configured. R2 only using the outside interface as the address to translate internal addresses to so there is no dynamic mapping.

3.2.19.3 Scripts

Router R1

```

enable
configure terminal
interface GigabitEthernet0/0/0
  ip nat inside
interface GigabitEthernet0/0/1
  ip nat inside
interface Serial0/1/0
  ip nat outside
  ip nat pool DYNAMIC 209.165.200.233 209.165.200.234 netmask 255.255.255.252
  ip nat inside source list 1 pool DYNAMIC overload
  access-list 1 permit 172.16.0.0 0.0.255.255
end

```

Router R2

```

enable
configure terminal
interface GigabitEthernet0/0/0
  ip nat inside

```

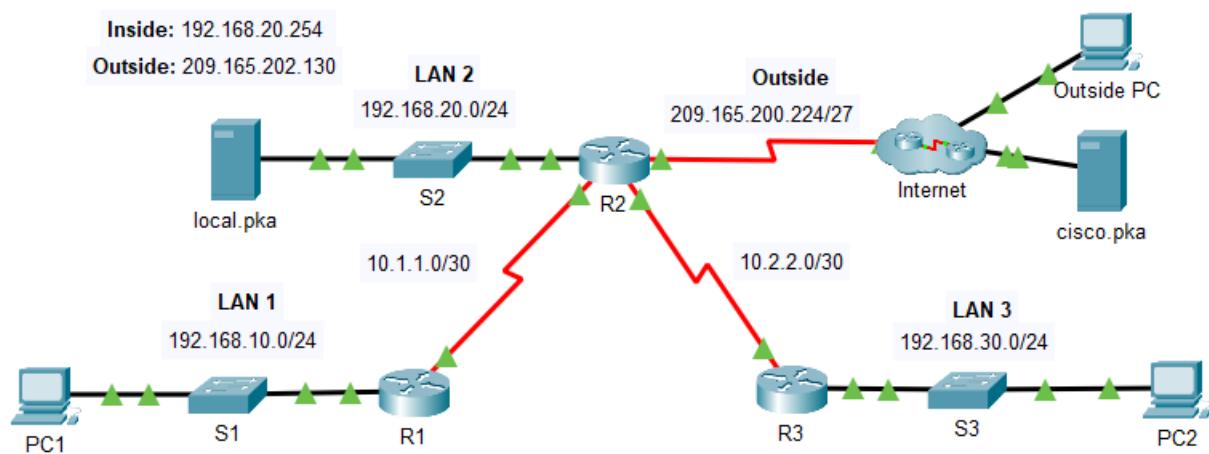
```

interface GigabitEthernet0/0/1
 ip nat inside
interface Serial0/1/1
 ip nat outside
ip nat inside source list 2 interface Serial0/1/1 overload
access-list 2 permit 172.17.0.0 0.0.255.255
end

```

3.2.20 Exercise 6.8.1 -Packet Tracer - Configure NAT for IPv4

3.2.20.1 Topology



3.2.20.2 Addressing Table

Device	Interface	IP Address
R1 <i>R1</i>	S0/0/0	10.1.1.1/30
	F0/0	192.168.10.1/24
R2 <i>R2</i>	S0/0/0	10.1.1.2/30
	S0/0/1	10.2.2.1/30

<i>R2</i>	S0/1/0	209.165.200.225/27
<i>R2</i>	F0/0/0	192.168.20.1/24
R3	S0/0/1	10.2.2.2/30
<i>R3</i>	F0/0	192.168.30.1/24
PC1	NIC	192.168.10.10/24
PC2	NIC	192.168.30.10/24
local.pka	NIC	192.168.20.254/24
Outside PC	NIC	209.165.201.14/28
cisco.pka	NIC	209.165.201.30/28

3.2.20.3 Objectives

- Configure Dynamic NAT with PAT
- Configure Static NAT

3.2.20.4 Background / Scenario

In this lab, you will configure a router with dynamic NAT with PAT. This will translate addresses from the three internal LANs to a single outside address. In addition, you will configure static NAT to translate an internal server address to an outside address.

3.2.20.5 Instructions

In this activity you will only configure router R2.

- Use a named ACL to permit the addresses from LAN1, LAN2, and LAN3 to be translated. Specify the LANs in this order. Use the name **R2NAT**. The name you use must match this name exactly.

```
R2(config)#ip access-list standard R2NAT
R2(config-std-nacl)#permit 192.168.10.0 0.0.0.255
R2(config-std-nacl)#permit 192.168.20.0 0.0.0.255
R2(config-std-nacl)#permit 192.168.30.0 0.0.0.255
```

- Create a NAT pool named **R2POOL**. The pool should use the **first** address from the **209.165.202.128/30** address space. The pool name you use must match this name exactly. All translated addresses must use this address as their outside address.

```
R2(config)#ip nat pool R2POOL 209.165.202.129 209.165.202.129 netmask 255.255.255.252
```

- Configure NAT with the ACL and NAT pool that you have created.

```
R2(config)#ip nat inside source list R2NAT pool R2POOL overload
```

- Configure static NAT to map the local.pka server inside address to the **second** address from the **209.165.202.128/30** address space.

```
R2(config)#ip nat inside source static 192.168.20.254 209.165.202.130
```

- Configure the interfaces that will participate in NAT.

```
R2(config)#interface FastEthernet0/0
R2(config-if)#ip nat inside
R2(config-if)#interface Serial0/0/0
```

```
R2(config-if)#ip nat inside
R2(config-if)#interface Serial0/0/1
R2(config-if)#ip nat inside
R2(config-if)#interface Serial0/1/0
R2(config-if)#ip nat outside
```

3.2.20.6 Scripts

```
!Router R2
enable
configure terminal
!
interface FastEthernet0/0
ip nat inside
!
interface Serial0/0/0
ip nat inside
!
interface Serial0/0/1
ip nat inside
!
interface Serial0/1/0
ip nat outside
!
ip nat pool R2POOL 209.165.202.129 209.165.202.129 netmask 255.255.255.252
ip nat inside source list R2NAT pool R2POOL overload
ip nat inside source static 192.168.20.254 209.165.202.130
!
ip access-list standard R2NAT

!
end
```

3.2.20.7 Test

Ping PC1 to cisco.pka

PC1

Physical Config Desktop Programming

Command Prompt X

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 209.165.201.30

Pinging 209.165.201.30 with 32 bytes of data:

Request timed out.
Reply from 209.165.201.30: bytes=32 time=62ms TTL=125
Reply from 209.165.201.30: bytes=32 time=15ms TTL=125
Reply from 209.165.201.30: bytes=32 time=2ms TTL=125

Ping statistics for 209.165.201.30:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 62ms, Average = 26ms

C:\>|
```

```
R2#show ip nat translations
Pro Inside global      Inside local        Outside local      Outside global
icmp 209.165.202.129:1 192.168.10.10:1    209.165.201.30:1  209.165.201.30:1
icmp 209.165.202.129:2 192.168.10.10:2    209.165.201.30:2  209.165.201.30:2
icmp 209.165.202.129:3 192.168.10.10:3    209.165.201.30:3  209.165.201.30:3
icmp 209.165.202.129:4 192.168.10.10:4    209.165.201.30:4  209.165.201.30:4
--- 209.165.202.130   192.168.20.254     ---           ---
R2#show ip nat statistics
Total translations: 5 (1 static, 4 dynamic, 4 extended)
Outside Interfaces: Serial0/1/0
Inside Interfaces: FastEthernet0/0 , Serial0/0/0 , Serial0/0/1
Hits: 3 Misses: 4
Expired Translations: 0
Dynamic mappings:
-- Inside Source
access-list R2NAT pool R2POOL refCount 4
pool R2POOL netmask 255.255.255.252
    start 209.165.202.129 end 209.165.202.129
    type generic, total addresses 1 , allocated 1 (100%), misses 0
R2#|
```

Ping local.pka to Outside PC

local.pka

Physical Config Services Desktop Programming

Command Prompt

```
Cisco Packet Tracer SERVER Command Line 1.0
C:\>ping 209.165.201.14

Pinging 209.165.201.14 with 32 bytes of data:

Request timed out.
Reply from 209.165.201.14: bytes=32 time=20ms TTL=126
Reply from 209.165.201.14: bytes=32 time=15ms TTL=126
Reply from 209.165.201.14: bytes=32 time=21ms TTL=126

Ping statistics for 209.165.201.14:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 15ms, Maximum = 21ms, Average = 18ms

c:\>
```

```
R2#show ip nat translations
Pro Inside global      Inside local      Outside local      Outside global
icmp 209.165.202.130:1 192.168.20.254:1  209.165.201.14:1  209.165.201.14:1
icmp 209.165.202.130:2 192.168.20.254:2  209.165.201.14:2  209.165.201.14:2
icmp 209.165.202.130:3 192.168.20.254:3  209.165.201.14:3  209.165.201.14:3
icmp 209.165.202.130:4 192.168.20.254:4  209.165.201.14:4  209.165.201.14:4
--- 209.165.202.130   192.168.20.254   ---   ---
```



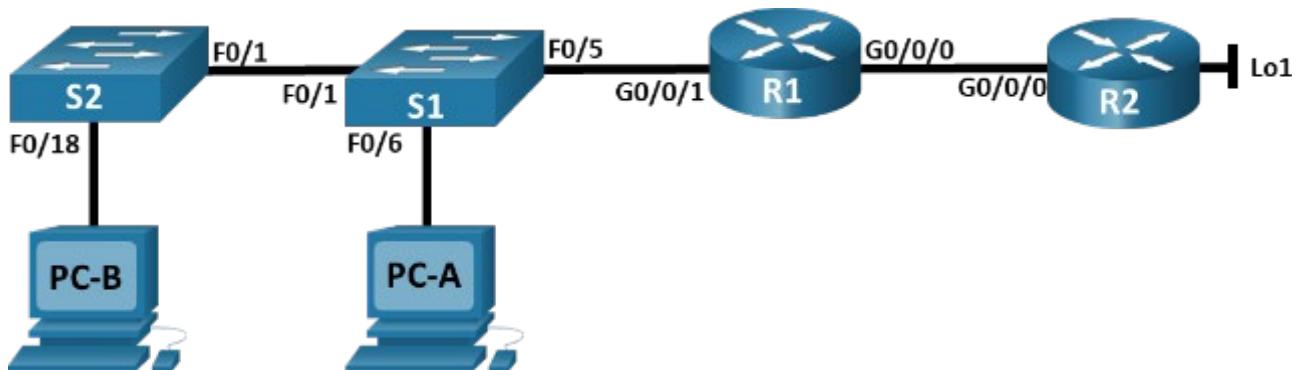
```
R2#show ip nat statistics
Total translations: 5 (1 static, 4 dynamic, 4 extended)
Outside Interfaces: Serial0/1/0
Inside Interfaces: FastEthernet0/0 , Serial0/0/0 , Serial0/0/1
Hits: 6 Misses: 8
Expired translations: 4
Dynamic mappings:
-- Inside Source
access-list R2NAT pool R2POOL refCount 0
  pool R2POOL: netmask 255.255.255.252
    start 209.165.202.129 end 209.165.202.129
      type generic, total addresses 1 , allocated 0 (0%), misses 0
```

Expand/Collapse All Show incorrect items

Assessment Items	Status	Points	Component(s)
Network			
R2			
ACL		0	ACL
R2NAT	Correct	20	ACL to Permit Translation
NAT			
Inside Source List		0	NAT
NAT Source Setting 1	Correct	15	Configure Dynamic NAT
Inside Source Static		0	NAT
NAT Source Setting 1	Correct	15	Configure Static NAT
Pools		0	NAT
Pool Name 1	Correct	15	Configure NAT Pool
Ports			
FastEthernet0/0		0	Other
NAT Mode	Correct	9	Configure NAT Interfaces
Serial0/0/0		0	Other
NAT Mode	Correct	9	Configure NAT Interfaces
Serial0/0/1		0	Other
NAT Mode	Correct	9	Configure NAT Interfaces
Serial0/1/0		0	Other
NAT Mode	Correct	8	Configure NAT Interfaces

3.2.21 Exercise 6.8.2 - Lab - Configure NAT for IPv4

3.2.21.1 Topology



3.2.21.2 Addressing Table

Device	Interface	IP Address	Subnet Mask
R1	G0/0/0	209.165.200.230	255.255.255.248
	G0/0/1	192.168.1.1	255.255.255.0
R2	G0/0/0	209.165.200.225	255.255.255.248
	Lo1	209.165.200.1	255.255.255.224
S1	VLAN 1	192.168.1.11	255.255.255.0
S2	VLAN 1	192.168.1.12	255.255.255.0
PC-A	NIC	192.168.1.2	255.255.255.0
PC-B	NIC	192.168.1.3	255.255.255.0

3.2.21.3 Objectives

Part 1: Build the Network and Configure Basic Device Settings Part 2:

Configure and verify NAT for IPv4

Part 3: Configure and verify PAT for IPv4

Part 4: Configure and verify Static NAT for IPv4

3.2.21.4 Background / Scenario

Network Address Translation (NAT) is the process where a network device, such as a Cisco router, assigns a public address to host devices inside a private network. The main reason to use NAT is to reduce the number of public IP addresses that an organization uses because the number of available IPv4 public addresses is limited.

An ISP has allocated the public IP address space of 209.165.200.224/29 to a company. This network is used to address the link between the ISP router (R2) and the company gateway (R1). The first address

(209.165.200.225) is assigned to the g0/0/0 interface on R2 and the last address (209.165.200.230) is assigned to the g0/0/0 interface on R1. The remaining addresses (209.165.200.226-209.165.200.229) will be used to provide internet access to the company hosts. A default route is used from R1 to R2. The internet is simulated by a loopback address on R2.

In this lab, you will configure various types of NAT. You will test, view, and verify that the translations are taking place, and you will interpret the NAT/PAT statistics to monitor the process.

Note: The routers used with CCNA hands-on labs are Cisco 4221 with Cisco IOS XE Release 16.9.3 (universalk9 image). The switches used in the labs are Cisco Catalyst 2960s with Cisco IOS Release 15.2(2) (lanbasek9 image). Other routers, switches, and Cisco IOS versions can be used. Depending on the model and Cisco IOS version, the commands available and the output produced might vary from what is shown in the labs. Refer to the Router Interface Summary Table at the end of the lab for the correct interface identifiers.

Note: Ensure that the routers and switches have been erased and have no startup configurations. If you are unsure contact your instructor.

3.2.21.5 Required Resources

- 2 Routers (Cisco 4221 with Cisco IOS XE Release 16.9.4 universal image or comparable)
- 2 Switches (Cisco 2960 with Cisco IOS Release 15.2(2) lanbasek9 image or comparable)
- 2 PCs (Windows with a terminal emulation program, such as Tera Term)
- Console cables to configure the Cisco IOS devices via the console ports
- Ethernet cables as shown in the topology

3.2.21.6 Instructions

Part 1: Build the Network and Configure Basic Device Settings

In Part 1, you will set up the network topology and configure basic settings on the PC hosts and switches.

Step 1: Cable the network as shown in the topology.

Attach the devices as shown in the topology diagram and cable as necessary.

Step 2: Configure basic settings for each router.

- a. Assign a device name to the router.

router(config)# hostname R1

- b. Disable DNS lookup to prevent the router from attempting to translate incorrectly entered commands as though they were host names.

R1(config)# no ip domain-lookup

- c. Assign **class** as the privileged EXEC encrypted password.

R1(config)# enable secret class

- d. Assign **cisco** as the console password and enable login.

R1(config)# line console 0

R1(config-line)# password cisco

```
R1(config-line)# login
e. Assign cisco as the VTY password and enable login.
R1(config)# line vty 0 4
R1(config-line)# password cisco
R1(config-line)# login
f. Encrypt the plaintext passwords.
R1(config)# service password-encryption
g. Create a banner that warns anyone accessing the device that unauthorized access is prohibited.
R1(config)# banner motd $ Authorized Users Only! $
h. Configure interface IP addressing as specified in the table above.

R1(config)# interface g0/0/0
R1(config-if)# ip address 209.165.200.230 255.255.255.248
R1(config-if)# no shutdown
R1(config-if)# interface g0/0/1
R1(config-if)# ip address 192.168.1.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
i. Configure a default route to R2 from R1.
R1(config)# ip route 0.0.0.0 0.0.0.0 209.165.200.225
j. Save the running configuration to the startup configuration file.

R1(config)# exit
R1# copy running-config startup-config
```

R1

```
enable
config terminal
hostname R1
no ip domain-lookup
enable secret class
line console 0
password cisco
login
line vty 0 4
password cisco
login
service password-encryption
banner motd $ Authorized Users Only! $
interface g0/0/0
ip address 209.165.200.230 255.255.255.248
no shutdown
interface g0/0/1
ip address 192.168.1.1 255.255.255.0
```

```
no shutdown
ip route 0.0.0.0 0.0.0.0 209.165.200.225
exit
copy running-config startup-config
```

R2

```
enable
config terminal
hostname R2
no ip domain-lookup
enable secret class
line console 0
password cisco
login
line vty 0 4
password cisco
login
service password-encryption
banner motd $ Authorized Users Only! $
interface g0/0/0
ip address 209.165.200.225 255.255.255.248
no shutdown
interface lo1
ip address 209.165.200.1 255.255.255.224
no shutdown
end
copy running-config startup-config
```

Step 3: Configure basic settings for each switch.

- a. Assign a device name to the switch.

switch(config)# hostname S1

- b. Disable DNS lookup to prevent the router from attempting to translate incorrectly entered commands as though they were host names.

S1(config)# no ip domain-lookup

- c. Assign **class** as the privileged EXEC encrypted password.

S1(config)# enable secret class

- d. Assign **cisco** as the console password and enable login.

S1(config)# line console 0

S1(config-line)# password cisco

S1(config-line)# login

- e. Assign **cisco** as the VTY password and enable login.

S1(config)# line vty 0 15

S1(config-line)# password cisco

S1(config-line)# login

- f. Encrypt the plaintext passwords.

```
S1(config)# service password-encryption
g. Create a banner that warns anyone accessing the device that unauthorized access is prohibited.
S1(config)# banner motd $ Authorized Users Only! $
h. Shutdown all interfaces that will not be used.
S1(config)# interface range f0/2-4, f0/7-24, g0/1-2
S1(config-if-range)# shutdown
i. Configure interface IP addressing as specified in the table above.
S1(config)# interface vlan 1
S1(config-if)# ip address 192.168.1.11 255.255.255.0
S1(config-if)# no shutdown
S1(config-if)# exit
S1(config)# ip default-gateway 192.168.1.1
j. Save the running configuration to the startup configuration file.
S1# copy running-config startup-config
```

S1

```
enable
config terminal
hostname S1
no ip domain-lookup
enable secret class
line 0
password cisco
login
line vty 0 15
password cisco
login
service password-encryption
banner motd $ Authorized Users Only! $
interface range f0/2-4, f0/7-24, g0/1-2
shutdown
interface vlan 1
ip address 192.168.1.11 255.255.255.0
no shutdown
ip default-gateway 192.168.1.1
end
copy running-config startup-config
```

S2

```
enable
config terminal
hostname S2
no ip domain-lookup
```

```

enable secret class
line console 0
password cisco
login
line vty 0 15
password cisco
login
service password-encryption
banner motd $ Authorized Users Only! $
interface range f0/2-17, f0/19-24, g0/1-2
shutdown
interface vlan 1
ip address 192.168.1.12 255.255.255.0
no shutdown
ip default-gateway 192.168.1.1
end
copy running-config startup-config

```

Part 2: Configure and verify NAT for IPv4

In Part 2, you will configure and verify NAT for IPv4.

Step 1: Configure NAT on R1 using a pool of three addresses, 209.165.200.226-209.165.200.228.

- Configure a simple access list that defines what hosts are going to be allowed for translation. In this case, all devices on the R1 LAN are eligible for translation.

```
R1(config)# access-list 1 permit 192.168.1.0 0.0.0.255
```

- Create the NAT pool, and give it a name and a range of addresses to use.

```
R1(config)# ip nat pool PUBLIC_ACCESS 209.165.200.226 209.165.200.228 netmask 255.255.255.248
```

Note: The netmask parameter is not an IP address delimiter. It should be the correct subnet mask for the addresses being assigned, even if you are not using all the subnet addresses in the pool.

- Configure the translation, associating the ACL and Pool to the translation process.

```
R1(config)# ip nat inside source list 1 pool PUBLIC_ACCESS
```

Note: Three very important points. First, the word 'inside' is critical to the operation of this kind of NAT. If you omit it, NAT will not work. Second, the list number is the ACL number configured in a previous step. Third, the pool name is case-sensitive.

- Define the inside interface.

```
R1(config)# interface g0/0/1
```

```
R1(config-if)# ip nat inside
```

- Define the outside interface.

```
R1(config)# interface g0/0/0
```

```
R1(config-if)# ip nat outside
```

Step 2: Test and Verify the configuration.

- From PC-B, ping the Lo1 interface (209.165.200.1) on R2. If the ping was unsuccessful, troubleshoot and correct the issues. On R1, display the NAT table on R1 with the command **show ip nat translations**.

```
^C
C:\>ping 209.165.200.1

Pinging 209.165.200.1 with 32 bytes of data:

Request timed out.
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254

Ping statistics for 209.165.200.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

```
R1# show ip nat translations
      Pro Inside global      Inside local      Outside local      Outside global
      --- 209.165.200.226   192.168.1.3      ---                  ---
          icmp 209.165.200.226:1 192.168.1.3:1  209.165.200.1:1  209.165.200.1:1

Total number of translations: 2
```

```
R1#
R1#show ip nat translations
R1#show ip nat translations
      Pro Inside global      Inside local      Outside local      Outside global
      icmp 209.165.200.230:4 192.168.1.3:4  209.165.200.1:4  209.165.200.1:4
      icmp 209.165.200.230:5 192.168.1.3:5  209.165.200.1:5  209.165.200.1:5
      icmp 209.165.200.230:6 192.168.1.3:6  209.165.200.1:6  209.165.200.1:6
      icmp 209.165.200.230:7 192.168.1.3:7  209.165.200.1:7  209.165.200.1:7

R1#
```

What was the inside local address of PC-B translated to? **Answer** 209.165.200.226

What type of NAT address is the translated address? **Answer** Inside global

- From PC-A, ping the Lo1 interface (**209.165.200.1**) on R2. If the ping was unsuccessful, troubleshoot and correct the issues. On R1, display the NAT table on R1 with the command **show ip nat translations**.

```
R1# show ip nat translations
```

Pro	Inside global	Inside local	Outside local	Outside global
---	209.165.200.227	192.168.1.2	---	---
---	209.165.200.226	192.168.1.3	---	---
icmp	209.165.200.227:1	192.168.1.2:1	209.165.200.1:1	209.165.200.1:1

```

        icmp 209.165.200.226:1    192.168.1.3:1          209.165.200.1:1      209.165.200.1:1
Total number of translations: 4

```

```

Cisco Packet Tracer PC Command Line 1.0
C:\>ping 209.165.200.1

Pinging 209.165.200.1 with 32 bytes of data:

Reply from 209.165.200.1: bytes=32 time<1ms TTL=254

Ping statistics for 209.165.200.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>

```

- c. Notice that the previous translation for PC-B is still in the table. From S1, ping the Lo1 interface (**209.165.200.1**) on R2. If the ping was unsuccessful, troubleshoot and correct the issues. On R1, display the NAT table on R1 with the command **show ip nat translations**.

```

R1# show ip nat translations
      Pro Inside global           Inside local           Outside local       local       Outside global
      --- 209.165.200.227        192.168.1.2            ---           ---           ---
      --- 209.165.200.226        192.168.1.3            ---           ---           ---
      --- 209.165.200.228        192.168.1.11           ---           ---           ---
      icmp 209.165.200.226:1   192.168.1.3:1         209.165.200.1:1   209.165.200.1:1
      icmp 209.165.200.228:0   192.168.1.11:0        209.165.200.1:0   209.165.200.1:0

Total number of translations: 5

```

```

R1#show ip nat translations
      Pro Inside global           Inside local           Outside local       Outside global
      icmp 209.165.200.230:1   192.168.1.2:1         209.165.200.1:1   209.165.200.1:1
      icmp 209.165.200.230:2   192.168.1.2:2         209.165.200.1:2   209.165.200.1:2
      icmp 209.165.200.230:3   192.168.1.2:3         209.165.200.1:3   209.165.200.1:3
      icmp 209.165.200.230:4   192.168.1.2:4         209.165.200.1:4   209.165.200.1:4

R1#show ip nat stats
R1#show ip nat statistics
Total translations: 4 (0 static, 4 dynamic, 4 extended)
Outside Interfaces: GigabitEthernet0/0/0
Inside Interfaces: GigabitEthernet0/0/1
Hits: 7  Misses: 8
Expired Translations: 4
Dynamic mappings:
R1#

```

- d. Now try and ping R2 Lo1 from S2. This time, the translations fail, and you get these messages (or similar) on the R1 console:

```

Sep 23 15:43:55.562: %IOSXE-6-PLATFORM: R0/0: cpp_cp: QFP:0.0 Thread:000
TS:00000001473688385900 %NAT-6-ADDR_ALLOC_FAILURE: Address allocation failed; pool 1
may be exhausted [2]

```

```

S2#ping 209.165.200.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.200.1, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 0/0/0 ms
S2#

```

- e. This is an expected result, because only 3 addresses are allocated, and we tried to ping Lo1 from four devices. Recall that NAT is a one-to-one translation. So how long are the translations allocated? Issue the command **show ip nat translations verbose** and you will see that the answer is for 24 hours.

```

R1# show ip nat translations verbose
      Pro Inside global           Inside local           Outside local           Outside global
      --- 209.165.200.226       192.168.1.3          ---                   ---
create: 09/23/19 15:35:27, use: 09/23/19 15:35:27, timeout: 23:56:42
Map-Id(In) : 1
<output omitted>

```

- f. Given that the pool is limited to three addresses, NAT to a pool of addresses is not adequate for our application. Clear the NAT translations and statistics and we will move on to PAT.

```

R1# clear ip nat translations *
R1# clear ip nat statistics

```

Part 3: Configure and verify PAT for IPv4

In Part 3, you will configure replace NAT with PAT to a pool of addresses, and then with PAT using an interface.

Step 1: Remove the translation command on R1.

The components of an Address Translation configuration are basically the same; something (an access-list) to identify addresses eligible to be translated, an optionally configured pool of addresses to translate them to, and the commands necessary to identify the inside and outside interfaces. From Part 1, our access-list (access-list 1) is still correct for the network scenario, so there is no need to recreate it. We are going to use the same pool of addresses, so there is no need to recreate that configuration either. Also, the inside and outside interfaces are not changing. To get started in Part 3, remove the command that ties the ACL and pool together.

```
R1(config)# no ip nat inside source list 1 pool PUBLIC_ACCESS
```

Step 2: Add the PAT command on R1.

Now, configure for PAT translation to a pool of addresses (remember, the ACL and Pool are already configured, so this is the only command we need to change from NAT to PAT).

```
R1(config)# ip nat inside source list 1 pool PUBLIC_ACCESS overload
```

Step 3: Test and Verify the configuration.

- a. Let's verify PAT is working. From PC-B, ping the Lo1 interface (209.165.200.1) on R2. If the ping was unsuccessful, troubleshoot and correct the issues. On R1, display the NAT table on R1 with the command **show ip nat translations**.

```
R1# show ip nat translations
```

```

Pro Inside global           Inside local            Outside local          Outside global
      icmp 209.165.200.226:1    192.168.1.3:1        209.165.200.1:1      209.165.200.1:1
Total number of translations: 1#

```

What was the inside local address of PC-B translated to? **Answer** - 209.165.200.226

What type of NAT address is the translated address?

Answer Inside Global

What is different about the output of the **show ip nat translations** command from the NAT exercise?

Answer - There is no dedicated translation between an inside and outside address listed.

- b. From PC-A, ping the Lo1 interface (209.165.200.1) on R2. If the ping was unsuccessful, troubleshoot and correct the issues. On R1, display the NAT table on R1 with the command **show ip nat translations**.

```
R1# show ip nat translations
```

```

Pro Inside global           Inside local            Outside local
      Outside global icmp 209.165.200.226:1
                           192.168.1.2:1        209.165.200.1:1
                           209.165.200.1:1

```

Total number of translations: 1

Notice that there is only one translation again. Send the ping once more, and quickly go back to the router and issue the command **show ip nat translations verbose** and you will see what happened.

As you can see, the translation timeout has been dropped from 24 hours to 1 minute.

- c. Generate traffic from multiple devices to observe PAT. On PC-A and PC-B, use the -t parameter with the ping command to send a non-stop ping to R2's Lo1 interface (**ping -t 209.165.200.1**), then go back to R1 and issue the **show ip nat translations** command:

Notice that the inside global address is the same for both sessions.

How does the router keep track of what replies go where? **Answer** -

Unique Port Numbers are assigned.

- d. PAT to a pool is a very effective solution for small-to-midsize organizations. However, there are unused IPv4 addresses involved in this scenario. We will move to PAT with interface overload to eliminate this waste of IPv4 addresses. Stop the pings on PC-A and PC-B with the Control-C key combination, then clear translations and translation statistics:

```

R1# clear ip nat translations *
R1# clear ip nat statistics

```

Step 4: On R1, remove the nat pool translation commands.

Once again, our access-list (access-list 1) is still correct for the network scenario, so there is no need to recreate it. Also, the inside and outside interfaces are not changing. To get started with PAT to an interface, clean up the configuration by removing the NAT Pool and the command that ties the ACL and pool together.

```
R1(config)# no ip nat inside source list 1 pool PUBLIC_ACCESS overload
R1(config)# no ip nat pool PUBLIC_ACCESS
```

Step 5: Add the PAT overload command by specifying the outside interface.

Add the PAT command that will cause overload to the outside interface.

```
R1(config)# ip nat inside source list 1 interface g0/0/0 overload
```

Step 6: Test and Verify the configuration.

- a. Let's verify PAT to the interface is working. From PC-B, ping the Lo1 interface (209.165.200.1) on R2. If the ping was unsuccessful, troubleshoot and correct the issues. On R1, display the NAT table on R1 with the command **show ip nat translations**.

```
R1# show ip nat translations
```

Pro	Inside global	Inside local	Outside local	Outside global
icmp	209.165.200.230:1	192.168.1.3:1	209.165.200.1:1	209.165.200.1:1

Total number of translations: 1

- b. Generate traffic from multiple devices to observe PAT. On PC-A and PC-B, use the -t parameter with the ping command to send a non-stop ping to R2's Lo1 interface (**ping -t 209.165.200.1**). On S1 and S2, issue the privileged exec command **ping 209.165.200.1 repeat 2000**. Then go back to R1 and issue the **show ip nat translations** command.

```
R1# show ip nat translations
```

Pro	Inside global	Inside local	Outside local	Outside global
icmp	209.165.200.230:3	192.168.1.11:1	209.165.200.1:1	209.165.200.1:3
icmp	209.165.200.230:2	192.168.1.2:1	209.165.200.1:1	209.165.200.1:2
icmp	209.165.200.230:4	192.168.1.3:1	209.165.200.1:1	209.165.200.1:4
icmp	209.165.200.230:1	192.168.1.12:1	209.165.200.1:1	209.165.200.1:1

Total number of translations: 4

Now all the Inside Global addresses are mapped to the g0/0/0 interface IP address. Stop all the pings. On PC-A and PC-B, using the CTRL-C key combination.

Part 4: Configure and verify Static NAT for IPv4

In Part 4, you will configure static NAT so that PC-A is directly reachable from the internet. PC-A will be reachable from R2 via the address 209.165.200.229.

Note: The configuration you are about to complete does not follow recommended practices for internet-connected gateways. This lab completely omits what would be standard security practices to focus on successful configuration of static NAT. In a production environment, careful coordination between the network infrastructure and security teams would be fundamental to supporting this requirement.

Step 1: On R1, clear current translations and statistics.

```
R1# clear ip nat translations *
R1# clear ip nat statistics
```

Step 2: On R1, configure the NAT command required to statically map an inside address to an outside address.

For this step, configure a static mapping between 192.168.1.11 and 209.165.200.1 using the following command:

```
R1(config)# ip nat inside source static 192.168.1.2 209.165.200.229
```

Step 3: Test and Verify the configuration.

- a. Let's verify the Static NAT is working. On R1, display the NAT table on R1 with the command **show ip nat translations**, and you should see the static mapping.

```
R1# show ip nat translations
```

Pro	Inside global global	Inside local	Outside local	Outside
---	209.165.200.229	192.168.1.2	---	---
Total number of translations: 1				

- b. The translation table shows the static translation is in effect. Verify this by pinging from R2 to 209.165.200.229. The pings should work.

Note: you may have to disable the PC firewall for the pings to work.

- c. On R1, display the NAT table on R1 with the command **show ip nat translations**, and you should see the static mapping and the port-level translation for the inbound pings.

```
R1# show ip nat translations
```

Pro	Inside global	Inside local	Outside local	Outside global
---	209.165.200.229	192.168.1.2	---	---
icmp	209.165.200.229:3	192.168.1.2:3	209.165.200.225:3	
209.165.200.225:3				
Total number of translations: 2				

This validates that the Static NAT is working.

3.2.21.7 Router Interface Summary Table

Router Model	Ethernet Interface #1	Ethernet Interface #2	Serial Interface #1	Serial Interface #2
1800	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)

Router Model	Ethernet Interface #1	Ethernet Interface #2	Serial Interface #1	Serial Interface #2
1900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2801	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)
2811	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
4221	Gigabit Ethernet 0/0/0 (G0/0/0)	Gigabit Ethernet 0/0/1 (G0/0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)
4300	Gigabit Ethernet 0/0/0 (G0/0/0)	Gigabit Ethernet 0/0/1 (G0/0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)

Note: To find out how the router is configured, look at the interfaces to identify the type of router and how

many interfaces the router has. There is no way to effectively list all the combinations of configurations for each router class. This table includes identifiers for the possible combinations of Ethernet and Serial interfaces in the device. The table does not include any other type of interface, even though a specific router may contain one. An example of this might be an ISDN BRI interface. The string in parenthesis is the legal abbreviation that can be used in Cisco IOS commands to represent the interface.

3.2.21.8 Scripts

```
! =====
! Router R1 Configuration Script
! =====

! -----
! Part 1: Basic Device Settings
! -----
enable
configure terminal
hostname R1
no ip domain-lookup
enable secret class
line console 0
password cisco
login
line vty 0 4
password cisco
login
service password-encryption
banner motd $ Authorized Users Only! $

! Configure Interfaces
interface g0/0/0
ip address 209.165.200.230 255.255.255.248
no shutdown
exit
interface g0/0/1
ip address 192.168.1.1 255.255.255.0
no shutdown
exit

! Configure Default Route
ip route 0.0.0.0 0.0.0.0 209.165.200.225
exit

! Save Configuration
write memory

! -----
! Part 2: Configure NAT for IPv4
! -----
enable
configure terminal
access-list 1 permit 192.168.1.0 0.0.0.255
ip nat pool PUBLIC_ACCESS 209.165.200.226 209.165.200.228 netmask
255.255.255.248
ip nat inside source list 1 pool PUBLIC_ACCESS
interface g0/0/1
ip nat inside
exit
interface g0/0/0
ip nat outside
exit
end
```

```

! -----
! Part 3: Configure PAT for IPv4
! -----
enable
configure terminal
no ip nat inside source list 1 pool PUBLIC_ACCESS
ip nat inside source list 1 pool PUBLIC_ACCESS overload
end

! -----
! Part 4: Configure PAT with Interface Overload
! -----
enable
configure terminal
no ip nat inside source list 1 pool PUBLIC_ACCESS overload
no ip nat pool PUBLIC_ACCESS
ip nat inside source list 1 interface g0/0/0 overload
end

! -----
! Part 5: Configure Static NAT for IPv4
! -----
enable
clear ip nat translations *
clear ip nat statistics
configure terminal
ip nat inside source static 192.168.1.2 209.165.200.229
end

! Save Configuration
write memory

! =====
! End of Script
! =====

! =====
! Router R2 Configuration Script
! =====

! -----
! Part 1: Basic Device Settings
! -----
enable
configure terminal
hostname R2
no ip domain-lookup
enable secret class
line console 0
password cisco
login
line vty 0 4
password cisco
login
service password-encryption
banner motd $ Authorized Users Only! $

! Configure Interfaces
interface g0/0/0
ip address 209.165.200.225 255.255.255.248
no shutdown
exit
interface lo1
ip address 209.165.200.1 255.255.255.224

```

```
no shutdown
exit
end
! Save Configuration
write memory

! =====
! End of Script
! =====

! =====
! Switch S1 Configuration Script
! =====

! -----
! Part 1: Basic Device Settings
! -----
enable
configure terminal
hostname S1
no ip domain-lookup
enable secret class
line console 0
password cisco
login
line vty 0 15
password cisco
login
service password-encryption
banner motd $ Authorized Users Only! $

! Shutdown Unused Interfaces
interface range f0/2-4, f0/7-24, g0/1-2
shutdown
exit

! Configure VLAN 1 Interface
interface vlan 1
ip address 192.168.1.11 255.255.255.0
no shutdown
exit

! Configure Default Gateway
ip default-gateway 192.168.1.1
exit

! Save Configuration
write memory

! =====
! End of Script
! =====

! =====
! Switch S2 Configuration Script
! =====

! -----
! Part 1: Basic Device Settings
! -----
enable
```

```

configure terminal
hostname S2
no ip domain-lookup
enable secret class
line console 0
password cisco
login
line vty 0 15
password cisco
login
service password-encryption
banner motd $ Authorized Users Only! $

! Shutdown Unused Interfaces
interface range f0/2-17, f0/19-24, g0/1-2
shutdown
exit

! Configure VLAN 1 Interface
interface vlan 1
ip address 192.168.1.12 255.255.255.0
no shutdown
exit

! Configure Default Gateway
ip default-gateway 192.168.1.1
exit

! Save Configuration
write memory

! =====
! End of Script
! =====

```

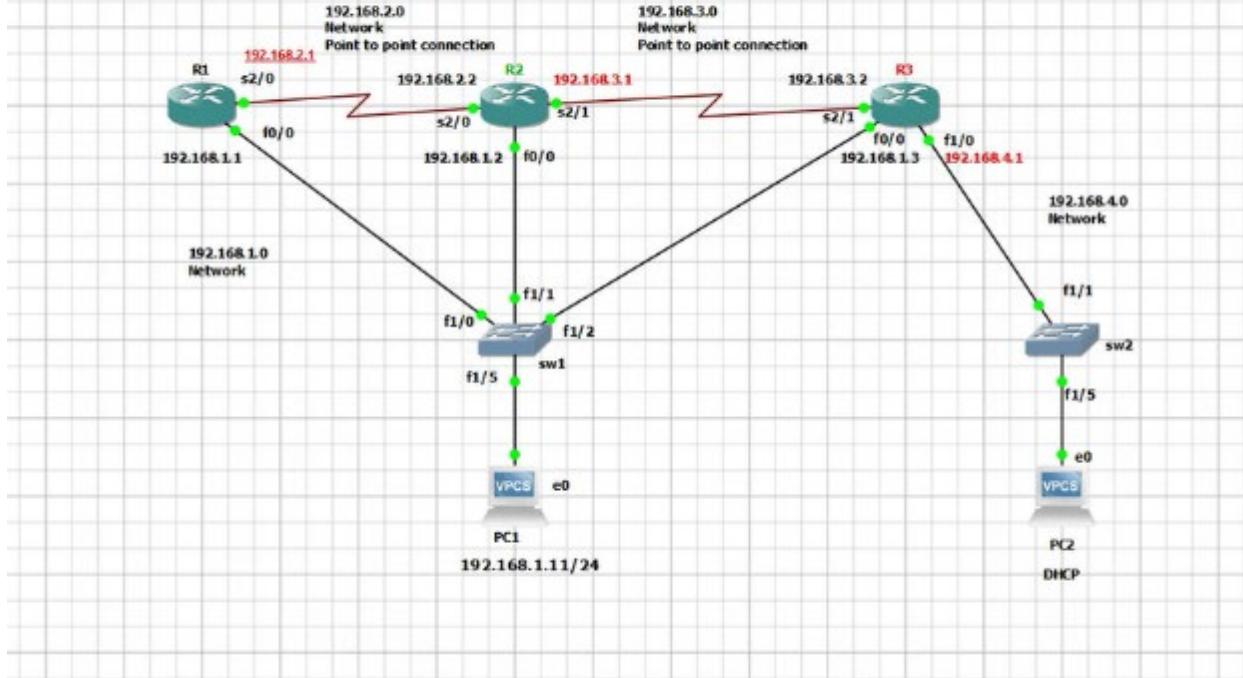
3.3 GNS3 Labs

3.3.1 OSPF and Packet Capture - DR BDR

3.3.1.1 Topology

OSPF and Packet Capture - DR BDR

OSPF 10 all are in Area 0



3.3.1.2 Addressing table

Network Element	Connection	Port	IP Address
R1	Connection to R2	S2/0	192.168.2.1/24
	Connection to SW1	F0/0	192.168.1.1/24
R2	Connection to R1	S2/0	192.168.2.2/24
	Connection to R3	S2/1	192.168.3.1/24
	Connection to SW1	F0/0	192.168.1.2/24
R3	Connection to R2	S2/1	192.168.3.2/24
	Connection to SW1	F0/0	192.168.1.3/24
	Connection to SW2	F1/0	192.168.4.1/24
SW1	Connection to R1	F1/0	N/A
	Connection to R2	F1/1	N/A
	Connection to R3	F1/2	N/A
	Connection to PC1	F1/5	N/A
SW2	Connection to R3	F1/0	N/A
	Connection to PC2	F1/5	N/A

PC's

PC1	Connection to SW1	NIC	192.168.1.11/24
PC2	Connection to SW2	NIC	DHCP-assigned

3.3.1.3 Configure topology

1. Create topology on GNS3 based on the diagram and addressing table
2. Basic configuration of network elements.

```
!!! ROUTER R1
enable
configure terminal
hostname R1
no ip domain lookup
banner motd #WARNING Authorized Users Only! #
end
write memory
```

```
!!! ROUTER R2
enable
configure terminal
hostname R2
no ip domain lookup
banner motd #WARNING Authorized Users Only! #
end
write memory
```

```
!!! ROUTER R3
enable
configure terminal
hostname R3
no ip domain lookup
banner motd #WARNING Authorized Users Only! #
end
write memory
```

```
!!! SWITCH SW1
enable
configure terminal
hostname SW1
no ip domain lookup
banner motd #WARNING Authorized Users Only! #
end
```

```
write memory
```

```
!!! SWITCH SW2
enable
configure terminal
hostname SW2
no ip domain lookup
banner motd #WARNING Authorized Users Only! #
end
write memory
```

3. Configure 1Pv4 address on PCl of l92.l68.l.ll with a default gateway of l92.l68.l.l, then save the address to NVRAM.

```
PC1> ip 192.168.1.11 255.255.255.0 192.168.1.1
PC1> save
PC1> show
```

4. Configure routers

```
!!!!!!!
!!!! Router R1:
!!!!!!!
enable
conf t

int s2/0
description Connection between R1 and R2
ip address 192.168.2.1 255.255.255.0
no shutdown
exit

int f0/0
description Connection between R1 and SW1
ip address 192.168.1.1 255.255.255.0
no shutdown
exit
end
write memory

show ip interface brief
```

```

WARNING Authorized Users Only!
R1#
R1#
R1#
R1#
R1#show ip interface brief
Interface          IP-Address      OK? Method Status          Protocol
FastEthernet0/0    192.168.1.1    YES manual up           up
FastEthernet1/0    unassigned      YES unset administratively down down
FastEthernet1/1    unassigned      YES unset administratively down down
Serial2/0          192.168.2.1    YES manual up           down
Serial2/1          unassigned      YES unset administratively down down
Serial2/2          unassigned      YES unset administratively down down
Serial2/3          unassigned      YES unset administratively down down
R1#
```

!!!!!!!!!!!!!!

!!! Router R2:

!!!!!!!!!!!!!!

```

enable
conf t
```

```

int s2/0
description Connection between R2 and R1
ip address 192.168.2.2 255.255.255.0
no shutdown
exit
```

```

int s2/1
description Connection between R2 and R3
ip address 192.168.3.1 255.255.255.0
no shutdown
exit
```

```

int f0/0
description Connection between R2 and SW1
ip add 192.168.1.2 255.255.255.0
no shut
exit
end
write memory
```

show ip interface brief

```

R2#
R2#show ip interface brief
Interface          IP-Address      OK? Method Status          Protocol
FastEthernet0/0    192.168.1.2    YES manual up           up
FastEthernet1/0    unassigned      YES unset administratively down down
FastEthernet1/1    unassigned      YES unset administratively down down
Serial2/0          192.168.2.2    YES manual up           up
Serial2/1          192.168.3.1    YES manual up           down
Serial2/2          unassigned      YES unset administratively down down
Serial2/3          unassigned      YES unset administratively down down
R2#
```

!!!!!!!!!!!!!!

!!! Router R3:

```
!!!!!!!!!!!!!!!
enable
conf t

int s2/1
description Connection between R3 and R2
ip add 192.168.3.2 255.255.255.0
no shut

int f0/0
description Connection between R3 and SW1
ip add 192.168.1.3 255.255.255.0
no shut

int f1/0
description Connection between R3 and SW2
ip add 192.168.4.1 255.255.255.0
no shut
exit
end
write memory
```

show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	192.168.1.3	YES	manual	up	up
FastEthernet1/0	192.168.4.1	YES	manual	up	up
FastEthernet1/1	unassigned	YES	unset	administratively down	down
Serial2/0	unassigned	YES	unset	administratively down	down
Serial2/1	192.168.3.2	YES	manual	up	up
Serial2/2	unassigned	YES	unset	administratively down	down
Serial2/3	unassigned	YES	unset	administratively down	down

3.3.1.4 Configure DHCP

- Configure R3 as DHCP server

```
!!!!!!!!!!!!!!!
!!! DHCP Configuration on R3:
!!!!!!!!

enable
config t
```

ip dhcp excluded-address 192.168.4.1 192.168.4.10

ip dhcp pool NETWORK_192.168.4.0
 network 192.168.4.0
 default-router 192.168.4.1
 domain-name cisco.com
 dns-server 8.8.8.8

```
exit  
end  
write memory
```

2. Verify DHCP configuration

```
show ip dhcp pool
```

```
R3#show ip dhcp pool

Pool NETWORK_192.168.4.0 :
  Utilization mark (high/low)      : 100 / 0
  Subnet size (first/next)        : 0 / 0
  Total addresses                : 254
  Leased addresses               : 0
  Excluded addresses             : 10
  Pending event                  : none
  1 subnet is currently in the pool :
    Current index          IP address range           Leased/Excluded/Total
    192.168.4.1            192.168.4.1 - 192.168.4.254     0 / 10 / 254
R3#
```

```
show ip dhcp binding
```

```
R3#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address      Client-ID/          Lease expiration       Type      State      Interface
                  Hardware address/
                  User name
R3#
```

```
show ip dhcp server statistics
```

```
R3#show ip dhcp server statistics
Memory usage          15808
Address pools          1
Database agents         0
Automatic bindings       0
Manual bindings         0
Expired bindings        0
Malformed messages      0
Secure arp entries      0
Renew messages          0
Workspace timeouts       0
Static routes           0
Relay bindings           0
Relay bindings active     0
Relay bindings terminated 0
Relay bindings selecting   0

Message                Received
BOOTREQUEST             0
DHCPDISCOVER             0
DHCPREQUEST              0
DHCPDECLINE              0
DHCPRELEASE              0
DHCPINFORM               0
DHCPVENDOR               0
BOOTREPLY                 0
DHCPOFFER                 0
DHCPACK                  0
DHCPNAK                  0

Message                Sent
BOOTREPLY                 0
DHCPOFFER                 0
DHCPACK                  0
DHCPNAK                  0

Message                Forwarded
BOOTREQUEST              0
DHCPDISCOVER              0
```

```

Message           Forwarded
BOOTREQUEST        0
DHCPDISCOVER       0
DHCPREQUEST        0
DHCPDECLINE        0
DHCPRELEASE         0
DHCPINFORM         0
DHCPVENDOR         0
BOOTREPLY          0
DHCPOFFER          0
DHCPACK            0
DHCPNAK            0

DHCP-DPM Statistics
Offer notifications sent      0
Offer callbacks received       0
Classname requests sent       0
Classname callbacks received   0

R3#
```

3. Configure PC2 to be addresses automatically by DHCP, then save address to NVRAM.

```

PC2> ip dhcp
PC2> save
PC2> show
```

```

PC2> show
NAME    IP/MASK          GATEWAY          MAC          LPORT  RHOST:PORT
PC2    0.0.0.0/0          0.0.0.0          00:50:79:66:68:01  20002  127.0.0.1:20003
      fe80::250:79ff:fe66:6801/64

PC2> ip dhcp
DDORA IP 192.168.4.11/24 GW 192.168.4.1

PC2> show
NAME    IP/MASK          GATEWAY          MAC          LPORT  RHOST:PORT
PC2    192.168.4.11/24    192.168.4.1    00:50:79:66:68:01  20002  127.0.0.1:20003
      fe80::250:79ff:fe66:6801/64

PC2> save
Saving startup configuration to startup.vpc
. done

PC2>
```

4. Check on Router 3 (DHCP server)

```
show ip dhcp pool
```

```
R3#show ip dhcp pool

Pool NETWORK_192.168.4.0 :
  Utilization mark (high/low)      : 100 / 0
  Subnet size (first/next)        : 0 / 0
  Total addresses                : 254
  Leased addresses               : 1 ←
  Excluded addresses             : 10
  Pending event                  : none
  1 subnet is currently in the pool :
    Current index          IP address range           Leased/Excluded/Total
    192.168.4.12          192.168.4.1 - 192.168.4.254   1 / 10 / 254
R3#
```

solarwinds | Solar-PuTTY free tool

show ip dhcp binding

```
R3#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address      Client-ID/          Lease expiration       Type      State      Interface
              Hardware address/
              User name
192.168.4.11   0100.5079.6668.01   Feb 25 2025 07:12 PM  Automatic  Active   FastEthernet1/0
R3#
```

show ip dhcp server statistics

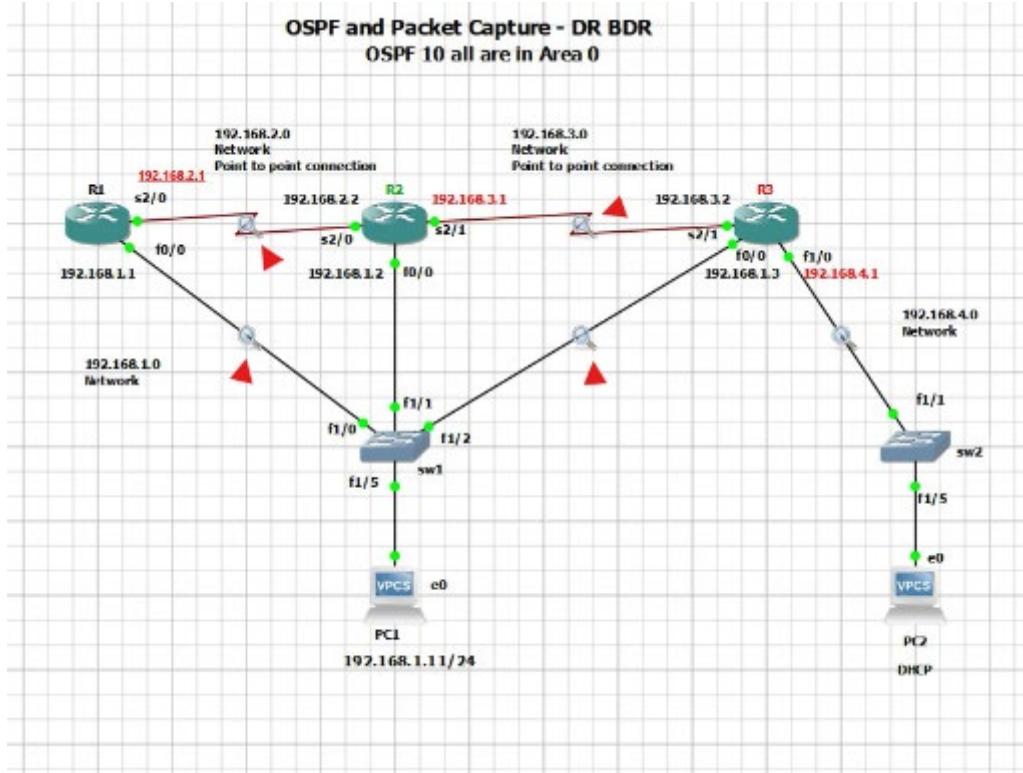
```
R3#show ip dhcp server statistics
Memory usage          16067
Address pools         1
Database agents       0
Automatic bindings    1
Manual bindings       0
Expired bindings      0
Malformed messages   0
Secure arp entries   0
Renew messages        0
Workspace timeouts   0
Static routes         0
Relay bindings        0
Relay bindings active 0
Relay bindings terminated 0
Relay bindings selecting 0
```

Message	Received
BOOTREQUEST	0
DHCPDISCOVER	2
DHCPPREQUEST	1
DHCPDECLINE	0
DHCPRLEASE	0
DHCPIINFORM	0
DHCPVENDOR	0
BOOTREPLY	0
DHCPOFFER	0
DHCPACK	0
DHCPNAK	0
Message	Sent
BOOTREPLY	0
DHCPOFFER	1
DHCPACK	1
DHCPNAK	0
Message	Forwarded
BOOTREQUEST	0
DHCPDISCOVER	0
DHCPPREQUEST	0
DHCPDECLINE	0
DHCPRLEASE	0
DHCPIINFORM	0
DHCPVENDOR	0
BOOTREPLY	0
DHCPOFFER	0
DHCPACK	0
DHCPNAK	0
DHCP-DPM Statistics	
Offer notifications sent	0
Offer callbacks received	0
Classname requests sent	0
Classname callbacks received	0

R3#

3.3.1.5 OSPF configuration

1. Initiate wireshark trace on Routers links



2. Configure OSPF 10 on all routers, then save configuration to NVRAM.

```
!!OSPF Router R1:  
enable  
config t  
router ospf 10  
network 192.168.1.0 0.0.0.255 area 0  
network 192.168.2.0 0.0.0.255 area 0  
end  
copy run start
```

```
R1#config t
R1#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#router ospf 10
R1(config-router)#area 0
R1(config-router)#network 192.168.1.0 0.0.0.255 area 0
R1(config-router)#end
R1(config router)#end
R1(config-router)#end
R1#copy run start
R1#copy run startup-config
Destination filename [startup-config]?
Feb 24 19:46:00.811: %SYS-5-CONFIG_I: Configured from console by console

Feb 24 19:46:15.075: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.3.1 on Serial2/0 from LOADING to FULL, Loading Done
Feb 24 19:46:15.487: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.3.1 on FastEthernet0/0 From LOADING to FULL, Loading Done
Feb 24 19:47:06.531: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.4.1 on FastEthernet0/0 From LOADING to FULL, Loading Done

Building configuration...
[ok]
R1#
R1#
```

!!!OSPF Router R2:

```
enable
conf t
router ospf 10
network 192.168.1.0 0.0.0.255 area 0
network 192.168.2.0 0.0.0.255 area 0
network 192.168.3.0 0.0.0.255 area 0
exit
end
write memory
```

```
R2#config t
R2#enable
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#router ospf 10
R2(config-router)#area 0
R2(config-router)#network 192.168.1.0 0.0.0.255 area 0
R2(config-router)#network 192.168.2.0 0.0.0.255 area 0
R2(config-router)#network 192.168.3.0 0.0.0.255 area 0
R2(config-router)#exit
R2(config)#write memory
% Invalid input detected at '^' marker.

R2(config)#
Feb 24 19:46:15.135: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.2.1 on Serial2/0 from LOADING to FULL, Loading Done
R2(config)#
R2(config)#
R2#w
Feb 24 19:46:24.959: %SYS-5-CONFIG_I: Configured from console by console
R2#write memory
building configuration...
[ok]
R2#
Feb 24 19:46:54.035: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.2.1 on FastEthernet0/0 From LOADING to FULL, Loading Done
R2#
Feb 24 19:46:54.819: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.4.1 on Serial2/1 From LOADING to FULL, Loading Done
R2#
Feb 24 19:47:06.541: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.4.1 on FastEthernet0/0 From LOADING to FULL, Loading Done
R2#
```

!!!! OSPF Router R3:

```
enable
conf t
router ospf 10
network 192.168.1.0 0.0.0.255 area 0
network 192.168.3.0 0.0.0.255 area 0
network 192.168.4.0 0.0.0.255 area 0
exit
end
write memory
```

```

R3#
R3>!!!! OSPF Router R3:
R3#enable
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#router ospf 10
R3(config-router)#network 192.168.1.0 0.0.0.255 area 0
R3(config-router)#network 192.168.3.0 0.0.0.255 area 0
R3(config-router)#network 192.168.4.0 0.0.0.255 area 0
R3(config-router)#exit
R3(config)#end
R3#
R3#write memory
Building configuration...
[OK]
R3#
*Feb 24 19:48:59.855: %SYS-5-CONFIG_I: Configured from console by console
*Feb 24 19:47:01.495: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.3.1 on Serial2/1 from LOADING to FULL, Loading Done
R3#
R3#
*Feb 24 19:47:06.543: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.5.1 on FastEthernet0/0 from LOADING to FULL, Loading Done
*Feb 24 19:47:06.567: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.2.1 on FastEthernet0/0 from LOADING to FULL, Loading Done
R3#

```

3. Verify routing table on all routers with the show ip route command

```

!R1
show ip route
show ip route ospf
show ip ospf neighbor
show ip ospf database
show ip ospf interface
show ip ospf

```

```

R1#show ip route
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
      + - replicated route, % - next hop override

Gateway of last resort is not set

      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, FastEthernet0/0
L        192.168.1.1/32 is directly connected, FastEthernet0/0
      192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.2.0/24 is directly connected, Serial2/0
L        192.168.2.1/32 is directly connected, Serial2/0
O  ▶  192.168.3.0/24 [110/65] via 192.168.1.3, 00:02:31, FastEthernet0/0
      [110/65] via 192.168.1.2, 00:02:51, FastEthernet0/0
O  ▶  192.168.4.0/24 [110/2] via 192.168.1.3, 00:02:31, FastEthernet0/0
R1#show ip route ospf
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
      + - replicated route, % - next hop override

Gateway of last resort is not set

O  ▶  192.168.3.0/24 [110/65] via 192.168.1.3, 00:02:49, FastEthernet0/0
      [110/65] via 192.168.1.2, 00:03:09, FastEthernet0/0
O  ▶  192.168.4.0/24 [110/2] via 192.168.1.3, 00:02:49, FastEthernet0/0
R1#

```

```
R1#show ip ospf neighbor
Neighbor ID      Pri  State      Dead Time     Address          Interface
192.168.3.1       0    FULL/ -      00:00:35     192.168.2.2      Serial2/0
192.168.3.1       1    FULL/DR      00:00:35     192.168.1.2      FastEthernet0/0
192.168.4.1       1    FULL/DROTHER  00:00:32     192.168.1.3      FastEthernet0/0
R1#show ip ospf database
              OSPF Router with ID (192.168.2.1) (Process ID 10)
              Router Link States (Area 0)
Link ID        ADV Router      Age      Seq#      Checksum Link count
192.168.2.1    192.168.2.1   218      0x80000003 0x00C3D8 3
192.168.3.1    192.168.3.1   214      0x80000003 0x00E5CC 5
192.168.4.1    192.168.4.1   203      0x80000003 0x001CF8 4
              Net Link States (Area 0)
Link ID        ADV Router      Age      Seq#      Checksum
192.168.1.2    192.168.3.1   207      0x80000002 0x00E131
R1#
R1#
R1#show ip ospf database
              OSPF Router with ID (192.168.2.1) (Process ID 10)
              Router Link States (Area 0)
Link ID        ADV Router      Age      Seq#      Checksum Link count
192.168.2.1    192.168.2.1   232      0x80000003 0x00C3D8 3
192.168.3.1    192.168.3.1   228      0x80000003 0x00E5CC 5
192.168.4.1    192.168.4.1   217      0x80000003 0x001CF8 4
              Net Link States (Area 0)
Link ID        ADV Router      Age      Seq#      Checksum
192.168.1.2    192.168.3.1   221      0x80000002 0x00E131
R1#
```

```
R1#show ip ospf interface
Serial2/0 is up, line protocol is up
  Internet Address 192.168.2.1/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.2.1, Network Type POINT TO POINT, Cost: 64
  Topology-MTID      Cost      Disabled      Shutdown      Topology Name
    0            64          no           no           Base
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 192.168.3.1
    Suppress hello for 0 neighbor(s)
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.1.1/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.2.1, Network Type BROADCAST Cost: 1
  Topology-MTID      Cost      Disabled      Shutdown      Topology Name
    0            1          no           no           Base
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 192.168.3.1, Interface address 192.168.1.2
  Backup Designated router (ID) 192.168.2.1, Interface address 192.168.1.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:08
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 192.168.3.1 ((Designated Router))
    Adjacent with neighbor 192.168.4.1
    Suppress hello for 0 neighbor(s)
R1#
```

```
R1#show ip ospf
Routing Process "ospf_10" with ID 192.168.2.1
Start time: 00:14:51.280, Time elapsed: 00:06:00.868
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
Supports area transit capability
Supports NSSA (compatible with RFC 3101)
Event-log enabled, Maximum number of events: 1000, Mode: cyclic
Router is not originating router-LSAs with maximum metric
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPFs 10000 msec
Maximum wait time between two consecutive SPFs 10000 msec
Incremental-SPF disabled
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msec
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msec
Retransmission pacing timer 66 msec
Number of external LSA 0. Checksum Sum 0x000000
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Number of areas transit capable is 0
External flood list length 0
IETF NSF helper support enabled
Cisco NSF helper support enabled
Reference bandwidth unit is 100 mbps
Area BACKBONE(0)
    Number of interfaces in this area is 2
    Area has no authentication
    SPF algorithm last executed 00:04:40.388 ago
    SPF algorithm executed 5 times
    Area ranges are
    Number of LSA 4. Checksum Sum 0x02A7CD
    Number of opaque link LSA 0. Checksum Sum 0x000000
    Number of DCbitless LSA 0
    Number of indication LSA 0
    Number of DoNotAge LSA 0
    Flood list length 0

R1#
```

```
!R2
show ip route
show ip route ospf
show ip ospf neighbor
show ip ospf database
show ip ospf interface
show ip ospf
```

```
R2#show ip route
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
      + - replicated route, % - next hop override

Gateway of last resort is not set

      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C          192.168.1.0/24 is directly connected, FastEthernet0/0
L          192.168.1.2/32 is directly connected, FastEthernet0/0
      192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C          192.168.2.0/24 is directly connected, Serial2/0
L          192.168.2.2/32 is directly connected, Serial2/0
      192.168.3.0/24 is variably subnetted, 2 subnets, 2 masks
C          192.168.3.0/24 is directly connected, Serial2/1
L          192.168.3.1/32 is directly connected, Serial2/1
O      192.168.4.0/24 [110/2] via 192.168.1.3, 00:18:00, FastEthernet0/0
R2#
```

```
R2#show ip route ospf
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
      + - replicated route, % - next hop override

Gateway of last resort is not set

O      192.168.4.0/24 [110/2] via 192.168.1.3, 00:18:12, FastEthernet0/0
R2#
```

```
R2#show ip ospf neighbor

Neighbor ID      Pri   State            Dead Time    Address          Interface
192.168.4.1      0     FULL/ -          00:00:34    192.168.3.2    Serial2/1
192.168.2.1      0     FULL/ -          00:00:31    192.168.2.1    Serial2/0
192.168.2.1      1     FULL/BDR ◀       00:00:35    192.168.1.1    FastEthernet0/0
192.168.4.1      1     FULL/DROTHER   00:00:38    192.168.1.3    FastEthernet0/0
R2#
R2#
R2#show ip ospf database

        OSPF Router with ID (192.168.3.1) (Process ID 10)

        Router Link States (Area 0)

Link ID          ADV Router      Age        Seq#      Checksum Link count
192.168.2.1     192.168.2.1    1145      0x80000003 0x00C3D8 3
192.168.3.1     192.168.3.1    1139      0x80000003 0x00E5CC 5
192.168.4.1     192.168.4.1    1129      0x80000003 0x001CF8 4

        Net Link States (Area 0)

Link ID          ADV Router      Age        Seq#      Checksum
192.168.1.2     192.168.3.1    1132      0x80000002 0x00E131
R2#
R2#
```

```
v 47 new 48 new 49 new 50 new 51 new 52 issl
R2#show ip ospf interface
Serial2/1 is up, line protocol is up
  Internet Address 192.168.3.1/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.3.1, Network Type POINT_TO_POINT, Cost: 64
  Topology-MTID   Cost   Disabled   Shutdown   Topology Name
    0       64      no        no          Base
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:07
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 3/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 192.168.4.1
  Suppress hello for 0 neighbor(s)
Serial2/0 is up, line protocol is up
  Internet Address 192.168.2.2/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.3.1, Network Type POINT_TO_POINT, Cost: 64
  Topology-MTID   Cost   Disabled   Shutdown   Topology Name
    0       64      no        no          Base
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:05
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 192.168.2.1
  Suppress hello for 0 neighbor(s)
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.1.2/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.3.1, Network Type BROADCAST, Cost: 1
  Topology-MTID   Cost   Disabled   Shutdown   Topology Name
    0       1      no        no          Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.3.1, Interface address 192.168.1.2
  Backup Designated router (ID) 192.168.2.1, Interface address 192.168.1.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:07
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 2
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 192.168.2.1 (Backup Designated Router)
    Adjacent with neighbor 192.168.4.1
  Suppress hello for 0 neighbor(s)
R2#
```

```
R2#show ip ospf
Routing Process "ospf_10" with ID 192.168.3.1
Start time: 00:15:06.160, Time elapsed: 00:20:18.916
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
Supports area transit capability
Supports NSSA (compatible with RFC 3101)
Event-log enabled, Maximum number of events: 1000, Mode: cyclic
Router is not originating router-LSAs with maximum metric
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPFs 10000 msec
Maximum wait time between two consecutive SPFs 10000 msec
Incremental-SPF disabled
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msec
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msec
Retransmission pacing timer 66 msec
Number of external LSA 0. Checksum Sum 0x000000
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Number of areas transit capable is 0
External flood list length 0
IETF NSF helper support enabled
Cisco NSF helper support enabled
Reference bandwidth unit is 100 mbps
Area BACKBONE(0)
    Number of interfaces in this area is 3
    Area has no authentication
    SPF algorithm last executed 00:19:13.280 ago
    SPF algorithm executed 4 times
    Area ranges are
        Number of LSA 4. Checksum Sum 0x02A7CD
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
    Flood list length 0
```

a.

```
! R3
show ip route
show ip route ospf
show ip ospf neighbor
show ip ospf database
show ip ospf interface
show ip ospf
```

```

R3#show ip route
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
      + - replicated route, % - next hop override

Gateway of last resort is not set

      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C          192.168.1.0/24 is directly connected, FastEthernet0/0
L          192.168.1.3/32 is directly connected, FastEthernet0/0
O <  192.168.2.0/24 [110/65] via 192.168.1.2, 00:34:43, FastEthernet0/0
                  [110/65] via 192.168.1.1, 00:34:43, FastEthernet0/0
      192.168.3.0/24 is variably subnetted, 2 subnets, 2 masks
C          192.168.3.0/24 is directly connected, Serial2/1
L          192.168.3.2/32 is directly connected, Serial2/1
      192.168.4.0/24 is variably subnetted, 2 subnets, 2 masks
C          192.168.4.0/24 is directly connected, FastEthernet1/0
L          192.168.4.1/32 is directly connected, FastEthernet1/0
R3#
R3#show ip route ospf
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
      + - replicated route, % - next hop override

Gateway of last resort is not set

O <  192.168.2.0/24 [110/65] via 192.168.1.2, 00:34:54, FastEthernet0/0
                  [110/65] via 192.168.1.1, 00:34:54, FastEthernet0/0
R3#

```

```

R3#
R3#show ip ospf neighbor
      Neighbor ID      Pri      State            Dead Time      Address          Interface
192.168.3.1          0      FULL/ -           00:00:34    192.168.3.1      Serial2/1
192.168.2.1          1      FULL/BDR -         00:00:37    192.168.1.1      FastEthernet0/0
192.168.3.1          1      FULL/DR -         00:00:31    192.168.1.2      FastEthernet0/0
R3#
R3#show ip ospf database
      OSPF Router with ID (192.168.4.1) (Process ID 10)

      Router Link States (Area 0)
      Link ID      ADV Router      Age      Seq#      Checksum Link count
192.168.2.1      192.168.2.1      235      0x80000004 0x00C1D9 3
192.168.3.1      192.168.3.1      231      0x80000004 0x00E3CD 5
192.168.4.1      192.168.4.1      185      0x80000004 0x001AF9 4

      Net Link States (Area 0)
      Link ID      ADV Router      Age      Seq#      Checksum
192.168.1.2      192.168.3.1      231      0x80000003 0x00DF32
R3#
R3#

```

```
R3#show ip ospf interface
FastEthernet1/0 is up, line protocol is up
  Internet Address 192.168.4.1/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.4.1, Network Type BROADCAST, Cost: 1
  Topology-MTID      Cost      Disabled      Shutdown      Topology Name
    0            1        no          no          Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.4.1, Interface address 192.168.4.1
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:00
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 3/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
Serial2/1 is up, line protocol is up
  Internet Address 192.168.3.2/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.4.1, Network Type POINT_TO_POINT, Cost: 64
  Topology-MTID      Cost      Disabled      Shutdown      Topology Name
    0            64        no          no          Base
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 192.168.3.1
  Suppress hello for 0 neighbor(s)
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.1.3/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.4.1, Network Type BROADCAST, Cost: 1
  Topology-MTID      Cost      Disabled      Shutdown      Topology Name
    0            1        no          no          Base
  Transmit Delay is 1 sec, State DROTHER, Priority 1
  Designated Router (ID) 192.168.3.1, Interface address 192.168.1.2
  Backup Designated router (ID) 192.168.2.1, Interface address 192.168.1.1
  Timer intervals configured, Hello 10, Dead 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 192.168.2.1 (Backup Designated Router)
    Adjacent with neighbor 192.168.3.1 (Designated Router)
  Suppress hello for 0 neighbor(s)
```

```

R3#show ip ospf
Routing Process "ospf 10" with ID 192.168.4.1
Start time: 00:15:50.940, Time elapsed: 00:38:41.824
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
Supports area transit capability
Supports NSSA (compatible with RFC 3101)
Event-log enabled, Maximum number of events: 1000, Mode: cyclic
Router is not originating router-LSAs with maximum metric
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPFs 10000 msec
Maximum wait time between two consecutive SPFs 10000 msec
Incremental-SPF disabled
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msec
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msec
Retransmission pacing timer 66 msec
Number of external LSA 0. Checksum Sum 0x000000
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Number of areas transit capable is 0
External flood list length 0
IETF NSF helper support enabled
Cisco NSF helper support enabled
Reference bandwidth unit is 100 mbps
Area BACKBONE(0)
    Number of interfaces in this area is 3
    Area has no authentication
    SPF algorithm last executed 00:38:25.208 ago
    SPF algorithm executed 2 times
    Area ranges are
    Number of LSA 4. Checksum Sum 0x029FD1
    Number of opaque link LSA 0. Checksum Sum 0x000000
    Number of DCbitless LSA 0
    Number of indication LSA 0
    Number of DoNotAge LSA 0
    Flood list length 0

```

3.3.1.6 Wireshark

The following Wireshark captures were taken at configuration of routers.

The address 224.0.0.5 is a multicast address used in OSPF (Open Shortest Path First) to communicate with all OSPF routers on the same network segment. This address is known as the **AllSPFRouters** address.

Key Points:

- Hello Packets: OSPF routers use 224.0.0.5 to send Hello packets to discover and maintain neighbor relationships.
- Link-State Advertisements (LSAs): The Designated Router (DR) and Backup Designated Router (BDR) use this address to send LSAs to all OSPF routers on the network.
- Efficiency: Using multicast addresses like 224.0.0.5 helps reduce network traffic by allowing a single packet to be received by multiple routers simultaneously

The address 224.0.0.6 is another multicast address used in OSPF (Open Shortest Path First) for communication between OSPF routers. Specifically, this address is known as the **AllDRouters** address.

Key Points:

- Designated Router (DR) and Backup Designated Router (BDR): OSPF routers use 224.0.0.6 to communicate with the DR and BDR on a multiaccess network.
- LSA Updates: When OSPF routers need to send updates to the DR and BDR, they use this multicast address to ensure the information is properly received and processed.

Summary

224.0.0.5: Used to send messages to **all OSPF routers**.

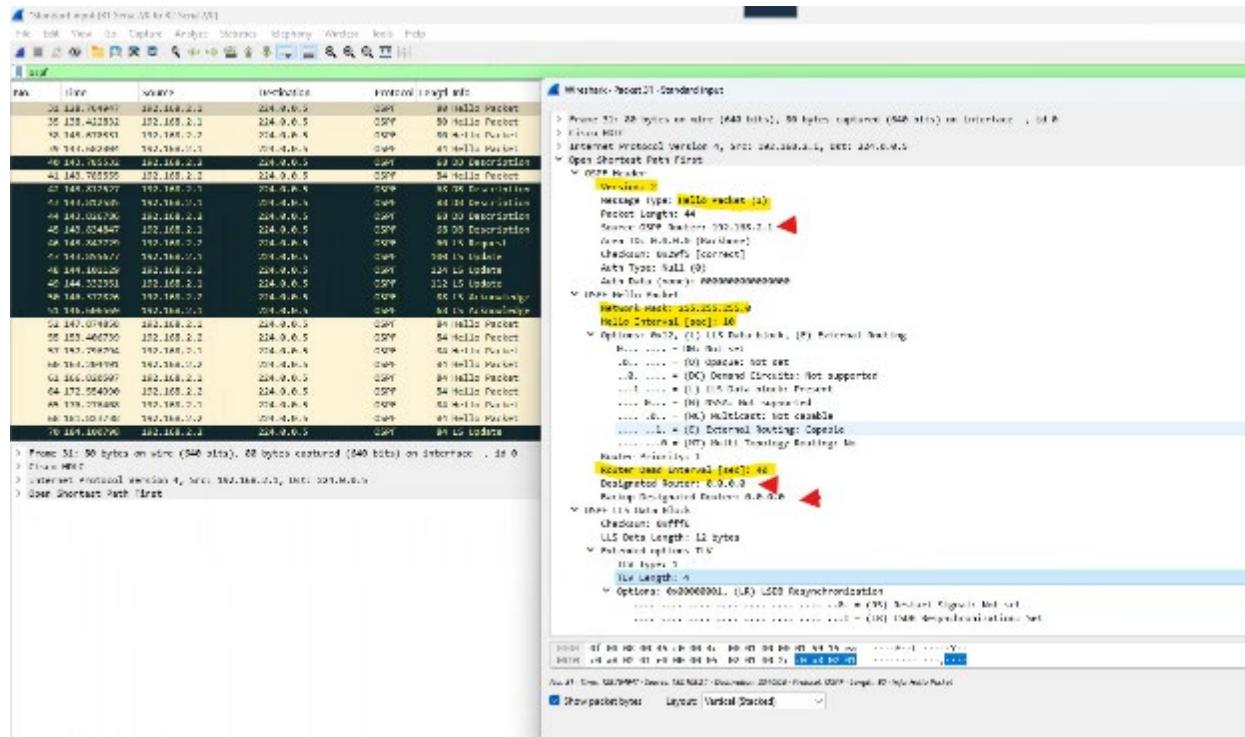
224.0.0.6: Used to send messages specifically to the **DR and BDR**.

This differentiation helps in reducing unnecessary traffic and improving the efficiency of OSPF operations on multiaccess networks.

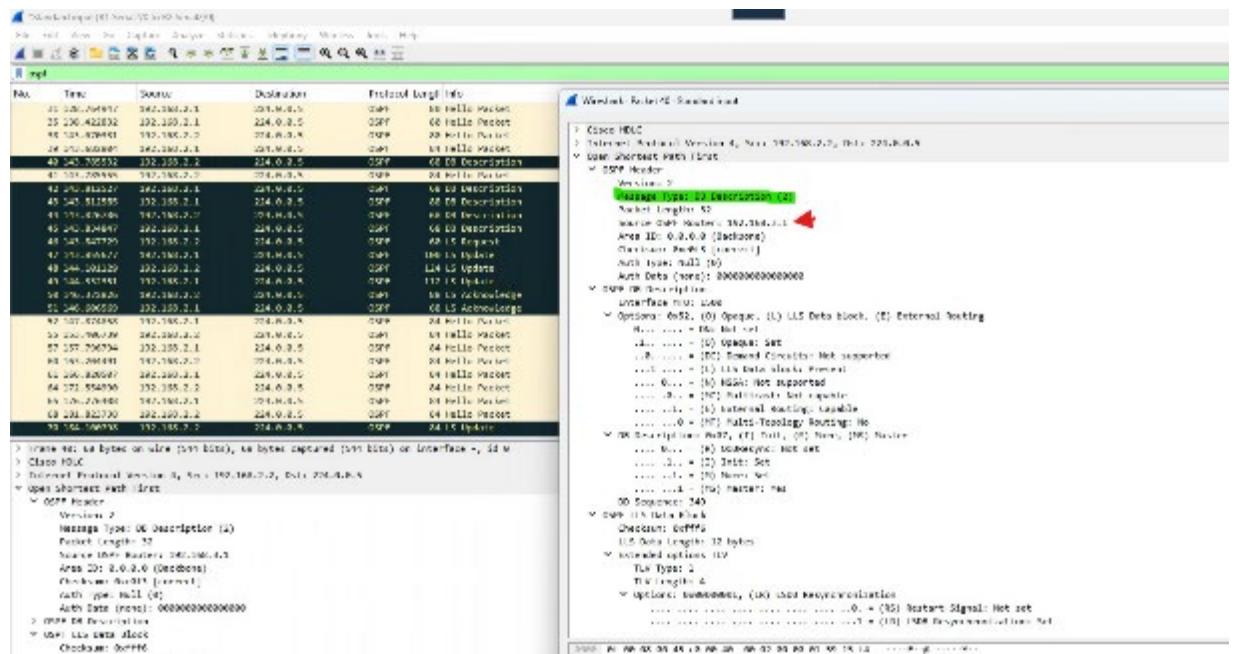
R1 to R2 - OSPF filter

Hello Packet

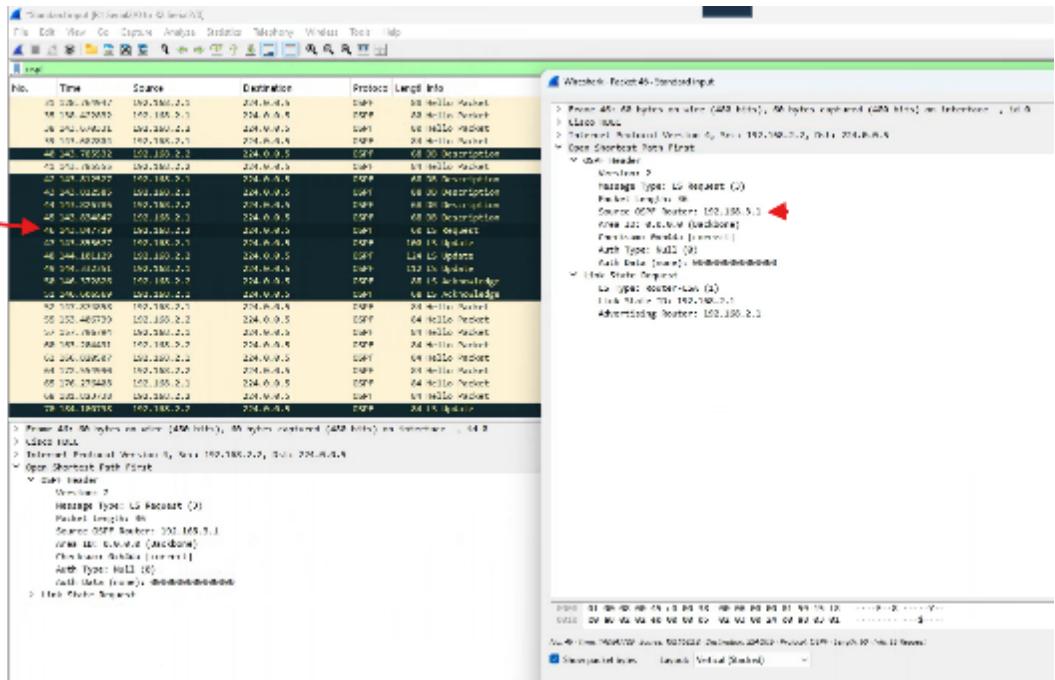
Note no DR no BDR (first message)



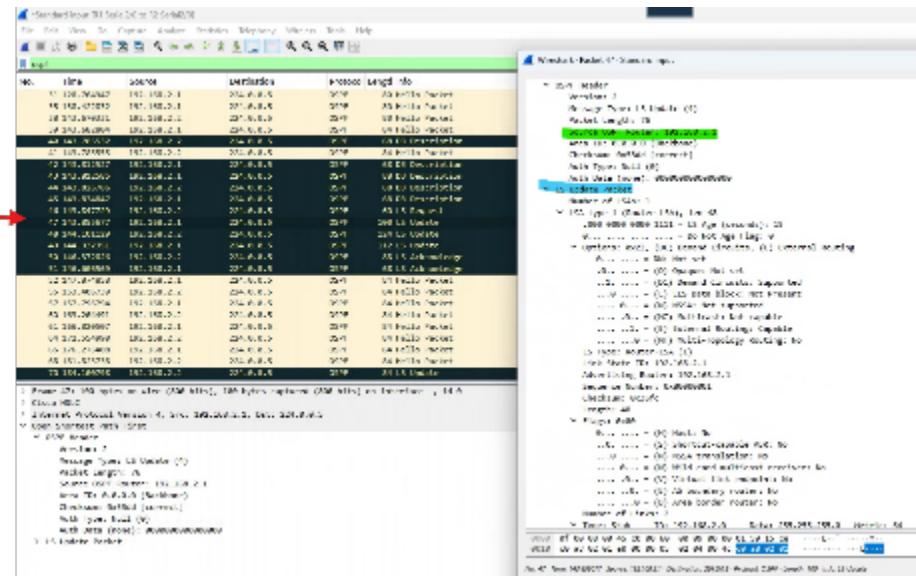
DB description message



LS request



LS UPDATE



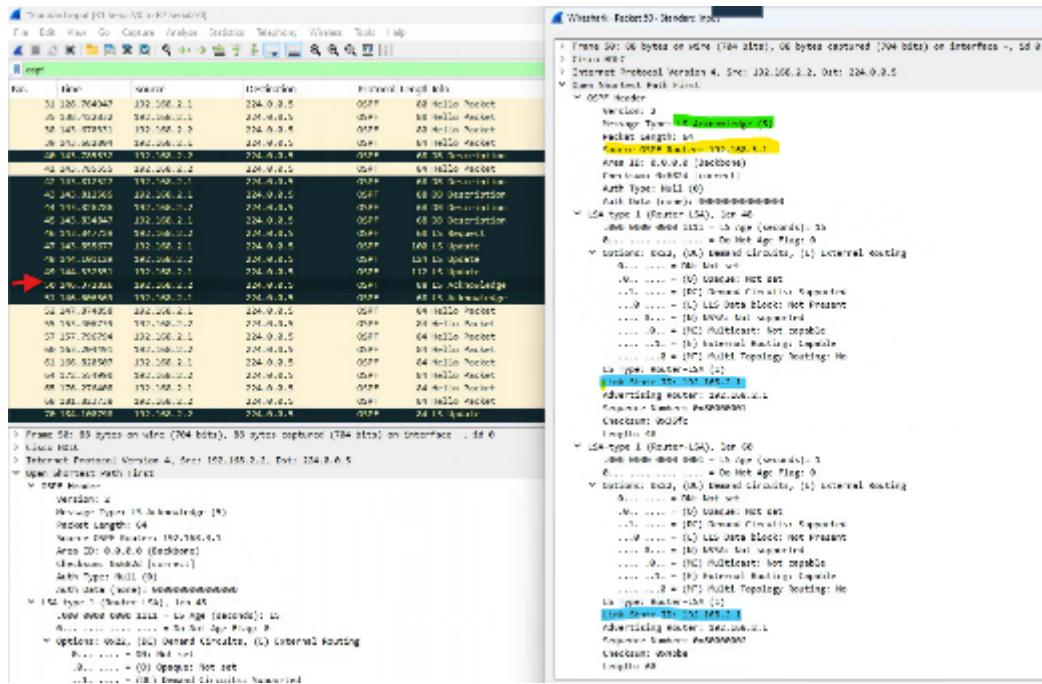
LSupdate cont.

```

.... .0. = (E) AS boundary router: No
.... .0 = (B) Area border router: No
Number of Links: 2
  ✓ Type: Stub ID: 192.168.2.0 Data: 255.255.255.0 Metric: 64
    Link ID: 192.168.2.0 - IP network/subnet number
    Link Data: 255.255.255.0
    Link Type: 3 - Connection to a stub network
    Number of Metrics: 0 - TOS
    0 Metric: 64
  ✓ Type: Stub ID: 192.168.1.0 Data: 255.255.255.0 Metric: 1
    Link ID: 192.168.1.0 - IP network/subnet number
    Link Data: 255.255.255.0
    Link Type: 3 - Connection to a stub network
    Number of Metrics: 0 - TOS
    0 Metric: 1

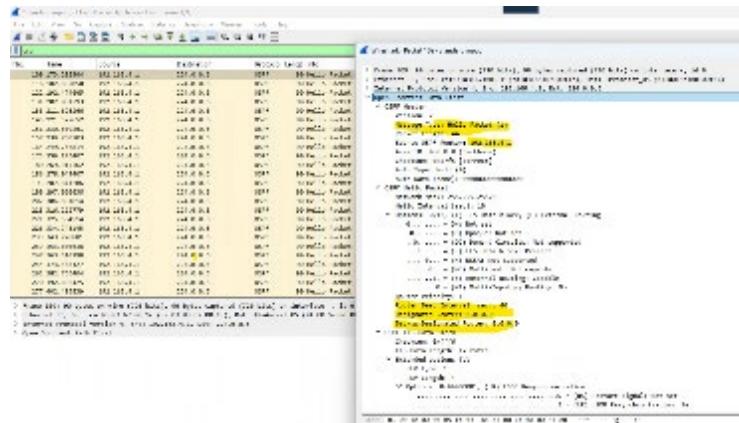
```

LS Acknowledge

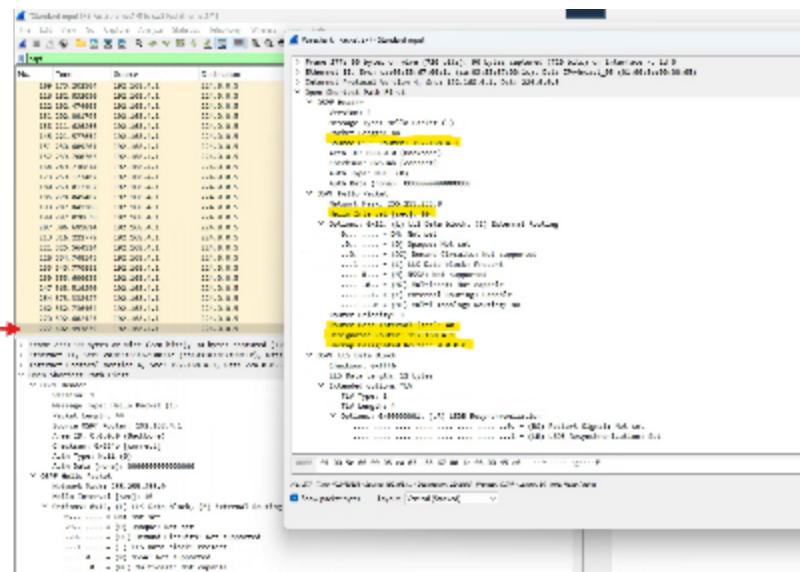


R3 to SW2

Hello Packet

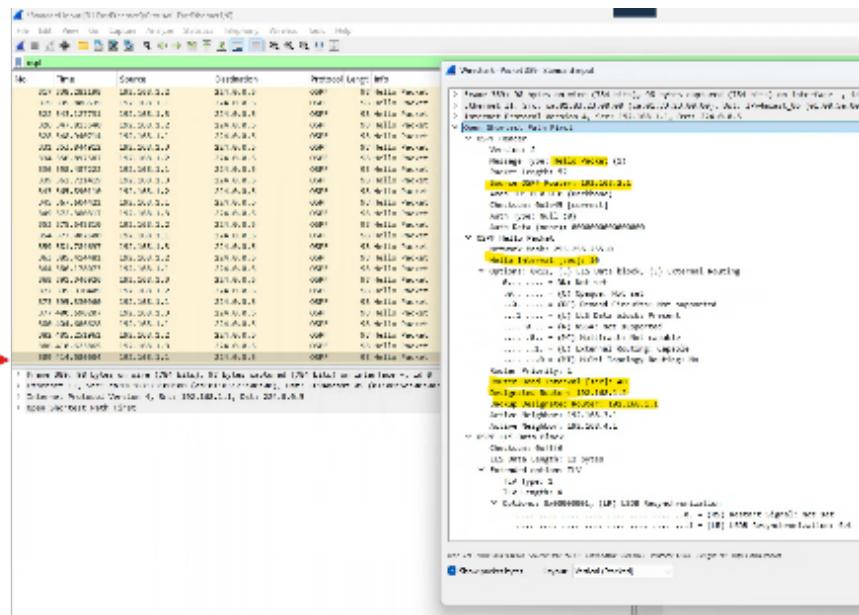


Hello Packet

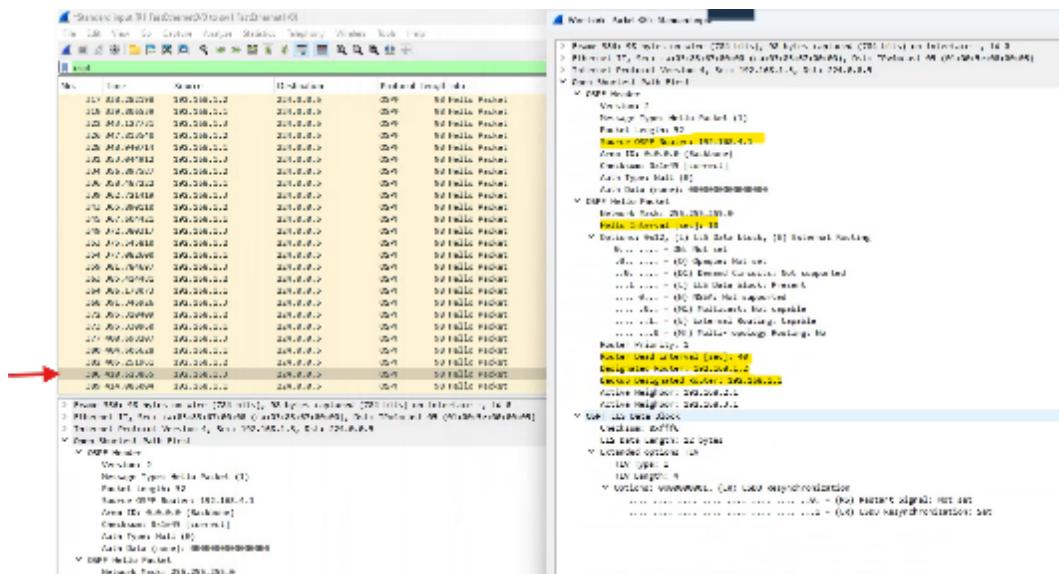


R1 to SW1

Hello Packet

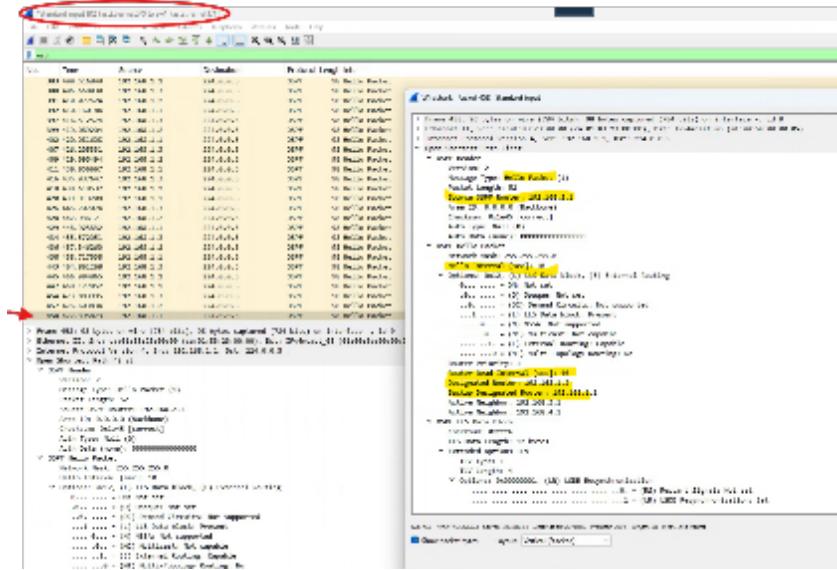


Hello Packet



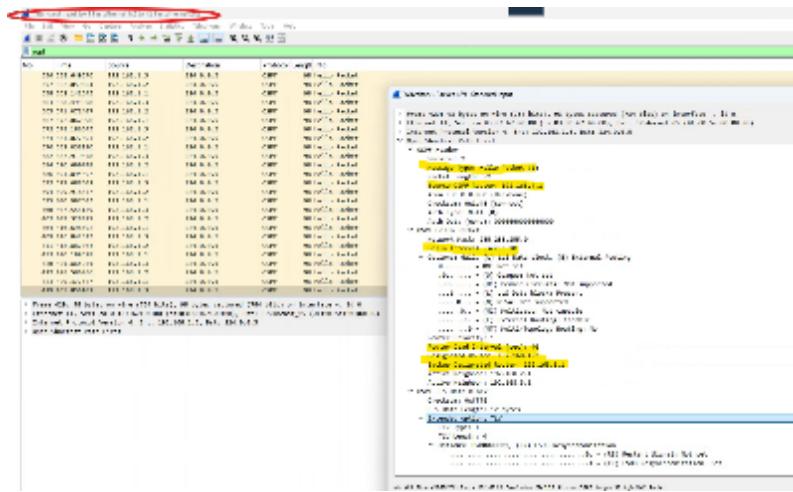
R2 to SW1

Hello Packet



SW1 to R3

Hello Packet



As we can see the DR and BDR are automatically configured to determine which node takes the role.

3.3.1.7 Giving loop back address to R2

A loopback address is a special IP address designated for testing and diagnostics on a network device like a router or a computer. It helps verify the internal workings of the device's IP stack without needing to access the external network.

IP Address 192.168.100.100 is given to be the highest value IP in the network and to be selected as DR.

```
!! loopback address R2
enable
conf t
interface lo0
description loopback address R2
ip address 192.168.100.100 255.255.255.255
end
copy running-config startup-config
```

Test loopback address from R2

```
ping 192.168.100.100
```

```
R2#
R2#!! loopback address R2
R2#enable
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#interface lo0
R2(config-if)# description loopback address R2
R2(config-if)# ip address 192.168.100.100 255.255.255.255
R2(config-if)#end
R2#
*Feb 24 22:17:34.851: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R2#
*Feb 24 22:17:35.307: %SYS-5-CONFIG_I: Configured from console by console
R2#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R2#
R2#ping 192.168.100.100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.100.100, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/8 ms
R2#
```

3.3.1.8 Printouts before change of DR BDR

Print router neighbours and identify DR and BDR.

R1

```
show history
R1#show ip ospf neighbor

Neighbor ID      Pri   State          Dead Time    Address      Interface
192.168.3.1      0     FULL/ -        00:00:34    192.168.2.2    Serial2/0
192.168.3.1      1     - FULL/DR      00:00:32    192.168.1.2    FastEthernet0/0
192.168.4.1      1     FULL/DROTHER  00:00:35    192.168.1.3    FastEthernet0/0
R1#
```

- R1 has established full OSPF adjacency with three neighbors.

- The neighbor with ID 192.168.100.100 (Priority 0) is not the DR or BDR on the Serial2/0 interface.
- The neighbor with ID 192.168.4.1 (Priority 1) is the Backup Designated Router (BDR) on the FastEthernet0/0 interface.
- The neighbor with ID 192.168.100.100 (Priority 1) is the Designated Router (DR) on the FastEthernet0/0 interface.

R2

```
show history
R2#show ip ospf neighbor

Neighbor ID      Pri  State          Dead Time   Address       Interface
192.168.4.1      0    FULL/         -           00:00:39   192.168.3.2   Serial2/1
192.168.2.1      0    FULL/         -           00:00:32   192.168.2.1   Serial2/0
192.168.2.1      1    FULL/BDR     -           00:00:32   192.168.1.1   FastEthernet0/0
192.168.4.1      1    FULL/DROTHER -           00:00:32   192.168.1.3   FastEthernet0/0
R2#
```

- The router has established full OSPF adjacency with four neighbors.
- Two neighbors (192.168.4.1 and 192.168.2.1) have a priority of 0, meaning they cannot be elected as DR or BDR.
- The neighbor with ID 192.168.4.1 (Priority 1) is the Backup Designated Router (BDR) on the FastEthernet0/0 interface.
- The neighbor with ID 192.168.2.1 (Priority 1) is a DROTHER on the FastEthernet0/0 interface.

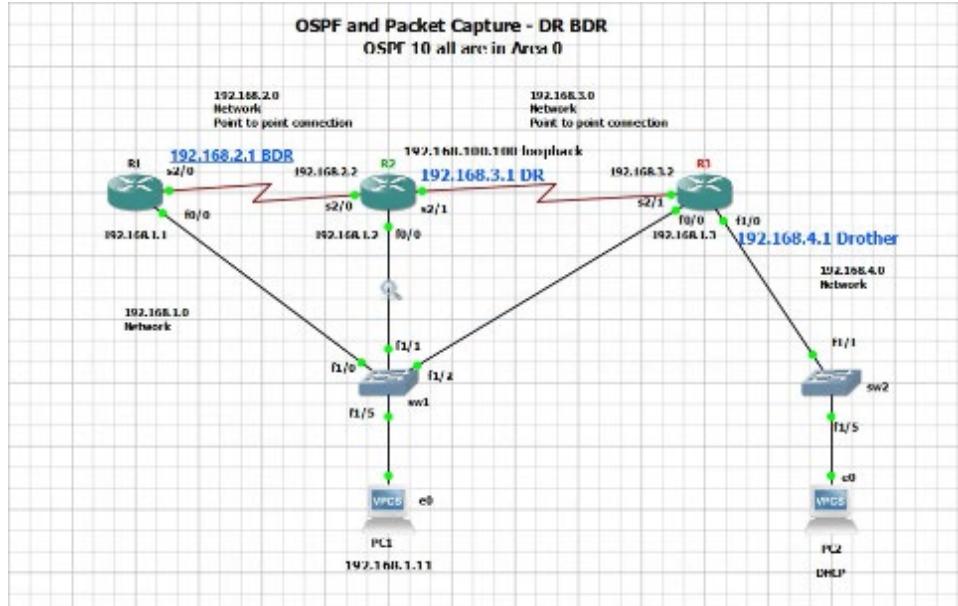
R3

```
[OK]
R3#show ip ospf neighbor

Neighbor ID      Pri  State          Dead Time   Address       Interface
192.168.3.1      0    FULL/         -           00:00:38   192.168.3.1   Serial2/1
192.168.2.1      1    FULL/BDR     -           00:00:38   192.168.1.1   FastEthernet0/0
192.168.3.1      1    FULL/DR      -           00:00:34   192.168.1.2   FastEthernet0/0
R3#
```

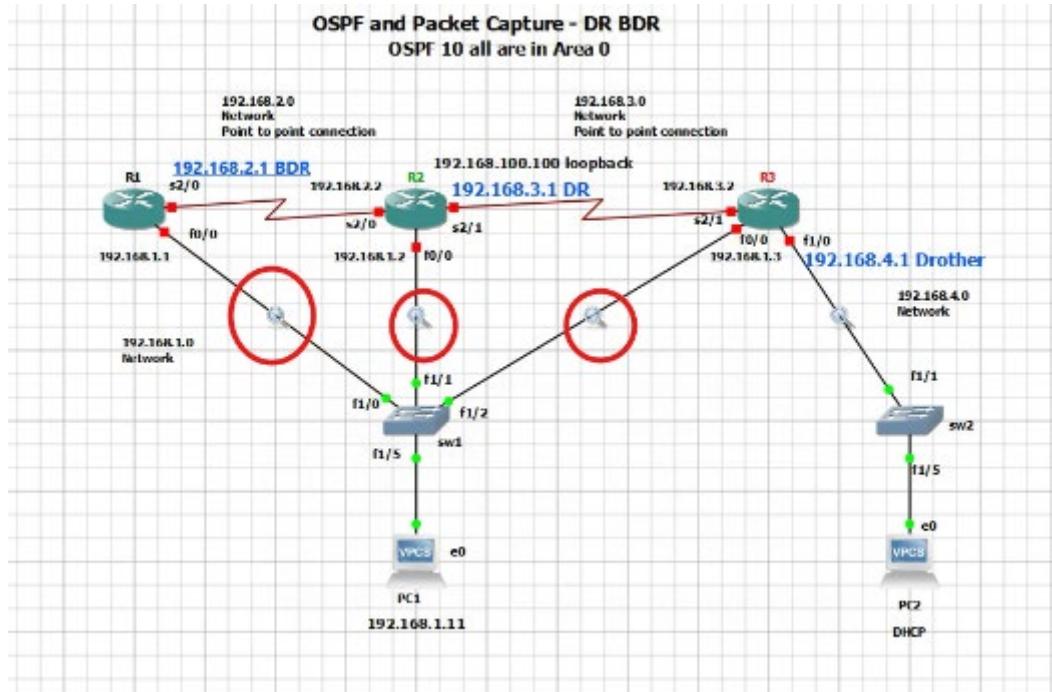
- The router has established full OSPF adjacency with four neighbors.
- Two neighbors (192.168.4.1 and 192.168.2.1) have a priority of 0, meaning they cannot be elected as DR or BDR.
- The neighbor with ID 192.168.4.1 (Priority 1) is the Backup Designated Router (BDR) on the FastEthernet0/0 interface.
- The neighbor with ID 192.168.2.1 (Priority 1) is a DROTHER on the FastEthernet0/0 interface.

Currently as per printouts the DR,BDR and Drother and loopback IP has been assigned.

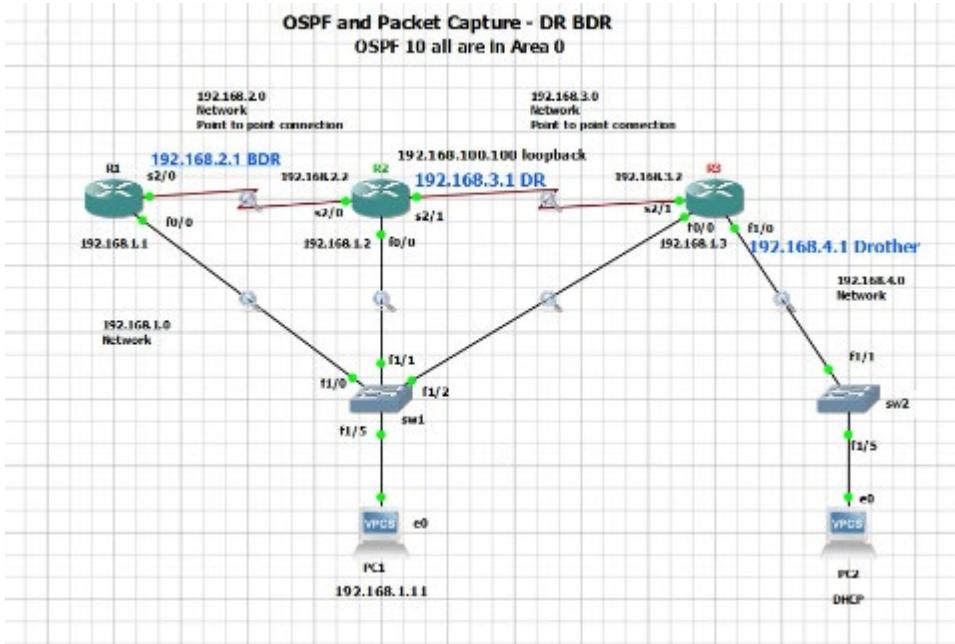


3.3.1.9 Change of DR and BDR

Power off the routers and start Wireshark



Power on the routers



Wait until it finishes powering up and synchronizing, this might take time.

3.3.1.10 Wireshark analysis loop back

See in Wireshark **224.0.0.6** messages appearing

224.0.0.6: Used to send messages specifically to the DR and BDR, when this address appears it means DR and BDR had been selected.

No.	Time	Source	Destination	Protocol	Length	Info
449	481.015697	192.168.1.3	224.0.0.5	OSPF	98	Hello Packet
453	484.566628	192.168.1.2	224.0.0.5	OSPF	98	Hello Packet
455	486.548555	192.168.1.1	224.0.0.5	OSPF	98	Hello Packet
458	490.245828	192.168.1.3	224.0.0.5	OSPF	98	Hello Packet
148	182.291234	192.168.1.1	224.0.0.6	OSPF	122	LS Update
152	184.774932	192.168.1.1	224.0.0.6	OSPF	98	LS Acknowledge
168	191.817218	192.168.1.1	224.0.0.6	OSPF	134	LS Update
171	191.849688	192.168.1.1	224.0.0.6	OSPF	98	LS Update
175	194.315873	192.168.1.1	224.0.0.6	OSPF	98	LS Acknowledge
461	493.988414	192.168.1.2	224.0.0.5	OSPF	98	Hello Packet
463	496.249577	192.168.1.1	224.0.0.5	OSPF	98	Hello Packet
466	499.651359	192.168.1.3	224.0.0.5	OSPF	98	Hello Packet
471	503.771122	192.168.1.2	224.0.0.5	OSPF	98	Hello Packet
474	506.231942	192.168.1.1	224.0.0.5	OSPF	98	Hello Packet
476	508.893538	192.168.1.3	224.0.0.5	OSPF	98	Hello Packet
480	513.481122	192.168.1.2	224.0.0.5	OSPF	98	Hello Packet
483	515.480685	192.168.1.1	224.0.0.5	OSPF	98	Hello Packet
486	518.712668	192.168.1.3	224.0.0.5	OSPF	98	Hello Packet

See the Hello Message after 224.0.0.6 shows DR and BDR

The figure displays two side-by-side network captures in Wireshark:

- Left Capture (Standard Input):** Shows standard Ethernet frames (Type: IEEE 802.3). It includes a list of 16 captured frames, a packet details pane, a bytes pane, and a timeline pane.
- Right Capture (Wi-Fi Standard Input):** Shows IEEE 802.11 wireless frames (Type: IEEE 802.11). It includes a list of 16 captured frames, a packet details pane, a bytes pane, and a timeline pane.

Both captures have their "Selected" column checked for the first frame. The right capture's details pane shows specific fields for IEEE 802.11 frames, such as "Frame Delimiter", "SSID", "BSSID", "Source MAC", "Destination MAC", "Protocol", "Length/Info", and "Auth Type". The right capture's timeline pane shows the sequence of frames over time.

Make sure to stop Wireshark captures.

3.3.1.11 Printouts OSPF

Printing ospf neighbour

show ip ospf neighbor

R1

```
R1#show ip ospf neighbor
R1#show ip ospf neighbor

Neighbor ID      Pri  State        Dead Time    Address          Interface
192.168.100.100  0    FULL/ -       00:00:38    192.168.2.2    Serial2/0
192.168.4.1      1    FULL/BDR     00:00:31    192.168.1.3    FastEthernet0/0
192.168.100.100  1    FULL/DR      00:00:37    192.168.1.2    FastEthernet0/0
R1#
```

- R1 has established full OSPF adjacency with three neighbors.
- The neighbor with ID 192.168.100.100 (Priority 0) is not the DR or BDR on the Serial2/0 interface.
- The neighbor with ID 192.168.4.1 (Priority 1) is the Backup Designated Router (BDR) on the FastEthernet0/0 interface.
- The neighbor with ID 192.168.100.100 (Priority 1) is the Designated Router (DR) on the FastEthernet0/0 interface.

R2

```
R2#show ip ospf neighbor
R2#show ip ospf neighbor

Neighbor ID      Pri  State        Dead Time    Address          Interface
192.168.4.1      0    FULL/ -       00:00:33    192.168.3.2    Serial2/1
192.168.2.1      0    FULL/ -       00:00:33    192.168.2.1    Serial2/0
192.168.2.1      1    FULL/DROTHER  00:00:33    192.168.1.1    FastEthernet0/0
192.168.4.1      1    FULL/BDR     00:00:34    192.168.1.3    FastEthernet0/0
R2#
```

- R2 has established full OSPF adjacency with four neighbors.
- The neighbor with ID 192.168.2.1 (Priority 0) is not the DR or BDR.
- The neighbor with ID 192.168.2.1 (Priority 1) is a DROTHER.
- The neighbor with ID 192.168.4.1 (Priority 1) is the Backup Designated Router (BDR) on the FastEthernet0/0 interface.

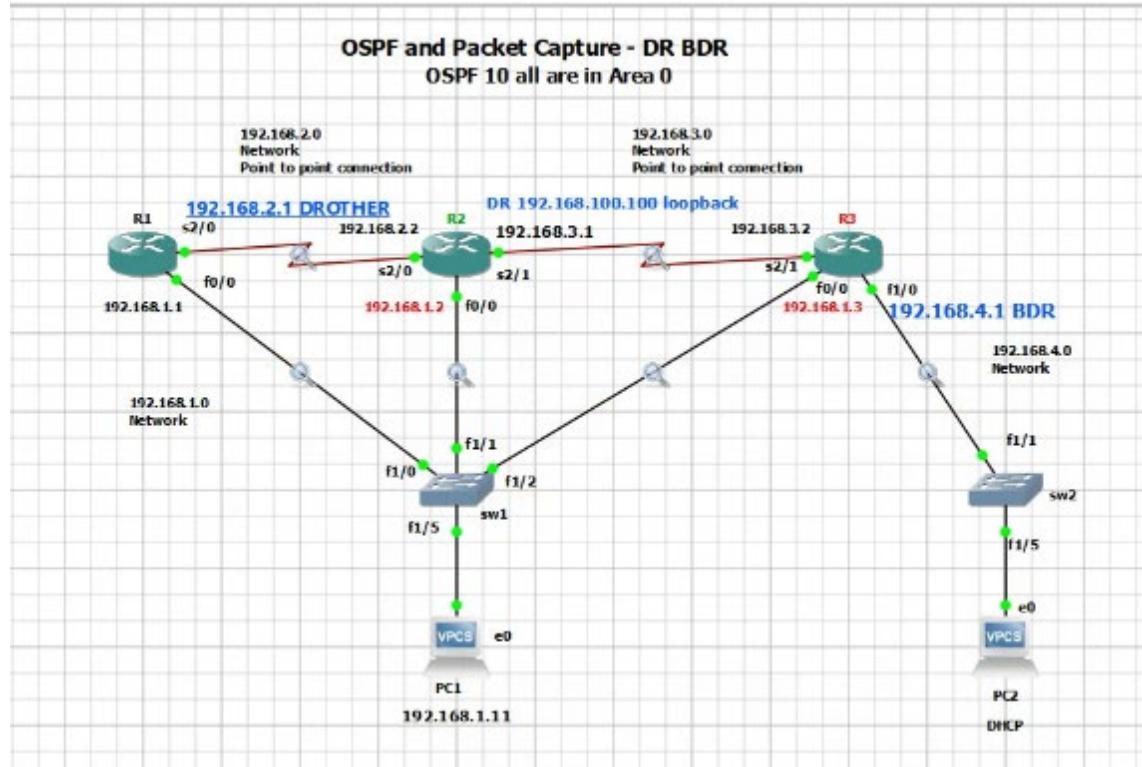
R3

```
R3#show ip ospf neighbor
R3#show ip ospf neighbor

Neighbor ID      Pri  State        Dead Time    Address          Interface
192.168.100.100  0    FULL/ -       00:00:33    192.168.3.1    Serial2/1
192.168.2.1      1    FULL/DROTHER  00:00:38    192.168.1.1    FastEthernet0/0
192.168.100.100  1    FULL/DR      00:00:35    192.168.1.2    FastEthernet0/0
R3#
```

- R3 has established **full** OSPF adjacency with three neighbors.
- One neighbor (192.168.100.100) is connected via both Serial2/1 and FastEthernet0/0.
- The neighbor with ID 192.168.2.1 (Priority 1) is not the DR or BDR.
- The neighbor with ID 192.168.100.100 (Priority 1) is the DR on the FastEthernet0/0 interface.

Show R2 is DR is highest loop back address



3.3.2 OSPF add Fedora VM with Apache SSH services – Access Control Lists and NAT

3.3.2.1 Preparation install Fedora with SSH and Apache

Install a new Fedora machine and add Apache and SSH services 3.1.1 Fedora

Apply workaround to avoid black screen before doing update/upgrade 3.2.2 Fedora fix to avoid black screen.

3.3.2.1.1 Install ssh and Apache services

```
dnf install httpd -y  
dnf install openssh-server  
systemctl enable httpd
```

```
systemctl enable sshd  
systemctl start httpd  
systemctl status httpd  
systemctl start sshd  
systemctl status sshd
```

Start enable sshd

```

root@fedora:/etc# dnf install openssh-server
Updating and loading repositories...
Repositories loaded.
Package "openssh-server-8.9pl1-3.fc41.x86_64" is already installed.

Nothing to do.

root@fedora:/etc# systemctl start sshd
root@fedora:/etc# systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; disabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/sshd.service.d
             └─ 10-timeout-abort.conf, 50-keep-wake.conf
     Active: active (running) since Wed 2025-02-26 10:22:46 EST; 8s ago
   Invocation: efbd0e4cd28e4d08ba4af42f88cabb
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 196473 (sshd)
      Tasks: 1 (limit: 9415)
        Memory: 1.4M (peak: 1.6M)
         CPU: 9ms
        CGroup: /system.slice/sshd.service
                  └─196473 "sshd: /usr/sbin/sshd -D [listener] 8 of 16-160 startups"

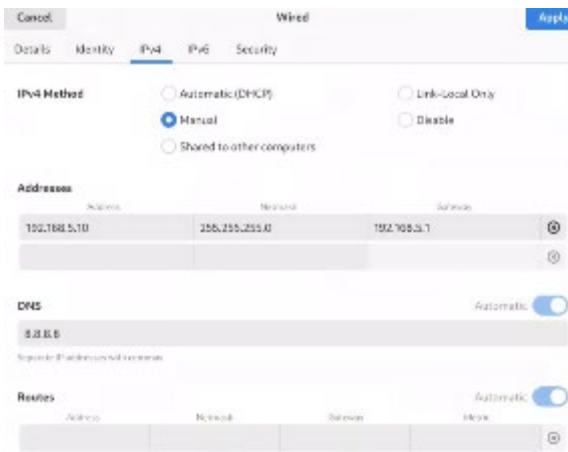
Feb 26 10:22:45 fedora systemd[1]: Starting sshd.service - OpenSSH server daemon...
Feb 26 10:22:46 fedora sshd[196473]: Server listening on 0.0.0.0 port 22.
Feb 26 10:22:46 fedora sshd[196473]: Server listening on :: port 22.
Feb 26 10:22:46 fedora systemd[1]: Started sshd.service - OpenSSH server daemon.
root@fedora:/etc# systemctl enable sshd
Created symlink '/etc/systemd/system/multi-user.target.wants/sshd.service' → '/usr/lib/systemd/system/sshd.service'.
root@fedora:/etc# systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/sshd.service.d
             └─ 10-timeout-abort.conf, 50-keep-wake.conf
     Active: active (running) since Wed 2025-02-26 10:22:46 EST; 23s ago
   Invocation: efbd0e4cd28e4d08ba4af42f88cabb
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 196473 (sshd)
      Tasks: 1 (limit: 9415)
        Memory: 1.4M (peak: 1.6M)
         CPU: 9ms
        CGroup: /system.slice/sshd.service
                  └─196473 "sshd: /usr/sbin/sshd -D [listener] 8 of 16-160 startups"

Feb 26 10:22:45 fedora systemd[1]: Starting sshd.service - OpenSSH server daemon...
Feb 26 10:22:46 fedora sshd[196473]: Server listening on 0.0.0.0 port 22.
Feb 26 10:22:46 fedora sshd[196473]: Server listening on :: port 22.
Feb 26 10:22:46 fedora systemd[1]: Started sshd.service - OpenSSH server daemon.
root@fedora:/etc# 

```

3.3.2.2 Fedora IP address

Change ip address in Fedora to static ip 192.168.5.10



Verify IP address

nmcli device show

```
10 Error: argument 'show' not understood. Try passing --help instead.
11 root@fedoralab:/home/student# nmcli device show
12 GENERAL.DEVICE:                         ens35
13 GENERAL.TYPE:                           ethernet
14 GENERAL.HWADDR:                         80:0C:29:6A:A6:90
15 GENERAL.MTU:                            1500
16 GENERAL.STATE:                          100 (connected)
17 GENERAL.CONNECTION:                     Wired connection 2
18 GENERAL.CON-PATH:                       /org/freedesktop/NetworkManager/ActiveConnection/2
19
20 IP4.ADDRESS[1]:                         192.168.5.10/24
21           IP4.GATEWAY:                      192.168.5.1
22           IP4.ROUTE[1]:                    dst = 0.0.0.0/8, nh = 192.168.5.1, mt = 26180
23           IP4.ROUTE[2]:                    dst = 192.168.5.0/24, nh = 0.0.0.0, mt = 108
24             IP4.DNS[1]:                   8.8.8.8
25           IP6.ADDRESS[1]:                  fe80::86a6:f22a:ee81:7e7b/64
26           IP6.GATEWAY:                     --
27           IP6.ROUTE[1]:                   dst = fe80::/64, nh = ::, mt = 1024
28
29 GENERAL.DEVICE:                         lo
30 GENERAL.TYPE:                           loopback
31 GENERAL.HWADDR:                         00:09:80:00:00:00
32 GENERAL.MTU:                            65535
33 GENERAL.STATE:                          100 (connected (externally))
34 GENERAL.CONNECTION:                     /org/freedesktop/NetworkManager/ActiveConnection/1
35 IP4.ADDRESS[1]:                         127.0.0.1/8
36 IP4.GATEWAY:                           --
37 IP6.ADDRESS[1]:                         ::1/128
38 IP6.GATEWAY:                           --
39
40 root@fedoralab:/home/student#
```

← mouse pointer: inside or press Ctrl+Q.

3.3.2.3 SSH user

Add user asmith/asmith

```
useradd asmith
pass asmith
```

Change user to asmith and connect via ssh

```
su asmith
ssh 192.168.5.10
## ssh is working
```

Create user asmith and test ssh connection

```
root@fedora:/etc# useradd asmith
useradd: user 'asmith' already exists
root@fedora:/etc# passwd asmith
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: password updated successfully
root@fedora:/etc#
root@fedora:/etc# su asmith
asmith@fedora:~/etc$ ssh 10.164.101.105
The authenticity of host '10.164.101.105 (10.164.101.105)' can't be established.
ED25519 key fingerprint is SHA256:btfZDDDx4TEPDV3b7hghibNOf4D8BEgVNRxNoJfNwuo.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.164.101.105' (ED25519) to the list of known hosts.
asmith@10.164.101.105's password:
Last login: Wed Feb 26 10:26:10 2025
asmith@fedora:~$
```

Get back to root

3.3.2.4 Install lynx

lynx 192.168.5.10

```
exit
root@fedora:/etc# lynx 192.168.5.10
bash: lynx: command not found...
Install package 'lynx' to provide command 'lynx'? [N/y] y

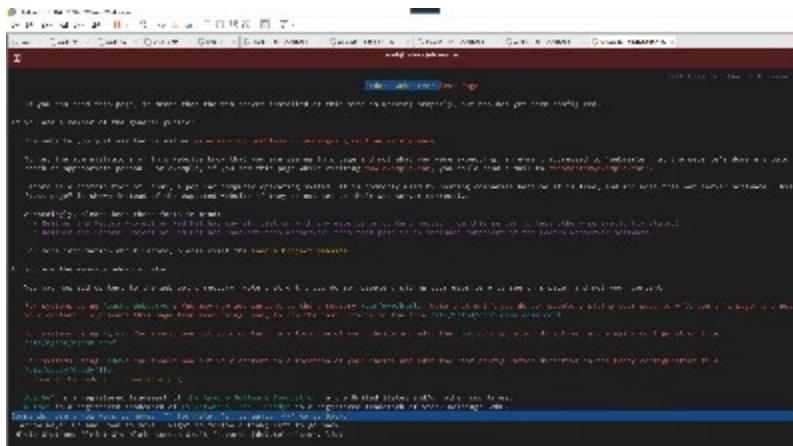
* Waiting in queue...
* Loading list of packages....
The following packages have to be installed:
lynx-2.9.2-2.fc41.x86_64          A text-based Web browser
Proceed with changes? [N/y] y

* Waiting in queue...
* Waiting for authentication...
* Waiting in queue...
* Downloading packages...
* Requesting data...
* Testing changes...
* Installing packages...

Exiting via interrupt: 2

root@fedora:/etc#
```

To close input to this VM, move the mouse pointer inside or press Ctrl+G.



Go out with Ctrl-x

3.3.2.5 Setup web page

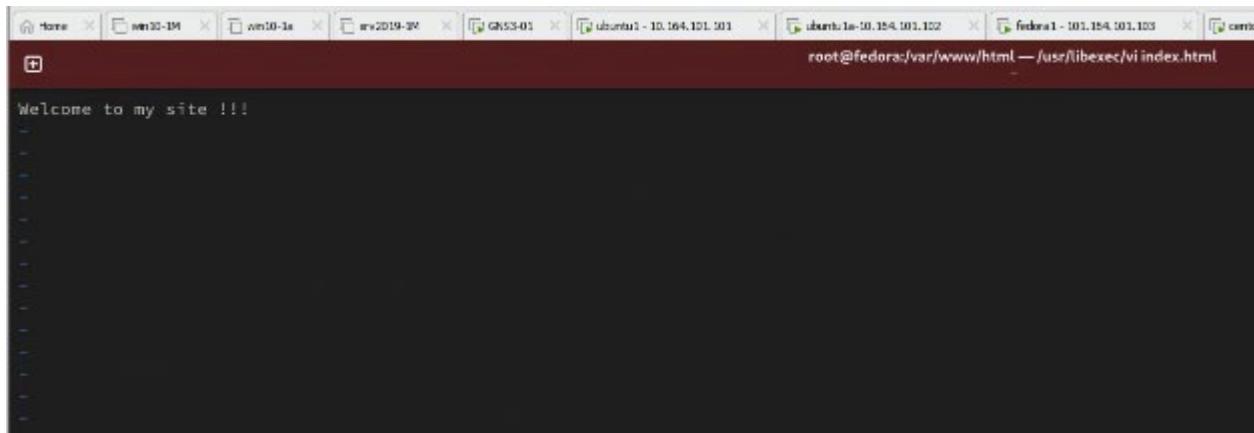
Create a test page

```
cd /var/www/html  
vi /index.html
```

```
root@fedora:/etc# cd /var/www/html  
root@fedora:/var/www/html# vi index.html  
root@fedora:/var/www/html#
```

To return to your computer, move the mouse pointer outside or press Ctrl+Alt.

