Aula 4

11 de novembro de 2020 11:16

Ex. 5:

Cliente.exe -ip 127.0.0.1 -port 6000 -msg Olá

Cliente.exe -port 6000 -ip 127.0.0.1 -msg Olá

Cliente.exe -msg Olá -port 6000 -ip 127.0.0.1

1 2 3 4 5 6 7 → argc = 7

argv[0] argv[1] argv[2] ... argv[6]

7

```
if(argc != 2){
        fprintf(stderr,"Sintaxe: %s frase_a_enviar\n",argv[0]);
        exit(EXIT_FAILURE);
    }
```

"Sintaxe: %s -ip ip-destino -port porto-destino -msg mensagem\n"

```
for(i=0; i<7; i++)
    se(comparar(argv[i], "-ip") == TRUE)
        ip_idx = i+1;
    se(comparar(argv[i], "-port") == TRUE)
        port_idx = i+1;
    se(comparar(argv[i], "-msg") == TRUE)
        msg_idx = i+1;
```

int

argv[ip_idx]

```
serv_addr.sin_addr.s_addr = inet_addr(SERV_HOST_ADDR); /*IP no formato "dotted decimal" => 32 bits*/
serv_addr.sin_port = htons(SERV_UDP_PORT); /*Host TO Netowork Short*/
```

htons (argv[port_idx])

converte de string para inteiro

```
msg_len = strlen(argv[1]);
if(sendto( sockfd , argv[1] , msg_len , 0 , (struct sockaddr*)&serv_addr , sizeof(serv_addr) ) == SOCKET_ERROR)
    Abort("SO nao conseguiu aceitar o datagram");
```

argv[msg_idx]

Ex. 6:

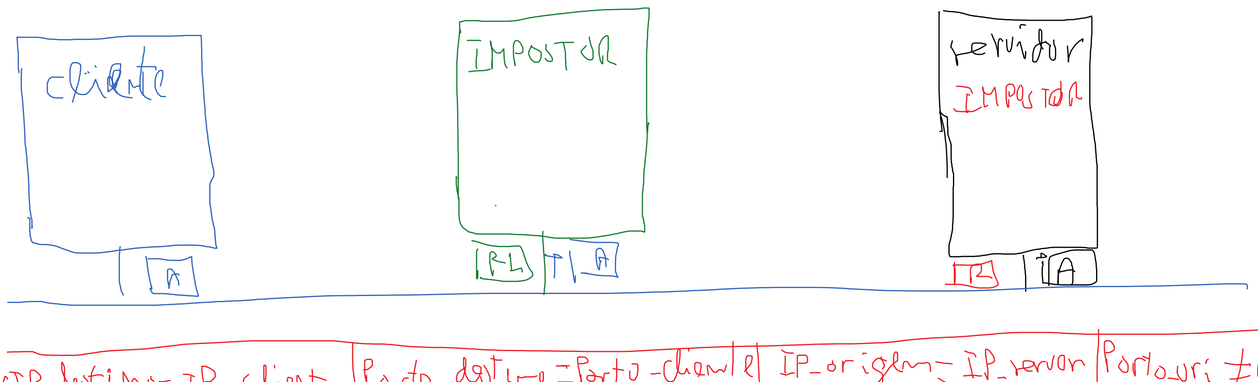
```
serv_check_len = sizeof( serv_check);
nbytes=recvfrom(sockfd , buffer , sizeof(buffer) , 0 , NULL , NULL);
```

&serv_check_len

int

(struct sockaddr*) &serv_check

struct sockaddr_in



se ((serv_check.sin_port==serv_addr.sin_port) && (comparar(inet_ntoa(serv_check.sin_addr), inet_ntoa(serv_addr.sin_addr)) == TRUE)

printf("Mensagem vem do servidor\n");

else printf("Mensagem vem de um impostor\n");