FIREBRAND

CISCO

CCNA Certification

ICND1 Lab Guide

Version 2.0 Issue 1.01

www.firebrandtraining.com

ICND1

Interconnecting Cisco Networking Devices, Part 1

Version 2.0

Lab Guide

Issue v1.01

Table of Contents

Lab 2-2: Internet connections.

Visual Topology

Command List

Task 1: Defining static IP addresses and setting a static default route.

Task 2: Configure NAT.

Task 3: Configure PAT.

Lab 3-1: Improving Device Security.

Visual Topology

Command List

Task 1: Device password protection.

Task 2: Remote access using Telnet and SSH.

Task 3: Limiting remote access based on source IP addresses.

Task 4: Creating a login banner page.

Lab 3-2: Device Hardening.

Visual Topology

Command List

Task 1: Managing unused ports on a Switch.

Task 2: Using Switchport port-security.

Task 3: Disable unused services.

Lab 3-3: Using ACLs to filter IP based traffic.

Visual Topology

Command Line

Task 1: Configuring an ACL (Access Control List).

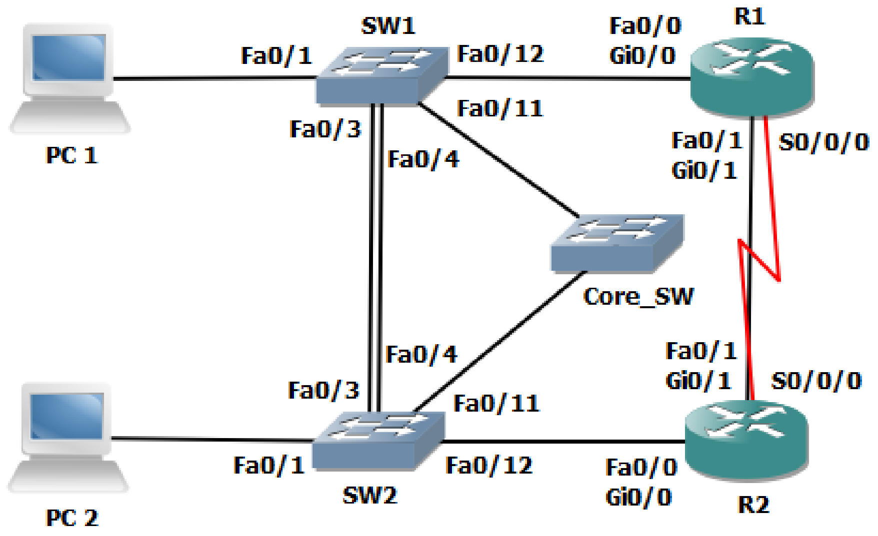
Lab 5-2: Configure IPv6 Routing.

Visual Topology

Command Line

Task 1: Enable OSPFv3.

Lab Answer Keys:



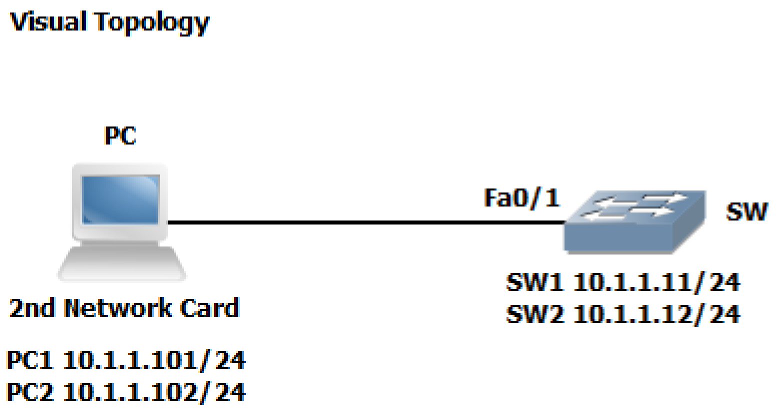
Connections Table

PC - Switch Fa0/1
SW1 Fa0/3 - SW2 Fa0/3
SW1 Fa0/4 - SW2 Fa0/4
SW1 Fa0/11 - Core_SW Fa0/X see note 1
SW2 Fa0/11 - Core_SW Fa0/X see note 1
SW1 Fa0/12 - R1 Fa0/0 or Gi0/0 see note 2
SW2 Fa0/12 - R2 Fa0/0 or Gi0/0 see note 2
R1 S0/0/0 - R2 S0/0/0
R1 Fa0/1 or Gi0/1 - R2 Fa0/1 or Gi0/1 see note 2

Note 1
This topology supports 2 students, each student gets to configure PC1, SW1 & R1 or PC2, SW2 & R2. The Core Switch is managed by the instructor for all student connections.
Students will need to work in pairs during some of the lab exercises.

Note 2
If R1 and R2 are 2811 routers then use Fa0/0 and Fa0/1
If R1 and R2 are 2901 routers then use Gi0/0 and Gi0/1

Lab 1-1: Switch Startup and Initial Configuration.



Command List

Command	Description
? or help	Lists available commands in your current mode
Clock set	Sets the system clock
Configure Terminal	Enters global configuration mode
Copy run start	Saves your dynamic running config to NVRAM
Delete *name*	Deletes a file
Do *command*	Allows for the execution of commands located in a different mode
Enable	Enters privileged EXEC mode from user EXEC mode
End	Terminates configuration mode
Erase startup-config	Erases the startup-configuration from NVRAM
Exit	Exits current configuration mode
Terminal history size *number*	Sets the number of lines held in the history buffer.
Hostname *name*	Sets a system name and is displayed within the system prompt
Interface Vlan 1	Enters the interface configuration (SVI) for Vlan 1 and allows you to set the management IP address for the switch.
IP address *address & mask*	Set an IP address and also the network/subnet mask
Line console 0	Enters line console configuration mode
Logging synchronous	Prevents unsolicited messages from interfering when typing in your commands
Reload	Restarts the device
Show clock	Displays the system clock
Show flash:	Displays the contents of the flash memory

Show startup-config	Displays the startup-config saved in NVRam
Show terminal	Displays the current settings for the terminal
Show version	Displays hardware and software information

Task 1: Reload and check that the Switch is set to factory defaults.

Step 1: Assign an IP address to your PC using the details listed in the visual topology diagram. The PC should be fitted with two network adapters check with the instructor if you are unsure which network adapter should be configured.

Step 2: Access the Switch Console port using the method and information provided by the instructor.

At the **Switch>** prompt (if you see any other prompt or are asked for a password contact the instructor), enter the **erase startup-config** command and make a note of the result.

Why did this fail?

Step 3: From the user prompt type in the command which enters privilege exec-mode.

Does the system prompt change and if so, how?

Now try and execute the **Erase startup-config** command, once again make a note of the output.

Do you see a different console message?

Step 4: Switches hold information about logical VLANs in a database stored in their flash memory and it is necessary to delete this database to reset the Switch back to factory defaults. **PLEASE BE VERY CAREFUL WHEN USING THE DELETE COMMAND.**

From Privilege mode type in the following command and follow the system messages (if you are unsure what to do, contact the instructor before answering any of the system messages).

Switch#**Delete flash:vlan.dat**

Step 5: Use the appropriate command to verify that the Switch doesn't have a current startup-configuration and use the appropriate show command to display information about the device hardware and software parameters.

Step 6: Reload the Switch.

Please note the Switch may take a few minutes to reload.

Task 2: Defining a hostname and enabling a management IP address.

Step 1: Change the hostname of the Switch to either **SW1** or **SW2**

Step 2: Assign your Switch a management IP address from the values identified in the visual topology diagram at the beginning of the lab exercise.

Step 3: Verify connectivity between your PC and the Switch using the **Ping** command, remember that your PC might have a personal firewall installed which could prevent the Switch from Pinging the PC.

Was the Ping successful ?

If not, investigate and correct the problem.

Task 3: Using context-sensitive help.

Step 1: Access the privilege mode on your Switch and enter **?** to list the available commands.

Step 2: Using the **?** navigate through the series of command options to set the system time to the current time and date. Note that the system will support abbreviated commands provided they are unique and using the **Tab** key will automatically complete the command.

Step 3: Use a command to show the current time and date.

Step 4: Type in the following command at the privilege command prompt.

! The next set of commands will configure the links to the core office

The **!** at the beginning of a line indicates that you are entering a comment into the running configuration, this can be very useful to other colleagues and engineers who are trying to ascertain the nature of the configuration.

Step 5: To help navigate around the CLI (command line interface) a number of key combinations can be used. Spend a few minutes trying these combinations out and make a note of what they appear to do, for the best result execute a few valid show commands first.

Ctrl P or the **up arrow** key

Ctrl A

Backspace

Task 4: Changing default CLI parameters.

Step 1: Using the **show terminal** command, verify that history is enabled and determine the current history size for the console line.

Step 2: Use the appropriate command to change the history size to a value of 100 for the console line.

Step 3: When you mistype a command, the system will try and translate it into an IP address which can take some time and be very annoying, however by issuing the **no IP domain-lookup** command it disables the translation look-up.

At the command prompt enter a mistyped command and monitor how long it takes for the system to return an error message, now execute the **no IP domain-lookup** command and try the mistyped command for a second time and notice the difference.

Step 4: When accessing the console port there is a default keyboard inactivity timeout of 10 minutes. Change this timer to **60 minutes.**

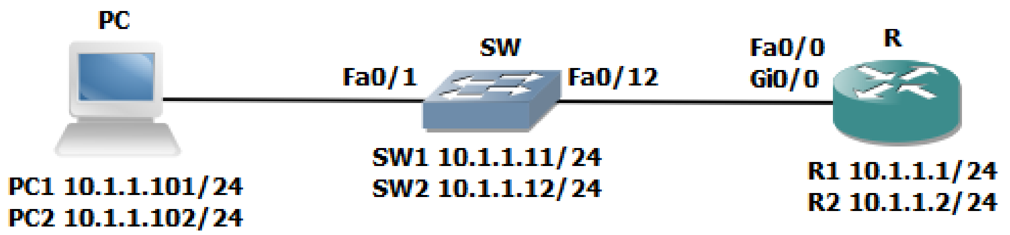
Step 5: What does the **logging synchronous** command do?

Enable this command on the **line console 0** port.

Step 6: Save your running-configuration.

Lab 1-2: Troubleshooting Switch Media Issues.

Visual Topology



NB. Switchports fa0/3, fa0/4 and fa0/11 should be shutdown for this Lab exercise

Router 2811 use Fa0/0
Router 2901 use Gi0/0

Command List

Commands	**Description**
Configure Terminal	Enters global configuration mode.
Copy run start	Saves the dynamic running-config to NVRAM.
Duplex *full / Half / Auto*	Enables the interface duplex setting.
Enable	Enters privilege EXEC mode
Interface Fastethernet 0/0	Specifies interface fa0/0
Interface Gigabitethernet 0/0	Specifies interface gi0/0
Shutdown/ No Shutdown	Disables or enable an interface
Ping *ip-address or hostname*	Checks IP connectivity
Show Interface Fastethernet 0/0	Displays information about interface fa0/0
Show Interface Gigabitethernet 0/0	Displays information about interface gi0/0
Show IP Interface Brief	Displays a brief summary of the device interfaces
Speed *10/100/1000/auto*	Sets the speed of an interface

Task 1: Lab setup.

Step 1: Make sure that interfaces fa0/3, fa0/4 and fa0/11 are shutdown on your Switch.

Task 2: Connectivity issues between the PC and the Switch.

Step 1: Check the connectivity between the PC and the Switch using the Ping command, if the Ping fails check the status of Switch interface fa0/1 and verify that its status is up/up. If the interface is administratively down issue the **No Shutdown** command to bring it up.

Step 2: Enter the correct interface mode for the Switch SVI (management interface) and shut the interface down. Check the IP connectivity between the PC and the Switch, this should now fail.

Enable the SVI (management interface) and check that IP connectivity has been restored.

Task 3: Connectivity issues between the Switch and the Router.

Step 1: Check that interface fa0/12 on the Switch isn't administratively shutdown, rectify if it is.

Step 2: Access the console port of the Router using the access method described by the instructor.

Give the router a hostname of R1 or R2

Step 3: Enter the interface configuration mode this will be either fa0/0 if you are using a 2811 Router or gi0/0 if you are using a 2901 Router.

Give the interface an IP address of either 10.1.1.1/24 (R1 only) or 10.1.1.2/24 (R2 only).

Try and Ping the IP address of your switch. Was this successful?

If not check the status of the interface, what do you notice?

Rectify the condition and try to Ping the switch again. Only when you have full IP connectivity between the Router and the Switch move on to the next step.

Step 4: Access the interface (fa0/0 or gi0/0) configuration mode on the Router and change the **speed** setting to **10**, now access the CLI on the Switch and enter the interface fa0/12 configuration mode and set the **speed** to **100.**

Check the status of the interfaces connecting the Switch and Router together, make a note of their layer 1 and layer 2 states.

Would you expect connectivity when there is a speed mis-match?

Reconfigure the Router (interface fa0/0 or gi0/0) to match the speed of the switch, remember best working practice suggests you shutdown the interface before making any changes and after you have reconfigured the interface enter the no shut command.

Verify connectivity before moving on to the next step.

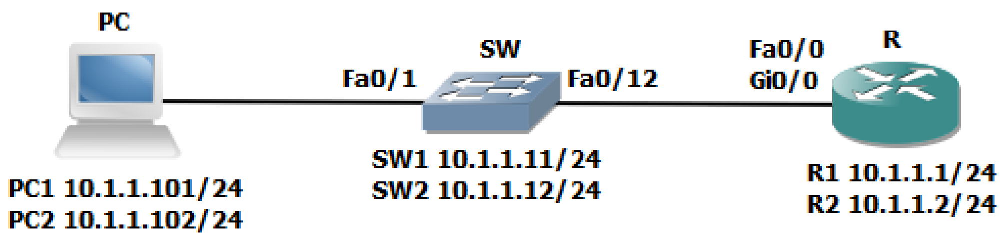
Step 5: Configure Switch interface fa0/12 to half duplex and configure Router interface (fa0/0 or gi0/0) to full duplex.

Check the layer 1 and layer 2 status of the connecting interfaces and record your results below.

Once you are ready to move on, reconfigure Switch interface fa0/12 to full duplex, check IP connectivity and save your running-config on both devices.

Lab 2-1: Router startup and Initial Configuration.

Visual Topology



NB. Switchports fa0/3, fa0/4 and fa0/11 should be shutdown for this Lab exercise

Router 2811 use Fa0/0
Router 2901 use Gi0/0

Command List.

Command	Description
Configure Terminal	Enters global configuration mode.
Copy run start	Saves the dynamic running-config to NVRAM.
Description	Adds a descriptive comment to an interface
Erase startup-config	Removes the saved startup-config from NVRAM
Exec-timeout	
Hostname *name*	Sets the system device name
Interface *type module/slot/port*	Enters interface mode
IP address *address/mask*	Sets an IP address and network/subnet mask.
[no] IP domain lookup	Translates host/FQDN to IP addresses
Line con 0	Enters the console configuration mode
Logging synchronous	Prevents unsolicited messages from interfering when typing in your commands
Ping *ip address/host*	Checks IP connectivity
Reload	Restarts the system
Show CDP	Displays CDP parameters
Show CDP neighbors [detail]	Displays the contents of the CDP dynamic table
Show interface	Displays interface parameters and status
Show version	Displays hardware and software information
[no] shutdown	Disables or enable an interface

Task 1: Router hardware and software inspection.

Step 1: Access the CLI of your router and enter privilege exec mode.

Step 2: Use the appropriate command to display the hardware and software properties of the router.

Fill in the table below.

Router Model	
System image file	
RAM	
Flash	
Software version	

Step 3: Check the NVRAM for a startup-config file using the **sh start** command and remove the startup-configuration if one exists using the **erase startup-config** command.

Step 4: Issue the **reload** command to restart the router and observe the boot process from the console.

Task 2: Create the initial Router configuration.

Step 1: Skip the initial configuration dialog, terminate the autoinstall and enter privilege EXEC mode.

Step 2: Set the system hostname to either R1 or R2

Does the system prompt change?

Step 3: Enter the correct configuration mode to add a description to the first ethernet interface on the router (fa0/0 or gi0/0) **Link to LAN Switch.**

Step 4: Configure an interface IP address and mask with reflects the values shown in the Visual Topology diagram.

R1 10.1.1.1/24

R2 10.1.1.2/24

What is the status of the interface?

Do you think we could ping the IP address of the switch?

Take any necessary steps to enable IP connectivity between the Router and the Switch before you move on to the next step.

Step 5: Save your running-config to NVRAM

Task 3: Changing default CLI parameters.

Step 1: Change the EXEC timeout on the console port to a value of 60 minutes

Step 2: Enter the **sh line con 0** command

Does this command verify the new timeout value?

Step 3: Improve the readability of the console access by synchronising unsolicited messages and debug outputs with the input of the CLI.

Step 4: Use the relevant command which prevents the system from translating a mistyped command to an IP address.

Step 5: Save your running-config to NVRAM

Task 4: Neighbour discovery using CDP.

