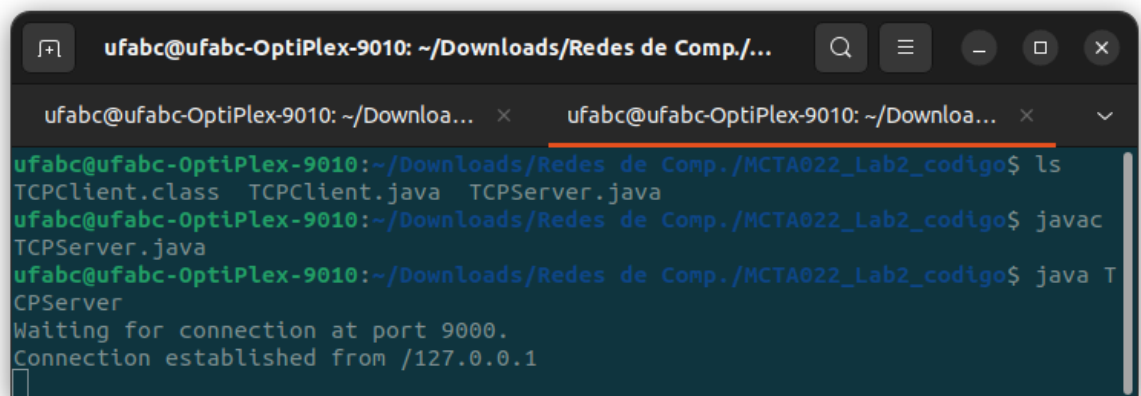


Redes de Computadores

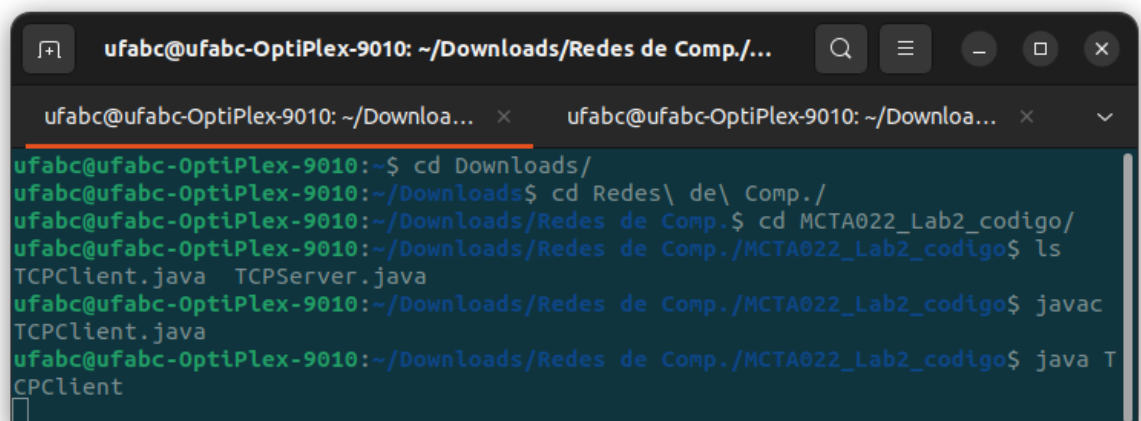
Náthaly Martins De Sá

Tópico 02

1. Aplicação cliente-servidor TCP .
 - 1.1



```
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./...
ufabc@ufabc-OptiPlex-9010: ~/Downloa... x  ufabc@ufabc-OptiPlex-9010: ~/Downloa... x
ufabc@ufabc-OptiPlex-9010:~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ ls
TCPClient.class TCPClient.java TCPServer.java
ufabc@ufabc-OptiPlex-9010:~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ javac
TCPServer.java
ufabc@ufabc-OptiPlex-9010:~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ java T
CPServer
Waiting for connection at port 9000.
Connection established from /127.0.0.1
```



```
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./...
ufabc@ufabc-OptiPlex-9010: ~/Downloa... x  ufabc@ufabc-OptiPlex-9010: ~/Downloa... x
ufabc@ufabc-OptiPlex-9010:~$ cd Downloads/
ufabc@ufabc-OptiPlex-9010:~/Downloads$ cd Redes\ de\ Comp./
ufabc@ufabc-OptiPlex-9010:~/Downloads/Redes de Comp.$ cd MCTA022_Lab2_codigo/
ufabc@ufabc-OptiPlex-9010:~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ ls
TCPClient.java TCPServer.java
ufabc@ufabc-OptiPlex-9010:~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ javac
TCPClient.java
ufabc@ufabc-OptiPlex-9010:~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ java T
CPClient
```

Ao executar primeiro o programa servidor e após o programa cliente, o servidor aguarda conexão com cliente, que ao ser executado se conecta instantaneamente com o servidor.

1.2

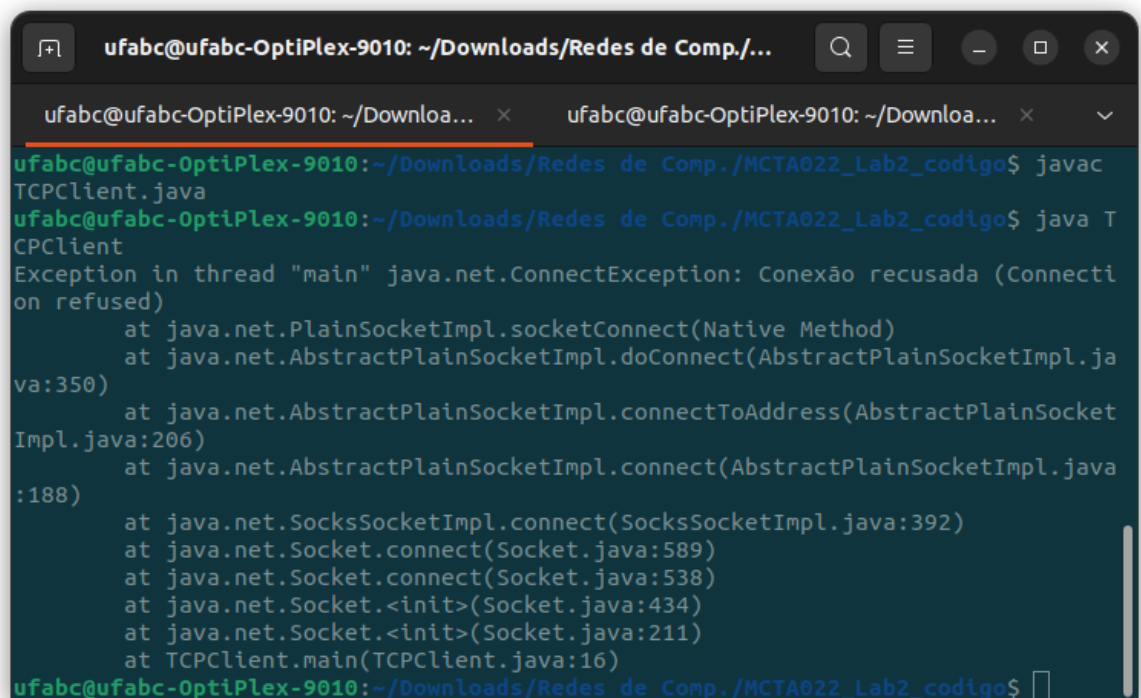
```
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./...  
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ java TCPClient  
Exception in thread "main" java.net.ConnectException: Conexão recusada (Connection refused)  
    at java.net.PlainSocketImpl.socketConnect(Native Method)  
    at java.net.AbstractPlainSocketImpl.doConnect(AbstractPlainSocketImpl.java:350)  
    at java.net.AbstractPlainSocketImpl.connectToAddress(AbstractPlainSocketImpl.java:206)  
    at java.net.AbstractPlainSocketImpl.connect(AbstractPlainSocketImpl.java:188)  
    at java.net.SocksSocketImpl.connect(SocksSocketImpl.java:392)  
    at java.net.Socket.connect(Socket.java:589)  
    at java.net.Socket.connect(Socket.java:538)  
    at java.net.Socket.<init>(Socket.java:434)  
    at java.net.Socket.<init>(Socket.java:211)  
    at TCPClient.main(TCPClient.java:16)  
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$
```

```
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./...  
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ java TCPServer  
Waiting for connection at port 9000.  
[ ]
```

Já ao gerar primeiro o programa cliente e somente após o programa servidor, o cliente gera uma falha, pois é necessário que o servidor já esteja aguardando conexão em tal porta para execução do código cliente, que obrigatoriamente assim que executar, precisa ter a conexão aceita.

1.3

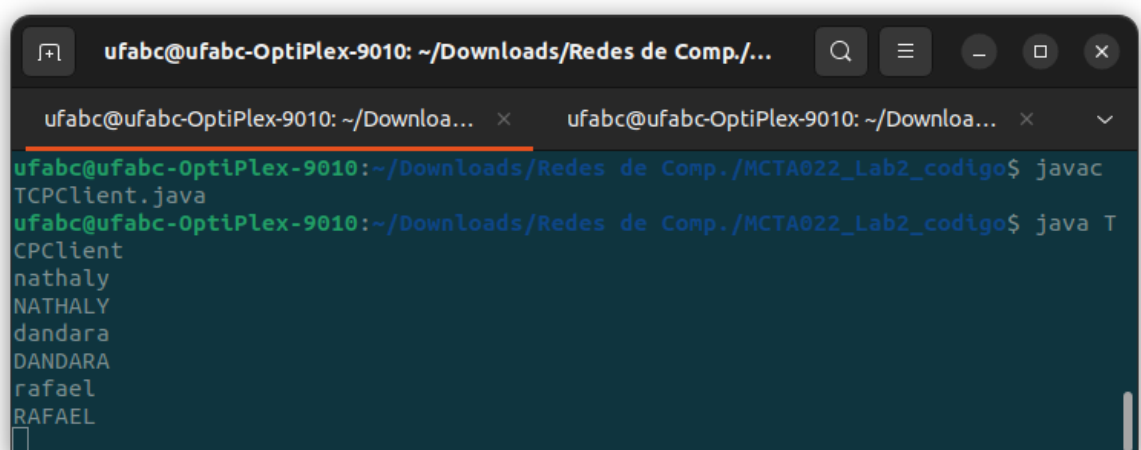
```
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./...  
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ javac TCPServer.java  
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ java TCPServer  
Waiting for connection at port 9800.  
[ ]
```

A terminal window titled 'ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./...' showing the compilation and execution of a Java program. The user runs 'javac TCPClient.java' and then 'java TCPClient'. The output shows a 'java.net.ConnectException: Conexão recusada (Connection refused)' error, with a stack trace indicating the failure occurred in the 'main' thread while attempting to connect to a remote address.

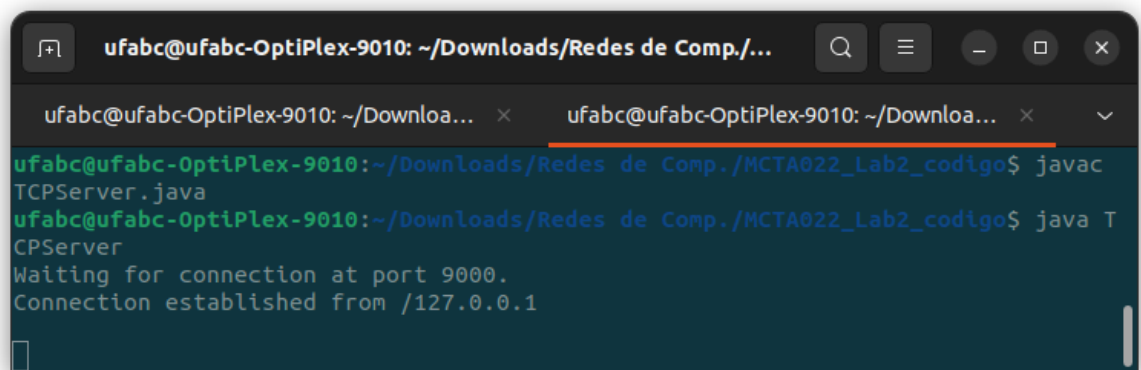
```
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ javac TCPClient.java
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ java TCPClient
Exception in thread "main" java.net.ConnectException: Conexão recusada (Connection refused)
    at java.net.PlainSocketImpl.socketConnect(Native Method)
    at java.net.AbstractPlainSocketImpl.doConnect(AbstractPlainSocketImpl.java:350)
    at java.net.AbstractPlainSocketImpl.connectToAddress(AbstractPlainSocketImpl.java:206)
    at java.net.AbstractPlainSocketImpl.connect(AbstractPlainSocketImpl.java:188)
    at java.net.SocksSocketImpl.connect(SocksSocketImpl.java:392)
    at java.net.Socket.connect(Socket.java:589)
    at java.net.Socket.connect(Socket.java:538)
    at java.net.Socket.<init>(Socket.java:434)
    at java.net.Socket.<init>(Socket.java:211)
    at TCPClient.main(TCPClient.java:16)
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$
```

Código alterado disponível para consulta. Sendo assim, com portas diferentes, não é possível prosseguir, pois obrigatoriamente assim que o cliente é executado, precisará ter a conexão aceita e com portas diferentes, os programas percorrem caminhos diferentes e a conexão não será bem sucedida.

