

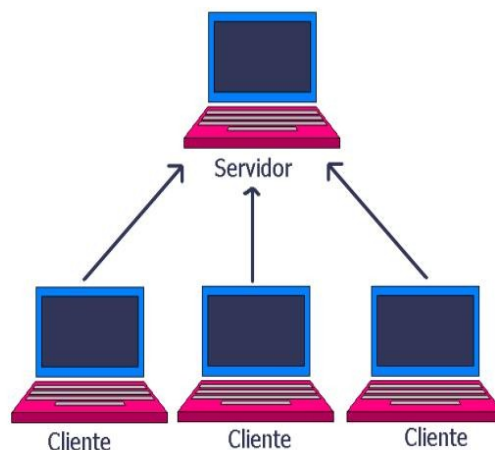
# Documentação

## Trabalho prático 2 de redes de computadores

Nome: Vinícius Moraes Silva - 2012050047

### Introdução:

Neste trabalho foi feito um chat, que simula o IRC, nele um cliente irá se conectar a um canal-servidor e poderá postar mensagens que serão repassadas a todos os outros usuários da rede.



### Desenvolvimento:

Foi criado um programa Cliente.c e um Servidor.c. No cliente o usuário executa o código com os parâmetros `./cliente <IP><PORTA><USUARIO>`, sendo o IP o número de ip, a PORTA o número da porta a ser conectado, e USUARIO o nome inicial e final do cliente. O servidor tem como entrada apenas o número da porta que usará para se conectar `./servidor<PORTA>`.

O Servidor irá realizar as seguintes funções:

1. Irá cadastrar um cliente que queira se juntar ao chat, ip e nomes de usuários iguais são rejeitados pelo servidor.
2. Ao receber um post de algum cliente, irá transmitir para os demais clientes
3. Se um cliente A post alguma mensagem, e se o cliente B mutou A, então B não irá receber o post de A
4. Mutar algum cliente A para o cliente B, e também desmutar. Essa função é pedida do cliente
5. Ao receber um comando close, irá deletar o cliente que enviou este comando da lista de clientes.

O cliente irá realizar as seguintes funções:

1. Ao ser executado o cliente automaticamente pede para o servidor para se cadastrar
2. O cliente pode postar um mensagem usando o comando `POST <mensagem>` que

- será enviados para todos os outros clientes.
3. Pode mutar algum usuário com o comando MUTE <nick> que irá impedir que receba mensagens do usuário que foi mutado
  4. Realizar o comando UNMUTE <nick> que faz que o usuário volte a receber as mensagens do usuário mutado.
  5. Realizar o comando CLOSE que irá encerrar a conexão.

## COMANDOS:

Servidor: Nenhum

Clientes:

1. POST <MENSAGEM>
2. MUTE <NICK ALVO>
3. UNMUTE <NICK ALVO>
4. CLOSE

## DETALHES DE IMPLEMENTAÇÃO:

Servidor:

1. O servidor possui uma estrutura “Client” que guarda as informações necessárias de cada clientes
2. O chat pode suportar até 1000 clientes simultaneamente
3. Como a função do soquete recvfrom() gera uma parada de sistema, então é criado uma thread para a função que captura as mensagens dos clientes.

## CLIENTE:

1. Para capturar as mensagens enviadas do servidor foi criada uma thread para a função escuta(). Enquanto um laço principal é executado para capturar os comandos do usuário.

## DECISÕES:

1. Não foi implementado a função NEW
2. Não foi verificado se uma mensagens chegou ou não ao seu destino, logo a perda de mensagens não foi tratada.

## Testes:

Os teste foram realizados em 4 terminais diferentes simultaneamente no mesmo computador, 1 para o servidor e 3 para os clientes.

### TERMINAL 1:

vini@vini-VirtualBox:~/Área de Trabalho/TPREDES2\$ ./servidor 9999

Sun Dec 20 17:37:04 2015

: USUARIO: juca entrou na sala 127.0.0.3

Sun Dec 20 17:37:07 2015

: USUARIO: maria entrou na sala 127.0.0.2

Sun Dec 20 17:37:12 2015

: USUARIO: joao entrou na sala 127.0.0.1

juca :: OI EU SOU JUCA

Sun Dec 20 17:37:34 2015

: Mandando de juca para maria

Sun Dec 20 17:37:34 2015

: Mandando de juca para joao

maria :: OLHA JUCA

Sun Dec 20 17:37:41 2015

: Mandando de maria para juca

Sun Dec 20 17:37:41 2015

: Mandando de maria para joao

joao :: OLA JUCA

Sun Dec 20 17:37:48 2015

: Mandando de joao para juca

Sun Dec 20 17:37:48 2015

: Mandando de joao para maria

Sun Dec 20 17:37:58 2015

: maria mutou juca com sucesso

juca :: ENTAO COMO VAI VCS?