Step 1: Using the **sh cdp** command fill in the table below

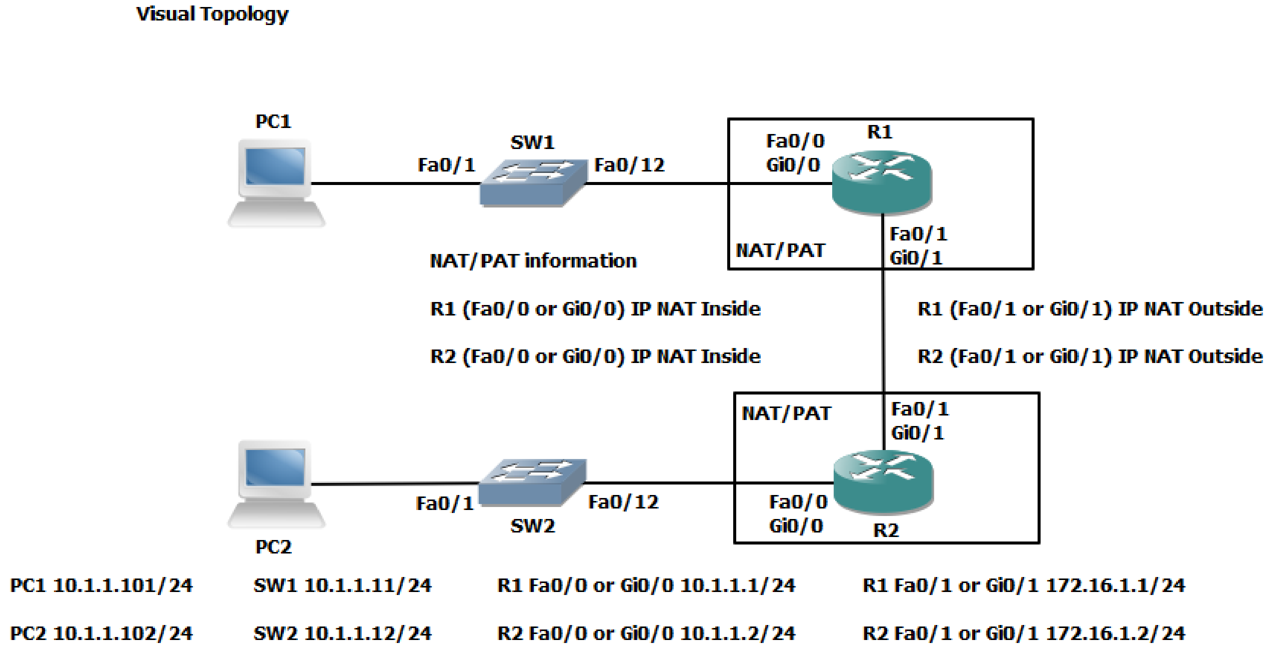
How often are CDP advertisements being sent	
How long will a CDP neighbour entry be held in the table without being refreshed.	
What version of CDP is currently running on your device	

Step 2: Issue the **sh cdp nei** command and if you see any devices fill in the table below.

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID

Step 3: Execute the **sh cdp nei detail** command, do you see any additional information not shown using the command in step 2.

Lab 2-2: Internet connections.



Command List

Command	Description
Access-list *acl id* permit *network wildcard mask*	Creates a standard IP access control list (ACL)
Debug IP ICMP	Displays real-time ICMP traffic activity
IP nat inside	Configures an interface as NAT inside
IP nat inside source list *acl id* pool *pool name*	Dynamic source NAT rule that translates candidates permitted by an ACL to a pool of global addresses.
IP nat inside source list *acl id* interface *name* overload	Dynamic source PAT rule which translates candidates permitted by an ACL to the IP address of the IP nat outside interface
IP nat outside	Configures an interface as NAT outside
IP nat pool *pool name start ip end ip* netmask *mask*	Configures a pool of global addresses
IP route *network mask [next hop/exitinterface]*	Creates a static route to a remote network
Sh ip int brief	Displays interface status and ip information
Show ip route	Displays a list of the best paths to networks
Show users	Information regarding active line connections
[no] shutdown	Disable or enables an interface
Telnet *ip address / hostname*	Telnet to a remote device

Terminal monitor	Redirects debugging output to your telnet session instead of the default console port
Undebug all	Disable all debugging commands

Task 1: Defining static IP addresses and setting a static default route.

Step 1: Access the CLI of the router and verify the current status of the ethernet interfaces using the **show ip int brief** command.

The first ethernet interface should already be configured with an IP address and a status of up/up, if not rectify this.

Step 2: Enter the configuration mode of the second Ethernet interface (fa0/1 or gi0/1), place the interface into a disable state and then manually assign an IP address which is listed in the Visual Topology diagram.

Step 3: Enable the second Ethernet interface and Ping the other router, remember both sides of the link will need to be configured and enabled before the Ping will be successful.

Step 4: Execute the command which allows you to view the contents of the routing table.

R1#sh ip route

or

R2#sh ip route

How many entries would you expect to see? Can you see any remote networks?

Task 2: Configure NAT.

Step 1: Access the CLI on the Router

Step 2: Configure a standard IP ACL using an ACL id of **1** and permit any device on subnet 10.1.1.0 /24

R1(config)#**access-list 1 permit 10.1.1.0 0.0.0.255**

or

R2(config)#**access-list 1 permit 10.1.1.0 0.0.0.255**

This ACL will be used to identify which source IP addresses are going to be translated using NAT, and this example allows any device from the 10.1.1.0 subnet.

Step 3: Create a dynamic NAT address pool, this will hold a list of inside global addresses.

Use this table and parameters on **Router R1 only**

Pool name	**NAT-POOL**
Starting IP address	**192.168.1.1**
Ending IP address	**192.168.1.14**
Network mask	**255.255.255.240**

R1(config)#**ip nat pool NAT-POOL 192.168.1.1 192.168.1.14 netmask 255.255.255.240**

Use this table and parameters on **Router R2 only**

Pool name	**NAT-POOL**
Starting IP address	**192.168.2.1**
Ending IP address	**192.168.2.14**
Network mask	**255.255.255.240**

R2(config)#**ip nat pool NAT-POOL 192.168.2.1 192.168.2.14 netmask 255.255.255.240**

Stop.....Have you configured the right set of parameters for your Router!

Step 4: Linking the nat pool to the ACL.

R1(config)#**ip nat inside source list 1 pool NAT-POOL**

or

R2(config)#**ip nat inside source list 1 pool NAT-POOL**

NB. Nat Pool names are case-sensitive

Step 5: Before any NAT translations occur we must identify at least two interfaces to be our inside and outside.

Source IP address will be translated when traffic traverses between the inside and outside interfaces and destination IP addresses will be translated in the opposite direction between the outside and inside.

Interface fa0/0 or gi0/0 will be our inside interface

Interface fa0/1 or gi0/1 will be our outside interface

Assign the following commands to the relevant interfaces.

The example shown illustrates the commands required on Router R1 which is a 2901 device, you may need to use fa0/0 and fa0/1 if you are using a 2811 device.

R1(config)#int gi0/0

R1(config-if)#ip nat inside

R1(config-if)#int gi0/1

R1(config-if)#ip nat outside

Step6: When we ping from PC1 to R2 or PC2 to R1 our original IP address will be translated from a 10.1.1.x to a 192.168.x.x address (x denotes a variable depending on the direction of the traffic)

Therefore we need to configure a static route back to the 192.168.x.x network.

R1 only...

R1(config)#**ip route 192.168.2.0 255.255.255.0 172.16.1.2**

R2 only...

R2(config)#**ip route 192.168.1.0 255.255.255.0 172.16.1.1**

Use the appropriate command to verify that they have been added to the routing table.

R1#**sh ip route**

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

******some output missing******

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks

C 10.1.1.0/24 is directly connected, GigabitEthernet0/0

L 10.1.1.1/32 is directly connected, GigabitEthernet0/0

172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks

C 172.16.1.0/24 is directly connected, GigabitEthernet0/1

L 172.16.1.1/32 is directly connected, GigabitEthernet0/1

S 192.168.2.0/24 [1/0] via 172.16.1.2

Step 7: We have now configured all of the NAT components and a static route to the translated addresses, the next stage is to test our configuration.

From your PC check you still have a valid **10.1.1.11** or **10.1.1.12** address using **ipconfig/all** from the command shell (cmd).

PC1 will require a default gateway address of 10.1.1.1

and

PC2 will require a default gateway address of 10.1.1.2

Verify and rectify if necessary.

Check you can ping your default gateway from your PC

If you are having problems open the command shell (cmd) and type in the following statements.

On PC1 only...

route -p add 10.1.1.0 mask 255.255.255.0 10.1.1.1

route -p add 172.16.1.0 mask 255.255.255.0 10.1.1.1

On PC2 only...

route -p add 10.1.1.0 mask 255.255.255.0 10.1.1.2

route -p add 172.16.1.0 mask 255.255.255.0 10.1.1.2

Your classroom PC might be fitted with dual interface cards and we need to direct our traffic out of the correct interface.

Step 7: Testing NAT translation.

Traffic will need to traverse across the inside and outside interfaces before any entries will be seen in the IP NAT Translation table.

From you classroom PC ping the IP address of the outside interface on the corresponding router.

Examples

From PC1 ping 172.16.1.2 (this is the outside interface on R2)

From PC2 ping 172.16.1.1 (this is the outside interface on R1)

Check the contents of the IP nat table using the following command.

sh ip nat trans

(remember show commands run from privilege EXEC mode)

Example from R1

R1#**sh ip nat trans**

Pro	Inside global	Inside local	Outside local	Outside global
icmp	192.168.1.1:10	10.1.1.11:10	172.16.1.2:10	172.16.1.2:10
icmp	192.168.1.1:11	10.1.1.11:11	172.16.1.2:11	172.16.1.2:11
icmp	192.168.1.1:12	10.1.1.11:12	172.16.1.2:12	172.16.1.2:12
icmp	192.168.1.1:9	10.1.1.11:9	172.16.1.2:9	172.16.1.2:9

Task 3: Configure PAT.

Task 1: Removing the previous NAT configuration so we can apply PAT using the same pair of interfaces.

R1 only...

Type in the following commands.

R1(config)#**no ip nat pool NAT-POOL 192.168.1.1 192.168.1.14 netmask 255.255.255.240**

R1(config)#**no ip nat inside source list 1 pool NAT-POOL**

R2 only...

Type in the following commands.

R2(config)#**no ip nat pool NAT-POOL 192.168.2.1 192.168.2.14 netmask 255.255.255.240**

R2(config)#**no ip nat inside source list 1 pool NAT-POOL**

These commands remove the dynamic pool of addresses used by NAT and the link between the ACL and NAT Pool.

We will still use the existing ACL and IP NAT Inside/outside interface statements when configuring PAT.

Step 2: Configure a dynamic PAT rule which translates your 10.1.1.0 subnet to the IP address configured on the Routers outside interface.

ip nat inside source list 1 interface fa0/1 overload

Or

ip nat inside source list 1 interface gi0/1 overload

What does the **list 1** part of the command relate to?

What does the key word **overload** do?

Step 3: Verifying your configuration by Pinging the IP address of the other Routers outside interface.

Use the appropriate command to view the contents of the IP translation table.

Do you see any output differences between the previously configured dynamic NAT pool and the newly configured PAT function? Pay attention to the inside global address!

Describe the following NAT/PAT terms

Inside local

Inside global

Outside global

Outside local

Step 4: Once you are satisfied that PAT is configured correctly we can now remove it from the system, remember to remove all components and you will need to be in the right configuration mode to execute these commands.

no ip nat inside

no ip nat outside

no access-list 1

no ip nat inside source list 1 interface fa0/1 overload

or

no ip nat inside source list 1 interface gi0/1 overload

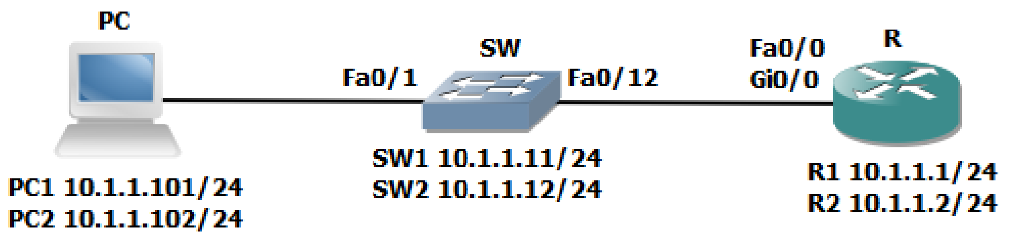
Step 5: Shutdown the fa0/1 or gi0/1 interface.

fa0/1 if you are using a 2811 router and gi0/1 if you are using a 2901 router

Step 6: Save your current configuration.

Lab 3-1: Improving Device Security.

Visual Topology



NB. Switchports fa0/3, fa0/4 and fa0/11 should be shutdown for this Lab exercise

Router 2811 use Fa0/0
Router 2901 use Gi0/0

Command List

Command	Description
Access-class *acl id* [in/out]	Applies an access-list to the VTY lines
Access-list *number* permit *ip address wildcard mask*	Creates a standard IP access control list (ACL)
Banner login	Message will be displayed just before login
Crypto key generate rsa	Generates a RSA Public/Private key pair used for asymmetrical encryption or authentication
Enable secret	Protects the privilege mode. Password is either put through a MD5 or SHA2 hashing algorithm dependant on the version of the IOS being used.
IP domain-name *name*	Applies a system domain name which is required for the process of generating the cryptographic keys
IP ssh version [1/2]	Specifies the version of SSH
line vty 0 4	Enters the configuration mode to manage remote telnet and ssh sessions.
login	Login process for the console or VTY lines, requires password command before the service will start.
login local	Login process for the console or VTY lines using a local authentication database
logout	Exits EXEC mode
Password *password*	Assigns a password to the console or VTY lines

show access-list	Displays the details of any ACLs configured on the device
show users	Display any users currently accessing the system via the console or VTY lines
ssh -l *username ip address*	Allows the system to remotely access a device via a ssh session.
Transport input [telnet / ssh / all]	Specifies which line protocols are permitted on the VTY lines, default is all.
username *username* secret *password*	Creates an entry into the local user database, can be used with ssh and login local functions.

Switches and routers behave differently when set to factory defaults.

Switches will allow all communications between the connecting devices, whereby routers require some initial configuration before any traffic traverses between any two interfaces, both switches and routers will require IP setup before you can remotely manage them via telnet, ssh or snmp.

It is also advisable to protect the privilege EXEC mode, console port and VTY lines using either a password or the credentials of a user.

Task 1: Device password protection.

Step 1: Access the console port of the router.

Step 2: Secure the console port with the password **cisco**

(please don't use any maverick passwords, only those specified in the lab instructions and passwords are case-sensitive)

Step 3: Verify your password by exiting from the line con 0 mode and then the user EXEC mode using the **End** and **Exit** commands.

Step 4: Enter the console password to return to user EXEC mode.

Step 5: Create a local user account with a username of **ccna** and a secret password of **cisco**

Step 6: Change the security method used on the console port to now prompt the administrator for a username and password.

Important...Step 5 must be completed before Step 6 otherwise you will lock yourself out of the system.

Step 7: Verify the security change by logging out of the system, you will notice this time you are asked for a username and password instead of just a password which doesn't identify the person accessing the system.

Task 2: Remote access using Telnet and SSH.

Using telnet (insecure) and ssh (secure) protocols allow administrators to access their devices remotely, providing IP connectivity exists between the telnet/ssh client and the telnet/ssh server.

In this task we are going to configure our router to support telnet and ssh sessions via the VTY lines.

Step 1: Access the router CLI and navigate to the VTY configuration mode, enter a command which forces the administrator to provide a username and password.

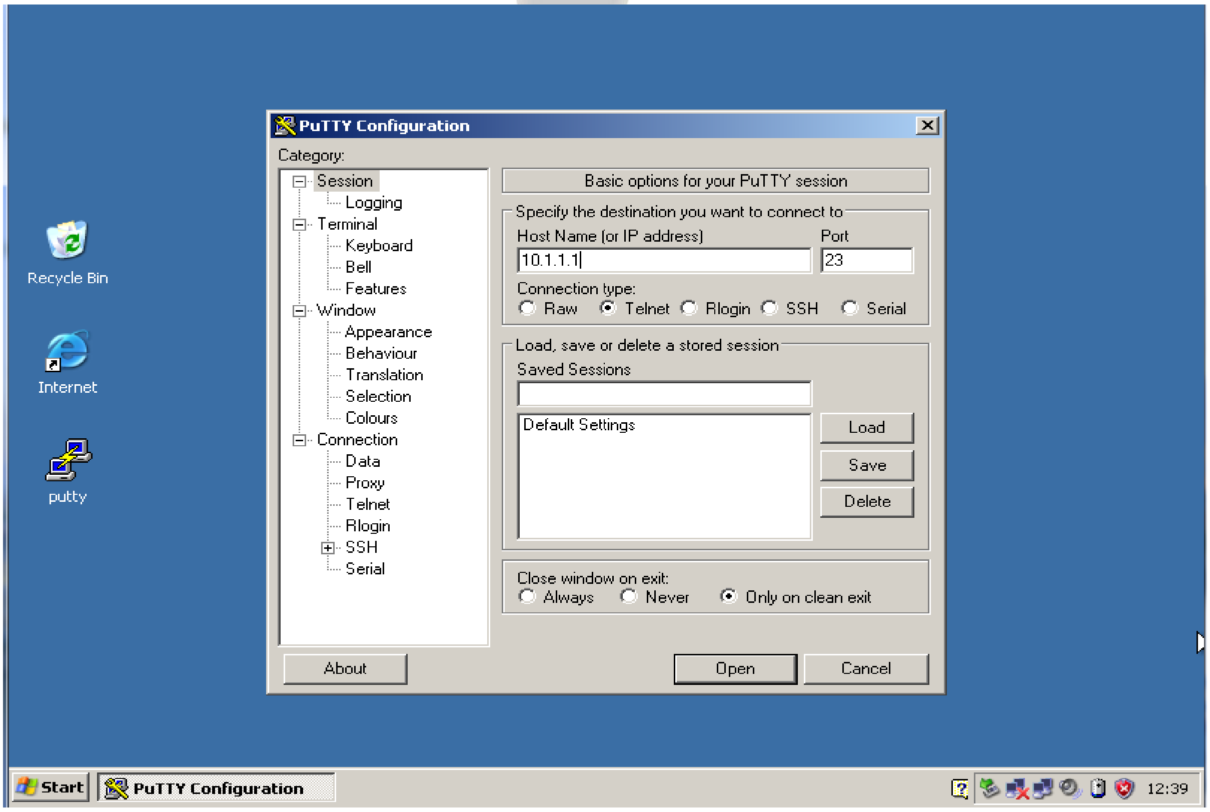
Hint: Used on the console port in the previous task.