Write Welcome to my site, save index.html file

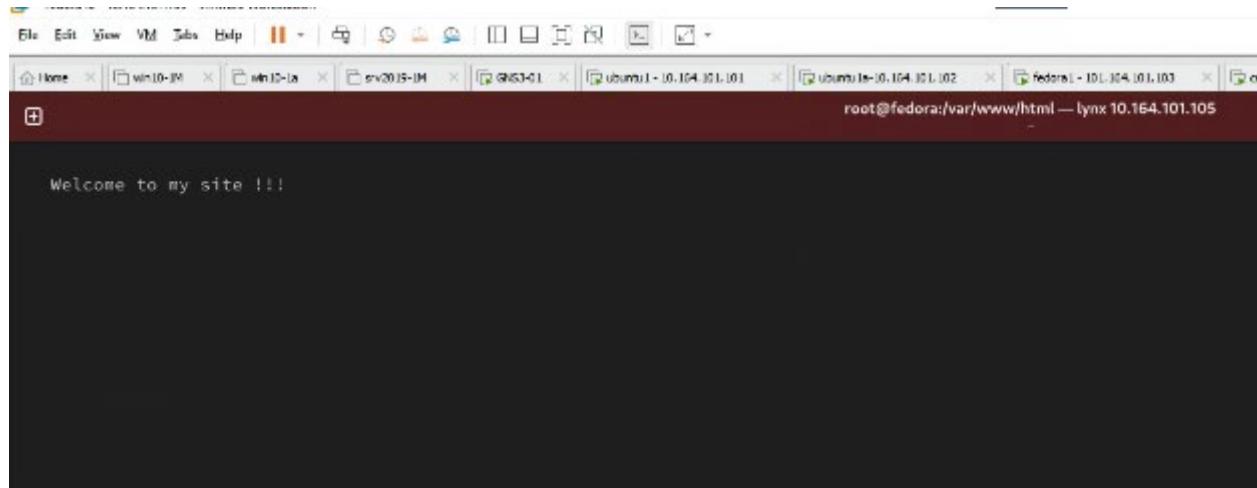


3.3.2.5.1 Test web

lynx 10.164.0.50

```
root@fedora:/etc# cd /var/www/html  
root@fedora:/var/www/html# vi index.html  
root@fedora:/var/www/html# lynx 10.164.101.105  
root@fedora:/var/www/html#
```

To return to your computer, move the mouse pointer outside or press Ctrl+Alt.



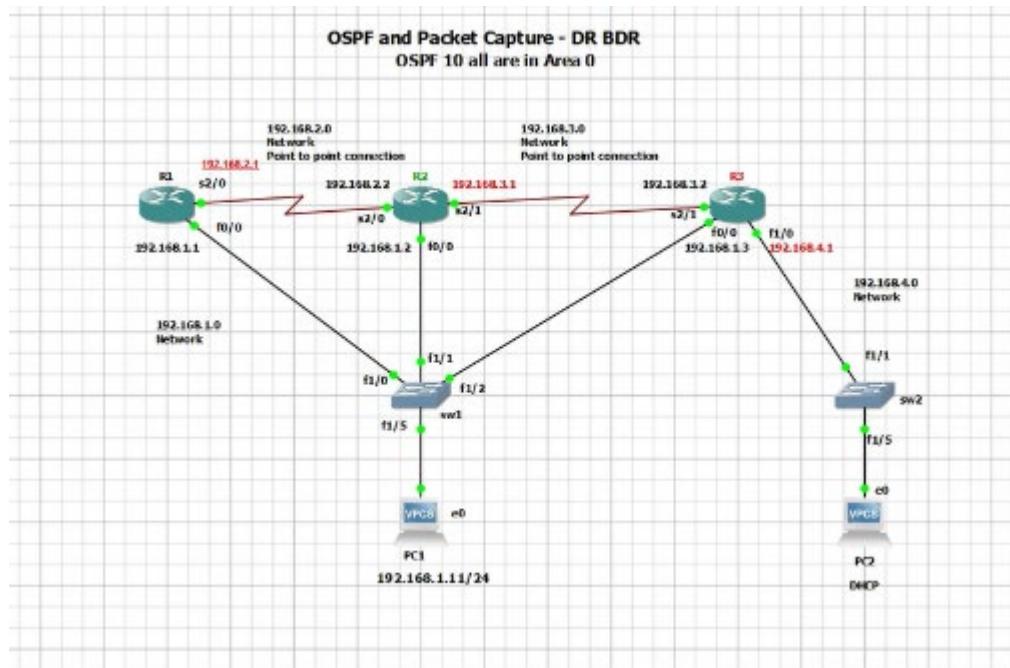
3.3.2.6 GNS3

Open up gns3

3.3.2.6.1 Topology

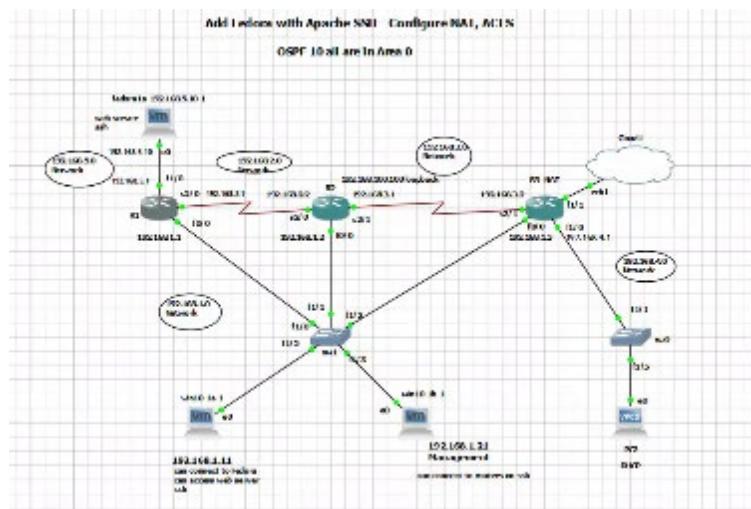
Open project OSPF

From this configuration



1. Add Fedora machine with IP 192.168.5.10/24 Gateway 192.168.5.1
2. Connect Fedora eth0 to R1 f1/0
3. Change PC1 and use win10-1a machine from Vmware
4. Change PC2 and use wun10-1b-1 machine from Vmware
5. Add cloud - From R3 add Cloud using f1/1 to connect to cloud

Resulting configuration



3.3.2.6.2 Addressing table

Network Element	Connection	Port	IP Address
R1	Connection to R2	S2/0	192.168.2.1/24
	Connection to SW1	F0/0	192.168.1.1/24
	Connection to Fedora1a	F1/0	192.168.5.1/24
R2	Connection to R1	S2/0	192.168.2.2/24
	Connection to R3	S2/1	192.168.3.1/24
	Connection to SW1	F0/0	192.168.1.2/24
R3	Connection to R2	S2/1	192.168.3.2/24
	Connection to SW1	F0/0	192.168.1.3/24
	Connection to SW2	F1/0	192.168.4.1/24
	Connection to Cloud	F1/1	DHCP
SW1	Connection to R1	F1/0	N/A
	Connection to R2	F1/1	N/A
	Connection to R3	F1/2	N/A
	Connection to win10-1a-1	F1/5	N/A
	Connection to win10-1b -1 Management	F1/3	N/A
SW2	Connection to R3	F1/0	N/A
	Connection to PC2	F1/5	N/A

PC's

win10-1a-1	Connection to SW1	NIC	192.168.1.11/24
win10-1b-1	Connection to SW1	NIC	192.168.1.21/24
PC2	Connection to SW2	NIC	DHCP-assigned

3.3.2.6.3 Add Fedora

3.3.2.6.3.1 R1 changes to add Fedora

Scripts to add new elements

```
enable
config t
```

```

int f1/0
description Connection between R1 and Fedora
ip address 192.168.5.1 255.255.255.0
no shutdown
exit

```

```

R1#enable
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#int f1/0
R1(config-if)# description Connection between R1 and Fedora
R1(config-if)# ip address 192.168.5.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)#exit
R1(config)#
R1(config)#
*Feb 28 03:04:24.275: %LINK-3-UPDOWN: Interface FastEthernet1/0, changed state to up
R1(config)#
*Feb 28 03:04:25.275: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/0, changed state to up
R1(config)#

```

Verify connection

```

R1#show ip interface brief
Interface          IP-Address      OK? Method Status           Protocol
FastEthernet0/0    192.168.1.1    YES NVRAM  up              up
FastEthernet1/0    192.168.5.1    YES manual  up              up
FastEthernet1/1    unassigned     YES NVRAM  administratively down  down
Serial2/0          192.168.2.1    YES NVRAM  up              up
Serial2/1          unassigned     YES NVRAM  administratively down  down
Serial2/2          unassigned     YES NVRAM  administratively down  down
Serial2/3          unassigned     YES NVRAM  administratively down  down
R1#

```

3.3.2.6.3.2 Test pings to Fedora

Test ping from R1

```

R1#ping 192.168.5.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.5.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/28/36 ms
R1#

```

3.3.2.6.3.3 Test ping in Fedora

Show ip in Fedora

```
error: argument '-r /etc/networks' has passing -f 'http://localhost:  
root@fedorate:~# rmcli  
ens35: connected to Mirred connection 2  
    "Intel PRO/100 MT"  
    ethernet (wired), 00:0C:29:84:48:90, hw, mtu 1500  
    ifp default  
    smtu 1500  
    route4 192.168.5.0/24 metric 20166  
    route4 192.168.5.0/24 metric 188  
    link fe80::6c2ff:fe84:91ff%eth0 brd ff:  
    route6 fe80::/64 metric 1824  
  
lo: connected (externally) to lo  
    "loop"  
    loopback (unknown), 00:00:00:00:00:00, sw, mtu 65536  
    inette 127.0.0.1/8  
    inette ::1/128  
  
NMS configuration:  
    servers 8.8.8.8  
    interfaces ens35  
  
Use 'rmcli device show' to get complete information about known devices and  
'rmcli connection show' to get an overview on active connection profiles.  
  
Consult rmcli(1) and rmcli-examples(7) manual pages for complete usage details.  
root@fedorate:~#
```

3.3.2.6.3.4 Test ping from Management PC

```
C:\Users\student>ipconfig
Windows IP Configuration

Ethernet adapter Ethernet0 2:

Connection-specific DNS Suffix  . :
Link-local IPv6 Address . . . . . : fe80::5113:86dc:b56b:5890%21
IPv4 Address. . . . . : 192.168.1.21
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.1.1

C:\Users\student>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:
Reply from 192.168.1.1: bytes=32 time=23ms TTL=255
Reply from 192.168.1.1: bytes=32 time=11ms TTL=255
Reply from 192.168.1.1: bytes=32 time=14ms TTL=255
Reply from 192.168.1.1: bytes=32 time=12ms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 23ms, Average = 15ms

C:\Users\student>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:
Reply from 192.168.2.1: bytes=32 time=13ms TTL=255
Reply from 192.168.2.1: bytes=32 time=6ms TTL=255
Reply from 192.168.2.1: bytes=32 time=3ms TTL=255
Reply from 192.168.2.1: bytes=32 time=7ms TTL=255

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

```
C:\Users\student>ping 192.168.3.1

Pinging 192.168.3.1 with 32 bytes of data:
Request timed out.
Reply from 192.168.3.1: bytes=32 time=32ms TTL=255
Reply from 192.168.3.1: bytes=32 time=23ms TTL=255
Reply from 192.168.3.1: bytes=32 time=32ms TTL=255

Ping statistics for 192.168.3.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 23ms, Maximum = 32ms, Average = 29ms

C:\Users\student>ping 192.168.4.1

Pinging 192.168.4.1 with 32 bytes of data:
Request timed out.
Reply from 192.168.4.1: bytes=32 time=22ms TTL=255
Reply from 192.168.4.1: bytes=32 time=23ms TTL=255
Reply from 192.168.4.1: bytes=32 time=27ms TTL=255

Ping statistics for 192.168.4.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 22ms, Maximum = 27ms, Average = 24ms

C:\Users\student>ping 192.168.5.1

Pinging 192.168.5.1 with 32 bytes of data:
Reply from 192.168.5.1: bytes=32 time=9ms TTL=255
Reply from 192.168.5.1: bytes=32 time=14ms TTL=255
Reply from 192.168.5.1: bytes=32 time=12ms TTL=255
Reply from 192.168.5.1: bytes=32 time=14ms TTL=255

Ping statistics for 192.168.5.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
```

Ping Fedora from Management PC

```
C:\Users\student>ping 192.168.5.10

Pinging 192.168.5.10 with 32 bytes of data:
Reply from 192.168.5.10: bytes=32 time=24ms TTL=63
Reply from 192.168.5.10: bytes=32 time=17ms TTL=63
Reply from 192.168.5.10: bytes=32 time=17ms TTL=63
Reply from 192.168.5.10: bytes=32 time=14ms TTL=63

Ping statistics for 192.168.5.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 14ms, Maximum = 24ms, Average = 18ms

C:\Users\student>
```

Ping PC1

```
C:\Users\student>ping 192.168.1.11

Pinging 192.168.1.11 with 32 bytes of data:
Reply from 192.168.1.11: bytes=32 time=3ms TTL=128
Reply from 192.168.1.11: bytes=32 time=2ms TTL=128
Reply from 192.168.1.11: bytes=32 time=1ms TTL=128
Reply from 192.168.1.11: bytes=32 time=15ms TTL=128

Ping statistics for 192.168.1.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 15ms, Average = 5ms

C:\Users\student>
```

Ping PC2

```
C:\Users\student>ping 192.168.4.11

Pinging 192.168.4.11 with 32 bytes of data:
Reply from 192.168.4.11: bytes=32 time=43ms TTL=63
Reply from 192.168.4.11: bytes=32 time=25ms TTL=63
Reply from 192.168.4.11: bytes=32 time=46ms TTL=63
Reply from 192.168.4.11: bytes=32 time=29ms TTL=63

Ping statistics for 192.168.4.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 25ms, Maximum = 46ms, Average = 35ms

C:\Users\student>
```

Test win10a connectivity

```
C:\Users\Administrator
C:\Users\Administrator>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet2 2:

  Connection-specific DNS Suffix  . :
  Link-local IPv6 Address . . . . . : fe80::ecbe:9bd1%5c0:8e21%5
  IPv4 Address . . . . . : 192.168.1.11
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 192.168.1.1

C:\Users\Administrator>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:
Reply from 192.168.1.1: bytes=32 time=8ms TTL=255
Reply from 192.168.1.1: bytes=32 time=10ms TTL=255
Reply from 192.168.1.1: bytes=32 time=9ms TTL=255
Reply from 192.168.1.1: bytes=32 time=6ms TTL=255

Ping statistics for 192.168.1.1:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milliseconds:
    Minimum = 6ms, Maximum = 10ms, Average = 8ms

C:\Users\Administrator>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:
Reply from 192.168.2.1: bytes=32 time=4ms TTL=255
Reply from 192.168.2.1: bytes=32 time=5ms TTL=255
Reply from 192.168.2.1: bytes=32 time=7ms TTL=255
Reply from 192.168.2.1: bytes=32 time=20ms TTL=255

Ping statistics for 192.168.2.1:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milliseconds:
    Minimum = 5ms, Maximum = 10ms, Average = 8ms
```

```
C:\Users\Administrator>ping 192.168.3.1

Pinging 192.168.3.1 with 32 bytes of data:
Request timed out.
Reply from 192.168.3.1: bytes=32 time=8ms TTL=255
Reply from 192.168.3.1: bytes=32 time=6ms TTL=255
Reply from 192.168.3.1: bytes=32 time=9ms TTL=255

Ping statistics for 192.168.3.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 6ms, Maximum = 9ms, Average = 7ms

C:\Users\Administrator>ping 192.168.4.1

Pinging 192.168.4.1 with 32 bytes of data:
Request timed out.
Reply from 192.168.4.1: bytes=32 time=10ms TTL=255
Reply from 192.168.4.1: bytes=32 time=8ms TTL=255
Reply from 192.168.4.1: bytes=32 time=5ms TTL=255

Ping statistics for 192.168.4.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 5ms, Maximum = 10ms, Average = 7ms

C:\Users\Administrator>
C:\Users\Administrator>
C:\Users\Administrator>ping 192.168.5.1

Pinging 192.168.5.1 with 32 bytes of data:
Reply from 192.168.5.1: bytes=32 time=15ms TTL=255
Reply from 192.168.5.1: bytes=32 time=3ms TTL=255
Reply from 192.168.5.1: bytes=32 time=26ms TTL=255
Reply from 192.168.5.1: bytes=32 time=7ms TTL=255

Ping statistics for 192.168.5.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 3ms, Maximum = 26ms, Average = 12ms
```

Test Fedora ping

```
C:\Users\Administrator>ping 192.168.5.10

Pinging 192.168.5.10 with 32 bytes of data:
Reply from 192.168.5.10: bytes=32 time=13ms TTL=63
Reply from 192.168.5.10: bytes=32 time=15ms TTL=63
Reply from 192.168.5.10: bytes=32 time=17ms TTL=63
Reply from 192.168.5.10: bytes=32 time=20ms TTL=63

Ping statistics for 192.168.5.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 13ms, Maximum = 20ms, Average = 16ms

C:\Users\Administrator>
```

```
C:\Users\Administrator>
C:\Users\Administrator>ping 192.168.4.11

Pinging 192.168.4.11 with 32 bytes of data:
Reply from 192.168.4.11: bytes=32 time=39ms TTL=63
Reply from 192.168.4.11: bytes=32 time=22ms TTL=63
Reply from 192.168.4.11: bytes=32 time=17ms TTL=63
Reply from 192.168.4.11: bytes=32 time=23ms TTL=63

Ping statistics for 192.168.4.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 17ms, Maximum = 39ms, Average = 25ms

C:\Users\Administrator>
```

```
Pinging 192.168.1.21 with 32 bytes of data:
Reply from 192.168.1.21: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.21:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\student>
```

3.3.2.6.4 SSH

3.3.2.6.4.1 Test SSH to Fedora NO ACL

Test from win10-1a (PC1)

```
C:\Users\Administrator>ssh 192.168.5.10 -l asmith
The authenticity of host '192.168.5.10 (192.168.5.10)' can't be established.
ED25519 key fingerprint is SHA256:btfZDDx4TEPDV3b7hghibNOf4D8BEgVRNxNoJFnwuo.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.5.10' (ED25519) to the list of known hosts.
asmith@192.168.5.10's password:
Last login: Wed Feb 26 10:26:43 2025 from 10.164.101.105
asmith@fedora1a:~$ ipconfig
bash: ipconfig: command not found...
asmith@fedora1a:~$ exit
logout
Connection to 192.168.5.10 closed.

C:\Users\Administrator>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet0 2:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::ecbe:9bd1:5c0:8e21%5
IPv4 Address. . . . . : 192.168.1.11
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.1.1

C:\Users\Administrator>
```

Test from Management PC

```
C:\Users\student>ssh 192.168.5.10 -l asmith
The authenticity of host '192.168.5.10 (192.168.5.10)' can't be established.
ECDSA key fingerprint is SHA256:MdLMA/ykGZ7TxT1PVRh4Zu5L4fSrwL+Klzc6DAchhYk.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.5.10' (ECDSA) to the list of known hosts.
asmith@192.168.5.10's password:
Last login: Thu Feb 27 23:08:33 2025 from 192.168.1.11
asmith@fedora1a:~$ exit
logout
Connection to 192.168.5.10 closed.

C:\Users\student>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet0 2:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::5113:86dc:b56b:5890%21
IPv4 Address. . . . . : 192.168.1.21
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.1.1

C:\Users\student>
```

We can not ssh from PC2 because is not supported in GNS3

3.3.2.6.5 WEB server

3.3.2.6.5.1 Modifications in Fedora to allow web server

Update Apache Configuration

1. Change httpd.conf file

```
sudo nano /etc/httpd/conf/httpd.conf
```

2. Find the Listen directive: Listen 80

If it's missing or says Listen [::]:80, change it to:

Listen 0.0.0.0:80

Save and exit (CTRL + X, then Y, then Enter).

3. Restart Apache

```
sudo systemctl restart httpd
```

4. Check if it's now listening on IPv4, you should see 0.0.0.0:80.:
sudo netstat -tulnp | grep :80

5. Check Firewall Settings

```
sudo firewall-cmd --list-services
```

6. If http is missing, allow it and reload:

```
sudo firewall-cmd --add-service=http --permanent
```

```
sudo firewall-cmd --reload
```

```

12 packets transmitted, 0 received, 100% packet loss, time 11308ms

root@fedora1a:~# sudo netstat -tulnp | grep httpd
tcp6      0      0 :::80          :::::::::::::::::::::::::::: LISTEN      1083/httpd
root@fedora1a:~# sudo nano /etc/httpd/conf/httpd.conf
root@fedora1a:~# systemctl restart httpd
root@fedora1a:~# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/service.d
             └─io-timeout-abort.conf, 50-keep-warm.conf
     Active: active (running) since Thu 2025-02-27 23:22:20 EST; 8s ago
   Invocation-ID: a325ff9bbe324183972b72b646ef89ee
     Docs: man:httpd.service(8)
   Main PID: 19819 (httpd)
     Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
     Tasks: 177 (limit: 9439)
     Memory: 14.1M (peak: 14.3M)
       CPU: 108ms
     CGroup: /system.slice/httpd.service
             ├─19819 /usr/sbin/httpd -DFOREGROUND
             ├─19827 /usr/sbin/httpd -DFOREGROUND
             ├─19831 /usr/sbin/httpd -DFOREGROUND
             ├─19832 /usr/sbin/httpd -DFOREGROUND
             └─19833 /usr/sbin/httpd -DFOREGROUND

Feb 27 23:22:20 fedora1a systemd[1]: Starting httpd.service - The Apache HTTP Server...
Feb 27 23:22:20 fedora1a (httpd)[19819]: httpd.service: Referenced but unset environment variable evaluates to an empty string: ORIGIN
Feb 27 23:22:20 fedora1a httpd[19819]: Server configured, listening on: port 80
Feb 27 23:22:20 fedora1a systemd[1]: Started httpd.service - The Apache HTTP Server.
root@fedora1a:~# sudo netstat -tulnp | grep httpd
tcp      0      0 0.0.0.0:80          0.0.0.0:*          LISTEN      19819/httpd
root@fedora1a:~# sudo firewall-cmd --list-services
dhcpv6-client mdns samba-client ssh
root@fedora1a:~# sudo firewall-cmd --list-services
dhcpv6-client mdns samba-client ssh
root@fedora1a:~# sudo firewall-cmd --add-service=http --permanent
success
root@fedora1a:~# sudo firewall-cmd reload
usage: 'firewall-cmd --help' for usage information or see firewall-cmd(1) man page
firewall-cmd: error: unrecognized arguments: reload
root@fedora1a:~# sudo firewall-cmd --reload
success
root@fedora1a:~# sudo firewall-cmd --list-services
dhcpv6-client http mdns samba-client ssh
root@fedora1a:~#

```

```

root@fedora1a:~# systemctl restart httpd
root@fedora1a:~# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/service.d
             └─io-timeout-abort.conf, 50-keep-warm.conf
     Active: active (running) since Thu 2025-02-27 23:22:30 EST; 8s ago
   Invocation-ID: a325ff9bbe324183972b72b646ef89ee
     Docs: man:httpd.service(8)
   Main PID: 19819 (httpd)
     Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
     Tasks: 177 (limit: 9439)
     Memory: 14.1M (peak: 14.3M)
       CPU: 108ms
     CGroup: /system.slice/httpd.service
             ├─19819 /usr/sbin/httpd -DFOREGROUND
             ├─19827 /usr/sbin/httpd -DFOREGROUND
             ├─19831 /usr/sbin/httpd -DFOREGROUND
             ├─19832 /usr/sbin/httpd -DFOREGROUND
             └─19833 /usr/sbin/httpd -DFOREGROUND

Feb 27 23:22:30 fedora1a systemd[1]: Starting httpd.service - The Apache HTTP Server...
Feb 27 23:22:30 fedora1a (httpd)[19819]: httpd.service: Referenced but unset environment variable evaluates to an empty string: ORIGIN
Feb 27 23:22:30 fedora1a httpd[19819]: Server configured, listening on: port 80
Feb 27 23:22:30 fedora1a systemd[1]: Started httpd.service - The Apache HTTP Server.
root@fedora1a:~# sudo netstat -tulnp | grep httpd
tcp      0      0 0.0.0.0:80          0.0.0.0:*          LISTEN      19819/httpd
root@fedora1a:~# sudo firewall-cmd --list-services
dhcpv6-client mdns samba-client ssh
root@fedora1a:~# sudo firewall-cmd --list-services
dhcpv6-client mdns samba-client ssh
root@fedora1a:~# sudo firewall-cmd --add-service=http --permanent
success
root@fedora1a:~# sudo firewall-cmd reload
usage: 'firewall-cmd --help' for usage information or see firewall-cmd(1) man page
firewall-cmd: error: unrecognized arguments: reload
root@fedora1a:~# sudo firewall-cmd --reload
success
root@fedora1a:~# sudo firewall-cmd --list-services
dhcpv6-client http mdns samba-client ssh
root@fedora1a:~#

```

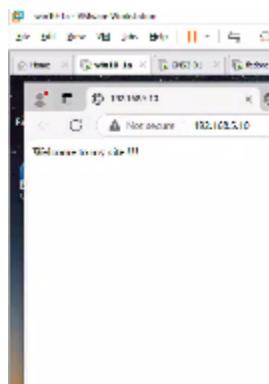
```
#ServerName www.example.com:80
root@fedora1a:~# cat /etc/httpd/conf/httpd.conf | grep 80
#Listen 12.34.56.78:80
#Listen 80
Listen 0.0.0.0:80
#ServerName www.example.com:80
root@fedora1a:~#
```

3.3.2.6.5.2 Test web server from windows NO ACL

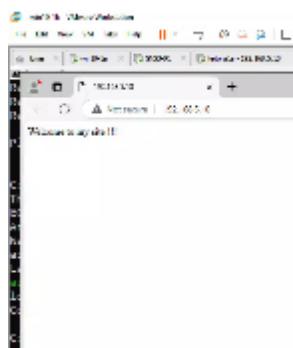
On your Windows 10 machine:

Open a web browser and go to http://192.168.5.10

Web server from win10-1a



Web server form win10-1b



3.3.2.6.6 SSH to ROUTERS

Modify routers to allow all kinds of transport input, this is not recommended for security reasons but can help for testing purposes

```
!!!! R1
enable
config t
username admin
ip domain name gns3.com
crypto key generate rsa general-keys modulus 2048
ip ssh version 2
line vty 0 15
  transport input all
end
```

```
!!!! R2
enable
config t
username admin
ip domain name gns3.com
crypto key generate rsa general-keys modulus 2048
ip ssh version 2
line vty 0 15
  transport input all
end
```

```
!!!! R3
enable
config t
username admin
ip domain name gns3.com
crypto key generate rsa general-keys modulus 2048
ip ssh version 2
line vty 0 15
  transport input all
end
```

3.3.2.6.6.1 Test ssh to routers from win-1a and win-1b

win10-a

R1

```
ssh -o MACs=hmac-sha1,hmac-sha1-96,hmac-md5,hmac-md5-96 -o  
Ciphers=aes128-cbc,3des-cbc,aes192-cbc,aes256-cbc -o HostKeyAlgorithms=ssh-  
rsa -o KexAlgorithms=diffie-hellman-group-exchange-sha1,diffie-hellman-group14-  
sha1,diffie-hellman-group1-sha1 192.168.1.1 -l admin
```

```
C:\Users\Administrator.ssh>ssh -o MACs=hmac-sha1,hmac-sha1-96,hmac-md5,hmac-md5-96 -o Ciphers=aes128-cbc,3des-cbc,aes192-cbc,aes256-cbc -o HostKeyAlgorithms=ssh-rsa -o KexAlgorithms=diffie-hellman-group-exchange-sha1,diffie-hellman-group1-sha1,diffie-hellman-group1-sha1 192.168.1.1 -l admin  
(admin@192.168.1.1) Password:  
WARNING: Authorized Users Only! R1>  
R1>
```

R2

```
ssh -o MACs=hmac-sha1,hmac-sha1-96,hmac-md5,hmac-md5-96 -o  
Ciphers=aes128-cbc,3des-cbc,aes192-cbc,aes256-cbc -o HostKeyAlgorithms=ssh-  
rsa -o KexAlgorithms=diffie-hellman-group-exchange-sha1,diffie-hellman-group14-  
sha1,diffie-hellman-group1-sha1 192.168.2.2 -l admin
```

```
C:\Users\Administrator.ssh>  
C:\Users\Administrator.ssh>ssh -o MACs=hmac-sha1,hmac-sha1-96,hmac-md5,hmac-md5-96 -o Ciphers=aes128-cbc,3des-cbc,aes192-cbc,aes256-cbc -o HostKeyAlgorithms=ssh-rsa -o KexAlgorithms=diffie-hellman-group-exchange-sha1,diffie-hellman-group1-sha1,diffie-hellman-group1-sha1 192.168.2.2 -l admin  
The authenticity of host '192.168.2.2 (192.168.2.2)' can't be established.  
RSA key fingerprint is SHA256:3M+g2HnAxDvSwqJW/hosttRulu=xYU08Zeng.  
This host key is known by the following other names/addresses:  
    C:\Users\Administrator.ssh\known_hosts.3: 192.168.2.1  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added "192.168.2.2" (RSA) to the list of known hosts.  
(admin@192.168.2.2) Password:  
WARNING: Authorized Users Only! R2>  
R2>  
R2>  
R2>  
R2>  
R2>exit  
Connection to 192.168.2.2 closed by remote host.  
Connection to 192.168.2.2 closed.  
Activate Windows  
Go to Settings to activate Windows
```

R3

```
ssh -o MACs=hmac-sha1,hmac-sha1-96,hmac-md5,hmac-md5-96 -o  
Ciphers=aes128-cbc,3des-cbc,aes192-cbc,aes256-cbc -o HostKeyAlgorithms=ssh-  
rsa -o KexAlgorithms=diffie-hellman-group-exchange-sha1,diffie-hellman-group14-  
sha1,diffie-hellman-group1-sha1 192.168.3.2 -l admin
```

```
C:\Users\Administrator.ssh>ssh -o MACs=hmac-sha1,hmac-sha1-96,hmac-md5,hmac-md5-96 -o Ciphers=aes128-cbc,3des-cbc,aes192-cbc,aes256-cbc -o HostKeyAlgorithms=ssh-rsa -o KexAlgorithms=diffie-hellman-group-exchange-sha1,diffie-hellman-group14-sha1,diffie-hellman-group1-sha1 192.168.3.2 -l admin  
The authenticity of host '192.168.3.2 (192.168.3.2)' can't be established.  
RSA key fingerprint is SHA256:3M+g2HnAxDvSwqJW/hosttRulu=xYU08Zeng.  
This host key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added "192.168.3.2" (RSA) to the list of known hosts.  
(admin@192.168.3.2) Password:  
WARNING: Authorized Users Only! R3>  
R3>  
R3>  
R3>exit  
Connection to 192.168.3.2 closed by remote host.  
Connection to 192.168.3.2 closed.  
C:\Users\Administrator.ssh>
```

win10-b

ssh connection to R1 , R2 and R3

3.3.2.6.7 ACL

3.3.2.6.7.1 Create ACL to allow access to Fedora to do ssh and web server to

Creates an extended ACL named WIN10a-TO-FEDORA, the ACL should:

- Allow SSH traffic (port 22) from win10-1a-1 (192.168.1.11) to Fedora (192.168.5.10).
 - Allow HTTP traffic (port 80) from win10-1a-1 to Fedora.

Apply the ACL to the inbound direction of the interface FastEthernet1/0 (the interface connected to Fedora).

```
!!! R1 !!!!
configure terminal
ip access-list extended WIN10a-TO-FEDORA
permit tcp host 192.168.1.11 host 192.168.5.10 eq 22
permit tcp host 192.168.1.11 host 192.168.5.10 eq 80
interface FastEthernet1/0
  ip access-group WIN10a-TO-FEDORA out
end
write memory
```

3.3.2.6.7.1.1 Verify the ACL

You can verify the ACL configuration in R1:

```
R1# show ip access-lists WIN10a-TO-FEDORA  
R1#show running-config | section interface
```

```
R1#show ip access-lists Win10a-Fedora
Extended IP access list Win10a-Fedora
    10 permit tcp host 192.168.1.11 host 192.168.5.10 eq 22
    20 permit tcp host 192.168.1.11 host 192.168.5.10 eq www
R1#
```

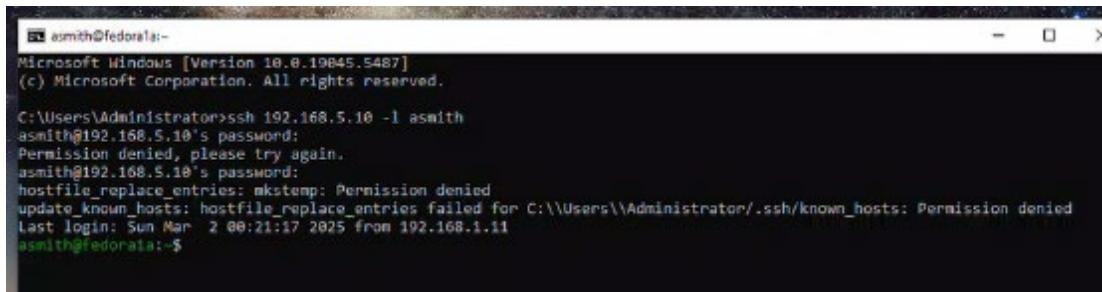
```
!
interface FastEthernet0/0
description Connection between R1 and SW1
ip address 192.168.1.1 255.255.255.0
duplex full
!
interface FastEthernet1/0
description Connection between R1 and Fedora
ip address 192.168.5.1 255.255.255.0
ip access-group WIN10a-TO-FEDORA out ←
speed auto
duplex auto
!
interface FastEthernet1/1
no ip address
shutdown
speed auto
duplex auto
!
interface Serial2/0
description Connection between R1 and R2
ip address 192.168.2.1 255.255.255.0
serial restart-delay 0
!
interface Serial2/1
no ip address
shutdown
serial restart-delay 0
```

3.3.2.6.7.1.2 Test the Configuration

Ensure that only win10-1a-1 can SSH into Fedora.

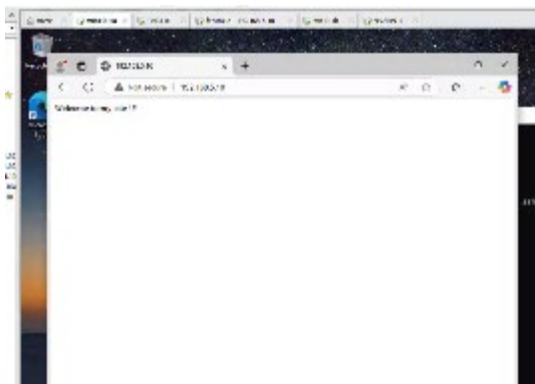
Win10-1a-1

```
ssh 192.168.5.10 -l asmith
```



```
asmith@fedorata:~  
Microsoft Windows [Version 10.0.19045.5487]  
(c) Microsoft Corporation. All rights reserved.  
C:\Users\Administrator>ssh 192.168.5.10 -l asmith  
asmith@192.168.5.10's password:  
Permission denied, please try again.  
asmith@192.168.5.10's password:  
hostfile_replace_entries: mkstemp: Permission denied  
update_known_hosts: hostfile_replace_entries failed for C:\Users\Administrator/.ssh/known_hosts: Permission denied  
Last login: Sun Mar 2 00:21:17 2025 from 192.168.1.11  
asmith@fedorata:~$
```

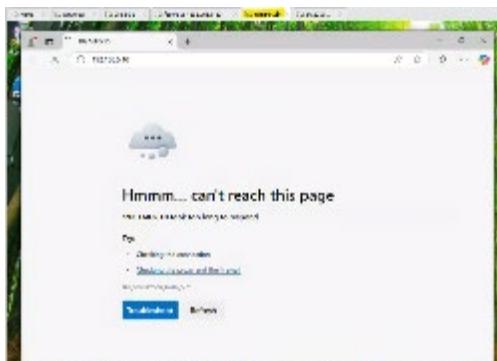
Make sure Win10-1a can access web server



Win10-1b-1



```
ssh 192.168.5.10 -l asmith  
C:\Users\student>  
C:\Users\student>ssh 192.168.5.10 -l asmith  
ssh: connect to host 192.168.5.10 port 22: Connection timed out  
C:\Users\student>
```



3.3.2.6.7.2 Create ACL to only allow win10-1b to access routers via ssh

!!! R1!!!

```
configure terminal
ip access-list extended WIN10b-SSH-ACCESS
permit tcp host 192.168.1.21 any eq 22
line vty 0 15
access-class WIN10b-SSH-ACCESS in
end
write memory
```

```
R1#!!!
R1#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)# ip access-list extended WIN10b-SSH-ACCESS
R1(config-ext-nacl)# permit tcp host 192.168.1.21 any eq 22
R1(config-ext-nacl)#line vty 0 15
R1(config-line)# access-class WIN10b-SSH-ACCESS in
R1(config-line)#end
R1#write memory
Building configuration...

*Mar  2 05:32:17.983: %SYS-5-CONFIG_I: Configured from console by console[OK]
R1#
```

```
R1#
R1#show ip access-lists
Extended IP access list WIN10a-TO-FEDORA
  10 permit tcp host 192.168.1.11 host 192.168.5.10 eq 22 (114 matches)
  20 permit tcp host 192.168.1.11 host 192.168.5.10 eq www
Extended IP access list WIN10b-SSH-ACCESS
  10 permit tcp host 192.168.1.21 any eq 22
R1#show running-config | section vty
line vty 0 4
  access-class WIN10b-SSH-ACCESS in
  login local
  transport input all
line vty 5 15
  access-class WIN10b-SSH-ACCESS in
  login local
  transport input all
R1#
```

!!! R2!!!

```
configure terminal
ip access-list extended WIN10b-SSH-ACCESS
permit tcp host 192.168.1.21 any eq 22
line vty 0 15
access-class WIN10b-SSH-ACCESS in
end
write memory
```

```
R2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)# ip access-list extended WIN10b-SSH-ACCESS
R2(config-ext-nacl)# permit tcp host 192.168.1.21 any eq 22
R2(config-ext-nacl)#line vty 0 15
R2(config-line)#access-class WIN10b-SSH-ACCESS in
R2(config-line)#end
R2#write memory
Building configuration...
[OK]
R2#
*Mar  2 05:32:35.467: %SYS-5-CONFIG_I: Configured from console by console
R2#
```

```
R2#show ip access-lists
Extended IP access list WIN10b-SSH-ACCESS
    10 permit tcp host 192.168.1.21 any eq 22
R2#show running-config | section vty
line vty 0 4
    access-class WIN10b-SSH-ACCESS in
    login local
    transport input all
line vty 5 15
    access-class WIN10b-SSH-ACCESS in
    login local
    transport input all
R2#
```

!!! R3!!!

```
configure terminal
ip access-list extended WIN10b-SSH-ACCESS
    permit tcp host 192.168.1.21 any eq 22
line vty 0 15
    access-class WIN10b-SSH-ACCESS in
end
write memory
```

```
R3!!!! R3!!!
R3#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)# ip access-list extended WIN10b-SSH-ACCESS
R3(config-ext-nacl)# permit tcp host 192.168.1.21 any eq 22
R3(config-ext-nacl)#line vty 0 15
R3(config-line)#access-class WIN10b-SSH-ACCESS in
R3(config-line)#end
R3#write memory
Building configuration...
[OK]
R3#
*Mar  2 05:32:56.655: %SYS-5-CONFIG_I: Configured from console by console
R3#
R3!!!! R3!!!
```

```
R3#show ip access-lists
Extended IP access list WIN10b-SSH-ACCESS
    10 permit tcp host 192.168.1.21 any eq 22
R3#show running-config | section vty
line vty 0 4
access-class WIN10b-SSH-ACCESS in
login local
transport input all
line vty 5 15
access-class WIN10b-SSH-ACCESS in
login local
transport input all
R3#
```

3.3.2.6.7.2.1 Test ACL for ssh to routers

Test access to router ssh from Win10-1a1 and win10-1b-1

Win10-1a-1

```
C:\Users\Administrator>ssh -o MACsha256-sha1,hmac-sha1-96,hmac-md5,hmac-md5-96 -o Ciphersaes128-cbc,3des-cbc,ses192-cbc,ses256-cbc -o HostKeyWigntthesessh-nso -o Ke
c-hellman-group-exchange-sha1,diffie-hellman-group14-sha1,diffie-hellman-group-sha1 192.168.1.1 1 admn
ssh: connect to host 192.168.1.1 port 22: connection refused

C:\Users\Administrator>ssh -o MACsha256-sha1,hmac-sha1-96,hmac-md5,hmac-md5-96 -o Ciphersaes128-cbc,3des-cbc,ses192-cbc,ses256-cbc -o HostKeyWigntthesessh-nso -o Ke
c-hellman-group-exchange-sha1,diffie-hellman-group14-sha1,diffie-hellman-group-sha1 192.168.1.2 1 admn
ssh: connect to host 192.168.1.2 port 22: connection refused

C:\Users\Administrator>ssh -o Cnfig
Windows IP Configuration

Ethernet adapter Ethernet 2:

  Connection-specific DNS Suffix . .
  Link-Local IPv6 Address . . . . . : fe80::ecbe:8bd! Sc0\8e2185
  IPv4 Address . . . . . : 192.168.1.11
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 192.168.1.1
```

Win10-1b-1

3.3.2.6.8 Connect to CLOUD and set NAT

R3 add cloud connection

```
!!!!!!!
!!! Router R3:
!!!!!!!
enable
conf t

int f1/1
description Connection between R3 and Cloud

ip add dhcp
no shut
exit
end
write memory

show ip interface brief
```

```
R3#!!!!!!!
R3#!!! Router R3:
R3#!!!!!!!
R3#enable
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#
R3(config)#
R3(config)#
R3(config)#int f1/1
R3(config-1f1)e description Connection between R3 and Cloud
R3(config-1f1)e description Connection between R3 and Cloud
R3(config-1f1)e
R3(config-1f1)ip add dhcp
R3(config-1f1)no shut
R3(config-1f1)no shutdown
R3(config-1f1)exit
R3(config)#
R3#
R3#
R3#
R3#
*Feb 28 03:44:01.010: %SYS-5-CONFIG_I: Configured from console by console
R3#
*Feb 28 03:44:01.047: %LINK-3-UPDOWN: Interface FastEthernet1/1, changed state to up
*Feb 28 03:44:01.047: %LINKDOWN-3-UPDOWN: Line protocol on Interface FastEthernet1/1, changed state to up
R3#
R3#show ip
*Feb 28 03:44:01.4.751: %R03-3-RESOLVE_FAILED: Adj resolve request failed to resolve 10.104.1.4 FastEthernet0/1
R3#show ip
*Feb 28 03:44:01.735: %R03-6-ADDRESS_ASSIGNED: Interface FastEthernet1/1 assigned IP address 10.104.0.27, mask 255.255.0.0, hostname
```

show ip interface brief

```

R3#
R3#show ip interface brief
Interface          IP-Address      OK? Method Status          Protocol
FastEthernet0/0    192.168.1.3    YES NVRAM up           up
FastEthernet1/0    192.168.4.1    YES NVRAM up           up
FastEthernet1/1    10.164.0.27   YES DHCP  up           up
Serial2/0          unassigned     YES NVRAM administratively down down
Serial2/1          192.168.3.2    YES NVRAM up           up
Serial2/2          unassigned     YES NVRAM administratively down down
Serial2/3          unassigned     YES NVRAM administratively down down
R3#

```

configure terminal

! Step 1: Define the internal network (NAT pool)

```

access-list 1 permit 192.168.1.0 0.0.0.255
access-list 1 permit 192.168.2.0 0.0.0.255
access-list 1 permit 192.168.3.0 0.0.0.255
access-list 1 permit 192.168.4.0 0.0.0.255
access-list 1 permit 192.168.5.0 0.0.0.255

```

! Step 2: Configure NAT overload (PAT) on the outside interface

```

interface FastEthernet1/1
ip nat outside

```

! Step 3: Configure the inside interfaces

```

interface FastEthernet0/0
ip nat inside

```

```

interface FastEthernet1/0
ip nat inside

```

! Step 4: Enable NAT overload (PAT) for the internal network

```

ip nat inside source list 1 interface FastEthernet1/1 overload

```

```

end

```

```

write memory

```

R1 forwards all traffic (default route 0.0.0.0/0) to R2 (192.168.2.2).

!!! R1 !!!

```

configure terminal

```

```

ip route 0.0.0.0 0.0.0.0 192.168.2.2

```

```

end

```

```

write memory

```

```

Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#ip route 0.0.0.0 0.0.0.0 192.168.2.2
R1(config)#end
R1#write memory
Building configuration...
[OK]
R1#
*Mar 2 06:37:40.099: %SYS-5-CONFIG_I: Configured from console by console
R1#\n
^
% Invalid input detected at '^' marker.

R1#
R1#show ip route
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISPs
      + - replicated route, % - next hop override

Gateway of last resort is 192.168.2.2 to network 0.0.0.0

S*   0.0.0.0/0 [1/0] via 192.168.2.2
     192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.1.0/24 is directly connected, FastEthernet0/0
L     192.168.1.1/32 is directly connected, FastEthernet0/0
     192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.2.0/24 is directly connected, Serial2/0
L     192.168.2.1/32 is directly connected, Serial2/0
O     192.168.3.0/24 [110/65] via 192.168.1.3, 2d04h, FastEthernet0/0
          [110/65] via 192.168.1.2, 2d04h, FastEthernet0/0
O     192.168.4.0/24 [110/2] via 192.168.1.3, 2d04h, FastEthernet0/0
     192.168.5.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.5.0/24 is directly connected, FastEthernet1/0
L     192.168.5.1/32 is directly connected, FastEthernet1/0
R1#

```

R2 forwards all traffic (default route 0.0.0.0/0) to R3 (192.168.3.2).

```

!!! R2 !!!
configure terminal
ip route 0.0.0.0 0.0.0.0 192.168.3.2
end
write memory

```

```

transport input all
R2#!!! R2 !!!
R2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#ip route 0.0.0.0 0.0.0.0 192.168.3.2
R2(config)#end
R2#write memory
Building configuration...
[OK]
R2#
*Mar  2 06:39:45.831: %SYS-5-CONFIG_I: Configured from console by console
R2#
R2#show ip route
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, 1 - LIS
      + - replicated route, % - next hop override

Gateway of last resort is 192.168.3.2 to network 0.0.0.0

S*   0.0.0.0/0 [1/0] via 192.168.3.2
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.1.0/24 is directly connected, FastEthernet0/0
L     192.168.1.2/32 is directly connected, FastEthernet0/0
      192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.2.0/24 is directly connected, Serial2/0
L     192.168.2.2/32 is directly connected, Serial2/0
      192.168.3.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.3.0/24 is directly connected, Serial2/1
L     192.168.3.1/32 is directly connected, Serial2/1
O     192.168.4.0/24 [110/2] via 192.168.1.3, 2d04h, FastEthernet0/0
      192.168.100.0/32 is subnetted, 1 subnets
C     192.168.100.100 is directly connected, Loopback0
R2#

```

3.3.2.6.8.1 NAT Test

A. Start NAT debug on R3

```
debug ip nat
```

B. Print statistics for NAT translations in R3

```
show ip nat statistics
```

When first checked the NAT translations, they are empty because the NAT entries are temporary and expire quickly after the ICMP exchange is complete.

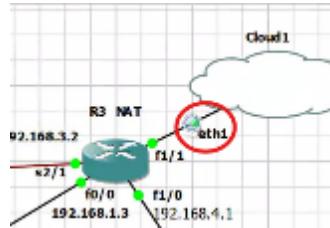
```
show ip nat translations
```

```

R3#show ip nat statistics
Total active translations: 0 (0 static, 0 dynamic; 0 extended)
Outside interfaces:
  FastEthernet1/1
Inside interfaces:
  FastEthernet0/0, FastEthernet1/0
Hits: 177  Misses: 0
CEF Translated packets: 177, CEF Punted packets: 6573
Expired translations: 77
Dynamic mappings:
-- Inside Source
[Id: 1] access-list 1 interface FastEthernet1/1 refcount 0
nat-limit statistics:
  max entry: max allowed 0, used 0, missed 0
R3#
R3#
R3#
R3#show ip nat translations
R3#
R3#
R3#
R3#
R3#

```

C. Start wireshark trace on R3 f1/1



D. Test ping from PC2 (192.168.4.11) to Fedora box (10.164.101.103)

```

PC2>
PC2> show
NAME      IP/MASK          GATEWAY        MAC          LPORT  RMOST:PORT
PC2      192.168.4.11/24   192.168.4.1   00:50:79:66:68:00  20053  127.0.0.1:20054
      fe80::250:79ff:fe66:6800/64

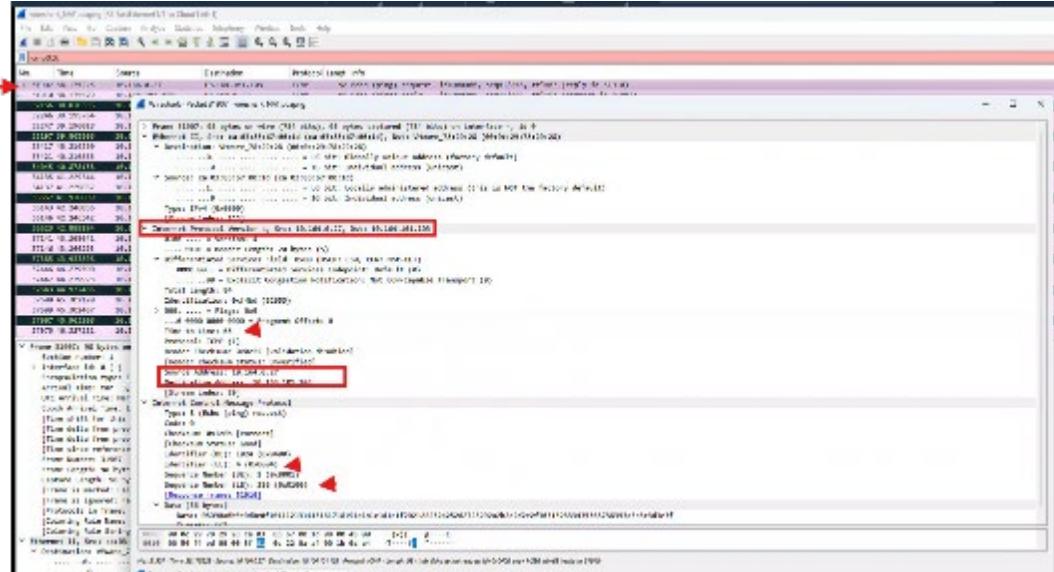
PC2>
PC2>
PC2>
PC2>
PC2> ping 10.164.101.103 -c 15
84 bytes from 10.164.101.103 icmp_seq=1 ttl=63 time=20.263 ms
84 bytes from 10.164.101.103 icmp_seq=2 ttl=63 time=21.132 ms
84 bytes from 10.164.101.103 icmp_seq=3 ttl=63 time=16.933 ms
84 bytes from 10.164.101.103 icmp_seq=4 ttl=63 time=12.261 ms
84 bytes from 10.164.101.103 icmp_seq=5 ttl=63 time=14.548 ms
84 bytes from 10.164.101.103 icmp_seq=6 ttl=63 time=19.815 ms
84 bytes from 10.164.101.103 icmp_seq=7 ttl=63 time=12.278 ms
84 bytes from 10.164.101.103 icmp_seq=8 ttl=63 time=19.808 ms
84 bytes from 10.164.101.103 icmp_seq=9 ttl=63 time=23.911 ms
84 bytes from 10.164.101.103 icmp_seq=10 ttl=63 time=12.671 ms
84 bytes from 10.164.101.103 icmp_seq=11 ttl=63 time=15.382 ms
84 bytes from 10.164.101.103 icmp_seq=12 ttl=63 time=13.814 ms
84 bytes from 10.164.101.103 icmp_seq=13 ttl=63 time=18.221 ms
84 bytes from 10.164.101.103 icmp_seq=14 ttl=63 time=21.355 ms
84 bytes from 10.164.101.103 icmp_seq=15 ttl=63 time=20.585 ms
PC2>

```

E. Stop Wireshark capture

F. Wireshark trace analysis

- 1) ICMP Echo Request (Ping Request): The Wireshark trace shows ICMP (ping) packets between the source (10.164.0.27, which is R3's outside interface) and the destination (10.164.101.103).



- id=0x0400: This is the ICMP identifier (1024 in decimal).
- seq=1/256: This is the sequence number (1 in decimal).
- ttl=63: Time to Live is 63, indicating the packet has traversed multiple hops.

2) ICMP Echo Reply (Ping Reply):

The destination (10.164.101.103) responds with an ICMP Echo Reply to the source (10.164.0.27).

- The id and seq match the request, confirming this is the reply to the earlier request.
- ttl=64: The TTL is 64, which is typical for the destination host.

G. NAT in Action:

- The Wireshark trace shows the packets after NAT has been applied. The original source IP (192.168.4.11) is translated to 10.164.0.27, which is why you see 10.164.0.27 as the source in the trace.
 - The ICMP identifiers and sequence numbers in the Wireshark trace match the NAT translations you see in R3's output.

No.	Time	Source	Destination	Protocol	Length	Info
344607	18:175599	18.164.0.37	18.164.181.183	TCP	94	Echo (ping) request Seq=0x486, seq=0/196, ttl=63 (reply in 31918)
351010	18:175579	18.164.181.183	18.164.0.27	TCP	95	Echo (ping) reply Seq=0x486, seq=0/256, ttl=64 (request in 31987)
32156	28.378355	18.164.0.37	18.164.1.4	TCP	70	Redirect (Redirect for network)
32246	29.395760	18.164.0.37	18.164.181.183	TCP	94	Echo (ping) request Seq=0x487, seq=0/196, ttl=63 (reply in 30087)
32247	30.196180	18.164.181.183	18.164.0.27	TCP	95	Echo (ping) reply Seq=0x481, seq=0/512, ttl=64 (request in 32246)
32247	30.196180	18.164.0.27	18.164.1.4	TCP	70	Redirect (Redirect for network)
33437	48.316338	18.164.0.37	18.164.181.183	TCP	94	Echo (ping) request Seq=0x482, seq=0/198, ttl=63 (reply in 33421)
33421	49.231665	18.164.181.183	18.164.0.27	TCP	95	Echo (ping) reply Seq=0x482, seq=0/765, ttl=64 (request in 33421)
34845	49.378138	18.164.0.37	18.164.1.4	TCP	70	Redirect (Redirect for network)
34335	49.322454	18.164.0.37	18.164.181.183	TCP	94	Echo (ping) request Seq=0x485, seq=0/194, ttl=63 (reply in 34137)
34327	50.212657	18.164.181.183	18.164.0.27	TCP	95	Echo (ping) reply Seq=0x480, seq=0/1034, ttl=64 (request in 34135)
35657	41.976669	18.164.0.37	18.164.1.4	TCP	70	Redirect (Redirect for network)
35843	42.248856	18.164.0.27	18.164.181.183	TCP	94	Echo (ping) request Seq=0x484, seq=0/1289, ttl=63 (reply in 36148)
36346	42.284627	18.164.181.183	18.164.0.27	TCP	95	Echo (ping) reply Seq=0x484, seq=0/1288, ttl=64 (request in 36149)
36623	42.881194	18.164.0.37	18.164.1.4	TCP	70	Redirect (Redirect for network)
37341	45.255841	18.164.0.27	18.164.181.183	TCP	94	Echo (ping) request Seq=0x485, seq=0/158, ttl=63 (reply in 37148)
37346	45.366393	18.164.181.183	18.164.0.27	TCP	95	Echo (ping) reply Seq=0x485, seq=0/153, ttl=64 (request in 37341)
37205	45.912200	18.164.0.37	18.164.1.4	TCP	70	Redirect (Redirect for network)
37446	46.275599	18.164.0.27	18.164.181.183	TCP	94	Echo (ping) request Seq=0x486, seq=0/1732, ttl=63 (reply in 37447)
38867	46.793503	18.164.181.183	18.164.0.27	TCP	95	Echo (ping) reply Seq=0x486, seq=0/1743, ttl=64 (request in 38846)
37645	44.9374405	18.164.0.27	18.164.1.4	TCP	70	Redirect (Redirect for network)
37680	45.380120	18.164.0.27	18.164.181.183	TCP	94	Echo (ping) request Seq=0x487, seq=0/2040, ttl=63 (reply in 37699)
37694	45.482707	18.164.181.183	18.164.0.27	TCP	95	Echo (ping) reply Seq=0x487, seq=0/2040, ttl=64 (request in 37688)
37667	45.483260	18.164.0.27	18.164.1.4	TCP	70	Redirect (Redirect for network)
37707	46.321725	18.164.0.27	18.164.181.183	TCP	94	Echo (ping) request Seq=0x488, seq=0/2024, ttl=63 (reply in 37792)

H. R3 printout

From the output in R3, R3 is performing NAT (Network Address Translation) for ICMP (ping) packets originating from PC2 (192.168.4.11) to the destination 10.164.101.103.

1. NAT Translations:

- R3 is translating the source IP address of PC2 (192.168.4.11) to its own outside interface IP address (10.164.0.27) when the packets go out to the destination (10.164.101.103).
- The NAT translations are dynamic and temporary, as they are created for each ICMP request and reply.

```
*Mar 3 18:28:12.842: NAT: Entry assigned id 102
*Mar 3 18:28:12.842: NAT*: ICMP id=48628->1024
*Mar 3 18:28:12.846: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.103 [62653]
*Mar 3 18:28:12.854: NAT*: ICMP id=1024->48628
*Mar 3 18:28:12.854: NAT*: s=10.164.101.103, d=10.164.0.27->192.168.4.11 [61339]
R3#
*Mar 3 18:28:13.866: NAT: Entry assigned id 103
*Mar 3 18:28:13.866: NAT*: ICMP id=48884->1025
*Mar 3 18:28:13.866: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.103 [62654]
*Mar 3 18:28:13.874: NAT*: ICMP id=1025->48884
*Mar 3 18:28:13.874: NAT*: s=10.164.101.103, d=10.164.0.27->192.168.4.11 [62062]
*Mar 3 18:28:14.866: NAT: Entry assigned id 104
*Mar 3 18:28:14.866: NAT*: ICMP id=49140->1026
*Mar 3 18:28:14.866: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.103 [62655]
*Mar 3 18:28:14.870: NAT*: ICMP id=1026->49140
*Mar 3 18:28:14.870: NAT*: s=10.164.101.103, d=10.164.0.27->192.168.4.11 [62332]
R3#
*Mar 3 18:28:15.906: NAT: Entry assigned id 105
*Mar 3 18:28:15.906: NAT*: ICMP id=49396->1027
*Mar 3 18:28:15.906: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.103 [62656]
*Mar 3 18:28:15.906: NAT*: ICMP id=1027->49396
*Mar 3 18:28:15.906: NAT*: s=10.164.101.103, d=10.164.0.27->192.168.4.11 [63084]
R3#
*Mar 3 18:28:16.918: NAT: Entry assigned id 106
*Mar 3 18:28:16.918: NAT*: ICMP id=49652->1028
*Mar 3 18:28:16.918: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.103 [62657]
*Mar 3 18:28:16.922: NAT*: ICMP id=1028->49652
*Mar 3 18:28:16.922: NAT*: s=10.164.101.103, d=10.164.0.27->192.168.4.11 [63778]
R3#
*Mar 3 18:28:17.934: NAT: Entry assigned id 107
*Mar 3 18:28:17.934: NAT*: ICMP id=49988->1029
*Mar 3 18:28:17.934: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.103 [62658]
*Mar 3 18:28:17.942: NAT*: ICMP id=1029->49988
*Mar 3 18:28:17.942: NAT*: s=10.164.101.103, d=10.164.0.27->192.168.4.11 [64379]
R3#
*Mar 3 18:28:18.954: NAT: Entry assigned id 108
*Mar 3 18:28:18.954: NAT*: ICMP id=50164->1030
*Mar 3 18:28:18.954: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.103 [62659]
*Mar 3 18:28:18.958: NAT*: ICMP id=1030->50164
```

```

*Mar 3 18:28:18.954: NAT: Entry assigned id 188
*Mar 3 18:28:18.954: NAT*: ICMP id=50164->1038
*Mar 3 18:28:18.954: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.183 [62659]
*Mar 3 18:28:18.958: NAT*: ICMP id=1038->50164
*Mar 3 18:28:18.958: NAT*: s=10.164.101.183, d=10.164.0.27->192.168.4.11 [65000]
R3A
*Mar 3 18:28:19.970: NAT: Entry assigned id 189
*Mar 3 18:28:19.970: NAT*: ICMP id=50628->1031
*Mar 3 18:28:19.970: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.183 [61660]
*Mar 3 18:28:19.978: NAT*: ICMP id=1031->50628
*Mar 3 18:28:19.978: NAT*: s=10.164.101.183, d=10.164.0.27->192.168.4.11 [188]
R3B
*Mar 3 18:28:20.994: NAT: Entry assigned id 110
*Mar 3 18:28:20.994: NAT*: ICMP id=50676->1032
*Mar 3 18:28:20.994: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.183 [61661]
*Mar 3 18:28:21.980: NAT*: ICMP id=1032->50676
*Mar 3 18:28:21.986: NAT*: s=10.164.101.183, d=10.164.0.27->192.168.4.11 [695]
R3C
*Mar 3 18:28:22.018: NAT: Entry assigned id 111
*Mar 3 18:28:22.018: NAT*: ICMP id=50932->1035
*Mar 3 18:28:22.018: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.183 [62662]
*Mar 3 18:28:22.022: NAT*: ICMP id=1033->50932
*Mar 3 18:28:22.022: NAT*: s=10.164.101.183, d=10.164.0.27->192.168.4.11 [837]
R3D
*Mar 3 18:28:23.034: NAT: Entry assigned id 112
*Mar 3 18:28:23.034: NAT*: ICMP id=51188->1034
*Mar 3 18:28:23.034: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.183 [62663]
*Mar 3 18:28:23.038: NAT*: ICMP id=1034->51188
*Mar 3 18:28:23.038: NAT*: s=10.164.101.183, d=10.164.0.27->192.168.4.11 [1197]
R3E
*Mar 3 18:28:24.046: NAT: Entry assigned id 113
*Mar 3 18:28:24.046: NAT*: ICMP id=51444->1035
*Mar 3 18:28:24.046: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.183 [62664]
*Mar 3 18:28:24.050: NAT*: ICMP id=1035->51444
*Mar 3 18:28:24.050: NAT*: s=10.164.101.183, d=10.164.0.27->192.168.4.11 [1585]
R3F

```

```

R3H
*Mar 3 18:28:25.866: NAT: Entry assigned id 114
*Mar 3 18:28:25.866: NAT*: ICMP id=51700->1036
*Mar 3 18:28:25.866: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.183 [62665]
*Mar 3 18:28:25.874: NAT*: ICMP id=1036->51700
*Mar 3 18:28:25.874: NAT*: s=10.164.101.183, d=10.164.0.27->192.168.4.11 [2565]
R3I
*Mar 3 18:28:26.878: NAT: Entry assigned id 115
*Mar 3 18:28:26.878: NAT*: ICMP id=51956->1037
*Mar 3 18:28:26.878: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.183 [62666]
*Mar 3 18:28:26.894: NAT*: ICMP id=1037->51956
*Mar 3 18:28:26.894: NAT*: s=10.164.101.183, d=10.164.0.27->192.168.4.11 [2626]
R3J
*Mar 3 18:28:27.186: NAT: Entry assigned id 116
*Mar 3 18:28:27.186: NAT*: ICMP id=52212->1038
*Mar 3 18:28:27.186: NAT*: s=192.168.4.11->10.164.0.27, d=10.164.101.183 [62667]
*Mar 3 18:28:27.114: NAT*: ICMP id=1038->52212
*Mar 3 18:28:27.114: NAT*: s=10.164.101.183, d=10.164.0.27->192.168.4.11 [2932]
R3K
R3L
R3M

```

Each ICMP packet gets a unique NAT entry with a specific **ICMP identifier** and **sequence number**. For example:

- Inside global: The translated source IP and port (10.164.0.27:1024).
- Inside local: The original source IP and port (192.168.4.11:48628).
- Outside local and Outside global: The destination IP and port (10.164.101.103:48628 and 10.164.101.103:1024).

NOTE - If check of NAT translations is done immediately after the ping, you will see the active translations, but if you wait a few seconds, they will expire and disappear.

