ICSI416/516 Project 3 – Network traffic analysis

By: Meet Rajivbhai Parikh (MP468851)
mparikh2@albany.edu

Analysis of Wireshark trace results:

On the high-level view and analysis of the given trace file, we can get following information,

Summary of the trace:

- There are total **1330 packets** in the trace.
- There are total of 67 hosts.
- There are total of 53 DLL/MAC addresses.
- There are total of 67 IP addresses.
- There are several types of packets in the given trace: DNS Query Response, HSRP etc.

Analysis:

The types of packets can be get by going through several IP and its protocols. The following figure describes all the types of packets in the trace file.

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	1330	100.0	329555	57 k	0	0	0
Ethernet	100.0	1330	100.0	329555	57 k	0	0	0
Logical-Link Control	2.0	26	0.8	2529	444	0	0	0
Spanning Tree Protocol	1.7	23	0.4	1380	242	23	1380	242
Cisco Discovery Protocol	0.2	3	0.3	1149	201	3	1149	201
Internet Protocol Version 6	5.4	72	4.0	13249	2326	0	0	0
User Datagram Protocol	3.2	42	3.2	10705	1879	0	0	0
Multicast Domain Name System	1.4	18	1.7	5630	988	18	5630	988
Link-local Multicast Name Resolution	0.9	12	0.3	1068	187	12	1068	187
DHCPv6	0.6	8	0.4	1179	207	8	1179	207
Data	0.3	4	0.9	2828	496	4	2828	496
Internet Control Message Protocol v6	2.3	30	0.8	2544	446	30	2544	446
Internet Protocol Version 4	80.3	1068	92.3	304045	53 k	0	0	0
User Datagram Protocol	45.7	608	22.7	74888	13 k	0	0	0
Network Time Protocol	1.2	16	0.4	1440	252	16	1440	252
NetBIOS Name Service	4.8	64	1.8	5888	1033	64	5888	1033
NetBIOS Datagram Service	1.1	15	1.1	3756	659	0	0	0
SMB (Server Message Block Protocol)	1.1	15	1.1	3756	659	0	0	0
SMB MailSlot Protocol	1.1	15	1.1	3756	659	0	0	0
Microsoft Windows Browser Protocol	1.1	15	1.1	3756	659	15	3756	659
Multicast Domain Name System	2.0	27	2.1	6766	1188	27	6766	1188
Link-local Multicast Name Resolution	0.9	12	0.3	828	145	12	828	145
Dropbox LAN sync Discovery Protocol	2.7	36	3.9	12836	2254	36	12836	2254
Domain Name System	9.2	122	5.6	18319	3216	122	18319	3216
Data	4.7	62	2.3	7715	1354	62	7715	1354
Cisco Hot Standby Router Protocol	17.9	238	4.5	14756	2591	238	14756	2591
Canon BJNP	0.9	12	0.2	720	126	12	720	126
Bootstrap Protocol	0.3	4	0.6	1864	327	4	1864	327
Transmission Control Protocol	29.5	393	67.9	223747	39 k	260	166696	29 k
Internet Group Management Protocol	0.2	2	0.0	108	18	2	108	18
Internet Control Message Protocol	4.9	65	1.6	5302	931	65	5302	931
Configuration Test Protocol (loopback)	0.3	4	0.1	240	42	0	0	0
Data	0.3	4	0.1	240	42	4	240	42
Address Resolution Protocol	12.0	160	2.9	9492	1666	160	9492	1666

Here, the details in every protocol shows the information about the types. The percent packets field shows how much percentage of total protocol is the specific type. The size and rate of sending and receiving of file is also easily described.

Several DLL and MAC addresses are also traceable from the endpoints in the trace file. It is accessible from the statistics of the trace file. If we can trace the MAC addresses of all the endpoints in the trace, we can get following addresses as result:

1. "00:00:0c:07:ac:01"
2. "00:03:ba:08:63:55"
3. "00:03:ba:09:1c:87"
4. "00:03:ba:12:db:29"
5. "00:08:e3:ff:fd:90"
6. "00:09:3d:12:2d:af"
7. "00:11:11:a3:77:30"
8. "00:11:43:ea:ea:1e"
9. "00:13:72:1a:2d:db"
10. "00:18:ba:84:dc:08"
11. "00:1a:4b:26:14:4d"
12. "00:22:41:27:b4:84"
13. "00:23:5e:69:8a:cc"
14. "00:23:5e:c8:b8:e6"
15. "00:23:5e:c8:b9:c2"
16. "01:00:0c:cc:cc:cc"
17. "01:00:5e:00:00:01"
18. "01:00:5e:00:00:02"
19. "01:00:5e:00:00:16"
20. "01:00:5e:00:00:fb"
21. "01:00:5e:00:00:fc"
22. "01:00:5e:7f:ff:fa"
23. "01:80:c2:00:00:00"
24. "08:00:20:f0:5c:f5"
25. "0c:4d:e9:c8:fb:2d"
26. "1c:c1:de:cb:39:28"
27. "28:d2:44:e5:1a:fe"
28. "30:8d:99:a8:fd:7c"
29. "33:33:00:00:00:02"
30. "33:33:00:00:00:0c"
31. "33:33:00:00:00:16"
32. "33:33:00:00:00:fb"
33. "33:33:00:01:00:02"
34. "33:33:00:01:00:03"
35. "33:33:ff:09:39:95"
36. "33:33:ff:6b:2d:95"

37. "34:17:eb:c3:73:73"
38. "34:17:eb:c3:c1:b7"
39. "34:17:eb:c3:c1:ca"
40. "3c:08:f6:8d:dd:56"
41. "74:26:ac:a3:2c:74"
42. "90:b1:1c:84:3c:bb"
43. "98:90:96:a6:9a:37"
44. "98:90:96:a6:9b:fe"
45. "98:90:96:c8:1e:b1"
46. "a8:20:66:3e:b6:9a"
47. "b8:ca:3a:b9:91:2f"
48. "d4:ae:52:c1:f3:60"
49. "d4:be:d9:a5:c7:cb"
50. "f0:de:f1:09:39:95"
51. "f8:b1:56:c0:25:d3"
52. "f8:b1:56:c0:2b:0f"
53. "ff:ff:ff:ff:ff:ff"

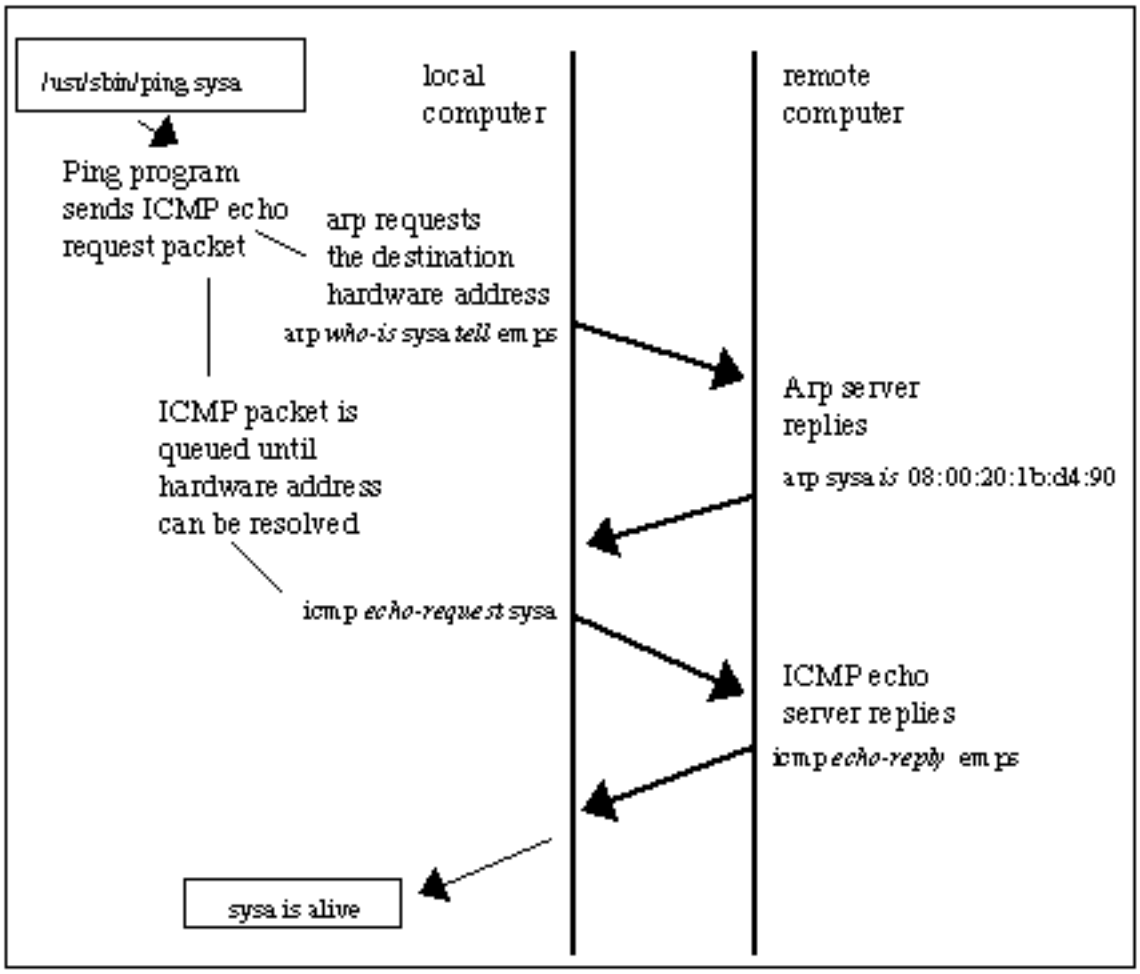
Here, a media access control address (MAC address), also called physical address, is a unique identifier assigned to network interfaces for communications on the physical network segment. All three numbering systems use the same format and differ only in the length of the identifier. So, these addresses are physical addresses of each machine being accessed in the route. Also, we can get the IP addresses of each host. These IP addresses can also be accessed with wireshark in the given trace. As stated earlier, the 67 IP addresses are also accessible by statistics of the given trace file.

1. "0.0.0.0"
2. "10.200.2.25"
3. "10.200.2.26"
4. "50.112.138.61"
5. "91.189.89.199"
6. "91.189.94.4"
7. "128.111.4.53"
8. "128.111.52.118"
9. "128.111.252.149"
10. "128.111.252.169"
11. "137.164.26.200"
12. "137.164.26.238"
13. "169.226.1.110"
14. "169.226.2.3"
15. "169.226.2.22"
16. "169.226.2.74"
17. "169.226.2.76"
18. "169.226.2.77"

```
19.     "169.226.2.78"
20.     "169.226.2.92"
21.     "169.226.2.93"
22.     "169.226.2.94"
23.     "169.226.2.95"
24.     "169.226.2.104"
25.     "169.226.2.131"
26.     "169.226.2.137"
27.     "169.226.2.147"
28.     "169.226.2.151"
29.     "169.226.2.165"
30.     "169.226.2.166"
31.     "169.226.2.255"
32.     "169.226.13.67"
33.     "169.226.13.75"
34.     "169.226.63.105"
35.     "169.226.255.255"
36.     "172.217.3.4"
37.     "172.217.3.14"
38.     "198.71.45.15"
39.     "198.71.45.16"
40.     "198.71.45.21"
41.     "199.109.8.25"
42.     "199.109.11.38"
43.     "216.58.217.174"
44.     "224.0.0.1"
45.     "224.0.0.2"
46.     "224.0.0.22"
47.     "224.0.0.251"
48.     "224.0.0.252"
49.     "239.255.255.250"
50.     "255.255.255.255"
51.     "::"
52.     "fe80::1ec1:deff:fecb:3928"
53.     "fe80::328d:99ff:fea8:fd7c"
54.     "fe80::412a:3007:21fb:a58a"
55.     "fe80::4805:8fb4:36a4:fbfe"
56.     "fe80::7626:acff:fea3:2c74"
57.     "fe80::b9cb:3a29:ec88:f2a2"
58.     "fe80::b9e5:7723:f358:4219"
59.     "fe80::f2de:f1ff:fe09:3995"
60.     "ff02::2"
61.     "ff02::c"
62.     "ff02::16"
```

63. "ff02::fb"
64. "ff02::1:2"
65. "ff02::1:3"
66. "ff02::1:ff09:3995"
67. "ff02::1:ff6b:2d95"

As we can see, there are total 67 IP addresses in the trace. Now, as stated above, the two types of addresses i.e. Physical and IP addresses we can get. The mapping of these two types of addresses can be done with ARP (Address Resolution Protocol). The Address Resolution Protocol (ARP) request is initiated for resolution of network layer addresses into link layer addresses, a critical function in multiple-access networks. ARP is used for mapping a network address (e.g. an IPv4 address) to a physical address like an Ethernet address (also named MAC address). ARP has been implemented with many combinations of network and data link layer technologies. If expected destination is not in reachable the packet will be dropped out. The ARP can be used when a computer tries to contact a remote computer on the same LAN (known as "sysa") using the "ping" program.



This image shows the working of ARP, as it requests and it is assumed that no previous IP datagrams have been received form this computer, and therefore ARP must first be used to identify the MAC address of the remote computer. Now, by the looking at trace file, we can say that Ethernet adapter was used to capture the trace, and not the 802.11 wireless card. By looking at the given trace, we can say that it was collected on a specific pattern: First, a host sends a packet to the local hosts. Now, these local hosts are selected from DNS server, i.e. 169.226.2.100. The range of this DNS server can be selected by its range of given DNS server address. After the local hosts are selected, the packets are transmitted to the hosts. Now, after packets have been successfully received at local hosts, the remote hosts can be chosen from the destination addresses. Now, the packets are sent to the remote host.

So, HOST → LOCAL HOSTS (Many) → REMOTE HOSTS (By Destination Address).

This methodology is used for sending packets from source to destination. We can also get the machine and IP address of the machine, on which the trace was collected. The MAC address is: fo:de:f1:09:39:95 and the IP address for the machine is: 169.226.2.166 . This can be accessed from 4th packet. From the trace, we can see that this packet is initializing the transmission. Here, the DHCP server is requesting. So, we can get DNS, MAC and IP address of the machine. So, we can say that the trace was collected on this machine.
The network mask is: 255.255.255.0

The default gateway is: 169.226.2.1

The DNS server is: 169.226.2.22 & 169.226.1.100

DHCP server is: 169.226.2.22

Following hosts are on the local network:

1. "169.226.2.3"
2. "169.226.2.22"
3. "169.226.2.74"
4. "169.226.2.76"
5. "169.226.2.77"
6. "169.226.2.78"
7. "169.226.2.92"
8. "169.226.2.93"
9. "169.226.2.94"
10. "169.226.2.95"
11. "169.226.2.104"
12. "169.226.2.131"
13. "169.226.2.137"
14. "169.226.2.147"
15. "169.226.2.151"

16. "169.226.2.165"
17. "169.226.2.166"
18. "169.226.2.255"
19. "2620:123:40ff:12d::c"
20. "2620:123:40ff:12d::6e"
21. "2620:123:40ff:2e::64"

So, there are 21 hosts on local network. There are total 67 hosts, and 21 are local. So, the remaining 46 entries in IP addresses are remote host. The list of remote hosts is:

1. "0.0.0.0"
2. "10.200.2.25"
3. "10.200.2.26"
4. "50.112.138.61"
5. "91.189.89.199"
6. "91.189.94.4"
7. "128.111.4.53"
8. "128.111.52.118"
9. "128.111.252.149"
10. "128.111.252.169"
11. "137.164.26.200"
12. "137.164.26.238"
13. "169.226.1.110"
14. "169.226.13.67"
15. "169.226.13.75"
16. "169.226.63.105"
17. "169.226.255.255"
18. "172.217.3.4"
19. "172.217.3.14"
20. "198.71.45.15"
21. "198.71.45.16"
22. "198.71.45.21"
23. "199.109.8.25"
24. "199.109.11.38"
25. "216.58.217.174"
26. "224.0.0.1"
27. "224.0.0.2"
28. "224.0.0.22"
29. "224.0.0.251"
30. "224.0.0.252"
31. "239.255.255.250"
32. "255.255.255.255"
33. "::"
34. "fe80::1ec1:deff:fecb:3928"

35. "fe80::328d:99ff:fea8:fd7c"
36. "fe80::412a:3007:21fb:a58a"
37. "fe80::4805:8fb4:36a4:fbfe"
38. "fe80::7626:acff:fea3:2c74"
39. "fe80::b9cb:3a29:ec88:f2a2"
40. "fe80::b9e5:7723:f358:4219"
41. "fe80::f2de:f1ff:fe09:3995"
42. "ff02::2"
43. "ff02::c"
44. "ff02::16"
45. "ff02::1:ff09:3995"
46. "ff02::1:ff6b:2d95"

Besides, there are several applications used on hosts:

1. Dropbox
2. Microsoft Windows Browser
3. Canon BJNP scanner

And, there are several services used, as well:

1. Apple
2. YouTube
3. Facebook
4. Google

There are 15 hops away from remote hosts. Total hops are 64. And Time to Live are 49. So, 64-49 = 15 hops are away from remote hosts. There are two most remote hosts: 91.189.94.4 and 91.189.89.199.

In the given trace file, there is no IP fragmentation which can be obtained by using the filter: ip.flags.mf == 1 or ip.frag_offset gt 0 . By the way, there are total 562 packets in which the “Don’t fragment” bit is set. This can be obtained by using the filter: ip.flags.df==1 .

Conclusion:

So, by this trace of given trace file, we can see that the path of a packet in route can be seen and analyzed by Wireshark. Here, as seen in above theories and descriptions, there are several questions, which can be answered by Network Traffic Analysis. I can conclude that UDP protocol was used in the given trace file. As discussed, the packet was transferred from source to local host to remote host by IP address. I believe this tracing can be used for Future Network Traffic Analysis.

References:

Image was taken from: https://scontent.xx.fbcdn.net/v/t1.0-9/13102640_1599403707055074_2280304566385241713_n.jpg?oh=b7ab16d532b7b309d7a9abf1d3e60815&oe=579F4C84&dl=1