Internet Traffic Analysis using Wireshark

Tan Wei Xuan (49003140)
tanweixuan@postech.ac.kr

May 3, 2019

1 Project Overview

Wireshark 3.0.1 is used to analyze the network traffic captured over a specified period of time.

2 Traffic Analysis using Wireshark

Analysis is done on the captured network traffic within the provided tracefile. Network Traffic is captured for a period of **59.088** seconds.

2.1 Total Number of Packets and Bytes

Statistics

Measurement	Captured	Displayed	Marked
Packets	4137680	4137680 (100.0%)	—
Time span, s	59.088	59.088	—
Average pps	70026.3	70026.3	—
Average packet size, B	754	754	—
Bytes	3120951509	3120951509 (100.0%)	0
Average bytes/s	52 M	52 M	—
Average bits/s	422 M	422 M	—

Figure 1: *Capture Statistics (Statistics → Capture File Properties)*

The total number of packets being captured between a **59 second period** is **4137680**. The total number of Bytes between captured is **3120951509**. This information can be obtained thorugh *Stastics → Capture File Properties*.

2.2 Time Difference between First and Last Packet

frame.number == 1 || frame.number == 4137680

No.	Time	Source	Destination	Protocol
1	0.000000	141.223.170.141	112.162.88.78	TCP
4137680	59.087530	95.39.36.34	141.223.60.4	SIP

We know that the total number of packets being captured is **4137680**. As such, the first frame be captured will be **1** and the last frame being captured will be **4137680**. We can filter out these two frames by applying the filter, ***(frame.number == 1) || (frame.number == 4137680)***. From the filtered results, we can see that the first packet is being transmitted at **0.0** seconds while the last packet is being transmitted at **59.087530** seconds. As such, the **time difference** between the **first** and **last** packet is **59.087530 seconds**

2.3 The number of packet and total bytes of TCP, UDP and ICMP traffic

Protocol	Percent Packets	Packets	Percent Bytes	Bytes
Frame	100.0	4137680	8.6	269186417
Ethernet	100.0	4137782	1.9	57928948
Internet Protocol Version 4	100.0	4137680	2.7	82753600
User Datagram Protocol	61.2	2533291	0.6	20266328
Transmission Control Protocol	37.9	1568769	1.4	43878139
Internet Protocol Version 6	0.1	3870	0.0	112314
Internet Control Message Protocol	0.8	31256	0.0	978571

Figure 2: *TCP,UDP and ICMP Proticol Hierarchy (Statistics $\rightarrow$ Protocol Hierarchy)*

The entirety of the network traffic is being transmitted through **IPv4** as it takes up **100%** of the total packets. The total number of packet and total bytes of IPv4 TCP, UDP and Internel Control Message Protocol (ICMP) traffic are as follow:

1. **TCP**
   The total number of packets being transmitted using TCP is **1568769** and the total number of bytes being transmitted is **43878139**. TCP takes up **37.9%** of total network traffic.

2. **UDP**
   The total number of packets being transmitted using UDP is **2533291** and the total number of bytes being transmitted is **20266328**. UDP takes up **61.2%** of total network traffic.

3. **ICMP**
   The total number of packets being transmitted using ICMP is **31256** and the total number of bytes being transmitted is **978571**. ICMP takes up **0.8%** of total network traffic.

For this captured netwrok traffic, IPv4 TCP and UDP take the majority of the percent of total packets, with **UDP taking up most of the traffic *(61%)***. Most of the traffic is probably allocated for UDP services such as Media Streaming, VoIP, etc. This information can be obtained thorugh *Statistics $\rightarrow$ Protocol Hierarchy*.

2.4 Total Number of Packets and Bytes of each end host

Ethernet · 8 | IPv4 · 113434 | IPv6 | TCP · 36092 | UDP · 149198

Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes
00:00:00_00:00:00	204	217 k	102	108 k	102	108 k
Cisco_ac:3a:80	4,137,668	3120 M	2,744,902	2214 M	1,392,766	906 M
Cisco_b5:fc:80	4,137,667	3120 M	1,392,778	906 M	2,744,889	2214 M
IPv4mcast_0a	25	1850	0	0	25	1850
43:9c:8c:42:1e:ab	1	74	1	74	0	0
47:86:02:8f:0e:25	1	74	0	0	1	74
94:37:f5:08:ae:3c	1	70	1	70	0	0
d8:78:02:10:9b:72	1	70	0	0	1	70

Figure 3: *End Points (Statistics → Endpoints)*

From the figure above, the main end hosts are **Cisco_ac:3a:80** and **Cisco_b5:fc:80**. As there **few Ethernet end nodes** with **many IP end nodes**, we have **two routers** that sends/receives packets from many remote devices during the entire capture period. For **Cisco_ac:3a:80**, the total amount of **transmitted packets** is **2,744,902** and the total amount of **received packets** is **1,392,766**. The total number of **transmitted bytes** is **2214MB** and the total number of **received bytes** is **906MB**.
For **Cisco_b5:fc:80**, the total amount of **transmitted packets** is **1,392,778** and the total amount of **received packets** is **2,744,902**. The total number of **transmitted bytes** is **906MB** and the total nnumber of **received bytes** is **2214MB**.
All transmitted packets &bytes of **Cisco_b5:fc:80** are received by **Cisco_b5:fc:80** and vice versa.

2.5 The number of packet and total bytes of FTP, SSH, DNS, and HTTP

In order to identify the total number of packets and bytes trasmitted using File Transfer Protocol (FTP), Secure Shell (SSH), Domain Name System (DNS) and Hypertext Transfer Protocol (HTTP), we have to know the reserved port numbres that are being allocated to these services. The ports allocated for each of these services are as shown in the diagram below:

1. **FTP**
   The port numbers that are being reserved for FTP is **20** and **21**. FTP uses two TCP connections for communication. Port 20 to pass control information and Port 21 to send the data files between the client and the server. The connection has to be established before the files can actually be sent across. As FTP is a TCP connection and thus in order to analyze only the traffic on ports 21 and 22, we apply a display filter to the entire captured traffic *(tcp.port == 20|| tcp.port == 21)*

Port	Usage
20	FTP - Data
21	FTP - Control
22	SSH
23	Telnet
25	SMTP
37	TIME
49	TACACS
53	DNS
67	DHCP Server (UDP)
68	DHCP Client (UDP)
69	TFTP (UDP)
79	Finger
80	HTTP
110	POP3
111	RPC (UDP)
119	NNTP
123	NTP
137-139	NetBIOS
161	SNMP
162	Trap (UDP)

Figure 4: *Well-Known Ports (https://networking.ringofsaturn.com/Protocols/wellknownports.php)*

(tcp.port==21 || tcp.port==20)

No.	Time	Source	Destination	Protocol	Length
509072	7.066944	121.180.215.243	141.223.49.78	FTP	90
1183616	16.735828	195.2.240.180	141.223.30.53	TCP	62
1183617	16.735835	195.2.240.180	141.223.30.53	TCP	62
1183628	16.735947	195.2.240.180	141.223.30.53	TCP	62
1183629	16.735954	195.2.240.180	141.223.30.53	TCP	62
1183641	16.736030	195.2.240.180	141.223.30.53	TCP	62
1183642	16.736036	195.2.240.180	141.223.30.53	TCP	62
1183647	16.736104	195.2.240.180	141.223.30.53	TCP	62

Figure 5: *Filter By FTP Port Numbers*

Statistics

Measurement	Captured	Displayed
Packets	4137680	127 (0.0%)
Time span, s	59.088	47.645
Average pps	70026.3	2.7
Average packet size, B	754	63
Bytes	3120951509	7946 (0.0%)
Average bytes/s	52 M	166
Average bits/s	422 M	1334

Figure 6: *FTP Capture Statistics (Stastics → Capture File Properties)*

From the capture statistics, we can tell that the total number of packets being captured for FTP is **127** and the total number of Bytes being captured is **7946**. FTP takes up close to **0%** of the entire network traffic. From Figure 4, we can tell that there are two Source Destination, **121.180.215 and 195.2.240.180** and two destination addresses, **141.223.49.78 and 141.223.30.53**.This sequence of captured traffic is probably the exchange of files (63.568kb) between POSTECH webpages and a client's computer as the IP prefix for POSTECH is *141.223.xx.xx*.

2. **SSH**

The port number that is being reserved for SSH is **22**. As SSH is a TCP connection and thus in order to analyze only the traffic on port 22, we apply a display filter to the entire captured traffic *(tcp.port == 22)*

(tcp.port==22)

No.	Time	Source	Destination	Protocol	Length
3843	0.053993	141.223.175.232	222.122.81.122	TCP	114
4476	0.062046	222.122.81.122	141.223.175.232	TCP	66
9437	0.126503	141.223.175.232	222.122.81.122	TCP	114
10007	0.133305	222.122.81.122	141.223.175.232	TCP	66
15535	0.205917	141.223.175.232	222.122.81.122	TCP	114
16123	0.213105	222.122.81.122	141.223.175.232	TCP	66
20789	0.268580	141.223.200.153	1.97.49.96	TCP	78
26633	0.345960	141.223.175.232	222.122.81.122	TCP	114
27053	0.352743	222.122.81.122	141.223.175.232	TCP	66

Figure 7: *Filter By SSH Port Numbers*

Statistics

Measurement	Captured	Displayed
Packets	4137680	799 (0.0%)
Time span, s	59.088	58.909
Average pps	70026.3	13.6
Average packet size, B	754	166
Bytes	3120951509	132704 (0.0%)
Average bytes/s	52 M	2252
Average bits/s	422 M	18 k

Figure 8: *SSH Capture Statistics (Statistics → Capture File Properties)*

From the capture statistics, we can tell that the total number of packets being captured for SSH is **799** and the total number of Bytes being captured is **132704**. SSH takes up close to **0%** of the entire network traffic. SSH is typically used to log into a remote machine and execute commands and can be used to transfer files using the associated SSH file transfer (SFTP) or secure copy protocols (SCP). SSH uses the client-server model.

Info
59320 → 22 [PSH, ACK] Seq=1 Ack=1 Win=1002 Len=48 TSval=13844…
22 → 59320 [ACK] Seq=1 Ack=49 Win=379 Len=0 TSval=2084580922 …
59320 → 22 [PSH, ACK] Seq=49 Ack=1 Win=1002 Len=48 TSval=1384…
22 → 59320 [ACK] Seq=1 Ack=97 Win=379 Len=0 TSval=2084580993 …
59320 → 22 [PSH, ACK] Seq=97 Ack=1 Win=1002 Len=48 TSval=1384…

Figure 9: *SSH Traffic Info*

From Figure 8, the [ACK] indicates that a host is acknowledging having received some data, and the [PSH,ACK] indicates the host is acknowledging receipt of some previous data and also transmitting some more data. This sequence of captured data is thus probably the transfers of files between a Postech Server (Identifiable by IP Address Prefix) and a Client's Computer.

3. **DNS**

A DNS server listens for requests on port **53 (both UDP and TCP)**. In order to analyze both TCP and UDP traffic on port 53, we apply a display filter to the entire captured traffic *(tcp.port* == 53 || *udp.port* == 53*)*

(tcp.port== 53 || udp.port == 53)

No.	Time	Source	Destination	Protocol	Length
98292	1.325406	141.223.82.102	168.126.63.2	DNS	72
98632	1.329624	141.223.1.33	192.112.36.4	DNS	87
98675	1.330268	141.223.1.33	192.112.36.4	DNS	87
98676	1.330301	141.223.1.33	192.112.36.4	DNS	88
98684	1.330425	141.223.1.33	192.112.36.4	DNS	88
98752	1.331466	141.223.1.34	192.43.172.30	DNS	90
98766	1.331598	141.223.1.34	192.43.172.30	DNS	90
98792	1.331966	141.223.1.34	119.205.216.45	DNS	91
98847	1.332616	141.223.1.34	119.205.216.45	DNS	84

Figure 10: *Filter By DNS Port Numbers*

Statistics

Measurement	Captured	Displayed
Packets	4137680	31076 (0.8%)
Time span, s	59.088	59.081
Average pps	70026.3	526.0
Average packet size, B	754	124
Bytes	3120951509	3850567 (0.1%)
Average bytes/s	52 M	65 k
Average bits/s	422 M	521 k

Figure 11: *DNS Capture Statistics (Statistics → Capture File Properties)*

From the capture statistics, we can tell that the total number of packets being captured for DNS is **31076** and the total number of Bytes being captured is **3850567**. DNS takes up about **0.8%** of the entire network traffic.

Statistics

Measurement	Captured	Displayed
Packets	4137680	27129 (0.7%)
Time span, s	59.088	59.081
Average pps	70026.3	459.2
Average packet size, B	754	131
Bytes	3120951509	3560143 (0.1%)
Average bytes/s	52 M	60 k
Average bits/s	422 M	482 k

Figure 12: *DNS (UDP) Capture Statistics*

Statistics

Measurement	Captured	Displayed
Packets	4137680	3947 (0.1%)
Time span, s	59.088	58.848
Average pps	70026.3	67.1
Average packet size, B	754	74
Bytes	3120951509	290424 (0.0%)
Average bytes/s	52 M	4935
Average bits/s	422 M	39 k

Figure 13: *DNS (TCP) Capture Statistics*

DNS realizes UDP as its main transport layer protocol as it is much faster than TCP, which requires a 3 way handshake. TCP is generally used for transmitted large amount of information (> 512 bytes). Comparing Figure 11 and 12, this is true for the captured network traffic as **more UDP packets (27129)** are being sent over the network as compared to TCP packets (3947).

4. **HTTP**

The port number that is being reserved for HTTP is **80**.As HTML is a TCP connection and thus in order to analyze only the traffic on port 80, we apply a display filter to the entire captured traffic *(tcp.port* == 80*)*

tcp.port == 80

No.	Time	Source	Destination	Protocol	Length
43	0.000653	208.72.192.133	141.223.159.200	TCP	66
62	0.000982	112.169.44.132	141.223.114.1	HTTP	834
129	0.002070	12.161.242.20	141.223.169.130	TCP	1434
131	0.002084	12.161.242.20	141.223.169.130	TCP	1434
139	0.002266	27.101.11.29	141.223.137.76	TCP	1514
152	0.002513	27.101.11.29	141.223.137.76	TCP	1514
160	0.002623	27.101.11.29	141.223.137.76	TCP	1230
162	0.002634	211.115.209.190	141.223.118.85	TCP	60
183	0.002928	66.249.67.66	141.223.114.1	TCP	66
184	0.002929	211.115.209.190	141.223.118.85	TCP	60

Figure 14: *Filter By HTTP Port Numbers*

Statistics

Measurement	Captured	Displayed
Packets	4137680	280541 (6.8%)
Time span, s	59.088	59.086
Average pps	70026.3	4748.0
Average packet size, B	754	646
Bytes	3120951509	181146248 (5.8%)
Average bytes/s	52 M	3065 k
Average bits/s	422 M	24 M

Figure 15: *HTTP Capture Statistics (Statistics → Capture File Properties)*

HTTP is used by the World Wide Web and it defines how messages are formatted and transmitted by browser. As such, we expect HTTP requests to take up a rather significant portion of the captured traffic. From the capture statistics, we can tell that the total number of packets being captured for HTTP is **280541** and the total number of Bytes being captured is **181146248**. HHTP takes up close to **8%** of the entire network traffic. From Figure 13, we can tell that most of the HTTP traffic are actually between POSTECH's webpages (Identifiable by the IP prefix) and a client's computer.

2.6 Select two applications other than the aforementioned applications, and print out the number of packets and the bytes of the traffic which allocates well-known port number (TCP/UDP 1 - 1024)

The two other applications that I have selected are **DHCP (Dynamic Host Configuration Protocol)** and **SMTP (Simple Mail Transfer Protocol)**.

1. **DHCP**

   The port numbers that are being reserved for DHCP are **67** and **68**. DHCP uses two UDP connections for communication., port 67 for the server and port 68 for the client. The interaction between DHCP clients and servers enables a client to obtain its IP address and corresponding configuration information from a DHCP server. As DHCP utilises a UDP connection, in order to analyze only the traffic on ports 67 and 68, we apply a display filter to the entire captured traffic *(udp.port* == 67|| *udp.port* == 68*)*

udp.port == 67 || udp.port == 68

No.	Time	Source	Destination	Protocol	Length	I
73513	1.005326000	141.223.65.67	203.81.166.2	BOOTP	1066	L
73519	1.005398000	141.223.65.180	203.81.166.3	BOOTP	1066	L
139155	1.877314000	141.223.65.48	203.81.166.3	BOOTP	1066	L
181510	2.479444000	141.223.126.150	141.223.255.255	BOOTP	590	[
181512	2.479454000	141.223.126.150	141.223.255.255	BOOTP	590	[
181539	2.479693000	141.223.126.150	141.223.255.255	BOOTP	590	[

Figure 16: *Filter By DHCP Port Numbers*

Statistics

Measurement	Captured	Displayed
Packets	4137680	499 (0.0%)
Time span, s	59.088	56.687
Average pps	70026.3	8.8
Average packet size, B	754	655
Bytes	3120951509	326778 (0.0%)
Average bytes/s	52 M	5764
Average bits/s	422 M	46 k

Figure 17: *DHCP Capture Statistics (Stastics → Capture File Properties)*

From the capture statistics, we can tell that the total number of packets being captured for DHCP is **499** and the total number of Bytes being captured is **326778**. DHCP takes up close to **0%** of the entire network traffic. The average bytes of each packet being transmitted is about 655.

2. **SMTP**

The port number that is being reserved for SMTP is **25**. As SMTP utilises both UDP and TCP Connections and thus in order to analyze only the traffic on port 25, we apply a display filter to the entire captured traffic *(tcp.port == 25 —— udp.port == 25)*

tcp.port == 25 || udp.port == 25

No.	Time	Source	Destination	Protocol	Length
88	0.001319000	141.223.1.112	74.125.155.27	TCP	66
372	0.005808000	178.236.49.96	141.223.1.8	TCP	60
1693	0.023879000	141.223.114.1	77.184.0.21	TCP	66
2185	0.031949000	141.223.1.8	112.225.174.196	SMTP	103
2951	0.041228000	91.220.127.217	141.223.1.8	TCP	66
4981	0.067551000	141.223.1.8	209.85.214.175	TCP	66
4997	0.067759000	141.223.1.8	209.85.214.175	TCP	66

Figure 18: *Filter By SMTP Port Numbers*

Statistics

Measurement	Captured	Displayed	Marked
Packets	4137680	6386 (0.2%)	—
Time span, s	59.088	58.982	—
Average pps	70026.3	108.3	—
Average packet size, B	754	352	—
Bytes	3120951509	2250069 (0.1%)	0
Average bytes/s	52 M	38 k	—
Average bits/s	422 M	305 k	—

Figure 19: *SMTP Capture Statistics (Statistics → Capture File Properties)*

From the capture statistics above, we can tell that the total number of packets being captured for SMTP is **6386** and the total number of Bytes being captured is **2250069**. SMTP takes up close to **0%** of the entire network traffic. SMTP is generally used for for **sending e-mail messages between servers *(POSTECH and several other clients***

for these capture traffic)

Statistics

Measurement	Captured	Displayed
Packets	4137680	29 (0.0%)
Time span, s	59.088	56.687
Average pps	70026.3	0.5
Average packet size, B	754	1066
Bytes	3120951509	30914 (0.0%)
Average bytes/s	52 M	545
Average bits/s	422 M	4362

Figure 20: *SMTP (UDP) Capture Statistics*

Statistics

Measurement	Captured	Displayed
Packets	4137680	6357 (0.2%)
Time span, s	59.088	58.982
Average pps	70026.3	107.8
Average packet size, B	754	349
Bytes	3120951509	2219155 (0.1%)
Average bytes/s	52 M	37 k
Average bits/s	422 M	300 k

Figure 21: *SMTP (TCP) Capture Statistics*

Although SMTP realises both TCP and UDP, it makes more sense to use TCP over UDP. SMTP is a mail transport protocol, and in mail every single packet is important. If you lose several packets in the middle of the message the recipient might not even receive the message and if they do they might be missing key information. This makes TCP more appropriate because it ensures that every packet is delivered. Comparing Figure 20 and 21. this is true for the capture network traffic as **TCP Packets (6357) and bytes (2219155)** are being sent over the network as compared to **UDP Packets (29) and bytes (30914)**