```
R3#show ip nat translations
Pro Inside global    Inside local      Outside local      Outside global
icmp 10.164.0.27:1024 192.168.4.11:48628 10.164.101.103:48628 10.164.101.103:1024
icmp 10.164.0.27:1025 192.168.4.11:48604 10.164.101.103:48604 10.164.101.103:1025
icmp 10.164.0.27:1026 192.168.4.11:49146 10.164.101.103:49146 10.164.101.103:1026
icmp 10.164.0.27:1027 192.168.4.11:49396 10.164.101.103:49396 10.164.101.103:1027
icmp 10.164.0.27:1028 192.168.4.11:49652 10.164.101.103:49652 10.164.101.103:1028
icmp 10.164.0.27:1029 192.168.4.11:49908 10.164.101.103:49908 10.164.101.103:1029
icmp 10.164.0.27:1030 192.168.4.11:50164 10.164.101.103:50164 10.164.101.103:1030
icmp 10.164.0.27:1031 192.168.4.11:50420 10.164.101.103:50420 10.164.101.103:1031
icmp 10.164.0.27:1032 192.168.4.11:50676 10.164.101.103:50676 10.164.101.103:1032
icmp 10.164.0.27:1033 192.168.4.11:50932 10.164.101.103:50932 10.164.101.103:1033
icmp 10.164.0.27:1034 192.168.4.11:51188 10.164.101.103:51188 10.164.101.103:1034
Pro Inside global    Inside local      Outside local      Outside global
icmp 10.164.0.27:1035 192.168.4.11:51444 10.164.101.103:51444 10.164.101.103:1035
icmp 10.164.0.27:1036 192.168.4.11:51700 10.164.101.103:51700 10.164.101.103:1036
icmp 10.164.0.27:1037 192.168.4.11:51956 10.164.101.103:51956 10.164.101.103:1037
icmp 10.164.0.27:1038 192.168.4.11:52212 10.164.101.103:52212 10.164.101.103:1038
R3#

```

1) NAT Statistics:

- The show ip nat statistics command shows that there are currently **15 active translations**, which correspond to the 15 ICMP requests and replies you sent from PC2.
- The Hits: 207 indicates that NAT has been applied to 207 packets, and Expired translations: 77 shows that 77 NAT entries have expired so far.

```
R3#show ip nat statistics
Total active translations: 15 (0 static, 15 dynamic; 15 extended)
Outside interfaces:
  FastEthernet1/1
Inside interfaces:
  FastEthernet0/0, FastEthernet1/0
Hits: 207  Misses: 0
CEF Translated packets: 207, CEF Punted packets: 6576
Expired translations: 77
Dynamic mappings:
-- Inside Source
[Id: 1] access-list 1 interface FastEthernet1/1 refcount 15
nat-limit statistics:
  max entry: max allowed 0, used 0, missed 0
R3#

```

2) NAT Expiry:

- After a short period (likely the default NAT timeout for ICMP, which is typically a few seconds), the NAT entries expire and are removed from the translation table.

```

*Mar  3 18:29:13.030: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1024 (48628)
*Mar  3 18:29:13.030: NAT: Freeing nat entry, id 102
R3#
*Mar  3 18:29:14.054: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1025 (48884)
*Mar  3 18:29:14.054: NAT: Freeing nat entry, id 103
R3#
*Mar  3 18:29:15.078: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1026 (49140)
*Mar  3 18:29:15.082: NAT: Freeing nat entry, id 104
R3#
*Mar  3 18:29:16.102: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1027 (49396)
*Mar  3 18:29:16.102: NAT: Freeing nat entry, id 105
R3#
*Mar  3 18:29:17.126: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1028 (49652)
*Mar  3 18:29:17.126: NAT: Freeing nat entry, id 106
R3#
*Mar  3 18:29:18.150: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1029 (49908)
*Mar  3 18:29:18.150: NAT: Freeing nat entry, id 107
R3#
*Mar  3 18:29:19.174: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1030 (50164)
*Mar  3 18:29:19.178: NAT: Freeing nat entry, id 108
R3#
*Mar  3 18:29:20.198: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1031 (50420)
*Mar  3 18:29:20.198: NAT: Freeing nat entry, id 109
R3#
*Mar  3 18:29:21.222: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1032 (50676)
*Mar  3 18:29:21.222: NAT: Freeing nat entry, id 110
R3#
*Mar  3 18:29:22.250: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1033 (50932)
*Mar  3 18:29:22.250: NAT: Freeing nat entry, id 111
R3#
*Mar  3 18:29:23.274: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1034 (51188)
*Mar  3 18:29:23.274: NAT: Freeing nat entry, id 112
R3#
*Mar  3 18:29:24.298: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1035 (51444)
*Mar  3 18:29:24.298: NAT: Freeing nat entry, id 113
R3#
*Mar  3 18:29:25.322: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1036 (51700)
*Mar  3 18:29:25.322: NAT: Freeing nat entry, id 114
R3#

```

```

*Mar  3 18:29:25.322: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1036 (51700)
*Mar  3 18:29:25.322: NAT: Freeing nat entry, id 114
R3#
*Mar  3 18:29:26.346: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1037 (51956)
*Mar  3 18:29:26.346: NAT: Freeing nat entry, id 115
R3#
*Mar  3 18:29:27.370: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1038 (52212)
*Mar  3 18:29:27.374: NAT: Freeing nat entry, id 116
R3#

```

Example, the printout below indicates that the NAT entry for the ICMP packet with identifier 1026 has expired and been removed.

```

*Mar  3 18:29:15.078: NAT: expiring 10.164.0.27 (192.168.4.11) icmp 1026 (49140)
*Mar  3 18:29:15.082: NAT: Freeing nat entry, id 104

```

I. Undebug NAT

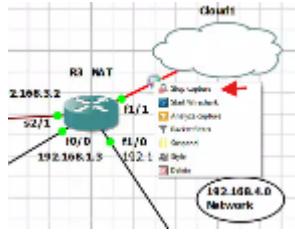
Undebug all

```

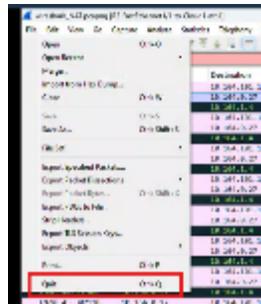
R3#
R3#undebug all
All possible debugging has been turned off
R3#

```

J. Stop Wireshark trace



K. Quit Wireshark

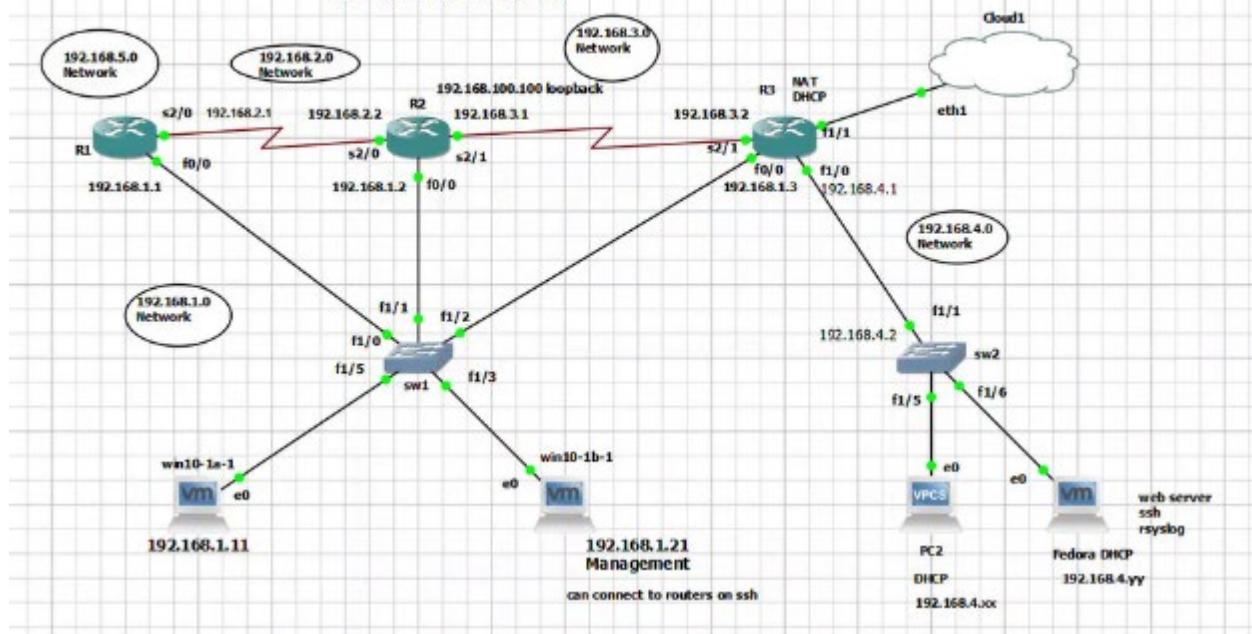


3.3.3 OSPF - ACLS , Fedora with Apache SSH RSYSLOG , Router DHCP Cloud NAT

3.3.3.1 Topology

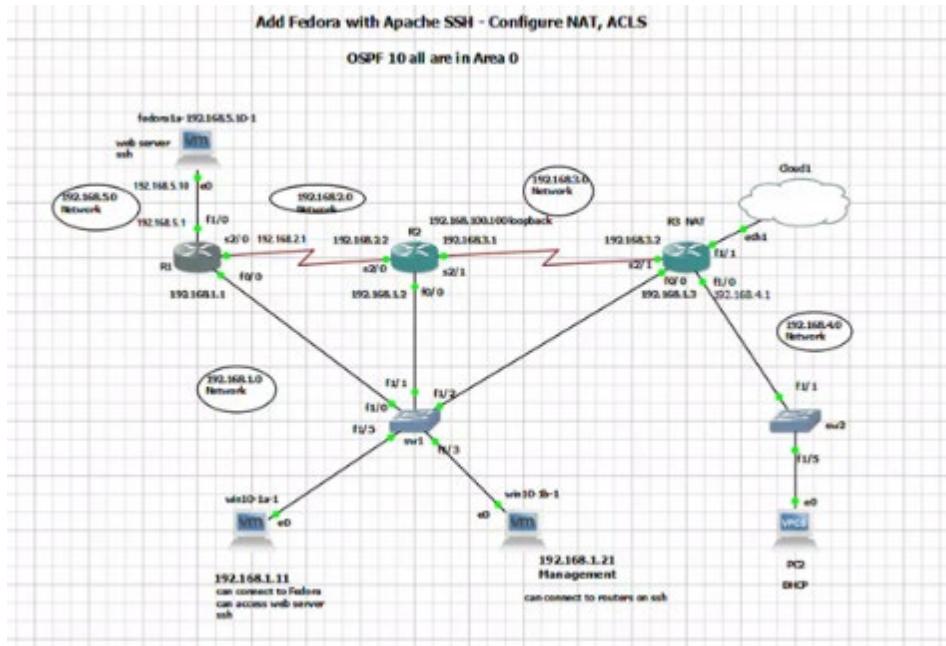
OSPF - Fedora with Apache SSH - DHCP Cloud NAT, ACLS - RSYSLOG

OSPF 10 all are in Area 0



3.3.3.2 Preparation

Having the following topology as a base



The topology in the figure above was created in exercise xxx

The following is considered for this Rsyslog exercise

- 1) OSPF is implemented
- 2) R3 is connected to the cloud (Internet)
- 3) R3 has NAT configured
- 4) R3 is DHCP server for NW 192.168.4.0

The topology changes will be done according to the following tables (see in yellow the Fedora connection in SW2)

Move Fedora from being connected in R1 to be connected to SW2

3.3.3.3 Addressing table

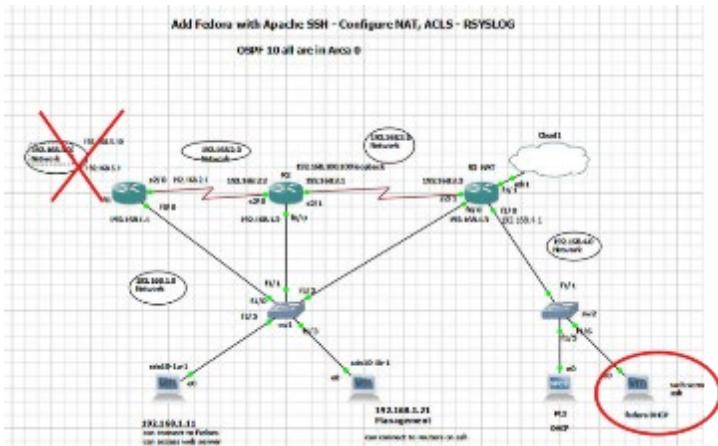
Network Element	Connection	Port	IP Address
R1	Connection to R2	S2/0	192.168.2.1/24
	Connection to SW1	F0/0	192.168.1.1/24
	Connection to Fedora1a	F1/0	192.168.5.1/24
R2	Connection to R1	S2/0	192.168.2.2/24

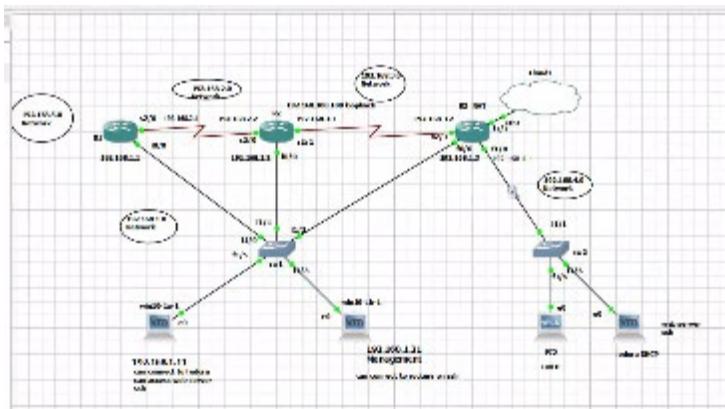
	Connection to R3	S2/1	192.168.3.1/24
	Connection to SW1	F0/0	192.168.1.2/24
R3	Connection to R2	S2/1	192.168.3.2/24
	Connection to SW1	F0/0	192.168.1.3/24
	Connection to SW2	F1/0	192.168.4.1/24
	Connection to Cloud	F1/1	DHCP
	Connection to R1	F1/0	N/A
SW1	Connection to R2	F1/1	N/A
	Connection to R3	F1/2	N/A
	Connection to win10-1a-1	F1/5	N/A
	Connection to win10-1b -1 Management	F1/3	N/A
	Connection to R3	F1/1	N/A
SW2	Connection to PC2	F1/5	N/A
	Connection to Fedora	F1/6	N/A

PC's and Linux boxes

win10-1a-1	Connection to SW1	NIC	192.168.1.11/24
win10-1b-1	Connection to SW1	NIC	192.168.1.21/24
PC2	Connection to SW2	NIC	DHCP-assigned
Fedora1a	Connection to SW2	NIC	DHCP-assigned

3.3.3.4 Change Fedora to be connected to SW2





Remove configuration for F1/0 connection between R1 and Fedora

```
R1#show ip interface brief
Interface          IP-Address      OK? Method Status      Protocol
FastEthernet0/0    192.168.1.1    YES NVRAM up        up
FastEthernet1/0    192.168.5.1    YES NVRAM up        up
FastEthernet1/1    unassigned     YES NVRAM administratively down down
Serial2/0          192.168.2.1    YES NVRAM up        up
Serial2/1          unassigned     YES NVRAM administratively down down
Serial2/2          unassigned     YES NVRAM administratively down down
Serial2/3          unassigned     YES NVRAM administratively down down
```

Remove Access list for 192.168.5.10 (Fedora connected to R1)

```
R1#show access-lists
Extended IP access list WIN10a-TO-FEDORA
  10 permit tcp host 192.168.1.11 host 192.168.5.10 eq 22
  20 permit tcp host 192.168.1.11 host 192.168.5.10 eq www
Extended IP access list WIN10b-SSH-ACCESS
  10 permit tcp host 192.168.1.21 any eq 22
R1#
```

!!!! R1 !!!!

```
enable
configure terminal
! Bring down the interface connected from R1 to Fedora
interface FastEthernet1/0
  no description Connection between R1 and Fedora
  no ip address 192.168.5.1 255.255.255.0
  no ip access-group WIN10a-TO-FEDORA out
  shutdown
  exit
! Remove the access list WIN10a-TO-FEDORA
no ip access-list extended WIN10a-TO-FEDORA
end
```

```
copy running-config startup-config
```

```
R1#!!!! R1 !!!!  
R1#enable  
R1#configure terminal  
Enter configuration commands, one per line. End with CNTL/Z.  
R1(config)#! Bring down the interface connected from R1 to Fedora  
R1(config)#interface FastEthernet1/0  
R1(config-if)# no description Connection between R1 and Fedora  
R1(config-if)# no ip address 192.168.5.1 255.255.255.0  
Invalid address  
R1(config-if)# no ip access-group WIN10a-TO-FEDORA out  
R1(config-if)# shutdown  
R1(config-if)# exit  
R1(config)#! Remove the access list WIN10a-TO-FEDORA  
R1(config)#no ip access-list extended WIN10a-TO-FEDORA  
R1(config)#end  
R1#
```

```
R1#show ip interface brief  
Interface          IP-Address      OK? Method Status          Protocol  
FastEthernet0/0    192.168.1.1    YES NVRAM  up           up  
FastEthernet1/0    unassigned     YES unset   administratively down down  
FastEthernet1/1    unassigned     YES NVRAM  administratively down down  
Serial2/0          192.168.2.1    YES NVRAM  up           up  
Serial2/1          unassigned     YES NVRAM  administratively down down  
Serial2/2          unassigned     YES NVRAM  administratively down down  
Serial2/3          unassigned     YES NVRAM  administratively down down  
R1#
```

```
R1#show access-l  
R1#show access-lists  
Extended IP access list WIN10b-SSH-ACCESS  
  10 permit tcp host 192.168.1.21 any eq 22  
R1#
```

3.3.3.4.1 Router R3

Remove no ip domain lookup

```
R3(config)#ip domain lookup  
R3(config)#+
```

Restart F1/1 interface

```
!!! Restart F1/1 interface in R3
enable
configure terminal
interface FastEthernet1/1
    shutdown
    no shutdown
exit
```

```
R3(config-if)#shutdown
R3(config-if)#no shut
*Mar 11 03:35:01.128: %LINK-5-CHANGED: Interface FastEthernet1/1, changed state to administratively down
*Mar 11 03:35:02.128: %LINEPROTO-5-UPDOWNN: Line protocol on Interface FastEthernet1/1, changed state to down
R3(config-if)#no shut
R3(config-if)#
R3(config-if)#
R3(config-if)#
R3(config-if)#
R3(config-if)#
*Mar 11 03:35:06.252: %LINK-3-UPDOWNN: Interface FastEthernet1/1, changed state to up
*Mar 11 03:35:07.256: %LINEPROTO-5-UPDOWNN: Line protocol on Interface FastEthernet1/1, changed state to up
R3(config-if)#
*Mar 11 03:35:08.000: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet1/1 assigned DHCP address 192.168.0.42, mask 255.255.0.0, hostname R3
R3(config-if)#end
R3#
*Mar 11 03:35:11.668: %SYS-5-CONFIG_I: Configured from console by console
R3#
R3#
```

3.3.3.4.2 SW2 Changes

Switch 2 modifications before connecting Fedora to SW2

Modify Switch 2 Interface to R3

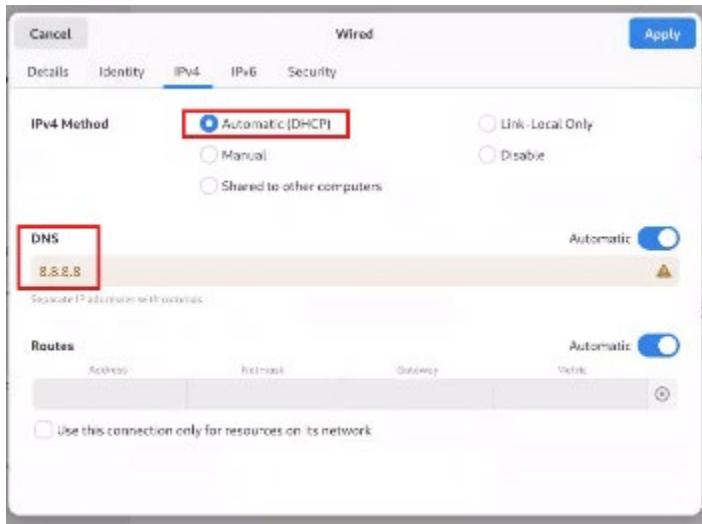
By assigning an IP address to VLAN 1, SW2 becomes a Layer 3 device for that VLAN.

This allows SW2 to communicate with R3 at the network layer (Layer 3), as both devices are now in the same subnet (192.168.4.0/24).

```
enable
configure terminal
interface Vlan1
    ip address 192.168.4.2 255.255.255.0
    no shutdown
exit
interface FastEthernet1/1
description Interface in SW2 connected to R3
! Set the port to access mode
switchport mode access
no shutdown
exit
end
```

3.3.3.4.3 Fedora

Assign fedora ip address via dhcp



Since we have DHCP assigned IP

Delete the hostname and ip reference for Fedora in hosts file input /etc/hosts

Turnoff firewall in Fedora

```
root@fedora1a:/etc# systemctl stop firewalld
root@fedora1a:/etc# systemctl disable firewalld
Removed '/etc/systemd/system/dbus-org.fedoraproject.FirewallD.service'.
Removed '/etc/systemd/system/multi-user.target.wants/firewalld.service'.
root@fedora1a:/etc# systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; preset: enabled)
   Drop-In: /usr/lib/systemd/system/service.d
             └─10-timeout-abort.conf, 50-keep-warm.conf
     Active: inactive (dead)
       Docs: man:firewalld(1)

Mar 18 12:00:08 fedora1a systemd[1]: Starting firewalld.service - firewalld - dynamic firewall daemon...
Mar 18 12:00:09 fedora1a systemd[1]: Started firewalld.service - firewalld - dynamic firewall daemon.
Mar 18 13:30:57 fedora1a systemd[1]: Stopping firewalld.service - firewalld - dynamic firewall daemon...
Mar 18 13:30:57 fedora1a systemd[1]: firewalld.service: Deactivated successfully.
Mar 18 13:30:57 fedora1a systemd[1]: Stopped firewalld.service - firewalld - dynamic firewall daemon.
root@fedora1a:/etc# firewall-cmd --state
not running
root@fedora1a:/etc#
```

Test ping google.com

```
root@fedora1a:/etc#
root@fedora1a:/etc# ping google.com
PING google.com (142.250.69.142) 56(84) bytes of data.
64 bytes from tzyula-ab-in-f14.1e100.net (142.250.69.142): icmp_seq=1 ttl=116 time=21.6 ms
64 bytes from tzyula-ab-in-f14.1e100.net (142.250.69.142): icmp_seq=2 ttl=116 time=21.1 ms
64 bytes from tzyula-ab-in-f14.1e100.net (142.250.69.142): icmp_seq=3 ttl=116 time=20.2 ms
64 bytes from tzyula-ab-in-f14.1e100.net (142.250.69.142): icmp_seq=4 ttl=116 time=30.4 ms
^C
--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3007ms
rtt min/avg/max/mdev = 20.159/23.317/30.485/4.124 ms
root@fedora1a:/etc#
```

3.3.3.5 Syslog

3.3.3.5.1 Server

Fedora Syslog Server

1. Start by installing rsyslog.

```
yum install rsyslog
```

```
[root@fedora8 ~]# yum install rsyslog
```

```
root@fedora8:/etc#
[root@fedora8 /etc]# yum install rsyslog -y
Updating and loading repositories:
  RPM Fusion for Fedora 41 - Nonfree - NVIDIA Driver
  RPM Fusion for Fedora 41 - Nonfree - Steam
  google-chrome
  Copr repo for PyCharm owned by phracek
  Fedora 41 - x86_64 - Updates
  Fedora 41 - x86_64
  RPM Fusion for Fedora 41 - Nonfree - NVIDIA Driver
  google-chrome
  Fedora 41 - x86_64 - Updates
  Repositories loaded.

Transaction Summary:
  Installing: 3 packages

Total size of inbound packages is 379 Kib. Need to download 879 Kib.
After this operation, 3 Mib extra will be used (install 3 Mib, remove 8 Kib).
[1/3] libusair-8.1.11-18.fc41.x86_64
[2/3] libfastjson-81.2384.8-5.fc41.x86_64
[3/3] rsyslog-8.8.2312.8-5.fc41.x86_64
[3/3] Total
Downloaded packages:
```

2. Once it is installed, enable and then start it. Check the status to confirm it's running correctly.

```
systemctl enable rsyslog
systemctl start rsyslog
systemctl status rsyslog
```

```
[root@fedora8 ~]# systemctl enable rsyslog
[root@fedora8 ~]# systemctl start rsyslog
[root@fedora8 ~]# systemctl status rsyslog
● rsyslog.service - System Logging Service
  Loaded: loaded (/usr/lib/systemd/system/rsysl
  Active: active (running) since Thu 2020-10-01
    Docs: man:rsyslog(3)
          man:rsyslogd(8)
```

```

root@federala:~# 
root@federala:/etc# systemctl enable rsyslog
root@federala:/etc# systemctl start rsyslog
root@federala:/etc# systemctl status rsyslog
● rsyslog.service - System Logging Service
   Loaded: loaded (/usr/lib/systemd/system/rsyslog.service; enabled; preset: enabled)
   Drop-In: /usr/lib/systemd/system/service.d
             └─18-timeout-abort.conf, 58-keep-warn.conf
     Active: active (running) since Tue 2025-03-11 09:25:48 EDT; 8s ago
   Invocation: 6984c559ecfe4c3d9b79fd48e79d271e
     Docs: man:rsyslogd(8)
           https://www.rsyslog.com/doc/
   Main PID: 206321 (rsyslogd)
     Tasks: 3 (limit: 9419)
    Memory: 10.6M (peak: 15.7M)
       CPU: 7.479s
      CGroup: /system.slice/rsyslog.service
                 └─206321 /usr/sbin/rsyslogd -n

Mar 11 09:25:47 federala systemd[1]: Starting rsyslog.service - System Logging Service...
Mar 11 09:25:48 federala systemd[1]: Started rsyslog.service - System Logging Service.
Mar 11 09:25:48 federala rsyslogd[206321]: [origin software="rsyslogd" swVersion="8.2312.0-5.fc41" x-pid="206321" x-info="https://www.rsyslog.com"] start
Mar 11 09:25:48 federala rsyslogd[206321]: imjournal: No statefile exists, /var/lib/rsyslog/imjournal.state will be created (ignore if this is first run)
Mar 11 09:25:49 federala rsyslogd[206321]: imjournal from <federala:kernel>: begin to drop messages due to rate-limiting

root@federala:/etc#

```

2. Check contents of /etc/syslog.conf

```
cat -n /etc/syslog.conf
```

```

root@federala:/etc# cat -n /etc/rsyslog.conf
1 # rsyslog configuration file
2
3 # For more information see /usr/share/doc/rsyslog-*/rsyslog_conf.html
4 # or latest version online at http://www.rsyslog.com/doc/rsyslog_conf.html
5 # If you experience problems, see http://www.rsyslog.com/doc/troubleshoot.html
6
7 ##### GLOBAL DIRECTIVES #####
8
9 # Where to place auxiliary files
10 global(workDirectory="/var/lib/rsyslog")
11
12 ##### MODULES #####
13
14 # Use default timestamp format
15 module(load="builtin:omfile" Template="RSYSLOG_TraditionalFileFormat")
16
17 module(load="imuxsock"      # provides support for local system logging (e.g. via logger command)
18         SysSock.Use="off") # Turn off message reception via local log socket;
19         # local messages are retrieved through imjournal now.
20 module(load="imjournal"      # provides access to the systemd journal
21         FileCreateMode="0600"      # Quiet warning and ensure privacy
22         StateFile="imjournal.state") # File to store the position in the journal
23
24 # Include all config files in /etc/rsyslog.d/
25 include(file="/etc/rsyslog.d/*.conf" mode="optional")
26
27 #module(load="imklog") # reads kernel messages (the same are read from journald)
28 #module(load="immark") # provides --MARK-- message capability
29
30 # Provides UDP syslog reception
31 # for parameters see http://www.rsyslog.com/doc/imudp.html
32 #module(load="imudp") # needs to be done just once
33 #input(type="imudp" port="514")
34
35 # Provides TCP syslog reception
36 # for parameters see http://www.rsyslog.com/doc/imtcp.html
37 #module(load="imtcp") # needs to be done just once
38 #input(type="imtcp" port="514")
39

```

```

40 ##### RULES #####
41
42 # Log all kernel messages to the console.
43 # Logging much else clutters up the screen.
44 #kern.*                                     /dev/console
45
46 # Log anything (except mail) of level info or higher.
47 # Don't log private authentication messages!
48 *.*;mail.none;authpriv.none;cron.none        /var/log/messages
49
50 # The authpriv file has restricted access.
51 authpriv.*                                    /var/log/secure
52
53 # Log all the mail messages in one place.
54 mail.*                                         -/var/log/maillog
55
56
57 # Log cron stuff
58 cron.*                                         /var/log/cron
59
60 # Everybody gets emergency messages
61 *.emerg                                         :omusrmsg:*
62
63 # Save news errors of level crit and higher in a special file.
64 uucp,news.crit                                /var/log/spooler
65
66 # Save boot messages also to boot.log
67 local7.*                                       /var/log/boot.log
68
69
70 # ##### sample forwarding rule #####
71 #action(type="omfwd"
72 # # An on-disk queue is created for this action. If the remote host is
73 # # down, messages are spooled to disk and sent when it is up again.
74 #queue.filename="fwdRule1"                      # unique name prefix for spool files
75 #queue.maxdiskspace="1g"                       # 1gb space limit (use as much as possible)
76 #queue.saveonshutdown="on"                     # save messages to disk on shutdown
77 #queue.type="LinkedList"                      # run asynchronously
78 #action.resumeRetryCount="-1"                  # infinite retries if host is down
79 # # Remote Logging (we use TCP for reliable delivery)
80 # # remote_host is: name/ip, e.g. 192.168.0.1, port optional e.g. 10514
81 #Target="remote_host" Port="XXX" Protocol="tcp")
root@fedoralalala:/etc#

```

3. Open the file '/etc/rsyslog.conf' for editing.

```
vi /etc/rsyslog.conf
```

```
[root@fedora8 ~]# vi /etc/rsyslog.conf
```

4. Set line numbers in vi

```
:set number
```

```
:set number
```

5. Locate the 'UDP syslog reception' section, and uncomment out the 'module' and 'input' lines.

```
/UDP  
  :/UDP
```

```
29  
30 # Provides UDP syslog reception  
31 # for parameters see http://www.rsyslog.com/doc/imudp.html  
32 module(load="imudp") # needs to be done just once  
33 [input(type="imudp" port="514")  
34
```

This indicate I will be running syslog on UDP and at port 514

6. Locate the ‘Log anything’ line and change the first word to ‘warn’.

```
45  
46 # Log anything (except mail) of level info or higher.  
47 # Don't log private authentication messages!  
48 *.warn;mail.none;authpriv.none;cron.none          /var/log/messages  
49
```

7. Save config file
8. Test to see if config file is correct

```
rsyslogd -f /etc/rsyslog.conf -N1
```

```
root@fedora1a:/etc# rsyslogd -f /etc/rsyslog.conf -N1  
rsyslogd: version 8.2312.0-5.fc41, config validation run (level 1), master config /etc/rsyslog.conf  
rsyslogd: End of config validation run. Bye.  
root@fedora1a:/etc# 
```

9. Check SELINUX config file

NOTE - You don't necessarily have to disable SELinux for the rsyslog server. However, SELinux can sometimes block certain features of rsyslog if the appropriate policies are not configured. This is because SELinux enforces strict access controls, which can interfere with rsyslog operations if the necessary permissions are not granted.

```
root@fedora1a:/etc# cat -n /etc/selinux/config  
1  
2 # This file controls the state of SELinux on the system.  
3 # SELINUX= can take one of these three values:  
4 #       enforcing - SELinux security policy is enforced.  
5 #       permissive - SELinux prints warnings instead of enforcing.  
6 #       disabled - No SELinux policy is loaded.  
7 # See also:  
8 # https://docs.fedoraproject.org/en-US/quick-docs/getting-started-with-selinux/#getting-  
started-with-selinux-selinux-states-and-modes  
9 #  
10 # NOTE: In earlier Fedora kernel builds, SELINUX=disabled would also
```

```

11 # fully disable SELinux during boot. If you need a system with SELinux
12 # fully disabled instead of SELinux running with no policy loaded, you
13 # need to pass selinux=0 to the kernel command line. You can use grubby
14 # to persistently set the bootloader to boot with selinux=0:
15 #
16 #     grubby --update-kernel ALL --args selinux=0
17 #
18 # To revert back to SELinux enabled:
19 #
20 #     grubby --update-kernel ALL --remove-args selinux
21 #
22 SELINUX=enforcing
23 # SELINUXTYPE= can take one of these three values:
24 #     targeted - Targeted processes are protected,
25 #     minimum - Modification of targeted policy. Only selected processes are protected.
26 #     mls - Multi Level Security protection.
27 SELINUXTYPE=targeted
28
29
root@fedora1a:/etc#

```

```

root@fedora1a:/etc# cat -n /etc/selinux/config
1
2 # This file controls the state of SELinux on the system.
3 # SELINUX can take one of these three values:
4 #     enforcing - SELinux security policy is enforced.
5 #     permissive - SELinux prints warnings instead of enforcing.
6 #     disabled - No SELinux policy is loaded.
7 # See also:
8 # https://docs.fedoraproject.org/en-US/quick-docs/getting-started-with-selinux/#getting-started-with-selinux-selinux-states-and-modes
9 #
10 # NOTE: In earlier Fedora kernel builds, SELINUX=disabled would also
11 # fully disable SELinux during boot. If you need a system with SELinux
12 # fully disabled instead of SELinux running with no policy loaded, you
13 # need to pass selinux=0 to the kernel command line. You can use grubby
14 # to persistently set the bootloader to boot with selinux=0:
15 #
16 #     grubby --update-kernel ALL --args selinux=0
17 #
18 # To revert back to SELinux enabled:
19 #
20 #     grubby --update-kernel ALL --remove-args selinux
21 #
22 SELINUX=enforcing
23 # SELINUXTYPE= can take one of these three values:
24 #     targeted - Targeted processes are protected,
25 #     minimum - Modification of targeted policy. Only selected processes are protected.
26 #     mls - Multi Level Security protection.
27 SELINUXTYPE=targeted
28
29
root@fedora1a:/etc#

```

8. Ensure that selinux is disabled.

If changed a reboot is needed.

```
[root@fedora8 ~]# vi /etc/selinux/config
```

```
20 #     grubby --update-kernel ALL --remove-args selinux
21 #
22 SELINUX=disabled
23 # SELINUXTYPE= can take one of these three values:
24 #     targeted - Targeted processes are protected,
25 #     minimum - Modification of targeted policy. Only se
26 #     mls - Multi Level Security protection.
27 SELINUXTYPE=targeted
28
```

sestatus

```
29
root@fedora1a:/etc# sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:              targeted
Current mode:                   enforcing
Mode from config file:          disabled
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:    actual (secure)
Max kernel policy version:     33
root@fedora1a:/etc#
root@fedora1a:/etc# reboot
```

10. Restart the rsyslog service.

systemctl restart rsyslog.service

```
root@fedora1a:/etc# systemctl restart rsyslog.service
[ ok ] Restarting rsyslog.service: rsyslog.service
```

11. See the port is open for rsyslog

netstat -anup

```
udp6      0      0 ::1:514          :::*          277111/rsyslogd
tcp6      0      0 ::1:5253          :::*          777/auditd
```

```

root@fedoraia:/etc# systemctl restart rsyslog.service
root@fedoraia:/etc# netstat -anup
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address          Foreign Address        State      PID/Program name
udp     0      0 127.0.0.54:53            0.0.0.0:*              ESTABLISHED  172856/systemd-reso
udp     0      0 127.0.0.53:53            0.0.0.0:*              ESTABLISHED  172856/systemd-reso
udp     0      0 192.168.4.12:68          192.168.4.1:67        ESTABLISHED  903/NetworkManager
udp     0      0 127.0.0.1:323           0.0.0.0:*              ESTABLISHED  800/chronynd
udp     0      0 0.0.0.0:514             0.0.0.0:*              ESTABLISHED  277111/rsyslogd
udp     0      0 0.0.0.0:54084           0.0.0.0:*              ESTABLISHED  773/avahi-daemon: r
udp     0      0 0.0.0.0:5353            0.0.0.0:*              ESTABLISHED  773/avahi-daemon: r
udp     0      0 0.0.0.0:5355            0.0.0.0:*              ESTABLISHED  172856/systemd-reso
udp     0      0 0.0.0.0:38513           0.0.0.0:*              ESTABLISHED  4144/python3
udp     0      0 192.168.4.12:3702         0.0.0.0:*              ESTABLISHED  4144/python3
udp     0      0 239.255.255.250:3702        0.0.0.0:*              ESTABLISHED  4144/python3
udp6    0      0 :::36961               ::::*                  ESTABLISHED  773/avahi-daemon: r
udp6    0      0 ::1:323                ::::*                  ESTABLISHED  800/chronynd
udp6    0      0 ::::514                ::::*                  ESTABLISHED  277111/rsyslogd
udp6    0      0 ::::5353               ::::*                  ESTABLISHED  773/avahi-daemon: r
udp6    0      0 ::::5355               ::::*                  ESTABLISHED  172856/systemd-reso
udp6    0      0 ::::48574               ::::*                  ESTABLISHED  4144/python3
udp6    0      0 fe80::86a6:f22a:ee:3702  ::::*                  ESTABLISHED  4144/python3
udp6    0      0 ff82::c:3702            ::::*                  ESTABLISHED  4144/python3
udp6    0      0 ff82::c:3702            ::::*                  ESTABLISHED  4144/python3
root@fedoraia:/etc#

```

3.3.3.5.2 Client

Router configuration R3

- Configure the basic syslog configuration including IP of the syslog server, trap level, and the source interface of the messages.

```

> logging SYSLOG_SERVER_IP
> logging trap #
> logging source-interface INT

```

```

R3(config)#logging 192.168.4.12
R3(config)#loggin
R3(config)#logging
*Mar 11 22:37:02.063: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 192.168.4.12 port 514 started -
CLI initiated
R3(config)#logging tra
R3(config)#logging trap ?
<0-7>          Logging severity level
alerts          Immediate action needed          (severity=1)
critical        Critical conditions          (severity=2)
debugging       Debugging messages          (severity=7)
emergencies     System is unusable          (severity=0)
errors          Error conditions          (severity=3)
informational   Informational messages      (severity=6)
notifications   Normal but significant conditions (severity=5)
warnings        Warning conditions          (severity=4)
<cr>

```

```

R3(config)#logging trap 4
R3(config)#logging source-interface f1/0
R3(config)#end

```

```

R3#
*Mar 11 23:07:33.159: %SYS-5-CONFIG_I: Configured from console by console
R3#
R3(config)#logging 192.168.4.12
R3(config)#login
R3(config)#logging
*Mar 11 22:37:02.063: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 192.168.4.12 port 514 started - CLI initiated
R3(config)#logging trap
R3(config)#logging trap ?
<0-7>          Logging severity level
alerts           Immediate action needed      (severity=1)
critical         Critical conditions        (severity=2)
debugging        Debugging messages         (severity=7)
emergencies      System is unusable       (severity=0)
errors           Error conditions         (severity=3)
informational    Informational messages   (severity=6)
notifications   Normal but significant conditions (severity=5)
warnings         Warning conditions       (severity=4)
<cr>

R3(config)#logging trap 4
R3(config)#logging sou
R3(config)#logging source-interface f 0/1
^
% Invalid input detected at '^' marker.

R3(config)#logging source-interface f1/0
R3(config)#end
R3#
*Mar 11 23:07:33.159: %SYS-5-CONFIG_I: Configured from console by console
R3#

```

2. Configure the router to timestamp syslog messages, as it does not do this by default.

> service timestamps log datetime

```

R3#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#service times
R3(config)#service timestamps log date
R3(config)#service timestamps log datetime
R3(config)#end
R3#
*Mar 12 00:12:06: %SYS-5-CONFIG_I: Configured from console by console
R3#

```

3. Complete the same steps on the other routers.

3.3.3.6 Testing

1. On fedora, you can view syslog messages live.

Since there is a lot of messages in /var/log/messages, we will filter on LINK to see link changes in R3.

```
tail -f /var/log/messages | grep LINK
```

```
root@fedora1a:/var/log# tail -f /var/log/messages | grep LINK
```

2. Go to R3 and restart an interface. You should receive a message in Fedora.

Selected interface is S2/0 currently unused

```
configure terminal  
interface s2/0  
no shutdown  
shutdown  
exit
```

```
R3#show ip interface brief  
Interface          IP-Address      OK? Method Status          Protocol  
FastEthernet0/0    192.168.1.3    YES NVRAM  up           up  
FastEthernet1/0    192.168.4.1    YES NVRAM  up           up  
FastEthernet1/1    10.164.0.42   YES DHCP   up           up  
Serial2/0          unassigned     YES NVRAM  administratively down down  
Serial2/1          192.168.3.2    YES NVRAM  up           up  
Serial2/2          unassigned     YES NVRAM  administratively down down  
Serial2/3          unassigned     YES NVRAM  administratively down down  
R3#config t  
Enter configuration commands, one per line. End with CNTL/Z.  
R3(config)#interface s2/0  
R3(config-if)#no shut  
R3(config-if)#no shutdown  
R3(config-if)#shut f  
*Mar 12 00:20:02: %LINK-3-UPDOWN: Interface Serial2/0, changed state to up  
R3(config-if)#shut f  
*Mar 12 00:20:03: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up  
R3(config-if)#shut down  
^  
% Invalid input detected at '^' marker.  
  
R3(config-if)#  
*Mar 12 00:20:08: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 192.168.4.12 port 514 started - reconnection  
R3(config-if)#shutdown  
R3(config-if)#end  
R3#  
*Mar 12 00:20:19: %SYS-5-CONFIG_I: Configured from console by console  
R3#  
*Mar 12 00:20:20: %LINK-5-CHANGED: Interface Serial2/0, changed state to administratively down  
*Mar 12 00:20:21: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down  
R3#
```

```

*Mar 12 00:19:16: %SYS-5-CONFIG_I: Configured from console byshow ip interface brief
R3#show ip interface brief
Interface          IP-Address      OK? Method Status        Protocol
FastEthernet0/0    192.168.1.3   YES NVRAM up           up
FastEthernet1/0    192.168.4.1   YES NVRAM up           up
FastEthernet1/1    10.164.0.42   YES DHCP  up           up
Serial2/0          unassigned    YES NVRAM administratively down down
Serial2/1          192.168.3.2   YES NVRAM up           up
Serial2/2          unassigned    YES NVRAM administratively down down
Serial2/3          unassigned    YES NVRAM administratively down down
R3#config t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#interface s2/0
R3(config-if)#no shut
R3(config-if)#no shutdown
R3(config-if)#shut f
*Mar 12 00:20:02: %LINK-3-UPDOWN: Interface Serial2/0, changed state to up
R3(config-if)#shut f
*Mar 12 00:20:03: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
R3(config-if)#shut down
^
% Invalid input detected at '^' marker.

R3(config-if)#
*Mar 12 00:20:08: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 192.168.4.12 port 514 started - reconnection
R3(config-if)#shutdown
R3(config-if)#end
R3#
*Mar 12 00:20:19: %SYS-5-CONFIG_I: Configured from console by console
R3#
*Mar 12 00:20:20: %LINK-5-CHANGED: Interface Serial2/0, changed state to administratively down
*Mar 12 00:20:21: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
R3#

```

3. See live messages in Fedora syslog server related to LINK

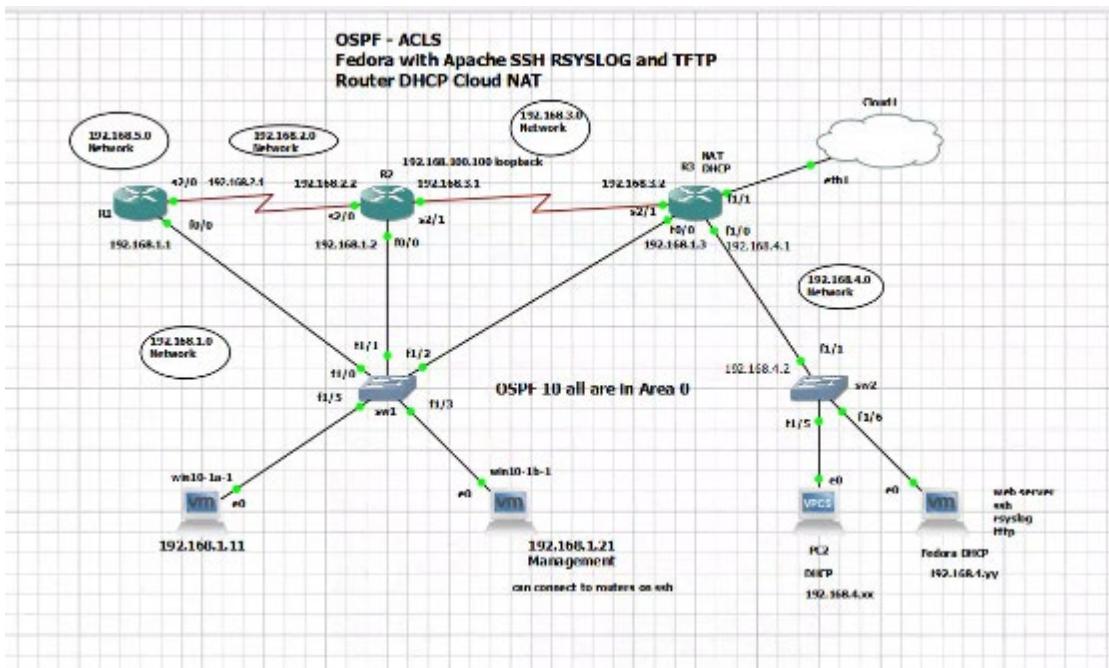
```

^C
root@fedora1a:/var/log# tail -f /var/log/messages | grep LINK
Mar 11 20:20:16 _gateway 45: *Mar 12 00:20:02: %LINK-3-UPDOWN: Interface Serial2/0, changed state to up
Mar 11 20:20:16 _gateway 45: *Mar 12 00:20:02: %LINK-3-UPDOWN: Interface Serial2/0, changed state to up
^C

```

3.3.4 OSPF - ACLS , Fedora with Apache SSH RSYSLOG and TFTP , Router DHCP Cloud NAT

3.3.4.1 Topology



3.3.4.2 Fedora – TFTP server

1. Install the tftp-server by entering `dnf install tftp-server tftp -y`

```
password:
root@fedora1a:~# dnf install tftp-server tftp -y
```

```
student@fedora1a:~$ su -
Password:
root@fedora1a:~# dnf install tftp-server tftp -y
Updating and loading repositories:
  Fedora 41 - x86_64 - Updates
  Fedora 41 openH264 (From Cisco) - x86_64
Repositories loaded.
Package           Arch      Version       Repository      Size
Installing:
  tftp            x86_64    5.2-44.fc41   fedora          32.7 KIB
  tftp-server     x86_64    5.2-44.fc41   fedora          64.1 KIB
Transaction Summary:
  Installing: 2 packages

Total size of inbound packages is 97 KIB. Need to download 73 KIB.
After this operation, 117 KIB extra will be used (install 117 KIB, remove 9 KIB).
[1/2] tftp-5.2-44.fc41.x86_64                                160B |  9.5 KIB/s | 32.7 KIB | 60x00s
[2/2] tftp-server-5.2-44.fc41.x86_64                            160B |  8.3 KIB/s | 46.3 KIB | 60x00s
[2/2] Total                                                 160B |  9.5 KIB/s | 72.0 KIB | 60x00s
Running transaction
[1/4] Verify package files                                     160B | 236.3 B/s | 2.8 B | 60x00s
[2/4] Prepare transaction                                    160B |  5.3 B/s | 2.8 B | 60x00s
[3/4] Installing tftp-5.2-44.fc41.x86_64                   160B |  1.7 MIB/s | 54.1 KIB | 60x00s
[4/4] Installing tftp-server-5.2-44.fc41.x86_64             160B |  44.5 KIB/s | 66.1 KIB | 60x01s
Complete!
```

2. Locate the tftp.service and .socket files in /usr/lib/systemd/system

```
ls -l /usr/lib/systemd/system/tftp*
```

```
root@fedora1a:~# ls -l /usr/lib/systemd/system/tftp*
-rw-r--r--. 1 root root 189 Jul 19 2024 /usr/lib/systemd/system/tftp.service
-rw-r--r--. 1 root root 112 Jul 19 2024 /usr/lib/systemd/system/tftp.socket
root@fedora1a:~#
```

3. Copy files

- /usr/lib/systemd/system/tftp.service
- /usr/lib/systemd/system/tftp.socket

to /etc/systemd/system directory.

```
cp /usr/lib/systemd/system/tftp.service /etc/systemd/system/tftp-server.service
```

```
cp /usr/lib/systemd/system/tftp.socket /etc/systemd/system/tftp-server.socket
```

4. Verify files are copied

```
ls -l /etc/systemd/system/tftp*
```

```
root@fedora1a:~# ls -l /etc/systemd/system/tftp*
-rw-r--r--. 1 root root 189 Mar 11 21:00 /etc/systemd/system/tftp-server.service
-rw-r--r--. 1 root root 112 Mar 11 21:00 /etc/systemd/system/tftp-server.socket
root@fedora1a:~#
```

5. Edit the files

Verify contents of the files before editing

```
root@fedora1a:~# cat /etc/systemd/system/tftp-server.service
[Unit]
Description=Tftp Server
Requires=tftp.socket
Documentation=man:in.tftpd
```

```
[Service]
ExecStart=/usr/sbin/in.tftpd -s /var/lib/tftpboot
StandardInput=socket

[Install]
Also=tftp.socket
```

```
root@fedora1a:~# cat /etc/systemd/system/tftp-server.socket
[Unit]
Description=Tftp Server Activation Socket

[Socket]
ListenDatagram=69

[Install]
WantedBy=sockets.target
root@fedora1a:~#
```

```
root@fedora1a:~#
root@fedora1a:~# cat /etc/systemd/system/tftp-server.service
[Unit]
Description=Tftp Server
Requires=tftp.socket
Documentation=man:in.tftpd

[Service]
ExecStart=/usr/sbin/in.tftpd -s /var/lib/tftpboot
StandardInput=socket

[Install]
Also=tftp.socket
root@fedora1a:~#
```

```
Also=tftp.socket
root@fedora1a:~# cat /etc/systemd/system/tftp-server.socket
[Unit]
Description=Tftp Server Activation Socket

[Socket]
ListenDatagram=69

[Install]
WantedBy=sockets.target
root@fedora1a:~#
```

6. Edit the **tftp-server.service** file we just copied

```
[Unit]
Description=Tftp Server
Requires=tftp.socket
Documentation=man:in.tftpd

[Service]
ExecStart=/usr/sbin/in.tftpd -c -p -s /var/lib/tftpboot
StandardInput=socket

[Install]
WantedBy=multi-user.target
Also=tftp-server.socket
```

The file should look like below after the changes

```
GNU nano 8.1
[Unit]
Description=Tftp Server
Requires=tftp-server.socket
Documentation=man:in.tftpd

[Service]
ExecStart=/usr/sbin/in.tftpd -c -p -s /var/lib/tftpboot
StandardInput=socket

[Install]
WantedBy=multi-user.target
Also=tftp-server.socket
```

User CTRL-S and CTRL-X to save and exit

7. Verify the changes

```
cat /etc/systemd/system/tftp-server.service
```

```
root@fedoralab:~# cat /etc/systemd/system/tftp-server.service
[Unit]
Description=Tftp Server
Requires=tftp.socket
Documentation=man:in.tftpd

[Service]
ExecStart=/usr/sbin/in.tftpd -c -p -s /var/lib/tftpboot
StandardInput=socket

[Install]
WantedBy=multi-user.target
Also=tftp-server.socket
root@fedoralab:~#
```

[Unit]

Description: Provides a brief description of the service. In this case, it describes the service as "Tftp Server."

Requires: Specifies that this service requires the tftp.socket to be active and running. If the tftp.socket is not active, the tftp-server service will not start.

Documentation: Provides a reference to the manual page for in.tftpd, which is the TFTP daemon.

[Service]

ExecStart: Specifies the command to start the TFTP daemon.

The options -c -p -s /var/lib/tftpboot configure the TFTP server as follows:

-c: Allows new files to be created.

-p: Use the "secure" TFTP mode, running TFTP only as a user with restricted privileges.

-s /var/lib/tftpboot: Specifies the root directory for TFTP server files.

StandardInput: Indicates that the service should use a socket for its standard input.

[Install]

WantedBy: Specifies the target to which this service should be enabled. The `multi-user.target` is a common target for multi-user systems without a graphical interface.

Also: Specifies that the `tftp-server.service` also needs to enable the `tftp-server.socket` unit.

Summary

This configuration file sets up the TFTP server to start with specific options and ensures it relies on the `tftp.socket` to function properly. It integrates the service with the `systemd` init system, making it possible to manage it using standard `systemctl` commands like `systemctl`.

8. Do `systemctl daemon-reload`

Systemctl daemon-reload

```
root@fedora1a:~# systemctl daemon-reload
```

9. Enable and start the TFTP server service immediately

`systemctl enable --now tftp-server`

```
root@fedora1a:~# systemctl enable --now tftp-server
Created symlink '/etc/systemd/system/multi-user.target.wants/tftp-server.service' → '/etc/systemd/system/tftp-server.service'.
Created symlink '/etc/systemd/system/sockets.target.wants/tftp-server.socket' → '/etc/systemd/system/tftp-server.socket'.
root@fedora1a:~#
```

10. Change the user privileges with `sudo chmod 777 /var/lib/tftpboot`

** `/var/lib/tftpboot`: The root directory for TFTP server files

```
root@fedora1a:~# ls -lqrtha /var/lib/tftp*
total 0
drwxr-xr-x. 1 root root 0 Jul 19 2024 .
drwxr-xr-x. 1 root root 1022 Mar 11 20:52 ..
root@fedora1a:~#
root@fedora1a:~#
```

```
root@fedora1a:~# sudo chmod 777 /var/lib/tftpboot
root@fedora1a:~#
root@fedora1a:~# ls -lqrtha /var/lib/tftp*
total 0
drwxrwxrwx. 1 root root 0 Jul 19 2024 .
drwxr-xr-x. 1 root root 1022 Mar 11 20:52 ..
root@fedora1a:~#
```

3.3.4.3 TEST

3.3.4.3.1 FEDORA prechecks

1. Create an empty file in the TFTP server's root directory. This file will receive the router's configuration.

```
root@fedora1a:/var/lib/tftpboot# touch r3_copy
```

```
drwxrwxrwx. 1 root root 70 Mar 11 23:41 .
root@fedora1a:/var/lib/tftpboot# touch r3_copy
root@fedora1a:/var/lib/tftpboot# ls -lqrtha
```

2. List the contents of the TFTP root directory to confirm the file was created.

```
root@fedora1a:/var/lib/tftpboot# ls -lqrtha
```

```
root@fedora1a:/var/lib/tftpboot# ls -lqrtha
total 4.0K
drwxr-xr-x. 1 root root 1022 Mar 11 23:20 ..
-rw-r--r--. 1 root root 14 Mar 11 23:28 test_tftpboot.txt
-rw-r--r--. 1 root root 0 Mar 11 23:41 test_tftp_file.txt
-rwxrwxrwx. 1 root root 0 Mar 11 23:46 r3_copy
drwxrwxrwx. 1 root root 84 Mar 11 23:46 .
```

3. Change the file's permissions to allow the TFTP server to write the router configuration to it.

```
root@fedora1a:/var/lib/tftpboot# chmod 777 r3_copy
```

```
root@fedora1a:/var/lib/tftpboot# chmod 777 r3_copy
```

4. List the directory contents again to check the updated permissions.

```
root@fedora1a:/var/lib/tftpboot# ls -lqrtha
```

```
root@fedora1a:/var/lib/tftpboot# ls -lqrtha
total 4.0K
drwxr-xr-x. 1 root root 1022 Mar 11 23:20 ..
-rw-r--r--. 1 root root 14 Mar 11 23:28 test_tftpboot.txt
-rw-r--r--. 1 root root 0 Mar 11 23:41 test_tftp_file.txt
-rwxrwxrwx. 1 root root 0 Mar 11 23:46 r3_copy
drwxrwxrwx. 1 root root 84 Mar 11 23:46 .
```

5. Verify TFTP service is running on the Fedora server.

```
root@fedora1a:/var/lib/tftpboot# systemctl status tftp
```

```
root@fedora1a:/var/lib/tftpboot# systemctl status tftp
● tftp.service - Tftp Server
  Loaded: loaded (/usr/lib/systemd/system/tftp.service; indirect; preset: disabled)
  Drop-In: /usr/lib/systemd/system/service.d
            └─10-timeout-abort.conf, 50-keep-warm.conf
    Active: active (running) since Tue 2025-03-11 23:04:51 EDT; 43min ago
      Invocation: 49374d48fd66413492b6e24c3e3c0b7f
  TriggeredBy: ● tftp.socket
    Docs: man:in.tftpd
    Main PID: 329252 (in.tftpd)
      Tasks: 1 (limit: 9419)
     Memory: 208K (peak: 1M)
        CPU: 27ms
      CGroup: /system.slice/tftp.service
              └─329252 /usr/sbin/in.tftpd -s /var/lib/tftpboot

Mar 11 23:04:51 fedora1a systemd[1]: Started tftp.service - Tftp Server.
Mar 11 23:23:19 fedora1a in.tftpd[332090]: Client ::1 finished test_tftp_get_file.txt
```

6. Display the service file to understand the TFTP server's settings, especially the root directory.

```
root@fedora1a:/var/lib/tftpboot# systemctl cat tftp.service
```

```
root@fedora1a:/var/lib/tftpboot# systemctl cat tftp.service
# /usr/lib/systemd/system/tftp.service
```

```
root@fedora1a:/var/lib/tftpboot# systemctl cat tftp.service
# /usr/lib/systemd/system/tftp.service
[Unit]
Description=Tftp Server
Requires=tftp.socket
Documentation=man:in.tftpd

[Service]
ExecStart=/usr/sbin/in.tftpd -s /var/lib/tftpboot
StandardInput=socket

[Install]
Also=tftp.socket

# /usr/lib/systemd/system/service.d/10-timeout-abort.conf
# This file is part of the systemd package.
# See https://fedoraproject.org/wiki/Changes/Shorter_Shutdown_Timer.
#
# To facilitate debugging when a service fails to stop cleanly,
# TimeoutStopFailureMode=abort is set to "crash" services that fail to
stop in
# the time allotted. This will cause the service to be terminated with
SIGABRT
# and a coredump to be generated.
#
# To undo this configuration change, create a mask file:
#   sudo mkdir -p /etc/systemd/system/service.d
#   sudo ln -sv /dev/null /etc/systemd/system/service.d/10-timeout-
abort.conf

[Service]
TimeoutStopFailureMode=abort

# /usr/lib/systemd/system/service.d/50-keep-warm.conf
# Disable freezing of user sessions to work around kernel bugs.
# See https://bugzilla.redhat.com/show_bug.cgi?id=2321268
[Service]
Environment=SYSTEMD_SLEEP_FREEZE_USER_SESSIONS=0
```

7. Check the security context to ensure SELinux is not blocking TFTP access.

```
root@fedora1a:/var/lib/tftpboot# ls -Z /var/lib/tftpboot/r3_copy
```

It should show tftpdirc_rw_t.

```
root@fedora1a:/var/lib/tftpboot# ls -Z /var/lib/tftpboot/r3_copy
unconfined_u:object_r:tftpdirc_rw_t:s0 /var/lib/tftpboot/r3_copy
```

8. Verify if the firewall is running and if the TFTP port is open. In this case the firewall is not running.

```
root@fedora1a:/var/lib/tftpboot# firewall-cmd --list-all
root@fedora1a:/var/lib/tftpboot# firewall-cmd --list-all
FirewallD is not running
```

9. Checking the SELinux status to see if it is enabled.

```
root@fedora1a:/var/lib/tftpboot# sestatus
```

```
root@fedora1a:/var/lib/tftpboot# sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   permissive ▲
Mode from config file:          disabled
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:    actual (secure)
Max kernel policy version:     33
```

If current mode is not permissive, change the mode without requiring a reboot (it will be reset upon the next system restart).

```
sudo setenforce 0
```

This command changes the SELinux mode to "Permissive". You can verify the change by running:

```
getenforce
```

```
root@fedora1a:/var/lib/tftpboot# getenforce
Permissive
root@fedora1a:/var/lib/tftpboot# █
```

3.3.4.3.2 ROUTER

1. Ping the TFTP server
R3#ping 192.168.4.12

```
R3#ping 192.168.4.12
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.4.12, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/36/96 ms
```

2. On the R3 router, we start the process of copying the running configuration to the TFTP server.

```
R3#copy running-config tftp
Address or name of remote host []? 192.168.4.12
Destination filename [r3-config]? r3_copy
!! 2481 bytes copied in 2.012 secs (1233 bytes/sec)
```

```
R3#
R3#copy running-config tftp
Address or name of remote host []? 192.168.4.12
Destination filename [r3-config]? r3_copy
!!
2481 bytes copied in 2.012 secs (1233 bytes/sec)
R3#
```

3.3.4.3.3 Fedora check after the transfer:

1. List the directory contents again to see the size of the configuration file after the router's transfer.

```
root@fedora1a:/var/lib/tftpboot# ls -lqrtha
```

```
MAX Kernel policy version.      55
root@fedora1a:/var/lib/tftpboot# ls -lqrtha
total 8.0K
drwxr-xr-x. 1 root root 1022 Mar 11 23:20 ..
-rw-r--r--. 1 root root   14 Mar 11 23:28 test_tftpboot.txt
-rw-r--r--. 1 root root    0 Mar 11 23:41 test_tftp_file.txt
drwxrwxrwx. 1 root root   84 Mar 11 23:46 .
-rwxrwxrwx. 1 root root 2.5K Mar 11 23:51 r3_copy
```

2. View the contents of the r3_copy file, which now contains the router's running configuration.

```
root@fedora1a:/var/lib/tftpboot# cat r3_copy
```

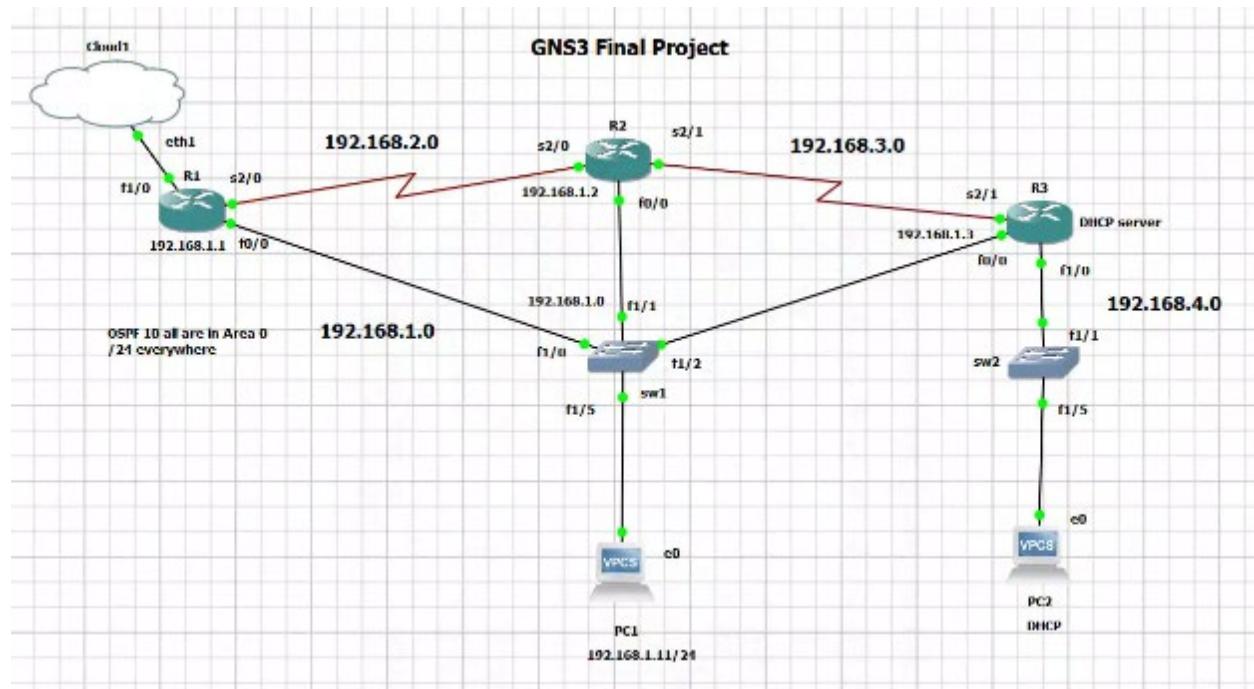
```

root@fedoralala:/var/lib/tftpboot# cat r3_copy

!
! Last configuration change at 00:20:19 UTC Wed Mar 12 2025
!
version 15.2
service timestamps debug datetime msec
service timestamps log datetime
!
```

3.3.5 Final GNS3 project CCNA 3

Topology



Network Element	Connection	Port	IP Address
R1	Connection to R2	S2/0	192.168.2.1/24
	Connection to SW1	F0/0	192.168.1.1/24

	Connection to Cloud	F1/0	DHCP
R2	Connection to R1	S2/0	192.168.2.2/24
	Connection to R3	S2/1	192.168.3.1/24
	Connection to SW1	F0/0	192.168.1.2/24
	Loopback	Lo0	192.168.100.100/32
R3	Connection to R2	S2/1	192.168.3.2/24
	Connection to SW1	F0/0	192.168.1.3/24
	Connection to SW2	F1/0	192.168.4.1/24
SW1	Connection to R1	F1/0	N/A
	Connection to R2	F1/1	N/A
	Connection to R3	F1/2	N/A
	Connection to PC1	F1/5	N/A
SW2	Connection to R3	F1/0	N/A
	Connection to PC2	F1/5	N/A

PC's

PC1	Connection to SW1	NIC	192.168.1.11/24
PC2	Connection to SW2	NIC	DHCP (192.168.4.0/24)

Configure topology

5. Create topology on GNS3 based on the diagram and addressing table
6. Basic configuration of network elements.

```
!!! ROUTER R1
enable
configure terminal
hostname R1
no ip domain lookup
banner motd #WARNING Authorized Users Only! #
end
write memory
```

```
!!! ROUTER R2
enable
configure terminal
hostname R2
no ip domain lookup
banner motd #WARNING Authorized Users Only! #
end
write memory
```

```
!!! ROUTER R3
enable
configure terminal
hostname R3
no ip domain lookup
banner motd #WARNING Authorized Users Only! #
end
write memory
```

```
!!! SWITCH SW1
enable
configure terminal
hostname SW1
no ip domain lookup
banner motd #WARNING Authorized Users Only! #
end
write memory
```

```
!!! SWITCH SW2
enable
configure terminal
hostname SW2
no ip domain lookup
banner motd #WARNING Authorized Users Only! #
end
write memory
```

7. Configure the IPv4 address on PC1 to 192.168.1.11 with a default gateway of 192.168.1.1, then save the configuration to NVRAM.

```
PC1> ip 192.168.1.11 255.255.255.0 192.168.1.1
PC1> save
PC1> show
```

8. Configure routers

```
!!!!!!!
!!!! Router R1:
!!!!!!!
enable
conf t

int s2/0
description Connection between R1 and R2
ip address 192.168.2.1 255.255.255.0
no shutdown
exit

int f0/0
description Connection between R1 and SW1
ip address 192.168.1.1 255.255.255.0
no shutdown
exit

int f1/0
description Connection between R1 and SW1
ip address dhcp
no shutdown
exit
end
write memory
```

show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	192.168.1.1	YES	manual	up	up
FastEthernet1/0	10.164.0.248	YES	DHCP	up	up
FastEthernet1/1	unassigned	YES	unset	administratively down	down
Serial2/0	192.168.2.1	YES	manual	up	up
Serial2/1	unassigned	YES	unset	administratively down	down
Serial2/2	unassigned	YES	unset	administratively down	down
Serial2/3	unassigned	YES	unset	administratively down	down

Note the IP assigned to F1/0.

```
R1# Mar 18 17:25:31.867: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet1/0 assigned DHCP address 10.164.0.248, mask 255.255.0.0, hostname R1
```

The IP assigned via DHCP is 10.164.0.248 the default gateway is assigned as gateway of last resort automatically.

```

R1#show ip route
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, I - LISP
      + - replicated route, % - next hop override

Gateway of last resort is 10.164.0.1 to network 0.0.0.0

S*   0.0.0.0/0 [254/0] via 10.164.0.1
    10.0.0.8 is variably subnetted, 3 subnets, 2 masks
S     10.162.240.52/32 [254/0] via 10.164.0.1, FastEthernet1/0
C     10.164.0.0/16 is directly connected, FastEthernet1/0
L     10.164.0.248/32 is directly connected, FastEthernet1/0
    192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.1.0/24 is directly connected, FastEthernet0/0
L     192.168.1.1/32 is directly connected, FastEthernet0/0
R1#

```

show dhcp lease

```

R1#show dhcp lease
Temp IP addr: 10.164.0.248  for peer on Interface: FastEthernet1/0
Temp sub net mask: 255.255.0.0
  DHCP Lease server: 10.162.240.52, state: 5 Bound
  DHCP transaction id: 10CE
  Lease: 3600 secs, Renewal: 1800 secs, Rebind: 3150 secs
Temp default-gateway addr: 10.164.0.1
Next timer fires after: 00:17:46
Retry count: 0  Client-ID: cisco-ca01.79c1.001c-Fa1/0
Client-ID hex dump: 636973636F2D636130312E373963312E
                           303031632D4661312F30
  Hostname: R1
R1#
R1#

```

```

!!!!!!!!!!!!!!!
!!! Router R2:
!!!!!!!!!!!!!!!
enable
conf t

int s2/0
description Connection between R2 and R1
ip address 192.168.2.2 255.255.255.0
no shutdown
exit

int s2/1
description Connection between R2 and R3
ip address 192.168.3.1 255.255.255.0
no shutdown
exit

```

```
int f0/0
description Connection between R2 and SW1
ip add 192.168.1.2 255.255.255.0
no shut
exit
end
write memory
```

```
show ip interface brief
```

```
R2#show ip interface brief
Interface          IP-Address      OK? Method Status          Protocol
FastEthernet0/0    192.168.1.2    YES manual up           up
FastEthernet1/0    unassigned      YES unset administratively down down
FastEthernet1/1    unassigned      YES unset administratively down down
Serial2/0          192.168.2.2    YES manual up           up
Serial2/1          192.168.3.1    YES manual up           up
Serial2/2          unassigned      YES unset administratively down down
Serial2/3          unassigned      YES unset administratively down down
R2#
*Mar 18 17:29:42.415: %LINEPROTO-5-UPDOWN: Line protocol on interface Serial2/1, changed state to down
R2#
```

```
!!!!!!!!!!!!!!
!!! Router R3:
!!!!!!!!!!!!!!
enable
conf t
```

```
int s2/1
description Connection between R3 and R2
ip add 192.168.3.2 255.255.255.0
no shut
```

```
int f0/0
description Connection between R3 and SW1
ip add 192.168.1.3 255.255.255.0
no shut
```

```
int f1/0
description Connection between R3 and SW2
ip add 192.168.4.1 255.255.255.0
no shut
exit
end
write memory
```

```
show ip interface brief
```

```
R3#show ip interface brief
Interface          IP-Address      OK? Method Status      Protocol
FastEthernet0/0    192.168.1.3    YES manual up       up
FastEthernet1/0    192.168.4.1    YES manual up       up
FastEthernet1/1    unassigned     YES unset administratively down down
Serial2/0          unassigned     YES unset administratively down down
Serial2/1          192.168.3.2    YES manual up       up
Serial2/2          unassigned     YES unset administratively down down
Serial2/3          unassigned     YES unset administratively down down
R3#
```

Configure DHCP

5. Configure R3 as DHCP server

```
!!!!!!!
!!! DHCP Configuration on R3:
!!!!!!!
enable
configure terminal
ip dhcp excluded-address 192.168.4.1 192.168.4.10
ip dhcp pool NETWORK_192.168.4.0
network 192.168.4.0 255.255.255.0
default-router 192.168.4.1
domain-name cisco.com
dns-server 8.8.8.8
exit
end
write memory
```

6. Verify DHCP configuration

```
show ip dhcp pool
```

```
R3#  
*Mar 18 17:48:53.627: %SYS-5-CONFIG_I: Configured from console by console  
R3#show ip dhcp pool  
  
Pool NETWORK_192.168.4.0 :  
  Utilization mark (high/low)      : 100 / 0  
  Subnet size (first/next)        : 0 / 0  
  Total addresses                : 254  
  Leased addresses               : 0  
  Excluded addresses             : 10  
  Pending event                  : none  
  1 subnet is currently in the pool :  
    Current index          IP address range           Leased/Excluded/Total  
    192.168.4.1            192.168.4.1       - 192.168.4.254     0      / 10      / 254  
R3#
```

show ip dhcp binding

```
R3#show ip dhcp binding  
Bindings from all pools not associated with VRF:  
IP address      Client-ID/          Lease expiration      Type      State      Interface  
          Hardware address/  
          User name  
R3#show ip dhcp server statistics
```

show ip dhcp server statistics

```

R3#show ip dhcp server statistics
Memory usage      15808
Address pools     1
Database agents   0
Automatic bindings 0
Manual bindings   0
Expired bindings  0
Malformed messages 0
Secure arp entries 0
Renew messages    0
Workspace timeouts 0
Static routes     0
Relay bindings    0
Relay bindings active 0
Relay bindings terminated 0
Relay bindings selecting 0

Message          Received
BOOTREQUEST      0
DHCPDISCOVER     0
DHCPREQUEST      0
DHCPDECLINE      0
DHCPRELEASE      0
DHCPINFORM       0
DHCPVENDOR       0
BOOTREPLY        0
DHCPOFFER        0
DHCPACK          0
DHCPNAK          0

Message          Sent
BOOTREPLY        0
DHCPOFFER        0
DHCPACK          0
DHCPNAK          0

Message          Forwarded
BOOTREQUEST      0
DHCPDISCOVER     0
DHCPREQUEST      0
DHCPDECLINE      0
DHCPRELEASE      0
DHCPINFORM       0
DHCPVENDOR       0
BOOTREPLY        0
DHCPOFFER        0
DHCPACK          0
DHCPNAK          0

Message          Received
BOOTREPLY        0
DHCPOFFER        0
DHCPACK          0
DHCPNAK          0

Message          Forwarded
BOOTREQUEST      0
DHCPDISCOVER     0
DHCPREQUEST      0
DHCPDECLINE      0
DHCPRELEASE      0
DHCPINFORM       0
DHCPVENDOR       0
BOOTREPLY        0
DHCPOFFER        0
DHCPACK          0
DHCPNAK          0

DHCP-DPM Statistics
Offer notifications sent      0
Offer callbacks received      0
Classname requests sent      0
Classname callbacks received  0

R3#

```

7. Configure PC2 to be addresses automatically by DHCP, then save address to NVRAM.

```

PC2> ip dhcp
PC2> save
PC2> show

```

```

PC2> show

NAME  IP/MASK           GATEWAY          MAC            LPORT  RHOST:PORT
PC2   0.0.0.0/0          0.0.0.0          00:50:79:66:68:01  20050  127.0.0.1:20051
      fe80::250:79ff:fe66:6801/64

PC2> ip dhcp
DDORA IP 192.168.4.11/24 GW 192.168.4.1

PC2> show

NAME  IP/MASK           GATEWAY          MAC            LPORT  RHOST:PORT
PC2   192.168.4.11/24    192.168.4.1     00:50:79:66:68:01  20050  127.0.0.1:20051
      fe80::250:79ff:fe66:6801/64

PC2>

```

See DHCP printouts

```
R3#show ip dhcp pool

Pool NETWORK_192.168.4.0 :
  Utilization mark (high/low)      : 100 / 0
  Subnet size (first/next)        : 0 / 0
  Total addresses                : 254
  Leased addresses               : 1
  Excluded addresses             : 10
  Pending event                  : none
  1 subnet is currently in the pool :
    Current index          IP address range           Leased/Excluded/Total
    192.168.4.12          192.168.4.1      - 192.168.4.254    1      / 10      / 254
R3#
```

```
R3#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address      Client-ID/          Lease expiration       Type      State      Interface
              Hardware address/
              User name
192.168.4.11    0100.5079.6668.01    Mar 19 2025 08:33 PM  Automatic  Active   FastEthernet1/0
R3#
```

R3#	
R3#show ip dhcp server statistics	
Memory usage	16067
Address pools	1
Database agents	0
Automatic bindings	1
Manual bindings	0
Expired bindings	0
Malformed messages	0
Secure arp entries	0
Renew messages	0
Workspace timeouts	0
Static routes	0
Relay bindings	0
Relay bindings active	0
Relay bindings terminated	0
Relay bindings selecting	0
Message Received	
BOOTREQUEST	0
DHCPDISCOVER	2
DHCPOREQUEST	1
DHCPODECLINE	0
DHCPORELEASE	0
DHCPIINFORM	0
DHCVPENDOR	0
BOOTREPLY	0
DHCPOFFER	0
DHCPACK	0
DHCPNAK	0
Message Sent	
BOOTREPLY	0
DHCPOFFER	1
DHCPACK	1
DHCPNAK	0
DHCPACK	1
DHCPNAK	0
Message Forwarded	
BOOTREQUEST	0
DHCPDISCOVER	0
DHCPOREQUEST	0
DHCPODECLINE	0
DHCPORELEASE	0
DHCPIINFORM	0
DHCVPENDOR	0
BOOTREPLY	0
DHCPOFFER	0
DHCPACK	0
DHCPNAK	0
DHCP-DPM Statistics	
Offer notifications sent	0
Offer callbacks received	0
Classname requests sent	0
Classname callbacks received	0
R3#	

OSPF configuration

Giving loop back address to R2

4. Configure loop back address

A loopback address is a special IP address designated for testing and diagnostics on a network device like a router or a computer. It helps verify the internal workings of the device's IP stack without needing to access the external network.

IP Address 192.168.100.100 is given to be the highest value IP in the network and to be selected as DR.

!! loopback address R2

```
enable
conf t
interface lo0
description loopback address R2
ip address 192.168.100.100 255.255.255.255
end
copy running-config startup-config
```

Test loopback address from R2

```
ping 192.168.100.100
```

```
R2#!! loopback address R2
R2#enable
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#interface lo0
R2(config-if)# description loopback address R2
R2(config-if)# ip address 192.168.100.100 255.255.255.255
R2(config-if)#end
R2#copy running-config startup-config
Destination filename [startup-config]?
*Mar 18 20:08:07.134: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
*Mar 18 20:08:07.590: %SYS-5-CONFIG_I: Configured from console by console

Building configuration...
[OK]
R2#
R2#
R2#ping 192.168.100.100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.100.100, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/5/8 ms
R2#
```

5. Check printouts

```
show ip interface brief
```

```
R2#show ip interface brief
Interface          IP-Address      OK? Method Status      Protocol
FastEthernet0/0    192.168.1.2    YES manual up       up
FastEthernet1/0    unassigned     YES unset  administratively down down
FastEthernet1/1    unassigned     YES unset  administratively down down
Serial2/0          192.168.2.2    YES manual up       up
Serial2/1          192.168.3.1    YES manual up       up
Serial2/2          unassigned     YES unset  administratively down down
Serial2/3          unassigned     YES unset  administratively down down
Loopback0          192.168.100.100 YES manual up       up
R2#
```

Configure OSPF

1. Configure OSPF 10 on all routers, then save configuration to NVRAM.

!!OSPF Router R1:

```
enable
config t
router ospf 10
network 192.168.1.0 0.0.0.255 area 0
network 192.168.2.0 0.0.0.255 area 0
default-information originate
end
copy run start
```

!!!OSPF Router R2:

```
enable
conf t
router ospf 10
network 192.168.1.0 0.0.0.255 area 0
network 192.168.2.0 0.0.0.255 area 0
network 192.168.3.0 0.0.0.255 area 0
network 192.168.100.100 0.0.0.0 area 0
exit
end
write memory
```

!!!! OSPF Router R3:

```
enable
conf t
router ospf 10
```

```
network 192.168.1.0 0.0.0.255 area 0
network 192.168.3.0 0.0.0.255 area 0
network 192.168.4.0 0.0.0.255 area 0
exit
end
write memory
```

2. Verify routing table on all routers with the show ip route command

!R1

```
show ip route
show ip route ospf
show ip ospf neighbor
show ip ospf database
show ip ospf interface
show ip ospf
show ip protocols
```

```
R1#show ip route
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
      + - replicated route, % - next hop override

Gateway of last resort is 10.164.0.1 to network 0.0.0.0

S*   0.0.0.0/0 [254/0] via 10.164.0.1
    10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
      S   10.162.240.52/32 [254/0] via 10.164.0.1, FastEthernet1/0
      C   10.164.0.0/16 is directly connected, FastEthernet1/0
      L   10.164.0.248/32 is directly connected, FastEthernet1/0
    192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
      C   192.168.1.0/24 is directly connected, FastEthernet0/0
      L   192.168.1.1/32 is directly connected, FastEthernet0/0
    192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
      C   192.168.2.0/24 is directly connected, Serial2/0
      L   192.168.2.1/32 is directly connected, Serial2/0
O     192.168.3.0/24 [110/65] via 192.168.1.3, 06:42:39, FastEthernet0/0
                  [110/65] via 192.168.1.2, 06:46:14, FastEthernet0/0
O     192.168.4.0/24 [110/2] via 192.168.1.3, 06:42:39, FastEthernet0/0
    192.168.100.0/32 is subnetted, 1 subnets
O       192.168.100.100 [110/2] via 192.168.1.2, 04:13:41, FastEthernet0/0
R1#
```

```
R1#show ip route ospf
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, 1 - LISP
      + - replicated route, % - next hop override

Gateway of last resort is 10.164.0.1 to network 0.0.0.0

O    192.168.3.0/24 [110/65] via 192.168.1.3, 06:43:21, FastEthernet0/0
                  [110/65] via 192.168.1.2, 06:46:56, FastEthernet0/0
O    192.168.4.0/24 [110/2] via 192.168.1.3, 06:43:21, FastEthernet0/0
      192.168.100.0/32 is subnetted, 1 subnets
O        192.168.100.100 [110/2] via 192.168.1.2, 04:14:23, FastEthernet0/0
```

```
R1#show ip ospf neighbor

Neighbor ID      Pri  State            Dead Time     Address          Interface
192.168.3.1      0    FULL/ -          00:00:38      192.168.2.2      Serial2/0
192.168.3.1      1    FULL/BDR         00:00:30      192.168.1.2      FastEthernet0/0
192.168.4.1      1    FULL/DROTHER     00:00:32      192.168.1.3      FastEthernet0/0
R1#
```

```
R1#show ip ospf database

OSPF Router with ID (192.168.2.1) (Process ID 10)

      Router Link States (Area 0)

Link ID        ADV Router      Age       Seq#      Checksum Link count
192.168.2.1    192.168.2.1    1327      0x8000000F 0x008FFF 3
192.168.3.1    192.168.3.1    1632      0x8000000F 0x004024 6
192.168.4.1    192.168.4.1    624       0x8000000F 0x00D534 4

      Net Link States (Area 0)

Link ID        ADV Router      Age       Seq#      Checksum
192.168.1.1    192.168.2.1    55        0x8000000E 0x00DE2A

      Type-5 AS External Link States

Link ID        ADV Router      Age       Seq#      Checksum Tag
0.0.0.0        192.168.2.1    571       0x80000008 0x00AF86 10
R1#
```

```
R1#show ip ospf interface
Serial2/0 is up, line protocol is up
  Internet Address 192.168.2.1/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.2.1, Network Type POINT_TO_POINT, Cost: 64
  Topology-MTID    Cost    Disabled    Shutdown    Topology Name
    0            64        no          no          Base
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
      Hello due in 00:00:04
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 192.168.3.1
  Suppress hello for 0 neighbor(s)

FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.1.1/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.2.1, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown    Topology Name
    0            1        no          no          Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.2.1, Interface address 192.168.1.1
  Backup Designated router (ID) 192.168.3.1, Interface address 192.168.1.2
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
      Hello due in 00:00:05
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 192.168.3.1 (Backup Designated Router)
    Adjacent with neighbor 192.168.4.1
  Suppress hello for 0 neighbor(s)

R1#
```

```
R1# show ip ospf
Routing Process "ospf 10" with ID 192.168.2.1
Start time: 03:33:14.136, Time elapsed: 06:55:55.824
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
Supports area transit capability
Supports NSSA (compatible with RFC 3101)
Event-log enabled, Maximum number of events: 1000, Mode: cyclic
It is an autonomous system boundary router
Redistributing External Routes from,
Router is not originating router-LSAs with maximum metric
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPFs 10000 msec
Maximum wait time between two consecutive SPFs 10000 msec
Incremental-SPF disabled
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msec
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msec
Retransmission pacing timer 66 msec
Number of external LSA 1. Checksum Sum 0x00AF86
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Number of areas transit capable is 0
External flood list length 0
IETF NSF helper support enabled
Cisco NSF helper support enabled
Reference bandwidth unit is 100 mbps
Area BACKBONE(0)
    Number of interfaces in this area is 2
    Area has no authentication
    SPF algorithm last executed 04:17:35.004 ago
    SPF algorithm executed 7 times
    Area ranges are
    Number of LSA 4. Checksum Sum 0x028381
    Number of opaque link LSA 0. Checksum Sum 0x000000
    Number of DCbitless LSA 0
    Number of indication LSA 0
    Number of DoNotAge LSA 0
    Flood list length 0
```

```
R1#show ip protocols
*** IP Routing is NSF aware ***

Routing Protocol is "ospf 10"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.2.1
  It is an autonomous system boundary router
  Redistributing External Routes from,
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    192.168.1.0 0.0.0.255 area 0
    192.168.2.0 0.0.0.255 area 0
  Routing Information Sources:
    Gateway          Distance      Last Update
    192.168.3.1      110          08:15:10
    192.168.4.1      110          10:44:08
  Distance: (default is 110)
```

```
R1#
```

!R2

```
show ip route
show ip route ospf
show ip ospf neighbor
show ip ospf database
show ip ospf interface
show ip ospf
show ip protocols
```

```

R2#show ip route
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
      + - replicated route, % - next hop override

Gateway of last resort is 192.168.1.1 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 192.168.1.1, 04:16:16, FastEthernet0/0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, FastEthernet0/0
L        192.168.1.2/32 is directly connected, FastEthernet0/0
      192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.2.0/24 is directly connected, Serial2/0
L        192.168.2.2/32 is directly connected, Serial2/0
      192.168.3.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.3.0/24 is directly connected, Serial2/1
L        192.168.3.1/32 is directly connected, Serial2/1
O        192.168.4.0/24 [110/2] via 192.168.1.3, 06:56:44, FastEthernet0/0
      192.168.100.0/32 is subnetted, 1 subnets
C        192.168.100.100 is directly connected, Loopback0
R2#

```

```

R2#show ip route ospf
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
      + - replicated route, % - next hop override

Gateway of last resort is 192.168.1.1 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 192.168.1.1, 04:16:44, FastEthernet0/0
O        192.168.4.0/24 [110/2] via 192.168.1.3, 06:57:12, FastEthernet0/0
R2#

```

```

R2#show ip ospf neighbor

```

Neighbor ID	Pri	State	Dead Time	Address	Interface
192.168.4.1	0	FULL/-	00:00:35	192.168.3.2	Serial2/1
192.168.2.1	0	FULL/-	00:00:30	192.168.2.1	Serial2/0
192.168.2.1	1	FULL/DR	00:00:37	192.168.1.1	FastEthernet0/0
192.168.4.1	1	FULL/DROTHER	00:00:33	192.168.1.3	FastEthernet0/0

```
R2#show ip ospf database

          OSPF Router with ID (192.168.3.1) (Process ID 10)

          Router Link States (Area 0)

Link ID        ADV Router      Age       Seq#      Checksum Link count
192.168.2.1   192.168.2.1   439       0x80000010 0x008D01 3
192.168.3.1   192.168.3.1   712       0x80000010 0x003E25 6
192.168.4.1   192.168.4.1   1746      0x8000000F 0x00D534 4

          Net Link States (Area 0)

Link ID        ADV Router      Age       Seq#      Checksum
192.168.1.1   192.168.2.1   1178      0x8000000E 0x00DE2A

          Type-5 AS External Link States

Link ID        ADV Router      Age       Seq#      Checksum Tag
0.0.0.0        192.168.2.1   1694      0x80000008 0x00AF86 10
R2#
```

```
R2#show ip ospf interface

Loopback0 is up, line protocol is up
  Internet Address 192.168.100.100/32, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.3.1, Network Type LOOPBACK, Cost: 1
  Topology-MTID  Cost  Disabled  Shutdown  Topology Name
    0      1      no       no       Base
  Loopback interface is treated as a stub Host
Serial2/1 is up, line protocol is up
  Internet Address 192.168.3.1/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.3.1, Network Type POINT_TO_POINT, Cost: 64
  Topology-MTID  Cost  Disabled  Shutdown  Topology Name
    0      64     no       no       Base
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:02
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 3/3, Flood queue length 0
  Nxtxt 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 192.168.4.1
  Suppress hello for 0 neighbor(s)
```

```
Serial2/0 is up, line protocol is up
  Internet Address 192.168.2.2/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.3.1, Network Type POINT_TO_POINT, Cost: 64
  Topology-MTID      Cost      Disabled      Shutdown      Topology Name
    0            64        no          no           Base
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:08
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 192.168.2.1
    Suppress hello for 0 neighbor(s)
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.1.2/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.3.1, Network Type BROADCAST, Cost: 1
  Topology-MTID      Cost      Disabled      Shutdown      Topology Name
    0            1        no          no           Base
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 192.168.2.1, Interface address 192.168.1.1
  Backup Designated router (ID) 192.168.3.1, Interface address 192.168.1.2
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:09
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 192.168.2.1 (Designated Router)
    Adjacent with neighbor 192.168.4.1
    Suppress hello for 0 neighbor(s)
R2#
```

```
R2#show ip ospf
Routing Process "ospf 10" with ID 192.168.3.1
Start time: 03:34:41.448, Time elapsed: 07:15:14.412
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
Supports area transit capability
Supports NSSA (compatible with RFC 3101)
Event-log enabled, Maximum number of events: 1000, Mode: cyclic
Router is not originating router-LSAs with maximum metric
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPFs 10000 msec
Maximum wait time between two consecutive SPFs 10000 msec
Incremental-SPF disabled
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msec
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msec
Retransmission pacing timer 66 msec
Number of external LSA 1. Checksum Sum 0x00AD87
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Number of areas transit capable is 0
External flood list length 0
IETF NSF helper support enabled
Cisco NSF helper support enabled
Reference bandwidth unit is 100 mbps
    Area BACKBONE(0)
        Number of interfaces in this area is 4 (1 loopback)
        Area has no authentication
        SPF algorithm last executed 04:38:15.992 ago
        SPF algorithm executed 5 times
        Area ranges are
        Number of LSA 4. Checksum Sum 0x027C85
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

R2#

```
R2#show ip protocols
*** IP Routing is NSF aware ***

Routing Protocol is "ospf 10"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.3.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    192.168.1.0 0.0.0.255 area 0
    192.168.2.0 0.0.0.255 area 0
    192.168.3.0 0.0.0.255 area 0
    192.168.100.100 0.0.0.0 area 0
  Routing Information Sources:
    Gateway          Distance      Last Update
    192.168.2.1      110          08:04:13
    192.168.4.1      110          10:44:40
  Distance: (default is 110)
```

```
R2#
```

! R3
show ip route
show ip route ospf
show ip ospf neighbor
show ip ospf database
show ip ospf interface
show ip ospf
show ip protocols

```
R3#show ip route
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, I - LISP
      + - replicated route, % - next hop override

Gateway of last resort is 192.168.1.1 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 192.168.1.1, 04:34:16, FastEthernet0/0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, FastEthernet0/0
L        192.168.1.3/32 is directly connected, FastEthernet0/0
O        192.168.2.0/24 [110/65] via 192.168.1.2, 07:14:42, FastEthernet0/0
              [110/65] via 192.168.1.1, 07:14:42, FastEthernet0/0
      192.168.3.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.3.0/24 is directly connected, Serial2/1
L        192.168.3.2/32 is directly connected, Serial2/1
      192.168.4.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.4.0/24 is directly connected, FastEthernet1/0
L        192.168.4.1/32 is directly connected, FastEthernet1/0
      192.168.100.0/32 is subnetted, 1 subnets
O        192.168.100.100 [110/2] via 192.168.1.2, 04:45:45, FastEthernet0/0
R3#
```

```
R3#show ip route ospf
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, I - LISP
      + - replicated route, % - next hop override

Gateway of last resort is 192.168.1.1 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 192.168.1.1, 04:34:47, FastEthernet0/0
O        192.168.2.0/24 [110/65] via 192.168.1.2, 07:15:13, FastEthernet0/0
              [110/65] via 192.168.1.1, 07:15:13, FastEthernet0/0
      192.168.100.0/32 is subnetted, 1 subnets
O        192.168.100.100 [110/2] via 192.168.1.2, 04:46:16, FastEthernet0/0
R3#
```

```
R3#show ip ospf neighbor

Neighbor ID      Pri  State            Dead Time    Address          Interface
192.168.3.1      0    FULL/ -          00:00:37     192.168.3.1    Serial2/1
192.168.2.1      1    FULL/DR         00:00:31     192.168.1.1    FastEthernet0/0
192.168.3.1      1    FULL/BDR        00:00:33     192.168.1.2    FastEthernet0/0
R3#
R3#show ip ospf database

OSPF Router with ID (192.168.4.1) (Process ID 10)

        Router Link States (Area 0)

Link ID          ADV Router      Age       Seq#      Checksum Link count
192.168.2.1      192.168.2.1   1197      0x80000010 0x008D01 3
192.168.3.1      192.168.3.1   1472      0x80000010 0x003E25 6
192.168.4.1      192.168.4.1   486       0x80000010 0x00D335 4

        Net Link States (Area 0)

Link ID          ADV Router      Age       Seq#      Checksum
192.168.1.1      192.168.2.1   1937      0x8000000E 0x00DE2A

        Type-5 AS External Link States

Link ID          ADV Router      Age       Seq#      Checksum Tag
0.0.0.0          192.168.2.1   439       0x80000009 0x00AD87 10
R3#
```

```
R3#show ip ospf interface
FastEthernet1/0 is up, line protocol is up
  Internet Address 192.168.4.1/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.4.1, Network Type BROADCAST, Cost: 1
  Topology-MTID      Cost      Disabled      Shutdown      Topology Name
    0            1        no          no          Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.4.1, Interface address 192.168.4.1
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:08
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 3/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
Serial2/1 is up, line protocol is up
  Internet Address 192.168.3.2/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.4.1, Network Type POINT TO POINT, Cost: 64
  Topology-MTID      Cost      Disabled      Shutdown      Topology Name
    0            64        no          no          Base
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:07
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 192.168.3.1
  Suppress hello for 0 neighbor(s)
```

```
Suppresses hello for 0 neighbor(s)
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.1.3/24, Area 0, Attached via Network Statement
  Process ID 10, Router ID 192.168.4.1, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown    Topology Name
    0          1        no        no           Base
  Transmit Delay is 1 sec, State DROTHER, Priority 1
  Designated Router (ID) 192.168.2.1, Interface address 192.168.1.1
  Backup Designated router (ID) 192.168.3.1, Interface address 192.168.1.2
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:09
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 192.168.2.1  (Designated Router)
    Adjacent with neighbor 192.168.3.1  (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
R3#
```

```
R3#show ip ospf
Routing Process "ospf 10" with ID 192.168.4.1
Start time: 03:38:06.252, Time elapsed: 07:25:40.268
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
Supports area transit capability
Supports NSSA (compatible with RFC 3101)
Event-log enabled, Maximum number of events: 1000, Mode: cyclic
Router is not originating router-LSAs with maximum metric
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPFs 10000 msec
Maximum wait time between two consecutive SPFs 10000 msec
Incremental-SPF disabled
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msec
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msec
Retransmission pacing timer 66 msec
Number of external LSA 1. Checksum Sum 0x00AD87
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Number of areas transit capable is 0
External flood list length 0
IETF NSF helper support enabled
Cisco NSF helper support enabled
Reference bandwidth unit is 100 mbps
Area BACKBONE(0)
    Number of interfaces in this area is 3
    Area has no authentication
    SPF algorithm last executed 04:52:16.216 ago
    SPF algorithm executed 4 times
    Area ranges are
    Number of LSA 4. Checksum Sum 0x027A86
    Number of opaque link LSA 0. Checksum Sum 0x000000
    Number of DCbitless LSA 0
    Number of indication LSA 0
    Number of DoNotAge LSA 0
    Flood list length 0
```

R3#

show ip protocols

```
R3#show ip protocols
*** IP Routing is NSF aware ***

Routing Protocol is "ospf 10"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.4.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    192.168.1.0 0.0.0.255 area 0
    192.168.3.0 0.0.0.255 area 0
    192.168.4.0 0.0.0.255 area 0
  Routing Information Sources:
    Gateway          Distance      Last Update
    192.168.3.1      110          08:16:00
    192.168.2.1      110          08:04:31
  Distance: (default is 110)
```

```
R3#
```

NAT

Configure NAT in R3

!!! R1

```
enable
configure terminal
access-list 1 permit 192.168.1.0 0.0.0.255
access-list 1 permit 192.168.2.0 0.0.0.255
access-list 1 permit 192.168.3.0 0.0.0.255
access-list 1 permit 192.168.4.0 0.0.0.255

ip nat inside source list 1 interface FastEthernet1/0 overload

interface FastEthernet0/0
  ip nat inside
  exit
interface FastEthernet1/0
  ip nat outside
  exit
```

```
end  
write memory
```

Test connectivity

PC1 to PC2

- ping 192.168.4.11

```
PC1> ping 192.168.4.11  
  
Redirect Network, gateway 192.168.1.1 -> 192.168.1.3  
84 bytes from 192.168.4.11 icmp_seq=1 ttl=63 time=23.748 ms  
84 bytes from 192.168.4.11 icmp_seq=2 ttl=63 time=33.007 ms  
84 bytes from 192.168.4.11 icmp_seq=3 ttl=63 time=36.422 ms  
84 bytes from 192.168.4.11 icmp_seq=4 ttl=63 time=25.440 ms  
84 bytes from 192.168.4.11 icmp_seq=5 ttl=63 time=16.791 ms  
  
PC1> █
```

PC2 to PC1

ping 192.168.1.11

```
PC2> ping 192.168.1.11  
  
84 bytes from 192.168.1.11 icmp_seq=1 ttl=62 time=33.856 ms  
84 bytes from 192.168.1.11 icmp_seq=2 ttl=62 time=48.530 ms  
84 bytes from 192.168.1.11 icmp_seq=3 ttl=62 time=42.323 ms  
84 bytes from 192.168.1.11 icmp_seq=4 ttl=62 time=45.755 ms  
84 bytes from 192.168.1.11 icmp_seq=5 ttl=62 time=23.757 ms  
  
PC2> █
```

From R1 to R2:

ping 192.168.2.2

```
R1#ping 192.168.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/37/64 ms
R1#
```

From R2 to R3:

ping 192.168.3.2

```
R2#ping 192.168.3.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/24/36 ms
R2#
```

From PC1 to its default gateway (R1):

ping 192.168.1.1

```
PC1> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=255 time=13.326 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=255 time=14.402 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=255 time=5.166 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=255 time=15.785 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=255 time=12.552 ms

PC1>
```

From PC2 to its default gateway (R3):

ping 192.168.4.1

```
PC2> ping 192.168.4.1

84 bytes from 192.168.4.1 icmp_seq=1 ttl=255 time=32.342 ms
84 bytes from 192.168.4.1 icmp_seq=2 ttl=255 time=4.712 ms
84 bytes from 192.168.4.1 icmp_seq=3 ttl=255 time=7.942 ms
84 bytes from 192.168.4.1 icmp_seq=4 ttl=255 time=8.752 ms
84 bytes from 192.168.4.1 icmp_seq=5 ttl=255 time=11.609 ms

PC2> █
```

Check NAT translations

```
PC1> ping 8.8.8.8

84 bytes from 8.8.8.8 icmp_seq=1 ttl=117 time=40.550 ms
84 bytes from 8.8.8.8 icmp_seq=2 ttl=117 time=37.292 ms
84 bytes from 8.8.8.8 icmp_seq=3 ttl=117 time=35.470 ms
84 bytes from 8.8.8.8 icmp_seq=4 ttl=117 time=15.344 ms
84 bytes from 8.8.8.8 icmp_seq=5 ttl=117 time=13.727 ms

PC1> █
```

Protocol	Inside IP	Inside Port	Outside IP	Outside Port
icmp	10.164.0.248:1024	192.168.1.11:43071	8.8.8.8:43071	8.8.8.8:1024
icmp	10.164.0.248:1025	192.168.1.11:43327	8.8.8.8:43327	8.8.8.8:1025
icmp	10.164.0.248:1026	192.168.1.11:43583	8.8.8.8:43583	8.8.8.8:1026
icmp	10.164.0.248:1027	192.168.1.11:43839	8.8.8.8:43839	8.8.8.8:1027
icmp	10.164.0.248:1028	192.168.1.11:44095	8.8.8.8:44095	8.8.8.8:1028

```
R1#
```

Show access-lists

```
R1#show access-lists
Standard IP access list 1
  10 permit 192.168.1.0, wildcard bits 0.0.0.255 (5 matches)
  20 permit 192.168.2.0, wildcard bits 0.0.0.255
  30 permit 192.168.3.0, wildcard bits 0.0.0.255
  40 permit 192.168.4.0, wildcard bits 0.0.0.255 (6 matches)
R1#
```

```
ping 8.8.8.8 source FastEthernet0/0
```

```
ping 8.8.8.8 source f1/0
```

```
R1#ping 8.8.8.8 source FastEthernet0/0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
Packet sent with a source address of 192.168.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/12/24 ms
R1#ping 8.8.8.8 source f1/0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
Packet sent with a source address of 10.164.0.248
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/12 ms
R1#
```

Scripts

```
!!!!!! R1 !!!!!!
enable
configure terminal
hostname R1
no ip domain lookup
banner motd #WARNING Authorized Users Only! #

interface Serial2/0
description Connection to R2
ip address 192.168.2.1 255.255.255.0
no shutdown
exit

interface FastEthernet0/0
description Connection to SW1
ip address 192.168.1.1 255.255.255.0
ip nat inside
no shutdown
exit

interface FastEthernet1/0
description Connection to Cloud
ip address dhcp
ip nat outside
no shutdown
exit

router ospf 10
network 192.168.1.0 0.0.0.255 area 0
network 192.168.2.0 0.0.0.255 area 0
default-information originate
exit

access-list 1 permit 192.168.1.0 0.0.0.255
access-list 1 permit 192.168.2.0 0.0.0.255
access-list 1 permit 192.168.3.0 0.0.0.255
access-list 1 permit 192.168.4.0 0.0.0.255
ip nat inside source list 1 interface FastEthernet1/0 overload

end
write memory
```

```
!!!!!! R2 !!!!!!
enable
configure terminal
hostname R2
no ip domain lookup
banner motd #WARNING Authorized Users Only! #

interface Serial2/0
description Connection to R1
ip address 192.168.2.2 255.255.255.0
no shutdown
exit

interface Serial2/1
description Connection to R3
ip address 192.168.3.1 255.255.255.0
no shutdown
exit

interface FastEthernet0/0
description Connection to SW1
ip address 192.168.1.2 255.255.255.0
no shutdown
exit

interface Loopback0
description Loopback address R2
ip address 192.168.100.100 255.255.255.255
exit

router ospf 10
network 192.168.1.0 0.0.0.255 area 0
network 192.168.2.0 0.0.0.255 area 0
network 192.168.3.0 0.0.0.255 area 0
network 192.168.100.100 0.0.0.0 area 0
exit

end
write memory
```

```
!!!!!! R3 !!!!!!
enable
configure terminal
hostname R3
no ip domain lookup
banner motd #WARNING Authorized Users Only! #

interface Serial2/1
description Connection to R2
ip address 192.168.3.2 255.255.255.0
no shutdown
exit

interface FastEthernet0/0
description Connection to SW1
ip address 192.168.1.3 255.255.255.0
no shutdown
exit

interface FastEthernet1/0
description Connection to SW2
ip address 192.168.4.1 255.255.255.0
no shutdown
exit

ip dhcp excluded-address 192.168.4.1 192.168.4.10
ip dhcp pool NETWORK_192.168.4.0
network 192.168.4.0 255.255.255.0
default-router 192.168.4.1
domain-name cisco.com
dns-server 8.8.8.8
exit

router ospf 10
network 192.168.1.0 0.0.0.255 area 0
network 192.168.3.0 0.0.0.255 area 0
network 192.168.4.0 0.0.0.255 area 0
exit

end
write memory
```

```
!!!!!!!
!!! SW1 !!!
!!!!!!!

enable
configure terminal
hostname SW1
no ip domain lookup
banner motd #WARNING Authorized Users Only! #

interface FastEthernet1/0
description Connection to R1
no shutdown
exit

interface FastEthernet1/1
description Connection to R2
no shutdown
exit

interface FastEthernet1/2
description Connection to R3
no shutdown
exit

interface FastEthernet1/5
description Connection to PC1
no shutdown
exit

end
write memory
```

```
!!!!!!!
!!! SW2 !!!
!!!!!!!

enable
```

```
configure terminal
hostname SW2
no ip domain lookup
banner motd #WARNING Authorized Users Only! #

interface FastEthernet1/0
description Connection to R3
no shutdown
exit

interface FastEthernet1/5
description Connection to PC2
no shutdown
exit

end
write memory
```

PC1

```
ip 192.168.1.11 255.255.255.0 192.168.1.1
save
```

PC2

```
ip dhcp
save
```

Commands

OSPF commands

```
show ip route
show ip route ospf
show ip ospf neighbor
show ip ospf database
show ip ospf interface
show ip ospf
show ip protocols
```

DHCP commands

```
show ip dhcp pool
```

```
show ip dhcp binding
show ip dhcp server statistics
show running-config | section dhcp
show ip interface brief
debug ip dhcp server events
debug ip dhcp server packet
clear ip dhcp binding *
```

Work around if ping does not work

In R3

```
release dhcp f1/0
renew dhcp f1/0
show dhcp lease
```

NAT commands

!R1

```
show ip nat translations
show ip nat statistics
show running-config | include nat
show ip interface brief
show access-lists
debug ip nat
no debug all.
clear ip nat translation *
ping 8.8.8.8 source
ping 8.8.8.8 source FastEthernet0/0
ping 8.8.8.8 source f1/0
```

3.1 Test Automation with Ansible

3.1.1 Setup

3.1.1.1 Install distros for ansible automation

Refer to course Operation systems I for Linux distributions installation in VMWare

```
└─▶ ubuntu1 - 10.164.101.101
  └─▶ ubuntu1a - 10.164.101.102
    └─▶ fedora1 - 101.164.101.103
      └─▶ centos1 - 10.164.101.104
      └─▶ opensuse1 - 101.164.101.105
```

1. ubuntu1.ansible1.com - 10.164.101.101 mask 255.255.0.0 default gateway 10.164.0.1
2. ubuntu1a.ansible1.com 10.164.101.102 mask 255.255.0.0 default gateway 10.164.0.1
3. fedora1.ansible1.com 10.164.101.103 mask 255.255.0.0 default gateway 10.164.0.1
4. centos1.ansible1.com 10.164.101.104 mask 255.255.0.0 default gateway 10.164.0.1
5. opensuse1.ansible.com 10.164.101.105 mask 255.255.0.0 default gateway 10.164.0.1

UBUNTU1

Install ansible on ubuntu1 machine

1. Install the software-properties-common package by entering:

```
sudo apt-get install software-properties-common
```

```
root@ubuntu1:/home/student/Desktop# sudo apt-get install software-properties-common

Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
software-properties-common is already the newest version (0.99.49.1).
software-properties-common set to manually installed.
The following packages were automatically installed and are no longer required:
  libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
root@ubuntu1:/home/student/Desktop#
```

2. Add a repository to find the applications we're going to need for Ansible by entering:

```
sudo apt-add-repository ppa:ansible/ansible
```

```
root@ubuntu1:/home/student/Desktop# sudo apt-add-repository ppa:ansible/ansible

Repository: 'types: deb
URIs: https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/
Suites: noble
Components: main
'
Description:
Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts or custom code to deploy and update your applications— automate in a language that approaches plain English, using SSH, with no agents to install on remote systems.

http://ansible.com/

If you face any issues while installing Ansible PPA, file an issue here:
https://github.com/ansible-community/ppa/issues
More info: https://launchpad.net/~ansible/+archive/ubuntu/ansible
Adding repository...
Hit:1 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:2 http://ca.archive.ubuntu.com/ubuntu noble InRelease
Hit:3 http://ca.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:4 http://ca.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:5 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/noble InRelease [17.8 kB]
Get:6 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/noble/main amd64 Packages [776 B]
Get:7 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/noble/main Translation-en [472 B]
Fetched 10.1 kB in 0s (21.5 kB/s)
Reading package lists... Done
root@ubuntu1:/home/student/Desktop#
```

3. Check for updates and upgrades by entering:

```
sudo apt-get update
```

```
Reading package lists... Done
root@ubuntu1:/home/student/Desktop# sudo apt-get update

Hit:1 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:2 http://ca.archive.ubuntu.com/ubuntu noble InRelease
Hit:3 http://ca.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:4 http://ca.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:5 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/noble InRelease
Reading package lists... Done
root@ubuntu1:/home/student/Desktop#
```

```
sudo apt-get upgrade
```

```
root@ubuntu1:/home/student/Desktop# sudo apt-get update
Hit:1 http://ca.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://ca.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://ca.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:5 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/noble InRelease
Reading package lists... 78%
Reading package lists... Done
root@ubuntu1:/home/student/Desktop#
root@ubuntu1:/home/student/Desktop#
root@ubuntu1:/home/student/Desktop#
root@ubuntu1:/home/student/Desktop# sudo apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@ubuntu1:/home/student/Desktop#
```

4. Install python by entering:

```
sudo apt-get install python3 -y
```

```
root@ubuntu1:/home/student/Desktop# sudo apt-get install python
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package python is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source
However the following packages replace it:
  2to3 python-is-python3

E: Package 'python' has no installation candidate
```

```
E: Package 'python' has no installation candidate
root@ubuntu1:/home/student/Desktop# sudo apt-get install python-is-python3

Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  liblvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  python-is-python3
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 2,684 B of archives.
After this operation, 15.4 kB of additional disk space will be used.
Get:1 http://ca.archive.ubuntu.com/ubuntu noble/main amd64 python-is-python3 all 3.11.4-1 [2,684 B]
Fetched 2,684 B in 0s (25.8 kB/s)
Selecting previously unselected package python-is-python3.
(Reading database ... 150039 files and directories currently installed.)
Preparing to unpack .../python-is-python3_3.11.4-1_all.deb ...
Unpacking python-is-python3 (3.11.4-1) ...
Setting up python-is-python3 (3.11.4-1) ...
Processing triggers for man-db (2.12.0-4build2) ...
root@ubuntu1:/home/student/Desktop# sudo apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  liblvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@ubuntu1:/home/student/Desktop# sudo apt-get update
Hit:1 http://ca.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:3 http://ca.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:4 http://ca.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:5 https://ppa.launchpadcontent.net/ansible/ubuntu noble InRelease
Reading package lists... Done
root@ubuntu1:/home/student/Desktop#
```

5. Install Ansible by entering:

```
sudo apt-get install ansible -y
```

6. Test Ansible by entering:

```
ansible localhost -m ping
```

```
root@ubuntu1:/home/student/Desktop# ##### TEST ANSIBLE
root@ubuntu1:/home/student/Desktop# ansible localhost -m ping
localhost | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
root@ubuntu1:/home/student/Desktop#
```

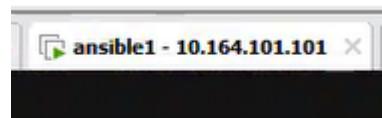
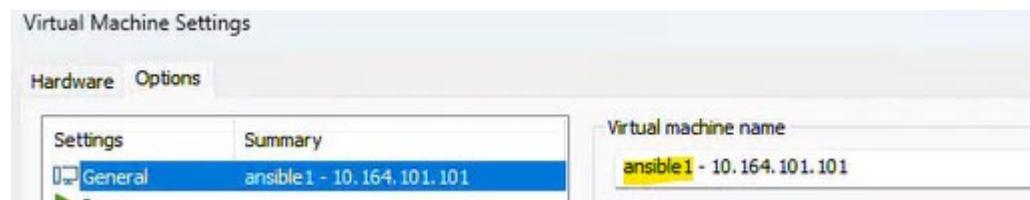
Repeat installation of python, ansible and test ansible for all other distros. For each distro make sure to update to latest software using the appropriate command.

3.1.1.2 Configure DNS zone Using Webmin

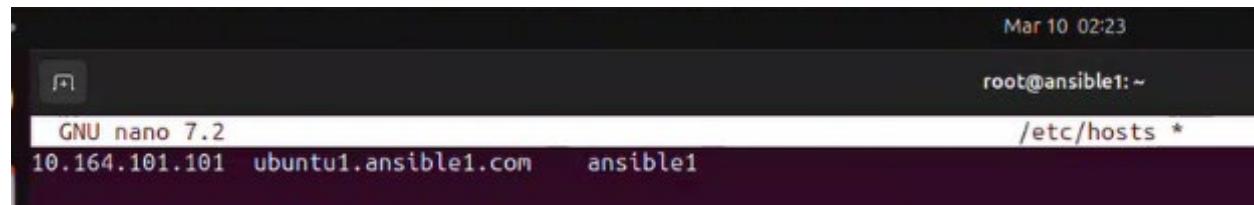
Video reference - <https://tutorialsonline.ca/courses/1330884/lectures/31411362>

Rename ansible server

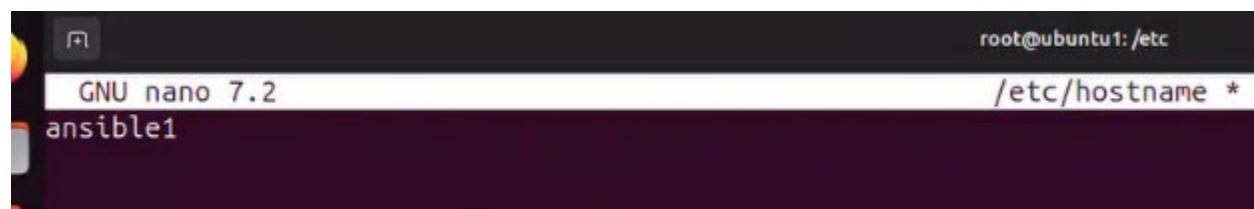
1. Rename VM



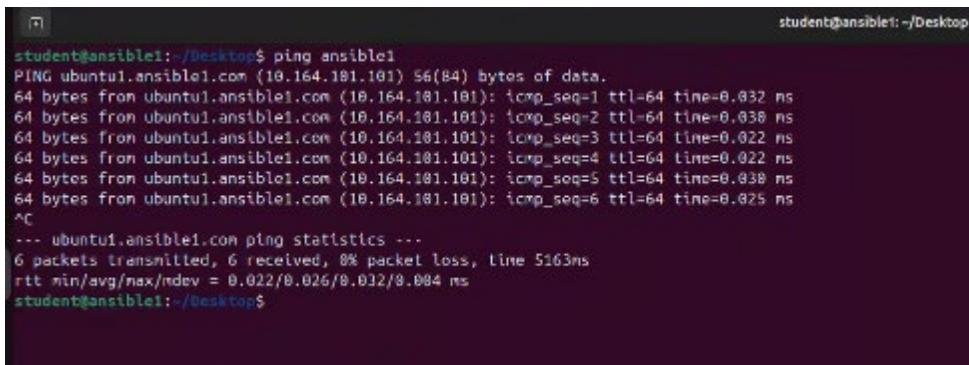
2. Change /etc/hosts



3. Change /etc/hostname



4. Reboot
5. After reboot ping ansible1



```
student@ansible1: ~/Desktop$ ping ansible1
PING ubuntu1.ansible1.com (10.164.181.101) 56(84) bytes of data.
64 bytes from ubuntu1.ansible1.com (10.164.181.101): icmp_seq=1 ttl=64 time=0.032 ns
64 bytes from ubuntu1.ansible1.com (10.164.181.101): icmp_seq=2 ttl=64 time=0.038 ns
64 bytes from ubuntu1.ansible1.com (10.164.181.101): icmp_seq=3 ttl=64 time=0.022 ns
64 bytes from ubuntu1.ansible1.com (10.164.181.101): icmp_seq=4 ttl=64 time=0.022 ns
64 bytes from ubuntu1.ansible1.com (10.164.181.101): icmp_seq=5 ttl=64 time=0.038 ns
64 bytes from ubuntu1.ansible1.com (10.164.181.101): icmp_seq=6 ttl=64 time=0.025 ns
^C
--- ubuntu1.ansible1.com ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5163ms
rtt min/avg/max/mdev = 0.022/0.026/0.032/0.004 ms
student@ansible1: ~/Desktop$
```

BIND DNS in ansible1

Open Webmin BIND DNS Server



The screenshot shows the Webmin interface for managing a BIND DNS server. The left sidebar contains a navigation tree with categories like Webmin, System, Servers, Tools, Networking, Hardware, Cluster, and Refresh Modules. The 'Servers' category is expanded, showing 'BIND DNS Server' as the selected item. The main content area is titled 'BIND DNS Server' and 'BIND version 9.18.30'. It features two rows of icons representing various server management functions.

