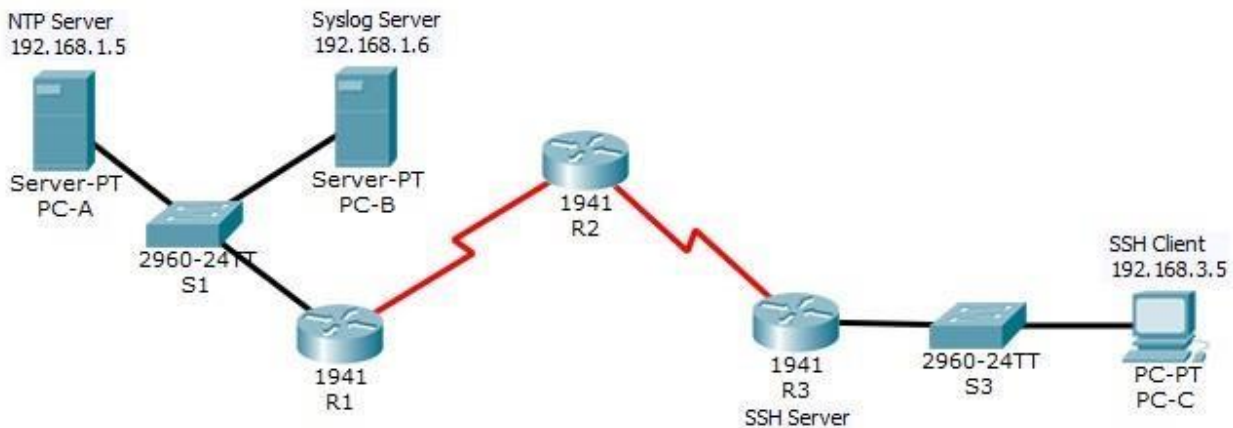


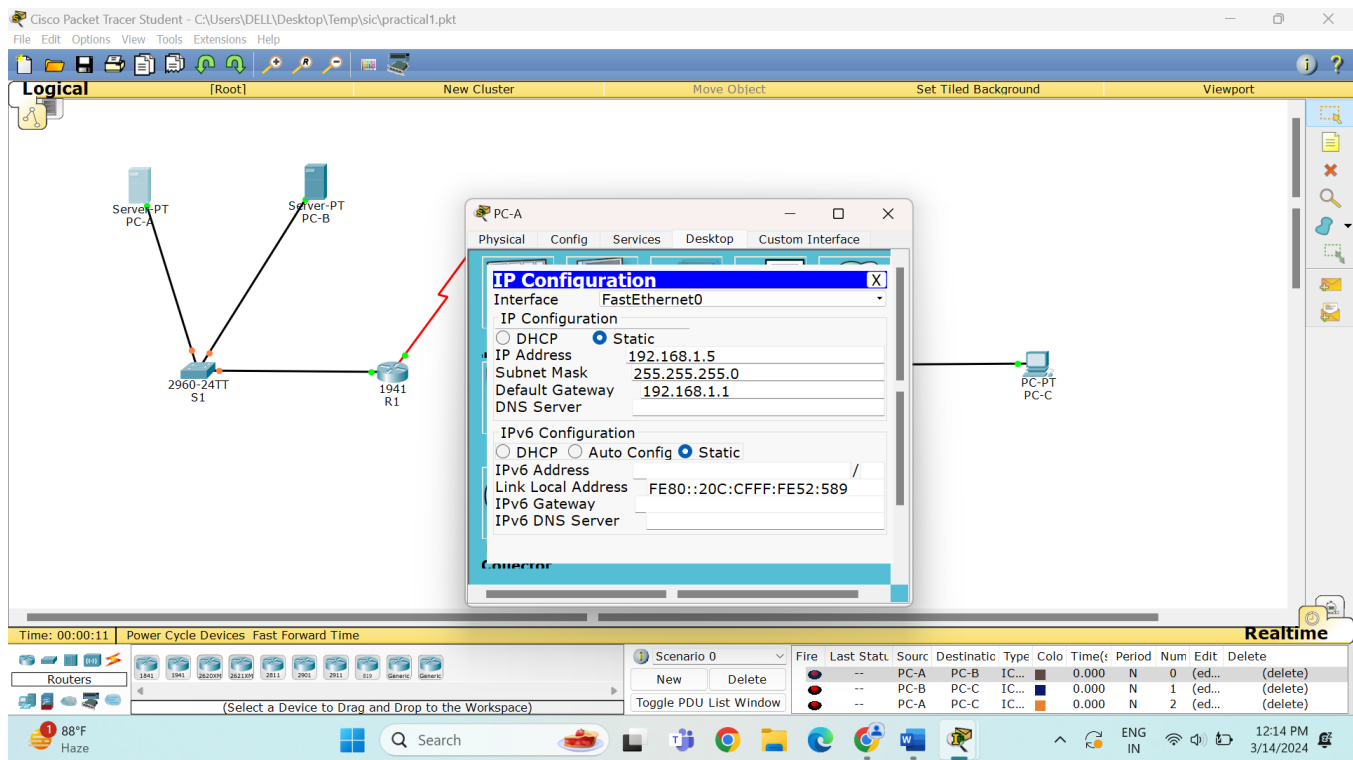
Practical 1: Packet Tracer - Configure Cisco Routers for Syslog, NTP, and SSH Operations

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway	Switch Port
R1	G0/1	192.168.1.1	255.255.255.0	N/A	S1 F0/5
	S0/0/0 (DCE)	10.1.1.1		N/A	N/A
R2	S0/0/0	10.1.1.2	255.255.255.252	N/A	N/A
	S0/0/1 (DCE)	10.2.2.2	255.255.255.252	N/A	N/A
R3	G0/1	192.168.3.1	255.255.255.0	N/A	S3 F0/5
	S0/0/1	10.2.2.1	255.255.255.252	N/A	N/A
PC-A	NIC	192.168.1.5	255.255.255.0	192.168.1.1	S1 F0/6
PC-B	NIC	192.168.1.6	255.255.255.0	192.168.1.1	S2 F0/18
PC-C	NIC	192.168.3.5	255.255.255.0	192.168.3.1	S3 F0/18



Objectives

- Configure OSPF MD5 authentication.
- Configure NTP.
- Configure routers to log messages to the syslog server.
- Configure R3 to support SSH connections.

Background / Scenario

In this activity, you will configure OSPF MD5 authentication for secure routing updates.

The NTP Server is the master NTP server in this activity. You will configure authentication on the NTP server and the routers. You will configure the routers to allow the software clock to be synchronized by NTP to the

time server. Also, you will configure the routers to periodically update the hardware clock with the time learned from NTP.

The Syslog Server will provide message logging in this activity. You will configure the routers to identify the remote host (Syslog server) that will receive logging messages.

You will need to configure timestamp service for logging on the routers. Displaying the correct time and date in Syslog messages is vital when using Syslog to monitor a network.

You will configure R3 to be managed securely using SSH instead of Telnet. The servers have been preconfigured for NTP and Syslog services respectively. NTP will not require authentication. The routers have been pre-configured with the following passwords:

- Enable password: **ciscoenpa55**
- Password for vty lines: **ciscovtypa55**

Note: Note: MD5 is the strongest encryption supported in the version of Packet Tracer used to develop this activity

(v6.2). Although MD5 has known vulnerabilities, you should use the encryption that meets the security requirements of your organization. In this activity, the security requirement specifies MD5.

