1 *The Router Functionality*

Figure 1 shows our topology with number 8868.

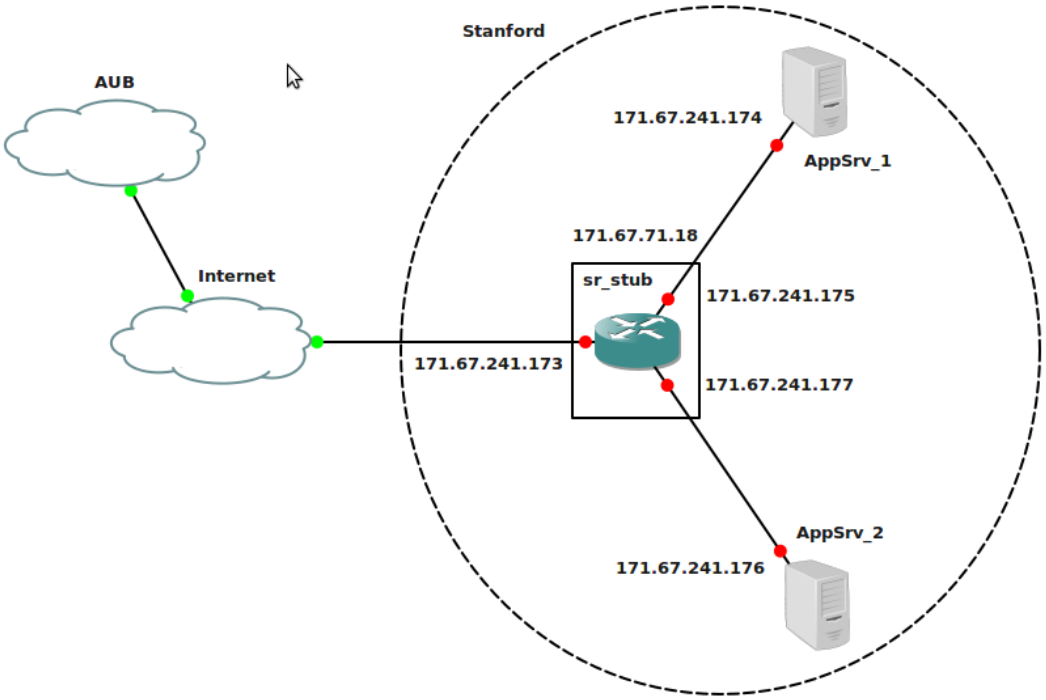

Figure 1: Topology # 8868

All the required specifications for the IP Router project are fully implemented and tested. The following sub-sections explain how each of these specifications is handled and implemented in the code:

1.1 Handling ARP Packets

The implemented router can successfully handle ARP REQUEST and REPLY packets. When an ARP REQUEST packet is received by the router, it automatically generates its corresponding ARP REPLY and sends it back to the sender.

However, receiving an ARP REPLY packet means that it is a reply of an ARP request packet sent by the router itself. In this case the router takes the required information, the destination MAC address, from this reply packet, pops a packet from the queue, updates its destination MAC address and sends it out on the appropriate interface.

1.2 Handling ICMP ECHO REQUEST Packets

The router successfully handles any ICMP ECHO REQUEST packet. When such packet is received, the router generates an ICMP ECHO REPLY packet, setting the destination MAC address to FF:FF:FF:FF:FF:FF, and pushes it in the queue of the output interface. After that, the router generates an ARP REQUEST packet and sends it to the next hop where the ICMP ECHO REPLY packet must be forwarded. When the ARP REPLY is received, the router pops the ICMP ECHO REPLY from the queue, replaces the destination MAC address from FF:FF:FF:FF:FF:FF to the MAC received in the ARP REPLY packet, and sends the ICMP ECHO REPLY packet via the appropriate interface.

1.3 Handling TCP/UDP Packets

The router handles any TCP/UDP packet that arrives at any of its interfaces and generates an ICMP PORT UNRECHABLE packet and sends it back to the TCP/UDP requester.

In order to test this functionality, a *telnet* or *ssh* session to one of the router's interfaces can be requested. Thus, the router returns ICMP PORT UNRECHABLE.

1.4 Forwarding Packets between the Firewall and the Application Servers

When a packet is received by the router, it first checks the *layer 2* and *layer 3* addresses to make sure that the packet is destined to one of the router's interfaces. If the destination MAC address of the packet is one of the router interfaces' MAC addresses and the destination IP address of the packet is not destined to one of the interfaces, the router checks its routing table to find the best path to the destination. If no route is found the packet will be sent to the default route.

In both cases, the router queues the packet that must be forwarded, and sends an ARP packet requesting the MAC address of the next hop where the packet must be forwarded. When an ARP REPLY is received from this next hop, the router pops the packet from the queue, modifies its destination MAC address with the one received in the ARP reply, and sends the packet out through the appropriate interface.

