Ejemplo Protocolo HTTP

El motivo de este ejemplo es ilustrar cómo se puede implementar un servidor HTTP básico utilizando el lenguaje de programación C. Este ejercicio busca demostrar el funcionamiento fundamental del protocolo HTTP, que es la base de la comunicación en la web, mediante la creación de un servidor que escucha en el puerto 80, acepta conexiones de clientes y les responde con una página HTML simple.

El protocolo HTTP, o Protocolo de Transferencia de Hipertexto, es un protocolo de la capa de aplicación que define la sintaxis y la semántica que deben utilizar los clientes web (como navegadores) y los servidores web para comunicarse entre sí. En este ejemplo, se utiliza un socket TCP, dado que HTTP es un protocolo orientado a la conexión que requiere una comunicación fiable entre el cliente y el servidor.

El servidor, una vez creado y configurado, entra en un bucle de espera donde acepta conexiones entrantes. Al recibir una conexión, responde con una página HTML que contiene un botón interactivo y un mensaje de bienvenida, demostrando así cómo se puede generar contenido dinámico y enviar respuestas HTTP desde un servidor escrito en C. Este ejemplo también destaca la facilidad con la que HTTP permite la transmisión de datos estructurados, como páginas web, y cómo se pueden manejar solicitudes básicas en un servidor a bajo nivel.

Para lograr este funcionamiento se debe de agregar el siguiente fragmento de código al archivo de ejemplohttp.c :

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <string.h>

int udp_socket, udp_cliente;
char Block[1000]="";
struct sockaddr_in servidor, cliente;

int main()
{

      int lbind, llisten;

      udp_socket = socket(AF_INET, SOCK_STREAM, 0);
      if(udp_socket==-1) {
            perror( "\nError al tratar de abrir socket...");
            exit(-1);
      }else{
                  perror( "\nExito al abrir socket...");
                  servidor.sin_family = AF_INET;
                  servidor.sin_port = htons(80);
                  servidor.sin_addr.s_addr = INADDR_ANY;
                  lbind = bind(udp_socket, (struct sockaddr *) &servidor,
sizeof(servidor) );
                  if (lbind==-1){
                  perror( "\nError en el bind");
```

```
                    exit(0);
              }else{
                    perror("\nExito en bind");
                    llisten = listen(udp_socket, 5);
                    if (llisten==-1){
                          perror("\nError en el listen");
                          exit(0);
                    }else{
                          perror("\nExito en el listen");
                               sprintf(Block, "HTTP/1.0 200 OK\r\n"
                               "Content-type: text/html\r\n"
                                  "Content-length: %d\r\n\r\n"
                                  "<!doctype html>"
                                  "<head>"
                                "<title>""HTTP""</title>"
                               "</head>"
                              "<body bgcolor=""orange"">"
                              "<h1>"
                              "Prueba practica HTTP"
                              "</h1>"
                              "<button type=button onclick=
document.getElementById('ejemplo').innerHTML=""Date()"">"
                              "Fecha y hora"
                              "</button>"
                              "<p id=""ejemplo"">"
                              "</p>"
                              "</body>"
                              "</html>",300);
                          while(1) {
                                      int size_cliente = sizeof(struct
sockaddr_in);
                                   udp_cliente = accept( udp_socket, (struct sockaddr
*) &cliente, &size_cliente);
                                      if (udp_cliente==-1){
                                      perror("\nError en el accept");
                                      exit(0);
                                      }else{
                                      perror("\nExito en el accept");
                                      write(udp_cliente,Block, strlen(Block));
                                      close(udp_cliente);
                                      }
                                }
                          close(udp_socket);
                          return 0;
                    }
              }
        }
}
```

Para poder probar el código se debe de compilar de la siguiente manera:

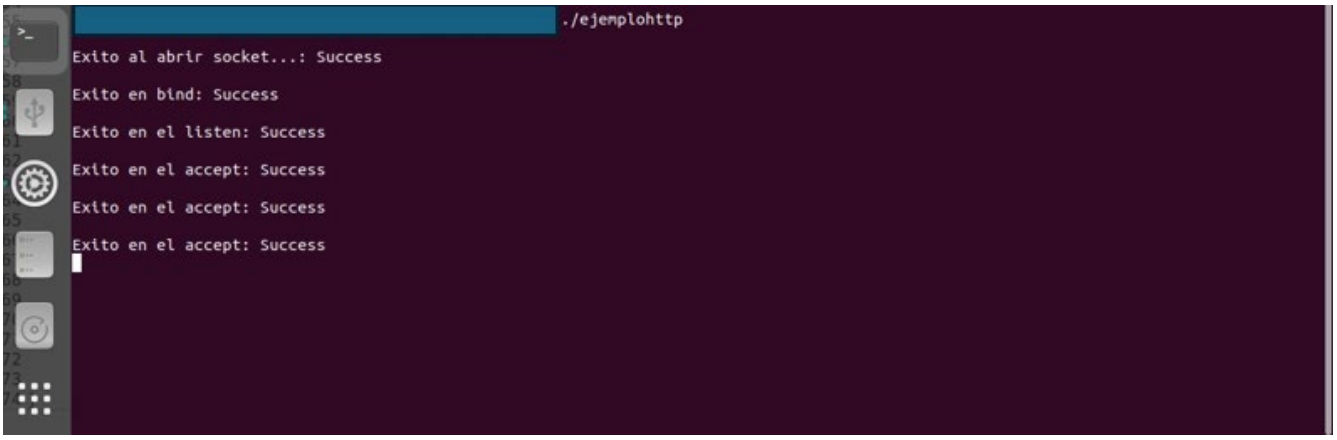
```
gcc ejemplohttp.c -o ejemplohttp
```

Suele denegar el acceso muchas veces por lo que recomiendo que en la terminal de Ubuntu se ingrese el siguiente comando:

```
Sudo su
```

Posterior a ello ejecutar con:

```
./ejemplohttp
```



Al consultar en el navegador accediendo con localhost:80 podremos ver la página que se esta invocando: