

# Program

## firewall.c

```
#include <netinet/in.h>
#include <errno.h>
#include <netdb.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <netinet/ip_icmp.h>
#include <netinet/udp.h>
#include <netinet/tcp.h>
#include <netinet/ip.h>
#include <netinet/if_ether.h>
#include <net/ethernet.h>
#include <sys/socket.h>
#include <arpa/inet.h>
#include <sys/ioctl.h>
#include <sys/time.h>
#include <sys/types.h>
#include <unistd.h>

void ProcessPacket(unsigned char *, int);
void print_ip_header(unsigned char *, int);
void print_tcp_packet(unsigned char *, int);
void print_udp_packet(unsigned char *, int);
void print_icmp_packet(unsigned char *, int);

struct sockaddr_in source, dest;
int tcp = 0, udp = 0, icmp = 0, others = 0, igmp = 0, total = 0, i, j;

int main()
{
    int saddr_size, data_size;
    struct sockaddr saddr;
    unsigned char *buffer = (unsigned char *)malloc(65536);
    printf("Starting...\n");

    int sock_raw = socket(AF_PACKET, SOCK_RAW, htons(ETH_P_ALL));
    printf("/nWeb socket active search something in the browser!");

    if (sock_raw < 0)
    {
        perror("Socket Error");
        return 1;
    }
    while (1)
    {
        saddr_size = sizeof saddr;
        data_size = recvfrom(sock_raw, buffer, 65536, 0, &saddr, (socklen_t*)&saddr_size);
```

```

        if (data_size < 0)
        {
            printf("Recvfrom error , failed to get packets\n");
            return 1;
        }
        ProcessPacket(buffer, data_size);
    }
    close(sock_raw);
    printf("Finished");
    return 0;
}

void ProcessPacket(unsigned char *buffer, int size)
{
    struct iphdr *iph = (struct iphdr *)(buffer + sizeof(struct ethhdr));
    ++total;
    switch (iph->protocol)
    {

    case 6:
        ++tcp;
        print_tcp_packet(buffer, size);
        break;

    case 17:
        ++udp;
        print_udp_packet(buffer, size);
        break;

    default:
        ++others;
        break;
    }
}

void print_ip_header(unsigned char *Buffer, int Size)
{
    unsigned short iphdrlen;

    struct iphdr *iph = (struct iphdr *)(Buffer + sizeof(struct ethhdr));
    iphdrlen = iph->ihl * 4;

    memset(&source, 0, sizeof(source));
    source.sin_addr.s_addr = iph->saddr;

    memset(&dest, 0, sizeof(dest));
    dest.sin_addr.s_addr = iph->daddr;

    printf("\n    Source IP          : %s\n", inet_ntoa(source.sin_addr));
}

void print_tcp_packet(unsigned char *Buffer, int Size)
{
    unsigned short iphdrlen;

```

```

    struct iphdr *iph = (struct iphdr *) (Buffer + sizeof(struct ethhdr));
    iphdrlen = iph->ihl * 4;

    struct tcphdr *tcph = (struct tcphdr *) (Buffer + iphdrlen +
    sizeof(struct ethhdr));

    int header_size = sizeof(struct ethhdr) + iphdrlen + tcph->doff * 4;
    int flag = 0;

    unsigned char *data = Buffer + header_size;
    for (i = 0; i < (Size - header_size); i++)
    {
        if ((data[i] == 99) && (data[i + 1] == 111) && (data[i + 2] == 114)
        && (data[i + 3] == 111) && (data[i + 4] == 110) && (data[i + 5] == 97))
        {
            printf("\nPacket contains information about corona. Be cautious
            of fake news. Stay safe.");
            flag = 1;
        }
        else if ((data[i] == 67) && (data[i + 1] == 79) && (data[i + 2] ==
        82) && (data[i + 3] == 79) && (data[i + 4] == 78) && (data[i + 5] == 65))
        {
            printf("\nPacket contains information about corona. Be cautious
            of fake news. Stay safe.");
            flag = 1;
        }

        else if ((data[i] == 67) && (data[i + 1] == 111) && (data[i + 2] ==
        114) && (data[i + 3] == 111) && (data[i + 4] == 110) && (data[i + 5] ==
        97))
        {
            printf("\nPacket contains information about corona. Be cautious
            of fake news. Stay safe.");
            flag = 1;
        }
    }
    if (flag == 1)
    {
        print_ip_header(Buffer, Size);
    }
}

void print_udp_packet(unsigned char *Buffer, int Size)
{

    unsigned short iphdrlen;

    struct iphdr *iph = (struct iphdr *) (Buffer + sizeof(struct ethhdr));
    iphdrlen = iph->ihl * 4;

    struct udphdr *udph = (struct udphdr *) (Buffer + iphdrlen +
    sizeof(struct ethhdr));