Part 1: Configure OSPF MD5 Authentication

Step 1: Test connectivity. All devices should be able to ping all other IP addresses.

Step 2: Configure OSPF MD5 authentication for all the routers in area 0.

Configure OSPF MD5 authentication for all the routers in area 0.

```
R1 (config) # router ospf 1
R1 (config-router) # area 0 authentication message-digest
R2 (config) # router ospf 1
R2 (config-router) # area 0 authentication message-digest
R3 (config) # router ospf 1
R3 (config-router) # area 0 authentication message-digest
```

Step 3: Configure the MD5 key for all the routers in area 0.

Configure an MD5 key on the serial interfaces on **R1**, **R2** and **R3**. Use the password **MD5pa55** for key 1.

```
R1 (config) # interface s0/0/0
R1 (config-if) # ip ospf message-digest-key 1 md5 MD5pa55

R2 (config) # interface s0/0/0
R2 (config-if) # ip ospf message-digest-key 1 md5 MD5pa55
R2 (config-if) # interface s0/0/1
R2 (config-if) # ip ospf message-digest-key 1 md5 MD5pa55

R3 (config) # interface s0/0/1
R3 (config-if) # ip ospf message-digest-key 1 md5 MD5pa55
```

Step 4: Verify configurations.

- Verify the MD5 authentication configurations using the commands **show ip ospf interface**.
- Verify end-to-end connectivity.

Part 2: Configure NTP

Step 1: Enable NTP authentication on PC-A.

- On **PC-A**, click **NTP** under the Services tab to verify NTP service is enabled.
- To configure NTP authentication, click **Enable** under Authentication. Use key **1** and password **NTPpa55** for authentication.

Step 2: Configure R1, R2, and R3 as NTP clients.

```
R1 (config) # ntp server 192.168.1.5
R2 (config) # ntp server 192.168.1.5
R3 (config) # ntp server 192.168.1.5
```

Verify client configuration using the command **show ntp status**.

Step 3: Configure routers to update hardware clock.

Configure R1, R2, and R3 to periodically update the hardware clock with the time learned from NTP.

```
R1 (config) # ntp update-calendar
```

```
R2(config)# ntp update-calendar
```

```
R3(config)# ntp update-calendar
```

Exit global configuration and verify that the hardware clock was updated using the command `show clock`.

Step 4: Configure NTP authentication on the routers. Configure NTP

authentication on **R1**, **R2**, and **R3** using key 1 and password **NTPpa55**.

```
R1(config)# ntp authenticate
```

```
R1(config)# ntp trusted-key 1
```

```
R1(config)# ntp authentication-key 1 md5 NTPpa55
```

```
R2(config)# ntp authenticate
```

```
R2(config)# ntp trusted-key 1
```

```
R2(config)# ntp authentication-key 1 md5 NTPpa55
```

```
R3(config)# ntp authenticate
```

```
R3(config)# ntp trusted-key 1
```

```
R3(config)# ntp authentication-key 1 md5 NTPpa55
```

Step 5: Configure routers to timestamp log messages. Configure timestamp service for logging on the routers.

```
R1(config)# service timestamps log datetime msec
```

```
R2(config)# service timestamps log datetime msec
```

```
R3(config)# service timestamps log datetime msec
```

Part 3: Configure Routers to Log Messages to the Syslog Server

Step 1: Configure the routers to identify the remote host (Syslog Server) that will receive logging messages.

```
R1(config)# logging host 192.168.1.6
```

```
R2(config)# logging host 192.168.1.6
```

```
R3(config)# logging host 192.168.1.6
```

The router console will display a message that logging has started.

Step 2: Verify logging configuration.

Use the command **show logging** to verify logging has been enabled.

Step 3: Examine logs of the Syslog Server.

From the **Services** tab of the **Syslog Server**'s dialogue box, select the **Syslog** services button. Observe the logging messages received from the routers.

Note: Log messages can be generated on the server by executing commands on the router. For example, entering and exiting global configuration mode will generate an informational configuration message. You may need to click a different service and then click **Syslog** again to refresh the message display.

Part 4: Configure R3 to Support SSH Connections

Step 1: Configure a domain name.

Configure a domain name of ccnasecurity.com on R3.

```
R3(config) # ip domain-name ccnasecurity.com
```

Step 2: Configure users for login to the SSH server on R3.

Create a user ID of SSHadmin with the highest possible privilege level and a secret password of **ciscosshpa55**.

```
R3(config) # username SSHadmin privilege 15 secret ciscosshpa55
```

Step 3: Configure the incoming vty lines on R3. Use the local user accounts for mandatory login and validation. Accept only SSH connections.

```
R3(config) # line vty 0 4
R3(config-line) # login local R3(config-line) #
transport input ssh
```

Step 4: Erase existing key pairs on R3. Any existing RSA

key pairs should be erased on the router.

```
R3(config) # crypto key zeroize rsa
```

Note: If no keys exist, you might receive this message: % No Signature RSA Keys found in configuration.

Step 5: Generate the RSA encryption key pair for R3.

The router uses the RSA key pair for authentication and encryption of transmitted SSH data. Configure the RSA keys with a modulus of **1024**. The default is 512, and the range is from 360 to 2048.

```
R3(config) # crypto key generate rsa
The name for the keys will be: R3.ccnasecurity.com
Choose the size of the key modulus in the range of 360 to 2048 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take a
few minutes.
```

```
How many bits in the modulus [512]: 1024
```

```
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]
```

Note: The command to generate RSA encryption key pairs for **R3** in Packet Tracer differs from those used in the lab.

Step 6: Verify the SSH configuration.

Use the **show ip ssh** command to see the current settings. Verify that the authentication timeout and retries are at their default values of 120 and 3.

Step 7: Configure SSH timeouts and authentication parameters.

The default SSH timeouts and authentication parameters can be altered to be more restrictive. Set the timeout to **90** seconds, the number of authentication retries to **2**, and the version to **2**.

```
R3(config)# ip ssh time-out 90
R3(config)# ip ssh authentication-retries 2
R3(config)# ip ssh version 2
```

Issue the **show ip ssh** command again to confirm that the values have been changed.

Step 8: Attempt to connect to R3 via Telnet from PC-C.

Open the Desktop of **PC-C**. Select the Command Prompt icon. From **PC-C**, enter the command to connect to **R3** via Telnet.

```
PC> telnet 192.168.3.1
```

This connection should fail because **R3** has been configured to accept only SSH connections on the virtual terminal lines.

Step 9: Connect to R3 using SSH on PC-C.

Open the Desktop of **PC-C**. Select the Command Prompt icon. From **PC-C**, enter the command to connect to **R3** via SSH. When prompted for the password, enter the password configured for the administrator **ciscosshpa55**.

```
PC> ssh -l SSHAdmin 192.168.3.1
```

Step 10: Connect to R3 using SSH on R2.

To troubleshoot and maintain **R3**, the administrator at the ISP must use SSH to access the router CLI. From the CLI of **R2**, enter the command to connect to **R3** via SSH version 2 using the **SSHAdmin** user account.

When prompted for the password, enter the password configured for the administrator: **ciscosshpa55**.

```
R2# ssh -v 2 -l SSHAdmin 10.2.2.1
```

Step 11: Check results.

Your completion percentage should be 100%. Click **Check Results** to view the feedback and verification of which required components have been completed.