Step 2: From the your PCs desktop launch the PuTTy application.

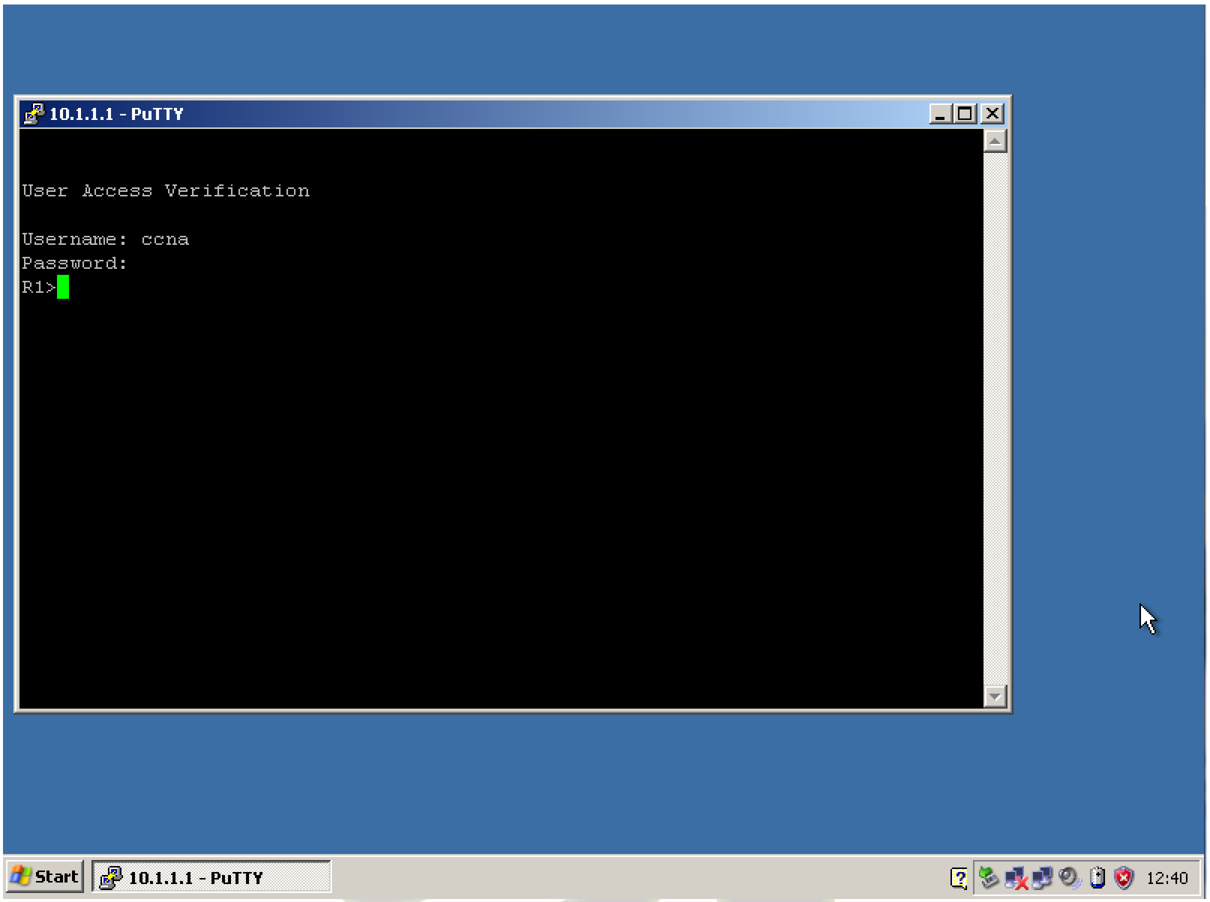
PuTTy is a freely available application which supports both telnet and ssh.

Select the connection type radio button for telnet and type in the IP address of routers interface, in other words your default gateway.

R1 10.1.1.1 or **R2 10.1.1.2**

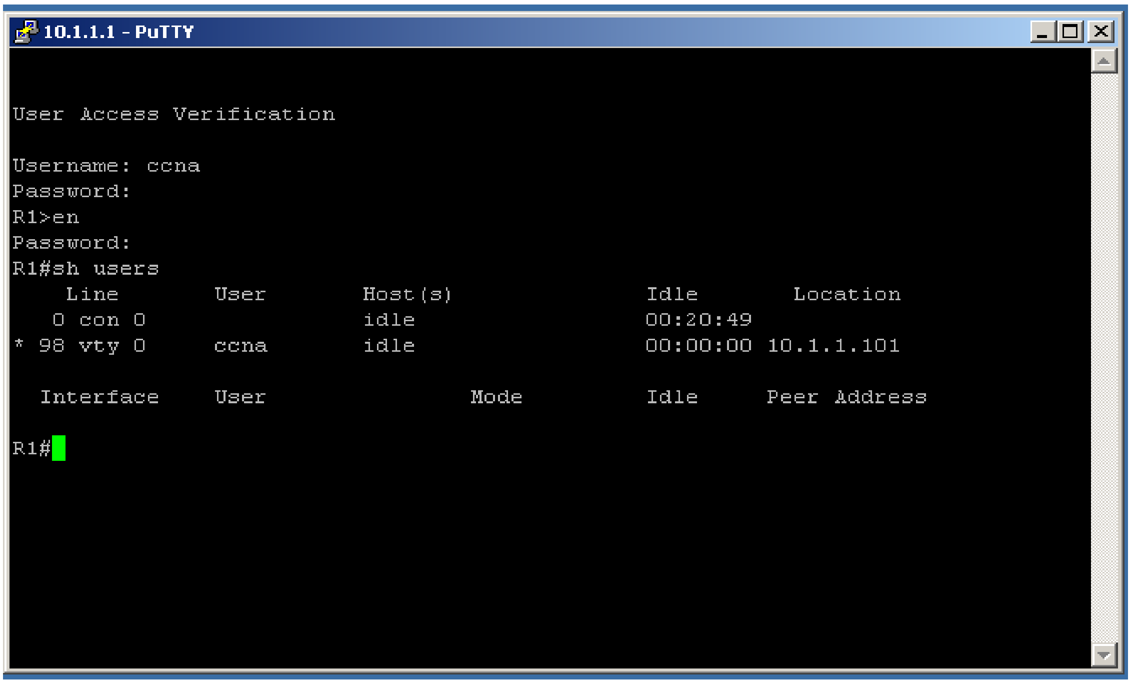


Click open and enter the username **ccna** and password **cisco**



Enter the command to gain access to the privilege mode and type in the secret password of **cisco**

Execute the **sh users** command the output should look like the image below



The output of the **sh users** command indicates that a user is accessing the device via the line VTY 0 with a username of **ccna** and a source IP address of **10.1.1.101.**

Step 3: Telnet provides a method of remote administration but unfortunately when you type in the authentication details, the username and password are sent in clear text therefore telnet should only be used on trusted interfaces or via a VPN encrypted tunnel. SSH version 2 is the preferred method because it can provide authentication and data protection via an encrypted channel.

To configure SSH we need to setup a domain name, generate our RSA public/private key pairs, recommended we support only version 2 and make sure that the VTY lines also support the SSH protocol.

Run the following commands from global configuration mode.

IP domain-name cisco.com

crypto key generate rsa

change the modulus size to **1024** when prompted (ssh version 2 needs a minimum modulus size of 768 bits or greater)

IP ssh version 2

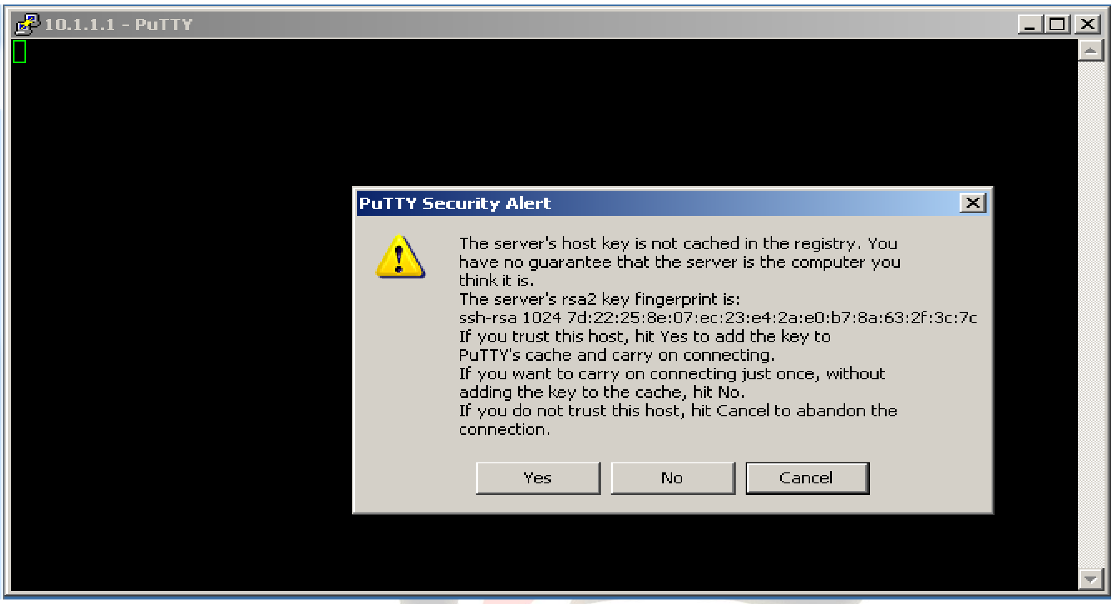
Optional, if you decided that your company policy dictates that all remote connections must be secure then you could use the following command on the VTY lines.

transport input ssh

This command disables all other protocols excluding ssh

Step 4: Check that ssh is working by opening a PuTTy session to your router from your PC but this time under the connection type select the **ssh** radio button, type in the IP address of the router and open the connection.

If you receive the following security alert click **yes**



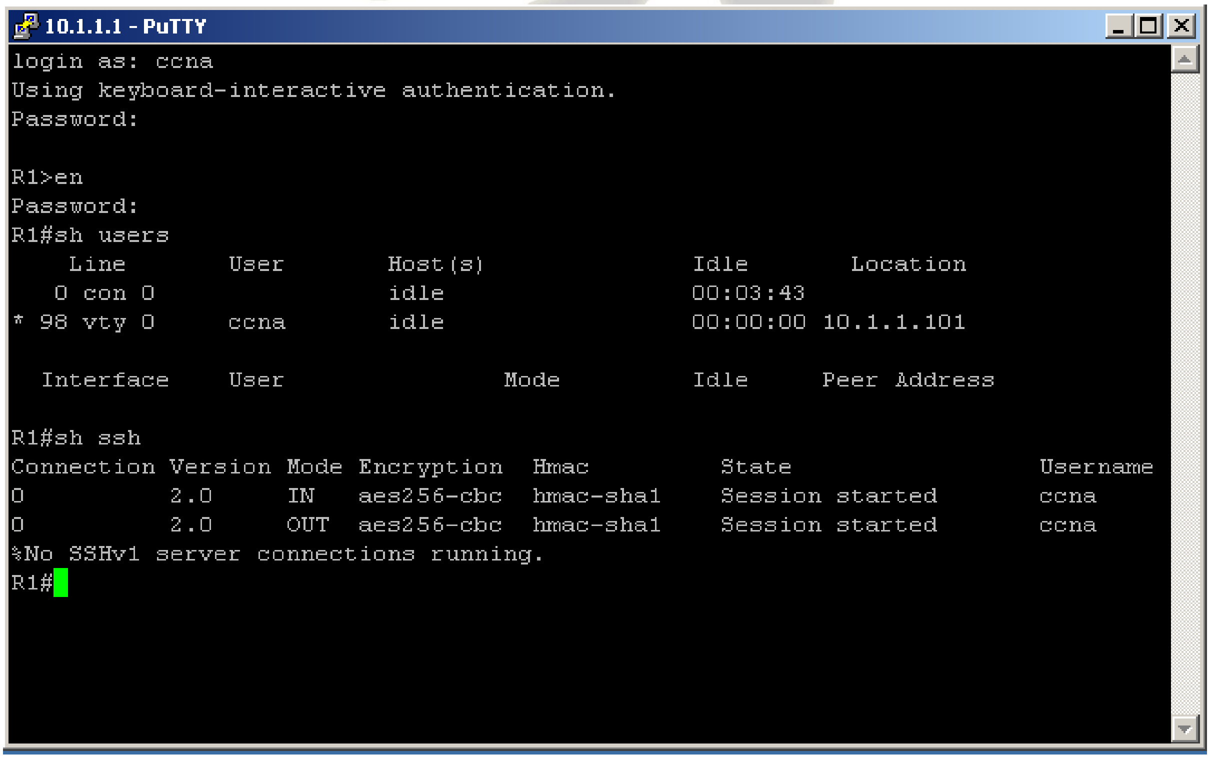
Login as requested using the local account credentials

Enter privilege mode

Execute the following two show commands

sh users

sh ssh



The output states that we are accessing the system via line VTY 0 and ssh version is being used.

If you decided to try the optional **transport input ssh** please reset this back to its default condition, **transport input all** once again enabling telnet and ssh at the same time.

What are the ports numbers used by telnet and ssh?

What command generates the Public/Private key pairs and what was the default modulus size?

Task 3: Limiting remote access based on source IP addresses.

It is possible to restrict which host or subnet a device is on when managing remote access. Standard IP ACLs can be used to identify the source IP address of a ssh or telnet client.

Step 1: Check you can still telnet or ssh into your router from your PC before you start the next step.

Step 2: Check the IP address of your PC and make a note of it below.

Step 3: Access the CLI on the router and navigate to the global configuration mode, this is where you will need to create a standard IP ACL which allows only your PC to telnet or ssh into the router, please use an ACL id of 2.

access-list 2 permit 10.1.1.101 0.0.0.0

or

access-list 2 permit 10.1.1.102 0.0.0.0

What does the wildcard mask **0.0.0.0** do ? Can you think of an alternative way of writing the ACL.

Step 4: Apply this ACL to the VTY lines using the appropriate command, use the command list if you are unsure.

Step 5: Telnet or ssh into your router from your PC, this should still work even after you have applied the ACL.

Step 6: Change the IP address on your PC

PC1 10.1.1.133/24

PC2 10.1.1.144/24

Now try and telnet or ssh to your router, should you be successful?

Step 7: Reset your PC to its original IP address, verify you can telnet or ssh to the router before moving on to the next task.

Task 4: Creating a login banner page.

Most systems have a mandatory security message displayed to anybody accessing the system.

Step 1: Access the router CLI and create a login message which advises only authorised users are permitted to access the system.

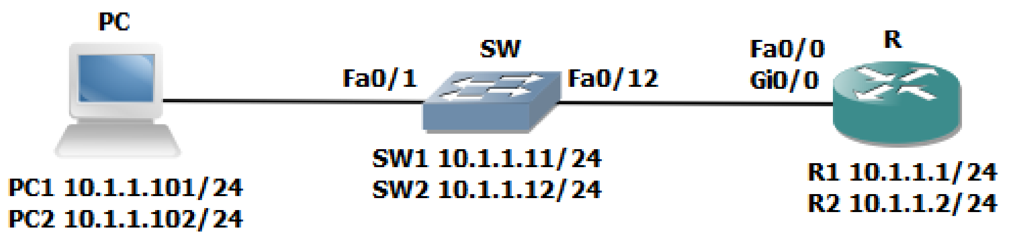
Hint: Remember to use delimiting characters to identify the beginning and end of the displayed message.

Step 2: Telnet or ssh to check your login message.

Step 3: Save your running-config

Lab 3-2: Device Hardening.

Visual Topology



NB. Switchports fa0/3, fa0/4 and fa0/11 should be shutdown for this Lab exercise

Router 2811 use Fa0/0
Router 2901 use Gi0/0

Command List

Command	Description
[no] cdp enable	Enables or disables CDP on an interface
show cdp neighbors [detail]	Displays CDP neighbours
Show mac-address	Displays the contents of the switches mac-address table.
show interfaces	Displays interface statistics
show interface status	Displays interface status
show port-security interface *interface*	Displays port security configured on an interface
show port-security address	Displays port-security MAC addresses
[no] shutdown	Disables or enable a switchport or interface
switchport mode access	Sets the switchport into access mode supporting only one data and one voice vlan
switchport port-security	Enables port-security
switchport port-security mac-address *mac-address*	Sets a static secure MAC address

Task 1: Managing unused ports on a Switch.