```

```

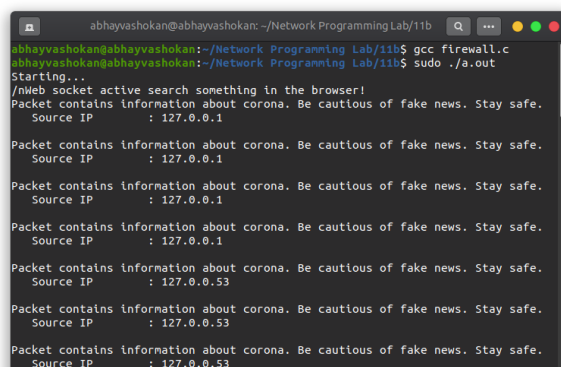
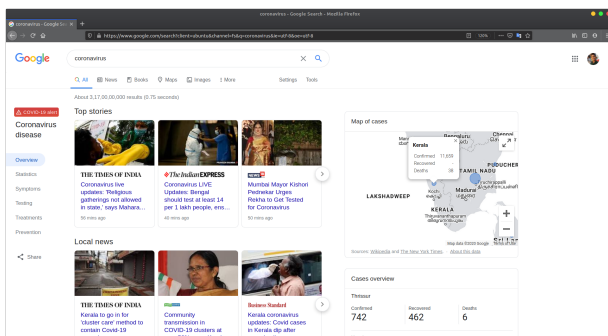
int header_size = sizeof(struct ethhdr) + iphdrlen + sizeof udph;
int flag = 0;

unsigned char *data = Buffer + header_size;
for (i = 0; i < (Size - header_size); i++)
{
    if ((data[i] == 99) && (data[i + 1] == 111) && (data[i + 2] == 114)
    && (data[i + 3] == 111) && (data[i + 4] == 110) && (data[i + 5] == 97))
    {
        printf("\nPacket contains information about corona. Be cautious
of fake news. Stay safe.");
        flag = 1;
    }
    else if ((data[i] == 67) && (data[i + 1] == 79) && (data[i + 2] ==
82) && (data[i + 3] == 79) && (data[i + 4] == 78) && (data[i + 5] == 65))
    {
        printf("\nPacket contains information about corona. Be cautious
of fake news. Stay safe.");
        flag = 1;
    }
    else if ((data[i] == 67) && (data[i + 1] == 111) && (data[i + 2] ==
114) && (data[i + 3] == 111) && (data[i + 4] == 110) && (data[i + 5] ==
97))
    {
        printf("\nPacket contains information about corona. Be cautious
of fake news. Stay safe.");
        flag = 1;
    }
}
if (flag == 1)
{
    print_ip_header(Buffer, Size);
}
}

```

## Output

### Screenshots



## Output

```
Starting...
Web socket active search something in the browser!
Packet contains information about corona. Be cautious of fake news. Stay
safe.
    Source IP          : 127.0.0.1

Packet contains information about corona. Be cautious of fake news. Stay
safe.
    Source IP          : 127.0.0.1

Packet contains information about corona. Be cautious of fake news. Stay
safe.
    Source IP          : 127.0.0.1

Packet contains information about corona. Be cautious of fake news. Stay
safe.
    Source IP          : 127.0.0.1

Packet contains information about corona. Be cautious of fake news. Stay
safe.
    Source IP          : 127.0.0.53

Packet contains information about corona. Be cautious of fake news. Stay
safe.
    Source IP          : 127.0.0.53

Packet contains information about corona. Be cautious of fake news. Stay
safe.
    Source IP          : 127.0.0.53
```

## ReadMe

1. Open first terminal
  1. `gcc firewall.c`
2. Open browser and search something related to coronavirus.
3. The packets containing the required information are recognized by the firewall.