

**Fr. Conceicao Rodrigues College of Engineering**

**Department of Computer Engineering**

**Academic Term : July-Nov 2023-24**

Class : T.E. (Computer B)

Subject Name: Computer network Lab

Subject Code : CSL 502

Experiment No:	8
Date of Performance:	14/9/2023
Roll No:	9614
Name of the Student:	Aqib Firdous Khan

**Objective of the Experiment:**

To find the subnet mask, the first address, the last address and the number of addresses from a given IP address.

**1. Theoretical description:**

A mask is a 32-bit number in which the  $n$  leftmost bits are 1s and the  $32-n$  rightmost bits are all 0s. It is convenient to give just the value of  $n$  preceded by a slash.

The address and the  $/n$  completely define the whole block. (first address, last address and the number of addresses)

The procedure to find the mask, the first address, the last address and the number of addresses is as follows:

**To find subnet mask:**

Set the  $n$  leftmost bits to 1s and the  $32-n$  rightmost bits to all 0s.

**To find the first address:**

Set the  $32-n$  rightmost bits in the binary notation of the address to 0s

OR

Bit by bit AND the given address with the mask.

**To find the last address:**

Set the  $32-n$  rightmost bits in the binary notation of the address to 1s.

OR

Bit by bit OR the given address with the complement of the mask.

**To find the number of addresses:**

The number of addresses can be found by using the formula  $2^{32-n}$ .

**3. Write a program to find the subnet mask, the first address, the last address and the number of addresses from a given IP address.**

CODE:

```
#include <stdio.h>
#include <stdint.h>
#include <arpa/inet.h>

int main() {
    char ip_address[16]; // Assuming IPv4 address format (xxx.xxx.xxx.xxx)
    int prefix_length;

    // Input IP address and subnet prefix length
    printf("Enter an IP address : ");
    scanf("%15s", ip_address);
    printf("Enter the subnet prefix length : ");
    scanf("%d", &prefix_length);

    // Calculate subnet mask
    uint32_t subnet_mask = 0xFFFFFFFFU << (32 - prefix_length);

    // Calculate first address
    struct in_addr ip;
    inet_aton(ip_address, &ip);
    uint32_t network_address = ntohl(ip.s_addr) & subnet_mask;

    // Calculate last address
    uint32_t broadcast_address = network_address | (~subnet_mask);

    // Calculate the number of addresses
    uint32_t num_addresses = broadcast_address - network_address + 1;

    // Print results
    printf("Subnet Mask: %s\n", inet_ntoa(*(struct in_addr*)&subnet_mask));
    printf("First Address: %s\n", inet_ntoa(*(struct in_addr*)&network_address));
    printf("Last Address: %s\n", inet_ntoa(*(struct in_addr*)&broadcast_address));
    printf("Number of Addresses: %u\n", num_addresses);

    return 0;
}
```

4. Attach the screenshot of the Program and output

## Output

```
/tmp/oSNJaY60Jv.o
```

```
Enter an IP address : 194.146.24.50
```

```
Enter the subnet prefix length : 25
```

```
Subnet Mask: 128.255.255.255
```

```
First Address: 0.24.146.194
```

```
Last Address: 127.24.146.194
```

```
Number of Addresses: 128
```

## 5. Conclusion

In conclusion, the program successfully calculates the subnet mask, first address, last address, and the number of addresses for the given IP address 194.146.24.50 with a subnet mask of /25. By performing the necessary bitwise operations, we have determined that the subnet mask is 255.255.255.128, the first address is 194.146.24.0, the last address is 194.146.24.127, and there are a total of 128 addresses in this subnet. This program provides valuable information for network administration and IP address management.

## 6. Post lab questions:

1. Explain NAT protocol in detail.

ANS: To access the Internet, one public IP address is needed, but we can use a private IP address in our private network. The idea of NAT is to allow multiple devices to access the Internet through a single public address. To achieve this, the translation of a private IP address to a public IP address is required. **Network Address Translation (NAT)** is a process in which one or more local IP address is translated into one or more Global IP address and vice versa in order to provide Internet access to the local hosts. Also, it does the translation of port numbers i.e. masks the port number of the host with another port number, in the packet that will be routed to the destination. It then makes the corresponding entries of IP address and port number in the NAT table. NAT generally operates on a router or firewall.

#### **Network Address Translation (NAT) working –**

Generally, the border router is configured for NAT i.e the router which has one interface in the local (inside) network and one interface in the global (outside) network. When a packet traverses outside the local (inside) network, then NAT converts that local (private) IP address to a global (public) IP address. When a packet enters the local network, the global (public) IP address is converted to a local (private) IP address.

If NAT runs out of addresses, i.e., no address is left in the pool configured then the packets will be dropped and an Internet Control Message Protocol (ICMP) host unreachable packet to the destination is sent.

#### **2. Differentiate between IPv4 and IPv6?**

ANS: **IPv4**

IPv4 address consists of two things that are the network address and the host address. It stands for **Internet Protocol version four**. It was introduced in 1981 by DARPA and was the first deployed version in 1982 for production on SATNET and on the ARPANET in January 1983.

IPv4 addresses are 32-bit integers that have to be expressed in Decimal Notation. It is represented by 4 numbers are separated by dots in the range of 0-255, which have to be converted to 0 and 1, to be understood by Computers. For Example, An IPv4 Address can be written as **189.123.123.90**.

#### **IPv4 Address Format**

IPv4 Address Format is a 32-bit Address that comprises binary digits separated by a dot (.).

#### **IPv6**

IPv6 is based on IPv4 and stands for Internet Protocol version 6. It was first introduced in December 1995 by Internet Engineering Task Force. IP version 6 is the new version of Internet Protocol, which is way better than IP version 4 in terms of complexity and efficiency. IPv6 is written as a group of 8 hexadecimal numbers separated by colon (:). It can be written as 128 bits of 0s and 1s.

#### **IPv6 Address Format**

IPv6 Address Format is a 128-bit IP Address, which is written in a group of 8 hexadecimal numbers separated by a colon (:).