Step 1: Access the CLI on your switch

Step 2: Ensure that the interface connecting your switch to your Router is enabled by using the **no shutdown** command, remember to be in the right configuration mode?

Task 2: Using Switchport port-security.

Step 1: Access the CLI on your Router

Step 2: Identify the MAC address of the interface used to connect to your switch.

What command could be used to achieve step 2 ?

Once you know the MAC address of the interface make a note of it below.

MAC Address:

Step 3: Enter the configuration mode which will allow you to change the interface parameters on Fa0/0 or Gi0/0.

Shutdown the interface and set the MAC address to the following value using these commands:

Rx(config-if)#**Shut**

Rx(config-if)#**mac-address 0000.0C12.ABCD**

Step 4: Enable the interface and generate some traffic by pinging the IP address of your switch.

Step 5: Access the switches CLI.

What command could we use to check the MAC address of the attached router?

Step 6: While still accessing the CLI on the switch, shutdown the interface fa0/12

Step 7: Enable switchport security using the following commands:

SWx(config-if)#**switchport mode access**

SWx(config-if)#**switchport port-security**

SWx(config-if)#**switchport port-security mac-address sticky**

Why have we used the first command **switchport mode access** and does it disable DTP ?

Step 8: Enable interface fa0/12 and generate some traffic between the switch and the router.

The connection between the switch and the router should be **up/up** use the **show interface fa0/12** to verify this.

What are the default values used by the system when the **switchport port-security** command is applied to an interface ?

Step 9: Access the CLI on the router and shutdown the interface connected to your switch.

Step 10: On the interface connected to your switch reset the MAC address to its original value.

Step 11: Enable the interface and generate some traffic between the router and the switch.

Step 12: Access the CLI on the switch and check the status of the interface connected to the router. You should now observe a **down/down (err-disabled)** state due to the port violation which occurred when you changed the MAC address of the router.

Step 13: Remove the switchport port-security and check you once again have connectivity between the switch and the router.

Task 3: Disable unused services.

Step 1: Access the CLI on the switch

Step 2: Run the appropriate command to view the CDP neighbours table (remember to use the American spelling)

Is CDP running?

Step 3: Disable CDP on the interface connected to the router and issue the command used in step 2.

Do you see a difference ?

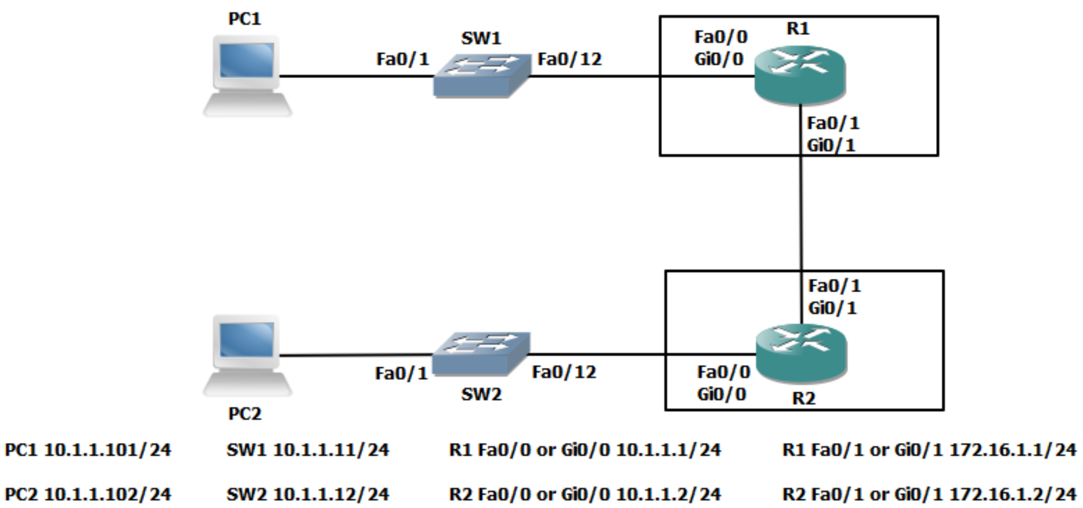
Remember the default timeout value for CDP is 180 seconds.

Step 4: Enable CDP on the interface.

Step 5: Save your configuration.

Lab 3-3: Using ACLs to filter IP based traffic.

Visual Topology



Command Line

Command	Description
Interface loopback0	Creates an internal interface
ip access-group *acl name* [in/out]	Binds an access control list to an interface
ip access-list extended *acl name*	Creates and enters the ACL configuration mode
{permit | deny} {*test conditions*}	Creates control statements within an ACL
show access-lists *acl name*	Displays any ACL located on the device
show ip interface *type/slot/number*	Displays any ACL bound to an interface

Task 1: Configuring an ACL (Access Control List).

Step 1: Access the CLI on your router

Step 2: Create a loopback interface using the following commands.

R1 only...

R1(config)#**interface loopback0**

R1(config-if)#**ip address 1.1.1.1 255.255.255.255**

R1(config-if)#**no shut**

Note we are using a 32 bit mask.

R2 only...

R2(config)#**interface loopback0**

R2(config-if)#**ip address 2.2.2.2 255.255.255.255**

R2(config-if)#**no shut**

Step 3: From your PC check that you can ping the IP address of the loopback interface you have just created and also your default gateway.

If two network cards are fitted you will need to type in the following commands.

PC1 only....

c:\>**router -p add 1.1.1.1 mask 255.255.255.255 10.1.1.1**

PC2 only.....

c:\>**router -p add 2.2.2.2 mask 255.255.255.255 10.1.1.2**

If the ping fails check the following.

Does your PC have the correct IP address ? hint... ipconfig

Can you ping your default gateway ? If not check that the router interface is up/up

Step 4: Create an extended ACL named PING that will prevent your PC from successfully pinging the default gateway, however you should be allowed to ping the loopback interface and all other IP traffic should be permitted through the router.

Step 5: Test your ACL and use the appropriate show commands to display the content of the configured ACL and write down your results below.

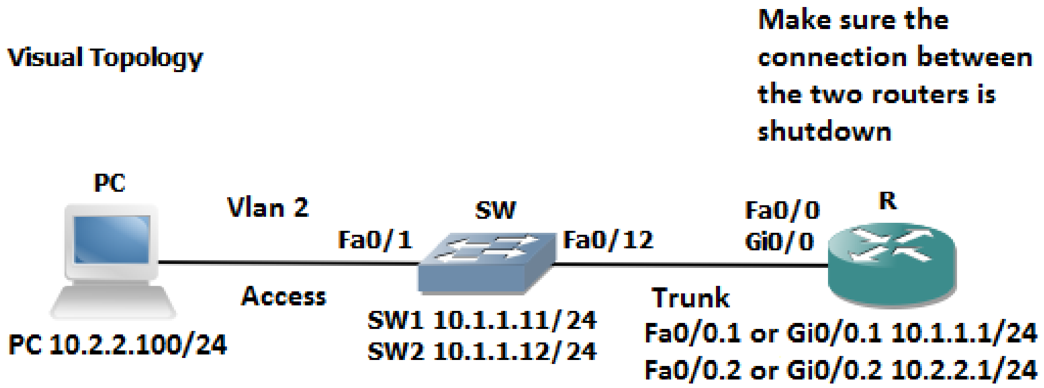
Rx#**show access-list PING**

Are access-list names case sensitive ?

Step 6: Once you have proved that the ACL works correctly then remove it from your configuration.

Step 7: Save your configuration

Lab 4-1: Enhancing a Switched Network.



NB. Switchports fa0/3, fa0/4 and fa0/11 should be shutdown for this Lab exercise

Router 2811 use Fa0/0
Router 2901 use Gi0/0

Command Line

Command	Description
encapsulation dot1q *vlan*	Enables IEEE 802.1Q encapsulation on a routers sub-interface
no ip address	Removes any ip address currently configured on the interface
show interfaces trunk	Displays trunking information
show vlan	Displays vlan information
show vlans	Verify the vlan and trunking configuration on a router on a stick
switchport access vlan *vlan*	Assign a port to a vlan
switchport mode *mode*	Defines DTP modes options available are **access, trunk, dynamic desirable or dynamic auto**
switchport trunk allowed vlan *vlan list*	Filters which vlans are permitted over a trunk connection.
vlan *number*	Creates a vlan

Task 1: Creating a VLAN and assigning Switchports.

Step 1: Access the CLI on the router and make sure the connection between the two routers is shutdown.

Step 2: Access the CLI on the switch.

Step 3: Create vlan **2** on your switch and name it **SALES**

SWx(config)#**vlan 2**

SWx(config-vlan)#**name SALES**

Step 4: Re-assign interface fa0/1 to vlan 2

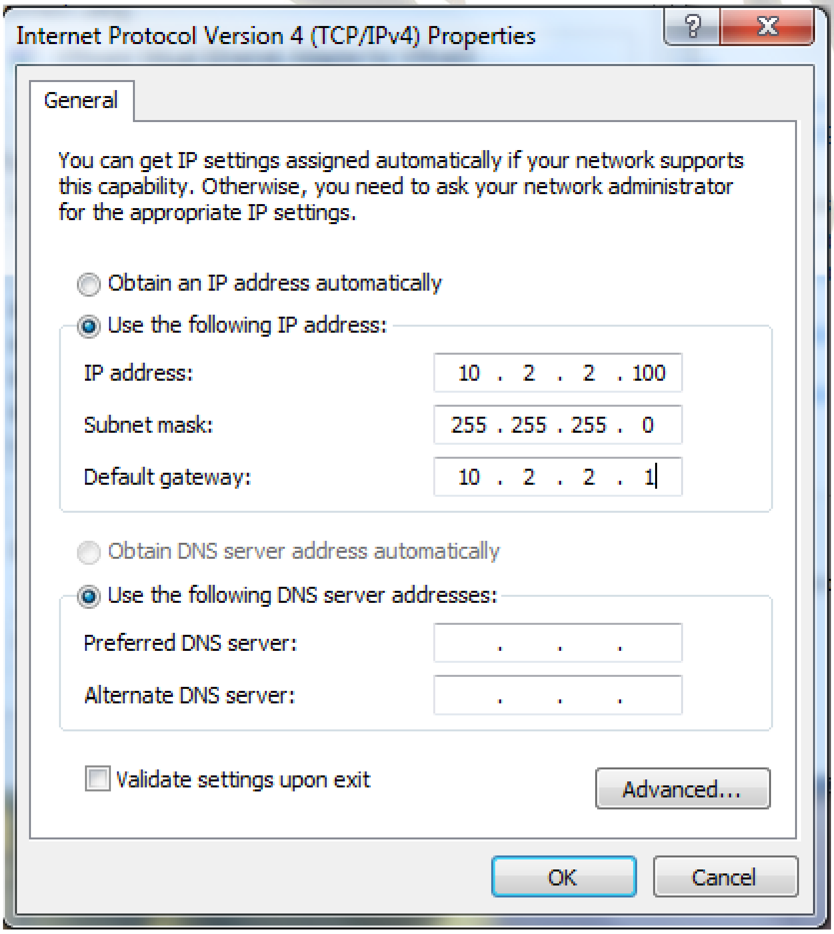
SWx(config-if)#**switchport mode access**

SWx(config-if)#**switchport access vlan 2**

Step 5: Configure Fa0/12 as a trunk connection

SWx(config-if)#**switchport mode trunk**

Step 6: Change the IP address on the PC to 10.2.2.100 with a 255.255.255.0 mask and a default gateway of 10.2.2.1



Step 7: From your PC try and Ping the IP address of your switch.

This should fail! Why ?

Task 2: Configure a Trunk connection on a Router.

Now that we have configured the switch to support a trunk connection between itself and the router, the next stage involves us setting up the router so it understands the IEEE 802.1Q frame encapsulation.

Step 1: Access the CLI on the router.

Step 2: Navigate to the interface mode which connects the router to the switch. Hint.. fa0/0 or gi0/0

Step 3: Shutdown the interface.

Step 4: Remove any current IP address using the **no ip address** command.

Step 5: Create a new sub-interface using the following command.

Rx(config)#**interface fa0/0.1**

or

Rx(config)#**interface gi0/0.1**

Step 6: Assign an ip address of 10.1.1.1 255.255.255.0

Step 7: Issue the following command to support IEEE 801.1Q encapsulation linking it to vlan 1 and make this the native vlan.

Rx(config-subif)#**encap dot1q 1 native**

Step 8: Create a second sub-interface

Rx(config)**interface fa0/0.2**

or

Rx(config)**interface gi0/0.2**

Step 8: Assign an ip address of 10.2.2.1 255.255.255.0

Step 9: Setup IEEE 802.1Q encapsulation with a link to vlan 2

Rx(config-subif)#**encap dot1q 2**

What is the difference between the native vlan and a non-native vlan ?

Step 10: Issue the **no shutdown** command on the physical interface, this will automatically enable all sub-interfaces.

Step 11: Check your PC can ping its default gateway.

Step 12: Try and ping the IP address of the switch, this should now be successful.

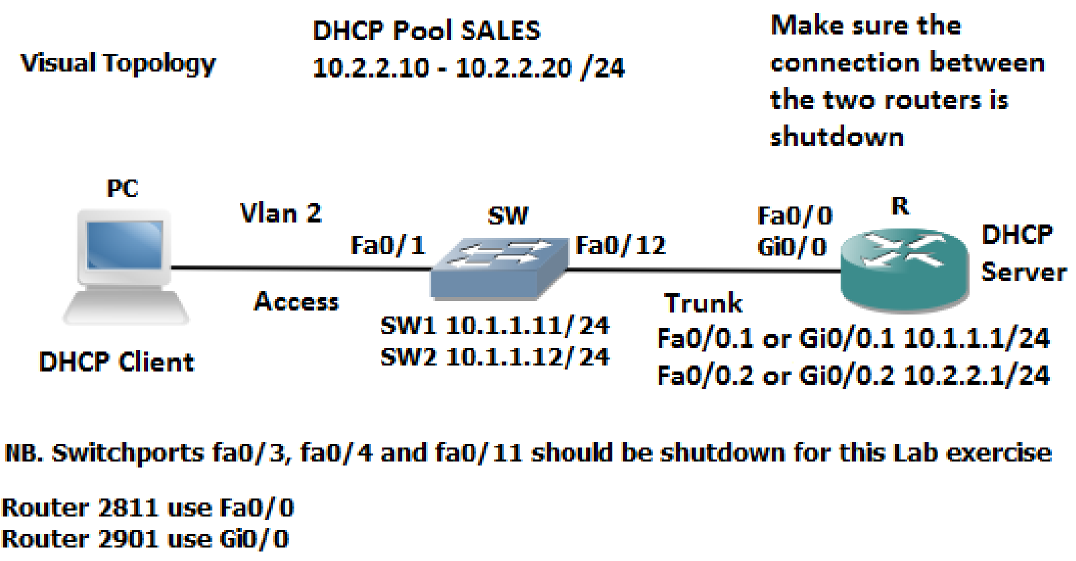
If it fails then check the following.

The switch will need a default gateway set to 10.1.1.1 because the path of the ping from the PC to the switch is via the router.

Explanation: The port attached to the PC has been assigned to Vlan 2 but the IP address of the switch is still in vlan 1, therefore the Ping packet will travel from the PC to its default gateway (the router) because the source IP address and the destination IP address are not located in the same IP subnet. The router upon receiving the Ping packet will direct it to sub-interface fa0/0.2 or gi0/0.2 because it's been linked to vlan 2, it will then examine the destination IP address after stripping the layer 2 header and redirect it out of sub-interface fa0/0.1 or gi0/0.1 but it will need to rebuild a new layer 2 header before going across the trunk to the switch.

Step 13: Save all your configs.

Lab 4-2: Using a Router to provide DHCP Services.



Command Line

Command	Description
default-router *address*	
dns-server *address*	
ip dhcp excluded-address *ip address [last ip address]*	
ip dhcp pool *name*	
ip helper-address *address*	
lease *{days[hours][minutes] | infinite}*	

Task 1: Setting-up DHCP address Pools

Step 1: Access the CLI of the router

Step 2: Configure a DHCP pool named SALES

Rx(config)#**ip dhcp pool SALES**

Step 3: Adding the IP network/subnet to the DHCP pool

Rx(dhcp-config)#**network 10.2.2.0 255.255.255.0**

Step 4: Adding the default gateway and dns server options to the DHCP pool

Rx(dhcp-config)#**default-router 10.2.2.1**

Rx(dhcp-config)#**dns-server 8.8.8.8**

Rx(dhcp-config)#**exit**

Task 2: Excluding static IP addresses from a DHCP Pool.

