

Análise de Técnicas de Qualidade de Serviço para Transmissão de Áudio e Vídeo em tempo Real

Objetivo:

O objetivo deste trabalho é aplicar, testar e analisar técnicas de Qualidade de Serviço (QoS) em um cenário realista de rede com transmissão de áudio e vídeo via RTP, simulada com o uso do Mininet. A implementação de QoS será realizada com o comando tc do Linux (ver Anexo). O foco será identificar quais técnicas de QoS são eficazes para garantir a qualidade da comunicação em situações de congestionamento de rede.

Descrição do trabalho:

Os alunos receberão um script base funcional que implementa:

- Uma topologia em Mininet com 4 hosts e 2 switches;
- Uma transmissão de vídeo + áudio via RTP de h1 para h2 usando ffmpeg e ffplay;
- Geração de tráfego de fundo (congestionamento) entre h3 e h4 usando iperf.

Os alunos deverão:

1. Estudar e compreender o script fornecido e como ocorre o congestionamento.
2. Aplicar técnicas de QoS estudadas na disciplina ao experimento, com o objetivo de:
 - a. Reduzir ou eliminar travamentos no vídeo/áudio;
 - b. Garantir desempenho aceitável mesmo sob congestionamento;
3. Analisar os resultados e apresentar evidências sobre o impacto das técnicas aplicadas (isto é, se funcionam ou não para resolver esse problema).
4. As técnicas podem ser testadas individualmente ou em combinação, conforme a estratégia de cada grupo.

Entrega e grupos:

O trabalho pode ser feito em grupos de até 3 integrantes. Cada grupo deverá entregar um relatório técnico com a descrição dos testes realizados, as técnicas de QoS aplicadas e evidências dos resultados (como gráficos, capturas de tela ou logs). Além disso, será feita uma apresentação em sala de aula, onde o grupo deverá explicar sua abordagem, mostrar os resultados obtidos e comentar sobre a eficácia das soluções adotadas.

ANEXO 1 – Comando tc

O comando tc (traffic control) é uma ferramenta do Linux utilizada para controlar o tráfego de rede em interfaces de rede. Ele permite aplicar diversas técnicas de Qualidade de Serviço (QoS), como programação (scheduling) para definir a prioridade de envio dos pacotes, formatação de tráfego (traffic shaping) para controlar rajadas e limitar taxas de transmissão, e a reserva de recursos para garantir banda mínima a fluxos sensíveis. O tc funciona organizando o tráfego em filas (queues), chamadas de qdiscs (queueing disciplines), e utilizando classes e filtros para determinar como os pacotes são tratados.

Principais conceitos no tc:

- Qdisc (Queueing Discipline): define como os pacotes são enfileirados e transmitidos.
- Class: subdivisões da largura de banda de uma interface. Podem ser criadas com regras distintas.
- Filter: define regras para classificar pacotes com base em critérios como endereço IP, porta ou protocolo.
- Rate / Ceil: definem a taxa mínima garantida e o máximo permitido em uma classe.
- Burst / Latency: usados para controlar picos de tráfego e atrasos em técnicas de shaping.

Técnicas de QoS que podem ser aplicadas com tc

- Programação (Scheduling): define como os pacotes são organizados e com que prioridade são transmitidos na interface de rede.
 - Formação de filas FIFO: comportamento padrão (pfifo_fast), onde os pacotes são processados na ordem de chegada.
 - Formação de filas por prioridade: Implementada com prio qdisc ou com htb usando diferentes prioridades (prio N) entre classes.
 - Formação de filas ponderadas: Implementada com htb, qfq, ou sfq, permitindo que diferentes fluxos compartilhem a banda com pesos distintos.
- Formatação de tráfego (Traffic Shaping): Controla a taxa de envio de pacotes, suavizando rajadas e garantindo conformidade com limites de banda.
 - Balde furado (leaky bucket): Simulado com tbf, limitando a saída a uma taxa constante com controle de latência e burst.
 - Balde de fichas (token bucket): Também com tbf, mas com configuração de burst maior, permitindo picos curtos acima da média permitida.
- Reserva de recursos (Resource Reservation): Pode ser simulada usando htb com classes com rate (mínimo garantido) para tráfego sensível (como RTP).