STATIC ROUTING

In this lab you will study the configuration of a network for static routing. We will also see a type of denial of service attack.

We have added 2 appendices to this lab that will **help** you with the experiment set ups. APPENDIX A shows you how you can quickly input command lines in the console window of routers, PCs and VMs. APPENDIX B illustrates how you can save a router configuration so that when you stop a GNS3 project and come back to it later, it will have stored all the values and settings for you.

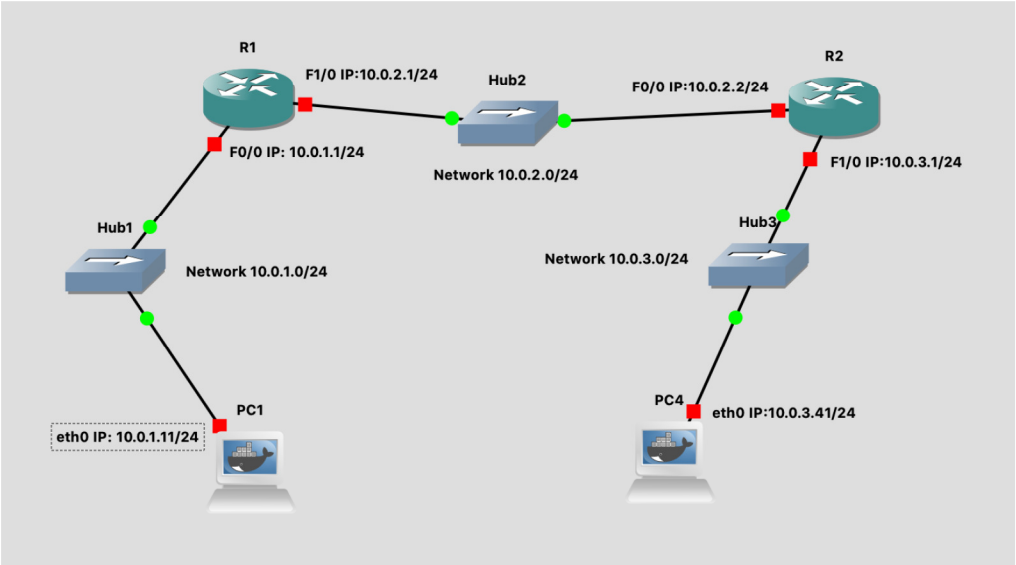


Figure 1 Network Topology

PCs	**eth0**	
PC1	10.0.1.11 / 24	
PC4	10.0.3.41 / 24	
Cisco Router	**FastEthernet0/0**	**FastEthernet1/0**
R1	10.0.1.1 / 24	10.0.2.1 / 24
R2	10.0.2.2/24	10.0.3.1/24

Table 1 IP addresses for Parts 1-4

PART 1. Configuring a CISCO Router

The setup of the Cisco router is more involved. There are different ways to connect to a Cisco router such as by the Serial or Ethernet ports or connections. The first step is to start the router in GNS3, and then open the console window so that the configuration commands can be entered. Once in the console you have to type IOS commands using the command line interface of IOS. The network setup for this part is as shown in Figure 1 and Table 1.

Exercise 1(A). Switching Cisco IOS Command Modes

This exercise demonstrates how to log into a router and how to work with the different Cisco IOS command modes. It is important to understand the different modes so you know where you are and what commands are accepted at any time.

1. Connect the Ethernet interfaces of the PCs and the Cisco router as shown in Figure 1. Do not turn on the PCs yet.

2. Right-click on router R1 and choose `Start`.

3. Right-click on router R1 and choose `Console`. Wait a few seconds until the router is initialized. If everything is fine, you should see the prompt shown below. This is the User EXEC mode. If the prompt does not appear, try to restart GNS3 and repeat the setup again.

   ```
   R1>
   ```

4. To see which commands are available in this mode, type "`?`":

   ```
   R1> ?
   ```

5. To view and change system parameters of a Cisco router, you must enter the Privileged EXEC mode by typing:

   ```
   R1> enable
   R1#
   ```

6. Type the following command to disable the Privileged EXEC mode

   ```
   R1# disable
   R1>
   ```

NOTE
The Cisco routers in GNS3 sometimes start up in Privileged instead of User EXEC mode. There is no explanation as to why that happens.

7. To modify system wide configuration parameters, you must enter the global configuration mode. This mode is entered by typing:

   ```
   R1> enable
   R1# configure terminal
   R1(config)#
   ```

Tip:
Almost all terminal commands can be reduced to shorter commands.
Example: `configure terminal` can be reduced to `conf t`

8. To make changes to a network interface, enter the interface configuration mode, with the command:

```
R1(config)# interface FastEthernet0/0
R1(config-if)#
```

   The name of the interface is provided as an argument. Here, the network interface that is configured is `FastEthernet0/0`.

9. To return from the interface configuration to the global configuration mode, or from the global configuration mode to the Privileged EXEC mode, use the `exit` command:

```
R1(config-if)# exit
R1(config)# exit
R1#
```

   The `exit` command takes you one step up in the command hierarchy. To directly return to the Privileged EXEC mode from any configuration mode, use the `end` command:

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

10. To terminate the console session from the User EXEC mode, type `logout` or `exit`:

```
R1# disable
R1> logout
R1 con0 is now available
Press RETURN to get started.

R1> exit
R1 con0 is now available
Press RETURN to get started.
```

Exercise 1(B). Configuring a Cisco Router via the console

The following exercises show the basic Cisco IOS commands that are used to configure a Cisco router.

1. Right-click on R1 and choose `Start`.

2. Right-click on R1 and choose `Console`. Wait some seconds until the initial console window is set up. When the router is ready to receive commands, proceed to the next step.

3. Configure R1 and R2 with the IP addresses given in Table 1. Below we show how to configure R1. Follow same steps for R2 with appropriate IP addresses.

IOS MODE: GLOBAL CONFIGURATION

```
ip routing
no ip routing
```

Enables or disables IP forwarding. When it is disabled, it also deletes the content of the routing table.

IOS MODE: INTERFACE CONFIGURATION

```
no shutdown
shutdown
```

Enables or disables, respectively, a network interface.

```
R1> enable
R1# configure terminal
R1(config)# no ip routing
R1(config)# ip routing
R1(config)# interface FastEthernet0/0
R1(config-if)# ip address 10.0.1.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# interface FastEthernet1/0
R1(config-if)# ip address 10.0.2.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# end
```

Tip:
"no ip routing" is used to guarantee that the routing cache is empty, not the routing table.

4. When you are done, use the following commands to check the changes you made to the router configuration:

```
R1# show interfaces
R1# show running-config
```

Exercise 1(C). Setting static routing table entries on a Cisco router

In this exercise, you will add static routes to the routing table of R1. The routing table must be configured so that it conforms to the network topology shown in Figure 1 and Table 1. The routes are configured manually, which is also referred to as *static routing*.

The IOS command to configure static routing is **`ip route`**. The command can be used to show, clear, add, or delete entries in the routing table. The commands are summarized in the list below.

IOS MODE: PREVILEGED EXEC

`show ip route`
Displays the contents of the routing table.

`clear ip route *`
Deletes all routing table entries.

`show ip cache`
Displays the routing cache.

IOS MODE: GLOBAL CONFIGURATION

`ip route` **`destination_prefix mask gw_address`**
`no ip route` **`destination mask gw_address`**

Adds or deletes a static routing table entry to **`destination`** with netmask **`mask`**. The argument **`gw_address`** is the IP address of the next-hop router.

`ip route` **`0.0.0.0 0.0.0.0 gw_address`**
`no ip route` **`0.0.0.0 0.0.0.0 gw_address`**

Adds or deletes a default routing table entry to a gateway where **`gw_address`** is the IP address of the next-hop router

`ip route` **`destination mask Iface`**
`no ip route` **`destination mask Iface`**

Adds or deletes a static routing table entry to **`destination`** with netmask **`mask`**. Here, the next-hop information is the name of a network interface (e.g. `FastEthernet0/0`).

Next we show some examples for adding and deleting routing table entries in Cisco IOS. Note that whenever an IP address is configured for a network interface on a router, routing table entries for the directly connected network are added automatically.

The command for adding a route on R1 for the network address `10.0.1.0/24` with `10.0.2.22` as the next-hop gateway IP address is

```
R1(config)# ip route 10.0.1.0 255.255.255.0 10.0.2.22
```

NOTE
This is very important because if you do not set up the IP routes between the routers, the routers will never be able to ping each other from remote networks.

The command below shows you how to add a host route to a host with IP address `10.0.2.65` with next-hop (gateway) set to `10.0.1.21`. In IOS, a host route is identified by a 32bit prefix.

```
R1(config)# ip route 10.0.2.65 255.255.255.255 10.0.1.21
```

The command to add e.g. the IP address `10.0.4.4` as the default gateway is done with the command

```
R1(config)# ip route 0.0.0.0 0.0.0.0 10.0.4.4
```

Finally, to delete any specific entry use the `no ip route` command. For example:

```
R1(config)# no ip route 10.0.1.0 255.255.255.0 10.0.2.22
R1(config)# no ip route 10.0.2.65 255.255.255.255 10.0.1.21
R1(config)# no ip route 0.0.0.0 0.0.0.0 10.0.4.4
```

1. Display the content of the routing table with `show ip route`. Note the routing entries that are already present. Save the output.

2. Add routing entries to R1 and R2, so that the routers forward datagrams and operate correctly for the configuration shown in Figure 1. Routing entries should exist for the following networks in each router (either directly connected or via a nexthop/gateway).

   a) `10.0.1.0/24`
   b) `10.0.2.0/24`
   c) `10.0.3.0/24`

3. Display the routing table again with `show ip route` and save the output.

Lab Questions

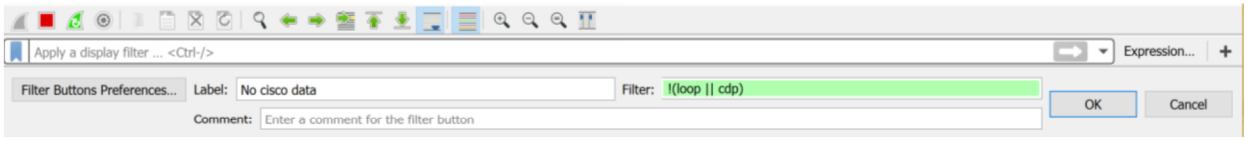
- Explain the fields of the routing table entries of the Cisco router.

PART 2. Configuring a PC with static routes

Exercise 2(A). Network setup

1. Start all the PCs on GNS3. Then, configure the IP addresses of the interfaces as given in Table 1.

2. Start Wireshark to capture traffic on PC1 link.
    a. Before you send ping commands, let's save a filter to the Wireshark capture window as discussed in Part 3 of Lab 1.
    b. Open the add filter toolbar to add the expression "`!(loop || cdp)`" and label it something simple like "No cisco data" or something along those lines. Save the expression.



    c. This will filter out the loop and cdp lines that the Cisco routers send periodically and can fill up your Wireshark capture data easily. Anytime we use routers in these labs be sure to add this command in.

3. Issue a `ping` command from PC1 to R1, R2 and PC4, respectively.

```
PC1% ping 10.0.1.1 -c 5
PC1% ping 10.0.2.2 -c 5
PC1% ping 10.0.3.41 -c 5
```

4. Save the captured Wireshark output.

Interesting things to study (if you are interested!)

Use the saved data to answer the following questions:

- What is the output on PC1 when the ping commands are issued?
- Which packets, if any, are captured by Wireshark?
- Are some of the destinations not reachable? If yes, which ones?

Exercise 2(B). Setting static routing table entries for a PC

Next, you will set up the routing tables of the PCs. The routing tables are configured so that they conform to the network topology shown in Figure 1 and Table 1.

Configuring static routes in Linux is done with the command `route`, which has numerous options for viewing, adding, deleting, or modifying routing entries. The various uses of the `route` command are summarized in the list below.

In Linux, there is no simple way to delete all entries in the routing table. When the commands are issued interactively in a Linux shell, the added entries are valid until Linux is rebooted. To

make static routes permanent, the routes need to be entered in the configuration file `/etc/sysconfig/static-routes`, which is read each time Linux is started.

`route add -net` **`netaddress`** `netmask` **`mask`** `gw` **`gw_address`**
`route add -net` **`netaddress`** `netmask` **`mask`** `dev` **`iface`**

Adds a routing table entry for the network prefix identified by IP address **`netaddress`** and netmask **`mask`**. The next-hop is identified by IP address **`gw_address`** or by interface **`iface`**.
Example: The command for adding a route for the network address 10.21.0.0/16 with next-hop address 10.11.1.4 is:
`route add -net` **`10.21.0.0`** `netmask` **`255.255.0.0`** `gw` **`10.11.1.4`**

`route add -host` **`hostaddress`** `gw` **`gw_address`**
`route add -host` **`hostaddress`** `dev` **`iface`**

Adds a host route entry for IP address **`hostaddress`** with the next-hop identified by IP address **`gw_address`** or by interface **`iface`**.

`route add default gw` **`gw_address`**

Sets the default route to IP address **`gw_address.`**

`route del -net` **`netaddress`** `netmask` **`mask`** `gw` **`gw_address`**
`route del -host` **`hostaddress`** `gw` **`gw_address`**
`route del default gw` **`gw_address`**

Deletes an existing route from the routing table with specific arguments.

`route -e`

Displays the current routing table with extended fields. The command is identical to the `netstat -r` command.

`ip route flush table main`

deletes all entries in the routing table on a PC

`ip route flush cache`

deletes all entries in a routing cache on a PC

Please note that when you flush the routing table, you cant add any entries to the routing table again. You need to STOP the PC and then restart it. It will need to be reconfigured.

`ip route get IPAddress`

displays the cached route for IPAddress

`ip route flush cache IPAddress`

flushes the cached route entry for IPAddress

Tip: The listed commands are helpful to get information on routing and to find mistakes in the routing setup. The `ping` command tests whether **`IPaddr`** can be reached or not, and the `traceroute` command displays the route to an **`IPaddr`**.

```
ping IPaddr
traceroute IPaddr
```

1. Configure the routing table entries of PC1 and PC4. You can either specify a default route or you can insert separate routing entries for each remote network. For this exercise, add a route for each individual remote network. Below we show you how to set up the routing configuration for PC4. Follow similar steps to setup the static routes on PC1.

```
PC4% route add -net 10.0.2.0 netmask 255.255.255.0 gw 10.0.3.1
PC4% route add -net 10.0.1.0 netmask 255.255.255.0 gw 10.0.3.1
```

2. Display the routing table of PC1 and PC4 with `netstat -rn` and screenshot the output.

Interesting things to study (if you are interested!)

- Explain the entries in the routing table and discuss the values of the fields for each entry.

PART 3. More on ROUTER Configuration

If the configuration of PC2 and R1 was done correctly, it is now possible to send IP datagrams between any two machines in the network shown in Figure 1. In most real network configurations, the network configuration requires additional changes before all hosts and routers can send and receive IP datagrams. However, if the network is not configured properly, you will need to debug and test your setup. The table below illustrates several common problems that may arise. Since it is impossible to cover all scenarios, network debugging is a crucial skill that you need to attain for your lab experiments to work well.