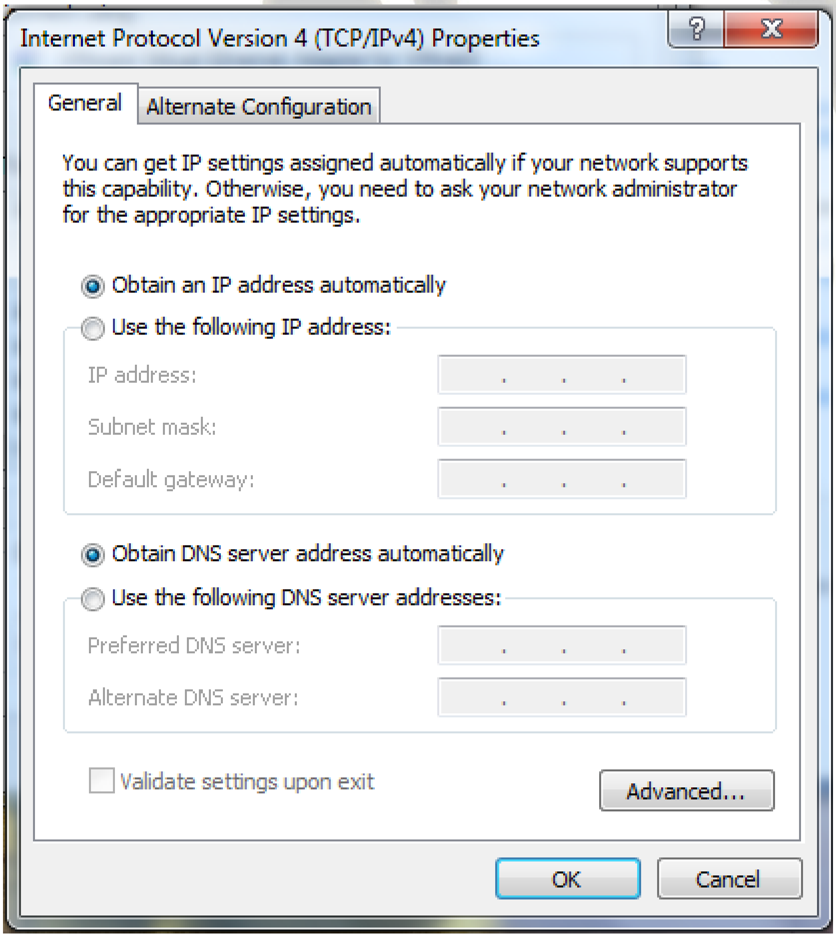
Step 1: Limiting the scope of the DHCP pool

Rx(config)#**ip dhcp excluded-address 10.2.2.1 10.2.2.9**

Rx(config)#**ip dhcp excluded-address 10.2.2.21 10.2.2.254**

Task 3: Testing the DHCP Service.

Step 1: Reconfigure the network properties on the PC to request an IP address from a DHCP server.



Step 2: From the PC command prompt verify your IP address.

```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Dave>ipconfig /all

Windows IP Configuration

        Host Name . . . . . . . . . . . . : virtual-xp
        Primary Dns Suffix  . . . . . . . :
        Node Type . . . . . . . . . . . . : Unknown
        IP Routing Enabled. . . . . . . . : No
        WINS Proxy Enabled. . . . . . . . : No

Ethernet adapter Local Area Connection:

        Connection-specific DNS Suffix  . :
        Description . . . . . . . . . . . : AMD PCNET Family PCI Ethernet Adapter
        Physical Address. . . . . . . . . : 08-00-27-39-C9-05
        Dhcp Enabled. . . . . . . . . . . : Yes
        Autoconfiguration Enabled . . . . : Yes
        IP Address. . . . . . . . . . . . : 10.2.2.10
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 10.2.2.1
        DHCP Server . . . . . . . . . . . : 10.2.2.1
        DNS Servers . . . . . . . . . . . : 8.8.8.8
        Lease Obtained. . . . . . . . . . : 13 August 2013 11:51:22
        Lease Expires . . . . . . . . . . : 14 August 2013 11:51:22
```

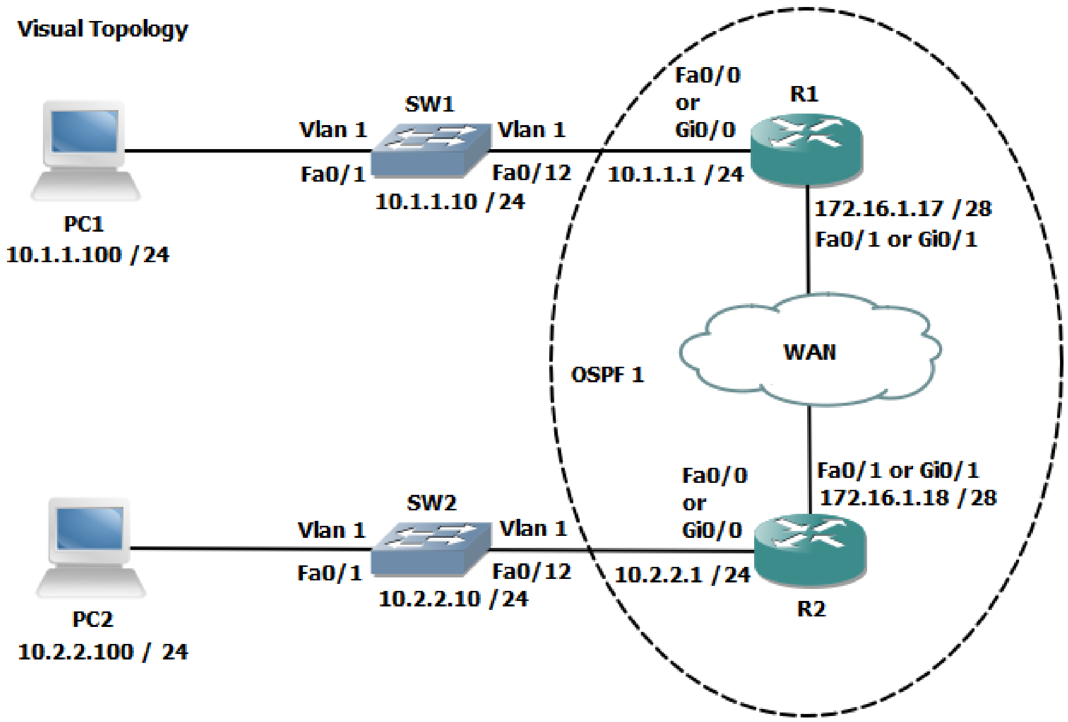
Remember this is an example output.

Step 3: Access the CLI on the router and execute the following command to display current IP address allocation.

Rx#**show ip dhcp binding**

```
R1
R1#
R1#
R1#
R1#
R1#
R1#
R1#
R1#
R1#
R1#
R1#
R1#
R1#sh ip dhcp binding
Bindings from all pools not associated with VRF:
IP address          Client-ID/              Lease expiration        Type
                    Hardware address/
                    User name
10.2.2.10           0108.0027.39c9.05       Aug 14 2013 12:02 PM    Automatic
R1#
R1#
R1#
R1#
R1#
R1#
```

Lab 4-3: Implementing OSPF.



Command Line

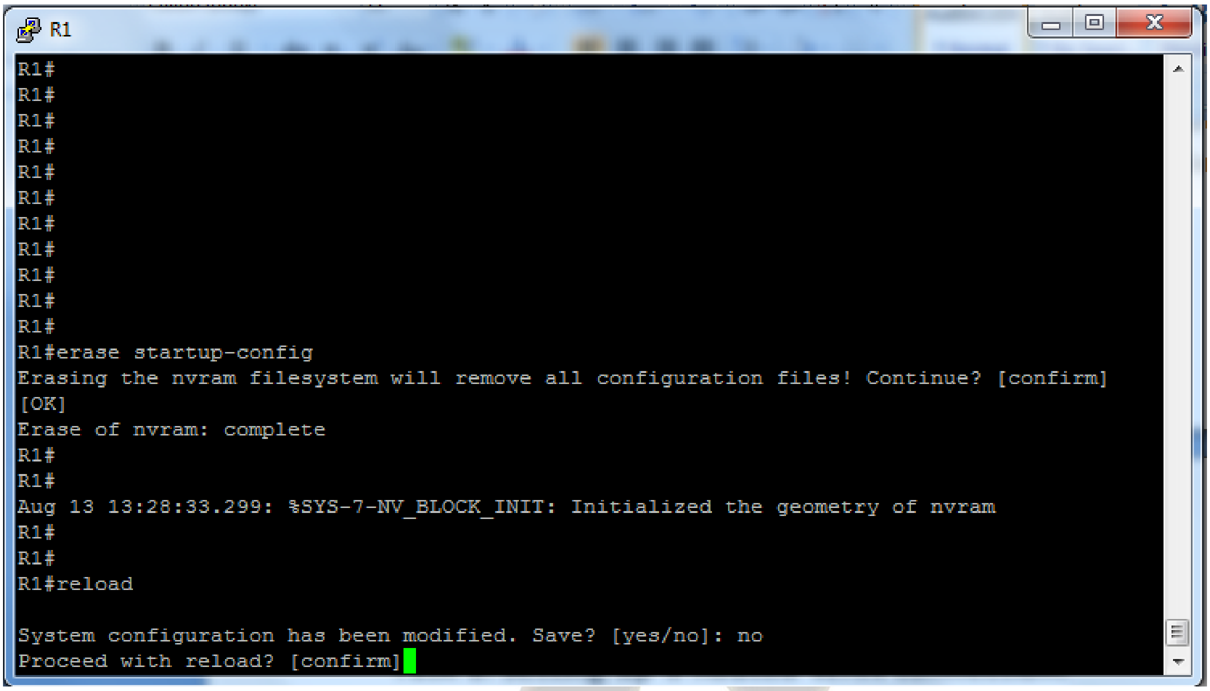
Command	Description
Erase startup-config or Write erase	Removes the startup-configuration from NVram
Hostname	Defines a system device name shown in the prompt
interface *name*	Enters interface configuration mode
ip address *ip address mask*	Assigns an IP address to an interface
Network {address [wildcard mask]} area *id*	Defines which interfaces are part of the OSPF routing process
router ospf *process id*	Enters the OSPF configuration mode and defines the locally significant process id
show ip ospf interface	Displays interface information related to OSPF
show ip ospf neighbor	Shows the output of the OSPF adjacency table
show ip route	Shows the contents of the IPv4 routing table
[no] shutdown	Disables or enables an interface

This lab exercise requires two students to work together to complete the tasks. Use the visual topology diagram to ascertain the correct IP addressing plan for your PC, Router and Switch.

Task 1: Setting-up a Routed WAN connection.

Step 1: Access the CLI on both your switch and router.

Step 2: Clear down their current configuration and reload the devices using the **erase startup-config** and **reload** commands. Make sure you do this on both the router and the switch.



If the system indicates that the system configuration has been modified and do you want to save ? Answer **no**

Confirm the reload.

Step 3: The devices have been set back to factory defaults (well almost) ignore and abort the setup dialogue options.

Using the information in the visual topology diagram, setup the correct IP addresses and hostnames on all 3 devices.

Hint....

Switch#**conf t**

Switch(config)#**hostname SW1**

SW1(config)#**interface vlan 1**

SW1(config-if)#**ip address 10.1.1.10 255.255.255.0**

SW1(config-if)#**no shut**

Router#**conf t**

Router(config)#**hostname R1**

R1(config)#**interface fa0/0**

R1(config-if)#**ip address 10.1.1.1 255.255.255.0**

R1(config-if)#**no shut**

R1(config-if)#**interface fa0/1**

R1(config-if)#**ip address 172.16.1.17 255.255.255.240**

R1(config-if)#**no shut**

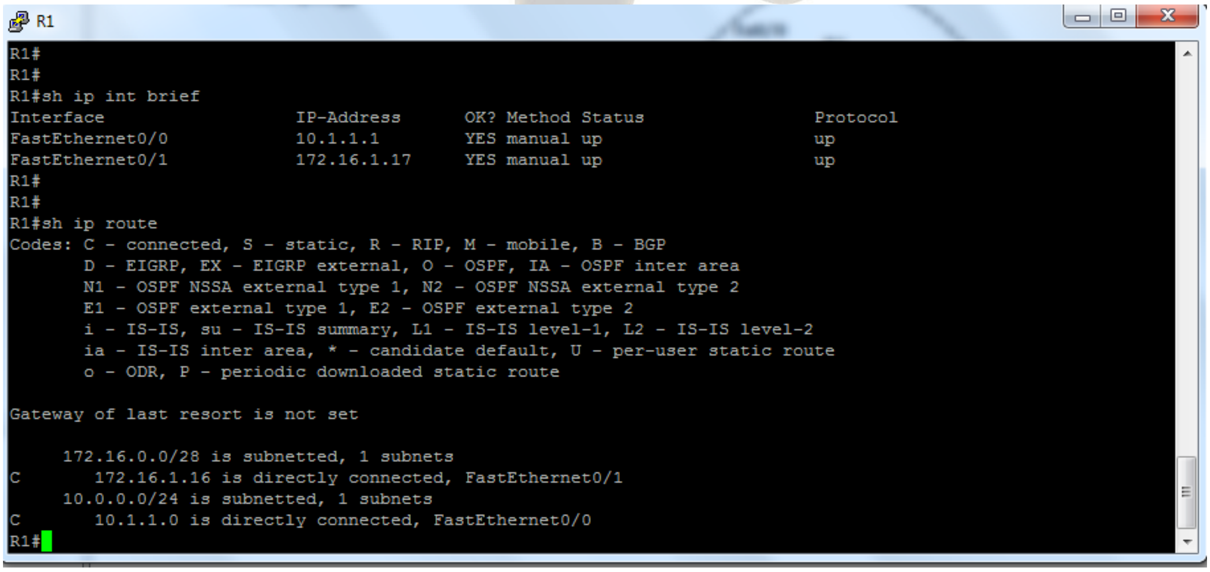
Check that the interfaces are **up/up**, troubleshoot any discrepancies.

Remember to set a static IP address on the PC interface and check connectivity between the PC and its default router.

Task 2: Configuring OSPF.

By default routers do not run any dynamic routing protocols, however routing between IPv4 locally connected interfaces is enabled by default.

Step 1: Execute the relevant **show** command to display the contents of the routing table.



```
R1
R1#
R1#
R1#sh ip int brief
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet0/0            10.1.1.1        YES manual up                    up
FastEthernet0/1            172.16.1.17     YES manual up                    up
R1#
R1#
R1#sh ip route
Codes: 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

Gateway of last resort is not set

     172.16.0.0/28 is subnetted, 1 subnets
C       172.16.1.16 is directly connected, FastEthernet0/1
     10.0.0.0/24 is subnetted, 1 subnets
C       10.1.1.0 is directly connected, FastEthernet0/0
R1#
```

The table displays only two connected subnets and therefore this router only has paths for subnets 10.1.1.0 and 172.16.1.16.

Step 2: Enter OSPF configuration mode and use a process id of 1.

Rx(config)#**Router ospf 1**

Step 3: Enable both interfaces for ospf and place them in the backbone area 0

R1 only.....

R1(config-router)#**network 10.1.1.1 0.0.0.0 area 0**

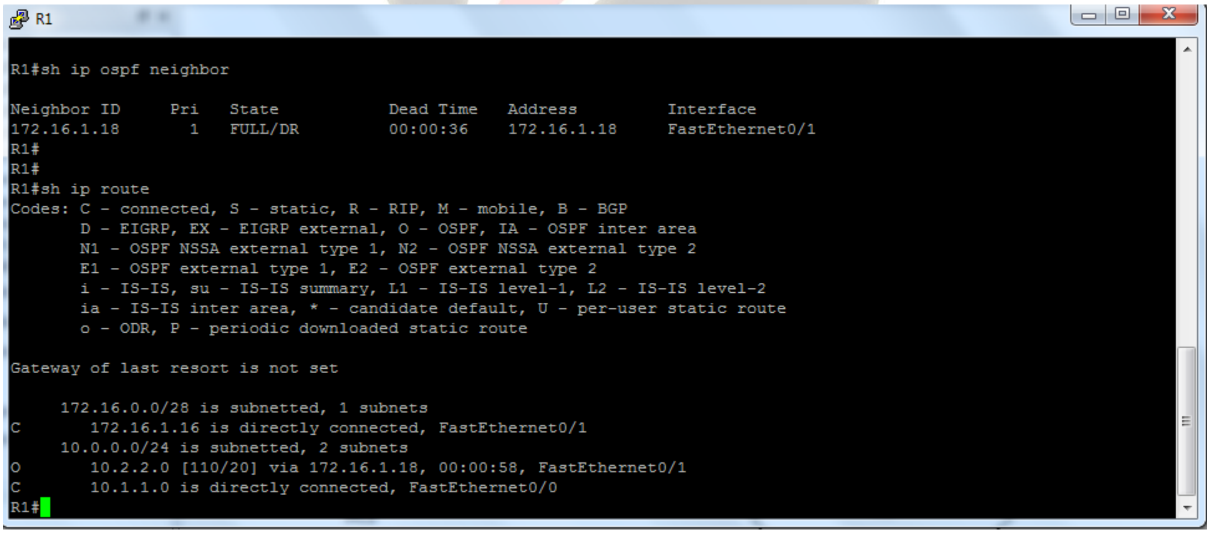
R1(config-router)#**network 172.16.1.17 0.0.0.0 area 0**

R2 only.....

R2(config-router)#**network 10.2.2.1 0.0.0.0 area 0**

R2(config-router)#**network 172.16.1.18 0.0.0.0 area 0**

Step 4: Check the contents of the routing table. Do you see any additional entries?



