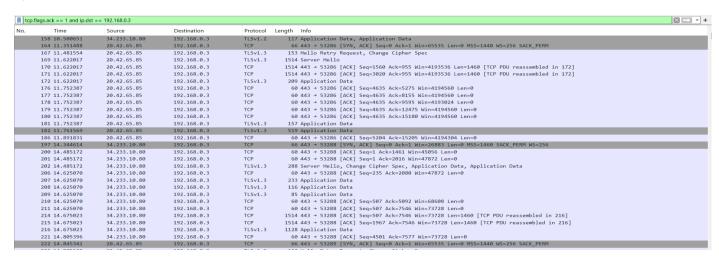
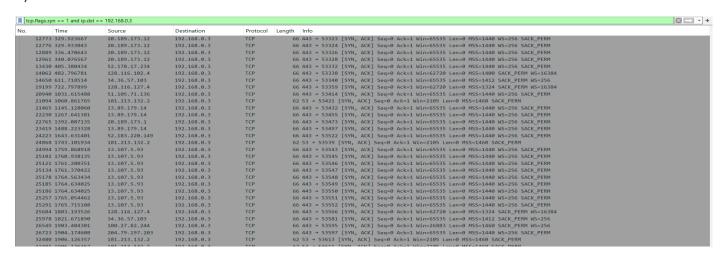
## Arthur de Sá Braz de Matos

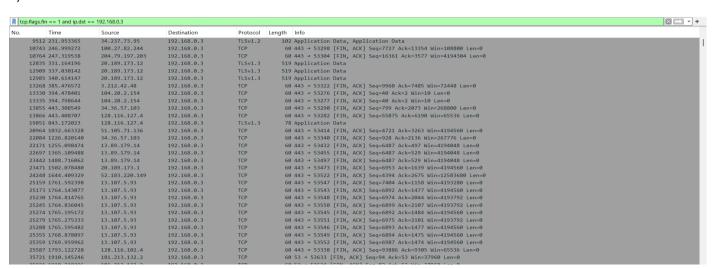
## a) Transferência Confiável



#### b) Início de Conexão



### c) Término de Conexão



#### d) Controle de Fluxo

p.w	vindow_size < 65535 and	d ip.dst == 192.168.0.3			le de la company	$\times \Rightarrow$
	Time	Source	Destination	Protocol	Length Info	
	23 2.642429	128.116.127.4	192.168.0.3	TLSv1.2	93 Application Data	
	24 2.642429	128.116.127.4	192.168.0.3	TLSv1.2	78 Application Data	
	25 2.642429	128.116.127.4	192.168.0.3	TCP	60 443 → 52344 [FIN, ACK] Seq=64 Ack=1 Win=4 Len=0	
	26 2.751966	128.116.127.4	192.168.0.3	TCP	66 443 → 53282 [SYN, ACK] Seq=0 Ack=1 Win=62720 Len=0 MSS=1324 SACK_PERM WS=16384	
			192.168.0.3		60 [TCP Retransmission] 443 → 52344 [FIN, ACK] Seq=64 Ack=1 Win=4 Len=0	
	95 3.783660	20.150.179.231	192.168.0.3	TLSv1.2	85 Encrypted Alert	
					117 [TCP Retransmission] 443 → 52344 [FIN, PSH, ACK] Seq=1 Ack=1 Win=4 Len=63	
	138 5.270888	128.116.127.4	192.168.0.3	TCP	117 [TCP Retransmission] 443 → 52344 [FIN, PSH, ACK] Seq=1 Ack=1 Win=4 Len=63	
	139 5.375317	34.36.57.103	192.168.0.3	TLSv1.2	127 Application Data	
	150 7.958067	34.237.73.95	192.168.0.3	TCP	60 443 → 53260 [ACK] Seq=1 Ack=241 Win=181 Len=0	
	151 7.987942	34.237.73.95	192.168.0.3	TLSv1.2	317 Application Data	
	158 10.500651	34.233.10.80	192.168.0.3	TLSv1.2	117 Application Data, Application Data	
	197 14.344614	34.233.10.80	192.168.0.3	TCP	66 443 → 53288 [SYN, ACK] Seq=0 Ack=1 Win=26883 Len=0 MSS=1460 SACK_PERM WS=256	
	200 14.485172	34.233.10.80	192.168.0.3	TCP	60 443 → 53288 [ACK] Seq=1 Ack=1461 Win=45056 Len=0	
	201 14.485172	34.233.10.80	192.168.0.3	TCP	60 443 → 53288 [ACK] Seq=1 Ack=2016 Win=47872 Len=0	
	202 14.485172	34.233.10.80	192.168.0.3	TLSv1.3	288 Server Hello, Change Cipher Spec, Application Data, Application Data	
	206 14.625070	34.233.10.80	192.168.0.3	TCP	60 443 → 53288 [ACK] Seq=235 Ack=2080 Win=47872 Len=0	
	207 14.625070	34.233.10.80	192.168.0.3	TLSv1.3	233 Application Data	
	208 14.625070	34.233.10.80	192.168.0.3	TLSv1.3	116 Application Data	
	209 14.625070	34.233.10.80	192.168.0.3	TLSv1.3	85 Application Data	
				TCP		
	245 19.719648	34.36.57.103		TCP	127 [TCP Retransmission] 443 → 52492 [PSH, ACK] Seq=1 Ack=1 Win=1047 Len=73	
	246 21.202119	34.237.73.95	192.168.0.3	TLSv1.2	78 Application Data	
	253 22.782637				78 [TCP Retransmission] 443 → 52511 [PSH, ACK] Seq=1 Ack=1 Win=375 Len=24	
	260 23.913816	34.237.73.95	192.168.0.3	TCP	60 443 → 53260 [ACK] Seq=264 Ack=481 Win=186 Len=0	
	261 23.913816	34.237.73.95	192.168.0.3	TLSv1.2	317 Application Data	
	265 24.384756	34.237.73.95	192.168.0.3	TCP	78 [TCP Retransmission] 443 → 52511 [PSH, ACK] Seq=1 Ack=1 Win=375 Len=24	

## a) Transferência Confiável

- 1. O emissor envia um segmento de dados com um número de sequência
- 2. O receptor responde com um ACK contendo o próximo número de sequência esperado
- 3. Se o emissor não receber um ACK dentro de um determinado tempo, ele retransmite o segmento
- 4. O receptor descarta segmentos duplicados usando os números de sequência

Esta mecânica garante que nenhum dado seja perdido e que todos os bytes cheguem na ordem correta.

# b) Início de Conexão

- Pacote SYN: O cliente inicia a conexão enviando um pacote com a flag SYN ativada e um número de sequência inicial (ISN)
- 2. Pacote SYN-ACK: O servidor responde com um pacote que tem as flags SYN e ACK ativadas, seu próprio ISN e um número de confirmação igual ao ISN do cliente + 1
- 3. Pacote ACK: O cliente completa o handshake enviando um pacote com a flag ACK ativada e um número de confirmação igual ao ISN do servidor + 1

# c) Término de Conexão

- 1. Pacote FIN: O lado que deseja encerrar a conexão envia um pacote com a flag FIN ativada
- 2. Pacote ACK: O outro lado confirma o recebimento do FIN com um ACK
- 3. Pacote FIN: O segundo lado também envia um FIN quando está pronto para encerrar
- 4. Pacote ACK: O lado que iniciou o encerramento confirma o FIN do segundo lado

### d) Controle de Fluxo

- 1. O receptor anuncia quanto espaço tem disponível em seu buffer (janela de recepção)
- 2. O emissor limita o número de bytes que envia sem confirmação a este valor
- 3. Quando o buffer do receptor começa a encher, ele reduz o tamanho da janela anunciada

4. O emissor deve respeitar este limite para evitar sobrecarga do receptor

Este mecanismo previne que um emissor rápido sobrecarregue um receptor mais lento, mantendo a eficiência da transferência de dados.