Análise de Segmentos TCP com Wireshark

Detalhamento Técnico de Segmentos TCP

Aluno: Vítor Carvalho Marx Lima
Matrícula: 11821ECP015
Data: 29 de Março de 2024

OBS: As questões foram respondidas a partir do documento fornecido pelo próprio livro de estudos (capturas realizadas pelo próprio autor)

1. Endereço IP e Porta de número TCP usada pelo cliente (Source):

1. Endereço do Computador Cliente: 192.168.1.102
2. Número da Porta TCP Usada pelo Cliente: 1161
3. O computador cliente está usando o endereço IP 192.168.1.102 e o número da porta 1161 para transferir o arquivo para gaia.cs.umass.edu.

2. Endereço IP de gaia.cs.umass.edu e o Número da Porta:

1. Endereço IP de gaia.cs.umass.edu: 128.119.245.12
2. Número da Porta em Que Está Enviando e Recebendo Segmentos TCP: 80
3. O endereço IP de gaia.cs.umass.edu é 128.119.245.12, e está enviando e recebendo segmentos TCP na porta número 80, que é a porta padrão para tráfego HTTP.

3. Estou utilizando os arquivos fornecidos pelo próprio livro para a realização dos laboratórios.

4. Número de Sequência do Segmento TCP SYN: O número de sequência utilizado para iniciar a conexão TCP entre o computador cliente (192.168.1.102) e gaia.cs.umass.edu (128.119.245.12) é 0. Este segmento é identificado como um segmento SYN porque tem o flag SYN definido, o que indica a tentativa de iniciar uma conexão TCP.

5. Número de Sequência do Segmento SYNACK:

1. O segmento SYNACK enviado por gaia.cs.umass.edu em resposta ao computador cliente possui um número de sequência de 0.
2. O valor do campo de Reconhecimento no segmento SYNACK é 1.
3. Gaia.cs.umass.edu determinou esse valor como sendo 1 a mais que o número de sequência do segmento SYN enviado pelo cliente, conforme o comportamento padrão do protocolo TCP, que reconhece o recebimento do segmento SYN incrementando seu número de sequência em 1.

4. Este segmento é identificado como um segmento SYNACK porque possui tanto o flag SYN quanto o ACK definidos, indicado pela notação [SYN, ACK] nas informações do pacote.

6. Número de Sequência do Segmento TCP Contendo o Comando HTTP POST: O número de sequência do segmento TCP que contém o comando HTTP POST é 1. Isso pode ser identificado inspecionando o conteúdo do pacote em busca de uma string "POST" dentro do seu campo de dados, conforme mostrado nas informações do pacote com a notação [PSH, ACK] e o exame detalhado do payload do pacote no documento.

7. Primeiro Segmento na Conexão TCP (Contendo HTTP POST):

OBS: Os prints dos pacotes utilizados estão logo abaixo da questão 7.

O primeiro segmento na conexão TCP que contém o comando HTTP POST tem o número de sequência 1.

1. tempo do pacote 1=0.026477
2. tempo do pacote 2=0.041737
3. tempo do pacote 3=0.053937
4. tempo do pacote 4=0.054026
5. tempo do pacote 5=0.054690
6. tempo do pacote 6=0.077294

Agora os RTTs:

1. RTT do segmento 1: 0.026477 segundos
2. RTT do segmento 2: 0.01526 segundos
3. RTT do segmento 3: 0.0122 segundos
4. RTT do segmento 4: 0.000089 segundos
5. RTT do segmento 5: 0.000664 segundos
6. RTT do segmento 6: 0.022604 segundos

Com base nesses RTTs, o valor do EstimatedRTT, calculado como a média dos RTTs observados, é aproximadamente 0.012882 segundos.

