

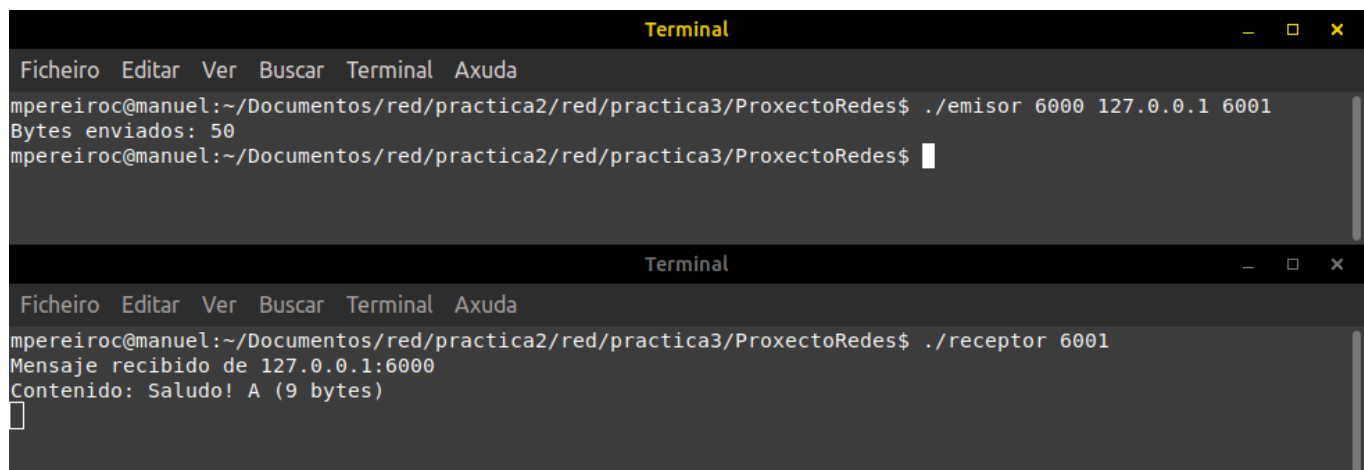
Informe

Informe Práctica3

Informe redactado por Manuel Pereiro Conde e Adrián Quiroga Linares

Exercicio 1.c

Enviamos unha mensaxe dende emisor.c que ocupa máis da mensaxe que é capaz de recibir o emisor, recibimos a mexaxe cortada. Non é posible recuperar os datos perdidos porque cada mensaxe enviada con sendto hay una única recepción con recvfrom.



```
Terminal
Ficheiro Editar Ver Buscar Terminal Axuda
mpereiroc@manuel:~/Documentos/red/practica2/red/practica3/ProxectoRedes$ ./emisor 6000 127.0.0.1 6001
Bytes enviados: 50
mpereiroc@manuel:~/Documentos/red/practica2/red/practica3/ProxectoRedes$

Terminal
Ficheiro Editar Ver Buscar Terminal Axuda
mpereiroc@manuel:~/Documentos/red/practica2/red/practica3/ProxectoRedes$ ./receptor 6001
Mensaje recibido de 127.0.0.1:6000
Contenido: Saludo! A (9 bytes)
```

Exercicio 1.d

Enviamos un array de floats, enviando cada float por separado empregando a función sprintf para convertir de float a string, cando acaba de enviar a string enviamos un \n para detectar o final da string. No receptor simplemente recibimos as strings e imprimimos os números, os bytes enviados e o numero de números recibidos.

```

float lista[5] = {1.1, 2.2, 3.3, 4.4, 5.5};
char mensaje[50];
for(int i = 0; i<5; i++){
    sprintf(mensaje, "%f", lista[i]);
    bytes = sendto(sockserv, mensaje, sizeof(float), 0, (struct sockaddr*) &iportRec, sizeof(iportRec));
    if (bytes < 0)
    {
        perror("Error al enviar el mensaje");
    }

    printf("Bytes enviados: %zd\n", bytes);
    sleep(1);
}

bytes = sendto(sockserv, "\n", sizeof(float), 0, (struct sockaddr*) &iportRec, sizeof(iportRec));
if (bytes < 0)
{
    perror("Error al enviar el mensaje");
}

printf("Bytes enviados: %zd\n", bytes);
close(sockserv);

```

```

do
{
    // Recibir mensaje
    bytesRecibidos = recvfrom(sockReceptor, bufferMensaje, sizeof(bufferMensaje) - 1, 0, (struct sockaddr *)&direccionEmisor, &tamanoEmisor);
    if (bytesRecibidos < 0)
    {
        perror("Error al recibir el mensaje");
        exit(EXIT_FAILURE);
    }

    bufferMensaje[bytesRecibidos] = '\0'; // Asegurar que el mensaje sea un string válido

    // Imprimir la IP, puerto y el mensaje recibido
    if (strcmp(bufferMensaje, "\n"))
    {
        printf("Mensaje recibido de %s:%d\n", inet_ntoa(direccionEmisor.sin_addr), ntohs(direccionEmisor.sin_port));
        printf("Contenido: %s (%d bytes)\n", bufferMensaje, bytesRecibidos);
        i++;
    }
} while (strcmp(bufferMensaje, "\n"));
printf("Números recibidos: %d.\n", i);
close(sockReceptor);
return 0;