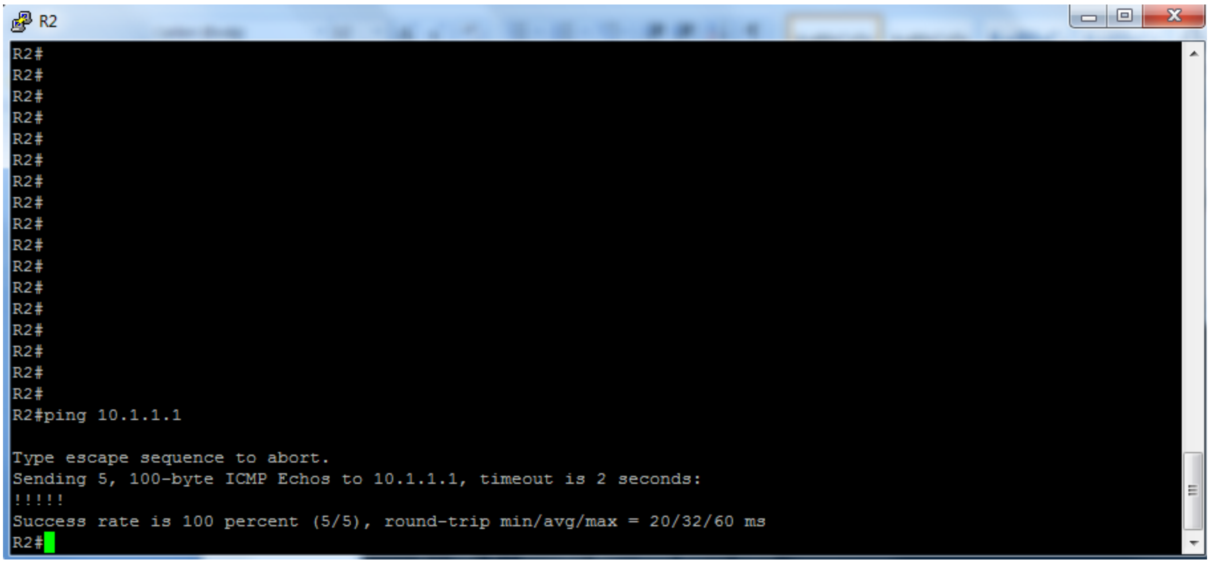
```
R1#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
172.16.1.18       1   FULL/DR         00:00:36    172.16.1.18     FastEthernet0/1
R1#
R1#
R1#sh ip route
Codes: 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

Gateway of last resort is not set

     172.16.0.0/28 is subnetted, 1 subnets
C       172.16.1.16 is directly connected, FastEthernet0/1
     10.0.0.0/24 is subnetted, 2 subnets
O       10.2.2.0 [110/20] via 172.16.1.18, 00:00:58, FastEthernet0/1
C       10.1.1.0 is directly connected, FastEthernet0/0
R1#
```

Step 5: Check connectivity by pinging the IP address of the other router's fa0/0 interface.



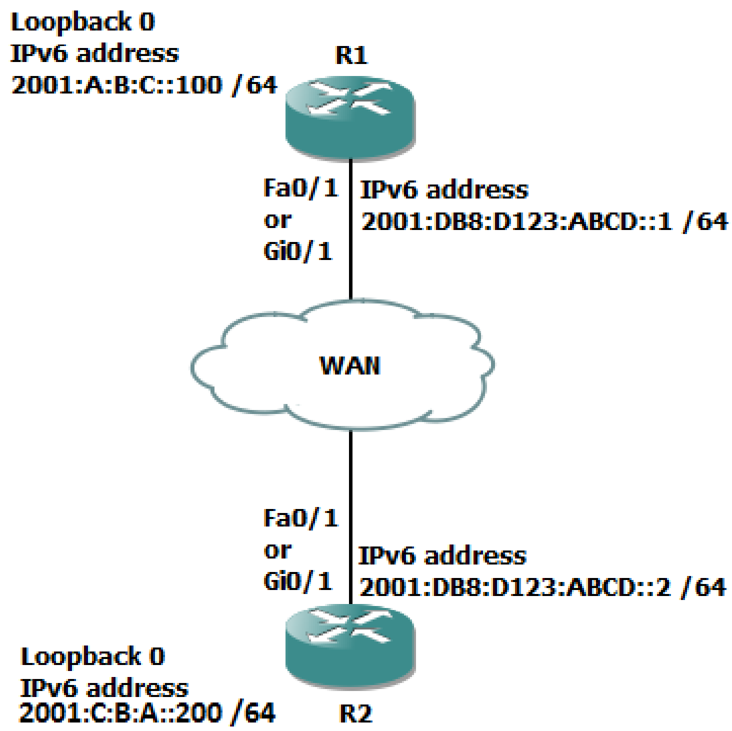
```
R2#
R2#
R2#
R2#
R2#
R2#
R2#
R2#
R2#
R2#
R2#
R2#
R2#
R2#
R2#
R2#
R2#
R2#ping 10.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/32/60 ms
R2#
```

Step 6: Save the configuration on both the switch and router.

Lab 5-1: Configure basic IPv6.

Visual Topology



Command Line

Command	Description
ipv6 address *address / mask*	
ipv6 unicast-routing	
ping *address*	
show ipv6 interface *interface*	
traceroute *address*	

Task 1: Enabling IPv6.

Step 1: Access the CLI on your router and enable IPv6 unicast routing

Step 2: Assign the following IPv6 addresses

R1 only…..

R1(config)#**interface fa0/1**

or

R1(config)#**interface gi0/1**

R1(config-if)#**ipv6 address 2001:DB8:D123:ABCD::1/64**

R1(config-if)#**no shut**

R1(config-if)#**interface loopback 0**

R1(config-if)#**ipv6 address 2001:A:B:C::100/64**

R2 only.....

R2(config)#**interface fa0/1**

or

R2(config)#**interface gi0/1**

R2(config-if)#**ipv6 address 2001:DB8:D123:ABCD::2/64**

R2(config-if)#**no shut**

R2(config-if)#**interface loopback 0**

R2(config-if)#**ipv6 address 2001:C:B:A::200/64**

Step 3: Issue the **show ipv6 interface** command

```
R1
FastEthernet0/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::C000:24FF:FE54:1
  Global unicast address(es):
    2001:DB8:D123:ABCD::1, subnet is 2001:DB8:D123:ABCD::/64
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::5
    FF02::6
    FF02::1:FF00:1
    FF02::1:FF54:1
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
Loopback0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::C000:24FF:FE54:0
  Global unicast address(es):
    2001:A:B:C::100, subnet is 2001:A:B:C::/64
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF00:100
    FF02::1:FF54:0
  MTU is 1514 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is not supported
  ND reachable time is 30000 milliseconds
  Hosts use stateless autoconfig for addresses.
R1#
```

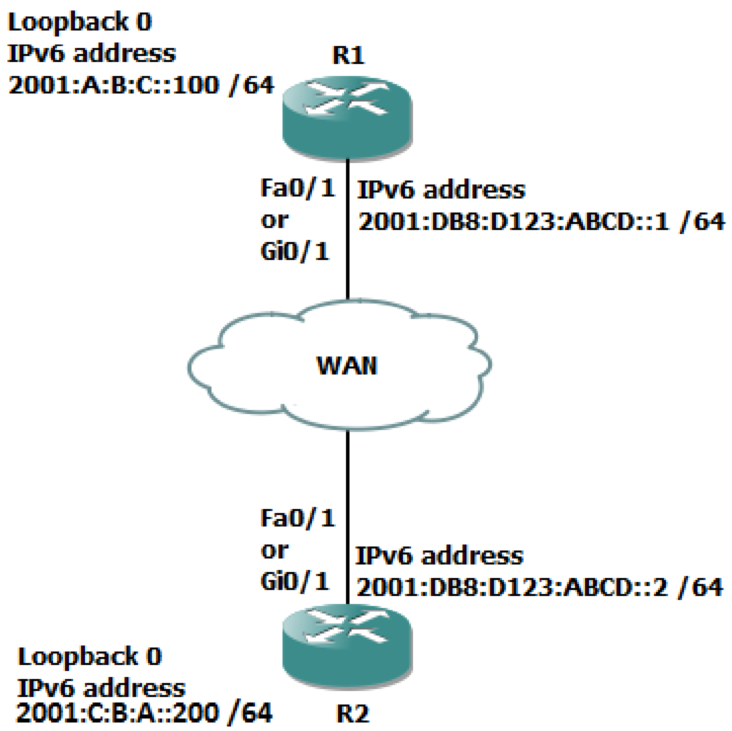
Where did the link-local address come from ?

What are the IPv6 addresses starting with FF02: ?

Step 4: Save your running configurations.

Lab 5-2: Configure IPv6 Routing.

Visual Topology



Command Line

Command	Description
ipv6 ospf *process id* area *id*	Enables OSPFv3 on the interface
ipv6 router ospf *process id*	Enters OSPFv3 router configuration mode
Ping *address*	Checks end to end connectivity
router-id *32bit id*	Assigns a 32 bit router-id in a dotted decimal format, example (1.1.1.1)
show ipv6 ospf	Displays OSPFv3 settings
show ipv6 ospf neighbor	Displays the contents of the OSPF adjacency table
show ipv6 route	Display the contents of the IPv6 routing table.

Task 1: Enable OSPFv3.

Step 1: Access the CLI of the router and check that you still have the IPv6 addresses configured on your router, if not, rectify.

Step 2: Ping the IPv6 address of the other routers fa0/1 or gi0/1 interface. If you have correctly configured both end of the directly connected link, then this should be **successful**.

Step 3: Ping the IPv6 address located on the other router, this should **fail** because it is not directly connected and just like in IPv4 no dynamic routing protocols are enabled by default to advertise it out.

Step 4: Enter OSPFv3 configuration mode and assign the following router ID's

R1 only.....

Router ID 1.1.1.1

R2 only.....

Router ID 2.2.2.2

Do you need to configure unique router ID's ?

Step 5: Enter the configuration mode for the interface directly connecting the two routers together, fa0/1 or gi0/1.

Step 6: Enable OSPFv3 on the interface and check you have an OSPF adjacency.

Step 7: Try pinging the loopback IPv6 address of the other router

Why would it fail?

Step 8: Enable OSPFv3 on the loopback interface and ask the other student to try and ping it, because you are now advertising it via OSPFv3 this should work.

Lab Answer Keys:

Please note that the Answer Keys only provide the Lab steps which require students to enter an answer to a question, or type in a command which isn't explicitly shown in the lab exercise notes.

Running your lab from the Answer Keys section of this lab guide, will result in steps being missed out!

Lab 1-1: Switch Startup and Initial Configuration.

Task 1: Reload and check that the Switch is set to factory defaults.

Step 2: Access the Switch Console port using the method and information provided by the instructor.

At the **Switch>** prompt (if you see any other prompt or are asked for a password contact the instructor), enter the **erase startup-config** command and make a note of the result.

Switch>**erase startup-config**

^

% Invalid input detected at '^' marker.

Why did this fail?

The **erase startup-config** command is not supported in **user mode.**

Step 3: From the user prompt type in the command which enters privilege exec-mode.

Switch>enable

Switch#

Does the system prompt change and if so, how?

Displays **hostname>** in user mode

Displays **hostname#** in privilege mode (sometimes called enable mode)

Now try and execute the **Erase startup-config** command, once again make a note of the output.

Do you see a different console message?

YES

Switch#erase startup-config

Erasing the nvram filesystem will remove all configuration files! Continue? [confirm]

Step 5: Use the appropriate command to verify that the Switch doesn't have a current startup-configuration and use the appropriate show command to display information about the device hardware and software parameters.

Switch#sh startup

startup-config is not present

Switch#sh version

Step 6: Reload the Switch.

Switch#reload

Proceed with reload? [confirm]

Task 2: Defining a hostname and enabling a management IP address.

Step 1: Change the hostname of the Switch to either **SW1** or **SW2**

Switch>enable

Switch#conf t

Switch(config)#hostname SW1

Step 2: Assign your Switch a management IP address from the values identified in the visual topology diagram at the beginning of the lab exercise.

SW1>enable

SW1#conf t

SW1(config)#interface vlan 1

SW1(config-if)#ip address 10.1.1.11 255.255.255.0

or

SW2(config-if)#ip address 10.1.1.12 255.255.255.0

Step 3: Verify connectivity between your PC and the Switch using the **Ping** command, remember that your PC might have a personal firewall installed which could prevent the Switch from Pinging the PC.

Was the Ping successful ?

YES

If not, investigate and correct the problem.

Task 3: Using context-sensitive help.

Step 1: Access the privilege mode on your Switch and enter **?** to list the available commands.

PLEASE NOTE FOR ILLUSRATION PURPOSES THAT THE PROMPT FOR THE REST OF THE DOCUMENT WILL BE SHOWN AS SW or R, THIS WILL DIFFER FROM YOUR OUTPUT.

SW#?

Exec commands:

<1-99>   Session number to resume

clear   Reset functions

clock   Manage the system clock

configure   Enter configuration mode

connect   Open a terminal connection

copy   Copy from one file to another

debug   Debugging functions (see also 'undebug')

delete   Delete a file

dir   List files on a filesystem

disable   Turn off privileged commands

disconnect Disconnect an existing network connection

enable   Turn on privileged commands

erase   Erase a filesystem

exit   Exit from the EXEC

logout   Exit from the EXEC

more   Display the contents of a file

no Disable debugging informations

ping Send echo messages

reload Halt and perform a cold restart

resume Resume an active network connection

setup Run the SETUP command facility

--More--

The above is an example output which may differ from the output you see.

Step 2: Using the **?** navigate through the series of command options to set the system time to the current time and date. Note that the system will support abbreviated commands provided they are unique and using the **Tab** key will automatically complete the command.

SW#clock ?

set Set the time and date

SW#clock set ?

hh:mm:ss Current Time

SW#clock set 20:30:00 ?

<1-31> Day of the month

MONTH Month of the year

SW#clock set 20:30:00 13 aug ?

<1993-2035> Year

SW#clock set 20:30:00 13 aug 2013

Step 3: Use a command to show the current time and date.

SW#show clock

Step 5: To help navigate around the CLI (command line interface) a number of key combinations can be used. Spend a few minutes trying these combinations out and make a note of what they appear to do, for the best result execute a few valid show commands first.

Ctrl P or the **up arrow** key	**Displays previous command entered**
Ctrl A	**Moves the cursor to the front of the command line**

Backspace **Deletes the previous last character**

Task 4: Changing default CLI parameters.

Step 1: Using the **show terminal** command, verify that history is enabled and determine the current history size for the console line.

History is enabled, history size is 10.

Step 2: Use the appropriate command to change the history size to a value of 100 for the console line.

SW#terminal history size 100

Step 4: When accessing the console port there is a default keyboard inactivity timeout of 10 minutes. Change this timer to **60 minutes.**

SW#conf t

SW(config)#line con 0

SW(config-line)#exec-timeout 60

Step 5: What does the **logging synchronous** command do?

Prevents unsolicited messages merging with the commands you type in.

Step 6: Save your running-configuration.

SW#copy run start

Lab 1-2: Troubleshooting Switch Media Issues.

Task 1: Lab setup.

Step 1: Make sure that interfaces fa0/3, fa0/4 and fa0/11 are shutdown on your Switch.

SW#conf t

SW(config)#interface range fa0/3-4, fa0/11

SW(config-if-range)#shutdown

Task 2: Connectivity issues between the PC and the Switch.

Step 2: Enter the correct interface mode for the Switch SVI (management interface) and shut the interface down. Check the IP connectivity between the PC and the Switch, this should now fail.

SW#conf t

SW(config)#int vlan 1

SW(config-if)#shut

Enable the SVI (management interface) and check that IP connectivity has been restored.

SW(config-if)#no shut

Task 3: Connectivity issues between the Switch and the Router.

Step 1: Check that interface fa0/12 on the Switch isn't administratively shutdown, rectify if it is.

SW#sh interface fa0/12

Step 2: Access the console port of the Router using the access method described by the instructor.

Give the router a hostname of R1 or R2

Router#conf t

Router(config)#host R1

or

Router(config)#host R1

Step 3: Enter the interface configuration mode this will be either fa0/0 if you are using a 2811 Router or gi0/0 if you are using a 2901 Router.

Give the interface an IP address of either 10.1.1.1/24 (R1 only) or 10.1.1.2/24 (R2 only).

R1 only.....

R1#conf t

R1(config)#interface fa0/0

or

R1(config)#interface gi0/0

R1(config-if)#ip address 10.1.1.1 255.255.255.0

R1(config-if)#no shut

R2 only.....

R2#conf t

R2(config)#interface fa0/0

or

R2(config)#interface gi0/0

R2(config-if)#ip address 10.1.1.2 255.255.255.0

R2(config-if)#no shut

Try and Ping the IP address of your switch. Was this successful?

The ping should work if the Switch interface and the Routers interface are both up/up

If not check the status of the interface, what do you notice?

Use the sh interface command to display their current status

Rectify the condition and try to Ping the switch again. Only when you have full IP connectivity between the Router and the Switch move on to the next step.

Step 4: Access the interface (fa0/0 or gi0/0) configuration mode on the Router and change the **speed** setting to **10**, now access the CLI on the Switch and enter the interface fa0/12 configuration mode and set the **speed** to **100.**

R(config-if)#speed 10

SW(config-if)#speed 100

Check the status of the interfaces connecting the Switch and Router together, make a note of their layer 1 and layer 2 states.

Use the either sh interface or sh ip interface brief (remember show commands are run from privilege mode)

Would you expect connectivity when there is a speed mis-match?

NO. Layer 1 connections depend on the same speed being used at both ends of the connection.

Reconfigure the Router (interface fa0/0 or gi0/0) to match the speed of the switch, remember best working practice suggests you shutdown the interface before making any changes and after you have reconfigured the interface enter the no shut command.

Verify connectivity before moving on to the next step.

Step 5: Configure Switch interface fa0/12 to half duplex and configure Router interface (fa0/0 or gi0/0) to full duplex.