Problem	Possible Causes	Debugging
Traffic does not reach destinations on local subnet.	Network interface not configured correctly.	Verify the interface configuration with `show protocols` (in IOS) or `ifconfig` (in Linux).
Traffic reaches router, but is not forwarded to remote subnets.	IP forwarding is not enabled. Routing tables are not configured correctly.	Use `show protocols` to display forwarding status in IOS and `sysctl` in Linux Display routing tables with `show ip route` (in IOS) or `netstat -rn` (in Linux). Run `traceroute` between all hosts and routers.
ICMP request messages reaches destination, but ICMP reply does not reach source.	Routing tables are not correctly configured for the reverse path.	Run `ping` and `traceroute` in both directions.
A change in the routing table has no effect on the flow of traffic.	The ARP cache has old entries.	Flush the ARP table. In Linux, delete entries with `arp -d IPAddress`. In IOS, use the command `clear arp`.

Exercise 3(A). Testing the network setup

1. Test the network configuration by issuing `ping` commands from each host and router to every other host and router. If some `ping` commands do not work, you need to modify the software configuration of routers and hosts. If all `ping` commands are successful, the network configuration is correct, and you can proceed to the next step.

2. Start Wireshark on PC1 link.

3. Execute a `traceroute` command from PC1 to PC4, and save the output.

   ```
   PC1% traceroute 10.0.3.41
   ```

4. Execute a `trace` command from R1 to PC4, and save the output.

   ```
   R1# trace 10.0.3.41
   ```

5. Stop Wireshark and save the captured traffic. Observe how `traceroute` commands gather route information.

6. Save the routing table of PC1, PC4, R1 and R2.

Interesting things to study (if you are interested!)

- Using the Wireshark output and the previously saved routing tables, explain the operation of `traceroute` command.

Exercise 3(C). Order of the routing table lookup

A router or host uses a routing table to determine the next hop of the path of an IP datagram. Generally, routing table entries are sorted in the order of decreasing prefix length, and are read from top to bottom. In this exercise, you determine how an IP router or PC resolves multiple matching entries in a routing table.

1. Add the following routes to the routing table of PC1:

   ```
   PC1% route add –net 10.0.0.0 netmask 255.255.0.0 gw 10.0.1.71
   PC1% route add –host 10.0.3.9 gw 10.0.1.81
   ```

   From Exercise 1(C), there should be a network route for the network prefix `10.0.3.0/24`. If there is no such route, then add the following entry:

   ```
   PC1% route add –net 10.0.3.0 netmask 255.255.255.0 gw 10.0.1.61
   ```

2. Referring to the routing table, determine how many matches exist for the following IP addresses:
   a) `10.0.3.9`
   b) `10.0.3.14`
   c) `10.0.4.1`

3. Start a Wireshark session on PC1, and issue the following `ping` commands from PC1:

```
PC1% ping 10.0.3.9 -c 5
PC1% ping 10.0.3.14 -c 5
PC1% ping 10.0.4.1 -c 5
```

Note that gateways with IP addresses `10.0.1.61, 10.0.1.71,` and `10.0.1.81` do not exist.

4. Save the output of Wireshark and PC1's routing table.

Interesting things to study (if you are interested!)

- Use the saved output to indicate the number of matches for each of the IP addresses above. Based upon what you have seen, explain how PC1 resolves multiple matches in the routing table. Depending on how you set up PC1's routing table, you will get different responses (i.e., if you used default route or explicit entries for 10.0.2.0 and 10.0.3.0).

Exercise 3(D). Default routes

1. Delete the routing table entries added to PC1 in Step 1 of Exercise 3(C) above using the "route del" command. (Otherwise, the entries will interfere with the remaining exercises in this lab.)

2. Add default routes on PC1 an PC4.

   a) On PC1, add a default route with interface FastEthernet0/0 on R1 as the default gateway.

   b) On PC4, add a default route with interface `FastEthernet1/0` of R2 as the default gateway.

3. Start Wireshark to capture traffic on PC1 link.

4. Issue a `ping` command from PC1 to a host on a network that does not exist, e.g.:

```
PC1% ping 10.0.10.110 -c 5
```

5. Save the Wireshark output.

Interesting things to study (if you are interested!)

Use the saved output to answer the following questions.

- What is the output on PC1, when the `ping` command is issued?
- Determine how far the ICMP Echo Request message travels.
- Which, if any, ICMP Echo Reply message returns to PC1?

PART 4. ICMP ROUTE REDIRECT

ICMP route redirect messages are sent from a router to a host, when a datagram should have been forwarded to a different router or interface. In Linux, an ICMP route redirect message updates the *routing cache*, but not the *routing table*.

Both the routing cache and the routing table contain information for forwarding traffic. Before a Linux system performs a routing table lookup, it first inspects the routing cache. If no matching entry is round in the cache, Linux performs a lookup in the routing table. After each routing table lookup, an entry is added to the routing cache. The routing cache does not aggregate table entries, and there is a separate entry for each destination IP address. As a consequence, a lookup in the routing cache does not require a longest prefix match. An entry in the routing cache is deleted if it has not been used for some time, usually after 10 minutes. When an ICMP Redirect message arrives, an entry is added to the routing cache, but no update is performed to the routing table.