The screenshot shows the Cisco Packet Tracer interface. The network topology includes two servers (Server-PT PC-A and Server-PT PC-B) connected to a switch (2960-24TT S1), which is connected to a router (1941 R1). R1 is connected to another router (1941 R2), which is connected to a third router (1941 R3). R3 is connected to a PC (PC-C). The CLI window for R3 is open, showing the following commands and output:

```

R3>en
R3#conf t
Enter configuration commands, one per line. End with
CTRL/Z.
R3(config)#router ospf q
% Invalid input detected at '^' marker.

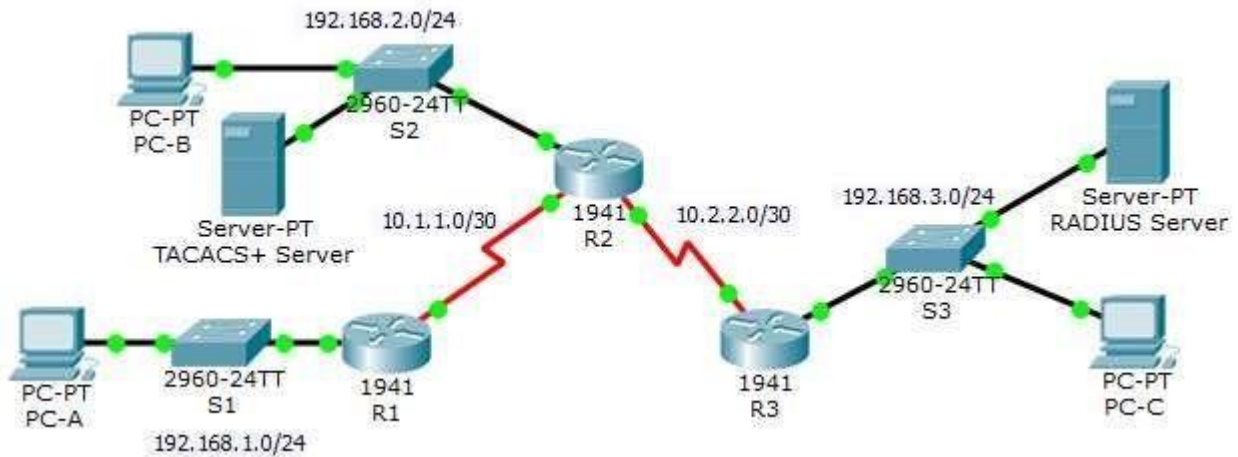
R3(config)#router ospf 1
R3(config-router)#network 192.168.3.0 0.0.0.255 area
0
R3(config-router)#network 10.2.2.0 0.255.255.255 area
0
R3(config-router)#
  
```

The bottom status bar shows the time as 00:05:21, the power cycle devices button, and the fast forward time button. The Realtime section shows a table of connections:

Fire	Last Stat.	Source	Destination	Type	Color	Time(s)	Period	Num	Edit	Delete
---	---	PC-A	PC-B	IC...	---	0.000	N	0	(ed...)	(delete)
---	---	PC-B	PC-C	IC...	---	0.000	N	1	(ed...)	(delete)
---	---	PC-A	PC-C	IC...	---	0.000	N	2	(ed...)	(delete)

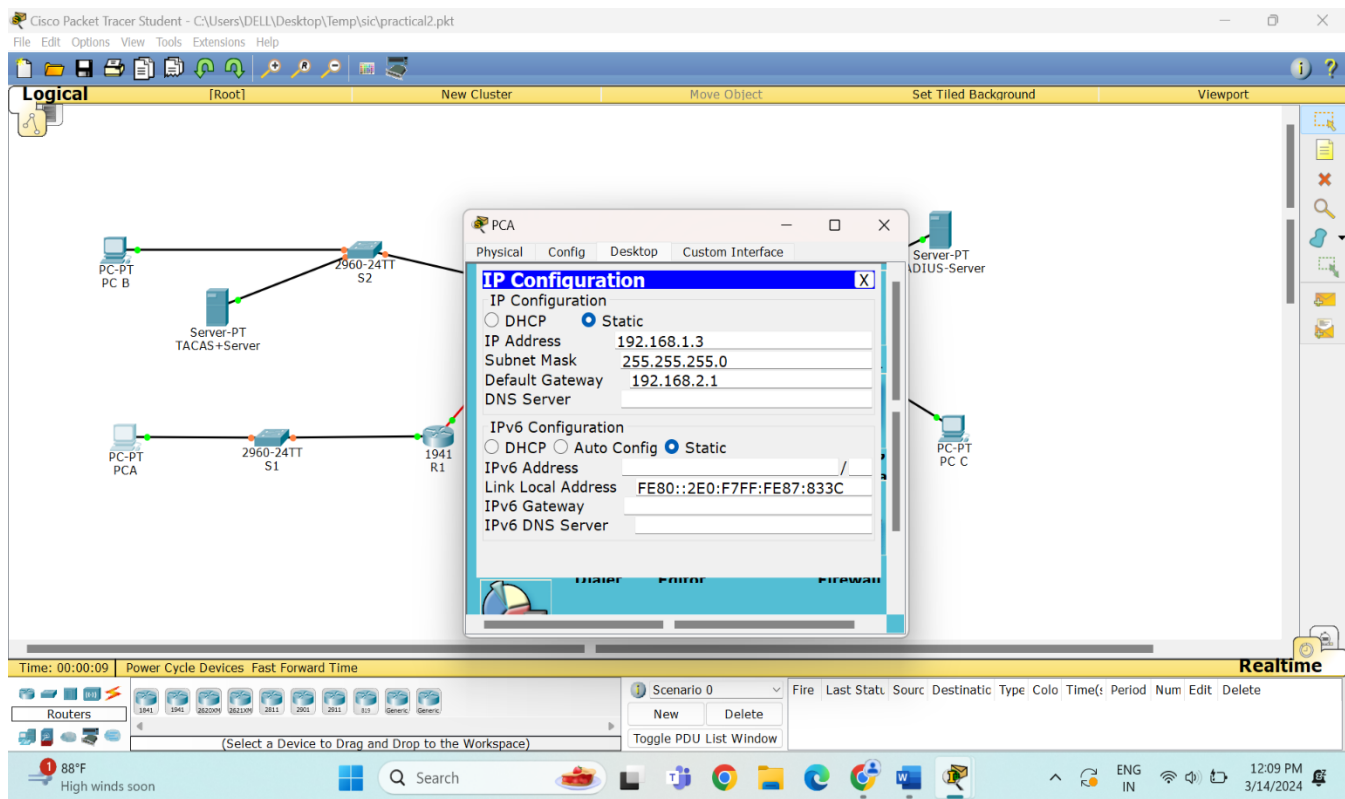
Practical 2: Packet Tracer - Configure AAA Authentication on Cisco Routers

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway	Switch Port
R1	G0/1		255.255.255.0	N/A	S1 F0/1
	S0/0/0 (DCE)	10.1.1.2	255.255.255.252	N/A	N/A
R2	G0/0	192.168.2.1	255.255.255.0	N/A	S2 F0/2
	S0/0/0	10.1.1.1	255.255.255.252	N/A	N/A
	S0/0/1 (DCE)	10.2.2.1	255.255.255.252	N/A	N/A
R3	G0/1	192.168.3.1	255.255.255.0	N/A	S3 F0/5
	S0/0/1	10.2.2.2	255.255.255.252	N/A	N/A
TACACS+ Server	NIC	192.168.2.2	255.255.255.0	192.168.2.1	S2 F0/6
RADIUS Server	NIC	192.168.3.2	255.255.255.0	192.168.3.1	S3 F0/1
PC-A	NIC	192.168.1.3	255.255.255.0	192.168.1.1	S1 F0/2
PC-B	NIC	192.168.2.3	255.255.255.0	192.168.2.1	S2 F0/1
PC-C	NIC	192.168.3.3	255.255.255.0	192.168.3.1	S3 F0/18



Objectives

- Configure a local user account on R1 and configure authenticate on the console and vty lines using local AAA.
- Verify local AAA authentication from the R1 console and the PC-A client.
- Configure server-based AAA authentication using TACACS+.
- Verify server-based AAA authentication from the PC-B client.
- Configure server-based AAA authentication using RADIUS.
- Verify server-based AAA authentication from the PC-C client.