Global Server Options							
Other DNS Servers	Logging and Errors	Access Control Lists	Files and Directories	Forwarding and Transfers	Addresses and Topology	Miscellaneous Options	Control Interface Options

Advanced Options							
DNS Keys	Zone Defaults	Cluster Slave Servers	Setup RNDC	DNSSEC Verification	DNSSEC Key Resigning	Check BIND Config	Edit Config File

Edit config file

See in Ubuntu is divided into multiple files:

```
include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";
```

Editing config file: /etc/bind/named.conf

```
1 // This is the primary configuration file for the BIND DNS server named.
2 //
3 // Please read /usr/share/doc/bind9/README.Debian for information on the
4 // structure of BIND configuration files in Debian, *BEFORE* you customize
5 // this configuration file.
6 //
7 // If you are just adding zones, please do that in /etc/bind/named.conf.local
8
9 include "/etc/bind/named.conf.options";
10 include "/etc/bind/named.conf.local";
11 include "/etc/bind/named.conf.default-zones";
12
```

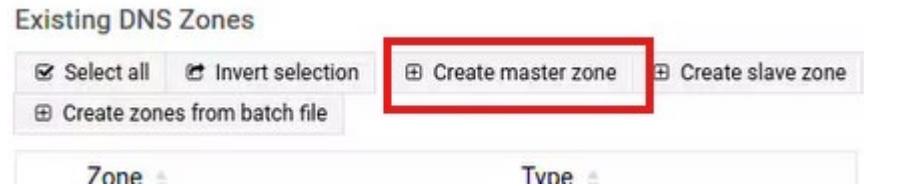
Editing config file: /etc/bind/named.conf

```
1 // This i
2 //
3 // Please
4 // struct
5 // this c
6 //
7 // If you are just adding zones, please do that in
8
9 include "/etc/bind/named.conf.options";
10 include "/etc/bind/named.conf.local";
11 include "/etc/bind/named.conf.default-zones";
12
```

The screenshot shows a dropdown menu in the file input field containing the following options:

- /etc/bind/named.conf
- /etc/bind/named.conf.options
- /etc/bind/named.conf.local
- /etc/bind/named.conf.default-zones

2. Create masterzone



```
student@ansible1:~/Desktop$ su -
Password:
root@ansible1:~# cat /etc/hosts
10.164.101.101 ansible1.ansible1.com ansible1
```

Verify settings of hostname:

```
student@ansible1:~/Desktop$ cat /etc/hosts
10.164.101.101 ubuntu1.ansible1.com ansible1
student@ansible1:~/Desktop$ cat /etc/hostname
ansible1
```

1. Set “Domain name / Network” = ansible1.com
2. Set “Master server “ = ubuntu1.ansible1.com
3. Set “Email address” = root.ansible1.com
4. Create
5. Start stop

Create A record

Name ansible1

Address 10.164.101.101

Address Records
In ansible1.com

Add Address Record

Name: ansible1

Time-To-Live: Default

Address: 10.164.101.101

Update reverse: Yes (and replace existing)

Create

Show records matching: Search

Create A record for centos

ansible1 - 10.164.101.101

centos1 - 10.164.101.104

Mar 10 02:44

BIND DNS Server/Address

Address Records
In ansible1.com

Add Address Record

Name: centos1

Address: 10.164.101.104

Update reverse: Yes (and replace existing)

Create

Show records matching: Search

Name centos1

Address 10.164.101.104

Address Records
In ansible1.com

Add Address Record

Name centos1	Address 10.164.101.104
Time-To-Live • Default seconds	Update reverse • Yes <input checked="" type="radio"/> Yes (and replace existing) <input type="radio"/> No
<input checked="" type="radio"/> Create	

Create A record for ubuntu1a

Name ubuntu1a

Address 10.164.101.102

Address Records
In ansible1.com

Add Address Record

Name ubuntu1a	Address 10.164.101.102
Time-To-Live • Default seconds	Update reverse • Yes <input checked="" type="radio"/> Yes (and replace existing) <input type="radio"/> No
<input checked="" type="radio"/> Create	

Create A record for fedora1

Name fedora1

Address 10.164.101.103

Address Records
In ansible1.com

Add Address Record

Name <input type="text" value="fedora1"/>	Address <input type="text" value="10.164.101.103"/>
Time-To-Live <input checked="" type="radio"/> Default <input type="radio"/> <input type="text"/> seconds ▾	Update reverse <input checked="" type="radio"/> Yes <input type="radio"/> Yes (and replace existing) <input type="radio"/> No
Create	
Show records matching: <input type="text"/> Search	

Opensuse

Edit Address Record

Name <input type="text" value="opensuse1.ansible1.com."/>	Address <input type="text" value="10.164.101.105"/>
Time-To-Live <input checked="" type="radio"/> Default (3600) <input type="radio"/> <input type="text"/> seconds ▾	Update reverse <input checked="" type="radio"/> Yes <input type="radio"/> No

We have five records for our servers

https://10.164.101.101:10000/bind9/edit_recs.cgi?zone=ansible1.com&view=any&type=A

Address Records

In ansible1.com

Add Address Record

Name:

Address:

Time-To-Live: Default
 seconds

Update reverse:
 Yes Yes (and replace existing) No

Create

Show records matching:

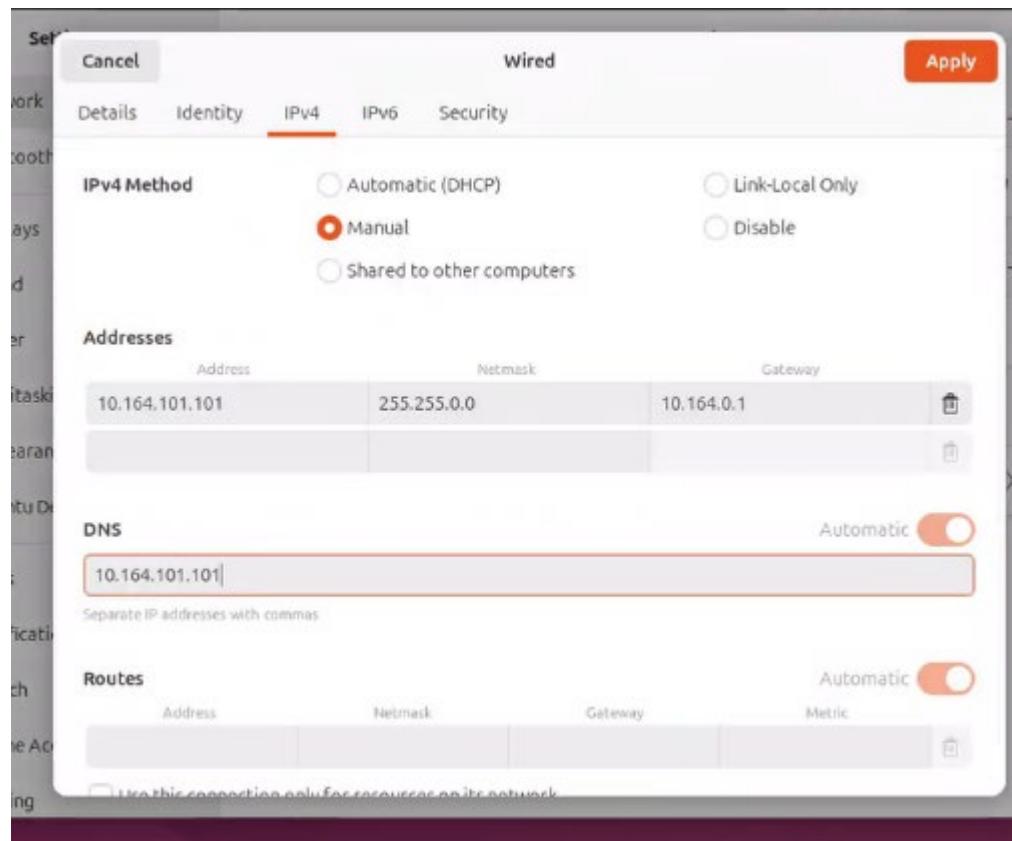
Select all Invert selection

Name	TTL	Address
ansible1 ansible1.com.	3600	10.164.101.101
centos1 ansible1.com.	3600	10.164.101.104
ubuntu1a ansible1.com.	3600	10.164.101.102
fedora1 ansible1.com.	3600	10.164.101.103
opensuse1 ansible1.com.	3600	10.164.101.105

Select all Invert selection

Delete Selected Delete reverses too?

Modify DNS server to self



Restart card

Check DNS server used

resolvectl status

```
root@ansible1:~# resolvectl status
Global
    Protocols: -LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
    resolv.conf mode: stub

Link 2 (ens33)
    Current Scopes: DNS
        Protocols: +DefaultRoute -LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
    Current DNS Server: 10.164.101.101
        DNS Servers: 10.164.101.101
root@ansible1:~#
```

Ping ansible1

```
root@ansible1:~# ping ansible1
PING ansible1.ansible1.com (10.164.101.101) 56(84) bytes of data.
64 bytes from ansible1.ansible1.com (10.164.101.101): icmp_seq=1 ttl=64 time=0.020 ms
64 bytes from ansible1.ansible1.com (10.164.101.101): icmp_seq=2 ttl=64 time=0.023 ms
64 bytes from ansible1.ansible1.com (10.164.101.101): icmp_seq=3 ttl=64 time=0.021 ms
^C
--- ansible1.ansible1.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2071ms
rtt min/avg/max/mdev = 0.020/0.021/0.023/0.001 ms
root@ansible1:~# cd /var/
```

Ping

```
root@ansible1:/etc/bind# ping ansible1.ansible1.com
PING ansible1.ansible1.com (10.164.101.101) 56(84) bytes of data.
64 bytes from ansible1.ansible1.com (10.164.101.101): icmp_seq=1 ttl=64 time=0.017 ms
64 bytes from ansible1.ansible1.com (10.164.101.101): icmp_seq=2 ttl=64 time=0.048 ms
64 bytes from ansible1.ansible1.com (10.164.101.101): icmp_seq=3 ttl=64 time=0.029 ms
64 bytes from ansible1.ansible1.com (10.164.101.101): icmp_seq=4 ttl=64 time=0.023 ms
64 bytes from ansible1.ansible1.com (10.164.101.101): icmp_seq=5 ttl=64 time=0.030 ms
64 bytes from ansible1.ansible1.com (10.164.101.101): icmp_seq=6 ttl=64 time=0.025 ms
64 bytes from ansible1.ansible1.com (10.164.101.101): icmp_seq=7 ttl=64 time=0.025 ms
^C
--- ansible1.ansible1.com ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6143ms
rtt min/avg/max/mdev = 0.017/0.028/0.048/0.009 ms
```

Ping google.com

```
root@ansible1:/etc/bind# ping google.com
PING google.com (142.250.69.142) 56(84) bytes of data.
64 bytes from 142.250.69.142: icmp_seq=1 ttl=117 time=1.98 ms
64 bytes from 142.250.69.142: icmp_seq=2 ttl=117 time=1.76 ms
64 bytes from 142.250.69.142: icmp_seq=3 ttl=117 time=1.91 ms
64 bytes from 142.250.69.142: icmp_seq=4 ttl=117 time=1.94 ms
64 bytes from 142.250.69.142: icmp_seq=5 ttl=117 time=1.82 ms
64 bytes from 142.250.69.142: icmp_seq=6 ttl=117 time=1.93 ms
^C
--- google.com ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5018ms
rtt min/avg/max/mdev = 1.757/1.890/1.983/0.078 ms
```

```
cat /var/lib/bind/ansible1.com.hosts
```

```
];
root@ansible1:/etc/bind# cat /etc/bind/named.conf.local
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "ansible1.com" {
    type master;
    file "/var/lib/bind/ansible1.com.hosts";
};

root@ansible1:/etc/bind# cat /var/lib/bind/ansible1.com.hosts
$ttl 3600
ansible1.com.    IN      SOA     ansible1.ansible1.com. root.ansible1.com (
                      2025031005
                      3600
                      600
                      1209600
                      3600 )
ansible1.com.    IN      NS       ansible1.ansible1.com.
ansible1.ansible1.com.  IN      A        10.164.101.101
centos1.ansible1.com. IN      A        10.164.101.104
ubuntu1a.ansible1.com. IN      A        10.164.101.102
fedora1.ansible1.com.  IN      A        10.164.101.103
root@ansible1:/etc/bind#
```

Change DNS for Fedora , ubuntu and centos

Ubuntu1a

Cancel **Wired** **Apply**

Details Identity **IPv4** IPv6 Security

IPv4 Method

Automatic (DHCP) Link-Local Only
 Manual Disable
 Shared to other computers

Addresses

Address	Netmask	Gateway	
10.164.101.102	255.255.0.0	10.164.0.1	

DNS Automatic
10.164.101.101
Separate IP addresses with commas

Routes Automatic

Address	Netmask	Gateway	Metric	

Fedora11

Cancel **Wired** **Apply**

Details Identity **IPv4** IPv6 Security

IPv4 Method

Automatic (DHCP) Link-Local Only
 Manual Disable
 Shared to other computers

Addresses

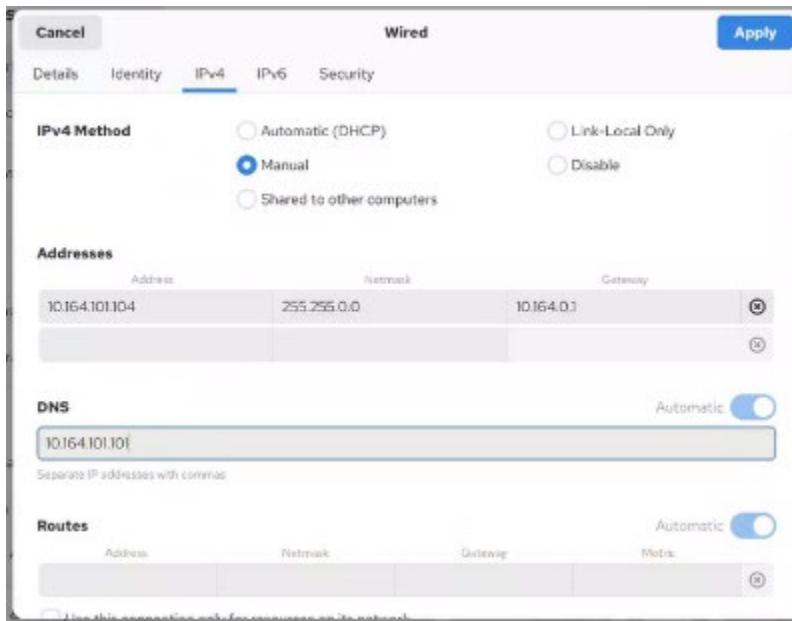
Address	Netmask	Gateway	
10.164.101.103	255.255.0.0	10.164.0.1	

DNS Automatic
10.164.101.101
Separate IP addresses with commas

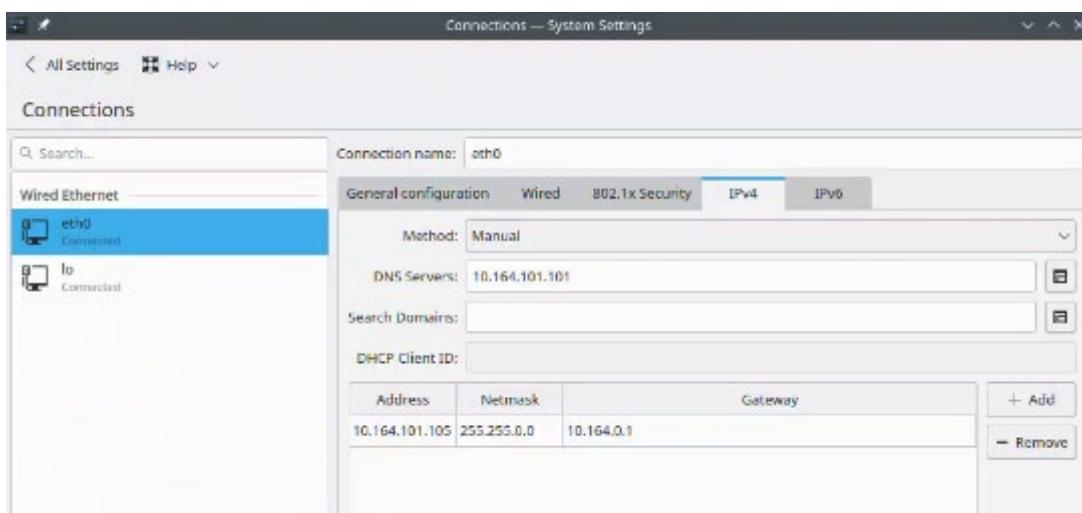
Routes Automatic

Address	Netmask	Gateway	Metric	

Centos1



OpenSUSE



3.1.1.3 Install ssh and Configure SSH keys

Verify ssh is installed up and running

Fedora

```
root@federal:~# systemctl status sshd.service
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: disabled)
  Drop-In: /usr/lib/systemd/system/service.d
    └─10-timeout-abort.conf, 50-keep-warm.conf
    Active: active (running) since Thu 2025-03-20 20:06:02 EDT; 30min ago
  Invocation: 93982d4883ad4e12bfada6fa99ea2a6e
    Docs: man:sshd(8)
          man:sshd_config(5)
  Main PID: 977 (sshd)
    Tasks: 1 (limit: 4587)
   Memory: 1.4M (peak: 1.7M)
      CPU: 25ms
     CGroup: /system.slice/sshd.service
             └─977 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Mar 20 20:06:02 federal systemd[1]: Starting sshd.service - OpenSSH server daemon...
Mar 20 20:06:02 federal sshd[977]: Server listening on 0.0.0.0 port 22.
Mar 20 20:06:02 federal sshd[977]: Server listening on :: port 22.
Mar 20 20:06:02 federal systemd[1]: Started sshd.service - OpenSSH server daemon.
root@federal:~# systemctl list-units | grep ssh
```

Centos

```
ansible@centos1:~$ systemctl status sshd.service
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: enabled)
  Active: active (running) since Thu 2025-03-20 20:06:11 EDT; 32min ago
  Invocation: 5f244ec3023b43b3b645eda535a5fd37
    Docs: man:sshd(8)
          man:sshd_config(5)
  Main PID: 1050 (sshd)
    Tasks: 1 (limit: 22916)
   Memory: 1.8M (peak: 2.1M)
      CPU: 40ms
     CGroup: /system.slice/sshd.service
             └─1050 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"
ansible@centos1:~$ █
```

Ansible (ubuntu)

```
root@ansible1:~# systemctl status ssh.socket
● ssh.socket - OpenBSD Secure Shell server socket
  Loaded: loaded (/usr/lib/systemd/system/ssh.socket; enabled; preset: enabled)
  Active: active (listening) since Thu 2025-03-20 16:42:30 EDT; 4h 4min ago
  Triggers: ● ssh.service
  Listen: [::]:22 (Stream)
    Tasks: 0 (limit: 9382)
   Memory: 8.0K (peak: 256.0K)
      CPU: 790us
     CGroup: /system.slice/ssh.socket

Mar 20 16:42:30 ansible1 systemd[1]: Listening on ssh.socket - OpenBSD Secure Shell server socket.
root@ansible1:~# █
```

Ubuntu1a

```

ansible@ubuntu1a:~/Desktop$ su -
Password:
root@ubuntu1a:~# systemctl status ssh.socket
● ssh.socket - OpenBSD Secure Shell server socket
  Loaded: loaded (/usr/lib/systemd/system/ssh.socket; enabled; preset: enabled)
  Active: active (listening) since Thu 2025-03-20 20:51:06 EDT; 1min 20s ago
  Triggers: ● ssh.service
  Listen: [::]:22 (Stream)
    Tasks: 0 (limit: 4551)
   Memory: 8.0K (peak: 256.0K)
     CPU: 663us
    CGroup: /system.slice/ssh.socket

Mar 20 20:51:06 ubuntu1a systemd[1]: Listening on ssh.socket - OpenBSD Secure Shell server socket.
root@ubuntu1a:~#

```

Opensuse

```

22 2025-03-20 20:00:47 history
opensuse1:~ # systemctl status sshd.service
● sshd.service - OpenSSH Daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: disabled)
  Active: active (running) since Thu 2025-03-20 16:42:49 EDT; 3h 18min ago
    Process: 1133 ExecStartPre=/usr/sbin/sshd-gen-keys-start (code=exited, status=0/SUCCESS)
    Process: 1145 ExecStartPre=/usr/sbin/sshd -t $SSHD_OPTS (code=exited, status=0/SUCCESS)
  Main PID: 1156 (sshd)
    Tasks: 1
      CPU: 35ms
     CGroup: /system.slice/sshd.service
             └─1156 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Mar 20 16:42:48 opensuse1 sshd-gen-keys-start[1133]: Checking for missing server keys in /etc/ssh
Mar 20 16:42:48 opensuse1 systemd[1]: Starting OpenSSH Daemon...
Mar 20 16:42:49 opensuse1 sshd[1156]: Server listening on 0.0.0.0 port 22.
Mar 20 16:42:49 opensuse1 systemd[1]: Started OpenSSH Daemon.
Mar 20 16:42:49 opensuse1 sshd[1156]: Server listening on :: port 22.
opensuse1:~ #

```

3.1.1.4 Setup ansible user with sudo privileges

CentOS

1. Create User with Password:

```

sudo useradd -m -s /bin/bash ansible
echo "Amf123456" | sudo passwd --stdin ansible

```

2. Grant Sudo Privileges with visudo:

```

sudo visudo

```

- Add the following line at the end of the file:

```

ansible ALL=(ALL) NOPASSWD:ALL

```

- Save and exit (:wq in vi).

Fedora

1. Create User with Password:

```
sudo useradd -m -s /bin/bash ansible  
echo "Amf123456" | sudo passwd --stdin ansible
```

2. Grant Sudo Privileges with visudo:

```
sudo visudo
```

- Add:

```
ansible ALL=(ALL) NOPASSWD:ALL
```

- Save and exit.

Ubuntu

1. Create User with Password:

```
sudo useradd -m -s /bin/bash ansible  
echo "ansible: Amf123456" | sudo chpasswd
```

2. Grant Sudo Privileges with visudo:

```
sudo visudo
```

- Add:

```
ansible ALL=(ALL) NOPASSWD:ALL
```

- Save and exit.

openSUSE

1. Create User with Password:

```
sudo useradd -m -s /bin/bash ansible  
echo "Amf123456" | sudo passwd --stdin ansible
```

2. Grant Sudo Privileges with visudo:

```
sudo visudo
```

- Add:

```
ansible ALL=(ALL) NOPASSWD:ALL
```

- Save and exit.

3.1.1.5 Create ansible key in ansible machine

```
cd /home/ansible/
```

```
ls -a
```

Create a ssh key

```
ssh-keygen
```

```
cd /home/ansible/.ssh
```

```
ls -a
```

```
### Look for public key file
```

```
8ZuLENEpxPIKespkLnZlpC
root@ansible1:/home/ansible/.ssh# ls
id_ed25519  id_ed25519.pub  known_hosts  known_hosts.old
root@ansible1:/home/ansible/.ssh# pwd
/home/ansible/.ssh
root@ansible1:/home/ansible/.ssh# █
```

Copy to ubuntu1a

```
ssh-copy-id -i .ssh/id_ed25519.pub ubuntu1a.ansible1.com
```

```
ssh ubuntu1a.ansible1.com
```

```
exit
```

```
cd /home/ansible/
```

```
Connection to opensuse1.ansible1.com closed.
ansible@ansible1:~/.ansible/tmp$ ssh ubuntu1a.ansible1.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

28 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

8 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Thu Mar 20 21:17:42 2025 from 10.164.101.101
ansible@ubuntu1a:~$ exit
logout
Connection to ubuntu1a.ansible1.com closed.
ansible@ansible1:~/.ansible/tmp$ █
```

Copy to fedora

ssh-copy-id -i .ssh/id_ed25519.pub fedora1.ansible1.com and now test

ssh fedora1.ansible1.com

exit

```
ansible@ansible1:~/.ansible/tmp$ ssh ubuntu1a.ansible1.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

28 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

8 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Mon Mar 17 01:46:45 2025 from 10.164.101.101
ansible@ubuntu1a:~$ pwd
/home/ansible
ansible@ubuntu1a:~$ ls
```

Centos

```
ssh-copy-id -i .ssh/id_ed25519.pub centos1.ansible1.com
```

```
ssh centos1.ansible1.com
```

```
ansible@ansible1:~/.ansible/tmp$ ssh centos1.ansible1.com
Last login: Thu Mar 20 20:37:26 2025
ansible@centos1:~$ ls
Desktop Documents Downloads folder1 folder2 folder3 Music
ansible@centos1:~$ exit
logout
Connection to centos1.ansible1.com closed.
ansible@ansible1:~/.ansible/tmp$
```

Opensuse

```
Allow ssh in Opensuse
```

```
sudo firewall-cmd --permanent --add-service=ssh
```

```
sudo firewall-cmd --reload
```

```
ssh-copy-id -i .ssh/id_ed25519.pub opensuse1.ansible1.com
```

```
ssh opensuse1.ansible1.com
```

```
ansible@ansible1:~/.ansible/tmp$ ssh opensuse1.ansible1.com
Have a lot of fun...
Last login: Mon Mar 17 01:46:59 2025 from 10.164.101.101
ansible@opensuse1:~> exit
logout
Connection to opensuse1.ansible1.com closed.
ansible@ansible1:~/.ansible/tmp$
```

3.1.1.6 Configuring an Inventory Inside the Ansible Hosts File

Video Reference

Lesson <https://tutorialsonline.ca/courses/ansible/lectures/31421094> 8 - Install and Configure Centos 8 and Configure Inventory inside hosts file

1- In ubuntu Ansible server, open /etc/ansible/hosts in text editor using sudo priviledges:

```
ansible@ansible1:/etc/ansible$ cat hosts
```

```
ansible@ansible1:/etc/ansible$ cat hosts
# This is the default ansible 'hosts' file.
#
# It should live in /etc/ansible/hosts
#
# - Comments begin with the '#' character
# - Blank lines are ignored
# - Groups of hosts are delimited by [header] elements
# - You can enter hostnames or ip addresses
# - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers:

## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group:

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern, you can specify
# them like this:

## www[001:006].example.com

# You can also use ranges for multiple hosts:

## db-[99:101]-node.example.com

# Ex 3: A collection of database servers in the 'dbservers' group:

## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Ex4: Multiple hosts arranged into groups such as 'Debian' and 'openSUSE':
```

```
## [Debian]
## alpha.example.org
## beta.example.org

## [openSUSE]
## green.example.com
## blue.example.com
```

sudo nano /etc/ansible/hosts

NOTE - Fedora uses /usr/bin/python3 as the system default, which points to Python 3.13.2. Setting ansible_python_interpreter=/usr/bin/python3 aligns with what's actually installed.

The other hosts (centos1, ubuntu1a, opensuse1) have Python 3.12 at /usr/bin/python3.12, so those settings are correct.

Add the following text at the bottom of the file:

```
[centoshosts]
centos1.ansible1.com ansible_python_interpreter=/usr/bin/python3.12

[fedorahosts]
fedora1.ansible1.com ansible_python_interpreter=/usr/bin/python3

[ubuntuhosts]
ubuntu1a.ansible1.com ansible_python_interpreter=/usr/bin/python3.12

[opensusehosts]
opensuse1.ansible1.com ansible_python_interpreter=/usr/bin/python3.12

[linuxservers:children]
centoshosts
fedorahosts
ubuntuhosts
opensusehosts
[linuxservers:children]
centoshosts
fedorahosts
```

ubuntuhosts

opensusehosts

Verify changes

cat hosts

```
# This is the default ansible 'hosts' file.
#
# It should live in /etc/ansible/hosts
#
#   - Comments begin with the '#' character
#   - Blank lines are ignored
#   - Groups of hosts are delimited by [header] elements
#   - You can enter hostnames or ip addresses
#   - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers:

## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group:

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern, you can specify
# them like this:

## www[001:006].example.com

# You can also use ranges for multiple hosts:

## db-[99:101]-node.example.com

# Ex 3: A collection of database servers in the 'dbservers' group:

## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Ex4: Multiple hosts arranged into groups such as 'Debian' and 'openSUSE':

## [Debian]
## alpha.example.org
## beta.example.org

## [openSUSE]
## green.example.com
## blue.example.com
```

```
[centoshosts]
centos1.ansible1.com ansible_python_interpreter=/usr/bin/python3.12

[fedorahosts]
fedoral.ansible1.com ansible_python_interpreter=/usr/bin/python3

[ubuntuhosts]
ubuntula.ansible1.com ansible_python_interpreter=/usr/bin/python3.12

[opensusehosts]
opensuse1.ansible1.com ansible_python_interpreter=/usr/bin/python3.12

[linuxservers:children]
centoshosts
fedorahosts
ubuntuhosts
opensusehosts
```

2. Test the file to see is correct

```
ansible-inventory --list -i /etc/ansible/hosts
```

```
ansible@ansible1:/etc/ansible$ ansible-inventory --list -i /etc/ansible/hosts
{
    "_meta": {
        "hostvars": {
            "centos1.ansible1.com": {
                "ansible_python_interpreter": "/usr/bin/python3.12"
            },
            "fedoral.ansible1.com": {
                "ansible_python_interpreter": "/usr/bin/python3"
            },
            "opensuse1.ansible1.com": {
                "ansible_python_interpreter": "/usr/bin/python3.12"
            },
            "ubuntula.ansible1.com": {
                "ansible_python_interpreter": "/usr/bin/python3.12"
            }
        }
    },
    "all": {
        "children": [
            "ungrouped",
            "linuxservers"
        ]
    },
    "centoshosts": {
        "hosts": [
            "centos1.ansible1.com"
        ]
    },
    "fedorahosts": {
        "hosts": [
            "fedoral.ansible1.com"
        ]
    },
    "linuxservers": {
        "children": [
            "centoshosts",
            "fedorahosts",
            "ubuntuhosts",
            "opensusehosts"
        ]
    }
}
```

```
        ],
    },
    "opensusehosts": {
        "hosts": [
            "opensuse1.ansible1.com"
        ]
    },
    "ubuntuhosts": {
        "hosts": [
            "ubuntu1a.ansible1.com"
        ]
    }
}
ansible@ansible1:/etc/ansible$
```

3. To test connectivity with each host, issue the following command:
ansible -m ping all

```
ansible@ansible1:/etc/ansible$ ansible -m ping all
centos1.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
opensuse1.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
ubuntu1a.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
fedora1.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
```

```
centoshosts
ansible@ansible1:/etc/ansible$ ansible -m ping all
centos1.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
opensuse1.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
ubuntu1a.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
fedora1.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
ansible@ansible1:/etc/ansible$
```

4. To test connectivity with groups, issue the following commands:

```
$ ansible -m ping centoshosts
$ ansible -m ping fedorahosts
$ ansible -m ping opensusehosts
$ ansible -m ping ubuntuhosts
```

```
centoshosts
ansible@ansible1:/etc/ansible$ ansible -m ping centoshosts
centos1.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
fedorahosts
fedora1.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
ubuntuhosts
ubuntu1a.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
opensusehosts
opensuse1.ansible1.com | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
ansible@ansible1:/etc/ansible$
```

3.1.1.7 Snapshots of VM setup for ansible automation

Create snapshots of all 5 ansible VMs and save each snapshot as "[OS]ansible4-afterinstall".

3.1.2 Ansible playbooks

3.1.2.1 Create and Run an Ansible Playbook that Prints the System Status, Info and Username to a text file in each Linux host

Reference

Lesson 10 - Ansible Playbooks

<https://tutorialsonline.ca/courses/1330884/lectures/31428925>

1. On your Ansible server, login as ansible, then move to /home/ansible/ and create a /home/ansible/Documents directory, then move to /home/ansible/Documents and create yaml file name "intro.yml":

```
root@ansiblesrv4:/# su ansible ansible@ansiblesrv4:/$ cd /home/ansible  
ansible@ansiblesrv4:/home/ansible$ mkdir Documents  
ansible@ansiblesrv4:/home/ansible$ cd Documents/  
ansible@ansiblesrv4:~/Documents$ vi intro.yml
```

2. Paste the following text into intro.yml:

```
---  
- name: get stats and write to desktop  
hosts: linuxservers  
tasks:  
- name: get system status and info  
shell: uname -a > /home/ansible/Desktop/output3.txt  
- name: print my username  
shell: whoami >> /home/ansible/Desktop/output3.txt
```

```
ansible@ansible1:/etc/ansible$ cd /home/ansible/
ansible@ansible1:$ mkdir Documents
ansible@ansible1:$ cd Documents/
ansible@ansible1:~/Documents$ vi intro.yml
ansible@ansible1:~/Documents$ cat intro.yml
---
- name: get stats and write to desktop
  hosts: linuxservers
  tasks:
    - name: get system status and info
      shell: uname -a > /home/ansible/Desktop/output3.txt
    - name: print my username
      shell: whoami  >> /home/ansible/Desktop/output3.txt
```

3. Create a /home/ansible/Documents directory **in each Linux server** by issuing the following command:

```
mkdir /home/ansible/Desktop
```

```
ansible@ansible1:~/Documents$ ssh centos1.ansible1.com
Last login: Sun Mar 16 19:18:06 2025 from 10.164.101.101
ansible@centos1:~$ pwd
/home/ansible
ansible@centos1:~$ mkdir Desktop
ansible@centos1:~$ ls
Desktop
ansible@centos1:~$ exit
logout
Connection to centos1.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh fedora1.ansible1.com
Last login: Sun Mar 16 19:18:22 2025 from 10.164.101.101
ansible@fedora1:~$ mkdir Desktop
ansible@fedora1:~$ ls
Desktop test.txt
ansible@fedora1:~$ exit
logout
Connection to fedora1.ansible1.com closed.
```

```
ansible@ansible1:~/Documents$ ssh ubuntu1a.ansible1.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

8 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Sun Mar 16 19:18:31 2025 from 10.164.101.101
ansible@ubuntu1a:~$ pwd
/home/ansible
ansible@ubuntu1a:~$ mkdir Desktop
ansible@ubuntu1a:~$ ls
Desktop  snap
ansible@ubuntu1a:~$ exit
logout
Connection to ubuntu1a.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh opensuse1.ansible1.com
Have a lot of fun...
Last login: Sun Mar 16 19:18:42 2025 from 10.164.101.101
ansible@opensuse1:~> pwd
/home/ansible
ansible@opensuse1:~> mkdir Desktop
ansible@opensuse1:~> ls
bin  Desktop
```

4. Go back to the Ansible server, and run the intro.yml ansible playbook by issuing the following command:

```
# ansible-playbook intro.yml
```

```

ansible@ansible1:~/Documents$ ansible-playbook intro.yml

PLAY [get stats and write to desktop] *****

TASK [Gathering Facts] *****
ok: [ubuntula.ansible1.com]
ok: [opensuse1.ansible1.com]
ok: [centos1.ansible1.com]
ok: [fedora1.ansible1.com]

TASK [get system status and info] *****
changed: [ubuntula.ansible1.com]
changed: [centos1.ansible1.com]
changed: [opensuse1.ansible1.com]
changed: [fedora1.ansible1.com]

TASK [print my username] *****
changed: [ubuntula.ansible1.com]
changed: [centos1.ansible1.com]
changed: [fedora1.ansible1.com]
changed: [opensuse1.ansible1.com]

PLAY RECAP *****
centos1.ansible1.com      : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
fedora1.ansible1.com     : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
opensuse1.ansible1.com   : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ubuntula.ansible1.com    : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ansible@ansible1:~/Documents$
```

```

ansible@ansible1:~/Documents$ ansible-playbook intro.yml

PLAY [get stats and write to desktop] *****

TASK [Gathering Facts] *****
ok: [ubuntula.ansible1.com]
ok: [opensuse1.ansible1.com]
ok: [centos1.ansible1.com]
ok: [fedora1.ansible1.com]

TASK [get system status and info] *****
changed: [ubuntula.ansible1.com]
changed: [centos1.ansible1.com]
changed: [opensuse1.ansible1.com]
changed: [fedora1.ansible1.com]

TASK [print my username] *****
changed: [ubuntula.ansible1.com]
changed: [centos1.ansible1.com]
changed: [fedora1.ansible1.com]
changed: [opensuse1.ansible1.com]

PLAY RECAP *****
centos1.ansible1.com      : ok=3    changed=2    unreachable=0    failed=0    skipped=8    rescued=8    ignored=8
fedora1.ansible1.com     : ok=3    changed=2    unreachable=0    failed=0    skipped=8    rescued=8    ignored=8
opensuse1.ansible1.com   : ok=3    changed=2    unreachable=0    failed=0    skipped=8    rescued=8    ignored=8
ubuntula.ansible1.com    : ok=3    changed=2    unreachable=0    failed=0    skipped=8    rescued=8    ignored=8

ansible@ansible1:~/Documents$
```

5. Verify that output1.txt has been created in the /home/ansible/Documents directory in each Linux server by issuing the following command on each server:

```

ls /home/ansible/Desktop
# cat /home/ansible/Desktop/output1.txt
```

```
ansible@ansible1:~/Documents$ ssh opensuse1.ansible1.com
```

```
Have a lot of fun...
Last login: Sun Mar 16 19:48:35 2025 from 10.164.101.101
ansible@opensuse1:~> ls
bin Desktop
ansible@opensuse1:~> cd Desktop/
ansible@opensuse1:~/Desktop> ls
output3.txt
ansible@opensuse1:~/Desktop> cat output3.txt
Linux opensuse1 6.4.0-150600.23.38-default #1 SMP PREEMPT_DYNAMIC Thu Feb 6 08:53:28 UTC
2025 (cb92f8c) x86_64 x86_64 x86_64 GNU/Linux
ansible
ansible@opensuse1:~/Desktop> exit
logout
Connection to opensuse1.ansible1.com closed.
```

```
ansible@ansible1:~/Documents$ ssh ubuntula.ansible1.com
```

```
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)
```

```
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro
```

```
Expanded Security Maintenance for Applications is not enabled.
```

```
11 updates can be applied immediately.
```

```
To see these additional updates run: apt list --upgradable
```

```
8 additional security updates can be applied with ESM Apps.
```

```
Learn more about enabling ESM Apps service at https://ubuntu.com/esm
```

```
Last login: Sun Mar 16 19:48:35 2025 from 10.164.101.101
```

```
ansible@ubuntula:~$ cd Desktop/
ansible@ubuntula:~/Desktop$ ls
output3.txt
ansible@ubuntula:~/Desktop$ cat output3.txt
Linux ubuntula 6.11.0-19-generic #19~24.04.1-Ubuntu SMP PREEMPT_DYNAMIC Mon Feb 17 11:51:52
UTC 2 x86_64 x86_64 x86_64 GNU/Linux
ansible
ansible@ubuntula:~/Desktop$ exit
logout
Connection to ubuntula.ansible1.com closed.
```

```
ansible@ansible1:~/Documents$ ssh fedoral.ansible1.com
```

```
Last login: Sun Mar 16 19:48:35 2025 from 10.164.101.101
```

```
ansible@fedoral:~$ cd Desktop/
ansible@fedoral:~/Desktop$ ls
output3.txt
ansible@fedoral:~/Desktop$ cat output3.txt
Linux fedoral 6.13.5-200.fc41.x86_64 #1 SMP PREEMPT_DYNAMIC Thu Feb 27 15:07:31 UTC 2025
x86_64 GNU/Linux
ansible
ansible@fedoral:~/Desktop$ exit
logout
Connection to fedoral.ansible1.com closed.
ansible@ansible1:~/Documents$
```

```
ansible@ansible1:~/Documents$ ssh centos1.ansible1.com
Last login: Sun Mar 16 19:48:35 2025 from 10.164.101.101
ansible@centos1:~$ cd Desktop/
ansible@centos1:~/Desktop$ ls
output3.txt
ansible@centos1:~/Desktop$ cat output3.txt
Linux centos1 6.12.0-59.el10.x86_64 #1 SMP PREEMPT_DYNAMIC Tue Mar 4 18:58:50 UTC 2025
x86_64 GNU/Linux
ansible
ansible@centos1:~/Desktop$
```

```
ansible@ansible1:~/Documents$ ssh opensuse1.ansible1.com
Have a lot of fun...
Last login: Sun Mar 16 19:48:35 2025 from 10.164.101.101
ansible@opensuse1:~$ ls
bin  Desktop
ansible@opensuse1:~$ cd Desktop/
ansible@opensuse1:~/Desktop$ ls
output3.txt
ansible@opensuse1:~/Desktop$ cat output3.txt
Linux openSUSE1 6.4.0-150600.23.38-default #1 SMP PREEMPT_DYNAMIC Thu Feb 6 08:53:28 UTC 2025 (cb92f8c) x86_64 x86_64 x86_64 GNU/Linux
ansible
ansible@opensuse1:~/Desktop$
```

```
connection to openSUSE1 closed.
ansible@ansible1:~/Documents$ ssh ubuntu1.ansible1.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

8 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Sun Mar 16 19:48:35 2025 from 10.164.101.101
ansible@ubuntu1:~$ cd Desktop/
ansible@ubuntu1:~/Desktop$ ls
output3.txt
ansible@ubuntu1:~/Desktop$ cat output3.txt
Linux ubuntu1 6.11.0-19-generic #19-24.04.1-Ubuntu SMP PREEMPT_DYNAMIC Mon Feb 17 11:51:52 UTC 2 x86_64 x86_64 x86_64 GNU/Linux
ansible
ansible@ubuntu1:~/Desktop$
```

```

Connection to ubuntu1.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh fedora1.ansible1.com
Last login: Sun Mar 16 19:48:35 2025 from 10.164.101.101
ansible@fedora1:~$ cd Desktop/
ansible@fedora1:~/Desktop$ ls
output3.txt
ansible@fedora1:~/Desktop$ cat output3.txt
Linux fedora 6.13.5-200.fc41.x86_64 #1 SMP PREEMPT_DYNAMIC Thu Feb 27 15:07:31 UTC 2025 x86_64 GNU/Linux
ansible
ansible@fedora1:~/Desktop$ exit
logout
Connection to fedora1.ansible1.com closed.
ansible@ansible1:~/Documents$ 
ansible@ansible1:~/Documents$ ssh centos1.ansible1.com
Last login: Sun Mar 16 19:48:35 2025 from 10.164.101.101
ansible@centos1:~$ cd Desktop/
ansible@centos1:~/Desktop$ ls
output3.txt
ansible@centos1:~/Desktop$ cat output3.txt
Linux centos1 6.12.0-59.el10.x86_64 #1 SMP PREEMPT_DYNAMIC Tue Mar 4 18:58:50 UTC 2025 x86_64 GNU/Linux
ansible
ansible@centos1:~/Desktop$ 

```

Sample Use of modules

1. To verify the uptime of all Linux servers, issue the following command:

```
ansible -m raw -a '/usr/bin/uptime' linuxservers
```

2. To verify which Linux servers are running Python3, issue the following command:

```
ansible -m raw -a 'python3 -V' linuxservers
```

3. To verify which user you are executing the commands as, issue the following command:

```
ansible all -a 'whoami'
```

4. To become the root user, and execute a command, use the "-b" option, which stands for "become":

```
ansible all -b -a 'whoami'
```

```

ansible@ansible1:~/Documents$ ansible -m raw -a '/usr/bin/uptime' linuxservers
opensuse1.ansible1.com | CHANGED | rc=0 >>
20:08:51 up 2:56, 4 users, load average: 0.00, 0.00, 0.00
Shared connection to opensuse1.ansible1.com closed.

centos1.ansible1.com | CHANGED | rc=0 >>
20:08:51 up 2:56, 4 users, load average: 0.05, 0.01, 0.00
Shared connection to centos1.ansible1.com closed.

fedora1.ansible1.com | CHANGED | rc=0 >>
20:08:51 up 2:56, 6 users, load average: 0.48, 0.19, 0.12
Shared connection to fedora1.ansible1.com closed.

ubuntula.ansible1.com | CHANGED | rc=0 >>
20:08:51 up 2:56, 2 users, load average: 0.02, 0.01, 0.00
Shared connection to ubuntula.ansible1.com closed.

ansible@ansible1:~/Documents$ ansible -m raw -a 'python3 -V' linuxservers
ubuntula.ansible1.com | CHANGED | rc=0 >>
Python 3.12.3

```

```
Shared connection to ubuntula.ansible1.com closed.

centos1.ansible1.com | CHANGED | rc=0 >>
Python 3.12.9
Shared connection to centos1.ansible1.com closed.

opensuse1.ansible1.com | CHANGED | rc=0 >>
Python 3.6.15
Shared connection to opensuse1.ansible1.com closed.

fedoral.ansible1.com | CHANGED | rc=0 >>
Python 3.13.2
Shared connection to fedoral.ansible1.com closed.

ansible@ansible1:~/Documents$ ansible all -a 'whoami'
ubuntula.ansible1.com | CHANGED | rc=0 >>
ansible
centos1.ansible1.com | CHANGED | rc=0 >>
ansible
fedoral.ansible1.com | CHANGED | rc=0 >>
ansible
opensuse1.ansible1.com | CHANGED | rc=0 >>
ansible
ansible@ansible1:~/Documents$ ansible all -b -a 'whoami'
ubuntula.ansible1.com | CHANGED | rc=0 >>
root
centos1.ansible1.com | CHANGED | rc=0 >>
root
opensuse1.ansible1.com | CHANGED | rc=0 >>
root
fedoral.ansible1.com | CHANGED | rc=0 >>
root
ansible@ansible1:~/Documents$
```

```
ansible@ansible1:~/Documents$ ansible -m raw -a '/usr/bin/uptime' linuxservers
opensuse1.ansible1.com | CHANGED | rc=0 >>
20:08:51 up 2:56, 4 users, load average: 0.00, 0.00, 0.00
Shared connection to opensuse1.ansible1.com closed.

centos1.ansible1.com | CHANGED | rc=0 >>
20:08:51 up 2:56, 4 users, load average: 0.05, 0.01, 0.00
Shared connection to centos1.ansible1.com closed.

fedoral.ansible1.com | CHANGED | rc=0 >>
20:08:51 up 2:56, 6 users, load average: 0.48, 0.19, 0.12
Shared connection to fedoral.ansible1.com closed.

ubuntula.ansible1.com | CHANGED | rc=0 >>
20:08:51 up 2:56, 2 users, load average: 0.02, 0.01, 0.00
Shared connection to ubuntula.ansible1.com closed.

ansible@ansible1:~/Documents$
```

```
ansible@ansible1:~/Documents$ ansible -m raw -a 'python3 -V' linuxservers
ubuntu1a.ansible1.com | CHANGED | rc=0 >>
Python 3.12.3
Shared connection to ubuntu1a.ansible1.com closed.

centos1.ansible1.com | CHANGED | rc=0 >>
Python 3.12.9
Shared connection to centos1.ansible1.com closed.

opensuse1.ansible1.com | CHANGED | rc=0 >>
Python 3.6.15
Shared connection to opensuse1.ansible1.com closed.

fedora1.ansible1.com | CHANGED | rc=0 >>
Python 3.13.2
Shared connection to fedora1.ansible1.com closed.

ansible@ansible1:~/Documents$
```

```
ansible@ansible1:~/Documents$ ansible all -a 'whoami'
ubuntu1a.ansible1.com | CHANGED | rc=0 >>
ansible
centos1.ansible1.com | CHANGED | rc=0 >>
ansible
fedora1.ansible1.com | CHANGED | rc=0 >>
ansible
opensuse1.ansible1.com | CHANGED | rc=0 >>
ansible
ansible@ansible1:~/Documents$ █
```

```
ansible
ansible@ansible1:~/Documents$ ansible all -b -a 'whoami'
ubuntu1a.ansible1.com | CHANGED | rc=0 >>
root
centos1.ansible1.com | CHANGED | rc=0 >>
root
opensuse1.ansible1.com | CHANGED | rc=0 >>
root
fedora1.ansible1.com | CHANGED | rc=0 >>
root
ansible@ansible1:~/Documents$
```

3.1.2.2 Create and Run an Ansible Playbook that creates a file in each Linux server as the Ansible User

1. On your Ansible server, login as ansible, then create yaml file name "sample1.yml" in /home/ansible/Documents:

```
sudo vi sample1.yml
```

2. Paste in the following text:

```
---
```

```
- hosts: linuxservers
  become: false
  tasks:
    - name: Create a file
      file: path=/home/ansible/test11.txt state=touch
```

```
ansible@ansible1:~/Documents$ cat sample1.yml
sample1.yml
```

```
---
```

```
- hosts: linuxservers
  become: true
  tasks:
    - name: Create a file
      file: path=/home/ansible/test2.txt state=touch
```

```
ansible@ansible1:~/Documents$
```

```
ansible@ansible1:~/Documents$ cat sample1.yml
---
- hosts: linuxservers
  become: true
  tasks:
    - name: Create a file
      file: path=/home/ansible/test2.txt state=touch

ansible@ansible1:~/Documents$
```

3. Run the sample1.yml ansible playbook by issuing the following command:
ansible-playbook sample1.yml

```

ansible@ansible1:~/Documents$ vi sample1.yml
ansible@ansible1:~/Documents$ ansible-playbook sample1.yml

PLAY [linuxservers] ****
TASK [Gathering Facts] ****
ok: [ubuntu1a.ansible1.com]
ok: [centos1.ansible1.com]
ok: [opensuse1.ansible1.com]
ok: [fedora1.ansible1.com]

TASK [Create a file] ****
changed: [ubuntu1a.ansible1.com]
changed: [centos1.ansible1.com]
changed: [opensuse1.ansible1.com]
changed: [fedora1.ansible1.com]

PLAY RECAP ****
centos1.ansible1.com    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
fedora1.ansible1.com    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
opensuse1.ansible1.com  : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ubuntu1a.ansible1.com   : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

```

```

ansible@ansible1:~/Documents$ ansible-playbook sample1.yml

PLAY [linuxservers] ****
TASK [Gathering Facts] ****
ok: [ubuntu1a.ansible1.com]
ok: [centos1.ansible1.com]
ok: [opensuse1.ansible1.com]
ok: [fedora1.ansible1.com]

TASK [Create a file] ****
changed: [ubuntu1a.ansible1.com]
changed: [centos1.ansible1.com]
changed: [opensuse1.ansible1.com]
changed: [fedora1.ansible1.com]

PLAY RECAP ****
centos1.ansible1.com    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
fedora1.ansible1.com    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
opensuse1.ansible1.com  : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ubuntu1a.ansible1.com   : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

```

4. Verify the output in each Linux machine

```

ansible@ansible1:~/Documents$ ssh fedora1.ansible1.com
Last login: Sun Mar 16 21:22:13 2025 from 10.164.101.101
ansible@fedora1:~$ ls
Desktop test2.txt test.txt
ansible@fedora1:~$ exit
logout
Connection to fedora1.ansible1.com closed.

ansible@ansible1:~/Documents$ ssh ubuntu1a.ansible1.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)

```

- * Documentation: <https://help.ubuntu.com>
- * Management: <https://landscape.canonical.com>

* Support: <https://ubuntu.com/pro>

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.

To see these additional updates run: apt list --upgradable

8 additional security updates can be applied with ESM Apps.

Learn more about enabling ESM Apps service at <https://ubuntu.com/esm>

Last login: Sun Mar 16 21:17:05 2025 from 10.164.101.101

ansible@ubuntu1a:~\$ ls

Desktop snap test2.txt

ansible@ubuntu1a:~\$ exit

logout

Connection to ubuntu1a.ansible1.com closed.

ansible@ansible1:~/Documents\$ ssh centos1.ansible1.com

Last login: Sun Mar 16 21:17:05 2025 from 10.164.101.101

ansible@centos1:~\$ ls

Desktop test2.txt

ansible@centos1:~\$ exit

logout

Connection to centos1.ansible1.com closed.

ansible@ansible1:~/Documents\$ ssh opensuse1.ansible1.com

Have a lot of fun...

Last login: Sun Mar 16 21:17:05 2025 from 10.164.101.101

ansible@opensuse1:~> ls

bin Desktop test2.txt

ansible@opensuse1:~> exit

logout

Connection to opensuse1.ansible1.com closed.

ansible@ansible1:~/Documents\$

```
Connection to fedora1.ansible1.com closed.
ansible@ansible1:~/Documents$ 
ansible@ansible1:~/Documents$ 
ansible@ansible1:~/Documents$ 
ansible@ansible1:~/Documents$ ssh fedora1.ansible1.com
Last login: Sun Mar 16 21:22:13 2025 from 10.164.101.101
ansible@fedora1:~$ ls
Desktop test2.txt test.txt
ansible@fedora1:~$ exit
logout
Connection to fedora1.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh ubuntu1a.ansible1.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

8 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Sun Mar 16 21:17:05 2025 from 10.164.101.101
ansible@ubuntu1a:~$ ls
Desktop snap test2.txt
ansible@ubuntu1a:~$ exit
logout
Connection to ubuntu1a.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh centos1.ansible1.com
Last login: Sun Mar 16 21:17:05 2025 from 10.164.101.101
ansible@centos1:~$ ls
Desktop test2.txt
ansible@centos1:~$ exit
```

```
ansible@ansible1:~/Documents$ ssh opensuse1.ansible1.com
Have a lot of fun...
Last login: Sun Mar 16 21:17:05 2025 from 10.164.101.101
ansible@opensuse1:~$ ls
bin Desktop test2.txt
ansible@opensuse1:~$ exit
logout
Connection to opensuse1.ansible1.com closed.
ansible@ansible1:~/Documents$ 
```

3.1.2.3 Create and Run an Ansible Playbook that Creates a Directory in Each Linux Host while Setting the Permissions, Ownership and Group for the Newly Created Directory

1. On your Ansible server, login as ansible, then create yaml file name "sample2.yml" in **/home/ansible/Documents:**

```
sudo vi sample2.yml
```

2. Paste in the following text:

```
---
```

```
- hosts: linuxservers
  become: true
  tasks:
    - name: Create directory
      file: path=/home/ansible/web state=directory mode=775 owner=ansible
            group=ansible
```

```
ansible@ansible1:~/Documents$ cat sample2.yml
---
- hosts: linuxservers
  become: true
  tasks:
    - name: Create directory
      file: path=/home/ansible/web state=directory mode=775 owner=ansible group=ansible
ansible@ansible1:~/Documents$
```

3. Run the script

```
ansible@ansible1:~/Documents$ ansible-playbook sample2.yml
PLAY [linuxservers]
*****
TASK [Gathering Facts] PLAY [linuxservers]
*****
ok: [ubuntula.ansible1.com]
ok: [centos1.ansible1.com]
ok: [federal.ansible1.com]
ok: [opensuse1.ansible1.com]

TASK [Create directory] PLAY [linuxservers]
*****
ok: [ubuntula.ansible1.com]
ok: [opensuse1.ansible1.com]
ok: [centos1.ansible1.com]
```

```

ok: [federal1.ansible1.com]

PLAY RECAP PLAY [linuxservers]
*****
centos1.ansible1.com      : ok=1    changed=0    unreachable=0    failed=0
skipped=0    rescued=0    ignored=0
federal1.ansible1.com     : ok=1    changed=0    unreachable=0    failed=0
skipped=0    rescued=0    ignored=0
opensuse1.ansible1.com    : ok=1    changed=0    unreachable=0    failed=0
skipped=0    rescued=0    ignored=0
ubuntula.ansible1.com     : ok=1    changed=0    unreachable=0    failed=0
skipped=0    rescued=0    ignored=0

ansible@ansible1:~/Documents$
```

4. Check results

```

ansible@ansible1:~/Documents$ ssh ubuntula.ansible1.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

8 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Sun Mar 16 21:48:30 2025 from 10.164.101.101
ansible@ubuntula:~$ ls
Desktop snap test2.txt web
ansible@ubuntula:~$ exit
logout
Connection to ubuntula.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh centos1.ansible1.com
Last login: Sun Mar 16 21:48:30 2025 from 10.164.101.101
ansible@centos1:~$ ls
Desktop test2.txt web
ansible@centos1:~$ exit
logout
Connection to centos1.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh federal1.ansible1.com
Last login: Sun Mar 16 21:48:30 2025 from 10.164.101.101
ansible@federal1:~$ ls
Desktop test2.txt test.txt web
ansible@federal1:~$ exit
logout
Connection to federal1.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh openssel.ansible1.com
ssh: Could not resolve hostname openssel.ansible1.com: Name or service not known
ansible@ansible1:~/Documents$ ssh opensuse1.ansible1.com
Have a lot of fun...
Last login: Sun Mar 16 21:48:30 2025 from 10.164.101.101
ansible@opensuse1:~> ls
bin Desktop test2.txt web
ansible@opensuse1:~> exit
```

```
logout
Connection to opensuse1.ansible1.com closed.
ansible@ansible1:~/Documents$
```

```
ansible@ansible1:~/Documents$ ssh ubuntu1a.ansible1.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

8 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Sun Mar 16 21:48:30 2025 from 10.164.101.101
ansible@ubuntu1a:~$ ls
Desktop snap test2.txt web
ansible@ubuntu1a:~$ exit
logout
Connection to ubuntu1a.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh centos1.ansible1.com
Last login: Sun Mar 16 21:48:30 2025 from 10.164.101.101
ansible@centos1:~$ ls
Desktop test2.txt web
ansible@centos1:~$ exit
logout
Connection to centos1.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh fedora1.ansible1.com
Last login: Sun Mar 16 21:48:30 2025 from 10.164.101.101
ansible@fedora1:~$ ls
Desktop test2.txt test.txt web
ansible@fedora1:~$ exit
logout
Connection to fedora1.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh opensse1.ansible1.com
ssh: Could not resolve hostname opensse1.ansible1.com: Name or service not known
ansible@ansible1:~/Documents$ ssh openuse1.ansible1.com
Have a lot of fun...
Last login: Sun Mar 16 21:48:30 2025 from 10.164.101.101
ansible@openuse1:~> ls
bin Desktop test2.txt web
ansible@openuse1:~> exit
logout
Connection to openuse1.ansible1.com closed.
```

3.1.2.4 Create and Run an Ansible Playbook that Creates Multiple Directories in Each Linux Host

1- On your Ansible server, login as ansible, then create yaml file name "sample3.yml" in /home/ansible/Documents:

```
sudo vi sample3.yml
```

2- Paste in the following text:

```
---
```

```
- hosts: linuxservers
  become: true
  tasks:
    - name: Create multiple directories
      file: path={{item}} state=directory
      with_items:
        - '/home/ansible/folder1'
        - '/home/ansible/folder2'
        - '/home/ansible/folder3'
```

```
ansible@ansible1:~/Documents$ cat sample3.yml
---
- hosts: linuxservers
  become: true
  tasks:
    - name: Create multiple directories
      file: path={{item}} state=directory
      with_items:
        - '/home/ansible/folder1'
        - '/home/ansible/folder2'
        - '/home/ansible/folder3'

ansible@ansible1:~/Documents$
```

Quit text editor and save changes.

3- Run the sample3.yml ansible playbook by issuing the following command:

```
# ansible-playbook sample3.yml
```

```
[[ Home | GNS3-01 | fedora1 - 10.104.101.100 | ansible1 - 10.104.101.101 | ubuntu1a - 10.104.101.102 | opensuse1 - 10.104.101.103 | centos1 - 10.104.101.104 ]]

ansible@ansible1:~/Documents$ ansible-playbook sample3.yml

PLAY [linuxservers] ****
TASK [Gathering Facts] ****
ok: [ubuntu1a.ansible1.com]
ok: [centos1.ansible1.com]
ok: [opensuse1.ansible1.com]
ok: [fedora1.ansible1.com]

TASK [Create multiple directories] ****
changed: [ubuntu1a.ansible1.com] => (item=/home/ansible/folder1)
changed: [centos1.ansible1.com] => (item=/home/ansible/folder1)
changed: [opensuse1.ansible1.com] => (item=/home/ansible/folder1)
changed: [fedora1.ansible1.com] => (item=/home/ansible/folder1)
changed: [ubuntu1a.ansible1.com] => (item=/home/ansible/folder2)
changed: [ubuntu1a.ansible1.com] => (item=/home/ansible/folder3)
changed: [centos1.ansible1.com] => (item=/home/ansible/folder2)
changed: [opensuse1.ansible1.com] => (item=/home/ansible/folder2)
changed: [fedora1.ansible1.com] => (item=/home/ansible/folder2)
changed: [centos1.ansible1.com] => (item=/home/ansible/folder3)
changed: [opensuse1.ansible1.com] => (item=/home/ansible/folder3)
changed: [fedora1.ansible1.com] => (item=/home/ansible/folder3)

PLAY RECAP ****
centos1.ansible1.com      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
fedora1.ansible1.com      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
opensuse1.ansible1.com    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ubuntu1a.ansible1.com    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ansible@ansible1:~/Documents$
```

4- Check results

```
ansible@ansible1:~/Documents$ ssh opensuse1.ansible1.com
Have a lot of fun...
Last login: Sun Mar 16 22:03:32 2025 from 10.164.101.101
ansible@opensuse1:~> ls
bin Desktop folder1 folder2 folder3 test2.txt web
ansible@opensuse1:~> exit
logout
Connection to opensuse1.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh centos1.ansible1.com
Last login: Sun Mar 16 22:03:31 2025 from 10.164.101.101
ansible@centos1:~$ ls
Desktop folder1 folder2 folder3 test2.txt web
ansible@centos1:~$ exit
logout
Connection to centos1.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh fedora1.ansible1.com
Last login: Sun Mar 16 22:03:32 2025 from 10.164.101.101
ansible@fedora1:~$ ls
Desktop folder1 folder2 folder3 test2.txt test.txt web
ansible@fedora1:~$ exit
logout
Connection to fedora1.ansible1.com closed.
ansible@ansible1: ~/Documents$ ssh ubuntu1a.ansible1.com
```

```
ansible@ansible1:~/Documents$ ssh ubuntu1a.ansible1.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

8 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Sun Mar 16 22:03:31 2025 from 10.164.101.101
ansible@ubuntu1a:~$ ls
Desktop folder1 folder2 folder3 snap test2.txt web
ansible@ubuntu1a:~$ exit
logout
Connection to ubuntu1a.ansible1.com closed.
ansible@ansible1:~/Documents$ █
```

3.1.2.5 Create and Run an Ansible Playbook that Creates a New Users in Each Linux Host

References

Lesson 11 - Create users Playbook

<https://tutorialsonline.ca/courses/1330884/lectures/31594800>

- 1- To gather facts on all Linux hosts, go to your Ansible server, login as ansible, then enter the following command:

```
$ ansible all -m gather_facts --tree /tmp/facts
```

```
ansible@ansiblesrv4:~/Documents$ ansible all -m gather_facts --tree  
/tmp/facts  
<output omitted>
```

- 2- To view the facts for each host, enter the following command:

```
$ ls /tmp/facts/
```

```
ansible@ansible1:~/Documents$ ls /tmp/facts  
centos1.ansible1.com fedora1.ansible1.com opensuse1.ansible1.com ubuntu1.ansible1.com  
ansible@ansible1:~/Documents$
```

```
ansible@ansible1:~/Documents$ ls /tmp/facts
centos1.ansible1.com  federal.ansible1.com  opensuse1.ansible1.com
ubuntula.ansible1.com
ansible@ansible1:~/Documents$
```

3- To create a hash encrypted password, first install whois by issuing the following command:

```
sudo apt install whois -y
```

```
ansible@ansible1:~/Documents$ sudo apt install whois -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  whois
0 upgraded, 1 newly installed, 0 to remove and 15 not upgraded.
Need to get 51.7 kB of archives.
After this operation, 279 kB of additional disk space will be used.
Get:1 http://ca.archive.ubuntu.com/ubuntu noble/main amd64 whois amd64 5.5.22 [51.7 kB]
Fetched 51.7 kB in 0s (115 kB/s)
Selecting previously unselected package whois.
(Reading database ... 254658 files and directories currently installed.)
Preparing to unpack .../whois_5.5.22_amd64.deb ...
Unpacking whois (5.5.22) ...
Setting up whois (5.5.22) ...
Processing triggers for man-db (2.12.0-4build2) ...
ansible@ansible1:~/Documents$
```

4- Create encrypted password using SHA-512 by issuing the following command:

```
$ mkpasswd --method=sha-512
```

```
ansible@ansible1:~/Documents$ mkpasswd --method=sha-512
Password:
$6$5c5AKxB5yBhfj07x$0XwfS/ppfquDtscQR6qum55Xkcy9wFEW1zU741oOF6SR8SUSrXEEoAU6nEgWR10Z6hDJsoaMgE9HBPluhnsS0
ansible@ansible1:~/Documents$
```

```
ansible@ansible1:~/Documents$ mkpasswd --method=sha-512
Password:
$6$5c5AKxB5yBhfj07x$0XwfS/ppfquDtscQR6qum55Xkcy9wFEW1zU741oOF6SR8SUSrXEEoAU6nEgWR10Z6hDJsoaMgE9HBPluhnsS0
```

```
ansible@ansible1:~/Documents$
```

- 5- Next, create yaml file name "createnewusers.yml" in /home/ansible/Documents:

```
sudo vi createnewuser.yml
```

The screenshot shows a terminal window titled "createuser.yml *". The window displays the contents of a YAML configuration file. The file defines a single task named "Create New Users" for hosts "linuxservers". The task uses "become: true" and "gather_facts: false". It contains a "user" block with "name: esmith", "state: present", and a hashed password. Other parameters like "groups", "shell", "system", "createhome", and "home" are also specified. The terminal window has a dark background with light-colored text.

```
---  
- name: Create New Users  
  hosts: linuxservers  
  become: true  
  gather_facts: false  
  tasks:  
    - name: create Users tasks  
      user:  
        name: esmith  
        state: present  
        password: '$6$5c5AKxB5yBhfj07x$0XwfS/ppfquDtscQR6qum55Xkcy9wFEW1zU741oOF6SR8SUSrXEEoAU6nEgWR10Z6hDJsoaMgE9HBPltuhnssS0'  
        groups: ansible # Empty by default.  
        shell: /bin/bash # Defaults to /bin/bash  
        system: no # Defaults to no  
        createhome: yes # Defaults to yes  
        home: /home/esmith # Defaults to /home/<username>
```

- 6- Paste in the following text with the hashed password for the password value:

```
---  
- name: Create New Users  
  hosts: linuxservers  
  become: true  
  gather_facts: false  
  tasks:  
    - name: create Users tasks  
      user:  
        name: esmith  
        state: present  
        password:  
          '$6$5c5AKxB5yBhfj07x$0XwfS/ppfquDtscQR6qum55Xkcy9wFEW1zU741oOF6SR8SUSrXEEoAU6nEgWR10Z6hDJsoaMgE9HBPltuhnssS0'  
        groups: ansible # Empty by default.  
        shell: /bin/bash # Defaults to /bin/bash  
        system: no # Defaults to no  
        createhome: yes # Defaults to yes  
        home: /home/esmith # Defaults to /home/<username>
```

```

ansible@ansible1:~/Documents$ cat createuser.yml
---
- name: Create New Users
  hosts: linuxservers
  become: true
  gather_facts: false
  tasks:
    - name: create Users tasks
      user:
        name: esmith
        state: present
        password: '$6$5cSAKxB5yBhfj07x$0XwFS/ppfquDtscQR6qun55Xkcy9wFEM1zU741oOF6SR8SUSrXEEoAU6nEgWR10Z6hDjsoaMgE9HBPtuhnsS0'
        groups: ansible # Empty by default.
        shell: /bin/bash # Defaults to /bin/bash
        system: no # Defaults to no
        createhome: yes # Defaults to yes
        home: /home/esmith # Defaults to /home/<username>
ansible@ansible1:~/Documents$
```

7- Run the createnewuser.yml ansible playbook by issuing the following command:

```
# ansible-playbook createnewuser.yml
```

```

ansible@ansible1:~/Documents$ ansible-playbook createuser.yml

PLAY [Create New Users] ****
TASK [create Users tasks] ****
changed: [ubuntula.ansible1.com]
changed: [opensuse1.ansible1.com]
changed: [centosi.ansible1.com]
changed: [fedora1.ansible1.com]

PLAY RECAP ****
centosi.ansible1.com : ok=1    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
fedora1.ansible1.com : ok=1    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
opensuse1.ansible1.com : ok=1    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ubuntula.ansible1.com : ok=1    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ansible@ansible1:~/Documents$
```

8. Verify results

```
ansible@ansible1:~/Documents$ ssh fedora1.ansible1.com
Last login: Sun Mar 16 22:35:13 2025 from 10.164.101.101
ansible@fedora1:~$ ls
Desktop folder1 folder2 folder3 test2.txt test.txt web
ansible@fedora1:~$ cd /home/
ansible@fedora1:/home$ ls
ansible esmith student
ansible@fedora1:/home$ exit
logout
Connection to fedora1.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh centos1.ansible1.com
Last login: Sun Mar 16 22:35:12 2025 from 10.164.101.101
ansible@centos1:~$ cd /home/
ansible@centos1:/home$ ls
ansible esmith student
ansible@centos1:/home$ exit
logout
Connection to centos1.ansible1.com closed.
```

```
ansible@ansible1:~/Documents$ ssh ubuntu1a.ansible1.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

8 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Sun Mar 16 22:35:13 2025 from 10.164.101.101
ansible@ubuntu1a:~$ cd /home/
ansible@ubuntu1a:/home$ ls
ansible esmith student
ansible@ubuntu1a:/home$ exit
logout
Connection to ubuntu1a.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh opensuse1.ansible1.com
Have a lot of fun...
Last login: Sun Mar 16 22:35:12 2025 from 10.164.101.101
ansible@opensuse1:~$ cd /home/
ansible@opensuse1:/home$ ls
ansible esmith student
ansible@opensuse1:/home$ exit
logout
Connection to opensuse1.ansible1.com closed.
```

3.1.2.6 Ansible Playbook to Install vsftpd Package

1. Here's the playbook named **ftp-install.yml**.

```
---
```

- name: Install ftp
hosts: linuxservers
become: true


```
tasks:
```

- name: Install vsftpd package in CentOS and Fedora
dnf:
 - name: vsftpd
 - state: latest

```
when: ansible_facts['os_family'] == "RedHat"
```

- name: Install vsftpd package in openSUSE
community.general.zypper:
 - name: vsftpd
 - state: present

```
when: ansible_facts['os_family'] == "Suse"
```

- name: Install vsftpd package in Ubuntu
apt:
 - name: vsftpd
 - state: latest

```
when: ansible_facts['os_family'] == "Debian"
```

- name: Start and Enable vsftpd service
systemd:
 - name: vsftpd
 - state: restarted
 - enabled: yes

- name: Store the status of service into a variable
shell: "systemctl status vsftpd"
register: status

- name: Print out the status service
debug:
 - var: status

- name: Show message # Another option of showing the status of
service. This is for future reference.
debug:

```
msg: "{{ status.stdout }}"
```

How script Works

Gathering Facts: Ansible runs an initial task to collect facts about each host (e.g., OS family, distribution), which are stored in `ansible_facts`.