1.5 Handling Traceroute Packets

When the router receives a packet with a TTL ≤ 1, it first checks if the packet is dentined to one of its interfaces. If this is the case, the router generates an ICMP HOST UNREACHABLE packet and sends it back to the source. Otherwise, it generates an ICMP TIME EXCEEDED packet with the source IP address being the address of the interface over which it received the packet and the destination IP address being that of the sender of this packet.

1.6 ARP Cache Implementation

The ARP cache is implemented by employing a linked list data structure where the insertion of elements at the beginning of the list occurs with a cost $\mathcal{O}(1)$. Each element in the linked list contains:

- *IP*
- *MAC*
- *Timestamp*: this is time of creation of the respective cache entry represented in the number of seconds from 1/1/1970 00:00:00.

Each cache entry expires after 15 secs and thus should be purged of the linked list data structure. This is achieved by a separate execution thread which compares the current time with the timestamp field in each cache entry. If the time difference is greater than or equal to 15 secs, the entry is deleted from the linked list.

1.7 Packet Queueing Implementation

The router queues all packets waiting for outstanding ARP replies. If a host does not respond to 5 consecutive ARP requests, the queued packet is dropped and an ICMP HOST UNREACHABLE message is sent back to the source of the queued packet. To test this functionality, we should be able to send an ARP request to a dead host. However, the structure of the topology we are experimenting on contains only two servers which are up all the time. For this reason and to simulate this functionality, we ignored all the ARP replies received by the router from the servers.

To link a queued packet to the ARP reply it is waiting for, we implemented three different queues for the three different router interfaces to ensure the proper handling of out of order ARP replies.

2 *Compiling and Running the Code*

The code was implemented using GNU C++ (g++ compiler version 4.3.3) under Linux Ubuntu 9.04 Kernel version 2.6.28-11.

The code is compiled using the make utility with the MakeFile provided by the stub with minor modifications to include the DEBUG feature. The follwoing table demonstrates how to compile the code with and without debugging.

With Debugging:	`make all`
Without Debugging:	`make all NO_DEBUG=1`

To obtain a verbose output with different *router debugging messages* displayed, use compilation with debugging.

3 *Code Optimization Strategy*

Problem	**Solution**
Waiting for ARP replies from unreachable hosts	The router queues all packets waiting for the corresponding ARP replies. The router sends an ICMP HOST UNREACHABLE after 5 unreplied ARP requests.
TTL ≤ 1	If the packet is received with a TTL ≤ 1, the router sends an ICMP PORT UNREACHABLE if the packet is destined to it, otherwise it sends an ICMP TIME EXCEEDED.
Invalid Checksum	If the router received a packet with an invalid checksum it simply drops it.
Association of Packets with ARP replies	The router implements three different queues for the three different interfaces. Each packet is queued in the interface's queue it is going to be forwarded on. In this way the router can associate each queued packet with the ARP reply it is waiting for without explicitly augmenting any new information to the packet.

Other Problems	**Solution**
Authentication File	sr_stub version 0.21 contains an *authentication file missing* bug. This problem was solved by using a newer version (0.22) issued by the Stanford VNS team.
Downloading large files from the servers	Based on feedback from the Stanford VNS team representative, technical problems with the two application servers is preventing the realization of this functionality.

4 *Tests and Results*

4.1 Successful pings of all router interfaces:

4.1.1 eth0: 171.67.241.173

Client Side:

```
root@cesar-laptop:~# ping -c 4 -i 5 -W 60 171.67.241.173
PING 171.67.241.173 (171.67.241.173) 56(84) bytes of data.
64 bytes from 171.67.241.173:  icmp_seq=1 ttl=49 time=2426 ms
64 bytes from 171.67.241.173:  icmp_seq=2 ttl=49 time=1218 ms
64 bytes from 171.67.241.173:  icmp_seq=3 ttl=49 time=1212 ms
64 bytes from 171.67.241.173:  icmp_seq=4 ttl=49 time=1289 ms

--- 171.67.241.173 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 15009ms
rtt min/avg/max/mdev = 1212.422/1536.800/2426.848/514.770 ms
```

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
---------------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
---------------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
***********************************
VNS Welcome Message
***********************************
System status should be normal.  Enjoy!

Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
```

```
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ECHO REPLY Packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****


Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98


*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.1.2 eth0: 171.67.241.175

Client Side:

```
root@cesar-laptop:~# ping -c 4 -i 5 -W 60 171.67.241.175
PING 171.67.241.175 (171.67.241.175) 56(84) bytes of data.
64 bytes from 171.67.241.175:  icmp_seq=1 ttl=49 time=2519 ms
64 bytes from 171.67.241.175:  icmp_seq=2 ttl=49 time=1353 ms
64 bytes from 171.67.241.175:  icmp_seq=3 ttl=49 time=1273 ms
64 bytes from 171.67.241.175:  icmp_seq=4 ttl=49 time=1223 ms

--- 171.67.241.175 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 15010ms
rtt min/avg/max/mdev = 1223.635/1592.608/2519.444/537.122 ms
```

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
---------------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
---------------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
***********************************
VNS Welcome Message
***********************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ECHO REPLY Packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****
```

```
Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98
Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98
Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98


*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.1.3 eth0: 171.67.241.177

Client Side:

```
root@cesar-laptop:~# ping -c 4 -i 5 -W 60 171.67.241.177
PING 171.67.241.177 (171.67.241.177) 56(84) bytes of data.
64 bytes from 171.67.241.177:  icmp_seq=1 ttl=49 time=2651 ms
64 bytes from 171.67.241.177:  icmp_seq=2 ttl=49 time=1256 ms
64 bytes from 171.67.241.177:  icmp_seq=3 ttl=49 time=1213 ms
64 bytes from 171.67.241.177:  icmp_seq=4 ttl=49 time=1351 ms

--- 171.67.241.177 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 15008ms
rtt min/avg/max/mdev = 1213.130/1618.326/2651.807/598.787 ms
```

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
-----------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
-----------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
**********************************
VNS Welcome Message
**********************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.7
4.17] -> ARP Cache entry NOT found
-> Pushing the ICMP ECHO REPLY Packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****
```

```
Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98


*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.2 Successful pings of two servers:

4.2.1 Application Server 1: 171.67.241.174

Client Side:

```
root@cesar-laptop:~# ping -c 4 -i 5 -W 60 171.67.241.174
PING 171.67.241.174 (171.67.241.174) 56(84) bytes of data.
64 bytes from 171.67.241.174:  icmp_seq=1 ttl=46 time=3709 ms
64 bytes from 171.67.241.174:  icmp_seq=2 ttl=46 time=1973 ms
64 bytes from 171.67.241.174:  icmp_seq=3 ttl=46 time=1873 ms
64 bytes from 171.67.241.174:  icmp_seq=4 ttl=46 time=1820 ms

--- 171.67.241.174 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 15004ms
rtt min/avg/max/mdev = 1820.654/2344.356/3709.039/789.820 ms
```

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
---------------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
---------------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
***********************************
VNS Welcome Message
***********************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)
```

```
*** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 1]
****


Received ARP REPLY Packet, length = 42
-> Updating the ARP Cache, [171.67.241.174, 00:bf:84:7c:36:24]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****


Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry found, [171.67.241.174, 00:bf:84:7c:36:24]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry found, [171.67.241.174, 00:bf:84:7c:36:24]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98
```

```
Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry found, [171.67.241.174, 00:bf:84:7c:36:24]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98


*** Removing cache entry, [171.67.241.174, 00:bf:84:7c:36:24] *****



*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.2.2 Application Server 2: 171.67.241.176

Client Side:

```
root@cesar-laptop:~# ping -c 4 -i 5 -W 60 171.67.241.176
PING 171.67.241.176 (171.67.241.176) 56(84) bytes of data.
64 bytes from 171.67.241.176:  icmp_seq=1 ttl=48 time=3746 ms
64 bytes from 171.67.241.176:  icmp_seq=2 ttl=48 time=1882 ms
64 bytes from 171.67.241.176:  icmp_seq=3 ttl=48 time=1858 ms
64 bytes from 171.67.241.176:  icmp_seq=4 ttl=48 time=1831 ms

--- 171.67.241.176 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 15006ms
rtt min/avg/max/mdev = 1831.969/2329.915/3746.572/818.104 ms
```

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
---------------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
---------------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
***********************************
VNS Welcome Message
***********************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [171.67.241.176]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)
```

```
**** Sending ARP REQUEST Packet to [171.67.241.176], length = 42 [Attempt 1]
****


Received ARP REPLY Packet, length = 42
-> Updating the ARP Cache, [171.67.241.176, 00:0c:1e:11:8c:f3]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****


Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [171.67.241.176]
-> ARP Cache entry found, [171.67.241.176, 00:0c:1e:11:8c:f3]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [171.67.241.176]
-> ARP Cache entry found, [171.67.241.176, 00:0c:1e:11:8c:f3]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98
```

```
Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [171.67.241.176]
-> ARP Cache entry found, [171.67.241.176, 00:0c:1e:11:8c:f3]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the forworded packet
-> Sending the forworded packet, length = 98


*** Removing cache entry, [171.67.241.176, 00:0c:1e:11:8c:f3] *****



*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.3 Successful traceroutes to all router interfaces:

4.3.1 eth0: 171.67.241.173

Client Side:

```
root@cesar-laptop:~# traceroute -w 15 171.67.241.173
traceroute to 171.67.241.173 (171.67.241.173), 30 hops max, 60 byte packets
1 169.254.0.1 (169.254.0.1) 431.012 ms 434.109 ms 827.957 ms
2 85.214.1.25 (85.214.1.25) 832.000 ms 831.931 ms 831.863 ms
3 85.214.0.177 (85.214.0.177) 831.877 ms 831.814 ms 831.749 ms
4 ae3-0.bln2-j.mcbone.net (62.104.199.89) 831.523 ms 831.460 ms 831.395 ms
5 ge-5-0-0-0.ffm4-j.mcbone.net (62.104.191.185) 847.614 ms 847.563 ms 847.498
ms
6 po4-2.core01.fra03.atlas.cogentco.com (130.117.14.125) 925.303 ms 593.483 ms
781.377 ms
7 te4-1.mpd03.fra03.atlas.cogentco.com (130.117.0.2) 798.191 ms 798.221 ms
798.204 ms
8 te7-1.mpd01.ymq02.atlas.cogentco.com (154.54.26.141) 892.951 ms 892.973 ms
892.970 ms
9 te9-2.mpd02.ord01.atlas.cogentco.com (154.54.28.9) 901.328 ms
te8-7.mpd02.ord01.atlas.cogentco.com (66.28.4.57) 1618.484 ms 1618.512 ms
10 te3-1.ccr02.mci01.atlas.cogentco.com (154.54.3.202) 1618.657 ms
te8-3.ccr02.mci01.atlas.cogentco.com
(154.54.7.165) 1618.671 ms te2-2.ccr02.mci01.atlas.cogentco.com (154.54.25.77)
1618.769 ms
11 te2-4.ccr02.sfo01.atlas.cogentco.com (154.54.24.109) 1618.867 ms 1618.878 ms
1618.830 ms
12 te9-1.mpd01.sjc04.atlas.cogentco.com (154.54.0.178) 973.328 ms 635.488 ms
2251.101 ms
13 Stanford_University2.demarc.cogentco.com (66.250.7.138) 2251.110 ms 621.825
ms 621.821 ms
14 serv-rtr.Stanford.EDU (171.67.255.138) 597.097 ms 597.100 ms 596.974 ms
15 * * *
16 171.67.241.173 (171.67.241.173) 3180.146 ms 3825.026 ms 3833.581 ms
```

Note that hop number 15, which corresponds to Stanford firewall does not respond to *traceroute*. For more information check the Internet Router Assignment FAQ page.

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
---------------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
---------------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
```

```
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
***********************************
VNS Welcome Message
***********************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 74
-> Packet dropped:  invalid TTL
-> Constructing ICMP PORT UNREACHABLE Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ERROR MESSAGE Packet in the queue, length = 70
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)

Received IP Packet, length = 74
-> Packet dropped:  invalid TTL
-> Constructing ICMP PORT UNREACHABLE Packet


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****

-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ERROR MESSAGE Packet in the queue, length = 70
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)

Received IP Packet, length = 74
-> The IP Packet is UDP Packet
-> Constructing ICMP PORT UNREACHABLE Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ERROR MESSAGE Packet in the queue, length = 70


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****

-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)

Received IP Packet, length = 74
-> The IP Packet is UDP Packet


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****

-> Constructing ICMP PORT UNREACHABLE Packet
-> Searching the ARP Cache for [172.24.74.17]
```

```
-> ARP Cache entry NOT found
-> Pushing the ICMP ERROR MESSAGE Packet in the queue, length = 70
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)

Received IP Packet, length = 74
-> Packet dropped:  invalid TTL


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****

-> Constructing ICMP PORT UNREACHABLE Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ERROR MESSAGE Packet in the queue, length = 70
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****


Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received IP Packet, length = 74
-> The IP Packet is UDP Packet
-> Constructing ICMP PORT UNREACHABLE Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
```

```
-> Updating the ICMP PORT UNREACHABLE Packet
-> Sending the ICMP PORT UNREACHABLE Packet, length = 74

Received IP Packet, length = 74
-> The IP Packet is UDP Packet
-> Constructing ICMP PORT UNREACHABLE Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP PORT UNREACHABLE Packet
-> Sending the ICMP PORT UNREACHABLE Packet, length = 74

Received IP Packet, length = 74
-> The IP Packet is UDP Packet
-> Constructing ICMP PORT UNREACHABLE Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP PORT UNREACHABLE Packet
-> Sending the ICMP PORT UNREACHABLE Packet, length = 74


