

# **Komunikasi Data**



**Dede Rizky Kurniawan**

**09011382025157**

**SK4U-B (Kelas Palembang)**

**Jurusan Sistem Komputer**

**Fakultas Ilmu Komputer**

**Universitas Sriwijaya**

1. Menjalankan aplikasi dan menrecord jaringan dengan wireshark

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.43.84	142.251.10.94	TCP	55	52950 → 443 [ACK] Seq=1 Ack=1 Win=254 Len=1 [TCP segment of a reassembled PDU]
2	0.078865	142.251.10.94	192.168.43.84	TCP	66	443 → 52950 [ACK] Seq=1 Ack=2 Win=355 Len=0 SLE=1 SRE=2
3	1.739753	192.168.43.84	140.213.63.45	QUIC	1292	Initial, DCID=3ca6483874d56de3, PKN: 1, PADDING, PING, PING, PADDING, CRYPTO, CRYPTO, PADDIN
4	1.743338	192.168.43.84	140.213.63.45	QUIC	117	0-RTT, DCID=3ca6483874d56de3
5	1.744374	192.168.43.84	140.213.63.45	QUIC	1288	0-RTT, DCID=3ca6483874d56de3
6	1.744623	192.168.43.84	140.213.63.45	QUIC	383	0-RTT, DCID=3ca6483874d56de3
7	1.867205	140.213.63.45	192.168.43.84	QUIC	1292	Handshake, SCID=3ca6483874d56de3
8	1.867721	140.213.63.45	192.168.43.84	QUIC	1292	Handshake, SCID=3ca6483874d56de3
9	1.867721	140.213.63.45	192.168.43.84	QUIC	1292	Handshake, SCID=3ca6483874d56de3
10	1.867721	140.213.63.45	192.168.43.84	QUIC	1292	Handshake, SCID=3ca6483874d56de3
11	1.868058	192.168.43.84	140.213.63.45	QUIC	83	Handshake, DCID=3ca6483874d56de3
12	1.868348	192.168.43.84	140.213.63.45	QUIC	83	Handshake, DCID=3ca6483874d56de3
13	1.869078	140.213.63.45	192.168.43.84	QUIC	431	Protected Payload (KP0)
14	1.869923	192.168.43.84	140.213.63.45	QUIC	130	Handshake, DCID=3ca6483874d56de3
15	1.870085	192.168.43.84	140.213.63.45	QUIC	1292	Protected Payload (KP0), DCID=3ca6483874d56de3
16	1.870175	192.168.43.84	140.213.63.45	QUIC	407	Protected Payload (KP0), DCID=3ca6483874d56de3
17	1.968795	140.213.63.45	192.168.43.84	QUIC	572	Protected Payload (KP0)
18	1.968795	140.213.63.45	192.168.43.84	QUIC	130	Protected Payload (KP0)
19	1.969585	192.168.43.84	140.213.63.45	QUIC	75	Protected Payload (KP0), DCID=3ca6483874d56de3
20	1.975380	140.213.63.45	192.168.43.84	QUIC	67	Protected Payload (KP0)
21	1.976302	140.213.63.45	192.168.43.84	QUIC	1292	Protected Payload (KP0)
22	1.976302	140.213.63.45	192.168.43.84	QUIC	1292	Protected Payload (KP0)
23	1.977120	192.168.43.84	140.213.63.45	QUIC	77	Protected Payload (KP0), DCID=3ca6483874d56de3
24	1.980117	140.213.63.45	192.168.43.84	QUIC	1292	Protected Payload (KP0)

Pada proses ini saya menjalankan wireshark kurang lebih 3 menit sambil menonton video dari youtube yang tersambung dengan jaringan internet. Dapat dilihat pada gambar dibawah ini bahwa wireshark merekam jaringan server yang tersambung pada video youtube yang saya jalankan.

