Pontifícia Universidade Católica do Rio Grande do Sul Escola Politécnica Laboratório de Redes de Computadores

Trabalho 1 Laboratório de Redes de Computadores

1) Execute o Wireshark para monitorar o tráfego UDP gerado pelo programa. Identifique os pacotes UDP que estão sendo enviados para cada um dos servidores. Quais portas de origem e destino estão sendo utilizadas pelos pacotes?

Porta de Origem: Normalmente, quando o cliente envia pacotes UDP, ele usa uma porta dinâmica, que será exibida na análise.

Porta de Destino: Será a porta em que o servidor está escutando, que no seu caso deve ser a 65000, conforme configurado no código.

User Datagram Protocol, Src Port: 33953, Dst Port: 65000
 Source Port: 33953

Destination Port: 65000

 User Datagram Protocol, Src Port: 36008, Dst Port: 65000 Source Port: 36008

Destination Port: 65000

2) Há diferença, em termos de volume de tráfego na rede, entre a aplicação com socket TCP e a aplicação com socket UDP?

Para a mesma quantidade de dados enviados, uma aplicação baseada em **UDP** vai gerar menos tráfego total do que uma baseada em **TCP**, devido ao menor overhead e à ausência de pacotes de controle/retransmissão. No entanto, isso vem ao custo de confiabilidade, pois o UDP não garante que todos os pacotes chegarão ao destino. para a mesma quantidade de solicitações, pacotes do UDP:

Г	29 11.447683524	10.32.143.16	10.32.143.11	UDP	164 36008 → 65000 Len=122			
	30 11.447997880	10.32.143.11	10.32.143.24	UDP	164 65000 → 33953 Len=122			
-	67 22.158458759	10.32.143.24	10.32.143.11	UDP	160 33953 → 65000 Len=118			
L	68 22.158768627	10.32.143.11	10.32.143.16	UDP	160 65000 → 36008 Len=118			
pacotes do TCP:								

304 80.338197732	10.32.143.16	10.32.143.11	TCP	159 38508 → 65001 [PSH, ACK] Seq=1 Ack=1 Win=502 Len=93 TSval=2586153122 TSecr=2208556170	
305 80.338240033	10.32.143.11	10.32.143.16	TCP	66 65001 → 38508 [ACK] Seq=1 Ack=94 Win=510 Len=0 TSval=2208738498 TSecr=2586153122	
306 80.338522812	10.32.143.11	10.32.143.24	TCP	159 65001 - 40704 [PSH, ACK] Seq=1 Ack=1 Win=509 Len=93 TSval=3826878682 TSecr=160791319	
307 80.338920906	10.32.143.24	10.32.143.11	TCP	66 40704 → 65001 [ACK] Seq=1 Ack=94 Win=502 Len=0 TSval=160973646 TSecr=3826878682	
371 90.685927733	10.32.143.24	10.32.143.11	TCP	161 40704 - 65001 [PSH, ACK] Seq=1 Ack=94 Win=502 Len=95 TSval=160983993 TSecr=3826878682	
372 90.685968320	10.32.143.11	10.32.143.24	TCP	66 65001 - 40704 [ACK] Seq=94 Ack=96 Win=509 Len=0 TSval=3826889029 TSecr=160983993	
373 90.686215524	10.32.143.11	10.32.143.16	TCP	161 65001 - 38508 [PSH, ACK] Seq=1 Ack=94 Win=510 Len=95 TSval=2208748846 TSecr=2586153122	
274 00 606647026	10 22 142 16	40 22 442 44	TCD	66 20500 65004 [ACK] Cog-04 Ack-06 Win-502 Lon-0 Toyol-2506462474 TCoox-2200740046	

3) Há diferença, em termos de desempenho da aplicação, entre a aplicação com socket TCP e a aplicação com socket UDP?

TCP oferece confiabilidade e integridade de dados ao custo de maior latência e overhead, tornando-o ideal para aplicações onde a precisão dos dados é crucial. UDP sacrifica confiabilidade para priorizar baixa latência e menor overhead, o que o torna ideal para aplicações em tempo real e aquelas que podem tolerar perda ocasional de pacotes.

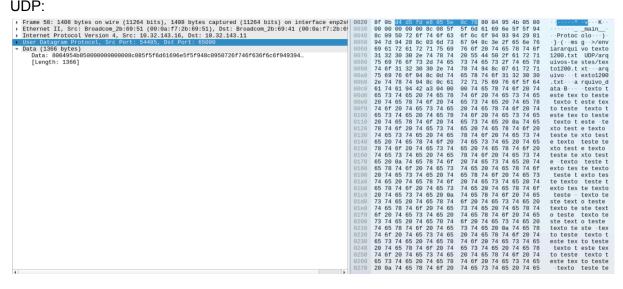
4) Compare a transmissão de um arquivo de 1200 bytes usando a socket TCP e socket UDP.