*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.3.2 eth1: 171.67.241.175

Client Side:

```
root@cesar-laptop:~# traceroute -w 15 171.67.241.175
traceroute to 171.67.241.175 (171.67.241.175), 30 hops max, 60 byte packets
1 169.254.0.1 (169.254.0.1) 421.826 ms 442.430 ms 442.436 ms
2 85.214.1.25 (85.214.1.25) 833.798 ms 833.810 ms 833.402 ms
3 85.214.0.177 (85.214.0.177) 837.531 ms 837.533 ms 837.441 ms
4 ae3-0.bln2-j.mcbone.net (62.104.199.89) 837.270 ms 837.277 ms 837.289 ms
5 ge-5-0-0-0.ffm4-j.mcbone.net (62.104.191.185) 853.735 ms 853.742 ms 853.736
ms
6 po4-2.core01.fra03.atlas.cogentco.com (130.117.14.125) 853.735 ms 440.131 ms
437.923 ms
7 te4-1.ccr02.fra03.atlas.cogentco.com (130.117.2.225) 442.028 ms 902.226 ms
902.236 ms
8 te7-2.ccr02.ymq02.atlas.cogentco.com (66.28.4.21) 968.018 ms 968.041 ms
968.042 ms
9 te9-8.ccr02.ord01.atlas.cogentco.com (66.28.4.137) 996.811 ms 996.909 ms
996.913 ms
10 te3-4.ccr02.mci01.atlas.cogentco.com (154.54.5.173) 1208.769 ms 1208.866 ms
1208.946 ms
11 te2-2.ccr02.sfo01.atlas.cogentco.com (154.54.6.42) 1208.438 ms 1037.933 ms
1208.370 ms
12 te9-1.mpd01.sjc04.atlas.cogentco.com (154.54.0.178) 857.320 ms 623.456 ms
624.254 ms
13 Stanford_University2.demarc.cogentco.com (66.250.7.138) 911.562 ms 911.572
ms 911.577 ms
14 serv-rtr.Stanford.EDU (171.67.255.138) 910.944 ms 624.163 ms 910.834 ms
15 * * *
16 171.67.241.175 (171.67.241.175) 3781.954 ms 4425.168 ms 4425.198 ms
```

Server Side:

It is the same as the case of tracerouting eth0

4.3.3 eth2: 171.67.241.177

Client Side:

```
root@cesar-laptop:~# traceroute -w 15 171.67.241.177
traceroute to 171.67.241.177 (171.67.241.177), 30 hops max, 60 byte packets
1 169.254.0.1 (169.254.0.1) 425.601 ms 425.612 ms 425.619 ms
2 85.214.1.25 (85.214.1.25) 429.607 ms 848.422 ms 852.627 ms
3 85.214.0.177 (85.214.0.177) 852.732 ms 852.740 ms 852.743 ms
4 ae3-0.bln2-j.mcbone.net (62.104.199.89) 852.556 ms 852.578 ms 852.580 ms
5 ge-5-0-0-0.ffm4-j.mcbone.net (62.104.191.185) 865.032 ms 865.044 ms 865.048
ms
6 po4-2.core01.fra03.atlas.cogentco.com (130.117.14.125) 865.052 ms 514.996 ms
514.991 ms
7 te8-8.mpd02.fra03.atlas.cogentco.com (130.117.0.246) 514.992 ms 442.431 ms
442.432 ms
8 te2-4.mpd01.ymq02.atlas.cogentco.com (154.54.28.93) 545.024 ms 545.204 ms
545.292 ms
9 te8-7.mpd02.ord01.atlas.cogentco.com (66.28.4.57) 562.114 ms 927.459 ms
927.550 ms
10 te2-2.ccr02.mci01.atlas.cogentco.com (154.54.25.77) 927.749 ms
te8-3.ccr02.mci01.atlas.cogentco.com (154.54.7.165) 927.606 ms
te2-2.ccr02.mci01.atlas.cogentco.com (154.54.25.77) 927.809 ms
11 te2-4.ccr02.sfo01.atlas.cogentco.com (154.54.24.109) 927.889 ms 927.973 ms
667.865 ms
12 te9-1.mpd01.sjc04.atlas.cogentco.com (154.54.0.178) 667.894 ms * 604.415 ms
13 Stanford_University2.demarc.cogentco.com (66.250.7.138) 608.782 ms 608.795
ms 608.884 ms
14 serv-rtr.Stanford.EDU (171.67.255.138) 608.652 ms 608.641 ms 604.345 ms
15 * * *
16 171.67.241.177 (171.67.241.177) 3201.966 ms 3836.620 ms 3844.847 ms
```

Server Side:

It is the same as the case of tracerouting eth0

4.4 Successful traceroutes to two servers:

4.4.1 Application Server 1: 171.67.241.174

Client Side:

```
root@cesar-laptop:~# traceroute -w 15 171.67.241.174
traceroute to 171.67.241.174 (171.67.241.174), 30 hops max, 60 byte packets
1 169.254.0.1 (169.254.0.1) 460.609 ms 460.607 ms 460.610 ms
2 85.214.1.25 (85.214.1.25) 460.612 ms 465.379 ms 465.385 ms
3 85.214.0.177 (85.214.0.177) 465.388 ms 465.390 ms 465.392 ms
4 ae3-0.bln2-j.mcbone.net (62.104.199.89) 900.293 ms 900.390 ms 900.393 ms
5 ge-5-0-0-0.ffm4-j.mcbone.net (62.104.191.185) 904.598 ms 904.605 ms 904.608
ms
6 po4-2.core01.fra03.atlas.cogentco.com (130.117.14.125) 904.610 ms 1526.464 ms
1526.497 ms
7 te4-1.ccr02.fra03.atlas.cogentco.com (130.117.2.225) 1526.429 ms 1526.474 ms
449.664 ms
8 te7-2.ccr02.ymq02.atlas.cogentco.com (66.28.4.21) 532.375 ms 532.404 ms
532.573 ms
9 te8-7.ccr02.ord01.atlas.cogentco.com (154.54.28.5) 561.416 ms 561.439 ms
561.442 ms
10 te7-4.ccr02.mci01.atlas.cogentco.com (66.28.4.33) 869.425 ms
te4-3.ccr02.mci01.atlas.cogentco.com (154.54.6.201) 869.535 ms 869.538 ms
11 te3-4.ccr02.sfo01.atlas.cogentco.com (154.54.6.161) 869.621 ms 869.707 ms
590.743 ms
12 te4-4.mpd01.sjc04.atlas.cogentco.com (154.54.7.174) 586.420 ms 590.711 ms
590.717 ms
13 Stanford_University2.demarc.cogentco.com (66.250.7.138) 942.278 ms 1221.126
ms 1216.877 ms
14 serv-rtr.Stanford.EDU (171.67.255.138) 942.006 ms 942.139 ms 942.088 ms
15 * * *
16 171.67.241.173 (171.67.241.173) 5565.480 ms 6817.278 ms 8152.774 ms
17 171.67.241.174 (171.67.241.174) 5164.472 ms 5791.798 ms 6834.911 ms
```

Note that hop number 15, which corresponds to Stanford firewall does not respond to *traceroute*. For more information check the Internet Router Assignment FAQ page.

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
---------------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
---------------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
```

```
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
*************************************
VNS Welcome Message
*************************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 74
-> Packet dropped:  invalid TTL
-> Constructing ICMP TIME EXCEEDED Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ERROR MESSAGE Packet in the queue, length = 70
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****


Received IP Packet, length = 74
-> Packet dropped:  invalid TTL
-> Constructing ICMP TIME EXCEEDED Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ERROR MESSAGE Packet in the queue, length = 70
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****


Received IP Packet, length = 74
-> Packet dropped:  invalid TTL
-> Constructing ICMP TIME EXCEEDED Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ERROR MESSAGE Packet in the queue, length = 70
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****


Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
```

```
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received IP Packet, length = 74
-> Forwarding packet, length = 74
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 74
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)

Received IP Packet, length = 74
-> Forwarding packet, length = 74
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 74
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 1]
****



*** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 1]
****


Received IP Packet, length = 74
-> Forwarding packet, length = 74
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 74
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)

Received IP Packet, length = 74
-> Forwarding packet, length = 74
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 74


*** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 1]
****

-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)
```

```
**** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 1]
****


Received ARP REPLY Packet, length = 42
-> Updating the ARP Cache, [171.67.241.174, 00:bf:84:7c:36:24]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 74
-> Updating the popped packet
-> Sending the popped packet, length = 74

Received ARP REPLY Packet, length = 42
-> Entry already exists in the ARP Cache, [171.67.241.174, 00:bf:84:7c:36:24]
-> Popping a packet from the queue, length = 74
-> Updating the popped packet
-> Sending the popped packet, length = 74

Received ARP REPLY Packet, length = 42
-> Entry already exists in the ARP Cache, [171.67.241.174, 00:bf:84:7c:36:24]
-> Popping a packet from the queue, length = 74
-> Updating the popped packet
-> Sending the popped packet, length = 74

Received ARP REPLY Packet, length = 42
-> Entry already exists in the ARP Cache, [171.67.241.174, 00:bf:84:7c:36:24]
-> Popping a packet from the queue, length = 74
-> Updating the popped packet
-> Sending the popped packet, length = 74

Received IP Packet, length = 70
-> Forwarding packet, length = 70
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the forworded packet
-> Sending the forworded packet, length = 70

Received IP Packet, length = 70
-> Forwarding packet, length = 70
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the forworded packet
-> Sending the forworded packet, length = 70

Received IP Packet, length = 70
-> Forwarding packet, length = 70
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the forworded packet
-> Sending the forworded packet, length = 70

Received IP Packet, length = 70
-> Forwarding packet, length = 70
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the forworded packet
-> Sending the forworded packet, length = 70
```

```
...
...
...
...
...

