**GUIA**

Como a quantidade de arquivos ficou meio grande, resolvi criar um guia para facilitar a leitura.

# Arquivo ANALISE.docx

Documento com a analise aprofundada do tráfego.

# Arquivo compiler.sh

Basicamente é só um script que compila todos os arquivos de uma vez.

# Arquivo ip.txt

**O ÚNICO QUE DEVE SER MODIFICADO**, aqui inserimos o IP do computador servidor.

# Arquivo list.txt

Esse é o arquivo que será enviado ao servidor repentinas vezes.

# Arquivos UdpClientTest.java e UdpServerTest.java

São os arquivos que controlam a aplicação UDP, executa na porta 9000, devem ser inicializados na seguinte ordem.

1. java UdpServerTest
2. java UdpClientTest

Internamente eles apenas inicializam os arquivos UdpServer e UdpClient respectivamente.

# Arquivos TcpClientTest.java e TcpServerTest.java

São os arquivos que controlam a aplicação TCP, executa na porta 9001, devem ser inicializados na seguinte ordem

1. java TcpServerTest
2. java TcpClientTest

Internamente eles apenas inicializam os arquivos TcpServer e TcpClient respectivamente.

# Diretório VisualAnalysis/

Toda a vez que finalizamos nossa análise, encerrando o processo TcpClientTest ou UdpClientTest, o programa salva a estrutura de logs (todos os window,timeout,perdas ao longo da execução), e insere essa informação no arquivo index.html dentro da pasta visualAnalysis, sendo assim, podemos arrastar o arquivo index.html ao navegador, e teremos uma forma de visualizar os dados que percorreram nosso programa.

O arquivo index.html, mostra 4 conjuntos de gráficos: os resultados obtidos através da conexão Udp, os resultados obtidos através da coneção Tcp, e os resultados obtidos através da conexão Udp ou Tcp com congestionamento pelo Iperf. Para identificarmos os resultados que estão sendo congestionados pelo Iperf devemos acrescentar o parâmetro 1 ao inicializarmos TcpClientTest ou UdpClientTest, como abaixo

java TcpClientTest 1 ou java UdpClientTest 1

Isso não adiciona automaticamente o IPERF ao nosso tráfego, ele deve ser criado manualtmente através de outros terminais, o parâmetro serve apenas para identificar se o teste pertence aos gráficos com congestionamento ou sem.

# Demais arquivos

Não precisam ser detalhados, pois fazem parte do funcionamento interno da aplicação.