2.7 Enumerate the average packet size, average packet inter-arrival time

1. **Average Packet Size**

Topic / Item	Count	Average	Min val	Max val	Rate (ms)	Percent	Burst rate	Burst start
Packet Lengths	4108237	759.03	60	1514	69.5280	100%	84.7000	33.585
0-19	0	-	-	-	0.0000	0.00%	-	-
20-39	0	-	-	-	0.0000	0.00%	-	-
40-79	882592	62.21	60	79	14.9370	21.48%	22.4900	2.355
80-159	115399	118.07	80	159	1.9530	2.81%	4.1100	39.135
160-319	65085	254.85	160	319	1.1015	1.58%	5.2900	44.907
320-639	1226380	419.86	320	639	20.7553	29.85%	22.6200	26.080
640-1279	401930	1056.08	640	1279	6.8023	9.78%	10.1400	44.490
1280-2559	1416851	1477.77	1280	1514	23.9788	34.49%	36.0700	18.385
2560-5119	0	-	-	-	0.0000	0.00%	-	-
5120 and greater	0	-	-	-	0.0000	0.00%	-	-

Figure 22: *Average Packet Size (Statistics → Packet Lengths)*

There were **4108237 packets** comprising roughly **3120.95MB** of traffic throughout the capture period. The **average packet size** for the these packets is **759.03 bytes**. Majority of the packets are between 1280 - 2559 bytes, and we can tell that higher the amount of byte being trasferred, the higher the transfer rate *(ms)* will be.

2. **Average Packet Inter-arrival Time**

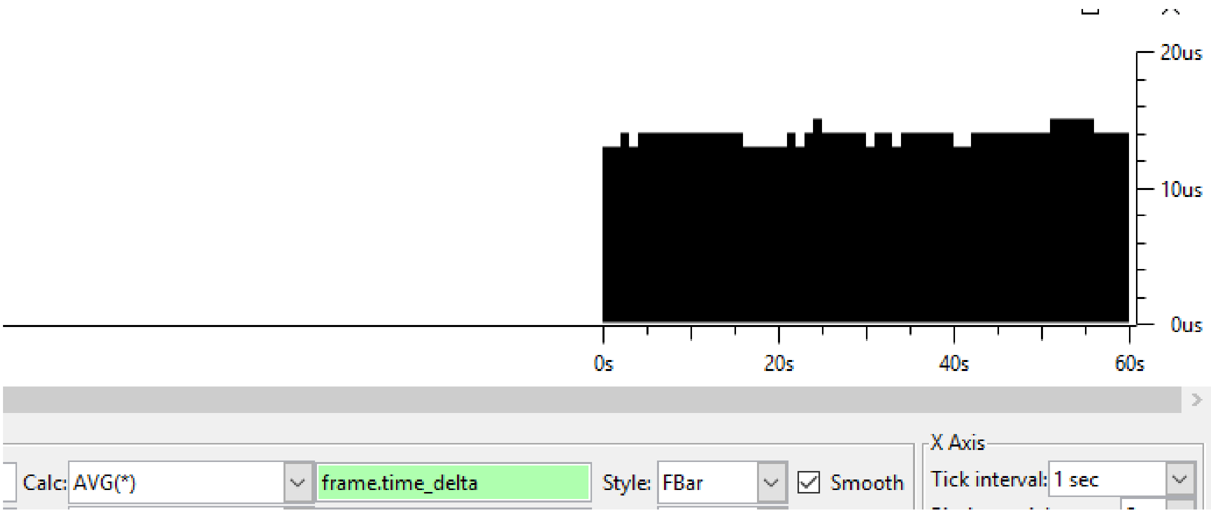

Figure 23: *Average Packet Inter-Arrival Time (Statistics → I/O Graph)*

From the figure above, we can tell that over the capture period of 60s, the average packet inter-arrival time is somewhere between **13 to 14μs**