Background / Scenario

The network topology shows routers R1, R2 and R3. Currently, all administrative security is based on knowledge of the enable secret password. Your task is to configure and test local and server-based AAA solutions.

You will create a local user account and configure local AAA on router R1 to test the console and vty logins. ○

User account: **Admin1** and password **admin1pa55**

You will then configure router R2 to support server-based authentication using the TACACS+ protocol. The TACACS+ server has been pre-configured with the following:

Configure AAA Authentication on Cisco Routers

- Client: **R2** using the keyword **tacacspa55**
 - User account: **Admin2** and password **admin2pa55**

Finally, you will configure router R3 to support server-based authentication using the RADIUS protocol.

The RADIUS server has been pre-configured with the following: equipment.

Part 1: Configure Local AAA Authentication for Console Access on R1

Step 1: Test connectivity.

- Ping from **PC-A** to **PC-B**.
 - Ping from **PC-B** to **PC-C**.
 -
- Client: **R3** using the keyword **radiuspa55** ○ User account: **Admin3** and password **admin3pa55**

The routers have also been pre-configured with the following:

- Enable secret password: **ciscoenpa55**
- OSPF routing protocol with MD5 authentication using password: **MD5pa55**

Note: The console and vty lines have not been pre-configured.

Note: IOS version 15.3 uses SCRYPT as a secure encryption hashing algorithm; however, the IOS version that is currently supported in Packet Tracer uses MD5. Always use the most secure option available on your

Step 2: Configure a local username on R1.

Configure a username of **Admin1** with a secret password of **admin1pa55**.

```
R1(config)# username Admin1 secret admin1pa55
```

Step 3: Configure local AAA authentication for console access on R1.

Enable AAA on R1 and configure AAA authentication for the console login to use the local database.

```
R1(config)# aaa new-model
```

```
R1(config)# aaa authentication login default local
```

Step 4: Configure the line console to use the defined AAA authentication method.

Enable AAA on R1 and configure AAA authentication for the console login to use the default method list.

```
R1(config)# line console 0
```

```
R1(config-line)# login authentication default
```

Step 5: Verify the AAA authentication method.

Verify the user EXEC login using the local database.

```
R1(config-line)# end
```

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

```
R1# exit
```

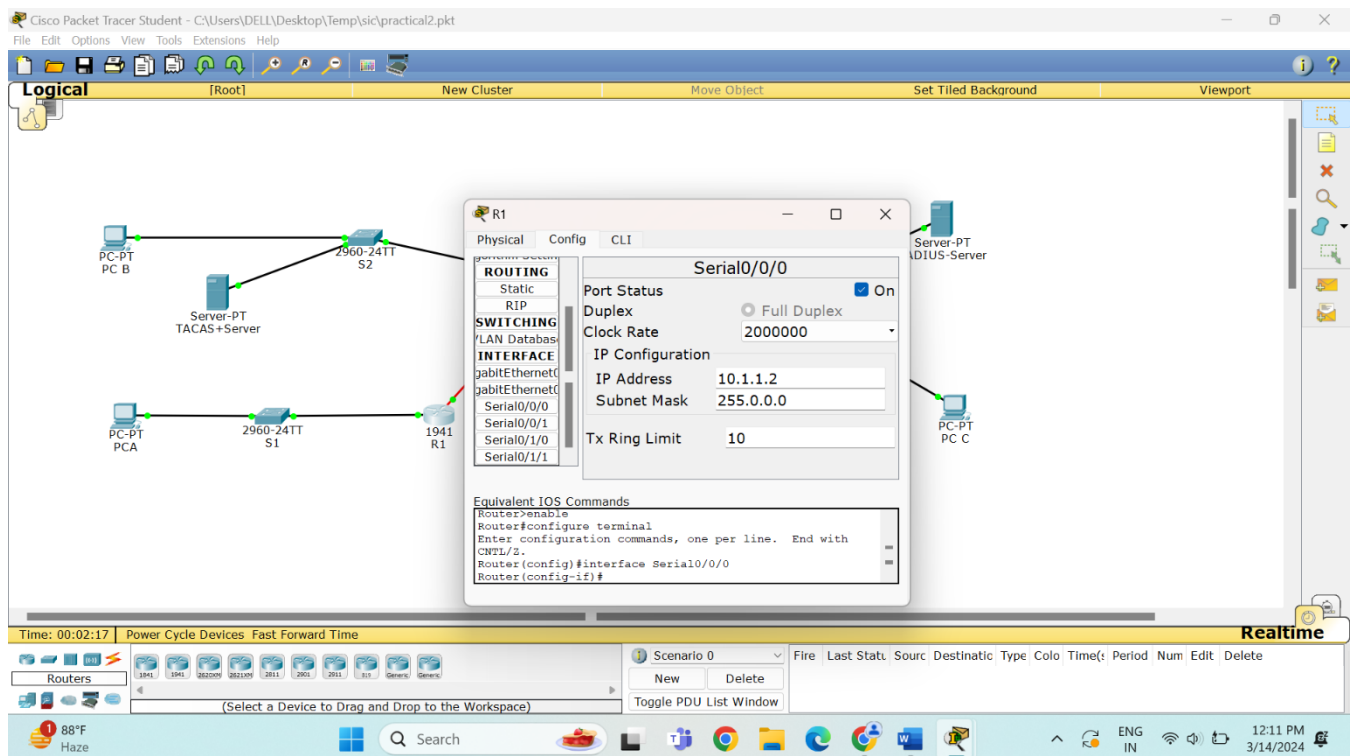
```
R1 con0 is now available Press RETURN
```

```
***** AUTHORIZED ACCESS ONLY *****  
UNAUTHORIZED ACCESS TO THIS DEVICE IS PROHIBITED.
```