Recall the following commands to display the contents of the routing cache in Linux:

```
ip route get IPAddress
```

To clear the route cache in Linux:

```
ip route flush cache IPAddress
```

or

```
ip route flush cache
```

Similarly, for IOS the commands are:

```
show ip cache
clear ip cache
```

In this part of the lab, you will use three Routers. Figure 2 and Table 2 describe the network configuration for the exercises below.

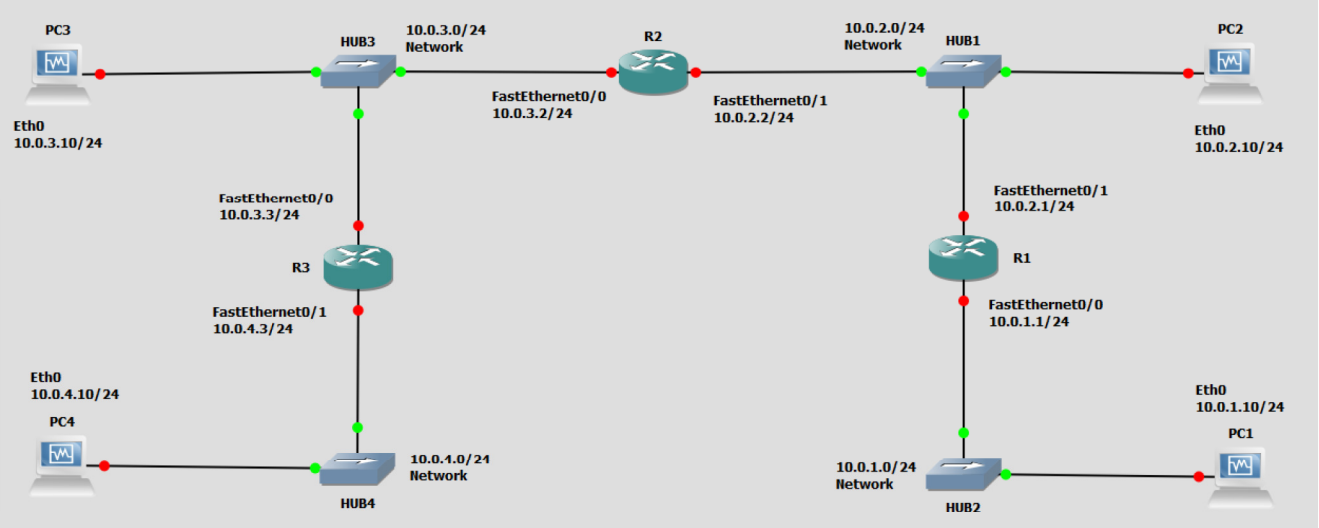


Figure 2 Network topology for Part 5

Cisco Routers	FastEthernet0/0	FastEthernet1/0
R1	10.0.1.1 / 24	10.0.2.1 / 24
R2	10.0.3.2 / 24	10.0.2.2 / 24
R3	10.0.3.3 / 24	10.0.4.3 / 24
PC	**Ethernet Interface eth0**	
PC1	10.0.1.10 / 24	
PC2	10.0.2.10 / 24	
PC3	10.0.3.10 / 24	
PC4	10.0.4.10 / 24	

Table 2 IP addresses for Part 5

Exercise 5. Observing ICMP Redirect

In the network shown in Figure 2, when PC2 sends datagrams with destination `10.0.3.10` (PC3) to `10.0.2.1` (R1), as opposed to `10.0.2.2` (R2), then R1 sends an ICMP route redirect message to PC2. The ICMP route redirect informs PC2 that it should send datagrams with destination `10.0.3.10` to R2 instead.

In this exercise, you will create the above scenario. You will trigger the transmission of an ICMP Route Redirect message and subsequently observe a change to the routing cache.

1. Connect the Ethernet interfaces of the routers and the hosts to the hubs as shown in Figure 2.

2. Delete the routing table entries, route caches and ARP caches on all PCs and on all Routers.

3. Build a new static routing entry on R1 for network 10.0.3.0/24 to R2 (FastEthernet1/0)

4. ICMP redirect messages can be used to attack a network. For this reason, hosts by default ignore ICMP redirect messages. On a Linux system, the `accept_redirects` variable controls whether the host can accept or not a redirect ICMP message.

   1) Use `sysctl` command to verify the current ICMP redirect status on PC2

      ```
      PC2% sysctl net.ipv4.conf.all.accept_redirects
      ```

   2) If the response is "0", then you need to enable it. E.g., enable PC2 to accept ICMP redirect messages.

      ```
      PC2% echo 1 | tee /proc/sys/net/ipv4/conf/*/accept_redirects
      ```

   3) Use the `sysctl` command again to ensure that the parameter change occurred.

5. Set up the routing table of PC2 in such a way that it provokes the transmission of an ICMP route redirect message as discussed above, i.e., make R1 the default router for PC2. In other words, force it to send the packet to a router other than the one you would expect it to use to get to PC3.

