

## Teste programa SD - Socket UDP em C

2 mensagens

Anny Caroline

9 de julho de 2019 13:30

Para: Alexandre Sztajnberg

Professor, acho que realmente tinha um problema no exemplo. Quando compilo utilizando a versão 7.4.0 do gcc, os arquivos udpClient.c e udpServer.c geram o seguinte erro:

```
anny@anny:~/Area de Trabalho/socket-c-udp$ gcc udpClient.c
udpClient.c: In function 'main':
udpClient.c:71:8: warning: implicit declaration of function 'inet_ntoa' [-Wimplicit-function-declaration]
    inet_ntoa(cli addr.sin addr), ntohs(cli addr.sin port));
    ^~~~~~
udpClient.c:70:42: warning: format '%s' expects argument of type 'char *', but argument 2 has type 'int' [-Wformat=]
    printf("Recebido resposta do servidor %s:%d\n",
                                   ~^
                                   %d
    inet_ntoa(cli addr.sin addr), ntohs(cli addr.sin port));
    ^~~~~~
udpClient.c:75:3: warning: implicit declaration of function 'close'; did you mean 'pclose'? [-Wimplicit-function-declaration]
    close(sock);
    ^~~~~
pclose
```

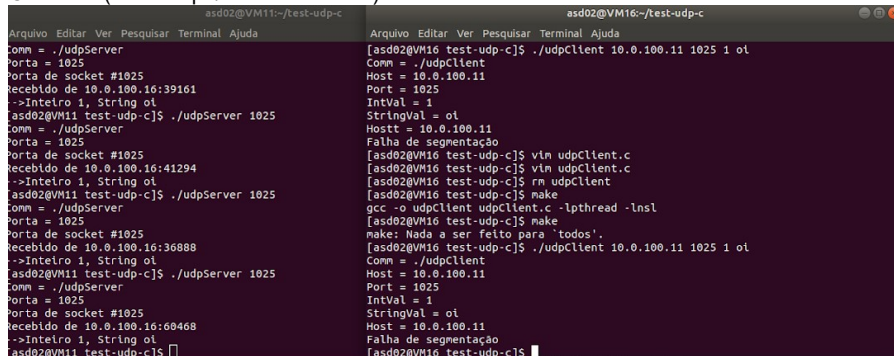
Para resolver, basta incluir arpa/inet.h e unistd.h:

```
#include <arpa/inet.h>
#include <unistd.h>
```

Testando na versão nas máquinas 11 e 16, que utilizam a versão 4.4.7-18 do gcc, nenhum erro é gerado na compilação, mas o cliente gera uma falha de segmentação na hora de criar o socket (apesar disso, a mensagem é recebida pelo servidor).

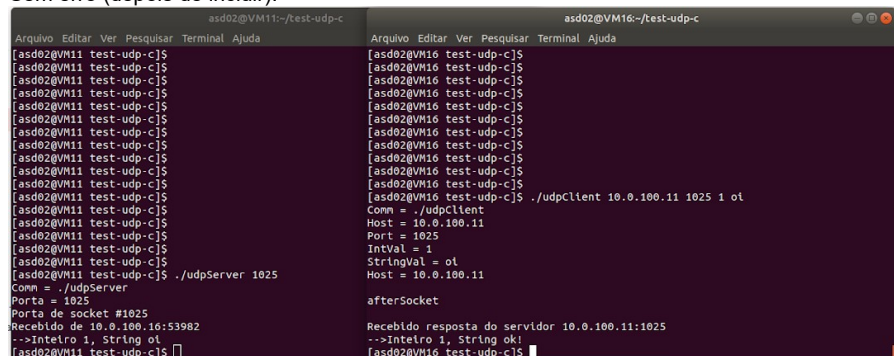
A solução é a mesma: **basta incluir arpa/inet.h e unistd.h nos arquivos udpClient.c e udpServer.c.**

Com erro (incluir arpa/inet.h e unistd.h):



```
as02@VM11:~/test-udp-c$ gcc udpClient.c
as02@VM16:~/test-udp-c$ gcc udpClient.c
[as02@VM16 test-udp-c]$ ./udpClient 10.0.100.11 1025 1 01
Comm = ./udpClient
Host = 10.0.100.11
Port = 1025
IntVal = 1
StringVal = 01
Hostt = 10.0.100.11
Falha de segmentação
[as02@VM16 test-udp-c]$ vim udpClient.c
[as02@VM16 test-udp-c]$ vim udpClient.c
[as02@VM16 test-udp-c]$ rm udpClient.c
[as02@VM16 test-udp-c]$ make
gcc -o udpClient udpClient.c -lpthread -lnsl
[as02@VM16 test-udp-c]$ make
make: Nada a ser feito para 'todos'.
Comm = ./udpClient
Host = 10.0.100.11
Port = 1025
IntVal = 1
StringVal = 01
Host = 10.0.100.11
Falha de segmentação
[as02@VM16 test-udp-c]$
```

Sem erro (depois de incluir):



```
as02@VM11:~/test-udp-c$ gcc udpClient.c
as02@VM16:~/test-udp-c$ gcc udpClient.c
[as02@VM16 test-udp-c]$ ./udpClient 10.0.100.11 1025 1 01
Comm = ./udpClient
Host = 10.0.100.11
Port = 1025
IntVal = 1
StringVal = 01
Host = 10.0.100.11
afterSocket
Recebido resposta do servidor 10.0.100.11:1025
-->Inteiro 1, String 01
[as02@VM16 test-udp-c]$
```

Note que eu adicionei um print logo depois da criação do socket do cliente, só para testar

```

printf("Host = %s \n", argv[1]);
printf("Port = %s \n", argv[2]);
printf("IntVal = %s \n", argv[3]);
printf("StringVal = %s \n", argv[4]);

srv_host=argv[1];
printf("Host = %s \n", srv_host);

/* cria socket UDP do cliente. Porta selecionada pelo host */
sock=socket(AF_INET, SOCK_DGRAM, 0);
printf("\nafterSocket\n\n");
if (sock<0) {
    perror ("Nao consegui abrir socket tipo DATAGRAMA");
    exit(1);
}

hp=gethostbyname(srv_host);
if (((char *)hp) == NULL) {
    printf("%s: host desconhecido", srv_host);
    exit(2);
}

```

--  
Anny Caroline Correa Chagas

Alexandre Sztajnberg <[REDACTED]>  
 Para: Anny Caroline <[REDACTED]>  
 Cc: Alexandre Sztajnberg <[REDACTED]>

9 de julho de 2019 14:28

Valeu!  
 [Texto das mensagens anteriores oculto]