1.4

A terminal window titled 'ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./...' showing the compilation and execution of a Java program. The user runs 'javac TCPClient.java' and then 'java TCPClient'. The output shows the program running successfully, displaying the names 'nathaly', 'NATHALY', 'dandara', 'DANDARA', 'rafael', and 'RAFAEL' on separate lines.

```
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ javac TCPClient.java
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ java TCPClient
nathaly
NATHALY
dandara
DANDARA
rafael
RAFAEL
```

A terminal window with a dark background. The title bar shows the user 'ufabc' on a machine named 'ufabc-OptiPlex-9010' in the directory '~/Downloads/Redes de Comp./...'. There are two tabs open, both with the same title. The terminal content shows the user running 'javac TCPServer.java' and then 'java TCPServer'. The output indicates the server is waiting for a connection at port 9000, and a connection is established from 127.0.0.1.

```
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ javac TCPServer.java
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ java TCPServer
Waiting for connection at port 9000.
Connection established from /127.0.0.1
```

Como mostrado acima, o servidor não tem o poder de atender a vários clientes ao mesmo tempo, pois TCP é formato unicast e ao tentar enviar mensagens de várias máquinas diferentes, o servidor irá estabelecer conexão com a primeira máquina, receber os pacotes e esperar o cliente desconectar para estabelecer conexão com a próxima máquina, processando assim uma conexão por vez.

1.5

A principal diferença notável entre os códigos é o protocolo de transporte usado, pois códigos em TCP usam um socket de fluxo (classe `ServerSocket`) para receber conexões de clientes e estabelecer uma comunicação bidirecional. É criado um socket de fluxo separado (classe `Socket`) para cada conexão de cliente e usa objetos `BufferedReader` e `DataOutputStream` para ler e escrever dados através do socket de conexão. Esse servidor espera por uma nova conexão do cliente e bloqueia até que uma conexão seja estabelecida.

Enquanto que em códigos em UDP, é usado um socket de datagrama (classe `DatagramSocket`) para receber datagramas de clientes e enviar datagramas de resposta de volta aos clientes. É bloqueado em um loop infinito até que um datagrama seja recebido. Quando um datagrama é recebido, ele lê os dados do pacote e cria um novo pacote para enviar de volta ao cliente. Ao contrário do TCP, o UDP não estabelece uma conexão antes de enviar dados.

Outras diferenças são:

Fluxo de dados: O TCP usa um fluxo de dados de entrada e saída para se comunicar com o servidor, enquanto o UDP usa datagramas de entrada e saída. O fluxo de dados é usado quando a comunicação é orientada a conexão, enquanto os datagramas são usados quando a comunicação é sem conexão.

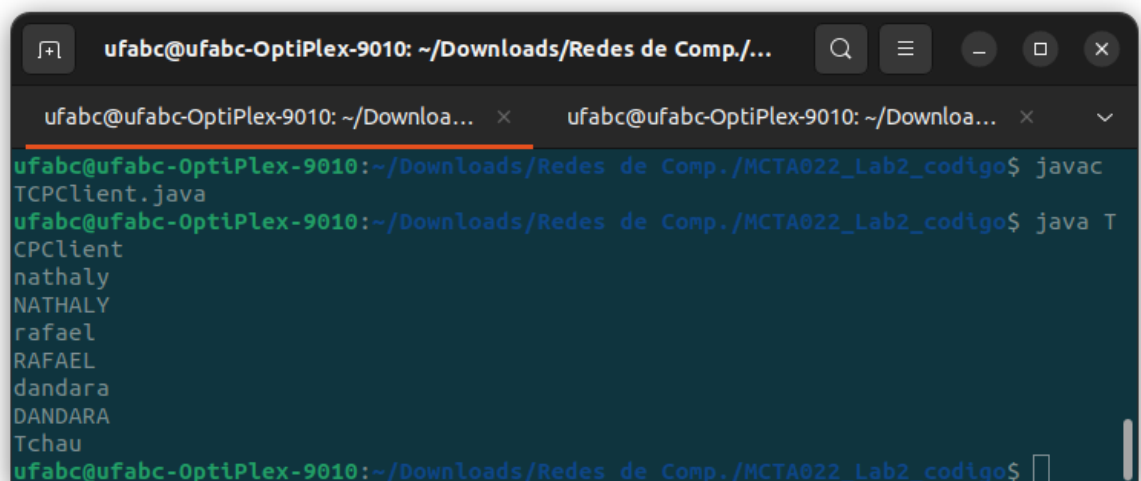
Tamanho dos pacotes: O tamanho dos pacotes em UDP é limitado a 65.507 bytes, enquanto o tamanho dos pacotes em TCP não tem um limite definido.