6. Save the contents of the routing table and the routing cache on each of R1, R2, and PC2.

7. Set up the routing table of PC3 and PC4 so that they can reach PC2.

8. Start Wireshark on PC2 link to capture ICMP messages and issue a `ping -c 5` from PC2 to PC3. Repeat for a `ping -c 5` from PC2 to PC4.

9. Stop Wireshark, save the capture data and the contents of the routing table and the routing cache of PC2, and routers R1, R2 after the ICMP redirect messages.

10. Wait a few minutes with no transmissions occurring and check the contents of the routing caches again. Save the output.

Interesting things to study (if you are interested!)

- Is there a difference between the contents of the routing table and the routing cache immediately after the ICMP route redirect message?
- When you viewed the cache a few minutes later, what did you observe?
- Describe how the ICMP route redirect works using the outputs you saved. Include only relevant data from your saved output to support your explanations.
- Explain how R1, in the above example, knows that datagrams destined to network `10.0.3.10` should be forwarded to `10.0.2.2`?

PART 4. ICMP Denial of Service (DoS) Attack

In this part of the lab, you will be using the Mallory as the attacker to attack a local area network by actively broadcasting ICMP requests to every single host in the network. This will overload the network and cause a denial of service for several time critical services running within the network.

`hping3` is a command for sending (almost) arbitrary TCP/IP packets to network hosts. This part of the Lab utilizes this command to send spoofed ICMP requests to flood the network. Note that to make it work, the PCs have to be able to accept broadcast ICMP messages. By default they don't. We show you how to enable that feature and observe the DoS attack.

Note: Packet flooding can cause GNS3 to run sluggishly or crash if it keeps running for a while!

`hping3 -1 --flood` **`IPAddress`**

Send ICMP packets flooding the target IP address, -1 flag here indicates that packets are sent using the ICMP protocol. The *flood* option here tells the command to send packets as fast as possible, ignoring incoming replies.

`hping3 -1 --flood --rand-source` **`IPAddress`**

This command does the same thing as the previous command, except sending the packets from a random source address

`hping3 -1 --flood -a` **`TargetIPAddress DestinationIPAddress`**

This command will send packets from **TargetIPAddress** to the **DestinationIPAddress** using the ICMP protocol, as well as sending as fast as possible.

For more info about the hping3, look up it's MAN page in the Linux manual:

https://linux.die.net/man/8/hping3

Exercise 7. Using the hping3 for Denial of Service Attack

1. For this exercise we will use the network configuration shown below in Figure 4, and Table 4. You will login to Mallory, and fire up the terminal. Remember to turn "Wired Connection" off. Now Mallory is ready to be configured. Always 'su root' for VMs.

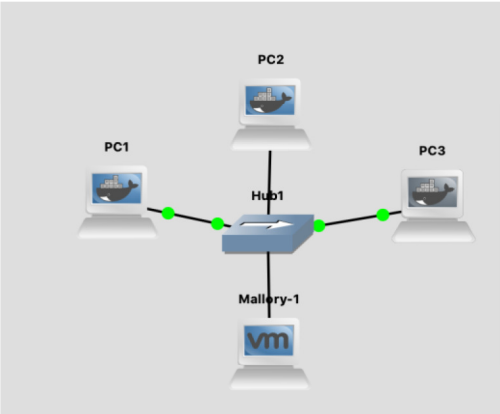


Figure 4

2. Assign IP addresses to the PCs and Mallory-1as shown in Table 4:

PCs	IP Address of eth0
PC1	10.0.1.11 / 24
PC2	10.0.1.12 / 24
PC3	10.0.1.13 / 24
Mallory	10.0.1.44 / 24

Table 4

3. Start GNS3. Wait a few minutes for VM Mallory to start.
4. Start Wireshark on the link between PC1 and the Hub.
5. Issue a `ping` command from PC2 to PC1, DO NOT terminate THE PING command.

```
PC2% ping 10.0.1.11
```

6. Now issue the following command on Mallory:

```
Mallory% hping3 -1 --flood 10.0.1.11
```

7. Do not stop the `hping3` command on Mallory. Now stop the `ping` command on PC2 (^C). Screenshot PC2's console window showing `ping` output.

8. Stop the `hping3` command on Mallory (^C).

9. Redo steps 4 to 8, with Mallory running the following command:

```
Mallory% hping3 -1 --flood --rand-source 10.0.1.11
```

10. Examine the packets captured in Wireshark.

11. By default the PCs ignore broadcast ICMP requests. To disable "ignore ICMP broadcast requests" execute the following command on PC1, PC2 and PC3

```
echo 0 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
```

12. Use `sysctl` command to verify the current ICMP redirect status on PC1, PC2, and PC3. It should be 0.

```
sysctl net.ipv4.icmp_echo_ignore_broadcasts
```

13. Now execute the following command on Mallory to spoof your IP address as PC1's IP address and send ICMP request to the broadcast address of the subnet.

```
Mallory% hping3 -1 --flood -a 10.0.1.11 10.0.1.255
```

14. Stop `hping3` on Mallory (^C).

15. Stop Wireshark capture. Save the output.
16. Stop GNS3, quit VirtualBox and quit GNS3.

Interesting things to study (if you are interested!)