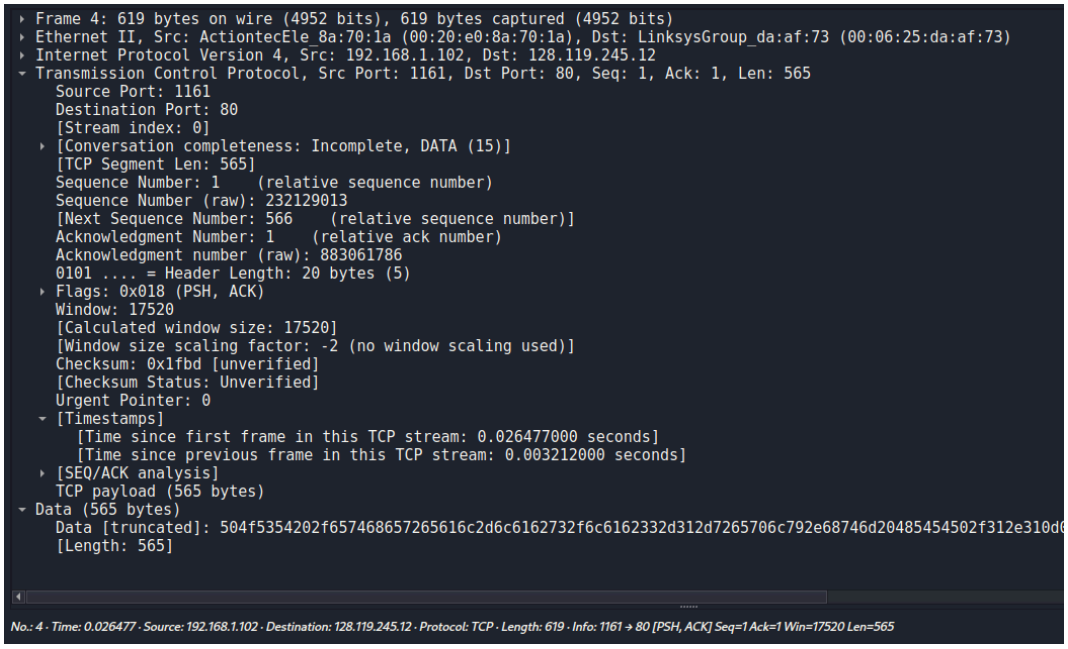


Figura 1: Primeiro pacote

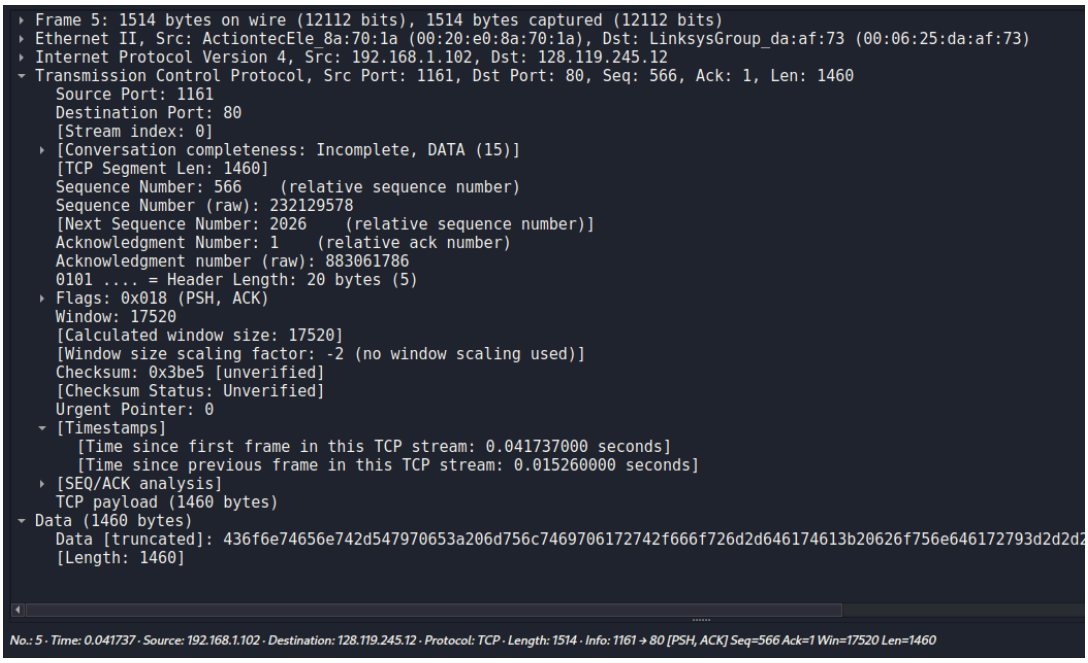