Conditional Execution: The `when` clauses ensure each package manager task runs only on the appropriate OS family:

- RedHat for CentOS and Fedora.
- Suse for openSUSE.
- Debian for Ubuntu.

Package Installation: The relevant task installs `vsftpd` using the host's package manager.

Service Management: The `systemd` task starts and enables the service.

Status Verification: The shell task captures the service status, and debug tasks display it in two formats for flexibility.

2. Verify the syntax of your playbook.

```
ansible@ansible1:~/Documents$ ansible-playbook ~/Documents/ftp-install.yml --syntax-check
playbook: /home/ansible/Documents/ftp-install.yml
```

No error with playbook syntax.

3. Execute the playbook.

Errors are found for Fedora

```
ansible@ansible1:~/Documents$ ansible-playbook ~/Documents/ftp-install.yml
PLAY [Install ftp] ****
TASK [Gathering Facts]
ok: [centos1 ansible1.com]
ok: [ubuntu1 ansible1.com]
ok: [opensuse1 ansible1.com]
ok: [fedora1 ansible1.com]

TASK [Install vsftpd package in CentOS and Fedora] ****
skipping: [opensuse1 ansible1.com]
skipping: [ubuntu1 ansible1.com]
fatal: [fedora1 ansible1.com]: FAILED! => {"changed": false, "failure": true, "msg": "Could not report the libtbsm python module (3.5.2 (win32, Feb 8 2015, 00:00:00) [GCC 4.8.2] 20150208 (Red Hat 4.8.2-17)) . Please install python-lttbtsm package or ensure you have specified the correct ansible_python_interpreter. (attempted '/usr/bin/python3', '/usr/bin/python3.5', '/usr/bin/python')", "unreachable": true}
changed: [centos1 ansible1.com]

TASK [Install vsftpd package in openSUSE] ****
skipping: [centos1 ansible1.com]
skipping: [ubuntu1 ansible1.com]
changed: [opensuse1 ansible1.com]

TASK [Install vsftpd package in Ubuntu] ****
skipping: [centos1 ansible1.com]
skipping: [opensuse1 ansible1.com]
changed: [ubuntu1 ansible1.com]
```

```

TASK [Install vsftpd package in Ubuntu] ****
skipping: [centos1.ansible.com]
skipping: [openSUSE1.ansible.com]
changed: [ubuntu.ansible.com]

TASK [Start and enable vsftpd service] ****
changed: [ubuntu.ansible.com]
changed: [centos1.ansible.com]
changed: [opensuse1.ansible.com]

TASK [Store the status of service into a variable] ****
changed: [ubuntu.ansible.com]
changed: [centos1.ansible.com]
changed: [openSUSE1.ansible.com]

TASK [Print out the status service] ****
ok: [centos1.ansible.com] => {
    "status": {
        "changed": true,
        "cmd": "systemctl status vsftpd",
        "exit_status": 0,
        "start": "2025-03-16 23:11:30.954997",
        "failed": false,
        "msg": "",
        "rc": 0,
        "start": "2025-03-16 23:11:18.930548",
        "state": "running",
        "stderr_lines": [],
        "stdout": "● vsftpd.service - Vsftpd Ftp daemon\n   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)\n      Active: active (running) since Sun 2025-03-16 23:11:17 EDT; 999ms ago\n        Docs: man:vsftpd(8)\n    Main PID: 12857 (vsftpd)\n      Tasks: 1 (limit: 2293)\n     Memory: 752K (peak: 10K)\n        CPU: 0ms\n       CGroup: /system.slice/vsftpd.service\n              └─12857 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf\n\nMar 16 23:11:17 centos1 systemd[1]: Starting vsftpd.service - Vsftpd Ftp daemon...\nMar 16 23:11:17 centos1 systemd[1]: Started vsftpd.service - Vsftpd Ftp daemon.\n"
        "stdout_lines": []
    }
}

```

Centos

```

TASK [Print out the status service] ****
ok: [centos1.ansible.com] => {
    "status": {
        "changed": true,
        "cmd": "systemctl status vsftpd",
        "exit_status": 0,
        "start": "2025-03-16 23:11:18.930548",
        "failed": false,
        "msg": "",
        "rc": 0,
        "start": "2025-03-16 23:11:18.930548",
        "state": "running",
        "stderr_lines": [],
        "stdout": "● vsftpd.service - Vsftpd Ftp daemon\n   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)\n      Active: active (running) since Sun 2025-03-16 23:11:17 EDT; 999ms ago\n        Docs: man:vsftpd(8)\n    Main PID: 12857 (vsftpd)\n      Tasks: 1 (limit: 2293)\n     Memory: 752K (peak: 10K)\n        CPU: 0ms\n       CGroup: /system.slice/vsftpd.service\n              └─12857 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf\n\nMar 16 23:11:17 centos1 systemd[1]: Starting vsftpd.service - Vsftpd Ftp daemon...\nMar 16 23:11:17 centos1 systemd[1]: Started vsftpd.service - Vsftpd Ftp daemon.\n"
        "stdout_lines": []
    }
}

```

```

    "Mar 16 23:11:17 centos1 systemd[1]: Starting vsftpd.service - Vsftpd Ftp daemon...",
    "Mar 16 23:11:17 centos1 systemd[1]: Started vsftpd.service - Vsftpd Ftp daemon."
]
}

```

```

ski [openSUSE.ansible.com] => [
  status: {
    "changed": true,
    "cmd": "systemctl status vsftpd",
    "delta": "0:00:00 01:41:56",
    "end": "2025-03-16 23:11:19.088487",
    "failed": false,
    "msg": "",
    "rc": 0,
    "start": "2025-03-16 23:11:18.986262",
    "stderr": [],
    "stdout": [
      "● vsftpd.service - Vsftpd ftp daemon    Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)\n      Active: active (running) since Sat 2025-03-16 23:11:18.941ms ago\n        Main PID: 25294 (vsftpd)\n          Tasks: 1 (limit: 4585)\n            CPU: 42ms\n            CGroup: /system.slice/vsftpd.service\n\n           vsftpd (pid 25294)\n              \u2192 /etc/vsftpd.conf[mar 16 23:11:18 openSUSE systemd[1]: Started vsftpd Ftp daemon.]"
    ],
    "stdout_lines": [
      "\u2192 /etc/vsftpd.conf[mar 16 23:11:18 openSUSE systemd[1]: Started vsftpd Ftp daemon.]"
    ]
  }
]

```

```

ski [ubuntu.ansible.com] => [
  status: {
    "changed": true,
    "cmd": "systemctl status vsftpd",
    "delta": "0:00:00 02:20:15",
    "end": "2025-03-16 23:11:18.766841",
    "failed": false,
    "msg": "",
    "rc": 0,
    "start": "2025-03-16 23:11:18.764826",
    "stderr": [],
    "stdout": [
      "● vsftpd.service - vsftpd FTP server\n    Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: enabled)\n      Active: active (running) since Sun 2025-03-16 23:11:17 UTC; 3s ago\n        Process: 10355 execStartPre/bin/mdir -p /var/run/vsftpd/empty (code-exited, status=0/SUCCESS)\n        Main PID: 10355 (vsftpd)\n          Tasks: 1 (limit: 4581)\n            Memory: 716.0K (peak: 1.0M)\n            CPU: 9ms\n            CGroup: /system.slice/vsftpd.service\n\n           vsftpd (pid 10355)\n              \u2192 /etc/vsftpd.conf[mar 16 23:11:17 ubuntu systemd[1]: Started vsftpd.service - vsftpd FTP server.]"
    ],
    "stdout_lines": [
      "\u2192 /etc/vsftpd.conf[mar 16 23:11:17 ubuntu systemd[1]: Started vsftpd.service - vsftpd FTP server.]"
    ]
  }
]

```

```

TASK [Show message] ****
ski [centos.ansible.com] => [
  msg: "\u2192 /etc/vsftpd.conf[mar 16 23:11:17 openSUSE systemd[1]: Starting vsftpd.service - Vsftpd ftp daemon...]"
]
ski [centos.ansible.com] => [
  msg: "\u2192 /etc/vsftpd.conf[mar 16 23:11:17 openSUSE systemd[1]: Started vsftpd.service - Vsftpd ftp daemon...]"
]
ski [ubuntu.ansible.com] => [
  msg: "\u2192 /etc/vsftpd.conf[mar 16 23:11:17 openSUSE systemd[1]: Starting vsftpd.service - vsftpd FTP server...]"
]
ski [ubuntu.ansible.com] => [
  msg: "\u2192 /etc/vsftpd.conf[mar 16 23:11:17 openSUSE systemd[1]: Started vsftpd.service - vsftpd FTP server...]"
]
ski [ubuntu.ansible.com] => [
  msg: "\u2192 /etc/vsftpd.conf[mar 16 23:11:18 openSUSE systemd[1]: Starting vsftpd.service - Vsftpd ftp daemon...]"
]
ski [ubuntu.ansible.com] => [
  msg: "\u2192 /etc/vsftpd.conf[mar 16 23:11:18 openSUSE systemd[1]: Started vsftpd.service - Vsftpd ftp daemon...]"
]

PLAY RECAP ****
centos.ansible.com : ok=6   changed=3   unreachable=0   failed=0   skipped=2   rescued=0   ignored=8
fedora.ansible.com  : ok=6   changed=3   unreachable=0   failed=1   skipped=8   rescued=0   ignored=8
openSUSE.ansible.com: ok=6   changed=3   unreachable=0   failed=0   skipped=0   rescued=0   ignored=8
ubuntu.ansible.com : ok=6   changed=3   unreachable=0   failed=0   skipped=2   rescued=0   ignored=8

```

Error with Fedora install

```
fatal: [fedora1.ansible1.com]: FAILED! => {"changed": false, "failures": [], "msg": "Could not import the libdnf5 python module using /usr/bin/python3 (3.13.2 (main, Feb 4 2025, 00:00:00) [GCC 14.2.1 20250110 (Red Hat 14.2.1-7)]). Please install python3-libdnf5 package or ensure you have specified the correct ansible_python_interpreter. (attempted ['/usr/libexec/platform-python', '/usr/bin/python3', '/usr/bin/python2', '/usr/bin/python'])"}
```

The error you're encountering in the `ftp-install.yml` playbook on `fedora1.ansible1.com` indicates that Ansible cannot execute the `dnf` module because the required `python3-libdnf5` package is missing on the Fedora host.

Fedora 41 (or later, based on the Python 3.13.2 version) uses `libdnf5` (a newer version of the DNF library) as part of its package management system.

The `python3-libdnf5` package provides the Python bindings that Ansible's `dnf` module needs to communicate with `dnf`.

On `centos1.ansible1.com`, this task succeeded because CentOS likely uses an older version of DNF (`libdnf`, not `libdnf5`), and the required bindings (`python3-dnf` or similar) are already installed.

Manual Installation

1. SSH into `fedora1`:

```
ssh fedora1.ansible1.com
```

2. Install `python3-libdnf5`:

```
sudo dnf install python3-libdnf5
```



```
Last login: Sun Mar 16 23:10:48 2025 from 10.164.101.101
[ansible@fedora1: ~]$ sudo dnf install python3-libdnf5
Updating and loading repositories...
Fedora 41 - x86_64
 0: https://dl.fedoraproject.org/pub/fedora/linux/releases/41/Everything/x86_64/os/Packages/
[...]
[  0.0%] 0B/s | 0.0  MiB/s | 0.0  B/s | 0.000s
[  0.0%] 0B/s | 0.0  MiB/s | 0.0  B/s | 0.000s
```

3. Verify the Installation:

```
python3 -c "import libdnf5"
```

If this command runs without errors, the module is installed correctly.



```
Complete!
[ansible@fedora1: ~]$ python3 -c "import libdnf5"
[ansible@fedora1: ~]$
```

4. Exit the SSH Session:

```
exit
```

5. Re-run the Playbook:

```
ansible-playbook ~/Documents/ftp-install.yml -v
```

Re run is done skipping the distros already installed but Fedora

```
ansible@ansible:~/Documents/ftp-install.yml
PLAY [Install Ftp]
...
TASK [Gathering Facts]
ok: [ubuntu ansible.com]
ok: [centos ansible.com]
ok: [openSUSE ansible.com]
ok: [fedora ansible.com]

TASK [Install vsftpd package in CentOS and Fedora]
skipping: [ubuntu ansible.com]
skipping: [openSUSE ansible.com]
ok: [centos ansible.com]
changed: [fedora ansible.com]

TASK [Install vsftpd package in openSUSE]
skipping: [centos ansible.com]
skipping: [fedora ansible.com]
skipping: [ubuntu ansible.com]
ok: [openSUSE ansible.com]

TASK [Install vsftpd package in Ubuntu]
skipping: [centos ansible.com]
skipping: [fedora ansible.com]
skipping: [openSUSE ansible.com]
ok: [ubuntu ansible.com]
```

```
[root@centos ~]# systemctl --no-pager -l vsftpd
● vsftpd.service - vsftpd ftp deamon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)
   Active: active (running) since Fri Mar 16 23:36:27 2018; 2s ago
     Tasks: 1 (limit: 22917)
      CPU: 1.200us
     Memory: 748K (peak: 1.25M)
      CGroup: /system.slice/vsftpd.service
           └─13374 /usr/sbin/vsftpd -c /etc/vsftpd/vsftpd.conf

Mar 16 23:36:27 centos[1]: Starting vsftpd service - vsftpd ftp deamon...
Mar 16 23:36:27 centos[1]: started vsftpd service.

```

```
ski:[abruno@archetek.com] ~> [status]
{
  "status": {
    "changed": true,
    "cmd": "systemctl status vsftpd",
    "delta": "0:00:00.015230",
    "end": "2025-03-16 23:36:26.82964",
    "failed": false,
    "mag": "+",
    "ret": 0,
    "start": "2025-03-16 23:36:26(26.824606)",
    "stder": "",
    "stder_lines": [],
    "stdout": [
      "● vsftpd.service - vsftpd FTP server\n    Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: enabled)\n      Active: active (running) since Su\nr 2025-03-16 23:36:24 EDT; 2s ago\n        Process: 18821 ExecStartPre=/bin/mkdir -p /var/run/vsftpd/empty (code=exited, status=0/SUCCESS)\n        Main PID: 18824 (vsftpd)\n          Tasks: 1 (limit: 4555)\n            Memory: 115.0M (peak: 1.0M)\n              CPU: 9ms\n            CGroup: /system.slice/vsftpd.service\n                └─18824 /usr/sbin/vsftpd /etc/vsftpd.conf"
    ],
    "stdout_lines": [
      "● vsftpd.service - vsftpd FTP server\n    Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: enabled)\n      Active: active (running) since Sun 2025-03-16 23:36:24 EDT; 2s ago\n        Process: 18821 ExecStartPre=/bin/mkdir -p /var/run/vsftpd/empty (code=exited, status=0/SUCCESS)\n        Main PID: 18824 (vsftpd)\n          Tasks: 1 (limit: 4555)\n            Memory: 115.0M (peak: 1.0M)\n              CPU: 9ms\n            CGroup: /system.slice/vsftpd.service\n                └─18824 /usr/sbin/vsftpd /etc/vsftpd.conf"
    ],
    "User": "root"
  }
}
Mar 16 23:36:24 ubuntu10 systemd[1]: Starting vsftpd service - vsftpd FTP server...
Mar 16 23:36:24 ubuntu10 systemd[1]: Started vsftpd.service - vsftpd FTP server.
```

```
ski:[opensesame:anshul1.com] ~> [status]
{
  "status": {
    "changed": true,
    "cmd": "systemctl status vsftpd",
    "delta": "0:00:00.014482",
    "end": "2025-03-16 23:36:27.106138",
    "failed": false,
    "mag": "+",
    "ret": 0,
    "start": "2025-03-16 23:36:27.01879",
    "stder": "",
    "stder_lines": [],
    "stdout": [
      "● vsftpd.service - Vsftpd ftp daemon\n    Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)\n      Active: active (running) since Sun 2025-03-16 23:36:27.01879\n        Main PID: 26299 (vsftpd)\n          Tasks: 1 (limit: 4555)\n            CPU: 38ms\n            CGroup: /system.slice/vsftpd.service\n                └─26299 /usr/sbin/vsftpd /etc/vsftpd.conf"
    ],
    "stdout_lines": [
      "● vsftpd.service - Vsftpd ftp daemon\n    Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)\n      Active: active (running) since Sun 2025-03-16 23:36:27.01879\n        Main PID: 26299 (vsftpd)\n          Tasks: 1 (limit: 4555)\n            CPU: 38ms\n            CGroup: /system.slice/vsftpd.service\n                └─26299 /usr/sbin/vsftpd /etc/vsftpd.conf"
    ],
    "User": "root"
  }
}
Mar 16 23:36:27 opensesame systemd[1]: Started Vsftpd ftp daemon.
```

```
ski:[decentralendecentral.com] ~> [status]
{
  "status": {
    "changed": true,
    "cmd": "systemctl status vsftpd",
    "delta": "0:00:00.020595",
    "end": "2025-03-16 23:36:27.380364",
    "failed": false,
    "mag": "+",
    "ret": 0,
    "start": "2025-03-16 23:36:27.293455",
    "stder": "",
    "stder_lines": [],
    "stdout": [
      "● vsftpd.service - Vsftpd ftp daemon\n    Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)\n      Drop-In: /usr/lib/systemd/system/vsftpd.service.d\n                  └─00-keep-warm.conf\n      Active: active (running) since Sun 2025-03-16 23:36:26 EDT; 1s ago\n        Invocation: systemctl start vsftpd\n        Main PID: 67601 (vsftpd)\n          Tasks: 1 (limit: 4537)\n            Memory: 748\nK (peak: 1.38M)\n              CPU: 7ms\n            CGroup: /system.slice/vsftpd.service\n                └─67601 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf\nMar 16 23:36:26 decentral systemd[1]: Starting vsftpd.service - Vsftpd ftp daemon."
    ],
    "stdout_lines": [
      "● vsftpd.service - Vsftpd ftp daemon\n    Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)\n      Drop-In: /usr/lib/systemd/system/vsftpd.service.d\n                  └─00-keep-warm.conf\n      Active: active (running) since Sun 2025-03-16 23:36:26 EDT; 1s ago\n        Invocation: systemctl start vsftpd\n        Main PID: 67601 (vsftpd)\n          Tasks: 1 (limit: 4537)\n            Memory: 748\nK (peak: 1.38M)\n              CPU: 7ms\n            CGroup: /system.slice/vsftpd.service\n                └─67601 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf"
    ],
    "User": "root"
  }
}
Mar 16 23:36:26 decentral systemd[1]: Started vsftpd.service - Vsftpd ftp daemon.
```

```

TASK [Show message] *****
ok: [centos.ansible1.com] => [
  "msg": "\u25bc vsftpd.service - Vsftpd ftp daemon\n  Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)\n    Active: active (running) since Sun 2025-03-16 23:36:25 EDT; 9min ago\n      Process: 13370 ExecStart=/usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf (code=exited, status=0/SUCCESS)\n      Main PID: 13374 (vsftpd)\n        Tasks: 1 (limit: 4587)\n       Memory: 748K (peak: 1.3M)\n      CPU: 8ms\n         CPU: /system.slice/vsftpd.service\n          └─13374 /usr/sbin/vsftpd\n\nvsftpd.service - Vsftpd ftp daemon...\n  Started vsftpd.service - Vsftpd ftp daemon.\n"
]
ok: [federal.ansible1.com] => [
  "msg": "\u25bc vsftpd.service - Vsftpd ftp daemon\n  Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)\n    Active: active (running) since Sun 2025-03-16 23:36:25 EDT; 9min ago\n      Process: 67600 ExecStart=/usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf (code=exited, status=0/SUCCESS)\n      Main PID: 67601 (vsftpd)\n        Tasks: 1 (limit: 4587)\n       Memory: 748K (peak: 1.3M)\n      CPU: 8ms\n         CPU: /system.slice/vsftpd.service\n          └─67601 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf\n\nvsftpd.service - Vsftpd ftp daemon...\n  Started vsftpd.service - Vsftpd ftp daemon.\n"
]
ok: [ubuntu.ansible1.com] => [
  "msg": "\u25bc vsftpd.service - Vsftpd FTP server\n  Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: enabled)\n    Active: active (running) since Sun 2025-03-16 23:36:24 EDT; 9min ago\n      Process: 10821 ExecStartPre=/bin/mkdir -p /var/run/vsftpd/empty (code=exited, status=0/SUCCESS)\n      Main PID: 10824 (vsftpd)\n        Tasks: 1 (limit: 4587)\n       Memory: 716.0K (peak: 1.1M)\n      CPU: 9ms\n         CPU: /system.slice/vsftpd.service\n          └─10824 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf\n\nvsftpd.service - Vsftpd FTP server...\n  Started vsftpd.service - Vsftpd FTP server.\n"
]
ok: [opensuse.ansible1.com] => [
  "msg": "\u25bc vsftpd.service - Vsftpd ftp daemon\n  Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)\n    Active: active (running) since Sun 2025-03-16 23:36:24 EDT; 9min ago\n      Process: 10821 ExecStartPre=/bin/mkdir -p /var/run/vsftpd/empty (code=exited, status=0/SUCCESS)\n      Main PID: 10824 (vsftpd)\n        Tasks: 1 (limit: 4587)\n       Memory: 716.0K (peak: 1.1M)\n      CPU: 9ms\n         CPU: /system.slice/vsftpd.service\n          └─10824 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf\n\nvsftpd.service - Vsftpd ftp daemon...\n  Started vsftpd.service - Vsftpd ftp daemon.\n"
]

```

```

PLAY RECAP *****
centos.ansible1.com : ok=6   changed=2   unreachable=0   failed=0   skipped=2   rescued=0   ignored=0
federal.ansible1.com : ok=6   changed=2   unreachable=0   failed=0   skipped=2   rescued=0   ignored=0
opensuse.ansible1.com : ok=6   changed=2   unreachable=0   failed=0   skipped=2   rescued=0   ignored=0
ubuntu.ansible1.com  : ok=6   changed=2   unreachable=0   failed=0   skipped=2   rescued=0   ignored=0

```

4. Currently we only have four machines as our ansible hosts, we can also verify the result from each machines.

```

[unknown]:~$ sudo vsftpd -v
ansible@federal:~$ systemctl status vsftpd
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/vsftpd.service.d
     └─10-timeout-abort.conf, 50-keep-warm.conf
     Active: active (running) since Sun 2025-03-16 23:36:26 EDT; 9min ago
   Invocation: 83aef3f56c9043109c62ed5579787bd
     Process: 67600 ExecStart=/usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf (code=exited, status=0/SUCCESS)
   Main PID: 67601 (vsftpd)
     Tasks: 1 (limit: 4587)
    Memory: 748K (peak: 1.3M)
      CPU: 7ms
     CGroup: /system.slice/vsftpd.service
           └─67601 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

```

```

ansible@ubuntu1a:~$ systemctl status vsftpd.service
● vsftpd.service - vsftpd FTP server
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: enabled)
   Active: active (running) since Sun 2025-03-16 23:36:24 EDT; 11min ago
     Process: 10821 ExecStartPre=/bin/mkdir -p /var/run/vsftpd/empty (code=exited, status=0/SUCCESS)
   Main PID: 10824 (vsftpd)
     Tasks: 1 (limit: 4587)
    Memory: 716.0K (peak: 1.1M)
      CPU: 9ms
     CGroup: /system.slice/vsftpd.service
           └─10824 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

```

```

connection to ansible1.ansible1.com closed.
ansible@ansible1:~/Documents$ ssh centos1.ansible1.com
Last login: Sun Mar 16 23:36:26 2025 from 10.164.101.101
ansible@centos1:~$ systemctl status vsftpd.service
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)
   Active: active (running) since Sun 2025-03-16 23:36:25 EDT; 11min ago
     Invocation: ba258decad2b48889b9cda2e6df96b3c
      Process: 13373 ExecStart=/usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf (code=exited, status=0/SUCCESS)
    Main PID: 13374 (vsftpd)
       Tasks: 1 (limit: 22917)
      Memory: 748K (peak: 1.2M)
        CPU: 8ms
       CGroup: /system.slice/vsftpd.service
               └─13374 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

ansible@centos1:~$
```

```

ansible@opensuse1:~> systemctl status vsftpd.service
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; preset: disabled)
   Active: active (running) since Sun 2025-03-16 23:36:24 EDT; 12min ago
     Main PID: 26299 (vsftpd)
        Tasks: 1 (limit: 4585)
       CPU: 38ms
      CGroup: /system.slice/vsftpd.service
              └─26299 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

ansible@opensuse1:~>
```

3.1.2.7 Ansible Playbook to Install Webmin

1. Here's the playbook for the installation of webmin.

```

---
- name: Install Webmin on Fedora, CentOS, openSUSE, and Ubuntu
hosts: linuxservers
become: true
gather_facts: yes

tasks:
- name: Ensure required packages are installed on all hosts
  ansible.builtin.package:
    name:
      - perl
      - openssl
    state: present
  when: ansible_facts['os_family'] in ["RedHat", "Suse", "Debian"]

# RedHat-based systems (Fedora, CentOS)
- name: Download Webmin RPM for RedHat-based systems
  ansible.builtin.get_url:
    url: https://prdownloads.sourceforge.net/webadmin/webmin-2.111-1.noarch.rpm
    dest: /tmp/webmin-2.111-1.noarch.rpm
    mode: '0644'
  when: ansible_facts['os_family'] == "RedHat"

- name: Import Webmin GPG key on RedHat-based systems
  ansible.builtin.rpm_key:
    state: present
    key: https://www.webmin.com/jcameron-key.asc
  when: ansible_facts['os_family'] == "RedHat"
  register: rpm_key_result
  failed_when: rpm_key_result.failed
```

```

retries: 3
delay: 5
until: rpm_key_result is success

- name: Install Webmin on RedHat-based systems using dnf/yum
  ansible.builtin.dnf:
    name: /tmp/webmin-2.111-1.noarch.rpm
    state: present
  when: ansible_facts['os_family'] == "RedHat"
  register: dnf_result
  failed_when: dnf_result.failed
  retries: 3
  delay: 5
  until: dnf_result is success
  notify: Restart Webmin service

# openSUSE
- name: Download Webmin RPM for openSUSE
  ansible.builtin.get_url:
    url: https://prdownloads.sourceforge.net/webadmin/webmin-2.111-1.noarch.rpm
    dest: /tmp/webmin-2.111-1.noarch.rpm
    mode: '0644'
  when: ansible_facts['os_family'] == "Suse"

- name: Import Webmin GPG key on openSUSE
  ansible.builtin.rpm_key:
    state: present
    key: https://www.webmin.com/jcameron-key.asc
  when: ansible_facts['os_family'] == "Suse"
  register: rpm_key_result
  failed_when: rpm_key_result.failed
  retries: 3
  delay: 5
  until: rpm_key_result is success

- name: Install Webmin on openSUSE using zypper
  community.general.zypper:
    name: /tmp/webmin-2.111-1.noarch.rpm
    state: present
  when: ansible_facts['os_family'] == "Suse"
  register: zypper_result
  failed_when: zypper_result.failed
  retries: 3
  delay: 5
  until: zypper_result is success

# Ubuntu
- name: Add Webmin repository for Ubuntu
  ansible.builtin.apt_repository:
    repo: 'deb http://download.webmin.com/download/repository sarge contrib'
    state: present
    filename: webmin
  when: ansible_facts['os_family'] == "Debian"
  register: apt_repo_result

- name: Install Webmin GPG key on Ubuntu
  ansible.builtin.apt_key:
    url: http://www.webmin.com/jcameron-key.asc
    state: present
  when: ansible_facts['os_family'] == "Debian"
  register: apt_key_result
  failed_when: apt_key_result.failed

- name: Update APT cache on Ubuntu
  ansible.builtin.apt:
    update_cache: yes
  when: ansible_facts['os_family'] == "Debian"
  register: apt_update_result
  failed_when: apt_update_result.failed
  retries: 3
  delay: 5
  until: apt_update_result is success

- name: Install Webmin on Ubuntu
  ansible.builtin.apt:
    name: webmin
    state: latest
    update_cache: "{{ apt_repo_result.changed or apt_key_result.changed }}"
    install_recommends: no

```

```

force: yes
dpkg_options: 'force-confdef,force-confold'
when: ansible_facts['os_family'] == "Debian"
register: apt_install_result
failed_when: apt_install_result.failed
retries: 3
delay: 5
until: apt_install_result is success
notify: Restart Webmin service

# Service Management
- name: Start and enable Webmin service on all hosts
  ansible.builtin.systemd:
    name: webmin
    state: restarted
    enabled: yes
  register: service_result

- name: Print Webmin service status
  ansible.builtin.debug:
    var: service_result

# Cleanup
- name: Cleanup temporary files on RedHat
  ansible.builtin.file:
    path: "{{ item }}"
    state: absent
  loop:
    - /tmp/setup-repos.sh
    - /tmp/webmin-2.111-1.noarch.rpm
  when: ansible_facts['os_family'] == "RedHat"

- name: Cleanup temporary files on openSUSE
  ansible.builtin.file:
    path: /tmp/webmin-2.111-1.noarch.rpm
    state: absent
  when: ansible_facts['os_family'] == "Suse"

handlers:
- name: Restart Webmin service
  ansible.builtin.systemd:
    name: webmin
    state: restarted
  when: ansible_facts['os_family'] in ["RedHat", "Suse", "Debian"]

```

2. Verify the syntax of the playbook.

```

ansible@ansubu16:~/Documents$ ansible-playbook webmin-install.yml --syntax-check
playbook: webmin-install.yml

```

Playbook syntax has been verified without error.

3. Execute the playbook.

4. Check webmin is installed

```

Max kernel policy version: 33
root@federal1:~# systemctl status webmin.service
● webmin.service - Webmin server daemon
   Loaded: loaded (/usr/lib/systemd/system/webmin.service; enabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/service.d
     └─no-timeout-abort.conf, 50-keep-warm.conf
     Active: active (running) since Mon 2025-03-17 01:46:54 EDT; 20s ago
       Process: 162535 ExecStart=/usr/libexec/webmin/miniserv.pl /etc/webmin/miniserv.conf (code=exited, status=0/SUCCESS)
      Main PID: 162537 (miniserv.pl)
        Tasks: 1 (limit: 4587)
       Memory: 44M (peak: 75.6M)
          CPU: 1.1eas
         CGroup: /system.slice/webmin.service
             └─162537 /usr/bin/perl /usr/libexec/webmin/miniserv.pl /etc/webmin/miniserv.conf

Mar 17 01:46:53 federal1 systemd[1]: Starting webmin.service - Webmin server daemon...
Mar 17 01:46:53 federal1 webmin[162535]: Webmin starting
Mar 17 01:46:54 federal1 systemd[1]: webmin.service: Can't open PID file /var/webmin/miniserv.pid (yet?) after start: No such file or directory
Mar 17 01:46:54 federal1 systemd[1]: Started webmin.service - Webmin server daemon.
root@federal1:~#

```

```

ansible@ubuntu1a:~/Desktop$ systemctl status webmin.service
● webmin.service - Webmin server daemon
   Loaded: loaded (/usr/lib/systemd/system/webmin.service; enabled; preset: enabled)
   Active: active (running) since Mon 2025-03-17 01:46:48 EDT; 1min 17s ago
     Process: 25307 ExecStart=/usr/share/webmin/miniserv.pl /etc/webmin/miniserv.conf (code=exited, status=0/SUCCESS)
    Main PID: 25308 (miniserv.pl)
      Tasks: 1 (limit: 4551)
     Memory: 26.8M (peak: 53.4M)
        CPU: 683ms
       CGroup: /system.slice/webmin.service
           └─25308 /usr/bin/perl /usr/share/webmin/miniserv.pl /etc/webmin/miniserv.conf

ansible@ubuntu1a:~/Desktop$ 

```

```

opensuse1:~# systemctl status webmin.service
● webmin.service - Webmin server daemon
   Loaded: loaded (/usr/lib/systemd/system/webmin.service; enabled; preset: disabled)
   Active: active (running) since Mon 2025-03-17 01:46:46 EDT; 48s ago
     Process: 39262 ExecStart=/usr/libexec/webmin/miniserv.pl /etc/webmin/miniserv.conf (code=exited, status=0/SUCCESS)
    Main PID: 39263 (miniserv.pl)
      Tasks: 1 (limit: 4585)
        CPU: 516ms
       CGroup: /system.slice/webmin.service
           └─39263 /usr/bin/perl /usr/libexec/webmin/miniserv.pl /etc/webmin/miniserv.conf

Mar 17 01:46:45 opensuse1 systemd[1]: Starting Webmin server daemon...
Mar 17 01:46:46 opensuse1 webmin[39262]: Webmin starting
Mar 17 01:46:46 opensuse1 systemd[1]: Started Webmin server daemon.
opensuse1:~# 

```

```

ansible@centos1:~/ $ systemctl status webmin.service
● webmin.service - Webmin server daemon
   Loaded: loaded (/usr/lib/systemd/system/webmin.service; enabled; preset: disabled)
   Active: active (running) since Mon 2025-03-17 01:35:51 EDT; 13min ago
     Invocation: 57f12573e1f646138823d239200e4eca
   Process: 22020 ExecStart=/usr/libexec/webmin/miniserv.pl /etc/webmin/miniserv.conf (code=exited, status=0/SUCCESS)
    Main PID: 22021 (miniserv.pl)
      Tasks: 1 (limit: 22917)
     Memory: 26.7M (peak: 56.5M)
        CPU: 2.434s
       CGroup: /system.slice/webmin.service
           └─22021 /usr/bin/perl /usr/libexec/webmin/miniserv.pl /etc/webmin/miniserv.conf

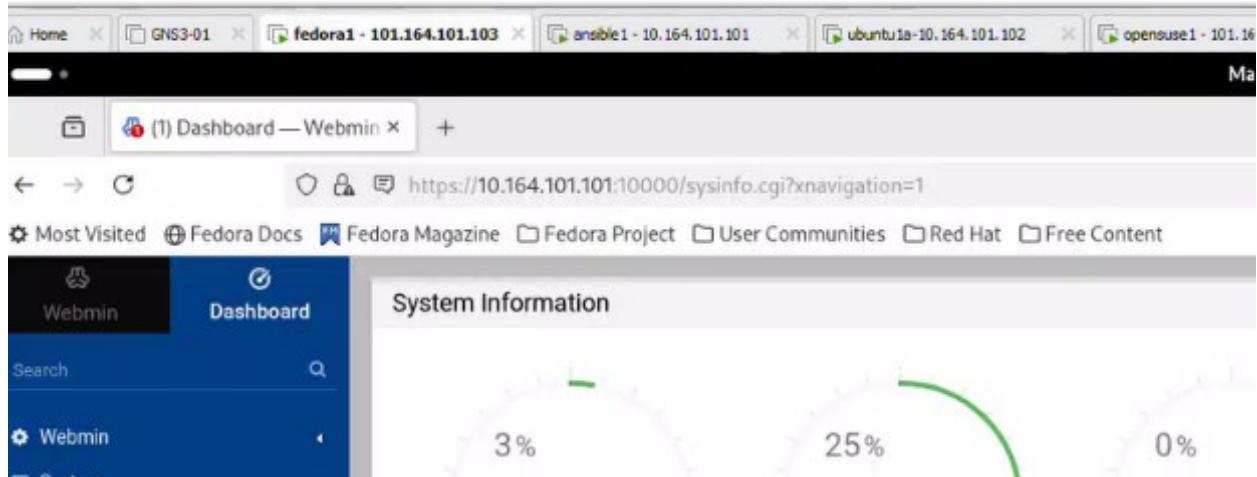
ansible@centos1:~/ $ 

```

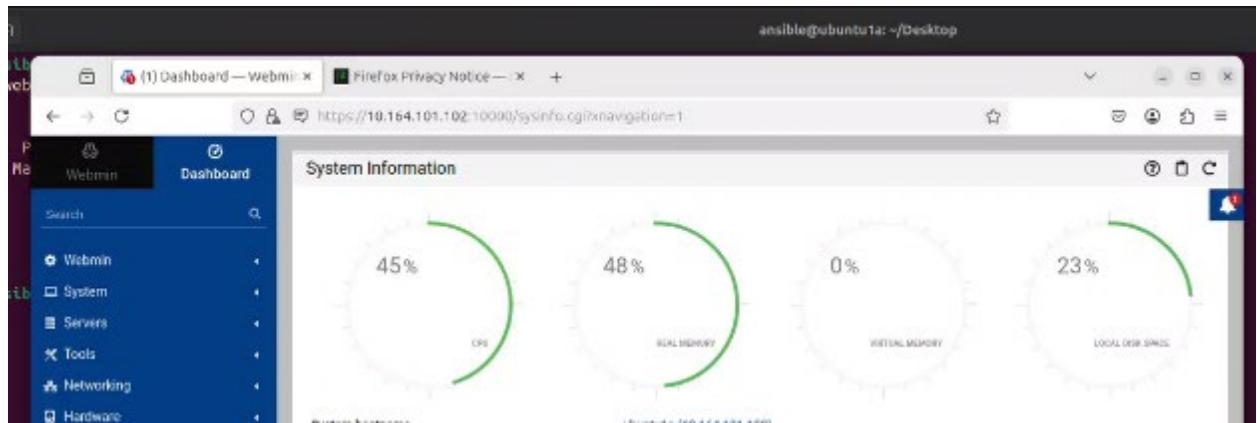
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

5. Test by accessing webmin.

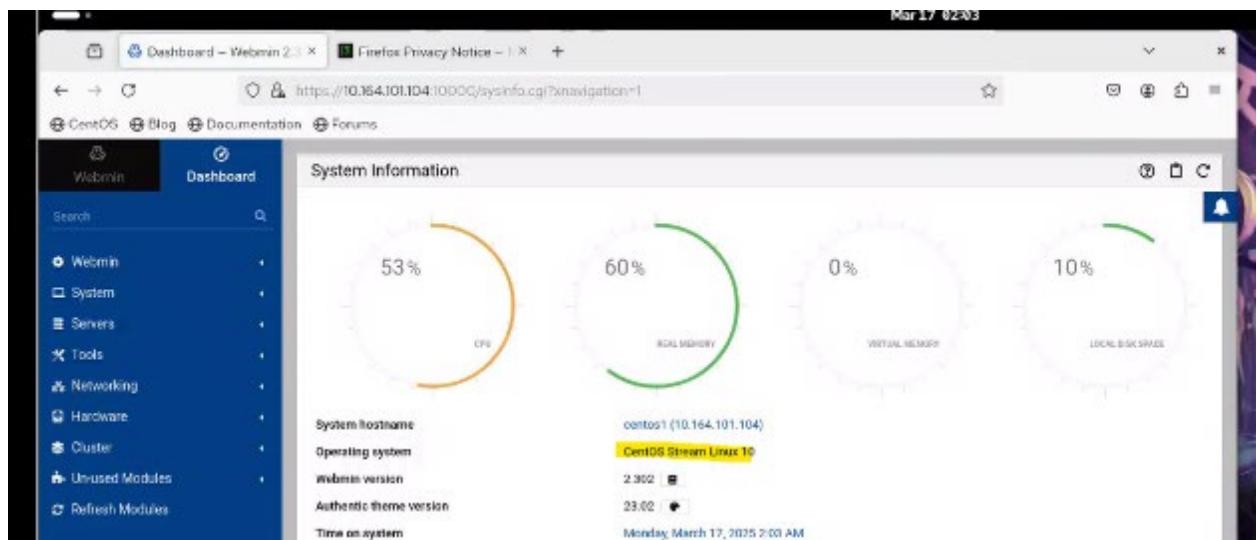
Fedor



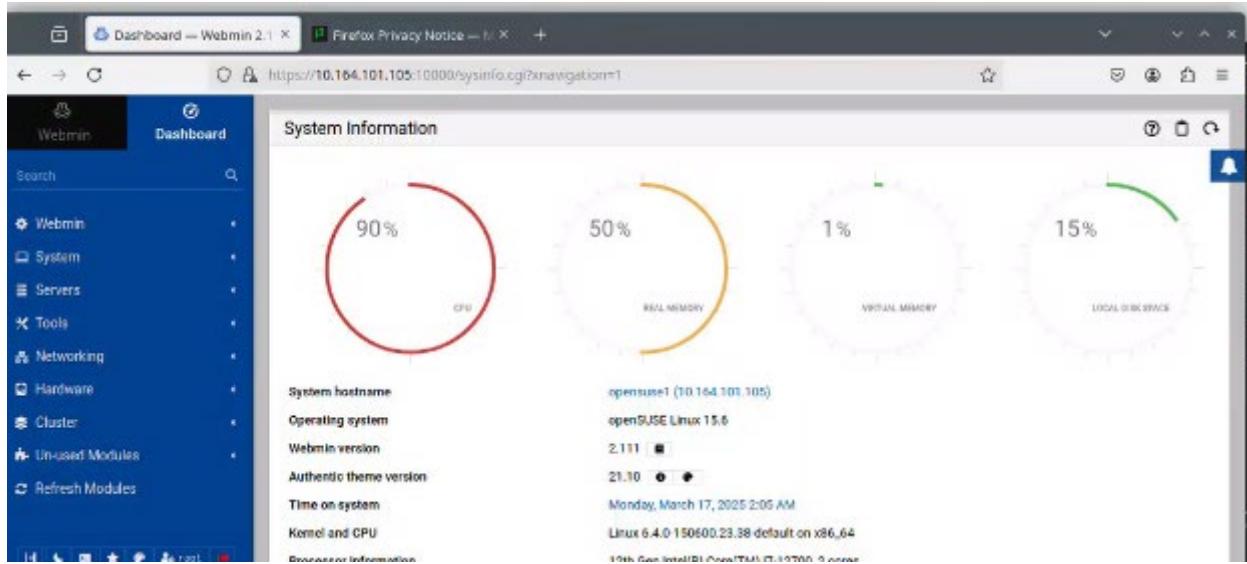
Ubuntu



Centos



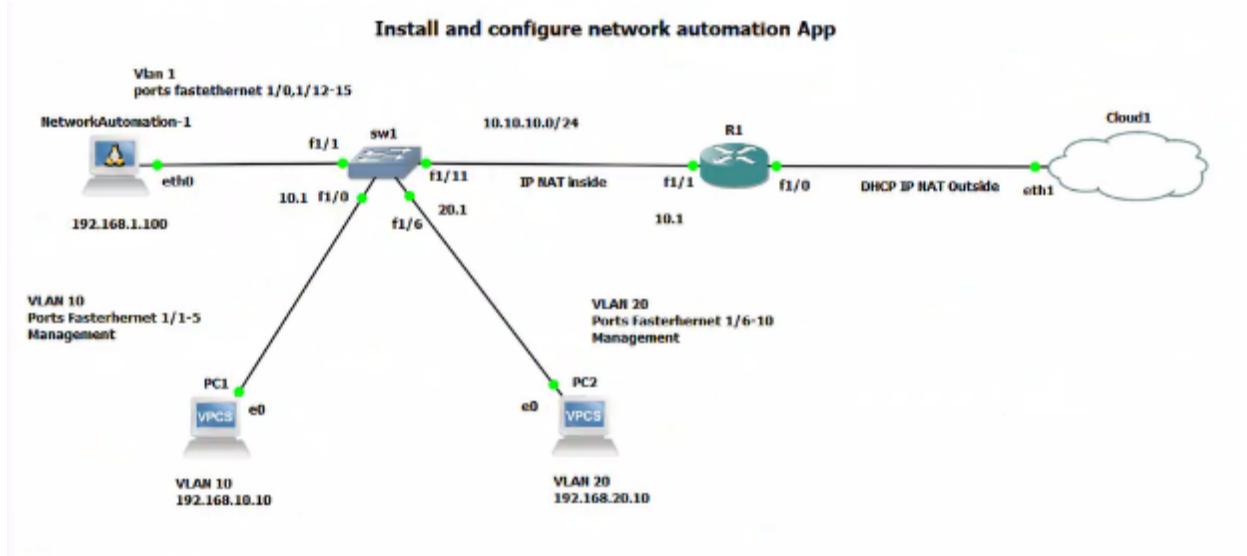
Opensuse



3.2 Test automation with python and GNS3 Network Automation App

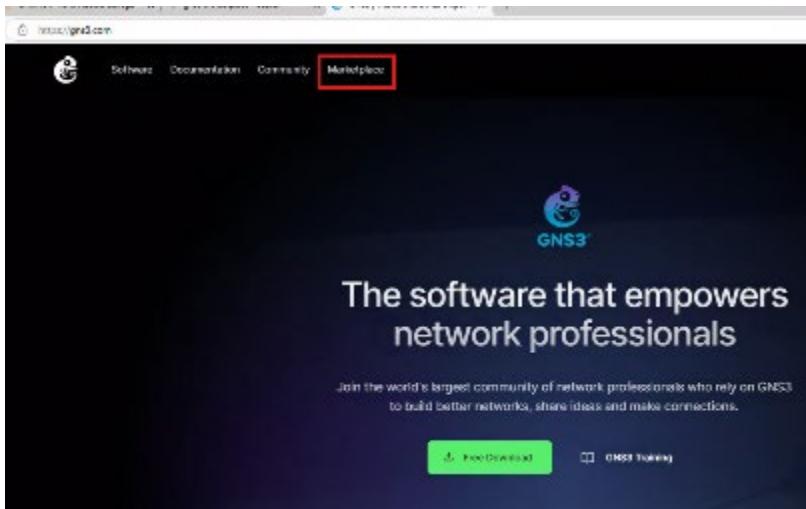
3.2.1 Install and configure Network automation APP

3.2.1.1 Topology



No shutdown

3.2.1.2 Appliances download



The screenshot shows the GNS3 Marketplace page. At the top, there is a navigation bar with links for Software, Documentation, Community, and Marketplace. The Marketplace link is highlighted with a red box. Below the navigation, the word "Marketplace" is prominently displayed. A sub-headline says "The one-stop networking shop for GNS3 Network Pros". On the left, there is a sidebar with categories: Featured (highlighted with a red box), Appliances (highlighted with a red box), Labs, and Software. In the center, there is a section titled "Appliances" with a sub-headline: "Easily add pre-configured appliances in GNS3 and integrate them to your projects and labs." Below this, there are two appliance cards: "FortiGate" and "Cisco 7200". Each card includes a small icon, the appliance name, a brief description, and interaction metrics like upvotes and views.

Locate appliance



Download it

Appliance

Network Automation

GNS3

Posted by Julien Duponchelle • June 29, 2017 at 12:25 UTC

[Download](#)

This container provides the popular tools used for network automation: Netmiko, NAPALM, Pyntc, and Ansible.

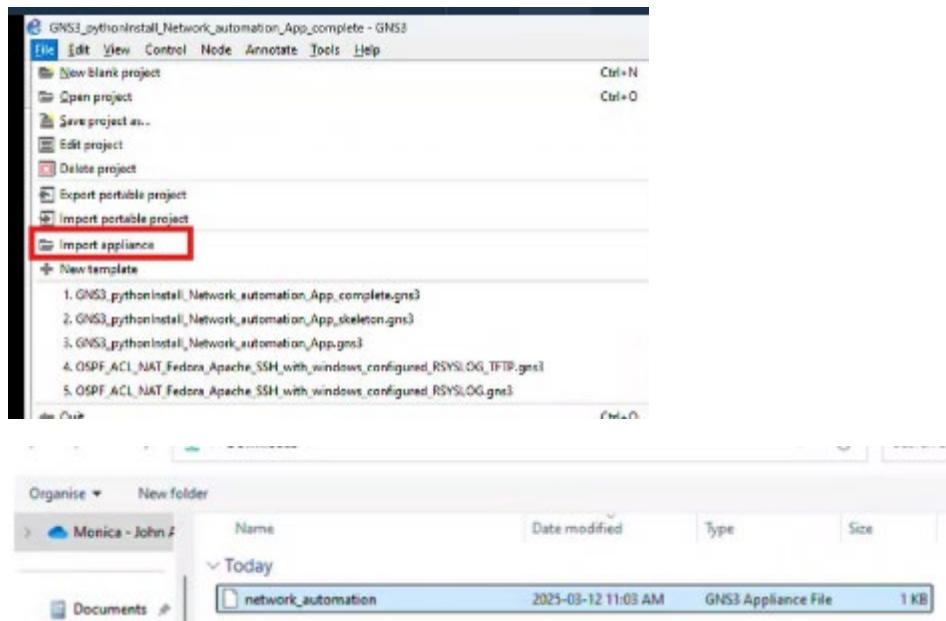
Views
78,758

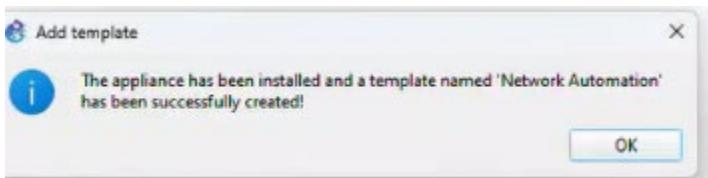
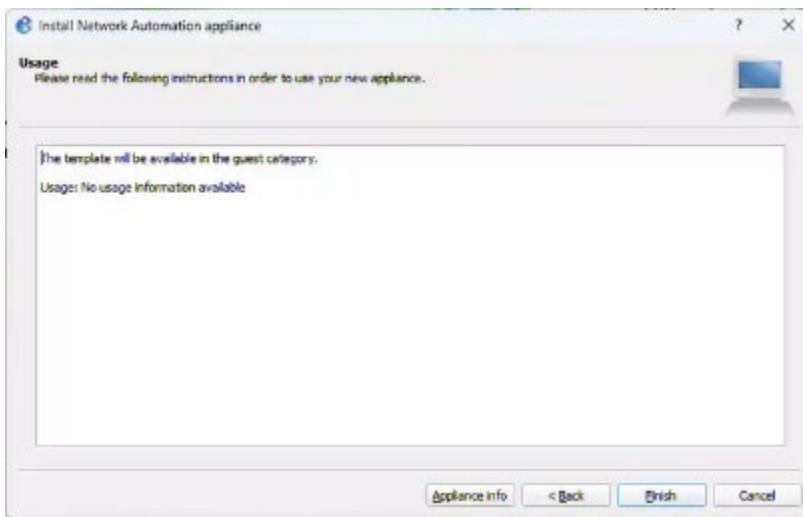
Replies
60

Last Updated
Dec 4, 2024

The appliance is downloaded

Import the appliance





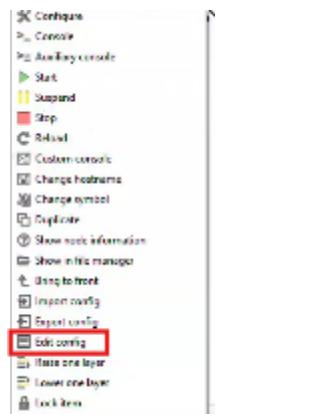
Drag appliance to topology



Configuration

Network Automation

Set ip address



```
# static config for eth0
auto eth0
iface eth0 inet static
    address 192.168.1.100
    netmask 255.255.255.0
    gateway 192.168.1.1
    up echo nameserver 8.8.8.8 > /etc/resolv.conf
# DHCP config for eth0
# auto eth0
# iface eth0 inet dhcp
```

```
# Static config for eth0
auto eth0
iface eth0 inet static
    address 192.168.1.100
    netmask 255.255.255.0
    gateway 192.168.1.1
    up echo nameserver 8.8.8.8 > /etc/resolv.conf
```

3.2.1.3 Scripts

```
!!!!!!!
!! SW1
!!!!!!!

enable
! VLANS
vlan database
vlan 10
vlan 20
exit

configure terminal
!
! Base Configuration
hostname sw1
no ip domain lookup
! Enable Layer 3 routing
ip routing
!
```

```

! Interface Configurations

interface range FastEthernet1/1 - 5
description Access ports for VLAN 10
switchport mode access
switchport access vlan 10
!
interface range FastEthernet1/6 - 10
description Access ports for VLAN 20
switchport mode access
switchport access vlan 20
!
interface FastEthernet1/11
description Layer 3 link to Router
no switchport
ip address 10.10.10.2 255.255.255.0
no shutdown
!

!
! VLAN Interfaces
interface Vlan1
description VLAN 1
ip address 192.168.1.1 255.255.255.0
no shutdown
!
interface Vlan10
description VLAN 10
ip address 192.168.10.1 255.255.255.0
no shutdown
!
interface Vlan20
description VLAN 20
ip address 192.168.20.1 255.255.255.0
no shutdown
!
! OSPF Configuration
router ospf 10
log-adjacency-changes
network 10.10.10.0 0.0.0.255 area 0
network 192.168.1.0 0.0.0.255 area 0
network 192.168.10.0 0.0.0.255 area 0
network 192.168.20.0 0.0.0.255 area 0
!
end

```

```

!!!!!!!
!! R1
!!!!!!!
enable
configure terminal
!
hostname R1
!

ip name-server 8.8.8.8
ip domain-lookup

! Interface Configurations

interface FastEthernet1/0

```

```
description connection R1 to cloud
ip address dhcp
ip nat outside
no shutdown
!
interface FastEthernet1/1
description connection R1 to SW1
ip address 10.10.10.1 255.255.255.0
ip nat inside
no shutdown
!
!
! OSPF Configuration
router ospf 10
network 10.10.10.0 0.0.0.255 area 0
default-information originate
!
! NAT and Routing
ip nat inside source list 1 interface FastEthernet1/0 overload
!
access-list 1 permit 10.10.10.0 0.0.0.255
access-list 1 permit 192.168.0.0 0.0.255.255
!
end
```

PC1

```
ip 192.168.10.10 255.255.255.0 192.168.10.1
```

```
save
```

PC2

```
ip 192.168.20.10 255.255.255.0 192.168.20.1
```

```
save
```

3.2.1.4 Connectivity test

PC1

Ping default gateway

PC1> ping 192.168.10.1

Ping PC2

PC1> ping 192.168.20.10

Ping Network automation

PC1> ping 192.168.1.100

Ping DNS

PC1> ping 8.8.8.8

NAME	IP/MASK	GATEWAY	MAC	LPORT	RHOST:PORT
PC1	192.168.10.10/24	192.168.10.1	00:50:79:66:68:00	20020	127.0.0.1:20021
	fe80::250:79ff:fe66:6800/64				

```
PC1>
PC1> ping 192.168.10.1

84 bytes from 192.168.10.1 icmp_seq=1 ttl=255 time=16.780 ms
84 bytes from 192.168.10.1 icmp_seq=2 ttl=255 time=1.037 ms
84 bytes from 192.168.10.1 icmp_seq=3 ttl=255 time=11.079 ms
84 bytes from 192.168.10.1 icmp_seq=4 ttl=255 time=3.893 ms
84 bytes from 192.168.10.1 icmp_seq=5 ttl=255 time=1.688 ms

PC1> ping 192.168.20.10

84 bytes from 192.168.20.10 icmp_seq=1 ttl=63 time=39.671 ms
84 bytes from 192.168.20.10 icmp_seq=2 ttl=63 time=21.170 ms
84 bytes from 192.168.20.10 icmp_seq=3 ttl=63 time=31.938 ms
84 bytes from 192.168.20.10 icmp_seq=4 ttl=63 time=26.594 ms
84 bytes from 192.168.20.10 icmp_seq=5 ttl=63 time=34.213 ms

PC1> ping 192.168.1.100

84 bytes from 192.168.1.100 icmp_seq=1 ttl=63 time=25.428 ms
84 bytes from 192.168.1.100 icmp_seq=2 ttl=63 time=17.601 ms
84 bytes from 192.168.1.100 icmp_seq=3 ttl=63 time=11.536 ms
84 bytes from 192.168.1.100 icmp_seq=4 ttl=63 time=22.804 ms
84 bytes from 192.168.1.100 icmp_seq=5 ttl=63 time=20.771 ms

PC1> ping 8.8.8.8

84 bytes from 8.8.8.8 icmp_seq=1 ttl=116 time=33.855 ms
84 bytes from 8.8.8.8 icmp_seq=2 ttl=116 time=40.082 ms
84 bytes from 8.8.8.8 icmp_seq=3 ttl=116 time=47.514 ms
84 bytes from 8.8.8.8 icmp_seq=4 ttl=116 time=28.437 ms
84 bytes from 8.8.8.8 icmp_seq=5 ttl=116 time=34.940 ms

PC1> [REDACTED]
```

PC2

Ping default gateway

PC1> ping 192.168.20.1

Ping PC1

PC1> ping 192.168.10.10

Ping Network automation

PC1> ping 192.168.1.100

Ping DNS

PC1> ping 8.8.8.8

```
PC2> show

NAME      IP/MASK          GATEWAY        MAC           LPORT   RHOST:PORT
PC2      192.168.20.10/24  192.168.20.1   00:50:79:66:68:01  20018  127.0.0.1:20019
        fe80::250:79ff:fe66:6801/64

PC2> ping 192.168.20.1

84 bytes from 192.168.20.1 icmp_seq=1 ttl=255 time=16.472 ms
84 bytes from 192.168.20.1 icmp_seq=2 ttl=255 time=13.835 ms
84 bytes from 192.168.20.1 icmp_seq=3 ttl=255 time=7.889 ms
84 bytes from 192.168.20.1 icmp_seq=4 ttl=255 time=10.869 ms
84 bytes from 192.168.20.1 icmp_seq=5 ttl=255 time=11.190 ms

PC2> ping 192.168.10.10

84 bytes from 192.168.10.10 icmp_seq=1 ttl=63 time=23.342 ms
84 bytes from 192.168.10.10 icmp_seq=2 ttl=63 time=52.455 ms
84 bytes from 192.168.10.10 icmp_seq=3 ttl=63 time=16.108 ms
84 bytes from 192.168.10.10 icmp_seq=4 ttl=63 time=26.288 ms
84 bytes from 192.168.10.10 icmp_seq=5 ttl=63 time=21.298 ms

PC2> ping 192.168.1.100

84 bytes from 192.168.1.100 icmp_seq=1 ttl=63 time=21.931 ms
84 bytes from 192.168.1.100 icmp_seq=2 ttl=63 time=20.706 ms
84 bytes from 192.168.1.100 icmp_seq=3 ttl=63 time=20.942 ms
84 bytes from 192.168.1.100 icmp_seq=4 ttl=63 time=18.108 ms
84 bytes from 192.168.1.100 icmp_seq=5 ttl=63 time=13.969 ms

PC2> ping 8.8.8.8

84 bytes from 8.8.8.8 icmp_seq=1 ttl=116 time=34.870 ms
84 bytes from 8.8.8.8 icmp_seq=2 ttl=116 time=40.989 ms
84 bytes from 8.8.8.8 icmp_seq=3 ttl=116 time=39.689 ms
84 bytes from 8.8.8.8 icmp_seq=4 ttl=116 time=54.908 ms
84 bytes from 8.8.8.8 icmp_seq=5 ttl=116 time=39.576 ms

PC2> █
```

3.2.1.5 Network Automation

Ping default gateway

ping 192.168.1.1

Ping PC1

ping 192.168.10.10

Ping PC2

ping 192.168.20.10

Ping DNS

```
ping 8.8.8.8
```

```
ping google.ca
```

```
root@NetworkAutomation-1:~# ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=255 time=7.31 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=255 time=11.1 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=255 time=11.6 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=255 time=15.6 ms
^C
--- 192.168.1.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3014ms
rtt min/avg/max/mdev = 7.313/11.373/15.571/2.925 ms
root@NetworkAutomation-1:~#
root@NetworkAutomation-1:~# ping 192.168.10.10
PING 192.168.10.10 (192.168.10.10) 56(84) bytes of data.
64 bytes from 192.168.10.10: icmp_seq=1 ttl=63 time=31.3 ms
64 bytes from 192.168.10.10: icmp_seq=2 ttl=63 time=20.9 ms
64 bytes from 192.168.10.10: icmp_seq=3 ttl=63 time=21.9 ms
^C
--- 192.168.10.10 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2009ms
rtt min/avg/max/mdev = 20.916/24.709/31.320/4.691 ms
root@NetworkAutomation-1:~# ping 192.168.20.10
PING 192.168.20.10 (192.168.20.10) 56(84) bytes of data.
64 bytes from 192.168.20.10: icmp_seq=1 ttl=63 time=28.9 ms
64 bytes from 192.168.20.10: icmp_seq=2 ttl=63 time=15.1 ms
64 bytes from 192.168.20.10: icmp_seq=3 ttl=63 time=17.8 ms
^C
--- 192.168.20.10 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 15.064/20.593/28.906/5.983 ms
root@NetworkAutomation-1:~# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=116 time=45.6 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=116 time=47.2 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 45.609/46.419/47.229/0.810 ms
```

```
root@NetworkAutomation-1:~# ping google.ca
PING google.ca (142.250.69.99) 56(84) bytes of data.
64 bytes from pnyula-ab-in-f3.1e100.net (142.250.69.99): icmp_seq=1 ttl=115 time=30.4 ms
64 bytes from pnyula-ab-in-f3.1e100.net (142.250.69.99): icmp_seq=2 ttl=115 time=42.0 ms
^C
--- google.ca ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 30.385/36.188/41.991/5.803 ms
root@NetworkAutomation-1:~#
```

3.2.1.6 Printouts

Test NAT

Pings from Network automation is seen

```
R1#show ip nat translations
Pro Inside global      Inside local        Outside local       Outside global
icmp 10.164.0.45:1024  192.168.1.100:21   142.250.69.99:21  142.250.69.99:1024
udp 10.164.0.45:4502   192.168.1.100:46602  8.8.8.8:53        8.8.8.8:53
udp 10.164.0.45:4501   192.168.1.100:56766  8.8.8.8:53        8.8.8.8:53
R1#
```

3.2.1.7 Workaround for DNS to work

Release and renew DHCP address in f1/0 connected to cloud

release dhcp f1/0

renew dhcp f1/0

```
R1#
R1#release dhcp F1/0
R1#renew dhcp F1/0
R1#
*Mar 13 19:12:11.523: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet1/0 assigned DHCP address 10.164.0.45, mask 255.255.0.0, hostname R1
R1#
```

Verify if Gateway of last resort route is present

```
R1#show ip route
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, 1 - LISP
      + - replicated route, % - next hop override

Gateway of last resort is 10.164.0.1 to network 0.0.0.0

S*  0.0.0.0/0 [254/0] via 10.164.0.1
    10.0.0.0/8 is variably subnetted, 5 subnets, 3 masks
C    10.10.10.0/24 is directly connected, FastEthernet1/1
L    10.10.10.1/32 is directly connected, FastEthernet1/1
S    10.162.240.51/32 [254/0] via 10.164.0.1, FastEthernet1/0
C    10.164.0.0/16 is directly connected, FastEthernet1/0
L    10.164.0.45/32 is directly connected, FastEthernet1/0
O    192.168.1.0/24 [110/2] via 10.10.10.2, 00:20:55, FastEthernet1/1
O    192.168.10.0/24 [110/2] via 10.10.10.2, 00:20:55, FastEthernet1/1
O    192.168.20.0/24 [110/2] via 10.10.10.2, 00:20:55, FastEthernet1/1
R1#
```

Verify the DHCP lease

```

R1#show dhcp lease
Temp IP addr: 10.164.0.45 for peer on Interface: FastEthernet1/0
Temp sub net mask: 255.255.0.0
    DHCP Lease server: 10.162.240.51, state: 5 Bound
    DHCP transaction id: 110A
    Lease: 3600 secs, Renewal: 1800 secs, Rebind: 3150 secs
Temp default-gateway addr: 10.164.0.1
Next timer fires after: 00:26:46
Retry count: 0 Client-ID: cisco-ca02.c7ec.001c-Fa1/0
Client-ID hex dump: 636973636F2D636130322E633765632E
                           303031632D4661312F30
Hostname: R1
R1#

```

Note Gateway in DHCP is Gateway of last resort automatically set.