- What difference did you notice when looking at the screen shots of the ping command output on PC2? Is there a huge difference between the time taken for each ping request? How would this impact the service provided by PC1?
- What difference did you notice in the output when doing step 9 compared to the packets captured from the first run of step 4 – 8?
- Use the packets captured in Wireshark in step 13 to explain why all the PCs on the network sent ICMP replies back to PC1 even though it was not sending any packets.

APPENDIX A. A TIP TO SPEED UP the SETUP of EXPERIMENTS

To configure the cisco routers and PCs, you are required to enter the commands manually in the console window. By now you will probably have realized that many of the commands are repeated. Also, you probably noticed that GNS3 resets PC and Cisco router configurations when you stop a GNS3 project. Saving the commands in a file and then using copy and paste of the command set will save you a lot of time.

Below we show you how you can save configuration commands in a Microsoft Excel Spread Sheet, for use at a later time.

1. Open an Excel spread sheet. Then, add a command in each row as shown in Figure A.1. The columns represent the different devices you are saving commands for.

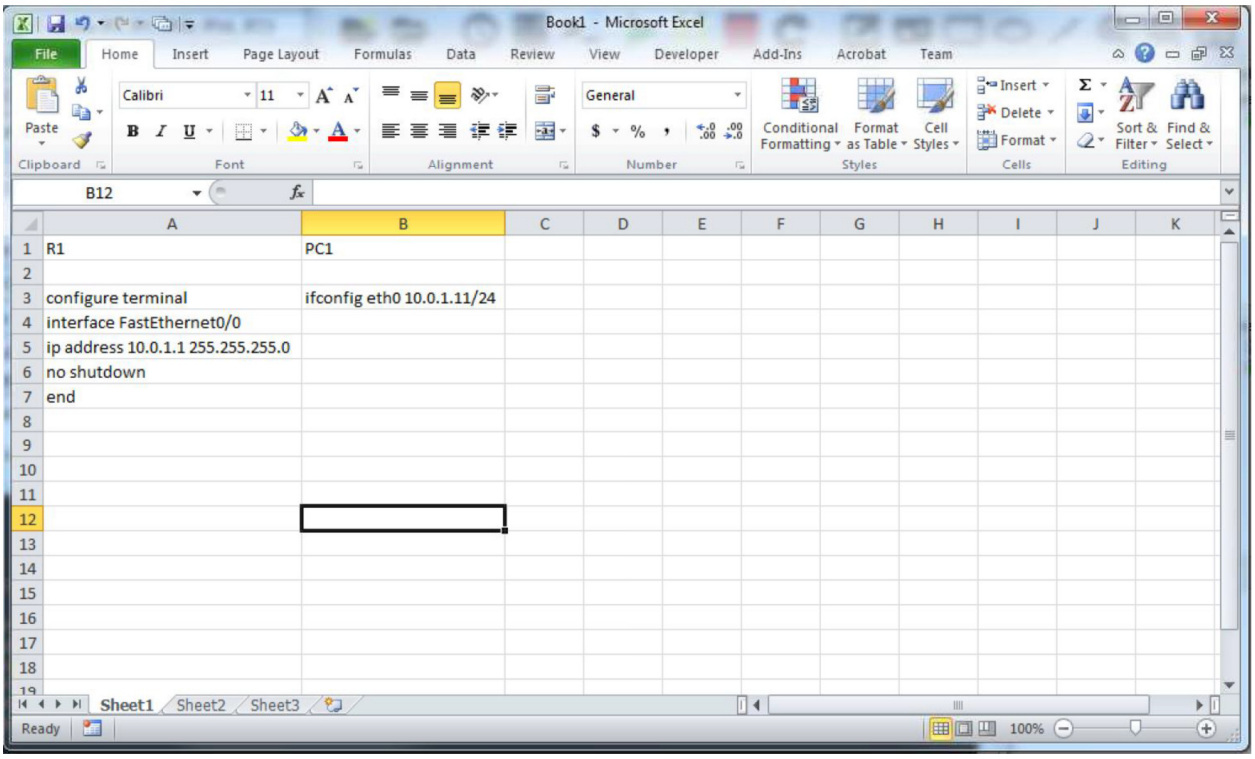


Figure A.1

2. Select a command set from a column and copy the block of commands by pressing the shortcut key "Ctrl + C" for Windows and "Command + C" for Mac. The command set will be highlighted as shown in Figure A.2.