A transmissão TCP é mais confiável, pois garante que os dados cheguem completos e na ordem certa, mas inclui verificações que aumentam o tamanho dos pacotes. No UDP, os dados são enviados mais rápido, mas sem garantia de entrega correta, resultando em pacotes menores e menos controle.

TCP:

```
77 2b 69 51 08 90 45 90
aa 96 0a 20 87 10 0a 20
17 58 9a 76 09 b1 80 18
80 0a 9a 28 8c d9 83 a9
00 00 00 00 00 8c 08 57
8c 09 50 72 6f 74 6f 63
94 7d 94 28 8c 03 6d 73
95 61 72 6f 172 71 75 69
31 32 30 30 2e 74 78 74
75 69 75 67 73 2d 74 65
74 6f 31 32 30 30 2e 74
75 69 75 67 94 8c 0d 71
2e 74 75 74 94 8c 0d 71
2e 74 75 74 94 8c 0d 71
2e 74 75 74 96 75 97 36 76
55 73 74 65 20 74 65 73 74
78 74 6f 20 74 65 73 74
Frame 58: 1416 bytes on wire (11328 bits), 1416 bytes captured (11328 bits) on interface enp2s(
Ethernet II, Src: Broadcom_Zb:69:51 (00:0a:r7:2b:69:51), Dst: Broadcom_Zb:69:41 (00:0a:r7:2b:6)
Internet Protocol Version 4, Src: 10:32.143.16, Dst: 10:32.143.11
Transmission Control Protocol, Src Port: 38508, Dst Port: 65001, Seq: 1, Ack: 1, Len: 1350
Data (1350 bytes)
Data: 8004953b95000000000000000808575f6d61696e575f948c0950726f746f636f6c6f949394...
[Length: 1350]
```

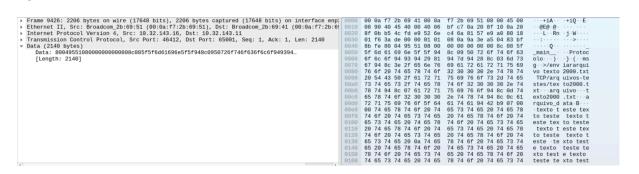
UDP:



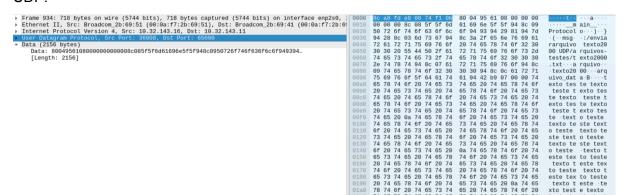
5) Compare a transmissão de um arquivo de 2000 bytes usando a socket TCP e socket UDP.

A transmissão via TCP é mais confiável, pois garante que os dados cheguem completos e na ordem correta, mas isso pode ser mais lento, pois inclui verificações e confirmações (ACKs). Já o UDP é mais rápido, mas não garante que os dados cheguem ou estejam completos, porque não faz verificações ou confirmações.

TCP:



UDP:



- 6) Configure a interface de rede da máquina para incluir perda de pacotes. Qual a diferença, em termos de tráfego na rede, entre o socket TCP e UDP? Houve alguma retransmissão usando TCP?
 - **TCP**: Quando os pacotes se perdem, o TCP percebe tenta retransmitir os dados perdidos para garantir que tudo chegue corretamente. Isso aumenta o tráfego, já que ele precisa enviar os pacotes de novo. Ele também ajusta a velocidade do envio para lidar com a perda, o que pode deixar a transmissão mais lenta.
 - **UDP**: O UDP, por outro lado, não faz essa verificação de perda de pacotes. Ele simplesmente continua enviando os dados, mesmo que alguns pacotes sejam perdidos. Isso resulta em menos tráfego, mas também não há garantias de que todos os dados vão chegar.

Resumo: O **TCP** gera mais tráfego quando há perda de pacotes, já que faz retransmissões, enquanto o **UDP** mantém o tráfego mais baixo, mas com risco de perder dados. E houve retransmissões usando o TCP.

- 7) Configurar a interface de rede da máquina para incluir latência variável. Qual a diferença, em termos de tráfego na rede, entre o socket TCP e UDP? Houve alguma retransmissão usando TCP?
 - TCP: Quando a latência aumenta, o TCP pode achar que os pacotes foram perdidos e, retransmite os dados, o que acaba gerando mais tráfego. Além disso, ele ajusta a velocidade de envio para evitar congestionamento, o que pode deixar a transmissão mais lenta quando a latência varia.
 - **UDP**: Já o UDP não se importa com a latência, ele simplesmente continua enviando os pacotes sem se preocupar se chegaram ou não. Isso mantém o tráfego constante, mas sem a garantia de que os dados vão chegar certinho no destino.

Resumo: Com latência variável, o **TCP** pode gerar mais tráfego por conta das retransmissões, enquanto o **UDP** mantém o tráfego estável, mas com maior chance de perder pacotes. E sim, houve retransmissões com o TCP.