No.	Time	Source	Destination	Protocol	Length	Info
6576	97.655141	192.168.43.84	192.168.43.1	DNS	81	Standard query 0x6f3f A config.edge.skype.com
6577	97.729088	192.168.43.1	192.168.43.84	DNS	241	Standard query response 0x6f3f A config.edge.skype.com CNAME config.edge.skype.com.trafficmanager.net CNAME 1-0007.config.skyp...
7145	110.614303	192.168.43.84	192.168.43.1	DNS	71	Standard query 0xa669 A youtube.com
7148	110.677371	192.168.43.1	192.168.43.84	DNS	135	Standard query response 0xa669 A youtube.com A 142.250.4.93 A 142.250.4.136 A 142.250.4.91 A 142.250.4.190
7169	117.957497	192.168.43.84	192.168.43.1	DNS	71	Standard query 0x35fe A api.msn.com
7170	118.016285	192.168.43.1	192.168.43.84	DNS	186	Standard query response 0x35fe A api.msn.com CNAME api.msn-com.a-0003.a-msedge.net CNAME icePrime.a-0003.dc-msedge.net CNAME a...
7222	119.875554	192.168.43.84	192.168.43.1	DNS	71	Standard query 0x05b3 A arc.msn.com
7223	119.954846	192.168.43.84	192.168.43.1	DNS	71	Standard query 0x05b3 A arc.msn.com
7224	120.066863	192.168.43.1	192.168.43.84	DNS	182	Standard query response 0x05b3 A arc.msn.com CNAME arc.trafficmanager.net CNAME iris-de-ppe-azsc-eas.eastasia.cloudapp.azure.c...
8086	155.684110	192.168.43.84	192.168.43.1	DNS	71	Standard query 0xd772 A arc.msn.com
8087	155.746729	192.168.43.1	192.168.43.84	DNS	182	Standard query response 0xd772 A arc.msn.com CNAME arc.trafficmanager.net CNAME iris-de-ppe-azsc-eas.eastasia.cloudapp.azure.c...
8120	156.420552	192.168.43.84	192.168.43.1	DNS	86	Standard query 0xe27b A ris.api.iris.microsoft.com
8121	156.477341	192.168.43.1	192.168.43.84	DNS	205	Standard query response 0xe27b A ris.api.iris.microsoft.com CNAME ris-prod.trafficmanager.net CNAME asf-ris-prod-easia-azsc.ea...

Saya menjalankan video youtube sehingga akan mendapatkan informasi mengenai server pada hasil record seperti gambar diatas, youtube.com.

Selain itu saya juga melakukan perhitungan throughput, packet loss, jitter, dan delay yang terjadi pada saat proses wireshark tersebut sedang berlangsung. Untuk hasil dan gambar untuk pencarian throughput, packet loss, dan jitter dapat dilihat pada penjelasan selanjutnya.

Mencari Throughput, Packet Loss, Jitter, dan Delay

Pertama buka hasil statistic dari record pada wireshark tadi seperti gambar dibawah ini.

Statistics			
Measurement	Captured	Displayed	Marked
Packets	9193	6 (0.1%)	—
Time span, s	213.895	122.471	—
Average pps	43.0	0.0	—
Average packet size, B	1048	54	—
Bytes	9636289	324 (0.0%)	0
Average bytes/s	45 k	2	—
Average bits/s	360 k	21	—

Diatas adalah hasil statistic dari record wireshark yang saya lakukan, terdapat 9193 Packets yang diproses dengan rentang waktu 213,895 second atau 3,5 menit

Untuk mencari Throughput bisa dilakukan dengan jumlah bytes data dibagi menjadi time span kemudian hasilnya yang bernilai bytes dikalikan dengan 8 bit. Sehingga didapat Throughput sekitar 360k. Selanjutnya

adalah mencari Packet Loss, untuk Packet Loss sendiri kita harus mencari dengan tcp.analysis.segment untuk packet lost seperti gambar dibawah berikut.

tcp.analysis.ack_lost_segment						
No.	Time	Source	Destination	Protocol	Length	Info
1811	37.312490	192.168.43.84	117.18.237.29	TCP	54	[TCP ACKed unseen segment] 52966 → 80 [ACK] Seq=1 Ack=2 Win=255 Len=0
1813	41.207512	192.168.43.84	117.18.232.200	TCP	54	[TCP ACKed unseen segment] 52972 → 443 [ACK] Seq=1 Ack=2 Win=1019 Len=0
6612	98.623235	192.168.43.84	117.18.237.29	TCP	54	[TCP Dup ACK 1811#1] [TCP ACKed unseen segment] 52966 → 80 [ACK] Seq=1 Ack=2 Win=255 Len=0
6651	102.643681	192.168.43.84	117.18.232.200	TCP	54	[TCP Dup ACK 1813#1] [TCP ACKed unseen segment] 52972 → 443 [ACK] Seq=1 Ack=2 Win=1019 Len=0
7147	110.618537	192.168.43.84	117.18.237.29	TCP	54	[TCP ACKed unseen segment] 52966 → 80 [ACK] Seq=1 Ack=3 Win=255 Len=0
8161	159.783483	192.168.43.84	117.18.232.200	TCP	54	[TCP ACKed unseen segment] 52972 → 443 [ACK] Seq=1 Ack=66 Win=1019 Len=0

Pada gambar tersebut ada 6 packet loss dari 9193 paket, sehingga untuk mencari persentasenya adalah (6\*100%) /9193 sehingga didapat Packet Loss nya lebih kurang 0,1%.