User Access Verification

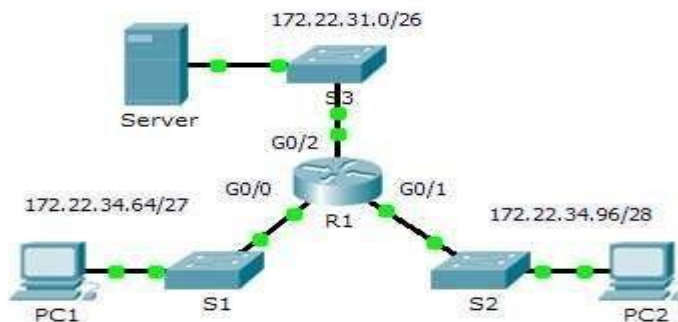
Username: **Adim1**

Password: **admin1pas55**



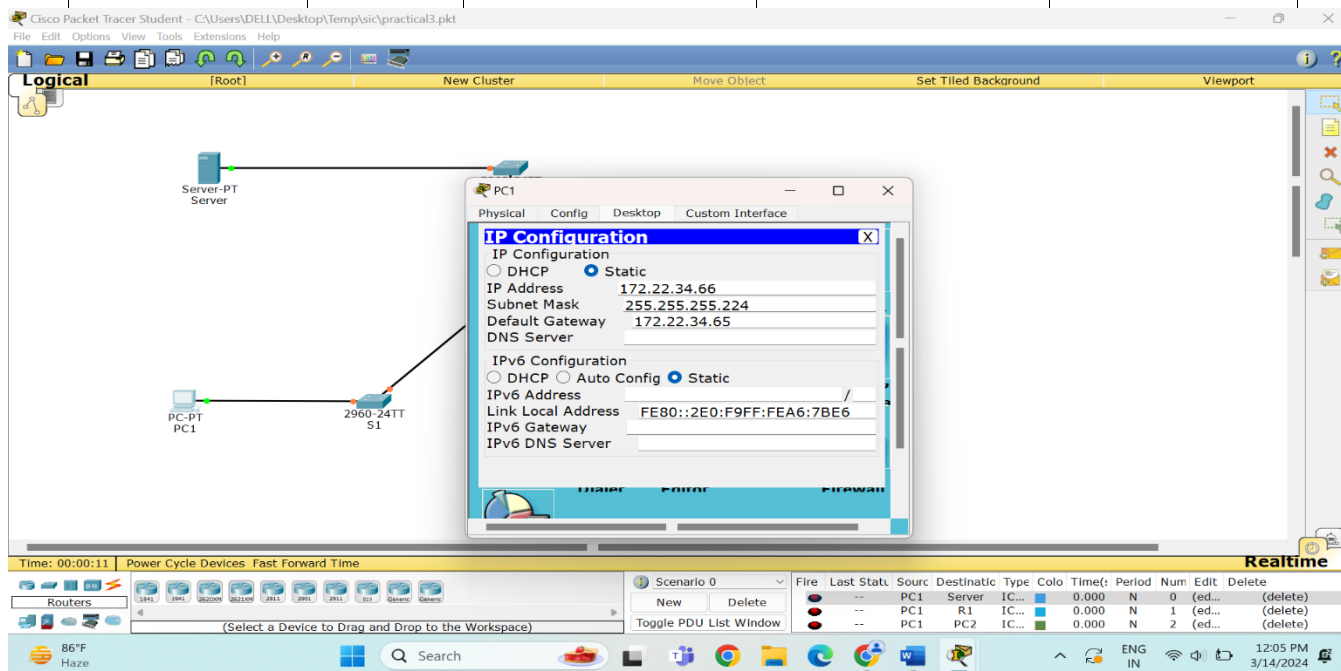
Practical 3: Configuring Extended ACLs - Scenario 1

Topology



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	N/A
	G0/2	172.22.34.1	255.255.255.192	N/A
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



Objectives

Part 1: Configure, Apply and Verify an Extended Numbered ACL

Part 2: Configure, Apply and Verify an Extended Named ACL

Background / Scenario

Two employees need access to services provided by the server. **PC1** needs only FTP access while **PC2** needs only web access. Both computers are able to ping the server, but not each other.

Part 1: Configure, Apply and Verify an Extended Numbered ACL

Step 1: Configure an ACL to permit FTP and ICMP.

- a. 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
- 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
- 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
- d. This ACL permits FTP and ICMP. ICMP is listed above, but FTP is not, because FTP uses TCP. Therefore, 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
- e. Notice that we could filter just for **PC1** by using the **host** keyword or we could allow **any** host. In this case, any device is allowed that has an address belonging to the 172.22.34.64/27 network. Enter the 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
- f. Calculate the wildcard mask determining the binary opposite of a subnet mask.
 11111111.11111111.11111111.11100000 = 255.255.255.224
 00000000.00000000.00000000.00011111 = 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 ?**

Configuring Extended ACLs - Scenario 1

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>
```

- i. 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 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 no particular type of ICMP traffic needs 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.

- a. Notice that one of the options is **<cr>** (carriage return). In other words, you can press **Enter** and the **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 Gigabit Ethernet 0/0 interface. Enter interface configuration mode and apply the ACL.

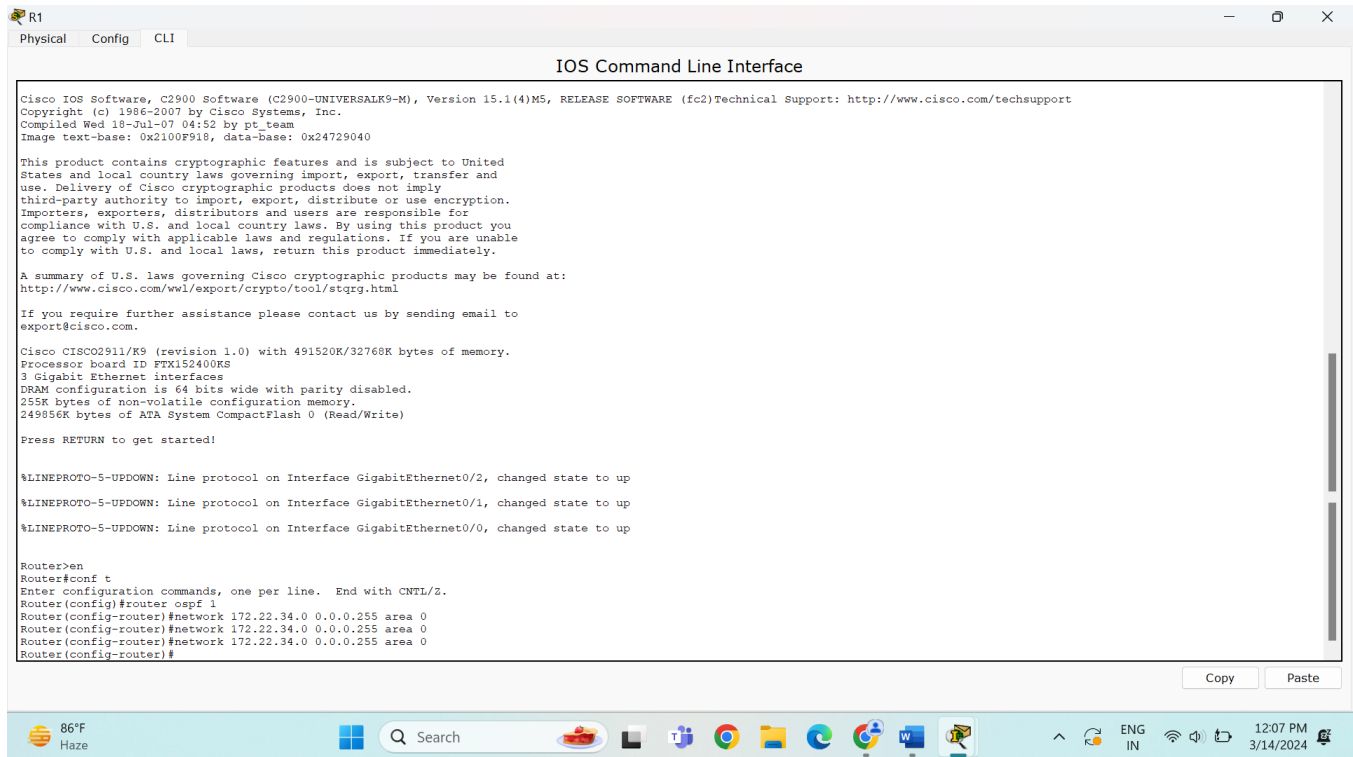
```
R1(config)# interface gigabitEthernet 0/0
```

```
R1(config-if)# ip access-group 100 in 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 of the **Server**.
- b.
- c. Open the web browser on **PC2** and enter the IP address of **Server** as the URL. The connection should be successful.



