

Pontifícia Universidade Católica – PUCRS

Trabalho 1 – Redes de Comunicação 2

Alunos: Fernanda Gallina e Thiago Subilhaga

Porto Alegre, 10 de novembro de 2015

Implementação

Para o desenvolvimento desse trabalho foram feitos dois arquivos C++, cliente e servidor. O servidor terá a função de se comunicar com um cliente e com outros servidores. Para a comunicação com o cliente foi usado protocolo TCP e entre os servidores UDP.

No servidor foram criadas 4 threads:

- HeartbeatCtrl: fica verificando nos servidores seus vetores de vizinhos. Foi feito um laço que percorre o vetor e verifica se o tempo que o último heartbeat foi enviado não ultrapassou 30 segundos, se sim, retira o mesmo servidor vizinho de sua lista.
- recvUDP: possui a função de receber as mensagens de START e HEARTBEAT dos servidores. Quando é recebido um START, o servidor adiciona o novo vizinho em sua lista, pondo o nome recebido junto e para tempo de heartbeat coloca-se o tempo atual, em seguida envia-se um comando HEARTBEAT ao servidor origem. Quando o comando é HEARTBEAT é atualizado o tempo de heartbeat(chamado no programa de beat) do vizinho.
- sendUDP: quando criado um servidor, o mesmo envia um comando de START para todos conectados na porta (broadcast), em seguida de 10 em segundos ele envia um comando de HEARTBEAT para seus vizinhos.
- TCP: primeiro é estabelecido contato com o cliente, em seguida o servidor fica esperando comando do cliente para poder responder em seguida. Para todos os comandos foi feito condições encadeadas.
 - o NAME: se a mensagem proveniente do cliente for apenas a palavra "NAME", o servidor envia o seu nome.
 - o NAME <name>: se houver na mensagem um nome junto com "NAME", o servidor muda seu nome.
 - o READ <byte/word> <address>: utilizando o programa dado pelo professor para auxiliary leitura e escrita num arquivo, quando a mensagem chega corretamente, verifica-se como deve ser lido do endereço pedido, em 'byte' ou 'word'.
 - o WRITE <byte/word> <address> <value>: função muito parecida com READ, mas ao invés de ler, deve escrever na memória. Por esse motivo vem na mensagem um atributo a mais.
 - o SHUTDOWN: quando recebida essa mensagem, o servidor encerra seus serviços.

- o HOSTS: lista todos os servidores vizinhos, com seu IP e nome, e o número de dispositivos.

Em todos comandos, a mensagem é montada bem antes de enviar, para não haver redundância no código é utilizado apenas um 'send'. Todos comandos, se estiverem corretos, retornam 200 OK. Se há algum problema em como foi escrito o comando, é retornado 400 Not Found. Se nas mensagens de READ/WRITE o endereço não é válido o retorno é: 404 Invalid address.

O cliente possui uma estrutura mais simples, ele estabelece conexão com o servidor, envia o comando desejado e fica esperando resposta do servidor, assim que recebida a conexão é encerrada e o cliente fecha.

Lado do cliente:

```
fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
HOSTS
200 OK
2
15.29.227.164 redes
15.29.227.164 2
fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
NAME
200 OK
redes
fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
WRITE byte 0 255
200 OK
byte 255
fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
READ word 255
200 OK
word 255
```

```
fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
NAME aula
200 OK
aula
fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
SHUTDOWN
200 OK
```

Lado do servidor:

```
fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./server1 8000 8000 redes
redes
START redes
received START from redes
Handling client 15.29.227.164
HEARTBEAT redes
received HEARTBEAT from redes
HEARTBEAT redes
received HEARTBEAT from redes
HEARTBEAT redes
received HEARTBEAT from redes
HEARTBEAT redes
received HEARTBEAT from redes
HEARTBEAT redes
received HEARTBEAT from redes
received START from 2
HEARTBEAT redes
received HEARTBEAT from redes
received HEARTBEAT from 2
```

```
fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./server1 8000 8001 2
2
START 2
received START from 2
HEARTBEAT 2
received HEARTBEAT from 2
received HEARTBEAT from redes
HEARTBEAT 2
received HEARTBEAT from 2
HEARTBEAT 2
received HEARTBEAT from 2
HEARTBEAT 2
received HEARTBEAT from 2
Apagando host redes
```

Obs.: para auxiliar na depuração do programa, foi habilitado o uso de mais de um servidor e mais de um cliente por porta.

Retorno para comandos errados:

```
Fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
name
400 Not Found
Fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
shutdown
400 Not Found
Fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
READ
400 Not Found
Fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
READ byte
400 Not Found
Fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
READ BYTE 0
400 Not Found
```

```
Fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
WRITE byte 0
400 Not Found
Fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
hosts
400 Not Found
Fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
WRITE word 257 257
404 Invalid address
```

```
Fernanda@fernanda-HP-Z600-Workstation:~/Desktop/REDES2-v2 $ ./client 15.29.227.164 8000
Comando desejado:
READ byte 257
404 Invalid address
```