Figura 2: Segundo Pacote

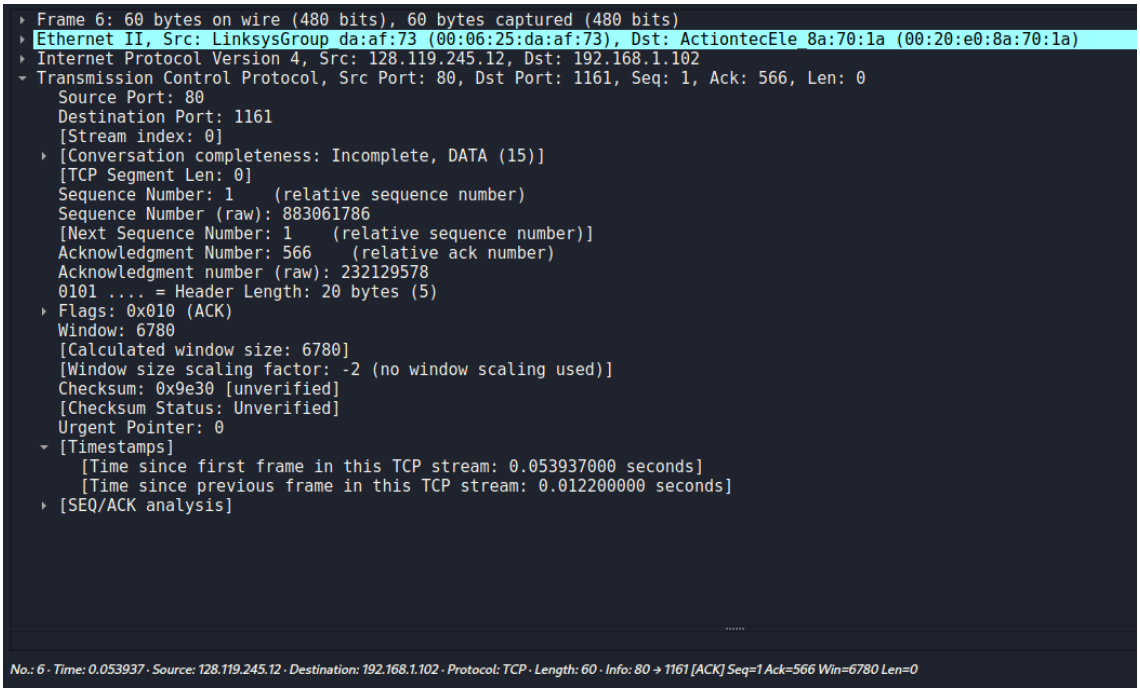


Figura 3: Terceiro Pacote

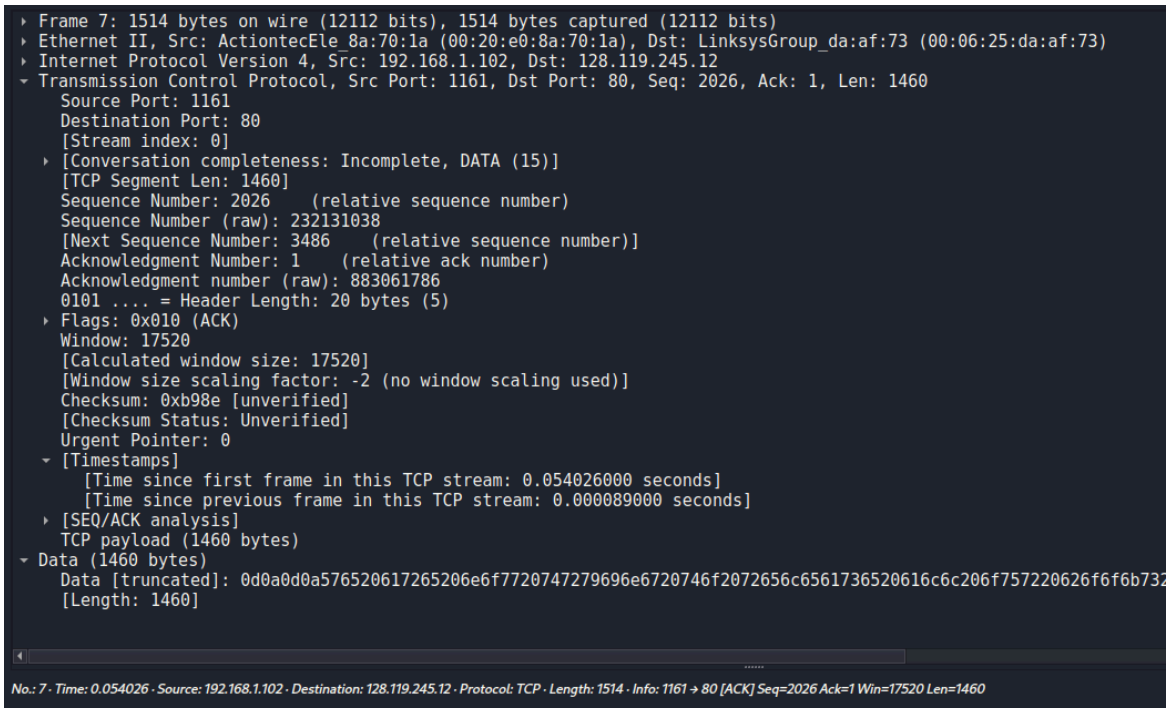


Figura 4: Quarto Pacote

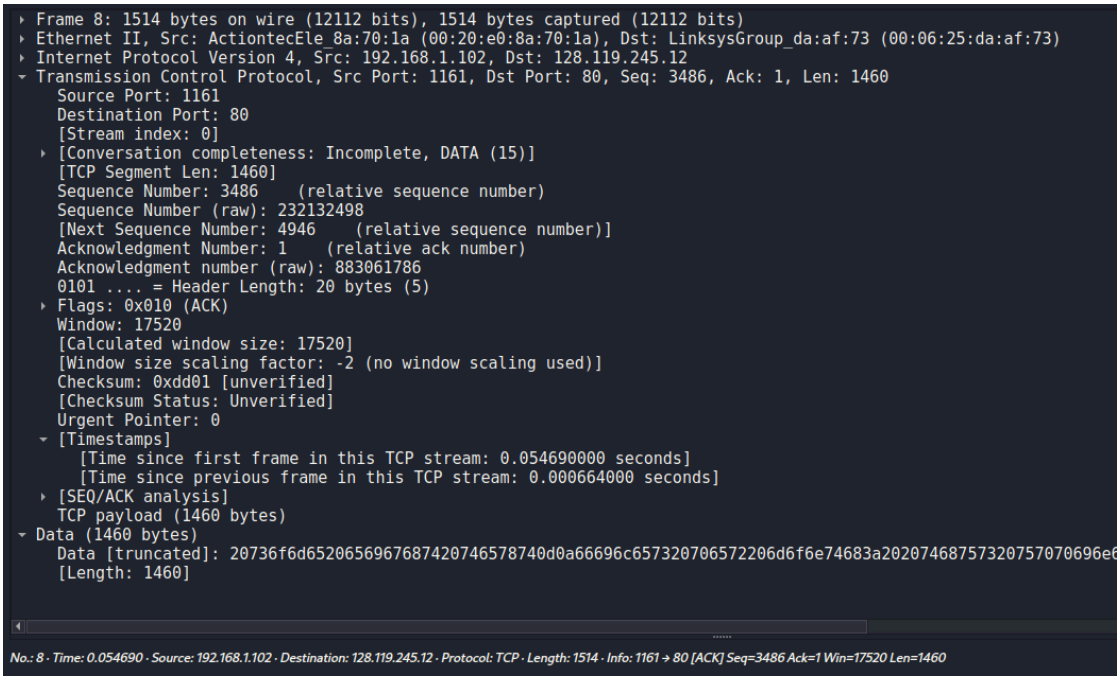


Figura 5: Quinto Pacote

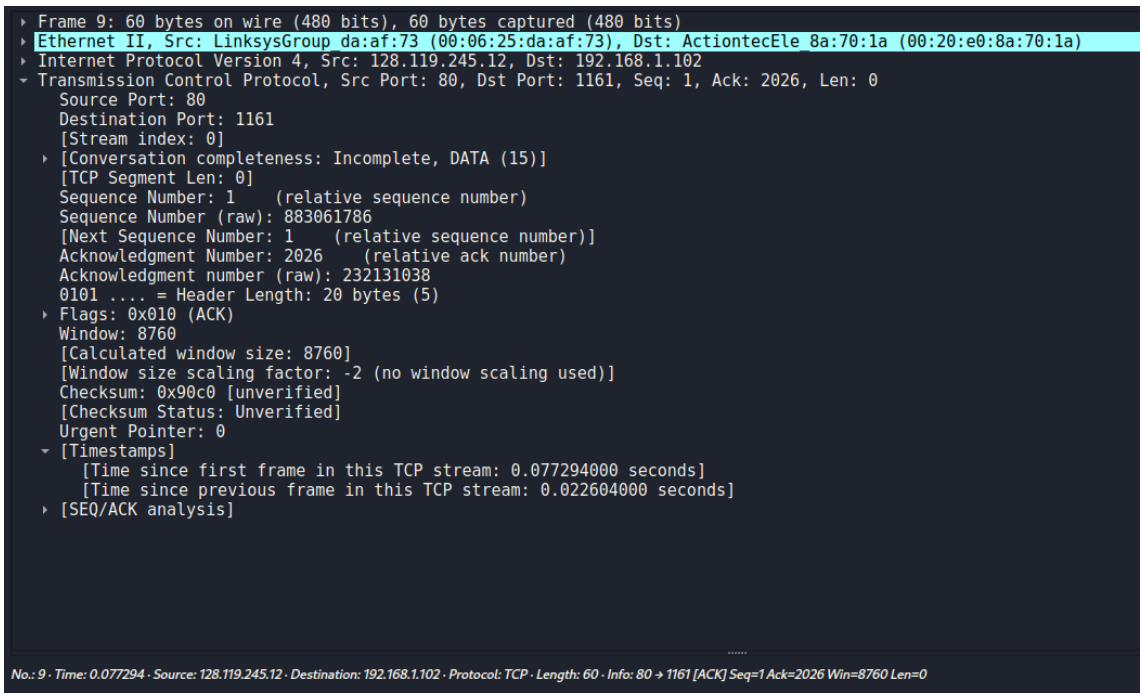
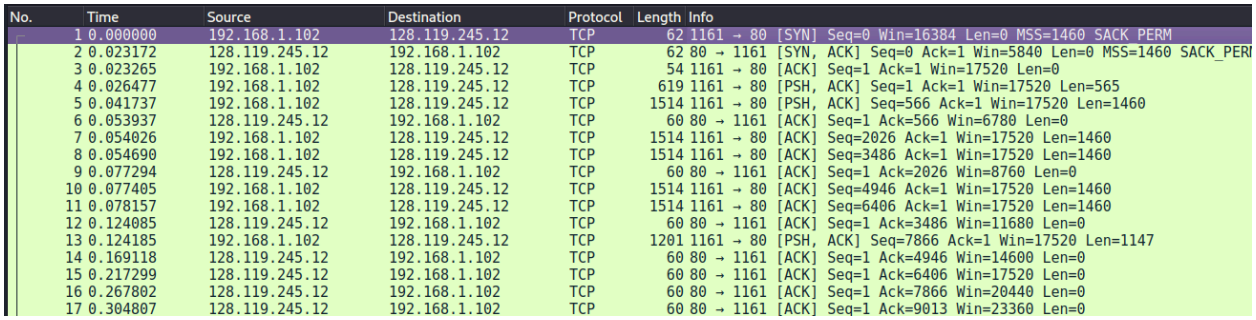


Figura 6: Sexto Pacote



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM
2	0.023172	128.119.245.12	192.168.1.102	TCP	62	80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM
3	0.023265	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	0.026477	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565
5	0.041737	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460
6	0.053937	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	0.054026	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460
8	0.054690	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460
9	0.077294	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	0.077405	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460
11	0.078157	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460
12	0.124085	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	0.124185	192.168.1.102	128.119.245.12	TCP	1201	1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147
14	0.169118	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15	0.217299	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
16	0.267802	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
17	0.304807	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0

Figura 7: Imagem geral das capturas TCP

8. Comprimento de cada um dos 6 pacotes:

1. 1° Pacote - 565 Bytes
2. 2° Pacote - 1460 Bytes
3. 3° Pacote - 60 Bytes
4. 4° Pacote - 1460 Bytes
5. 5° Pacote - 1460 Bytes
6. 6° Pacote - 60 Bytes

9. Mínimo de Buffer disponível:

O menor espaço disponível de buffer anunciado pelo receptor (window size) é o que possui o menor valor nos segmentos TCP. Entre os pacotes fornecidos, o menor valor anunciado é de 6780 bytes (encontrado no Frame 3).

10. Existem segmentos retransmitidos? O que foi avaliado no Trace para definir isto?

Verificamos por números de sequência duplicados ou por ACKs que parecem estar reconhecendo dados já recebidos anteriormente. Nos 6 pacotes analisados, não existe retransmissão, já que cada segmento possui um número de sequência único.

11. O quanto o receptor tipicamente reconhece em um ACK? Consegue identificar casos onde o receptor está reconhecendo (ACKing) a cada próximo segmento recebido?

Um receptor TCP geralmente reconhece todos os dados recebidos até um determinado ponto sequencial. Por exemplo, se o receptor envia um ACK com número de sequência X, está confirmando que recebeu todos os bytes até o byte X-1. Nas imagens acima, os ACKs estão reconhecendo os dados sequencialmente (1, 566, 2026, 3486, 6406), o que indica que cada ACK está reconhecendo os dados enviados em um único segmento anterior.

12. Qual o Throughput da conexão TCP? Como esse valor foi calculado?

O throughput é calculado com base no total de dados transferidos dividido pelo tempo total da transferência.

O tempo total da transferência pode ser aproximado como o tempo do último pacote menos o tempo do primeiro pacote de dados, que é aproximadamente (0.077294 - 0.026477) = 0.050817 segundos considerando os 6 pacotes analisados.

A quantidade total de dados pode ser somada a partir dos tamanhos dos segmentos de dados:

(565+3*1460+2*60) = 5065 Bytes.

Logo: Throughput= (Total de Bytes)/(Tempo Final−Tempo Inicial) = 5065/0.050817

Throughput = 99671.37 Bytes/Segundo (Bps).

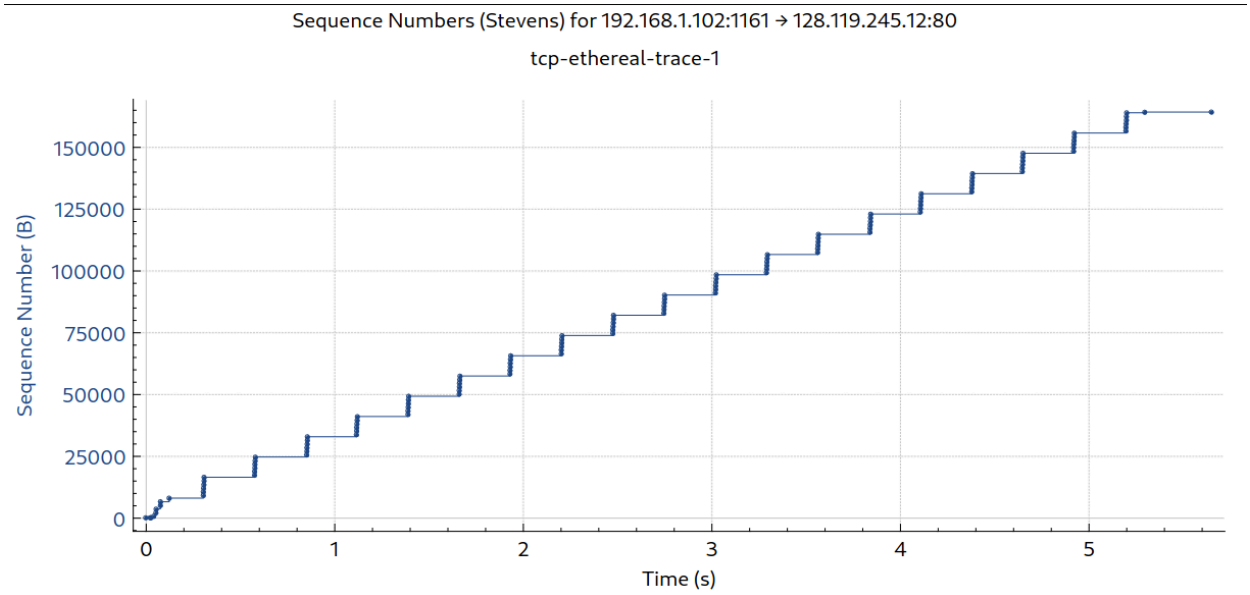


Imagem 8: Sequence Numbers (Stevens) para a questão 13.

13. É possível identificar o intervalo onde ocorre o "slowstart"? E onde o período de anti-congestão (congestion avoidance) ocorre?

A partir do gráfico de Número de Sequência versus Tempo (conhecido como gráfico de Stevens) para segmentos enviados do cliente para o servidor, podemos tentar identificar as fases do início lento (slow start) e da evitação de congestionamento (congestion avoidance) do TCP.

No TCP, a fase de início lento começa com o início da transferência de dados, onde o tamanho da janela de congestionamento (cwnd) começa em um ou alguns segmentos e dobra a cada Round-Trip Time (RTT) até que ocorra perda de pacotes (indicando congestionamento) ou até atingir o limiar de início lento (ssthresh). Após esse ponto, se o TCP entra na fase de evitação de congestionamento,

o crescimento da janela de congestionamento se torna mais conservador, aumentando linearmente em vez de exponencialmente.

No gráfico, é possível verificar o **período de "slowstart"** começando no 0 e indo até aproximadamente 180ms, após isso, se inicia o período de **"congestion avoidance"**.

14. Responda cada uma das questões acima com os dados do Trace coletado:

As questões foram respondidas utilizando os Traces fornecidos pelo próprio autor do livro, como comentado na observação no início deste arquivo.