The screenshot shows a Windows desktop environment. At the top, there is a window titled 'R1' with tabs for 'Physical', 'Config', and 'CLI'. The 'CLI' tab is active, displaying the 'IOS Command Line Interface'. The text in the CLI window includes:

```
Cisco IOS Software, C2900 Software (C2900-UNIVERSALK9-M), Version 15.1(4)M5, RELEASE SOFTWARE (fc2) Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Wed 18-Jul-07 04:52 by pt_team
Image text-base: 0x2100F918, data-base: 0x24729040

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wll/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

Cisco C1802911/K9 (revision 1.0) with 491520K/32768K bytes of memory.
Processor board ID FTX152400RS
3 Gigabit Ethernet interfaces
DRAM configuration is 64 bits wide with parity disabled.
255K bytes of non-volatile configuration memory.
249856K bytes of ATA System CompactFlash 0 (Read/Write)

Press RETURN to get started!

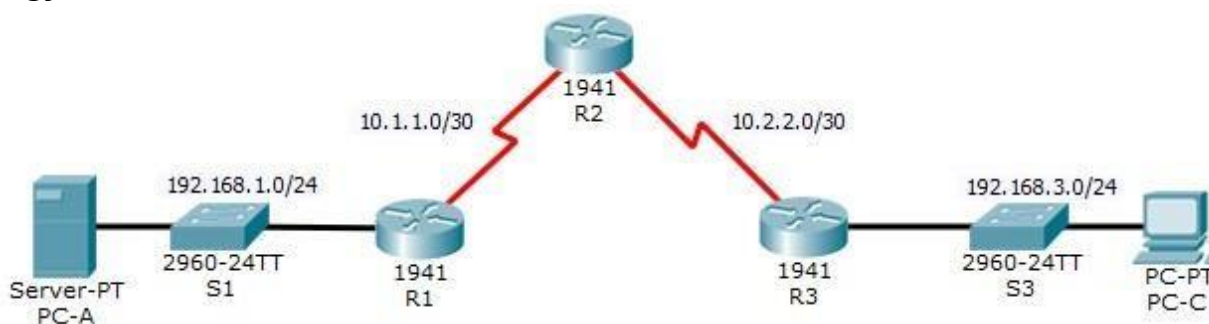
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/2, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 1
Router(config-router)#network 172.22.34.0 0.0.0.255 area 0
Router(config-router)#network 172.22.34.0 0.0.0.255 area 0
Router(config-router)#network 172.22.34.0 0.0.0.255 area 0
Router(config-router)#
```

At the bottom of the screen, the Windows taskbar is visible, showing the system tray with the date and time (12:07 PM, 3/14/2024) and various icons including the Start button, search bar, and application icons for File Explorer, Microsoft Edge, and others.

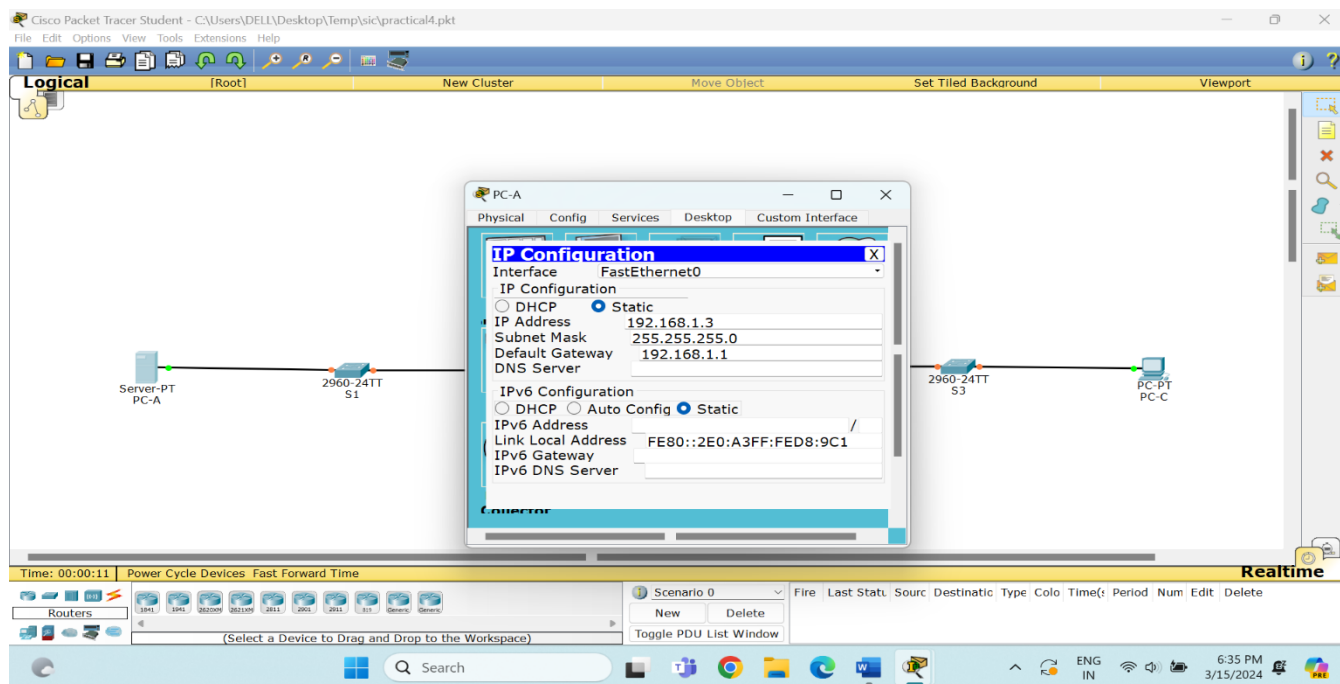
Practical 4: Configure IP ACLs to Mitigate Attacks.

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway	Switch Port
R1	G0/1	192.168.1.1	255.255.255.0	N/A	S1 F0/5
	S0/0/0 (DCE)	10.1.1.1	255.255.255.252	N/A	N/A
R2	S0/0/0	10.1.1.2	255.255.255.252	N/A	N/A
	S0/0/1 (DCE)	10.2.2.2	255.255.255.252	N/A	N/A
	Lo0	192.168.2.1		N/A	N/A
R3	G0/1	192.168.3.1	255.255.255.0	N/A	S3 F0/5
	S0/0/1	10.2.2.1	255.255.255.252	N/A	N/A
PC-A	NIC	192.168.1.3	255.255.255.0	192.168.1.1	S1 F0/6
PC-C	NIC	192.168.3.3	255.255.255.0	192.168.3.1	S3 F0/18



Objectives

- Verify connectivity among devices before firewall configuration.
- Use ACLs to ensure remote access to the routers is available only from management station PC-C.
- Configure ACLs on R1 and R3 to mitigate attacks.
- Verify ACL functionality.

