

Atividade 5 - Comparação protocolos TCP e UDP

O objetivo deste roteiro é analisar como os protocolos TCP e UDP se comportam em diferentes cenários de rede usando as ferramentas `iperf`, `netem` e Wireshark em dois computadores Linux. Os testes simulam situações como congestionamento, perda de pacotes, atrasos e reordenação, permitindo observar diferenças de desempenho, confiabilidade e controle de congestionamento. As ferramentas permitem avaliar visualmente como cada protocolo reage e ajudam a entender seus usos ideais em aplicações reais.

iperf: Teste de Vazão de Rede

O `iperf` é uma ferramenta para medir a vazão (throughput) de rede entre dois computadores. Ele permite testar o desempenho de conexões TCP e UDP, avaliando velocidade de transmissão, perdas e jitter.

Como testar a vazão entre dois computadores

1. No computador servidor (destino):

```
iperf -s
```

2. No computador cliente (origem):

```
iperf -c <IP_do_servidor>          # Teste com TCP
iperf -c <IP_do_servidor> -u       # Teste com UDP
```

Para UDP, é necessário definir a taxa de envio com `-b`, pois o protocolo não controla congestionamento e não reduz a velocidade mesmo com perdas:

```
iperf -c <IP_do_servidor> -u -b 10M
```

O valor de `-b` (ex: `10M` para 10 megabits por segundo) deve ser escolhido com base na capacidade real da rede. Se for muito alto, os pacotes podem ser descartados pelos buffers da rede. Se for muito baixo, não utilizará todo o potencial disponível.

netem: Simulação de Condições de Rede

O `netem` permite simular problemas de rede como perda, atraso, limitação de banda, jitter, duplicação e reordenação de pacotes. É útil para testar como os protocolos se comportam em redes instáveis.

Nos exemplo a seguir, substitua `eth0` pela interface real do seu sistema (ex: `enp4s0`, `wlan0`). Use `ip link` ou `ifconfig` para descobrir o nome correto da interface em uso.

Exemplos de uso:

- Remover a configuração antes de um novo teste:

```
sudo tc qdisc del dev eth0 root
```

Sempre limpe a configuração anterior antes de aplicar uma nova simulação.

- **Limitar a banda para 2 Mbps:**

```
sudo tc qdisc add dev eth0 root netem rate 2mbit
```

- **Simular perda de 5% dos pacotes:**

```
sudo tc qdisc add dev eth0 root netem loss 5%
```

- **Simular atraso fixo de 200ms:**

```
sudo tc qdisc add dev eth0 root netem delay 200ms
```

- **Simular atraso com variação (jitter) de ± 20 ms:**

```
sudo tc qdisc add dev eth0 root netem delay 100ms 20ms
```

- **Simular pacotes fora de ordem para 25% dos pacotes:**

```
sudo tc qdisc add dev eth0 root netem delay 10ms reorder 25%
```

- **Simular duplicação de 2% dos pacotes:**

```
sudo tc qdisc add dev eth0 root netem duplicate 2%
```

- **Corromper 1% dos pacotes:**

```
sudo tc qdisc add dev eth0 root netem corrupt 1%
```

Experimentos Sugeridos (Fazer em aula)

Esta seção propõe experimentos práticos para comparar o comportamento dos protocolos TCP e UDP em diferentes condições de rede. Todos os testes devem ser realizados entre dois computadores Linux, utilizando `iperf`, `netem` e, opcionalmente, Wireshark para análise.

Entrega: Ao final da aula, entregue um documento com capturas de tela que evidenciem o comportamento esperado em cada experimento.

Experimento 1: Controle de congestionamento com múltiplos fluxos

Este experimento analisa como os protocolos TCP e UDP se comportam quando há múltiplos fluxos ativos ao mesmo tempo, disputando a mesma rede.

1. Inicie o servidor `iperf` no computador de destino:

```
iperf -s
```

2. No computador de origem, execute os testes:

TCP:

```
# 1 fluxo TCP
iperf -c <IP> -t 10

# 2 fluxos TCP simultâneos
iperf -c <IP> -t 10 & iperf -c <IP> -t 10

# 3 fluxos TCP simultâneos
iperf -c <IP> -t 10 & iperf -c <IP> -t 10 & iperf -c <IP> -t 10
```

UDP:

Use o parâmetro `-b` para limitar a taxa de envio e evitar perdas excessivas. Para evitar saturar a rede, a soma das taxas dos fluxos UDP deve ficar próxima ou abaixo da capacidade da rede, ex: 100 Mbps.

```
# 1 fluxo UDP (até 90% da capacidade da rede)
iperf -c <IP> -u -b 90M -t 10

# 2 fluxos UDP (45 Mbps cada)
iperf -c <IP> -u -b 45M -t 10 & iperf -c <IP> -u -b 45M -t 10

# 3 fluxos UDP (30 Mbps cada)
iperf -c <IP> -u -b 30M -t 10 & iperf -c <IP> -u -b 30M -t 10 & iperf -c <IP> -u -b 30M -t 10
```

Comportamento esperado:

- **TCP:**
 - Divide a banda disponível automaticamente entre os fluxos.
 - Aplica controle de congestionamento e retransmissão de pacotes em caso de perda.
 - O desempenho tende a ser estável, mesmo com múltiplas conexões.
- **UDP:**
 - Cada fluxo envia na taxa fixa especificada com `-b`, sem considerar congestionamento.
 - Se a soma das taxas exceder 100 Mbps, ocorrerá perda de pacotes.
 - A rede pode ficar saturada rapidamente se os valores de `-b` não forem ajustados corretamente.

Experimento 2: Controle de congestionamento com rede limitada

Simule uma rede com capacidade reduzida (ex: 2 Mbps) usando `netem` e observe como TCP e UDP se comportam.

```
sudo tc qdisc add dev eth0 root netem rate 2mbit
iperf -c <IP> -t 10 # TCP
```

```
iperf -c <IP> -u -b 10M -t 10 # UDP
sudo tc qdisc del dev eth0 root
```

Resultado esperado:

- TCP reduz a taxa para se adaptar.
- UDP tenta manter os 10 Mbps e pode causar perda significativa.

Experimento 3: Perda de pacotes com TCP e UDP

Simule uma rede com 10% de perda de pacotes usando `netem` e observe como TCP e UDP se comportam.

```
sudo tc qdisc add dev eth0 root netem loss 10%
iperf -c <IP> -t 10
iperf -c <IP> -u -b 5M -t 10
sudo tc qdisc del dev eth0 root
```

Resultado esperado:

- TCP reduz a taxa e retransmite.
- UDP mantém a taxa e ignora perdas.