Sun Dec 20 17:38:10 2015

: maria estah mutato para juca

Sun Dec 20 17:38:10 2015

: Mandando de juca para joao

joao :: VOU BEM

Sun Dec 20 17:38:20 2015

: Mandando de joao para juca

Sun Dec 20 17:38:20 2015

: Mandando de joao para maria

Sun Dec 20 17:38:47 2015

: maria desmutou juca com sucesso

juca :: IREI SAIR AKI BJS AT+

Sun Dec 20 17:39:02 2015

: Mandando de juca para maria

Sun Dec 20 17:39:02 2015

: Mandando de juca para joao

Sun Dec 20 17:39:06 2015

: usuario: juca saiu do chat  
Sun Dec 20 17:39:24 2015  
: usuario: joao saiu do chat  
Sun Dec 20 17:39:32 2015  
: usuario: maria saiu do chat

#### TERMINAL 2:

vini@vini-VirtualBox:~/Área de Trabalho/TPREDES2\$ ./cliente 127.0.0.1 9999 joao

Sun Dec 20 17:37:34 2015  
juca :: OI EU SOU JUCA  
Sun Dec 20 17:37:41 2015  
maria :: OLHA JUCA  
POST OLA JUCA

Sun Dec 20 17:38:10 2015  
juca :: ENTAO COMO VAI VCS?  
POST VOU BEM

Sun Dec 20 17:39:02 2015  
juca :: IREI SAIR AKI BJS AT+  
Sun Dec 20 17:39:06 2015  
: USUARIO: juca saiu da sala  
Sun Dec 20 17:39:06 2015  
: USUARIO: juca saiu da sala

CLOSE

#### TERMINAL 3:

vini@vini-VirtualBox:~/Área de Trabalho/TPREDES2\$ ./cliente 127.0.0.2 9999 maria

Sun Dec 20 17:37:12 2015  
: USUARIO: joao entrou na sala  
Sun Dec 20 17:37:12 2015  
: USUARIO: joao entrou na sala

Sun Dec 20 17:37:34 2015  
juca :: OI EU SOU JUCA  
POST OLHA JUCA

Sun Dec 20 17:37:48 2015  
joao :: OLA JUCA  
MUTE juca

Sun Dec 20 17:38:20 2015  
joao :: VOU BEM  
UNMUTE juca

Sun Dec 20 17:39:02 2015  
juca :: IREI SAIR AKI BJS AT+

Sun Dec 20 17:39:06 2015  
: USUARIO: juca saiu da sala

Sun Dec 20 17:39:24 2015  
: USUARIO: joao saiu da sala

CLOSE

TERMINAL 4:

vini@vini-VirtualBox:~/Área de Trabalho/TPREDES2\$ ./cliente 127.0.0.3 9999 juca

Sun Dec 20 17:37:07 2015  
: USUARIO: maria entrou na sala

Sun Dec 20 17:37:12 2015  
: USUARIO: joao entrou na sala

POST OI EU SOU JUCA

Sun Dec 20 17:37:41 2015  
maria :: OLHA JUCA  
Sun Dec 20 17:37:48 2015  
joao :: OLA JUCA  
POST ENTAO COMO VAI VCS?

Sun Dec 20 17:38:20 2015  
joao :: VOU BEM  
POST IREI SAIR AKI BJS AT+

CLOSE

## **Análise:**

Os testes foram bem sucedidos, os clientes conseguem conectar no servidor com um nick e um IP, os POST de cada Cliente é recebido por todos os outros, o comando MUTE e UNMUTE não apresentaram problemas e funcionam corretamente. O servidor subtrai os clientes que enviaram o comando CLOSE.

Os únicos problemas encontrados é a duplicata de mensagens que ocorre quando alguém entra na sala ou sai da sala.

## **Conclusão:**

A partir deste trabalho é possível analisar como um chat em geral funciona, como as mensagens são enviadas e recebidas e como o uso de sockets funciona para UDP. As únicas dificuldades apresentadas foi estabelecer várias conexões simultaneamente usando threads.

## **Dados Técnicos:**

Tanto o programa quanto os teste foram realizados dentro de uma maquina virtual Ubuntu 14.04

## **Referências:**

- 1- <http://www.cplusplus.com>
- 2- <https://www.cs.rutgers.edu/~pxk/417/notes/sockets/udp.html>

## **Executando:**

make para compilar, gera um arquivo “cliente” e outro “servidor”  
make clean para limpar os executaveis