Background/Scenario

Access to routers R1, R2, and R3 should only be permitted from PC-C, the management station. PC-C is also used for connectivity testing to PC-A, which is a server providing DNS, SMTP, FTP, and HTTPS services. Standard operating procedure is to apply ACLs on edge routers to mitigate common threats based on source and destination IP address. In this activity, you will create ACLs on edge routers R1 and R3 to achieve this goal. You will then verify ACL functionality from internal and external hosts. The routers have been pre-configured with the following:

Configure IP ACLs to Mitigate Attacks

- Enable password: **ciscoenpa55**
- Password for console: **ciscoconpa55**
- SSH logon username and password:
SSHadmin/ciscosshpa55
- IP addressing
- Static routing

Part 1: Verify Basic Network Connectivity

Verify network connectivity prior to configuring the IP ACLs.

Step 1: From PC-A, verify connectivity to PC-C and R2.

- From the command prompt, ping **PC-C** (192.168.3.3).
- From the command prompt, establish an SSH session to **R2** Lo0 interface (192.168.2.1) using username **SSHadmin** and password **ciscosshpa55**. When finished, exit the SSH session. `SERVER> ssh -l SSHadmin 192.168.2.1`

Step 2: From PC-C, verify connectivity to PC-A and R2.

- From the command prompt, ping **PC-A** (192.168.1.3).
- From the command prompt, establish an SSH session to R2 Lo0 interface (192.168.2.1) using username SSHadmin and password ciscosshpa55. Close the SSH session when finished. `PC> ssh -l SSHadmin 192.168.2.1`
- Open a web browser to the PC-A server (192.168.1.3) to display the web page. Close the browser when done.

Part 2: Secure Access to Routers

Step 1: Configure ACL 10 to block all remote access to the routers except from PC-C. Use the **access-list** command to create a numbered IP ACL on **R1**, **R2**, and **R3**.

```
R1(config)# access-list 10 permit host 192.168.3.3
R2(config)# access-list 10 permit host 192.168.3.3
R3(config)# access-list 10 permit host 192.168.3.3
```


Step 2: Apply ACL 10 to ingress traffic on the VTY lines. Use the **access-class** command to apply the access list to incoming traffic on the VTY lines.

```
R1(config-line)# access-class 10 in
R2(config-line)# access-class 10 in
R3(config-line)# access-class 10 in
```

Step 3: Verify exclusive access from management station PC-C.

- a. Establish an SSH session to 192.168.2.1 from **PC-C** (should be successful).

```
PC> ssh -l SSHadmin 192.168.2.1
```

Configure IP ACLs to Mitigate Attacks

- b. Establish an SSH session to 192.168.2.1 from **PC-A** (should fail).

Part 3: Create a Numbered IP ACL 120 on R1

Create an IP ACL numbered 120 with the following rules:

- o Permit any outside host to access DNS, SMTP, and FTP services on server **PC-A**.
- o Deny any outside host access to HTTPS services on **PC-A**.
- o Permit **PC-C** to access **R1** via SSH.

Note: Check Results will not show a correct configuration for ACL 120 until you modify it in Part 4.

Step 1: Verify that PC-C can access the PC-A via HTTPS using the web browser.

Be sure to disable HTTP and enable HTTPS on server **PC-A**.

Step 2: Configure ACL 120 to specifically permit and deny the specified traffic. Use the **access-list** command to create a numbered IP ACL.

```
R1(config)# access-list 120 permit udp any host 192.168.1.3 eq domain
R1(config)# access-list 120 permit tcp any host 192.168.1.3 eq smtp
R1(config)# access-list 120 permit tcp any host 192.168.1.3 eq ftp
R1(config)# access-list 120 deny tcp any host 192.168.1.3 eq 443
R1(config)# access-list 120 permit tcp host 192.168.3.3 host 10.1.1.1 eq 22
```

Step 3: Apply the ACL to interface S0/0/0. Use the **ip access-group** command to apply the access list to incoming traffic on interface S0/0/0.

```
R1(config)# interface s0/0/0
R1(config-if)# ip access-group 120 in
```

Step 4: Verify that PC-C cannot access PC-A via HTTPS using the web browser.

Part 4: Modify an Existing ACL on R1

Permit ICMP echo replies and destination unreachable messages from the outside network (relative to **R1**).
Deny all other incoming ICMP packets.

Step 1: Verify that PC-A cannot successfully ping the loopback interface on R2.

Step 2: Make any necessary changes to ACL 120 to permit and deny the specified traffic. Use the `access-list` command to create a numbered IP ACL.

```
R1(config)# access-list 120 permit icmp any any echo-reply
R1(config)# access-list 120 permit icmp any any unreachable
R1(config)# access-list 120 deny icmp any any
R1(config)# access-list 120 permit ip any any
```

Step 3: Verify that PC-A can successfully ping the loopback interface on R2.

The screenshot displays the Cisco Packet Tracer interface with R1 configuration and CLI output.

Top Panel: R1 Configuration

- Physical Tab:** Shows the router's physical components.
- Config Tab:** Shows the configuration for GigabitEthernet0/1.
- CLI Tab:** Shows the command-line interface.

Configuration Details:

- Port Status:** On
- Bandwidth:** 1000 Mbps
- Duplex:** Full Duplex
- MAC Address:** 000A.F3A6.A602
- IP Configuration:**
 - IP Address: 192.168.1.1
 - Subnet Mask: 255.255.255.0
- Tx Ring Limit:** 10

Equivalent IOS Commands:

```
!LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up
00:00:10: %OSPF-5-ADJCHG: Process 1, Nbr 100.2.2.2 on Serial0/0/0 from LOADING to FULL, Loading Done

Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface GigabitEthernet0/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface GigabitEthernet0/1
Router(config-if)#
```

Bottom Panel: IOS Command Line Interface

States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at: <http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to export@cisco.com.

Cisco CISC01941/K9 (revision 1.0) with 491520K/32768K bytes of memory.
Processor board ID FTK152400KS
2 Gigabit Ethernet interfaces
4 Low-speed serial(sync/async) network interface(s)
DRAM configuration is 64 bits wide with parity disabled.
255K bytes of non-volatile configuration memory.
249856K bytes of ATA System CompactFlash 0 (Read/Write)

Press RETURN to get started!

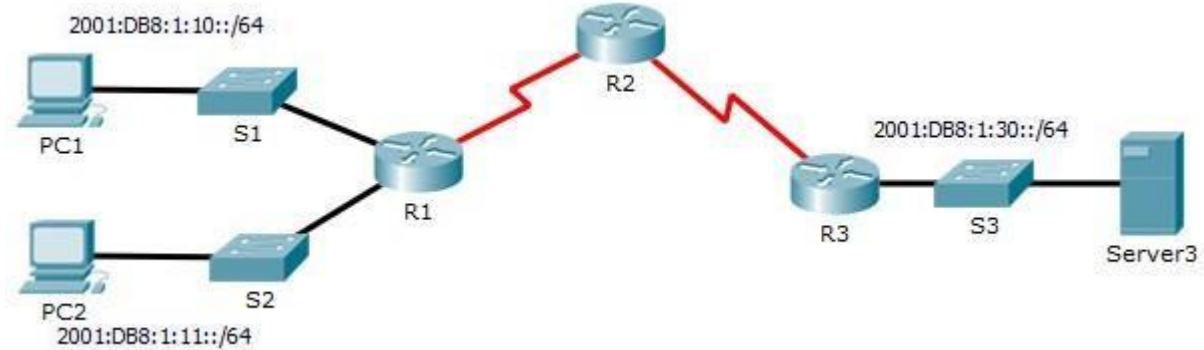
```
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up
%LINK-5-CHANGED: Interface Serial0/0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up
00:00:10: %OSPF-5-ADJCHG: Process 1, Nbr 100.2.2.2 on Serial0/0/0 from LOADING to FULL, Loading Done

Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface GigabitEthernet0/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface GigabitEthernet0/1
Router(config-if)#exit
Router(config)#router ospf 1
Router(config-router)#network 192.168.1.0 0.0.0.255 area 0
Router(config-router)#network 10.1.1.0 0.255.255.255 area 0
Router(config-router)#
```

System Tray: Shows system status (86°F, Air: Poor), search bar, and system clock (11:55 AM, 3/14/2024).