```

```
Terminal
Ficheiro Editar Ver Buscar Terminal Axuda
mpereiroc@manuel:~/Documentos/red/practica2/red/practica3/ProyectoRedes$ ./emisor 6000 127.0.0.1 6001
Bytes enviados: 4
Bytes enviados: 4
Bytes enviados: 4
Bytes enviados: 4
Bytes enviados: 4
Bytes enviados: 4
mpereiroc@manuel:~/Documentos/red/practica2/red/practica3/ProyectoRedes$

Terminal
Ficheiro Editar Ver Buscar Terminal Axuda
mpereiroc@manuel:~/Documentos/red/practica2/red/practica3/ProyectoRedes$ ./receptor 6001
Mensaje recibido de 127.0.0.1:6000
Contenido: 1.10 (4 bytes)
Mensaje recibido de 127.0.0.1:6000
Contenido: 2.20 (4 bytes)
Mensaje recibido de 127.0.0.1:6000
Contenido: 3.30 (4 bytes)
Mensaje recibido de 127.0.0.1:6000
Contenido: 4.40 (4 bytes)
Mensaje recibido de 127.0.0.1:6000
Contenido: 5.50 (4 bytes)
Números recibidos: 5.
mpereiroc@manuel:~/Documentos/red/practica2/red/practica3/ProyectoRedes$
```

Exercicio

No cliente poñemos un sleep antes de empezar o bucle que envía a mensaxe e outro despois de envialo, permitindo así que soporte varios clientes ao mesmo tempo. O servidor procesa as mensaxes de forma simultánea.

```
Terminal
Ficheiro Editar Ver Buscar Terminal Axuda
mpereiroc@manuel:~/Documentos/red/practica2/red/practica3/ProyectoRedes$ ./receptor 6001
Mensaje recibido de 127.0.0.1:7002
Contenido convertido: LINEA 1
Mensaje recibido de 127.0.0.1:7002
Contenido convertido: LINEA 2
Mensaje recibido de 127.0.0.1:7002
Contenido convertido: LINEA 3
Mensaje recibido de 127.0.0.1:7002
Contenido convertido: ASDASD
Mensaje recibido de 127.0.0.1:7002
Contenido convertido: ASDASD
Mensaje recibido de 127.0.0.1:7002
Contenido convertido: LINEA 1
Mensaje recibido de 127.0.0.1:8002
Contenido convertido: ARCHIVO2 LINEA1
Mensaje recibido de 127.0.0.1:7002
Contenido convertido: LINEA 2
Mensaje recibido de 127.0.0.1:8002
Contenido convertido: ARCHIVO2 LINEA2
Mensaje recibido de 127.0.0.1:7002
Contenido convertido: LINEA 3
Mensaje recibido de 127.0.0.1:8002
Contenido convertido: SADASDSAD
Mensaje recibido de 127.0.0.1:7002
Contenido convertido: ASDASD
Mensaje recibido de 127.0.0.1:8002
Contenido convertido: ASDASDFG
Mensaje recibido de 127.0.0.1:7002
Contenido convertido: ASDASD
Mensaje recibido de 127.0.0.1:8002
Contenido convertido: ASDASD
Mensaje recibido de 127.0.0.1:8002
Contenido convertido: LDJFDIF

Terminal
Ficheiro Editar Ver Buscar Terminal Axuda
Socket UDP creado.
cliente asociado al puerto 7002.
Mensaje enviado: linea 1
Bytes: 1000
Mensaje recibido: LINEA 1
Bytes: 999
Mensaje enviado: linea 2
Bytes: 1000
Mensaje recibido: LINEA 2
Bytes: 999
Mensaje enviado: linea 3
Bytes: 1000
Mensaje recibido: LINEA 3
Bytes: 999
Mensaje enviado: asdasd
Bytes: 1000
Mensaje recibido: ASDASD
Bytes: 999
Mensaje enviado: asdasd
Bytes: 1000
Mensaje recibido: ASDASD
Bytes: 999
Mensaje enviado: asdasd
Bytes: 1000
Mensaje recibido: ASDASD
Bytes: 999
Fin de la transmisión.

Terminal
Ficheiro Editar Ver Buscar Terminal Axuda
mpereiroc@manuel:~/Documentos/red/practica2/red/practica3/ProyectoRedes$ ./clienteudp 127.0.0.1 7001 8002 archivo2.txt
Socket UDP creado.
cliente asociado al puerto 8002.
Mensaje enviado: archivo2 linea1
Bytes: 1000
Mensaje recibido: ARCHIVO2 LINEA1
Bytes: 999
Mensaje enviado: archivo2 linea2
Bytes: 1000
Mensaje recibido: ARCHIVO2 LINEA2
Bytes: 999
Mensaje enviado: sadasdsad
Bytes: 1000
Mensaje recibido: SADASDSAD
Bytes: 999
Mensaje enviado: asdasdfg
Bytes: 1000
Mensaje recibido: ASDASDFG
Bytes: 999
Mensaje enviado: ldjfdif
Bytes: 1000
Mensaje recibido: LDJFDIF
Bytes: 999
Fin de la transmisión.
mpereiroc@manuel:~/Documentos/red/practica2/red/practica3/ProyectoRedes$
```