3.2.1.8 Small test with Python3

Disable asking for login password run in SW1 and R1

```

enable
configure terminal
line vty 0 4
no login
end

```

Write script in Network-automationm-1

nano getroutes_sw1_r1.py

```

root@NetworkAutomation-1:~# nano getroutes_sw1_r1.py

```

```

import telnetlib
import time

# Device details
DEVICES = [
    {"host": "192.168.1.1", "name": "SW1"},
    {"host": "10.10.10.1", "name": "R1"}
]

for device in DEVICES:
    HOST = device["host"]
    NAME = device["name"]

    try:
        # Establish Telnet connection
        print(f"\nConnecting to {NAME} ({HOST})...")
        tn = telnetlib.Telnet(HOST, timeout=10)

        # No login or enable password prompts
        tn.write(b"enable\n")

```

```

time.sleep(0.5)

# Send show ip route command
tn.write(b"terminal length 0\n")
time.sleep(0.5)
tn.write(b"show ip route\n")
time.sleep(1)

# Exit and read output
tn.write(b"exit\n")
output = tn.read_all().decode('ascii')

# Print the results
print(f"\nRouting Table for {NAME}:")

print("=" * 50)
print(output)
print("=" * 50)

except Exception as e:
    print(f"Error connecting to {HOST}: {str(e)}")
finally:
    try:
        tn.close()
    except:
        pass

```

This script automates the process of connecting to network devices (SW1 and R1 in your GNS3 setup) via Telnet, executing the show ip route command, and displaying their routing tables. Here's a step-by-step breakdown:

1. Imports Libraries:

- o import telnetlib: Provides the Telnet client functionality to connect to devices.
- o import time: Allows pausing execution (e.g., time.sleep()) to give devices time to respond.

2. Defines Devices:

- o DEVICES = [...]: A list of dictionaries containing the IP addresses and names of the devices to connect to:
 - SW1 at 192.168.1.1
 - R1 at 10.10.10.1

3. Loops Through Devices:

- o for device in DEVICES:: Iterates over each device (SW1 and R1) to perform the same actions on both.

4. Sets Variables:

- o HOST = device["host"]: Extracts the IP address (e.g., "192.168.1.1" for SW1).
- o NAME = device["name"]: Extracts the device name (e.g., "SW1").

5. Tries to Connect and Execute Commands:

- o Error Handling: The try block ensures that if something goes wrong (e.g., connection failure), it catches the error gracefully.
- o print(f"\nConnecting to {NAME} ({HOST})..."): Prints a message indicating which device it's connecting to.
- o tn = telnetlib.Telnet(HOST, timeout=10): Opens a Telnet connection to the device's IP address with a 10-second timeout.

6. Sends Commands:

- tn.write(b"enable\n"): Sends the enable command to enter privileged EXEC mode (sw1# or R1#). Since you specified no enable password, it assumes no prompt follows.
 - time.sleep(0.5): Waits half a second to ensure the device processes the command.
- tn.write(b"terminal length 0\n"): Disables output pagination so the entire routing table is returned without pauses.
 - time.sleep(0.5): Another brief wait.
- tn.write(b"show ip route\n"): Sends the command to display the routing table.
 - time.sleep(1): Waits 1 second for the device to generate and send the output.

7. Exits and Reads Output:

- tn.write(b"exit\n"): Exits the Telnet session.
- output = tn.read_all().decode('ascii'): Reads all output from the session (commands and responses) and converts it from bytes to a readable string.

8. Displays Results:

- print(f"\nRouting Table for {NAME}:") : Labels the output with the device name.
- print("=" * 50): Prints a line of 50 equals signs as a separator.
- print(output): Prints the full Telnet session output, including the routing table.
- print("=" * 50): Another separator.

9. Handles Errors:

- except Exception as e:: Catches any errors (e.g., connection refused) and prints them (e.g., Error connecting to 192.168.1.1: [error message]).

10. Closes Connection:

- finally:: Ensures the Telnet connection is closed, even if an error occurs.
- tn.close(): Closes the Telnet session.
- The try/except around tn.close() prevents errors if the connection was never opened.

3.2.1.9 Run script

Python3 getroutes_sw1_r1_.py

```
[root@NetworkAutomation-1:~# python3 getroutes_sw1_r1.py
```

Output

```
=====
Connecting to R1 (10.10.10.1)...

Routing Table for R1:
=====

R1>enable
% No password set
R1>terminal length 0
R1>show ip route
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
      i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
      + - replicated route, % - next hop override

Gateway of last resort is 10.164.0.1 to network 0.0.0.0

S*   0.0.0.0/0 [254/0] via 10.164.0.1
    10.0.0.8 is variably subnetted, 5 subnets, 3 masks
C     10.10.10.0/24 is directly connected, FastEthernet1/1
L     10.10.10.1/32 is directly connected, FastEthernet1/1
S     10.162.240.51/32 [254/0] via 10.164.0.1, FastEthernet1/0
C     10.164.0.0/16 is directly connected, FastEthernet1/0
L     10.164.0.45/32 is directly connected, FastEthernet1/0
O     192.168.1.0/24 [110/2] via 10.10.10.2, 00:55:22, FastEthernet1/1
O     192.168.10.0/24 [110/2] via 10.10.10.2, 00:55:22, FastEthernet1/1
O     192.168.20.0/24 [110/2] via 10.10.10.2, 00:55:22, FastEthernet1/1
R1>exit
=====

root@NetworkAutomation-1:~#
```

Restore login

```
enable
configure terminal
line vty 0 4
login
end
```

If script tried to run after login is set

```
root@NetworkAutomation-1:~# python3 getroutes_sw1_r1.py
Connecting to SW1 (192.168.1.1)...
Routing Table for SW1:
=====
Password required, but none set
=====
Connecting to R1 (10.10.10.1)...
Routing Table for R1:
=====
Password required, but none set
=====
root@NetworkAutomation-1:~#
```

3.2.2 GNS3 Python excercises

References

1. Video youtube GNS3 Talks: Python for Network Engineers with GNS3 (Part 1). Network programmability made easy. https://www.youtube.com/watch?v=IhroIrV9_7w
2. Video youtube GNS3 Talks: Python for Network Engineers with GNS3 (Part 2) - Configure VLANs on switches.
https://www.youtube.com/watch?v=_XGtQRUWasQ
3. Video youtube GNS3 Talks: Python for Network Engineers with GNS3 (Part 3) - Remove Passwords and improve scripts
<https://www.youtube.com/watch?v=ViGoll0-g7s>
4. Video youtube GNS3 Talks: Python for Network Engineers with GNS3 (Part 4)
<https://www.youtube.com/watch?v=dE2afwB9d5U>

Tutorials on line

- 1 - GNS3 Talks: Python for Network Engineers with GNS3 (Part 1). Network programmability made easy
<https://tutorialsonline.ca/courses/1585373/lectures/36674154>

2 - GNS3 - Install and Configure Network Automation App and Project Config Overview

<https://tutorialsonline.ca/courses/1585373/lectures/36723212>

5 - GNS3 GNS3 Talks: Python for Network Engineers with GNS3 (Part 3) - Remove Passwords and improve scripts

<https://tutorialsonline.ca/courses/1585373/lectures/36716485>

6 - GNS3 Talks: Python for Network Engineers with GNS3 (Part 4) - Create switch VLANs using loops

<https://tutorialsonline.ca/courses/1585373/lectures/36717271>

8 - GNS3 Talks: Python for Network Engineers with GNS3 (Part 5) - Multiple switches, multiple VLANs

<https://tutorialsonline.ca/courses/1585373/lectures/36738254>

3.2.2.1 Setting up python script to create/configure R1 loopback interface

1. Configure login and telnet remote access for R1.

```
!!!!!!!
!!! R1 !!!
!!!!!!!
configure terminal
enable pass
enable password cisco
username student password cisco
line vty 0 4
  login local
  transport input all
End
```

```
R1#
R1#!!!!!!!
R1#!!! R1 !!!
R1#!!!!!!!
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#enable pass
% Incomplete command.

R1(config)#enable password cisco
R1(config)#username student password cisco
R1(config)#line vty 0 4
R1(config-line)# login local
R1(config-line)# transport input all
R1(config-line)#End
R1#
*Mar 14 14:45:10.196: %SYS-5-CONFIG_I: Configured from console by console
R1#
R1#
R1#
```

2. Before running the script, verify that you have telnet access to router.

Use : student/cisco

```
Enable password : cisco
```

```
telnet 10.10.10.1
```

```
root@UbuntuDockerGuest-1:~# telnet 10.10.10.1
Trying 10.10.10.1...
Connected to 10.10.10.1
Escape character is '^]'.
```

```
User Access Verification
```

```
Username: student
Password:
R1>enable
Password:
R1#
R1#exit
```

```
root@NetworkAutomation-1:~# telnet 10.10.10.1
Trying 10.10.10.1...
Connected to 10.10.10.1.
Escape character is '^]'.
```

```
User Access Verification
```

```
Username: student
Password:
R1>enable
Password:
R1#
R1#exit
Connection closed by foreign host.
root@NetworkAutomation-1:~#
```

3. Create a file to input our script. We will name it **pythonR1script1**.

```

GNU nano 4.8                                         pythonR1script.py
import getpass
import telnetlib
import time

HOST = "10.10.10.1" # R1

try:
    # Get credentials
    user = input("Enter your telnet username: ") # Changed from raw_input to input
    password = getpass.getpass("Enter your telnet password: ")
    enable_password = getpass.getpass("Enter your enable password: ")

    # Connect to R1
    print(f"Connecting to {HOST}...")
    tn = telnetlib.Telnet(HOST, timeout=10)

    # Authenticate
    tn.read_until(b"Username: ", timeout=5)
    tn.write(user.encode('ascii') + b"\n")
    tn.read_until(b>Password: ", timeout=5)
    tn.write(password.encode('ascii') + b"\n")

    # Enter enable mode
    tn.write(b"enable\n")
    tn.read_until(b>Password: ", timeout=5)
    tn.write(enable_password.encode('ascii') + b"\n")
    time.sleep(0.5)

    # Configure loopbacks and OSPF
    commands = [
        b"conf t\n",
        b"int loop 0\n",
        b"ip address 1.1.1.1 255.255.255.255\n",
        b"int loop 1\n",
        b"ip address 2.2.2.2 255.255.255.255\n",
    ]

```

[Read 59 lines]

```

import getpass
import telnetlib
import time

HOST = "10.10.10.1" # R1

try:
    # Get credentials
    user = input("Enter your telnet username: ") # Changed from raw_input to input
    password = getpass.getpass("Enter your telnet password: ")
    enable_password = getpass.getpass("Enter your enable password: ")

    # Connect to R1
    print(f"Connecting to {HOST}...")
    tn = telnetlib.Telnet(HOST, timeout=10)

    # Authenticate
    tn.read_until(b"Username: ", timeout=5)
    tn.write(user.encode('ascii') + b"\n")
    tn.read_until(b>Password: ", timeout=5)
    tn.write(password.encode('ascii') + b"\n")

    # Enter enable mode
    tn.write(b"enable\n")
    tn.read_until(b>Password: ", timeout=5)
    tn.write(enable_password.encode('ascii') + b"\n")
    time.sleep(0.5)

```

```

# Configure loopbacks and OSPF
commands = [
    b"conf t\n",
    b"int loop 0\n",
    b"ip address 1.1.1.1 255.255.255.255\n",
    b"int loop 1\n",
    b"ip address 2.2.2.2 255.255.255.255\n",
    b"router ospf 1\n",
    b"network 0.0.0.0 255.255.255.255 area 0\n",
    b"end\n",
    b"exit\n"
]

for cmd in commands:
    tn.write(cmd)
    time.sleep(0.5)

# Read and print output
output = tn.read_all().decode('ascii')
print("\nConfiguration Output for R1:")
print("=" * 50)
print(output)
print("=" * 50)

except Exception as e:
    print(f"Error connecting to {HOST}: {str(e)}")
finally:
    try:
        tn.close()
    except:
        pass
    print(f"Error connecting to {HOST}: {str(e)}")
finally:
    try:
        tn.close()
    except:
        pass

```

4. Run the script and provide the needed information to access and configure the router:

```

root@NetworkAutomation-1:~# python3 pythonR1script.py
Enter your telnet username: student
Enter your telnet password:
Enter your enable password:
Connecting to 10.10.10.1...

Configuration Output for R1:
=====
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int loop 0
R1(config-if)#ip address 1.1.1.1 255.255.255.255
R1(config-if)#int loop 1
R1(config-if)#ip address 2.2.2.2 255.255.255.255
R1(config-if)#router ospf 1
R1(config-router)#network 0.0.0.0 255.255.255.255 area 0
R1(config-router)#end
R1#exit

```

```

=====
root@NetworkAutomation-1:~# python3 pythonR1script.py
Enter your telnet username: student
Enter your telnet password:
Enter your enable password:
Connecting to 10.10.10.1...

Configuration Output for R1:
=====

R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#int loop 0
R1(config-if)#ip address 1.1.1.1 255.255.255.255
R1(config-if)#int loop 1
R1(config-if)#ip address 2.2.2.2 255.255.255.255
R1(config-if)#router ospf 1
R1(config-router)#network 0.0.0.0 255.255.255.255 area 0
R1(config-router)#end
R1#exit

=====
root@NetworkAutomation-1:~# 

```

See the output in R1

```

"""
R1#
*Mar 14 15:27:09.636: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R1#
*Mar 14 15:27:10.860: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1, changed state to up
R1#
*Mar 14 15:27:12.684: %SYS-5-CONFIG_I: Configured from console by student on vty0 (192.168.1.100)
R1# 

```

See printout show ip interface brief loopback is there

```

R1>show ip int
R1#show ip interface brief
Interface          IP-Address      OK? Method Status          Protocol
FastEthernet0/0    unassigned     YES NVRAM administratively down down
FastEthernet1/0    10.164.0.45   YES DHCP    up             up
FastEthernet1/1    10.10.10.1   YES NVRAM up             up
Serial2/0          unassigned     YES NVRAM administratively down down
Serial2/1          unassigned     YES NVRAM administratively down down
Serial2/2          unassigned     YES NVRAM administratively down down
Serial2/3          unassigned     YES NVRAM administratively down down
Loopback0          1.1.1.1       YES manual  up             up
Loopback1          2.2.2.2       YES manual  up             up
R1# 

```

3.2.2.2 Switch script create VLANS 1 by 1

1. Configure login and telnet remote access for S1.

```
!!!!!!!
!!! S1 !!!
!!!!!!!
configure terminal
enable pass
enable password cisco
username student password cisco
line vty 0 4
  login local
  transport input all
End
```

2. Before running the script, verify that you have telnet access to router.

Use : student/cisco

Enable password : cisco

```
root@NetworkAutomation-1:~# telnet 192.168.1.1
Trying 192.168.1.1...
Connected to 192.168.1.1.
Escape character is '^]'.
```

User Access Verification

```
Username: student
Password:
sw1>enable
Password:
sw1#
sw1#exit
Connection closed by foreign host.
root@NetworkAutomation-1:~#
```

```
root@NetworkAutomation-1:~# telnet 192.168.1.1
Trying 192.168.1.1...
Connected to 192.168.1.1.
Escape character is '^]'.

User Access Verification

Username: student
Password:
sw1>enable
Password:
sw1#
sw1#exit
```

3. Create script

```
#!/usr/bin/env python3
import getpass
import telnetlib
import time

HOST = "192.168.1.1" # SW1

# Get credentials
user = input("Enter your telnet username: ")
password = getpass.getpass("Enter your telnet password: ")
enable_password = getpass.getpass("Enter your enable password: ")

# Connect to SW1
print(f"Connecting to {HOST}...")
tn = telnetlib.Telnet(HOST, timeout=10)

# Authenticate
tn.read_until(b"Username: ") # Wait for username prompt
tn.write((user + "\n").encode('ascii'))
time.sleep(0.5) # Small delay to ensure password prompt appears

if password:
    tn.read_until(b"Password: ") # Wait for password prompt
    tn.write((password + "\n").encode('ascii'))
time.sleep(0.5) # Small delay to ensure enable prompt appears

# Enter enable mode
tn.write(b"enable\n")
tn.read_until(b"Password: ") # Wait for enable password prompt
tn.write((enable_password + "\n").encode('ascii'))
```

```
time.sleep(0.5)

# Create VLANs using vlan database
tn.write(b"vlan database\n")
time.sleep(0.5)
```

4. Test script

```
./pythonS1script.py
```

```
root@NetworkAutomation-1:~# ./pythonS1script.py
Enter your telnet username: student
Enter your telnet password:
Enter your enable password:
Connecting to 192.168.1.1...

sw1#vlan database
sw1(vlan)#vlan 2
VLAN 2 added:
    Name: VLAN0002
sw1(vlan)#vlan 3
VLAN 3 added:
    Name: VLAN0003
sw1(vlan)#vlan 4
VLAN 4 added:
    Name: VLAN0004
sw1(vlan)#vlan 5
VLAN 5 added:
    Name: VLAN0005
sw1(vlan)#vlan 6
VLAN 6 added:
    Name: VLAN0006
sw1(vlan)#exit
APPLY completed.
Exiting....
sw1#exit
```

```
root@NetworkAutomation-1:~#
```

5. Print the vlans in SW1 to verify Vlans were defined

```

SW1# show vlan-switch

VLAN Name          Status    Ports
----  -----
1    default        active    Fa1/0, Fa1/12, Fa1/13, Fa1/14
                           Fa1/15
2    VLAN0002       active
3    VLAN0003       active
4    VLAN0004       active
5    VLAN0005       active
6    VLAN0006       active
10   VLAN0010       active    Fa1/1, Fa1/2, Fa1/3, Fa1/4
                           Fa1/5
20   VLAN0020       active    Fa1/6, Fa1/7, Fa1/8, Fa1/9
                           Fa1/10
1002 fddi-default  active
1003 token-ring-default  active
1004 fdnet-default   active
1005 trnet-default   active

VLAN Type  SAID      MTU  Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
----  -----
1    enet  100001    1500  -     -     -     -     -     1002  1003
2    enet  100002    1500  -     -     -     -     -     0     0

VLAN Type  SAID      MTU  Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
----  -----
3    enet  100003    1500  -     -     -     -     -     0     0
4    enet  100004    1500  -     -     -     -     -     0     0
5    enet  100005    1500  -     -     -     -     -     0     0
6    enet  100006    1500  -     -     -     -     -     0     0
10   enet  100010    1500  -     -     -     -     -     0     0
20   enet  100020    1500  -     -     -     -     -     0     0
1002 fddi 101002    1500  -     -     -     -     -     1     1003
1003 tr   101003    1500  1005  0     -     -     srb   1     1002
1004 fdnet 101004   1500  -     -     1     ibm   -     0     0
1005 trnet 101005   1500  -     -     1     ibm   -     0     0
sw1#

```

3.2.2.3 Remove password for user student and Create multiple vlans (vlan loop)

1. Remove password for user student

User account named student is assigned privilege level 15, which is the highest privilege level on a Cisco device. This essentially gives the student user full administrative access to the device, allowing them to execute all commands.

```

!!!!!!!
!!! SW1
!!!!!!!
config t
user student privilege 15
end

```

2. Create script

```
nano pythonS1_vlan_loop.py
```

script

```
#!/usr/bin/env python3
import getpass
import telnetlib
import time

HOST = "192.168.1.1" # SW1

# Get credentials
user = input("Enter your telnet username: ")
password = getpass.getpass("Enter your telnet password: ")

# Connect to SW1
print(f"Connecting to {HOST}...")
tn = telnetlib.Telnet(HOST, timeout=10)

# Authenticate
tn.read_until(b"Username: ") # Wait for username prompt
tn.write((user + "\n").encode('ascii'))
time.sleep(0.5) # Small delay to ensure password prompt appears

if password:
    tn.read_until(b"Password: ") # Wait for password prompt
    tn.write((password + "\n").encode('ascii'))
    time.sleep(0.5) # Small delay to ensure enable prompt appears

# Create VLANs using vlan database
tn.write(b"vlan database\n")
time.sleep(0.5)

# Create VLANs 30 to 39
for n in range(30, 40):
    tn.write(("vlan " + str(n) + "\n").encode('ascii'))
    time.sleep(0.1)

# Exit vlan database and session
tn.write(b"exit\n") # Exit vlan database mode
time.sleep(0.5)
tn.write(b"exit\n") # Exit session

# Print output
print(tn.read_all().decode('ascii'))
```

2. Give execution permissions to script

```
chmod +x pythonS1_vlan_loop
```

3. Test script

```
root@NetworkAutomation-1:~# nano pythonS1_vlan_loop.py
root@NetworkAutomation-1:~# ./pythonS1_vlan_loop.py
Enter your telnet username: student
Enter your telnet password:
Connecting to 192.168.1.1...

sw1#vlan database
sw1(vlan)#vlan 30
VLAN 30 modified:
sw1(vlan)#vlan 31
VLAN 31 modified:
sw1(vlan)#vlan 32
VLAN 32 modified:
sw1(vlan)#vlan 33
VLAN 33 modified:
sw1(vlan)#vlan 34
VLAN 34 modified:
sw1(vlan)#vlan 35
VLAN 35 modified:
sw1(vlan)#exit
APPLY completed.
Exiting.....
sw1#exit

root@NetworkAutomation-1:~#
```

```
Enter your telnet username: student
Enter your telnet password:
Enter your enable password:
Connecting to 192.168.1.1...

sw1#vlan database
sw1(vlan)#vlan 30
VLAN 30 added:
    Name: VLAN0030
sw1(vlan)#vlan 31
VLAN 31 added:
    Name: VLAN0031
sw1(vlan)#vlan 32
VLAN 32 added:
    Name: VLAN0032
sw1(vlan)#vlan 33
VLAN 33 added:
    Name: VLAN0033
sw1(vlan)#vlan 34
VLAN 34 added:
    Name: VLAN0034
sw1(vlan)#vlan 35
VLAN 35 added:
    Name: VLAN0035
sw1(vlan)#vlan 36
VLAN 36 added:
    Name: VLAN0036
sw1(vlan)#vlan 37
VLAN 37 added:
    Name: VLAN0037
sw1(vlan)#vlan 38
VLAN 38 added:
    Name: VLAN0038
sw1(vlan)#vlan 39
VLAN 39 added:
    Name: VLAN0039
sw1(vlan)#exit
APPLY completed.
Exiting....
```

3. Print the vlans in SW1 to verify Vlans were defined

```

EXXITING.....
sw1#show vlan-switch

VLAN Name          Status    Ports
---- -- -- -- --
1   default         active    Fa1/0, Fa1/12, Fa1/13, Fa1/14
                           Fa1/15
2   VLAN0002        active
3   VLAN0003        active
4   VLAN0004        active
5   VLAN0005        active
6   VLAN0006        active
10  VLAN0010        active    Fa1/1, Fa1/2, Fa1/3, Fa1/4
                           Fa1/5
20  VLAN0020        active    Fa1/6, Fa1/7, Fa1/8, Fa1/9
                           Fa1/10
1002 fddi-default   active
1003 token-ring-default active
1004 fddinet-default active
1005 trnet-default   active

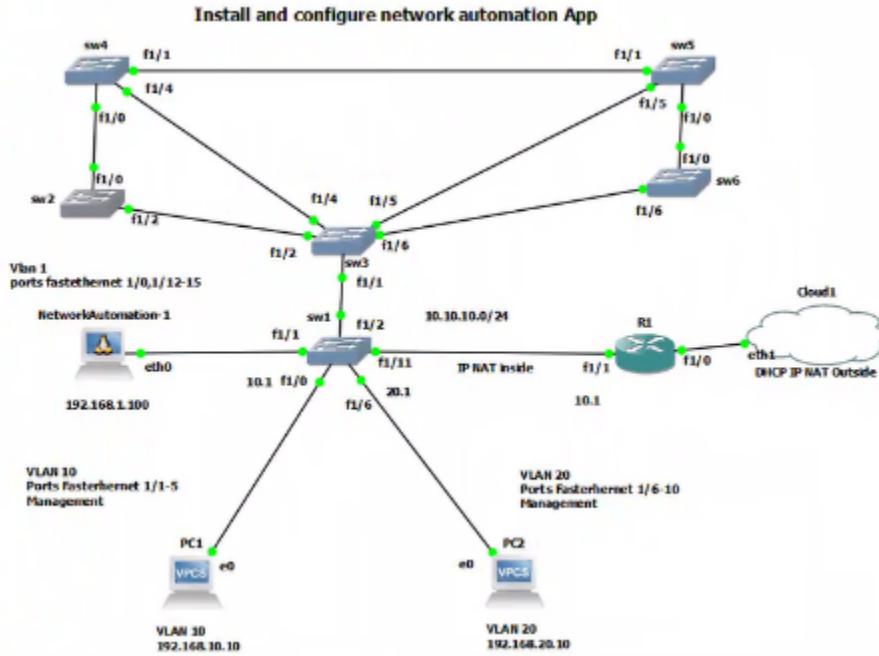
VLAN Type  SAID      MTU  Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
1   enet  100001    1500 -     -     -     -     -     1002  1003
2   enet  100002    1500 -     -     -     -     -     0     0

VLAN Type  SAID      MTU  Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
3   enet  100003    1500 -     -     -     -     -     0     0

```

3.2.2.4 Python for Network Engineers with GNS3 (Part 5) - Multiple switches, multiple VLANs

3.2.2.4.1 Topology



3.2.2.4.2 Addressing Table

NE	Port	Connected To	IP Address	Description
SW1	FastEthernet1/1-5	End devices	N/A	Access ports for VLAN 10
SW1	FastEthernet1/6-10	End devices	N/A	Access ports for VLAN 20
SW1	FastEthernet1/11	R1 (FastEthernet1/1)	10.10.10.2/24	Layer 3 link to Router
SW1	FastEthernet1/2	SW3 (FastEthernet1/1)		
SW1	VLAN1	Default	192.168.1.1/24	Default VLAN
SW1	Vlan10	VLAN 10 devices	192.168.10.1/24	VLAN 10
SW1	Vlan20	VLAN 20 devices	192.168.20.1/24	VLAN 20
SW3	VLAN1	Default	192.168.1.30	Default VLAN
SW3	FastEthernet1/1	SW1 (FastEthernet1/2)		
SW3	FastEthernet1/2	SW2 (FastEthernet1/2)		
SW3	FastEthernet1/4	SW4 FastEthernet1/4		
SW3	FastEthernet1/5	SW5 FastEthernet1/5		
SW3	FastEthernet1/6	SW6 FastEthernet1/6		
SW2	FastEthernet1/2	SW3 (FastEthernet1/2)		
SW2	FastEthernet1/0	SW4 FastEthernet1/0		
SW4	FastEthernet1/0	SW2 (FastEthernet1/0)		
SW4	FastEthernet1/1	SW5 FastEthernet1/1		
SW4	FastEthernet1/4	SW3 FastEthernet1/4		
SW5	FastEthernet1/0	SW6 (FastEthernet1/0)		
SW5	FastEthernet1/1	SW4 FastEthernet1/1		
SW5	FastEthernet1/5	SW3 FastEthernet1/5		
SW6	FastEthernet1/0	SW5 (FastEthernet1/0)		
SW6	FastEthernet1/6	SW3 FastEthernet1/6		
SW6	VLAN1	Default	192.168.1.60	
SW5	VLAN1	Default	192.168.1.50	
SW4	VLAN1	Default	192.168.1.40	
SW2	VLAN1	Default	192.168.1.20	
R1	FastEthernet1/0	Cloud/ISP	DHCP (10.164.0.45)	Connection R1 to cloud (NAT outside)
R1	FastEthernet1/1	SW1 (FastEthernet1/11)	10.10.10.1/24	Connection R1 to SW1 (NAT inside)

PC1	eth0	SW1 (FastEthernet1/1)	IP: 192.168.10.10/24, Gateway: 192.168.10.1	
PC2	eth0	SW1 (FastEthernet1/6)	IP: 192.168.20.10/24, Gateway: 192.168.20.1	
NetworkAutomation	eth0	SW1 (FastEthernet1/11)	IP: 192.168.1.100/24, Gateway:192.168.1.1	

3.2.2.4.3 Scripts

SW1

```
!!!!!!!
!! SW1
!!!!!!!

enable
! VLANs
vlan database
vlan 10
vlan 20
exit

configure terminal
!
! Base Configuration
hostname sw1
no ip domain lookup
! Enable Layer 3 routing
ip routing
!

! Interface Configurations

interface range FastEthernet1/1
description Access ports for VLAN 10
switchport mode access
switchport access vlan 10
!
interface range FastEthernet1/3 - 5
description Access ports for VLAN 10
```

```
switchport mode access
switchport access vlan 10
!
interface FastEthernet1/2
description Link to SW3
switchport access vlan 10
switchport mode trunk
!
interface range FastEthernet1/6 - 10
description Access ports for VLAN 20
switchport mode access
switchport access vlan 20
!
interface FastEthernet1/11
description Layer 3 link to Router
no switchport
ip address 10.10.10.2 255.255.255.0
no shutdown
!
!

!
! VLAN Interfaces
interface Vlan1
description VLAN 1
ip address 192.168.1.1 255.255.255.0
no shutdown
!
interface Vlan10
description VLAN 10
ip address 192.168.10.1 255.255.255.0
no shutdown
!
interface Vlan20
description VLAN 20
ip address 192.168.20.1 255.255.255.0
no shutdown
!
!
! OSPF Configuration
router ospf 10
log-adjacency-changes
network 10.10.10.0 0.0.0.255 area 0
network 192.168.1.0 0.0.0.255 area 0
network 192.168.10.0 0.0.0.255 area 0
network 192.168.20.0 0.0.0.255 area 0
!
line vty 0 4
password cisco123
login
exit
end
```

R1

```
!!!!!!!
!! R1
!!!!!!!
enable
configure terminal
!
hostname R1
!

ip name-server 8.8.8.8
ip domain-lookup
enable password cisco
username student password cisco

! Interface Configurations

interface FastEthernet1/0
description connection R1 to cloud
ip address dhcp
ip nat outside
no shutdown
!
interface FastEthernet1/1
description connection R1 to SW1
ip address 10.10.10.1 255.255.255.0
ip nat inside
no shutdown
!

! OSPF Configuration
router ospf 10
network 10.10.10.0 0.0.0.255 area 0
default-information originate
!
! NAT and Routing
ip nat inside source list 1 interface FastEthernet1/0 overload

!
access-list 1 permit 10.10.10.0 0.0.0.255
access-list 1 permit 192.168.0.0 0.0.255.255
!
line vty 0 4
password cisco
login local
transport input all
exit
```

```
end
```

SW2

```
!!!!!!!
!!! SW2
!!!!!!!

enable
configure terminal
!
hostname SW2
no ip domain-lookup
enable password cisco
username student password cisco
username student privilege 15
!
! VLAN Configuration
interface Vlan1
  description VLAN1
  ip address 192.168.1.20 255.255.255.0
  no shutdown
exit
!
! Trunk Links
interface FastEthernet1/2
  description Link to SW3
  switchport mode trunk
  no shutdown
!
interface FastEthernet1/0
  description Link to SW4
  switchport mode trunk
  no shutdown
!
line vty 0 4
password cisco
login local
transport input all
exit
end
```

SW3

```
!!!!!!!
!!! SW3
```

```
!!!!!!!!

enable
configure terminal
!
hostname SW3
no ip domain-lookup
enable password cisco
username student password cisco
username student privilege 15
!
! VLAN Configuration
interface Vlan1
description Management VLAN
ip address 192.168.1.30 255.255.255.0
no shutdown
exit
!
! Trunk Links
interface FastEthernet1/1
description Link to SW1
switchport mode trunk
no shutdown
!
interface FastEthernet1/2
description Link to SW2
switchport mode trunk
no shutdown
!
interface FastEthernet1/4
description Link to SW4
switchport mode trunk
no shutdown
!
interface FastEthernet1/5
description Link to SW5
switchport mode trunk
no shutdown
!
interface FastEthernet1/6
description Link to SW6
switchport mode trunk
no shutdown
!
line vty 0 4
password cisco
login local
transport input all
exit
end
```

SW4

```
!!!!!!!
!!! SW4
!!!!!!!

enable
configure terminal
!
hostname SW4
no ip domain-lookup
enable password cisco
username student password cisco
username student privilege 15
!
! VLAN Configuration
interface Vlan1
description Management VLAN
ip address 192.168.1.40 255.255.255.0
no shutdown
exit
!
! Trunk Links
interface FastEthernet1/0
description Link to SW2
switchport mode trunk
no shutdown
!
interface FastEthernet1/1
description Link to SW5
switchport mode trunk
no shutdown
!
interface FastEthernet1/4
description Link to SW3
switchport mode trunk
no shutdown
!
line vty 0 4
password cisco
login local
transport input all
exit
end
```

SW5

```
!!!!!!!
!!! SW5
!!!!!!!
enable
configure terminal
!
hostname SW5
no ip domain-lookup
enable password cisco
username student password cisco
username student privilege 15
!
! VLAN Configuration
interface Vlan1
description Management VLAN
ip address 192.168.1.50 255.255.255.0
no shutdown
exit
!
! Trunk Links
interface FastEthernet1/0
description Link to SW6
switchport mode trunk
no shutdown
!
interface FastEthernet1/1
description Link to SW4
switchport mode trunk
no shutdown
!
interface FastEthernet1/5
description Link to SW3
switchport mode trunk
no shutdown
!
line vty 0 4
password cisco
login local
transport input all
exit
end
```

SW6

```
!!!!!!!
!!! SW6
!!!!!!!
```

```

enable
configure terminal
!
hostname SW6
no ip domain-lookup
enable password cisco
username student password cisco
username student privilege 15
!
! VLAN Configuration
interface Vlan1
  description Management VLAN
  ip address 192.168.1.60 255.255.255.0
  no shutdown
exit
!
! Trunk Links
interface FastEthernet1/0
  description Link to SW5
  switchport mode trunk
  no shutdown
!
interface FastEthernet1/6
  description Link to SW3
  switchport mode trunk
  no shutdown
!
line vty 0 4
  password cisco
  login local
  transport input all
exit
end

```

Python script

```

#!/usr/bin/env python3
import getpass
import telnetlib
import time

# Base IP prefix
IP_BASE = "192.168.1."

# Specific last octets for SW1-SW6
HOST_OCTETS = [1, 20, 30, 40, 50, 60]

```

```

# Get credentials
user = input("Enter your telnet username: ")  # Python 3 uses input, not raw_input
password = getpass.getpass()

# Loop through specific IPs
for n in HOST_OCTETS:
    HOST = IP_BASE + str(n)
    print("Telnet to host " + HOST)
    tn = telnetlib.Telnet(HOST)

    tn.read_until(b"Username: ")
    tn.write((user + "\n").encode('ascii'))

    if password:
        tn.read_until(b"Password: ")
        tn.write((password + "\n").encode('ascii'))

    tn.write(b"conf t\n")
    time.sleep(0.5)

    # Create VLANs using vlan database
    tn.write(b"vlan database\n")
    time.sleep(0.5)

    for n in range(30, 36):
        tn.write(("vlan " + str(n) + "\n").encode('ascii'))
        tn.write(("name Python_VLAN_" + str(n) + "\n").encode('ascii'))
        time.sleep(0.1)
        tn.write(b"exit\n")  # Exit VLAN context
        time.sleep(0.1)

    tn.write(b"end\n")
    time.sleep(0.5)
    tn.write(b"exit\n")
    time.sleep(0.5)

    output = tn.read_all().decode('ascii')
    print(f"Output from {HOST}:\n{output}\n")
    tn.close()

print("VLAN configuration completed on all switches.")

```

nano python_multiple_sws.py

```
root@NetworkAutomation-1:~# nano python_multiple_sws.py
```

chmod +x nano python_multiple_sws.py

3.2.2.4.4 Run script

```
root@NetworkAutomation-1:~# ./python_multiple_sws.py
```

```
root@NetworkAutomation-1:~# ./python_multiple_sws.py
Enter your telnet username: student
Password:
Telnet to host 192.168.1.1
Output from 192.168.1.1:

sw1#vlan database
sw1(vlan)#vlan 30
VLAN 30 modified:
sw1(vlan)#vlan 31
VLAN 31 modified:
sw1(vlan)#vlan 32
VLAN 32 modified:
sw1(vlan)#vlan 33
VLAN 33 modified:
sw1(vlan)#vlan 34
VLAN 34 modified:
sw1(vlan)#vlan 35
VLAN 35 modified:
sw1(vlan)#exit
APPLY completed.
Exiting....
sw1#exit
```

```
Telnet to host 192.168.1.20
Output from 192.168.1.20:

SW2#vlan database
SW2(vlan)#vlan 30
VLAN 30 added:
    Name: VLAN0030
SW2(vlan)#vlan 31
VLAN 31 added:
    Name: VLAN0031
SW2(vlan)#vlan 32
VLAN 32 added:
    Name: VLAN0032
SW2(vlan)#vlan 33
VLAN 33 added:
    Name: VLAN0033
SW2(vlan)#vlan 34
VLAN 34 added:
    Name: VLAN0034
SW2(vlan)#vlan 35
VLAN 35 added:
    Name: VLAN0035
SW2(vlan)#exit
APPLY completed.
Exiting....
SW2#exit
```

```
Telnet to host 192.168.1.30
Output from 192.168.1.30:

SW3#vlan database
SW3(vlan)#vlan 30
VLAN 30 added:
    Name: VLAN0030
SW3(vlan)#vlan 31
VLAN 31 added:
    Name: VLAN0031
SW3(vlan)#vlan 32
VLAN 32 added:
    Name: VLAN0032
SW3(vlan)#vlan 33
VLAN 33 added:
    Name: VLAN0033
SW3(vlan)#vlan 34
VLAN 34 added:
    Name: VLAN0034
SW3(vlan)#vlan 35
VLAN 35 added:
    Name: VLAN0035
SW3(vlan)#exit
APPLY completed.
Exiting....
SW3#exit
```

```
Telnet to host 192.168.1.40
Output from 192.168.1.40:

SW4#vlan database
SW4(vlan)#vlan 30
VLAN 30 added:
    Name: VLAN0030
SW4(vlan)#vlan 31
VLAN 31 added:
    Name: VLAN0031
SW4(vlan)#vlan 32
VLAN 32 added:
    Name: VLAN0032
SW4(vlan)#vlan 33
VLAN 33 added:
    Name: VLAN0033
SW4(vlan)#vlan 34
VLAN 34 added:
    Name: VLAN0034
SW4(vlan)#vlan 35
VLAN 35 added:
    Name: VLAN0035
SW4(vlan)#exit
APPLY completed.
Exiting....
SW4#exit
```

```
Telnet to host 192.168.1.50
Output from 192.168.1.50:

SW5#vlan database
SW5(vlan)#vlan 30
VLAN 30 added:
    Name: VLAN0030
SW5(vlan)#vlan 31
VLAN 31 added:
    Name: VLAN0031
SW5(vlan)#vlan 32
VLAN 32 added:
    Name: VLAN0032
SW5(vlan)#vlan 33
VLAN 33 added:
    Name: VLAN0033
SW5(vlan)#vlan 34
VLAN 34 added:
    Name: VLAN0034
SW5(vlan)#vlan 35
VLAN 35 added:
    Name: VLAN0035
SW5(vlan)#exit
APPLY completed.
Exiting...
SW5#exit
```

```
Telnet to host 192.168.1.60
Output from 192.168.1.60:

SW6#vlan database
SW6(vlan)#vlan 30
VLAN 30 added:
  Name: VLAN0030
SW6(vlan)#vlan 31
VLAN 31 added:
  Name: VLAN0031
SW6(vlan)#vlan 32
VLAN 32 added:
  Name: VLAN0032
SW6(vlan)#vlan 33
VLAN 33 added:
  Name: VLAN0033
SW6(vlan)#vlan 34
VLAN 34 added:
  Name: VLAN0034
SW6(vlan)#vlan 35
VLAN 35 added:
  Name: VLAN0035
SW6(vlan)#exit
APPLY completed.
Exiting....
SW6#exit
```

VLAN configuration completed on all switches.

3.2.2.4.5 Verify VLANS are defined

SW1

```
sw1#show vlan-switch

VLAN Name          Status    Ports
---- -----
1    default        active    Fa1/0, Fa1/12, Fa1/13, Fa1/14
                           Fa1/15
2    VLAN0002       active
3    VLAN0003       active
4    VLAN0004       active
5    VLAN0005       active
6    VLAN0006       active
10   VLAN0010       active    Fa1/1, Fa1/3, Fa1/4, Fa1/5
20   VLAN0020       active    Fa1/6, Fa1/7, Fa1/8, Fa1/9
                           Fa1/10
30   VLAN0030       active
31   VLAN0031       active
32   VLAN0032       active
33   VLAN0033       active
34   VLAN0034       active
35   VLAN0035       active
1002 fddi-default  active
1003 token-ring-default  active
1004 fddinet-default  active
1005 trnet-default  active
```

SW2

```
SW2#show vlan-switch

VLAN Name          Status     Ports
----- -----
1    default        active     Fa1/1, Fa1/3, Fa1/4, Fa1/5
                           Fa1/6, Fa1/7, Fa1/8, Fa1/9
                           Fa1/10, Fa1/11, Fa1/12, Fa1/13
                           Fa1/14, Fa1/15
30   VLAN0030       active
31   VLAN0031       active
32   VLAN0032       active
33   VLAN0033       active
34   VLAN0034       active
35   VLAN0035       active
1002 fddi-default  active
1003 token-ring-default  active
1004 fddinet-default  active
1005 trnet-default   active
```

SW3

```
SW3#show vlan-switch

VLAN Name          Status     Ports
----- -----
1    default        active     Fa1/0, Fa1/3, Fa1/7, Fa1/8
                           Fa1/9, Fa1/10, Fa1/11, Fa1/12
                           Fa1/13, Fa1/14, Fa1/15
30   VLAN0030       active
31   VLAN0031       active
32   VLAN0032       active
33   VLAN0033       active
34   VLAN0034       active
35   VLAN0035       active
1002 fddi-default  active
1003 token-ring-default  active
1004 fddinet-default  active
1005 trnet-default   active

VLAN Type  SAID      MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
----- -----
1  static  100001    1500
```

SW4

```
SW4#show vlan-switch
```

VLAN	Name	Status	Ports
1	default	active	Fa1/2, Fa1/3, Fa1/5, Fa1/6 Fa1/7, Fa1/8, Fa1/9, Fa1/10 Fa1/11, Fa1/12, Fa1/13, Fa1/14 Fa1/15
30	VLAN0030	active	
31	VLAN0031	active	
32	VLAN0032	active	
33	VLAN0033	active	
34	VLAN0034	active	
35	VLAN0035	active	
1002	fdci-default	active	
1003	token-ring-default	active	
1004	fdinnet-default	active	
1005	trnet-default	active	

VLAN	Type	SAID	MTU	Parent	RingNo	BridgeNo	Stp	BrdgMode	Trans1	Trans2
1	enet	100001	1500	-	-	-	-	1002	1003	

SW5

```
SW5#show vlan-switch
```

VLAN	Name	Status	Ports
1	default	active	Fa1/2, Fa1/3, Fa1/4, Fa1/6 Fa1/7, Fa1/8, Fa1/9, Fa1/10 Fa1/11, Fa1/12, Fa1/13, Fa1/14 Fa1/15
30	VLAN0030	active	
31	VLAN0031	active	
32	VLAN0032	active	
33	VLAN0033	active	
34	VLAN0034	active	
35	VLAN0035	active	
1002	fdci-default	active	
1003	token-ring-default	active	
1004	fdinnet-default	active	
1005	trnet-default	active	

VLAN	Type	SAID	MTU	Parent	RingNo	BridgeNo	Stp	BrdgMode	Trans1	Trans2
1	enet	100001	1500	-	-	-	-	1002	1003	
30	enet	100030	1500	-	-	-	-	0	0	
31	enet	100031	1500	-	-	-	-	0	0	

solarwinds | Solar-PUTTY (vnc sess)

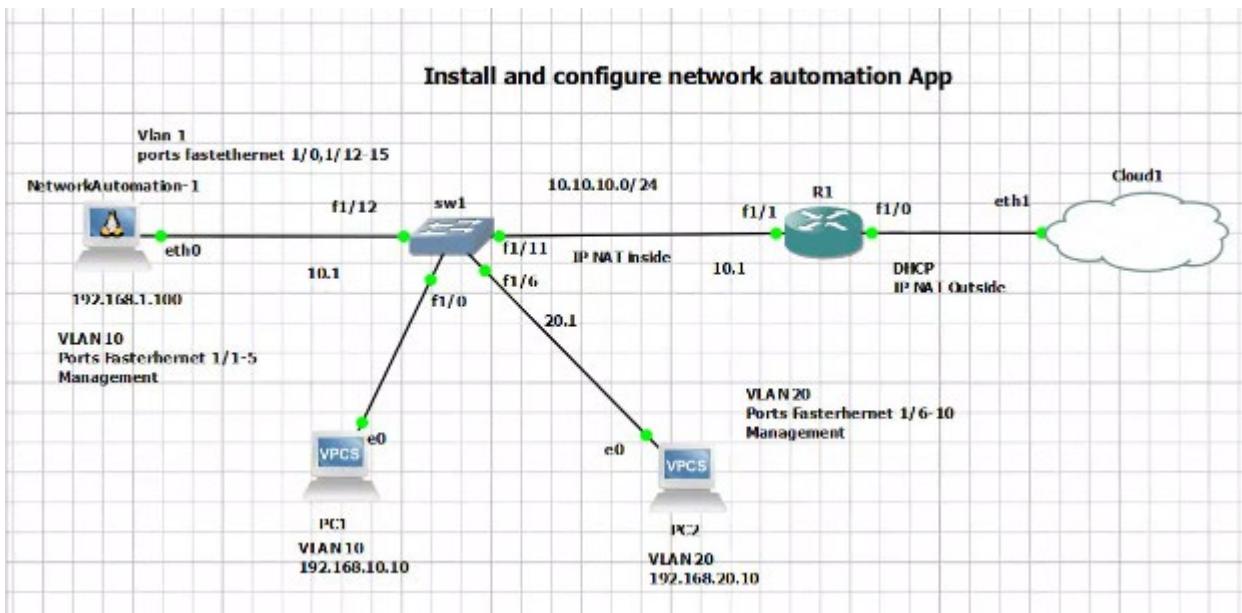
SW6

VLAN Name	Status	Ports
1 default	active	Fa1/1, Fa1/2, Fa1/3, Fa1/4 Fa1/5, Fa1/7, Fa1/8, Fa1/9 Fa1/10, Fa1/11, Fa1/12, Fa1/13 Fa1/14, Fa1/15
30 VLAN0030	active	
31 VLAN0031	active	
32 VLAN0032	active	
33 VLAN0033	active	
34 VLAN0034	active	
35 VLAN0035	active	
1002 fddi-default	active	
1003 token-ring-default	active	
1004 fddinet-default	active	
1005 trnet-default	active	
VLAN Type SAID	MTU	Parent RingNo BridgeNo Stp BrdgMode Trans1 Trans2

3.2.2.5 Final Python automation project Create vlans loops

3.2.2.5.1 Configure GNS3 project

Topology



Configuration

Network Automation

Set ip address

SC Configure
Console
Audrey console
Start
Suspend
Stop
Reload
Custom config
Change hostname
Change ipaddr
Duplicate
Show network information
Show in this manager
Bring to front
Import config
Export config
Edit config
Save one layer
Lower one layer
Lock item
Delete

```
#  
# Static config for eth0  
auto eth0  
iface eth0 inet static  
    address 192.168.1.100  
    netmask 255.255.255.0  
    gateway 192.168.1.1  
    up echo nameserver 8.8.8.8 > /etc/resolv.conf  
  
# DHCP config for eth0  
# auto eth0  
# iface eth0 inet dhcp
```

```
# Static config for eth0  
auto eth0  
iface eth0 inet static  
    address 192.168.1.100  
    netmask 255.255.255.0  
    gateway 192.168.1.1  
    up echo nameserver 8.8.8.8 > /etc/resolv.conf
```

PC1

```
ip 192.168.10.10 255.255.255.0 192.168.10.1
```

```
save
```

PC2

```
ip 192.168.20.10 255.255.255.0 192.168.20.1
```

```
save
```

```
!!!!!!  
!!! SW1  
!!!!!!  
enable  
! VLANs  
vlan database
```

```
vlan 10
vlan 20
exit

configure terminal
!
! Base Configuration
hostname sw1
enable password cisco
username student privilege 15 password 0 cisco
no ip domain lookup

! Enable Layer 3 routing
ip routing

!
! Interface configurations
interface FastEthernet0/0
no ip address
shutdown

interface FastEthernet0/1
no ip address
shutdown

! VLAN 10 interfaces (Fa1/0 to Fa1/5)
interface range FastEthernet1/0 - 5
description Access ports for VLAN 10
switchport access vlan 10

! VLAN 20 interfaces (Fa1/6 to Fa1/10)
interface range FastEthernet1/6 - 10
description Access ports for VLAN 20
switchport access vlan 20

interface FastEthernet1/11
description Layer 3 link to Router
no switchport
ip address 10.10.10.2 255.255.255.0

! VLAN interfaces
interface Vlan1
description VLAN 1
ip address 192.168.1.1 255.255.255.0

interface Vlan10
description VLAN 10
ip address 192.168.10.1 255.255.255.0
```

```
interface Vlan20
description VLAN 20
ip address 192.168.20.1 255.255.255.0

! OSPF configuration
router ospf 10
log-adjacency-changes
network 10.10.10.0 0.0.0.255 area 0
network 192.168.1.0 0.0.0.255 area 0
network 192.168.10.0 0.0.0.255 area 0
network 192.168.20.0 0.0.0.255 area 0

! Line configurations
line con 0
exec-timeout 0 0
privilege level 15
logging synchronous

line aux 0
exec-timeout 0 0
privilege level 15
logging synchronous

line vty 0 4
login local
transport input all

end
```

```
!!!!!!!
!!! R1
!!!!!!!

enable
configure terminal

! Basic system settings
hostname R1
enable password cisco
username student privilege 15 password 0 cisco
no ip domain lookup
ip name-server 8.8.8.8

! Interface configurations
```

```
interface FastEthernet0/0
no ip address
shutdown
duplex full

interface FastEthernet1/0
description connection R1 to cloud
ip address dhcp
ip nat outside
speed auto
duplex auto

interface FastEthernet1/1
description connection R1 to SW1
ip address 10.10.10.1 255.255.255.0
ip nat inside
speed auto
duplex auto

interface Serial2/0
no ip address
shutdown
serial restart-delay 0

interface Serial2/1
no ip address
shutdown
serial restart-delay 0

interface Serial2/2
no ip address
shutdown
serial restart-delay 0

interface Serial2/3
no ip address
shutdown
serial restart-delay 0

! OSPF configuration
router ospf 10
network 10.10.10.0 0.0.0.255 area 0
default-information originate

! NAT configuration
ip nat inside source list 1 interface FastEthernet1/0 overload
access-list 1 permit 10.10.10.0 0.0.0.255
access-list 1 permit 192.168.0.0 0.0.255.255
```

```
! Additional settings
no ip http server
no ip http secure-server
ip forward-protocol nd

! Line configurations
line con 0
exec-timeout 0 0
privilege level 15
logging synchronous
stopbits 1

line aux 0
exec-timeout 0 0
privilege level 15
logging synchronous
stopbits 1

line vty 0 4
login local
transport input all

end
```

Connectivity test

PC1

Ping default gateway

PC1> ping 192.168.10.1

Ping PC2

PC1> ping 192.168.20.10

Ping Network automation

PC1> ping 192.168.1.100

Ping DNS

PC1> ping 8.8.8.8

3.2.2.5.2 Python script

1. Create script

```
nano pythonS1_vlan_loop_1sw.py
```

```
pythonS1_vlan_loop_1sw.py
root@NetworkAutomation-1:~# cat pythonS1_vlan_loop_1sw.py
#!/usr/bin/env python3
```

Script

```
#!/usr/bin/env python3
import getpass
import telnetlib
import time

HOST = "192.168.1.1" # SW1

# Get credentials
user = input("Enter your telnet username: ")
password = getpass.getpass("Enter your telnet password: ")

# Connect to SW1
print(f"Connecting to {HOST}...")
tn = telnetlib.Telnet(HOST, timeout=10)

# Authenticate
tn.read_until(b"Username: ") # Wait for username prompt
tn.write((user + "\n").encode('ascii'))
time.sleep(0.5) # Small delay to ensure password prompt appears

if password:
    tn.read_until(b"Password: ") # Wait for password prompt
    tn.write((password + "\n").encode('ascii'))
    time.sleep(0.5) # Small delay to ensure enable prompt appears

# Create VLANs using vlan database
tn.write(b"vlan database\n")
time.sleep(0.5)

# Create VLANs 30 to 39 with names
for n in range(30, 40):
    vlan_name = f"python_vlan_{n}" # Example: python_vlan_30, python_vlan_31, etc.
    vlan_command = f"vlan {n} name {vlan_name}\n".encode('ascii')
    tn.write(vlan_command)
    time.sleep(0.1) # Small delay between VLAN creations

# Exit vlan database and session
tn.write(b"exit\n") # Exit vlan database mode
time.sleep(0.5)
```

```
tn.write(b"exit\n") # Exit session  
  
# Print output  
print(tn.read_all().decode('ascii'))
```

2. Give execution permissions to script

```
chmod +x pythonS1_vlan_loop
```

```
root@NetworkAutomation-1:~# chmod +x pythonS1_vlan_loop_1sw.py
```

3. Test telnet connection manually

```
root@NetworkAutomation-1:~# telnet 192.168.1.1  
Trying 192.168.1.1...  
Connected to 192.168.1.1.  
Escape character is '^]'.  
  
User Access Verification  
  
Username: student  
Password:  
sw1#enable  
sw1#exit  
Connection closed by foreign host.
```

4. Test script

Use user student/cisco

```
root@NetworkAutomation-1:~# ./pythonS1_vlan_loop_1sw.py  
Enter user name: student
```

```
root@NetworkAutomation-1:~# ./pythons1_vlan_loop_1sw.py
Enter your telnet username: student
Enter your telnet password:
Connecting to 192.168.1.1...

sw1#vlan database
sw1(vlan)#vlan 30 name python_vlan_30
VLAN 30 added:
  Name: python_vlan_30
sw1(vlan)#vlan 31 name python_vlan_31
VLAN 31 added:
  Name: python_vlan_31
sw1(vlan)#vlan 32 name python_vlan_32
VLAN 32 added:
  Name: python_vlan_32
sw1(vlan)#vlan 33 name python_vlan_33
VLAN 33 added:
  Name: python_vlan_33
sw1(vlan)#vlan 34 name python_vlan_34
VLAN 34 added:
  Name: python_vlan_34
sw1(vlan)#vlan 35 name python_vlan_35
VLAN 35 added:
  Name: python_vlan_35
sw1(vlan)#vlan 36 name python_vlan_36
VLAN 36 added:
  Name: python_vlan_36
sw1(vlan)#vlan 37 name python_vlan_37
VLAN 37 added:
  Name: python_vlan_37
sw1(vlan)#vlan 38 name python_vlan_38
VLAN 38 added:
  Name: python_vlan_38
sw1(vlan)#vlan 39 name python_vlan_39
VLAN 39 added:
  Name: python_vlan_39
sw1(vlan)#exit
APPLY completed.
Exiting....
sw1#exit

root@NetworkAutomation-1:~#
```

5. Verify vlans are created in SW1

```
SW1#show vlan-switch

VLAN Name                               Status    Ports
---- -----
1   default                             active    Fa1/12, Fa1/13, Fa1/14, Fa1/15
10  VLAN0010                           active    Fa1/0, Fa1/1, Fa1/2, Fa1/3
                                         Fa1/4, Fa1/5
20  VLAN0020                           active    Fa1/6, Fa1/7, Fa1/8, Fa1/9
                                         Fa1/10
30  python_vlan_30                     active
31  python_vlan_31                     active
32  python_vlan_32                     active
33  python_vlan_33                     active
34  python_vlan_34                     active
35  python_vlan_35                     active
36  python_vlan_36                     active
37  python_vlan_37                     active
38  python_vlan_38                     active
39  python_vlan_39                     active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                 active
1005 trnet-default                   active

VLAN Type    SAID      MTU    Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
```