Confiabilidade: O TCP é um protocolo confiável, o que significa que ele garante que todos os dados sejam recebidos e na ordem correta. O UDP é um protocolo não confiável, o que significa que não há garantia de que todos os dados sejam recebidos ou na ordem correta.

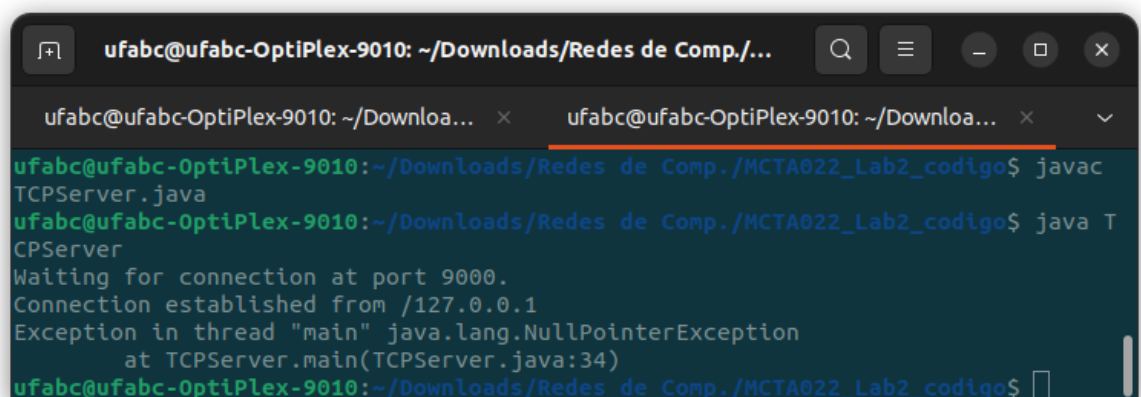
Overhead: O TCP tem um overhead maior do que o UDP devido ao estabelecimento de conexão e controle de fluxo e congestionamento. O UDP tem um overhead menor, já que é um protocolo sem conexão.

Em resumo, a escolha entre TCP e UDP depende dos requisitos da aplicação. Se a confiabilidade/conexão for importante e os dados precisarem ser recebidos na ordem correta, sem nenhuma perda, o TCP é mais adequado. Porém, se a velocidade for importante e a perda de alguns pacotes for aceitável, o UDP pode ser usado.

1.6



```
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./...  
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ javac TCPClient.java  
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ java TCPClient  
nathaly  
NATHALY  
rafael  
RAFAEL  
dandara  
DANDARA  
Tchau  
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$
```



```
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./...  
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ javac TCPServer.java  
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$ java TCPServer  
Waiting for connection at port 9000.  
Connection established from /127.0.0.1  
Exception in thread "main" java.lang.NullPointerException  
    at TCPServer.main(TCPServer.java:34)  
ufabc@ufabc-OptiPlex-9010: ~/Downloads/Redes de Comp./MCTA022_Lab2_codigo$
```

Código alterado disponível para consulta e funcionando normalmente como apresentado acima. Sendo assim, ao ser enviado a mensagem “Tchau” pelo cliente, a conexão é desfeita, encerrando tanto o cliente quanto o servidor.

2. Desenvolvimento de um servidor TCP com múltiplos clientes.

2.1 Código alterado disponível para consulta, porém não foi possível captura de tela, pois infelizmente não tive acesso a outras máquinas até o momento.