

Teleprocessamento e Redes - Relatório do trabalho final

Leonardo Ribeiro Santiago (120036072)
João Matheus Nascimento Gonçalves (117209640)
Esteves Emmanuel Melo Ferreira (120023786)

1 Introdução

Neste relatório iremos responder as perguntas relacionadas à parte 2 do trabalho final. O código está disponível no seguinte repositório do github: github.com/o-santi/redes.

Para reproduzir os resultados, deve-se instanciar uma máquina virtual Ubuntu usando Vagrant, assim como descrito em github.com/kaichengyan/mininet-vagrant. Uma vez dentro da VM, clonamos o repositório git para uma pasta interna, e rodamos o arquivo `run.sh`.

```
git clone https://github.com/o-santi/redes.git ~/redes
cd ~/redes
chmod +x ./bufferbloat/run.sh
sudo ./bufferbloat/run.sh
```

Isto irá rodar os dois casos de teste (`max_queue=20` e `max_queue=100`) e gerar os gráficos citados neste relatório.

2 Parte 2

- 2.1 Qual é o tempo médio de busca da página da web e seu desvio padrão quando `q=20` e `q=100`?
- 2.2 Por que você vê uma diferença nos tempos de busca de páginas da Web com buffers de roteador curtos e grandes?
- 2.3 Bufferbloat pode ocorrer em outros lugares, como sua placa de interface de rede (NIC). Verifique a saída de `ifconfig eth0` de sua VM mininet. Qual é o comprimento (máximo) da fila de transmissão na interface de rede relatada pelo `ifconfig`? Para esse tamanho de fila, se você assumir que a fila é “drenada” a 100 Mb/s, qual é o tempo máximo que um pacote pode esperar na fila antes de sair da NIC?
- 2.4 Como o RTT relatado pelo ping varia com o tamanho da fila? Descreva a relação entre os dois.
- 2.5 Identifique e descreva duas maneiras de mitigar o problema de bufferbloat.