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received ARP REPLY Packet, length = 60
-> Entry already exists in the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70


*** Removing cache entry, [171.67.241.174, 00:bf:84:7c:36:24] *****



*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.4.2 Application Server 2: 171.67.241.176

Client Side:

```
root@cesar-laptop:~# traceroute -w 15 171.67.241.176
traceroute to 171.67.241.176 (171.67.241.176), 30 hops max, 60 byte packets
1 169.254.0.1 (169.254.0.1) 441.364 ms 441.062 ms 440.913 ms
2 85.214.1.25 (85.214.1.25) 440.838 ms 448.932 ms 448.906 ms
3 85.214.0.177 (85.214.0.177) 448.890 ms 448.831 ms 448.744 ms
4 ae3-0.bln2-j.mcbone.net (62.104.199.89) 448.475 ms 448.445 ms 448.375 ms
5 ge-5-0-0-0.ffm4-j.mcbone.net (62.104.191.185) 855.831 ms 855.787 ms 855.566
ms
6 po4-2.core01.fra03.atlas.cogentco.com (130.117.14.125) 855.492 ms 526.774 ms
526.758 ms
7 te9-1.ccr01.fra03.atlas.cogentco.com (130.117.1.30) 526.428 ms 526.451 ms
461.424 ms
8 te2-4.ccr02.ymq02.atlas.cogentco.com (38.20.46.205) 680.228 ms 680.243 ms
680.247 ms
9 te8-7.ccr02.ord01.atlas.cogentco.com (154.54.28.5) 581.516 ms 581.542 ms
581.544 ms
10 te4-3.ccr02.mci01.atlas.cogentco.com (154.54.6.201) 589.836
ms te7-4.ccr02.mci01.atlas.cogentco.com (66.28.4.33) 589.776 ms
te3-4.ccr02.mci01.atlas.cogentco.com (154.54.5.173) 589.707 ms
11 te8-4.ccr02.sfo01.atlas.cogentco.com (154.54.24.117) 651.267 ms 651.289 ms
631.362 ms
12 te4-4.mpd01.sjc04.atlas.cogentco.com (154.54.7.174) 631.113 ms * *
13 Stanford_University2.demarc.cogentco.com (66.250.7.138) 613.140 ms 613.135
ms 1106.840 ms
14 serv-rtr.Stanford.EDU (171.67.255.138) 1102.575 ms 1102.474 ms 1102.499 ms
15 * * *
16 171.67.241.173 (171.67.241.173) 4484.536 ms 5100.730 ms 3814.238 ms
17 171.67.241.176 (171.67.241.176) 5923.121 ms 5306.785 ms 6572.401 ms
```

Note that hop number 15, which corresponds to Stanford firewall does not respond to *traceroute*. For more information check the Internet Router Assignment FAQ page.

Server Side:

It is the same as the case of tracerouting Application Server 1

4.5 Unsuccessful *telnet* connections to the application servers:

4.5.1 Application Server 1: 171.67.241.174

Client Side:

```
root@cesar-laptop:~# telnet 171.67.241.174
Trying 171.67.241.174...
telnet:  Unable to connect to remote host:  Connection refused
```

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
-------------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
-------------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
***********************************
VNS Welcome Message
***********************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 74
-> Forwarding packet, length = 74
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 74
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 1]
****
```

```
Received ARP REPLY Packet, length = 42
-> Updating the ARP Cache, [171.67.241.174, 00:bf:84:7c:36:24]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 74
-> Updating the popped packet
-> Sending the popped packet, length = 74

Received IP Packet, length = 70
-> Forwarding packet, length = 70
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 70
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****


Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

Received IP Packet, length = 74
-> Forwarding packet, length = 74
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry found, [171.67.241.174, 00:bf:84:7c:36:24]
-> Updating the forworded packet
-> Sending the forworded packet, length = 74

Received IP Packet, length = 70
-> Forwarding packet, length = 70
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the forworded packet
-> Sending the forworded packet, length = 70


*** Removing cache entry, [171.67.241.174, 00:bf:84:7c:36:24] *****


*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.5.2 Application Server 2: 171.67.241.176

Client Side:

```
root@cesar-laptop:~# telnet 171.67.241.176
Trying 171.67.241.176...
telnet:  Unable to connect to remote host:  Connection refused
```

Server Side:

It is the same as the case of *telneting* Application Server 1

4.6 Unsuccessful TCP connections to router:

4.6.1 telnet

Client Side:

```
root@cesar-laptop:~# telnet 171.67.241.173
Trying 171.67.241.173...
telnet:  Unable to connect to remote host:  Connection refused
```

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
-----------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
-----------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
***********************************
VNS Welcome Message
***********************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 74
-> The IP Packet is TCP Packet
-> Constructing ICMP PORT UNREACHABLE Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ERROR MESSAGE Packet in the queue, length = 70
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****
```

```
Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70


*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.6.2 ssh

Client Side:

```
root@cesar-laptop:~# ssh 171.67.241.173
ssh:  connect to host 171.67.241.173 port 22:  Connection refused
```

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
-----------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
-----------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
***********************************
VNS Welcome Message
***********************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 74
-> The IP Packet is TCP Packet
-> Constructing ICMP PORT UNREACHABLE Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ERROR MESSAGE Packet in the queue, length = 70
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)

*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****

Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 70
-> Updating the popped packet
-> Sending the popped packet, length = 70

*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.7 Successful handling of ARP requests and replies:

4.7.1 For existing destinations

Client Side:

```
root@cesar-laptop:~# ping -c 1 171.67.241.174
PING 171.67.241.174 (171.67.241.174) 56(84) bytes of data.
64 bytes from 171.67.241.174:  icmp_seq=1 ttl=46 time=4714 ms
--- 171.67.241.174 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 4714.721/4714.721/4714.721/0.000 ms
```

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
-----------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
-----------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
***********************************
VNS Welcome Message
***********************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 1]
****
```

```
Received ARP REPLY Packet, length = 42
-> Updating the ARP Cache, [171.67.241.174, 00:bf:84:7c:36:24]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****


Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98


*** Removing cache entry, [171.67.241.174, 00:bf:84:7c:36:24] *****



*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.7.2 For non-existing destinations

Client Side:

```
root@cesar-laptop:~# ping -c -W 120 1 171.67.241.174
PING 171.67.241.174 (171.67.241.174) 56(84) bytes of data.
64 bytes from 171.67.241.174:  icmp_seq=1 ttl=46 time=4714 ms
--- 171.67.241.174 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 4714.721/4714.721/4714.721/0.000 ms
```

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
-----------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
-----------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
***********************************
VNS Welcome Message
***********************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 98
-> Forwarding packet, length = 98
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry NOT found
-> Pushing forwarded packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 1]
****
```

```
**** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 2]
****



*** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 3]
****



*** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 4]
****



*** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 5]
****

-> Constructing ICMP HOST UNREACHABLE Packet
-> Searching the ARP Cache for [171.67.241.174]
-> ARP Cache entry NOT found
-> Pushing the ICMP ERROR MESSAGE Packet in the queue, length = 70
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [171.67.241.174], length = 42 [Attempt 1]
****


Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98


*** Removing cache entry, [171.67.241.174, 00:bf:84:7c:36:24] *****



*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```

4.7.3 ARP table entries time out

Client Side:

```
root@cesar-laptop:~# ping -c 5 -i 5 171.67.241.173
PING 171.67.241.173 (171.67.241.173) 56(84) bytes of data.
64 bytes from 171.67.241.173:  icmp_seq=1 ttl=49 time=9614 ms
64 bytes from 171.67.241.173:  icmp_seq=2 ttl=49 time=7234 ms
64 bytes from 171.67.241.173:  icmp_seq=3 ttl=49 time=5463 ms
64 bytes from 171.67.241.173:  icmp_seq=4 ttl=49 time=6267 ms
64 bytes from 171.67.241.173:  icmp_seq=5 ttl=49 time=10221 ms

--- 171.67.241.173 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 20016ms
rtt min/avg/max/mdev = 5463.599/7760.379/10221.693/1858.822 ms
```

Server Side:

```
Using VNS sr stub code revised 2009-10-31 (rev 0.22)
Loading routing table
-----------------------------------------
Destination     Gateway     Mask     Iface
0.0.0.0     172.24.74.17     0.0.0.0     eth0
171.67.241.174     171.67.241.174     255.255.255.255     eth1
171.67.241.176     171.67.241.176     255.255.255.255     eth2
-----------------------------------------
Client root connecting to Server 171.67.71.18:3250
Requesting topology 8868
(falling back to original port for -t type connection)
Router interfaces:
eth0 HWaddr00:07:72:dd:dc:29
inet addr 171.67.241.173
eth1 HWaddr00:85:bf:09:7a:ee
inet addr 171.67.241.175
eth2 HWaddr00:94:5f:85:7f:54
inet addr 171.67.241.177
<-- Ready to process packets -->
**********************************
VNS Welcome Message
**********************************
Received ARP REQUEST Packet, length = 42
-> Constructing ARP REPLY Packet
-> Sending ARP REPLY Packet, length = 42

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ECHO REPLY Packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)

*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****
```

```
Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98

Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry found, [172.24.74.17, 00:e0:81:04:26:d9]
-> Updating the ICMP ECHO REPLY Packet
-> Sending the ICMP ECHO REPLY Packet, length = 98


*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] ****


Received IP Packet, length = 98
-> The IP Packet is ICMP ECHO REQUEST
-> Constructing ICMP ECHO REPLY Packet
-> Searching the ARP Cache for [172.24.74.17]
-> ARP Cache entry NOT found
-> Pushing the ICMP ECHO REPLY Packet in the queue, length = 98
-> Constructing ARP REQUEST Packet
-> Running the ARP REQUESTs thread for 5 attempt(s)


*** Sending ARP REQUEST Packet to [172.24.74.17], length = 42 [Attempt 1] ****


Received ARP REPLY Packet, length = 60
-> Updating the ARP Cache, [172.24.74.17, 00:e0:81:04:26:d9]
-> Stopping the ARP REQUESTs thread
-> Popping a packet from the queue, length = 98
-> Updating the popped packet
-> Sending the popped packet, length = 98


*** Removing cache entry, [172.24.74.17, 00:e0:81:04:26:d9] *****
```