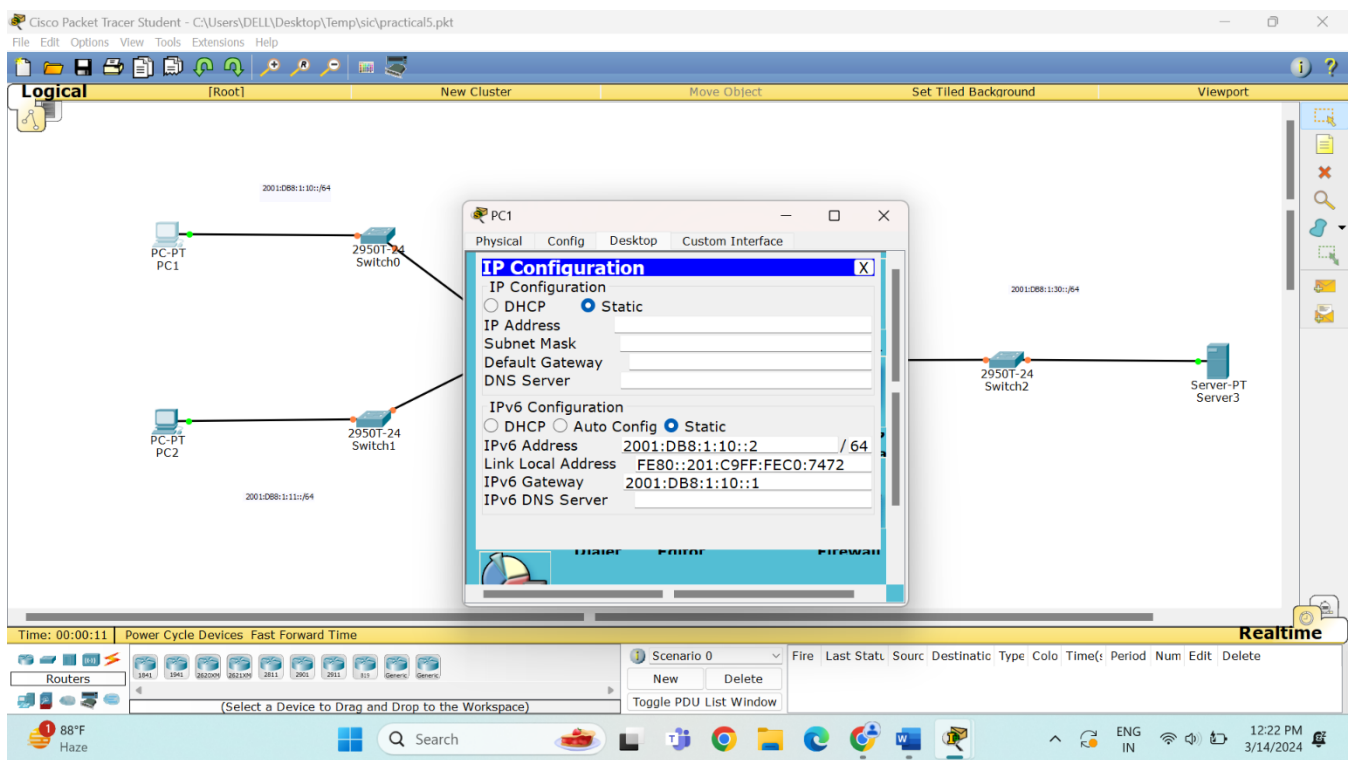
Practical 5: Configuring IPv6 ACLs

Topology



Addressing Table

Device	Interface	IPv6 Address/Prefix	Default Gateway
Server3	NIC	2001:DB8:1:30::30/64	FE80::30



Objectives

Part 1: Configure, Apply, and Verify an IPv6 ACL

Part 2: Configure, Apply, and Verify a Second IPv6 ACL

Part 1: Configure, Apply, and Verify an IPv6 ACL

Logs indicate that a computer on the 2001:DB8:1:11::0/64 network is repeatedly refreshing a web page. This is causing a Denial-of-Service (DoS) attack against **Server3**. Until the client can be identified and cleaned, you must block HTTP and HTTPS access to that network with an access list.

Step 1: Configure an ACL that will block HTTP and HTTPS access.

Configure an ACL named **BLOCK_HTTP** on **R1** with the following statements. a. Block HTTP and HTTPS traffic from reaching **Server3**.

```
R1(config)# deny tcp any host 2001:DB8:1:30::30 eq www
R1(config)# deny tcp any host 2001:DB8:1:30::30 eq 443
```

b. Allow all other IPv6 traffic to pass.

```
R1(config)# permit ipv6 any any
```

Step 2: Apply the ACL to the correct interface. Apply the ACL on the interface closest to the source of the traffic to be blocked.

```
R1(config)# interface GigabitEthernet0/1
R1(config-if)# ipv6 traffic-filter BLOCK_HTTP in
```

Step 3: Verify the ACL implementation.

Verify that the ACL is operating as intended by conducting the following tests:

- Open the **web browser** of **PC1** to http://2001:DB8:1:30::30 or https://2001:DB8:1:30::30. The website should appear.
- Open the **web browser** of **PC2** to http://2001:DB8:1:30::30 or https://2001:DB8:1:30::30. The website should be blocked.
- Ping from **PC2** to 2001:DB8:1:30::30. The ping should be successful.

Part 2: Configure, Apply, and Verify a Second IPv6 ACL

The logs now indicate that your server is receiving pings from many different IPv6 addresses in a Distributed Denial of Service (DDoS) attack. You must filter ICMP ping requests to your server.

Step 1: Create an access list to block ICMP.

Configure an ACL named **BLOCK_ICMP** on **R3** with the following statements: a. Block all ICMP traffic from any hosts to any destination.

```
R3(config)# deny icmp any any
```

b. Allow all other IPv6 traffic to pass.

```
R3(config)# permit ipv6 any any
```

Step 2: Apply the ACL to the correct interface.

In this case, ICMP traffic can come from any source. To ensure that ICMP traffic is blocked, regardless of its source or any changes that occur to the network topology, apply the ACL closest to the destination.

```
R3(config) # interface GigabitEthernet0/0
R3(config-if) # ipv6 traffic-filter BLOCK_ICMP out
```

Step 3: Verify that the proper access list functions.

- Ping from **PC2** to 2001:DB8:1:30::30. The ping should fail.
 - Ping from **PC1** to 2001:DB8:1:30::30. The ping should fail.
- Open the **web browser** of **PC1** to <http://2001:DB8:1:30::30> or <https://2001:DB8:1:30::30>. The website should display.