Switch configuration

SW#conf t

SW(config)#int fa0/12

SW(config-if)#shut

SW(config-if)#duplex half

SW(config-if)#no shut

Router configuration

R#conf t

R(config)#int fa0/0

or

R(config)#int gi0/0

R(config-if)#shut

R(config-if)#duplex full

R(config-if)#no shut

Check the layer 1 and layer 2 status of the connecting interfaces and record your results below.

Duplex mis-matches produce intermittent results but does allow traffic to pass between devices.

Once you are ready to move on, reconfigure Switch interface fa0/12 to full duplex, check IP connectivity and save your running-config on both devices.

#copy run start

Lab 2-1: Router startup and Initial Configuration.

Task 1: Router hardware and software inspection.

Step 2: Use the appropriate command to display the hardware and software properties of the router.

R#sh ver

Fill in the table below.

Step 3: Check the NVRAM for a startup-config file using the **sh start** command and remove the startup-configuration if one exists using the **erase startup-config** command.

R#sh start

R#erase startup-config

Step 4: Issue the **reload** command to restart the router and observe the boot process from the console.

R#reload

confirm

Task 2: Create the initial Router configuration.

Step 1: Skip the initial configuration dialog, terminate the autoinstall and enter privilege EXEC mode.

Router>enable

Router#

Step 2: Set the system hostname to either R1 or R2

Router#conf t

Router(config)#host R1

or

Router(config)#host R2

Does the system prompt change?

YES

R1(config)#

or

R2(config)#

Step 3: Enter the correct configuration mode to add a description to the first ethernet interface on the router (fa0/0 or gi0/0) **Link to LAN Switch.**

R(config)#int fa0/0

or

R(config)#int gi0/0

R(config-if)description Link to LAN Switch

Step 4: Configure an interface IP address and mask with reflects the values shown in the Visual Topology diagram.

R1 10.1.1.1/24

R2 10.1.1.2/24

R1 only.....

R1#conf t

R1(config)#int fa0/0 or **R1(config)#int gi0/0**

R1(config-if)#ip address 10.1.1.1 255.255.255.0

R2 only.....

R2#conf t

R2(config)#int fa0/0 or **R2(config)#int gi0/0**

R2(config-if)#ip address 10.1.1.2 255.255.255.0

What is the status of the interface?

administrative down

Use one of the following commands to determine the status.

R#sh ip int brief

or

R#sh int fa0/0

or

R#sh int gi0/0

The majority of Layer 3 interfaces are shut down by default, whereas layer 2 switch ports are enabled by default.

Do you think we could ping the IP address of the switch?

NO

Take any necessary steps to enable IP connectivity between the Router and the Switch before you move on to the next step.

R(config-if)#no shut

Step 5: Save your running-config to NVRAM

R#copy run start

Task 3: Changing default CLI parameters.

Step 1: Change the EXEC timeout on the console port to a value of 60 minutes

R#conf t

R(config)#line con 0

R(config-line)# (you are now in the console line configuration mode)

R(config-line)#exec-timeout 60

Step 2: Enter the **sh line con 0** command

Does this command verify the new timeout value?

Yes idle exec 01:00:00

Step 3: Improve the readability of the console access by synchronising unsolicited messages and debug outputs with the input of the CLI.

R(config-line)#logging synchronous

Step 4: Use the relevant command which prevents the system from translating a mistyped command to an IP address.

R(config)#no ip domain-lookup

Step 5: Save your running-config to NVRAM

R#copy run start

Task 4: Neighbour discovery using CDP.

Step 1: Using the **sh cdp** command fill in the table below

How often are CDP advertisements being sent	**60 seconds**
How long will a CDP neighbour entry be held in the table without being refreshed.	**180 seconds**
What version of CDP is currently running on your device	**CDP v2**

Step 3: Execute the **sh cdp nei detail** command, do you see any additional information not shown using the command in step 2.

The **detail** option can be used to display layer 3 information such as an IP address.

Lab 2-2: Internet connections.

Task 1: Defining static IP addresses and setting a static default route.

Step 2: Enter the configuration mode of the second Ethernet interface (fa0/1 or gi0/1), place the interface into a disable state and then manually assign an IP address which is listed in the Visual Topology diagram.

R1 only.....

R1(config-if)#ip address 172.16.1.1 255.255.255.0

R1(config-if)#no shut

R2 only.....

R2(config-if)#ip address 172.16.1.2 255.255.255.0

R2(config-if)#no shut

Step 4: Execute the command which allows you to view the contents of the routing table.

R1#sh ip route

or

R2#sh ip route

How many entries would you expect to see? Can you see any remote networks?

2, subnets 10.1.1.0 /24 and 172.16.1.0 /24 both local to the router

The router will calculate these subnets based on the active IP addresses configured on the local interfaces.

No remote networks are present until a static route or dynamic routing protocols are configured.

Task 2: Configure NAT.

Step 2: Configure a standard IP ACL using an ACL id of **1** and permit any device on subnet 10.1.1.0 /24

R1(config)#access-list 1 permit 10.1.1.0 0.0.0.255

or

R2(config)#access-list 1 permit 10.1.1.0 0.0.0.255

This ACL will be used to identify which source IP addresses are going to be translated using NAT, and this example allows any device from the 10.1.1.0 subnet.

Step 3: Create a dynamic NAT address pool, this will hold a list of inside global addresses.

Use this table and parameters on **Router R1 only**

Pool name	**NAT-POOL**
Starting IP address	**192.168.1.1**
Ending IP address	**192.168.1.14**
Network mask	**255.255.255.240**

R1(config)#ip nat pool NAT-POOL 192.168.1.1 192.168.1.14 netmask 255.255.255.240

Use this table and parameters on **Router R2 only**

Pool name	**NAT-POOL**
Starting IP address	**192.168.2.1**
Ending IP address	**192.168.2.14**
Network mask	**255.255.255.240**

R2(config)#ip nat pool NAT-POOL 192.168.2.1 192.168.2.14 netmask 255.255.255.240

Stop.....Have you configured the right set of parameters for your Router!

Step 4: Linking the nat pool to the ACL.

R1(config)#ip nat inside source list 1 pool NAT-POOL

or

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

NB. Nat Pool names are case-sensitive

Step 5: Before any NAT translations occur we must identify at least two interfaces to be our inside and outside.

Source IP address will be translated when traffic traverses between the inside and outside interfaces and destination IP addresses will be translated in the opposite direction between the outside and inside.

Interface fa0/0 or gi0/0 will be our inside interface

Interface fa0/1 or gi0/1 will be our outside interface

Assign the following commands to the relevant interfaces.

The example shown illustrates the commands required on Router R1 which is a 2901 device, you may need to use fa0/0 and fa0/1 if you are using a 2811 device.

R1(config)#int gi0/0

R1(config-if)#ip nat inside

R1(config-if)#int gi0/1

R1(config-if)#ip nat outside

Step6: When we ping from PC1 to R2 or PC2 to R1 our original IP address will be translated from a 10.1.1.x to a 192.168.x.x address (x denotes a variable depending on the direction of the traffic)

Therefore we need to configure a static route back to the 192.168.x.x network.

R1 only...

R1(config)#ip route 192.168.2.0 255.255.255.0 172.16.1.2

R2 only...

R2(config)#ip route 192.168.1.0 255.255.255.0 172.16.1.1

Use the appropriate command to verify that they have been added to the routing table.

R1#sh ip route

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

******some output missing******

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks

C 10.1.1.0/24 is directly connected, GigabitEthernet0/0

L 10.1.1.1/32 is directly connected, GigabitEthernet0/0

172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks

C 172.16.1.0/24 is directly connected, GigabitEthernet0/1

L 172.16.1.1/32 is directly connected, GigabitEthernet0/1

S 192.168.2.0/24 [1/0] via 172.16.1.2

Step 7: We have now configured all of the NAT components and a static route to the translated addresses, the next stage is to test our configuration.

From your PC check you still have a valid **10.1.1.11** or **10.1.1.12** address using **ipconfig/all** from the command shell (cmd).

PC1 will require a default gateway address of 10.1.1.1

and

PC2 will require a default gateway address of 10.1.1.1

Verify and rectify if necessary.

Check you can ping your default gateway from your PC

If you are having problems open the command shell (cmd) and type in the following statements.

On PC1 only...

route -p add 10.1.1.0 mask 255.255.255.0 10.1.1.1

route -p add 172.16.1.0 mask 255.255.255.0 10.1.1.1

On PC2 only...

route -p add 10.1.1.0 mask 255.255.255.0 10.1.1.2

route -p add 172.16.1.0 mask 255.255.255.0 10.1.1.2

Your classroom PC might be fitted with dual interface cards and we need to direct our traffic out of the correct interface.

Step 7: Testing NAT translation.

Traffic will need to traverse across the inside and outside interfaces before any entries will be seen in the IP NAT Translation table.

From you classroom PC ping the IP address of the outside interface on the corresponding router.

Examples

From PC1 ping 172.16.1.2 (this is the outside interface on R2)

From PC2 ping 172.16.1.1 (this is the outside interface on R1)

Check the contents of the IP nat table using the following command.

sh ip nat trans

(remember show commands run from privilege EXEC mode)

Example from R1

R1#sh ip nat trans

Pro	Inside global	Inside local	Outside local	Outside global
icmp	192.168.1.1:10	10.1.1.11:10	172.16.1.2:10	172.16.1.2:10
icmp	192.168.1.1:11	10.1.1.11:11	172.16.1.2:11	172.16.1.2:11
icmp	192.168.1.1:12	10.1.1.11:12	172.16.1.2:12	172.16.1.2:12
icmp	192.168.1.1:9	10.1.1.11:9	172.16.1.2:9	172.16.1.2:9

Task 3: Configure PAT.

Task 1: Removing the previous NAT configuration so we can apply PAT using the same pair of interfaces.

R1 only...

Type in the following commands.

R1(config)#no ip nat pool NAT-POOL 192.168.1.1 192.168.1.14 netmask 255.255.255.240

R1(config)#no ip nat inside source list 1 pool NAT-POOL

R2 only...

Type in the following commands.

R2(config)#no ip nat pool NAT-POOL 192.168.2.1 192.168.2.14 netmask 255.255.255.240

R2(config)#no ip nat inside source list 1 pool NAT-POOL

These commands remove the dynamic pool of addresses used by NAT and the link between the ACL and NAT Pool.

We will still use the existing ACL and IP NAT Inside/outside interface statements when configuring PAT.

Step 2: Configure a dynamic PAT rule which translates your 10.1.1.0 subnet to the IP address configured on the Routers outside interface.

R(config)#ip nat inside source list 1 interface fa0/1 overload

Or

R(config)#ip nat inside source list 1 interface gi0/1 overload

What does the **list 1** part of the command relate to?

Links to the ACL id which identifies which candidates are allowed

What does the key word **overload** do?

Turns PAT on

Step 3: Verifying your configuration by Pinging the IP address of the other Routers outside interface.

Use the appropriate command to view the contents of the IP translation table.

R#sh ip nat translation

Do you see any output differences between the previously configured dynamic NAT pool and the newly configured PAT function? Pay attention to the inside global address!

The inside global address will be the same value for all translations.

Describe the following NAT/PAT terms

Inside local

Original address on the inside interface (usually private)

Inside global

Translated address which appears to the outside network (usually public)

Outside global

Address we are trying to contact (usually public)

Outside local

A destination address which is hidden behind a NAT or PAT function (usually private)

Step 4: Once you are satisfied that PAT is configured correctly we can now remove it from the system, remember to remove all components and you will need to be in the right configuration mode to execute these commands.

R(config-if)#no ip nat inside

R(config-if)#no ip nat outside

R(config)#no access-list 1

R(config)#no ip nat inside source list 1 interface fa0/1 overload

or

R(config)#no ip nat inside source list 1 interface gi0/1 overload

Step 5: Shutdown the fa0/1 or gi0/1 interface.

fa0/1 if you are using a 2811 router and gi0/1 if you are using a 2901 router

R(config-if)#shut

Step 6: Save your current configuration.

R#copy run start

Lab 3-1: Improving Device Security.

Task 1: Device password protection.

Step 1: Access the console port of the router.

R#conf t

R(config)#line con 0

R(config-line)#

Step 2: Secure the console port with the password **cisco**

R(config-line)#password cisco

R(config-line)#login

(please don't use any maverick passwords, only those specified in the lab instructions and passwords are case-sensitive)

Step 3: Verify your password by exiting from the line con 0 mode and then the user EXEC mode using the **End** and **Exit** commands.

Step 4: Enter the console password to return to user EXEC mode.

Step 5: Create a local user account with a username of **ccna** and a secret password of **cisco**

R(config)#username ccna secret cisco

Step 6: Change the security method used on the console port to now prompt the administrator for a username and password.

R(config)#line con 0

R(config-line)#login local

Important...Step 5 must be completed before Step 6 otherwise you will lock yourself out of the system.

Step 7: Verify the security change by logging out of the system, you will notice this time you are asked for a username and password instead of just a password which doesn't identify the person accessing the system.

Task 2: Remote access using Telnet and SSH.

Using telnet (insecure) and ssh (secure) protocols allow administrators to access their devices remotely, providing IP connectivity exists between the telnet/ssh client and the telnet/ssh server.

In this task we are going to configure our router to support telnet and ssh sessions via the VTY lines.

Step 1: Access the router CLI and navigate to the VTY configuration mode, enter a command which forces the administrator to provide a username and password.

R(config)#line vty 0 4

R(config-line)#login local

Hint: Used on the console port in the previous task.

Step 3: Telnet provides a method of remote administration but unfortunately when you type in the authentication details, the username and password are sent in clear text therefore telnet should only be used on trusted interfaces or via a VPN encrypted tunnel. SSH version 2 is the preferred method because it can provide authentication and data protection via an encrypted channel.

To configure SSH we need to setup a domain name, generate our RSA public/private key pairs, recommended we support only version 2 and make sure that the VTY lines also support the SSH protocol.

Run the following commands from global configuration mode.

R(config)#IP domain-name cisco.com

R(config)#crypto key generate rsa

change the modulus size to **1024** when prompted (ssh version 2 needs a minimum modulus size of 768 bits or greater)

R(config)#IP ssh version 2

Optional, if you decided that your company policy dictates that all remote connections must be secure then you could use the following command on the VTY lines.

R(config)#line vty 0 4

R(config-line)#transport input ssh

This command disables all other protocols excluding ssh

Step 4: Check that ssh is working by opening a PuTTy session to your router from your PC but this time under the connection type select the **ssh** radio button, type in the IP address of the router and open the connection.

If you receive the following security alert click **yes**

Login as requested using the local account credentials

Username **ccna** password **cisco**

Enter privilege mode

R>enable

Execute the following two show commands

R#sh users

R#sh ssh

The output states that we are accessing the system via line VTY 0 and ssh version is being used.

If you decided to try the optional **transport input ssh** please reset this back to its default condition, **transport input all** once again enabling telnet and ssh at the same time.

R(config)#line vty 0 4

R(config-line)#transport input all

What are the ports numbers used by telnet and ssh?

Telnet **TCP 23**

SSH **TCP 22**

What command generates the Public/Private key pairs and what was the default modulus size?

R(config)#crypto key generate rsa

Default modulus **512 bits** (Cisco now recommend a minimum of 2048 bits)

Task 3: Limiting remote access based on source IP addresses.

It is possible to restrict which host or subnet a device is on when managing remote access. Standard IP ACLs can be used to identify the source IP address of a ssh or telnet client.

Step 2: Check the IP address of your PC and make a note of it below.

ipconfig /all

Step 3: Access the CLI on the router and navigate to the global configuration mode, this is where you will need to create a standard IP ACL which allows only your PC to telnet or ssh into the router, please use an ACL id of 2.

R(config)#access-list 2 permit 10.1.1.101 0.0.0.0

or

R(config)#access-list 2 permit 10.1.1.102 0.0.0.0

What does the wildcard mask **0.0.0.0** do ? Can you think of an alternative way of writing the ACL.

Matches all 32 bits to identify an individual host.

Step 4: Apply this ACL to the VTY lines using the appropriate command, use the command list if you are unsure.

R(config)#line vty 0 4

R(config-line)#access-class 2 in

Step 6: Change the IP address on your PC

PC1 10.1.1.133/24

PC2 10.1.1.144/24

Now try and telnet or ssh to your router, should you be successful?

NO the ACL will block your access

Task 4: Creating a login banner page.

Most systems have a mandatory security message displayed to anybody accessing the system.

Step 1: Access the router CLI and create a login message which advises only authorised users are permitted to access the system.

Example

R(config)#banner login # authorised access only! #

Hint: Remember to use delimiting characters to identify the beginning and end of the displayed message.

Step 3: Save your running-config

R#copy run start

Lab 3-2: Device Hardening.

Task 1: Managing unused ports on a Switch.

Step 2: Ensure that the interface connecting your switch to your Router is enabled by using the **no shutdown** command, remember to be in the right configuration mode?

SW>en

SW#conf t

SW(config)#int fa0/12

SW(config-if)#no shut

Task 2: Using Switchport port-security.

Step 2: Identify the MAC address of the interface used to connect to your switch.

What command could be used to achieve step 2 ?

R#sh int fa0/0

or

R#sh int gi0/0

Once you know the MAC address of the interface make a note of it below.

MAC Address:

Step 3: Enter the configuration mode which will allow you to change the interface parameters on Fa0/0 or Gi0/0.

R#conf t

R(config)#int fa0/0

or

R(config)#int gi0/0

Shutdown the interface and set the MAC address to the following value using these commands:

R(config-if)#Shut

R(config-if)#mac-address 0000.0C12.ABCD

Step 4: Enable the interface and generate some traffic by pinging the IP address of your switch.

