

### Trabalho final

**Objetivo:** Aplicar os conhecimentos adquiridos para reprodução e análise de cenários reais de congestionamento.

#### Implementação e formato de entrega

- O trabalho final poderá ser implementado por uma dupla de alunos (no máximo). A nota é individual.
- Informe a dupla (por e-mail) até 13/junho/2023 23:59. A ausência de informação será interpretada como implementação individual.
- Para entrega final, a dupla deverá demonstrar o trabalho operacional. Usaremos os equipamentos da sala de aula para realizar a demonstração.
- Data limite para entrega e apresentação do trabalho: 04/julho/2023 e 06/julho/2023. As datas serão sorteadas.
- Com qual linguagem devo implementar o trabalho? Você escolhe.

#### Como entregar o trabalho?

- Durante a apresentação, além do código funcional, apresente um pequeno conjunto de slides descrevendo o procedimento realizado, os resultados obtidos e principalmente as observações e conclusões.

#### Cenário e descrição do trabalho

- Exemplo inicial. Considerando uma interconexão de 100Mbps entre os computadores 1 e 2 (cenário da sala de aula).
- Quais ferramentas devo utilizar? iperf3 (<https://iperf.fr/iperf-doc.php>), ss (<https://man7.org/linux/man-pages/man8/ss.8.html>) e WireShark.



- 1) A comunicação entre os processos PA e PB será utilizada como gerador de tráfego de *background* não controlado, ou seja, utilizará UDP. Utilize a ferramenta iperf para gerar o tráfego UDP de forma controlada.
- 2) Os processos P0, P1, ... Pn, Pn+1 serão utilizados para geração de tráfego TCP e coleta de dados. Utilize a ferramenta iperf para gerar o tráfego TCP e a ferramenta ss para coletar os dados internos de controle.

## Plano de testes

- 1) A base do cenário experimental será sorteada no dia 14/junho/2023. Cada dupla receberá seu cenário. As possibilidades são:
  - a. Estudo aumentando gradativamente a carga UDP. Qual o impacto no TCP? Ocorre *starvation*?
  - b. Estudo comparando CUBIC e NEW RENO variando o número de processos e a vazão disponível. Use a ferramenta netem (<https://wiki.linuxfoundation.org/networking/netem>) para configurar o cenário.
  - c. Estudo comparando CUBIC e NEW RENO variando o número de processos e a latência máxima. Use a ferramenta netem (<https://wiki.linuxfoundation.org/networking/netem>) para configurar o cenário.
  - d. Estudo comparando tráfego em rajadas e tráfego elefante.
- 2) Executando o cenário
  - a. Após montar o cenário experimental, identifique quais métricas você irá coletar. Opções: cwnd, ssthresh, vazão total, vazão útil, bytes perdidos, entre outros.
  - b. Cada métrica deverá ser analisada em um gráfico específico. Exemplos de gráficos:
    - i. Evolução temporal: o eixo X indica a passagem do tempo, enquanto o eixo Y apresenta os dados obtidos (sendo comparados).
    - ii. Comparações: média e desvio padrão.
  - c. Selecione a ferramenta apropriada para gerar os gráficos.
  - d. Para cada cenário, elabore um gráfico da vazão obtida pelo tráfego de *background* (UDP).