Untuk Jitter dan Delay dapat dicari dengan mengexport hasil record wireshark menjadi CSV atau file dengan format excel seperti gambar dibawah ini

	A	B	C	D	E	F	G	H	I	J	K
1	No.	Time	Time2	Time1	Delay	Delay2	Delay1	Jitter			
2	1	0	0.078865		0	0.078865	1.660888	-1.58202	3.242911	Delay	
3	2	0.078865	1.739753	0.078865	1.660888	0.003585	1.657303	-1.65372		total delay	213.8952
4	3	1.739753	1.743338	1.739753	0.003585	0.001036	0.002549	-0.00151		rata-rata delay	0.02327
5	4	1.743338	1.744374	1.743338	0.001036	0.000249	0.000787	-0.00054		Jitter	
6	5	1.744374	1.744623	1.744374	0.000249	0.122582	-0.12233	0.244915		total jiter	213.8001
7	6	1.744623	1.867205	1.744623	0.122582	0.000516	0.122066	-0.12155		rata-rata Jitter	0.023259
8	7	1.867205	1.867721	1.867205	0.000516	0	0.000516	-0.00052			
9	8	1.867721	1.867721	1.867721	0	0	0	0			
10	9	1.867721	1.867721	1.867721	0	0.000337	-0.00034	0.000674			
11	10	1.867721	1.868058	1.867721	0.000337	0.00029	4.7E-05	0.000243			
12	11	1.868058	1.868348	1.868058	0.00029	0.00073	-0.00044	0.00117			
13	12	1.868348	1.869078	1.868348	0.00073	0.000845	-0.00011	0.00096			
14	13	1.869078	1.869923	1.869078	0.000845	0.000162	0.000683	-0.00052			
15	14	1.869923	1.870085	1.869923	0.000162	9E-05	7.2E-05	1.8E-05			
16	15	1.870085	1.870175	1.870085	9E-05	0.09862	-0.09853	0.19715			
17	16	1.870175	1.968795	1.870175	0.09862	0	0.09862	-0.09862			
18	17	1.968795	1.968795	1.968795	0	0.00079	-0.00079	0.00158			
19	18	1.968795	1.969585	1.968795	0.00079	0.005795	-0.00501	0.0108			
20	19	1.969585	1.97538	1.969585	0.005795	0.000922	0.004873	-0.00395			
21	20	1.97538	1.976302	1.97538	0.000922	0	0.000922	-0.00092			
22	21	1.976302	1.976302	1.976302	0	0.000818	-0.00082	0.001636			
23	22	1.976302	1.97712	1.976302	0.000818	0.002997	-0.00218	0.005176			
24	23	1.97712	1.980117	1.97712	0.002997	0.000487	0.00251	-0.00202			
25	24	1.980117	1.980604	1.980117	0.000487	0.003889	-0.0034	0.007291			
26	25	1.980604	1.984493	1.980604	0.003889	0.002509	0.00138	0.001129			
27	26	1.984493	1.987002	1.984493	0.002509	0.000676	0.001833	-0.00116			
28	27	1.987002	1.987678	1.987002	0.000676	0.000367	0.000309	5.8E-05			
29	28	1.987678	1.988045	1.987678	0.000367	0.00244	-0.00207	0.004513			
30	29	1.988045	1.990485	1.988045	0.00244	0.000382	0.002058	-0.00168			

Mencari Delay sendiri adalah dengan pengurangan, yaitu waktu pertama dibagi waktu kedua kemudian ditotalkan, untuk rata-ratanya dibagi total Packet dikurangi 1 jadi 9192. Untuk jitter sendiri juga sama namun yang dikurangkan adalah delay kedua dikurangi delay pertama kemudian ditotalkan, untuk rata-rata jitter dibagi dengan total Packet lalu dikurangi 1 jadi 9192.

Proses komunikasi data yang terlihat adalah dimana setiap jaringan memiliki ip source dan destinationnya. Ip yang sama kebanyakan adalah ip dari komputer atau device kita sendiri seperti pada record wireshark saya ip saya adalah 192.168.43.84 , sementara untuk ip yang berbeda beda adalah ip dari server atau web yang saya kunjungi atau terjadi dalam proses penggunaan aplikasi yang menggunakan jaringan didalamnya. Selain itu,

kesamaan ip tidak menjamin proses mengirim file yang sama aka nada port yang menjadi penentu arah transport dari data data yang diproses seperti pada gambar dibawah.

```
Source Port: 53
Destination Port: 58625
Length: 101
Checksum: 0x106d [unverified]
[Checksum Status: Unverified]
[Stream index: 10]
> [Timestamps]
  UDP payload (93 bytes)
> Domain Name System (response)
```

Ip yang sama tidak selamanya mengirim data yang sama, port menjadi pembeda dalam proses transportasi data.

Link Repository Github <https://github.com/dede1409/Network-Traffic-Analysis>