LAB SEVEN – NAT & DHCP

NAT (Network Address Translation) refers to a function that replaces the IP addresses (and possibly the port numbers) of IP datagrams. NAT is run on routers that connect private networks to the public Internet, to replace the IP address-port pair of an IP packet with another IP address-port pair. Generally, the operations of NAT are specified in terms of a set of rules which determines how IP addresses are to be replaced.

Often, a NAT device is referred to as a *NAT box.* One of the reasons for using NAT is that it conserves IP addresses. NAT allows hosts in a private network to share public IP addresses, or to limit the use of public IP addresses to a small number of hosts in the private network.

Private networks may have IP addresses that are non-Internet routable, as specified in RFC 1918. This means that the Internet Rs do not have entries in their routing tables for these addresses.

The Dynamic Host Configuration Protocol (DHCP) can be used to dynamically set and change configuration parameters of Internet hosts, including IP address, netmask, default router, and DNS server. DHCP is based on a client-server model. DHCP clients send requests to a DHCP server via UDP to port 67, and the server responds with an allocation of IP addresses and other configuration parameters to port 68 on the client.

# PART 1. Network Address Translation (NAT)

Figure 7.1 shows one private network which is connected via a router to a public network. The router acts as the NAT device.

In the network in Figure 7.1, R1 is set up as a NAT router. With NAT, the hosts in the private network 10.0.1.0/24 can access the public network, given here as 128.195.7.0/24.

In the private network, Alice, Bob and PC1 are used as hosts and R2 is a router on the public network that we will configure as a Telnet server for the hosts on the private network.



Figure 7.1 Configuration for Part 1

* On R1, you will use Cisco IOS commands to configure the NAT rules.
* R2 will be configured for interface FastEthernet0/0 on the public network. It will have no forwarding function.
* The assignment of IP addresses and default gateways for all hosts and routers are shown in Table 7.1 and Table 7.2.

|  |  |  |
| --- | --- | --- |
| **Hosts** | **IP address** | **Default Gateway** |
| Alice | 10.0.1.2/24 | 10.0.1.1 |
| Bob | 10.0.1.3/24 | 10.0.1.1 |
| PC1 | 10.0.1.4/24 | 10.0.1.1 |

Table 7.1 IP addresses and gateway assignments of all hosts for Part 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Cisco Rs** | **FastEthernet0/0** | **FastEthernet1/0** | **Default Gateway** |
| R1 | 10.0.1.1/24 | 128.195.7.1/24 | None |
| R2 | 128.195.7.2/24 | None | None |

Table 7.2 IP addresses and gateway assignments of routers for Part 1

**Exercise 1(A). Network setup**

Configure the network in Figure 7.1 with the IP address configuration shown in Tables 7.1 and 7.2. The following commands review the steps involved in the configuration.

1. Just as for all the other labs, you need to be “root” on the VMs to configure their interfaces. Use the “su root” command.
2. Use ifconfig to configure the IP addresses on Alice, Bob and PC1. Recall that the interface for Ubuntu hosts is “ens33”. Add a default gateway on each host as given in Table 7.1. Here shown how for Alice:

**Alice%** route add default gw 10.0.1.1

1. Configure the IP addresses of interfaces FastEthernet0/0 and FastEthernet0/1 on R1. Following is the sample configuration for R1

**R1#** enable

**R1#** configure terminal

**R1(config)#** no ip routing

**R1(config)#** ip routing

**R1(config)#** interface FastEthernet0/0

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

**R1(config-if)#** ip address 10.0.1.1 255.255.255.0

**R1(config-if)#** interface FastEthernet1/0

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

**R1(config-if)#** ip address 128.195.7.1 255.255.255.0

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

1. Repeat for R2 but only configure interface FastEthernet0/0. After the configuration, you should be able to issue successful ping commands between Alice, Bob and PC1 in the private network and between the two routers in the public network.

### Exercise 1(B). Configuration of NAT on a Cisco Router

In this exercise we look at two scenarios: 1) **Static** NAT, where a public address owned by the private network, is assigned (mapped) to one of the internal hosts so that outside devices can reach that host (e.g., a Web server), and 2) **Dynamic** NAT, where multiple private IP addresses are mapped to a single public IP address (that of the NAT router). The latter use of NAT is called *IP masquerading, port and network address translation (PNAT) or Network Address and Port Translation (NAPT)*. IP Masquerading, besides mapping the private IP source address of a host to that of the public IP address of the router, also modifies the port number of packets so that the single public IP address can be “overloaded”, i.e., used by multiple private addresses (the port numbers allow the multiplexing of many private connections to one single public IP address, the port numbers are used to demultiplexing the traffic streams back to the original private IP source addresses).

In this exercise, R1 will be configured to perform the IP masquerading. Network 10.0.1.0 is the private network, henceforth referred to as the *inside* network. The following steps illustrate how to enable “NAT overload” configuration. With NAT overloading, multiple internal host addresses can use the single public IP address that is assigned to the NAT router. This is achieved by translating source IP addresses and UDP/TCP ports in the packets to the NAT router IP address and a randomly assigned port number. The “translations/mappings” are maintained in a NAT table in the router.

A Cisco router can be set up to run as a NAT router. In Cisco IOS, the private network is referred to as *inside*, and the public network is referred to as *outside*. An IP address that is seen by hosts on the inside is called a *local address*, and an IP address that is seen by hosts on the outside is called a *global address*. There are four different types of addresses:

|  |
| --- |
| 1. An *inside local address* is an address in the private network that is not visible in the public network.
2. An *inside global address* can be used in the public network for devices in the private network.
3. An *outside global address* is an address in the public network that is not made known in the private network.
4. An *outside local address* is used by devices in the private network to addresses in the public network.
 |

Using this terminology, a NAT router translates inside local addresses (source addresses) to outside global addresses for outgoing packets and outside global addresses (destination addresses) to inside local addresses for incoming packages.

In the box below, we show the commands that you will use for setting up NAT functionality on a Cisco router.

|  |
| --- |
| **IOS MODE: PRIVILEGED EXEC**Displays the content of the NAT table.show ip nat translations**IOS MODE: INTERFACE CONFIGURATION**Specifies that an interface is connected to the private network.ip nat insideSpecifies that an interface is connected to the public network.ip nat outside**IOS MODE: GLOBAL CONFIGURATION**Adds a rule so that the private IP address IPaddr1 is mapped to a public IP address IPaddr2.ip nat inside source static **IPaddr1 IPaddr2****Example:** The command bellow maps the private address 10.0.1.2 to the public address 128.195.7.3.ip nat inside source static **10.0.1.2 128.195.7.3**Adds a rule to define an access listip access-list standard ***access-list-name***permit ***network-id*** ***wild-card***deny any**Example:** The command bellow defines the access list, i.e., those that can use NAT. This permits any internal addresses in the subnet 10.0.1.0 and denies any other addresses not in 10.0.1.0/24.ip access-list standard **nat1**permit **10.0.1.0 0.0.0.255**deny anyThis maps the access list to the router’s interface. “overload” below specifies the use of PNAT.ip nat inside source list ***access-list-name*** interface ***Interface*** overload**Example:** The command bellow maps the access list to the router’s interface FastEthernet 0/0ip nat inside source list **nat1** interface **FastEthernet0/0** overload |

1. **Set Up the NAT table of R1:** Use the following commands to set up R1 as a NAT router. A static NAT rule is added so that the private IP address of Alice, 10.0.1.2, is translated to the public address 128.195.7.3. Table 7.3 shows the static NAT mapping for Alice.

|  |  |  |
| --- | --- | --- |
| **VM** | **Inside Local Address** | **Outside Global Address** |
| Alice | 10.0.1.2/24 | 128.195.7.3/24 |

Table 7.3 Private and public address of Alice

The IOS commands for R1 are as follows:

**R1#** enable

**R1#** show ip nat translations

**R1#** configure terminal

**R1(config)#** interface FastEthernet0/0

**R1(config-if)#** ip nat inside

**R1(config-if)#** interface FastEthernet1/0

**R1(config-if)#** ip nat outside

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

**R1(config)#** ip nat inside source static 10.0.1.2 128.195.7.3

**R1(config)#** ip access-list standard nat1

**R1(config-std-nacl)#** permit 10.0.1.0 0.0.0.255

**R1(config-std-nacl)#** deny any

**R1(config-std-nacl)#** exit

**R1(config)#** ip nat inside source list nat1 interface f1/0 overload

**R1(config)#** end

**R1#** show ip nat translations

After configuring R1 as a NAT router, display the content of the NAT table and save a screenshot to a file.

1. **Observe traffic at a NAT Router:** To observe the IP address translation, issue ping commands between machines in the public and private network.
2. Start a Wireshark session on the link R1- Hub2 to capture the traffic on the private network.
3. Start a Wireshark session on the link R1 - Hub1 to capture the traffic on the public network.
4. Issue the following ping commands on Alice, Bob, PC1 and R2, respectively

**Alice%** ping 10.0.1.3 –c 5
**Alice%** ping 128.195.7.2 –c 5

**Bob%** ping 10.0.1.2 -c 5
**Bob%** ping 128.195.7.2 -c 5

**PC1%** ping 10.0.1.2 -c 5
**PC1%** ping 128.195.7.2 -c 5

**R2#** ping 10.0.1.2
**R2#** ping 128.195.7.3

1. Screenshot the ping outputs on each console and save to answer questions below.
2. Stop the Wireshark capture and save the data.
3. We will now set R2 up as a Telnet server to which the hosts can login.
4. First, we have to create a default username “**user**” and password “**password**” on **R2.**

**R2#** config terminal

**R2(config)#** username **user** secret **password**

1. Now we enable a Telnet Server on **R2**.

**R2#** configure terminal

**R2(config-if)#** line vty 0 15

**R2(config-line)#** login local

**R2(config-line)#** end

1. Start a Wireshark capture session on R1 - Hub1 link and on R1 – Hub 2 link as before. Use the following commands to initiate a Telnet connection from Alice and from Bob to R2 Telnet server:

**Alice%** telnet 128.195.7.2

**Bob%** telnet 128.195.7.2

1. Type some data on the command line of each Telnet session so that you can obverse the telnet TCP data packets on the Wireshark data capture.
2. Stop both Wireshark captures and save.
3. Stop GNS3 and Quit.

**Lab Questions:**

* For each of the preceding Telnet and ping commands above, which succeeded and which failed. Provide an explanation why the commands succeeded or failed using the saved Wireshark data.
* For each successful Telnet session, trace 1 packet on both the private and public network. Expand the datagram and include the IP header of the packet on the private network and on the public network.
* For each successful ping command, include the IP header data of an outgoing ICMP Request message (on the private network) and its corresponding incoming ICMP reply message (on the public network).
* How does router R1 know that a packet coming from the public network is destined to a host in the private network?

# PART 2. Dynamic Host Configuration Protocol (DHCP)

In this part of the lab you will observe how Dynamic Host Configuration Protocol (DHCP) is initialized and used for the distribution of IP addresses.

You will also learn about DHCP relay agents. When the DHCP client and DHCP server are not on the same IP network, DHCP relay agents can act as routers of DHCP messages. A DHCP relay agent can forward DHCP requests from a DHCP client to a DHCP server, and it can forward the reply messages from the DHCP server to the DHCP client.

The network configuration for Part 2 is shown in Figure 7.2. PC1, Alice, and Bob are set up as DHCP clients and initially do not have IP addresses. R1 is configured as a DHCP server, which listens for DHCP requests on all of its interfaces and transmits network configuration parameters to clients requesting an IP address.

Table 7.6 lists the range of addresses that are associated at the DHCP server R1 with each network.



Figure 7.2 Network topology for Part 2

|  |  |  |
| --- | --- | --- |
| **Hosts** | **IP address** | **Default Gateway** |
| PC1 | 10.0.2.2/24 | 10.0.2.1 |
| Alice | none | none |
| Bob | none | none |

Table 7.4 Configuration of the hosts in Part 2

### Exercise 2(A). Hosts and router R1 configuration

1. Start GNS3 and open a new project. Do not continue from Part1.
2. Set up the network topology as shown in Figure 7.2. Configure the IP addresses of the PCs and routers as shown in the Table 7.4 and 7.5.

|  |  |  |  |
| --- | --- | --- | --- |
| **Cisco R** | **FastEthernet0/0** | **FastEthernet1/0** | **Default Gateway** |
| R1 | 10.0.1.1/24 | 10.0.2.1/24 | None |

Table 7.5 Router R1 configuration parameters for Part 2

1. **NOTE**: Alice and Bob **should not** have a default route. Check each of their routing tables respectively. On Alice, this is done with the command:

**Alice%** netstat -rn

1. **NOTE**: Alice and Bob **should not** have an assigned IP address. Use the ifconfig -a command to check that there is no IP address associated with ens33 on each host. If there is, assign the IP address 0.0.0.0 on ens33 using the following command shown here for Alice:

**Alice%** ifconfig ens33 0.0.0.0 up

### Exercise 2(B). Configuring and starting a DHCP server on a Cisco Router

On a Cisco router, a DHCP server is started with the command ip dhcp pool. The DHCP server configuration steps follow after the command. The configuration contains information on available IP addresses, as well as other configuration information.

|  |
| --- |
| **IOS MODE: INTERFACE CONFIGURATION**ip dhcp pool **N**Configures dhcp address pool N, where **N** is a number, e.g., 1, or 5,…**IOS MODE: DHCP CONFIGURATION** network **subnet** **subnetmask**Configures **subnet** with **subnetmask** as the subnetwork number for the dhcp address pool “ip dhcp pool N”default-router **IPaddr**Sets the default router for DHCP clients to **IPaddr**lease **a b c**Configures the lease duration for an IP address with the options **a** days (optional) **b** hours (optional) **c** minutes |

1. The following is an example configuration for a DCHP server for R1:

**R1>** enable

**R1#** configure terminal

**R1(config)#** ip dhcp pool 1

**R1(dhcp-config)#** network 10.0.1.0 255.255.255.0

**R1(dhcp-config)#** default-router 10.0.1.1

**R1(dhcp-config)#** lease 0 0 10

**R1(dhcp-config)#** end

A DHCP client is assigned an IP address for a period of time that is known as a lease. The preceding configuration assigns IP addresses for a lease time of 10 minutes (0day 0hour and 10minutes). For DHCP requests on network 10.0.1.0/24, the DHCP server assigns IP addresses in the range 10.0.1.1 to 10.0.1.254. Note that it assigns IP address 10.0.1.1 as the default gateway. See Table 7.6 below.

From the above we see thatAlice and Bob will be assigned IP addresses from the 10.0.1.1 – 10.0.1.254 pool. They will also be assigned with default gateway 10.0.1.1.

|  |  |  |
| --- | --- | --- |
| **Subnet** | **Range of Addresses** | **Default Router** |
| 10.0.1.0/24 | 10.0.1.1 to 10.0.1.254 | 10.0.1.1 |

Table 7.6 DHCP server configuration – Address Pool and Default router

1. Start a Wireshark session on link interfaces FastEthernet0/0 and FastEthernet1/0 of R1.
2. **Start a DHCP client**: To start the dhcp client on Alice and Bob on interface ens33, run the following command:

**Alice%** dhclient ens33

**Bob%** dhclient ens33

1. Issue pings (-c 5) from Alice and Bob to PC1 and from PC1 to Alice and Bob. Are the pings successful? Screenshot your output.
2. **Renewing leases of IP addresses:** The DHCP client is assigned an IP address for a limited period of time, which is called a *lease*. You can observe the maximum time of a lease assigned to a client by a server, by using the following command:

**Alice%** cat /var/lib/dhcp/dhclient.leases

1. If you set the lease time to say “2mins” instead of 10 as given in example above, and you ping for a while on the console of Alice to Bob, you should see the lease extension negotiation process.
2. Stop the Wireshark capture and save the data.
3. Stop GNS3.

**Lab Questions**:

Answer the following questions using the captured traffic files:

* Which IP address is assigned to Alice? and to Bob?
* Were all the pings successful. If not which ones? And why?
* Observe the source and destination IP addresses of the packets that are sent between DHCP client and DHCP server.
* How can a host send and receive DHCP packets even though it does not have an IP address?
* Do you observe any ARP packets? If so, explain the function of ARP in this context.
* Observe and interpret the output of the DHCP packets. You should see the following packet types: DHCP Discover, DHCP Offer, DHCP Request, DHCP ACK.
* Identify and interpret the option fields (default router, DNS server, subnet, …) in the DHCP packet types that you observe.
* Observe how a DHCP client renews a lease and saves the captured traffic to a file.
* What type of DHCP messages are exchanged?
* How long does a DHCP client wait until it attempts to renew its lease?

**Exercise 2(C). DHCP relay agent**

A DHCP relay agent can forward DHCP packets when the DHCP server and the DHCP client are not on the same network. Note that the role of a DHCP relay agent is not entirely trivial, since it acts as a router for a host that does not have an IP address. Here you explore how packets from the client reach the server on another network and how the response from the server reaches the DHCP client.

The network configuration for this part is given in Figure 7.3 below. It is the same as the previous topology given in Figure 7.2, but with the addition of R2 as the relay agent and moving host Bob to R2’s network.



Figure 7.3. Adding a DHCP relay router R2 to the configuration setup in Figure 7.2.

|  |  |  |
| --- | --- | --- |
| **Hosts** | **IP address** | **Default Gateway** |
| PC1 | 10.0.2.2/24 | 10.0.2.1 |
| Alice | none | none |
| Bob | none | none |

Table 7.7 Host IP Address assignments for network topology shown in Figure 7.3

|  |  |  |  |
| --- | --- | --- | --- |
| **Cisco R** | **FastEthernet0/0** | **FastEthernet1/0** | **Default Gateway** |
| R1 | 10.0.1.1/24 | 10.0.2.1/24 | 10.0.1.4 |
| R2 | 10.0.3.1/24 | 10.0.1.4/24 | 10.0.1.1 |

Table 7.8 Router IP Address assignments for network topology shown in Figure 7.3

1. The following commands are used to set up R2 as a DHCP relay agent and also configure R 1 to act as a DHCP server for network 10.0.3.0 in addition to network 10.0.1.0. Note that we have to assign a default router for R1, interface f1/0 on R2. The command to configure R1 and R2 are as follows:

**R1>** enable

**R1#** configure terminal

**R1(config)#** ip route 0.0.0.0 0.0.0.0 10.0.1.4

**R1(config)#** ip dhcp pool 1

**R1(dhcp-config)#** network 10.0.1.0 255.255.255.0

**R1(dhcp-config)#** default-router 10.0.1.1

**R1(dhcp-config)#** lease 0 0 2

**R1(dhcp-config)#** end

**R1#** configure terminal

**R1(config)#** ip dhcp pool 2

**R1(dhcp-config)#** network 10.0.3.0 255.255.255.0

**R1(dhcp-config)#** default-router 10.0.3.1

**R1(dhcp-config)#** lease 0 0 2

**R1(dhcp-config)#** end

**R2>** enable

**R2#** configure terminal

**R2(config)#** ip forward-protocol udp

**R2(config)#** interface FastEthernet0/0

**R2(config-if)#** ip address 10.0.3.1 255.255.255.0

**R2(config-if)#** ip helper-address 10.0.1.1

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

**R2(config-if)#** interface FastEthernet1/0

**R2(config-if)#** ip address 10.0.1.4 255.255.255.0

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

**R2(config-if)#** end

1. Start Wireshark capture on R1 f0/0 and R2 f0/0.
2. Set up PC1 with the address and default gateway as given in Table 7.7.
3. Start DHCP clients on Bob and Alice:

**Alice%** dhclient ens33

**Bob%** dhclient ens33

1. Verify that an IP address has been assigned to Alice and Bob using the ifconfig command.
2. Issue pings (-c 5) from Alice and Bob to PC1 and from PC1 to Alice and Bob. Are the pings successful? Screenshot your output.
3. Stop the Wireshark data capture and save the files.
4. Display the NAT table of R1. Screenshot and save.

**R1#** show ip nat translations

1. Stop GNS3 and Quit.

**Lab Questions:**

* Does the DHCP relay agent modify DHCP packets or the IP header? If so, what are the modifications?
* How does the relay agent redirect the replies from the DHCP server? Does it broadcast them or unicast them to the DHCP client?
* Is there a difference in the response of the DHCP server when compared to the DHCP configuration of Alice? If so, explain the difference.
* How does the DHCP server know on which network Bob is located when it receives the DHCP request?
* What is the destination IP address of the first DHCP packet that the DHCP server sends to Bob?

# PART 3. Combining NAT and DHCP and DHCP relaying

# Figure 7.4 shows a network configuration which is very typical of many SOHO (small office, home office) networks.

# The SOHO network is a private network with multiple hosts (Alice and Bob) and two IP routers (R1 and R2).

# The IP router of the SOHO network, R1, provides access to the public Internet by connecting to a router of an ISP. The SOHO router is assigned a single IP address on the “public” interface of the SOHO network via a static assignment from the ISP provider on the public network.

# The SOHO router R1 works as a DHCP server and NAT server for the hosts in the SOHO network. R2 works as a relay agent in the SOHO network.



Figure 7.4. Network Configuration for Part 3

|  |  |  |
| --- | --- | --- |
| **Hosts** | **IP address** | **Default Gateway** |
| PC1 | 128.195.7.33/24 | none |
| Alice | none | none |
| Bob | none | none |

Table 7.9 IP addresses of hosts for network shown in Figure 7.4

|  |  |  |  |
| --- | --- | --- | --- |
| **Cisco R** | **FastEthernet0/0** | **FastEthernet1/0** | **Default Gateway** |
| R1 | 10.0.1.1/24 | 128.195.7.2/24 |  |
| R2 | 10.0.3.1/24 | 10.0.1.4/24 | 10.0.1.1 |
| R3 | 128.195.7.1/24 | none | none |

Table 7.10 IP addresses of routers for network shown in Figure 7.4

**Exercise 3(A).**

The network configuration is as shown in Figure 7.4. It is similar to network configuration of Figure 7.3 except that we have changed R1 f1/0 address to a public IP address as shown in table 7.10 and added R3 to serve as a Telnet server. Start GNS3.

1. Configure the hosts and routers as shown in Table 7.9 and 7.10. Make sure that you add a route on R1 for network 10.0.3.0 via R2.

**Router1(config)#** ip route 10.0.3.0 255.255.255.0 10.0.1.4

1. Set up the Telnet server on R3 using the commands:
	* First, we have to create a default username “**user**” and password “**password**” on **R1**.

**R3#** configure terminal

**R3(config)#** username **user** secret **password**

* + Now we enable a Telnet Server on **R3**.

**R3#** configure terminal

**R3(config)#** line vty 0 15

**R3(config-line)#** login local

**R3(config-line)#** end

1. Configure R1 as a NAT router:

**Router1#** configure terminal

**Router1(config)#** no ip routing

**Router1(config)#** ip routing

**Router1(config)#** interface FastEthernet0/0

**Router1(config-if)#** ip nat inside

**Router1(config-if)#** no shutdown

**Router1(config-if)#** ip address 10.0.1.1 255.255.255.0

**Router1(config-if)#** interface FastEthernet1/0

**Router1(config-if)#** ip nat outside

**Router1(config-if)#** no shutdown

**Router1(config-if)#** ip address 128.195.7.2 255.255.255.0

**Router1(config-if)#** exit

**Router1(config)#** ip nat inside source list nat1 interface FastEthernet1/0 overload

**Router1(config)#** ip access-list standard nat1

**Router1(config-std-nacl)#** permit 10.0.0.0 0.0.255.255

**Router1(config-std-nacl)#** deny any

**Router1(config-std-nacl)#** exit

1. Configure and start a DHCP server on R1 as shown:

**R1>** enable

**R1#** configure terminal

**R1(config)#** ip dhcp pool 1

**R1(dhcp-config)#** network 10.0.1.0 255.255.255.0

**R1(dhcp-config)#** default-router 10.0.1.1

**R1(dhcp-config)#** lease 0 0 10

**R1(dhcp-config)#** end

**R1#** configure terminal

**R1(config)#** ip dhcp pool 2

**R1(dhcp-config)#** network 10.0.3.0 255.255.255.0

**R1(dhcp-config)#** default-router 10.0.3.1

**R1(dhcp-config)#** lease 0 0 10

**R1(dhcp-config)#** end

1. Configure R2 asa DHCP relay agent as shown:

**R2>** enable

**R2#** configure terminal

**R2(config)#** ip forward-protocol udp

**R2(config)#** interface FastEthernet0/0

**R2(config-if)#** ip address 10.0.3.1 255.255.255.0

**R2(config-if)#** ip helper-address 10.0.1.1

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

**R2(config-if)#** interface FastEthernet1/0

**R2(config-if)#** ip address 10.0.1.4 255.255.255.0

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

**R2(config-if)#** end

1. Start Wireshark data capture on R1 f1/0 and R1 f0/0.
2. Alice and Bobare hosts in the SOHO network. PC1 is a host in the public network.
3. Set up Alice and Bob as DHCP clients on interfaces ens33.

**Alice%** dhclient ens33

**Bob%** dhclient ens33

1. Verify that an IP address has been assigned to Alice and Bob using the ifconfig command. Display the routing tables of Alice and Bob with netstat –rn. Take screen shots of both outputs on each host. Save.
2. Issue pings (-c 5) from Alice and Bob to PC1 and from PC1 to Alice and Bob. Are the pings successful? Screenshot your output.
3. Now start a telnet session Alice and Bob to R3 using the following command

**Alice%** telnet 128.195.7.1

**Bob%** telnet 128.195.7.1

Type a few lines on each console for each session. Then exit telnet.

1. Display the NAT table of R1. Screenshot and save.

**R1#** show ip nat translations

1. Stop Wireshark data capture and save the output.

**Lab Questions:**

* + What are the IP addresses assigned to Alice and Bob?
* Are all the pings successful? If not, why?
* How are the IP addresses mapped onto the public IP address of the NAT server R1 for the pings? And for the Telnet session?
* Show the NAT table of R1. What do you see?