R(config-if)#no shut

Step 5: Access the switches CLI.

What command could we use to check the MAC address of the attached router?

SW#sh mac-address

Step 6: While still accessing the CLI on the switch, shutdown the interface fa0/12

SW#conf t

SW(config)#int fa0/12

SW(config-if)#shut

Step 7: Enable switchport security using the following commands:

SW(config-if)switchport mode access

SW(config-if)switchport port-security

Why have we used the first command **switchport mode access** and does it disable DTP ?

Switchport port security is not supported over trunk connections

NO it doesn't disable DTP

Step 8: Enable interface fa0/12 and generate some traffic between the switch and the router.

SW(config-if)#no shut

The connection between the switch and the router should be **up/up** use the **show interface fa0/12** to verify this.

SW(config-if)#end

SW#show int fa0/12

What are the default values used by the system when the **switchport port-security** command is applied to an interface ?

Max of 1 supported MAC address with a violation policy of shutdown

Step 9: Access the CLI on the router and shutdown the interface connected to your switch.

R(config)#int fa0/0

or

R(config)#int gi0/0

R(config-if)#shut

Step 10: On the interface connected to your switch reset the MAC address to its original value.

R(config-if)default mac-address

or

R(config-if)no mac-address

Step 11: Enable the interface and generate some traffic between the router and the switch.

R(config-if)no shut

Step 12: Access the CLI on the switch and check the status of the interface connected to the router. You should now observe a **down/down (err-disabled)** state due to the port violation which occurred when you changed the MAC address of the router.

Resetting the violated port

SW(config)#int fa0/12

SW(config-if)#shut

SW(config-if)#no shut

Step 13: Remove the switchport port-security and check you once again have connectivity between the switch and the router.

SW(config-if)#no switchport port-security

Task 3: Disable unused services.

Step 2: Run the appropriate command to view the CDP neighbours table (remember to use the American spelling)

SW#sh cdp nei

Is CDP running? **YES**

Step 3: Disable CDP on the interface connected to the router and issue the command used in step 2.

SW#conf t

SW(config)#int fa0/12

SW(config-if)#no cdp enable

Do you see a difference ? **YES**

Remember the default timeout value for CDP is 180 seconds.

Step 4: Enable CDP on the interface.

SW(config-if)#cdp enable

Step 5: Save your configuration.

SW(config-if)end

SW#copy run start

Lab 3-3: Using ACLs to filter IP based traffic

Task 1: Configuring an ACL (Access Control List).

Step 2: Create a loopback interface using the following commands.

R1 only...

R1>en

R1#conf t

R1(config)#interface loopback0

R1(config-if)#ip address 1.1.1.1 255.255.255.255

R1(config-if)#no shut

Note we are using a 32 bit mask.

R2 only...

R2>en

R2#conf t

R2(config)#interface loopback0

R2(config-if)#ip address 2.2.2.2 255.255.255.255

R2(config-if)#no shut

Step 3: From your PC check that you can ping the IP address of the loopback interface you have just created and also your default gateway. **This should work!**

If the ping fails check the following.

Does your PC have the correct IP address ? hint... ipconfig

Can you ping your default gateway ? If not check that the router interface is up/up

Step 4: Create an extended ACL named PING that will prevent your PC from successfully pinging the default gateway, however you should be allowed to ping the loopback interface and all other IP traffic should be permitted through the router.

R1 only....

R1#conf t

R1(config)#ip access-list extended PING

R1(config-ext-nacl)#deny icmp 10.1.1.101 0.0.0.0 10.1.1.1 0.0.0.0 echo

R1(config-ext-nacl)#permit ip any any

R1(config-ext-nacl)#int fa0/0

or

R1(config-ext-nacl)#int gi0/0

R1(config-if)ip access-group PING in

R2 only....

R2#conf t

R2(config)#ip access-list extended PING

R2(config-ext-nacl)#deny icmp 10.1.1.102 0.0.0.0 10.1.1.2 0.0.0.0 echo

R2(config-ext-nacl)#permit ip any any

R2(config-ext-nacl)#int fa0/0

or

R2(config-ext-nacl)#int gi0/0

R2(config-if)ip access-group PING in

Step 5: Test your ACL and use the appropriate show commands to display the content of the configured ACL and write down your results below.

Rx#**show access-list PING**

Are access-list names case sensitive ?

YES

Step 6: Once you have proved that the ACL works correctly then remove it from your configuration.

R(config-if)#no ip access-group PING in

R(config-if)#exit

R(config)#no ip access-list ext PING

Step 7: Save your configuration

R(config)#exit

R#copy run start

Lab 4-1: Enhancing a Switched Network.

Task 1: Creating a VLAN and assigning Switchports.

Step 2: Access the CLI on the switch.

Step 3: Create vlan **2** on your switch and name it **SALES**

SW>en

SW#conf t

SW(config)#vlan 2

SW(config-vlan)#name SALES

Step 4: Re-assign interface fa0/1 to vlan 2

SW(config-vlan)#int fa0/1

SW(config-if)#switchport mode access

SW(config-if)#switchport access vlan 2

Step 5: Configure Fa0/12 as a trunk connection

SW(config-if)#int fa0/12

SW(config-if)#switchport mode trunk

Step 7: From your PC try and Ping the IP address of your switch.

This should fail! Why ?

Because the IP address of the switch is assigned to Vlan 1 and the port connected to the PC is now in Vlan 2 (logically separated) Routing is required between the two Vlan's.

Task 2: Configure a Trunk connection on a Router.

Now that we have configured the switch to support a trunk connection between itself and the router, the next stage involves us setting up the router so it understands the IEEE 802.1Q frame encapsulation.

Step 1: Access the CLI on the router.

Step 2: Navigate to the interface mode which connects the router to the switch. Hint.. fa0/0 or gi0/0

R>en

R#conf t

R(config)#int fa0/0

or

R(config)#int gi0/0

Step 3: Shutdown the interface.

R(config-if)#shut

Step 4: Remove any current IP address using the **no ip address** command.

R(config-if)#no ip address

Step 5: Create a new sub-interface using the following command.

R(config)interface fa0/0.1

or

R(config)interface gi0/0.1

Step 6: Assign an ip address of 10.1.1.1 255.255.255.0

R(config-subif)#ip address 10.1.1.1 255.255.255.0

Step 7: Issue the following command to support IEEE 801.1Q encapsulation linking it to vlan 1 and make this the native vlan.

R(config-subif)#encap dot1q 1 native

Step 8: Create a second sub-interface

R(config)#interface fa0/0.2

or

R(config)#interface gi0/0.2

Step 8: Assign an ip address of 10.2.2.1 255.255.255.0

R(config-subif)#ip address 10.2.2.1 255.255.255.0

Step 9: Setup IEEE 802.1Q encapsulation with a link to vlan 2

R(config-subif)#encap dot1q 2

What is the difference between the native vlan and a non-native vlan ?

Native is untagged across a trunk connection, default id vlan 1.

Step 10: Issue the **no shutdown** command on the physical interface, this will automatically enable all sub-interfaces.

R(config-subif)#exit

R(config)#int fa0/0

or

R(config)#int gi0/0

R(config-if)#no shut

Step 11: Check your PC can ping its default gateway.

Step 12: Try and ping the IP address of the switch, this should now be successful.

If it fails then check the following.

The switch will need a default gateway set to 10.1.1.1 because the path of the ping from the PC to the switch is via the router.

Explanation: The port attached to the PC has been assigned to Vlan 2 but the IP address of the switch is still in vlan 1, therefore the Ping packet will travel from the PC to its default gateway (the router) because the source IP address and the destination IP address are not located in the same IP subnet. The router upon receiving the Ping packet will direct it to sub-interface fa0/0.2 or gi0/0.2 because it's been linked to vlan 2, it will then examine the destination IP address after stripping the layer 2 header and redirect it out of sub-interface fa0/0.1 or gi0/0.1 but it will need to rebuild a new layer 2 header before going across the trunk to the switch.

Lab 4-2: Using a Router to provide DHCP Services.

Task 1: Setting-up DHCP address Pools

Step 2: Configure a DHCP pool named SALES

R>en

R#conf t

R(config)#ip dhcp pool SALES

Step 3: Adding the IP network/subnet to the DHCP pool

R(dhcp-config)#network 10.2.2.0 255.255.255.0

Step 4: Adding the default gateway and dns server options to the DHCP pool

R(dhcp-config)#default-router 10.2.2.1

R(dhcp-config)#dns-server 8.8.8.8

R(dhcp-config)#exit

Task 2: Excluding static IP addresses from a DHCP Pool.

Step 1: Limiting the scope of the DHCP pool

R(config)#ip dhcp excluded-address 10.2.2.1 10.2.2.9

R(config)#ip dhcp excluded-address 10.2.2.21 10.2.2.254

Task 3: Testing the DHCP Service.

Step 2: From the PC command prompt verify your IP address.

IPconfig /all

Step 3: Access the CLI on the router and execute the following command to display current IP address allocation.

R(config)#exit

R#show ip dhcp binding

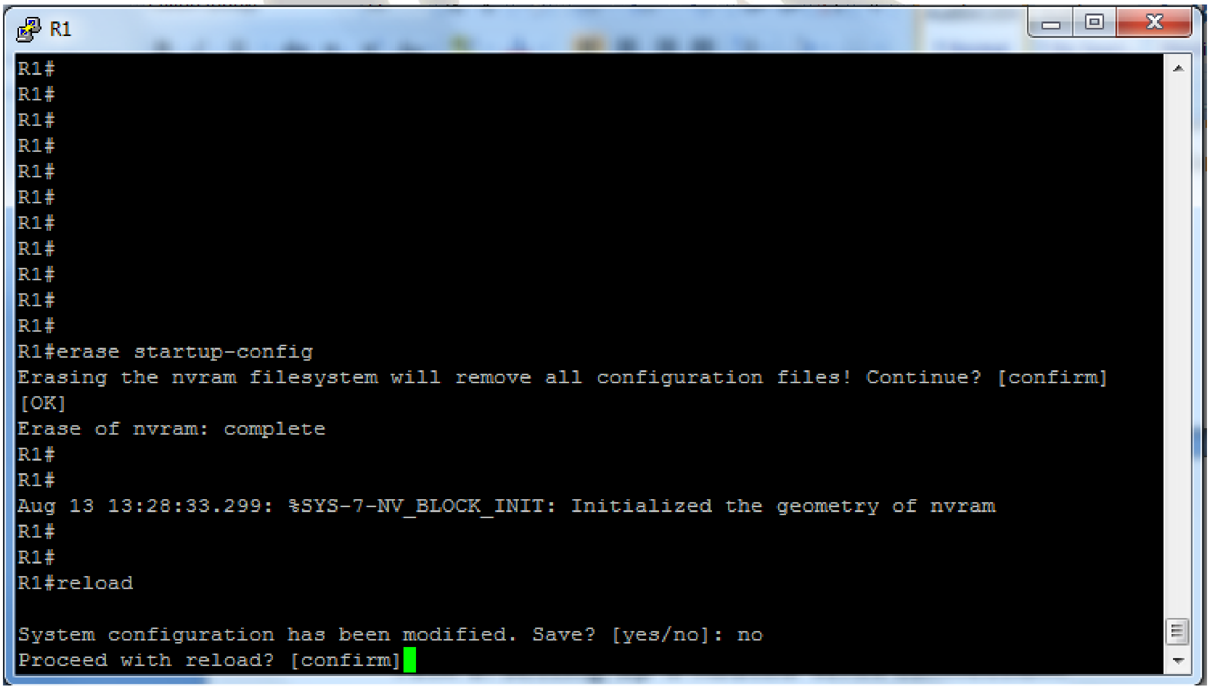
Lab 4-3: Implementing OSPF.

This lab exercise requires two students to work together to complete the tasks. Use the visual topology diagram to ascertain the correct IP addressing plan for your PC, Router and Switch.

Task 1: Setting-up a Routed WAN connection.

Step 1: Access the CLI on both your switch and router.

Step 2: Clear down their current configuration and reload the devices using the **erase startup-config** and **reload** commands. Make sure you do this on both the router and the switch.



```
R1
R1#
R1#
R1#
R1#
R1#
R1#
R1#
R1#
R1#
R1#
R1#
R1#erase startup-config
Erasing the nvram filesystem will remove all configuration files! Continue? [confirm]
[OK]
Erase of nvram: complete
R1#
R1#
Aug 13 13:28:33.299: %SYS-7-NV_BLOCK_INIT: Initialized the geometry of nvram
R1#
R1#
R1#reload

System configuration has been modified. Save? [yes/no]: no
Proceed with reload? [confirm]
```

If the system indicates that the system configuration has been modified and do you want to save ? Answer **no**

Confirm the reload.

Step 3: The devices have been set back to factory defaults (well almost) ignore and abort the setup dialogue options.

Using the information in the visual topology diagram, setup the correct IP addresses and hostnames on all 3 devices.

Hint....

Switch>en

Switch#conf t

Switch(config)#hostname SW1

SW1(config)#interface vlan 1

SW1(config-if)#ip address 10.1.1.10 255.255.255.0

SW1(config-if)#no shut

Router>en

Router#conf t

Router(config)#hostname R1

R1(config)#interface fa0/0

R1(config-if)#ip address 10.1.1.1 255.255.255.0

R1(config-if)#no shut

R1(config-if)#interface fa0/1

R1(config-if)#ip address 172.16.1.17 255.255.255.240

R1(config-if)#no shut

Check that the interfaces are **up/up**, troubleshoot any discrepancies.

Remember to set a static IP address on the PC interface and check connectivity between the PC and its default router.

Task 2: Configuring OSPF.

By default routers do not run any dynamic routing protocols, however routing between IPv4 locally connected interfaces is enabled by default.

Step 1: Execute the relevant **show** command to display the contents of the routing table.

R#sh ip route

The table displays only two connected subnets and therefore this router only has paths for subnets 10.1.1.0 and 172.16.1.16.

Step 2: Enter OSPF configuration mode and use a process id of 1.

R(config)#Router ospf 1

Step 3: Enable both interfaces for ospf and place them in the backbone area 0

R1 only.....

R1(config-router)#network 10.1.1.1 0.0.0.0 area 0

R1(config-router)#network 172.16.1.17 0.0.0.0 area 0

R2 only.....

R2(config-router)#network 10.2.2.1 0.0.0.0 area 0

R2(config-router)#network 172.16.1.18 0.0.0.0 area 0

Step 4: Check the contents of the routing table. Do you see any additional entries?

Yes you now see an OSPF advertised route from the neighbouring router

Step 6: Save the configuration on both the switch and router.

R#copy run start

Lab 5-1: Configure basic IPv6.

Task 1: Enabling IPv6.

Step 1: Access the CLI on your router and enable IPv6 unicast routing

R>en

R#conf t

R(config)#ipv6 unicast-routing

Step 2: Assign the following IPv6 addresses

R1 only.....

R1(config)#interface fa0/1

or

R1(config)#interface gi0/1

R1(config-if)#ipv6 address 2001:DB8:D123:ABCD::1/64

R1(config-if)#no shut

R1(config-if)#interface loopback 0

R1(config-if)#ipv6 address 2001:A:B:C::100/64

R2 only.....

R2(config)#interface fa0/1

or

R2(config)#interface gi0/1

R2(config-if)#ipv6 address 2001:DB8:D123:ABCD::2/64

R2(config-if)#no shut

R2(config-if)#interface loopback 0

R2(config-if)#ipv6 address 2001:C:B:A::200/64

Step 3: Issue the **show ipv6 interface** command

R(config-if)#end

R#sh ipv6 interface

Where did the link-local address come from ?

Automatically assigned by the system

What are the IPv6 addresses starting with FF02: ?

IPv6 multicast addresses

Step 4: Save your running configurations.

R#copy run start

Lab 5-2: Configure IPv6 Routing.

Task 1: Enable OSPFv3.

Step 1: Access the CLI of the router and check that you still have the IPv6 addresses configured on your router, if not, rectify.

R>en

R#sh ipv6 int brief

Step 2: Ping the IPv6 address of the other routers fa0/1 or gi0/1 interface. If you have correctly configured both end of the directly connected link, then this should be **successful**.

Step 3: Ping the IPv6 address located on the other router, this should **fail** because it is not directly connected and just like in IPv4 no dynamic routing protocols are enabled by default to advertise it out.

Step 4: Enter OSPFv3 configuration mode and assign the following router ID's

R1 only.....

R1#conf t

R1(config)#ipv6 router ospf 1

R1(config-router)#Router-ID 1.1.1.1

R2 only.....

R2#conf t

R2(config)#ipv6 router ospf 1

R2(config-router)#Router-ID 2.2.2.2

Do you need to configure unique router ID's ?

Yes, each ospf router needs a unique router ID

Step 5: Enter the configuration mode for the interface directly connecting the two routers together, fa0/1 or gi0/1.

R(config-router)#int fa0/1

or

R(config-router)#int gi0/1

R(config-if)#

Step 6: Enable OSPFv3 on the interface and check you have an OSPF adjacency.

R(config-if)#ipv6 ospf 1 area 0

R(config-if)#exit

R#sh ipv6 ospf nei

Step 7: Try pinging the loopback IPv6 address of the other router

Why would it fail?

OSPF is not advertising the loopback interfaces

Step 8: Enable OSPFv3 on the loopback interface and ask the other student to try and ping it, because you are now advertising it via OSPFv3 this should work.

R#conf t

R(config)#int loop 0

R(config-if)# ipv6 ospf 1 area 0