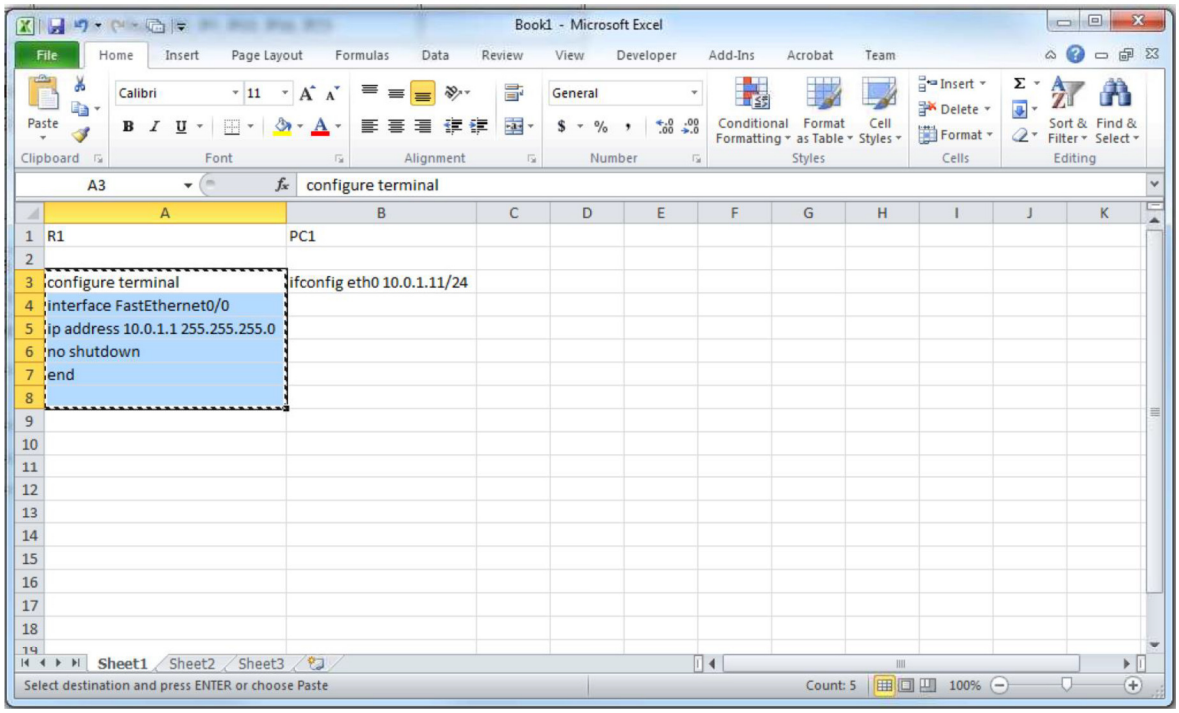


Figure A.2

3. Start a Cisco router R1 and a PC1 in GNS3.

4. Open console windows for each of R1 and PC1.

5. Now you paste the commands by right click on the console windows for Windows users. For Mac users, paste the commands with the shortcut key "Command + V". You will see that the commands are executing as shown in Figure A.3 for R1.

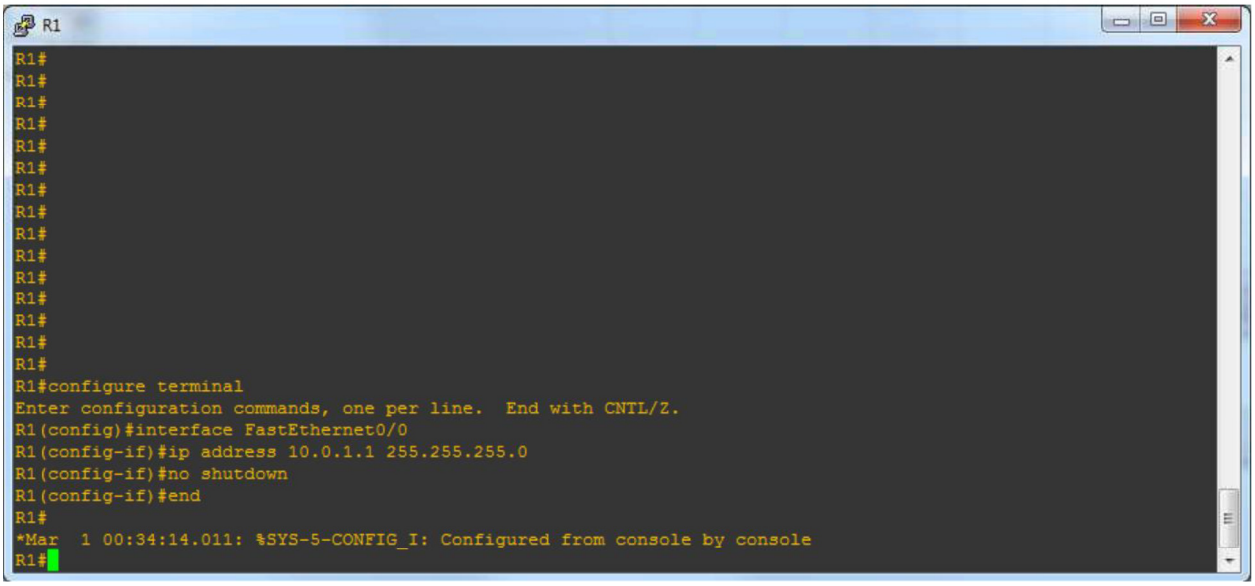


Figure A.3

6. Repeat the above step to configure PC1.

7. Please make sure that the commands do not contain any typos.

8. Save the excel spread sheet. You can use for the next experiment if the configuration is similar. If not you can edit the saved file, e.g., IP address may have changed. And repeat the copy-paste routing to configure your devices.

APPENDIX B. Saving a Router Configuration in GNS3 Project

For routers, you can use the command ***# copy running-config startup-config*** to save the configurations, and then save your GNS3 project. This will ensure your IP address configurations don't get wiped off when you restart GNS3 every time.