# CSE 4512 [Computer Networks Lab]

# Lab # 02

## 1. Objectives:

- Understand the basics of IP Subnetting
- Learn to subnet a network following given specifications
- Understand Variable Length Subnet Mask (VLSM) addressing scheme
- Learn to design and implement VLSM in a network
- Get to know Secure Shell (SSH) and Telnet basics

## 2. Theory:

This lab assumes that you know binary arithmetic operations and have basic knowledge about IPv4 addressing scheme. If not, you're requested to go through your class lectures and online reference materials understand these concepts.

### IP Subnetting:

In the early days of networking, organizations could only use one network from the class A, B or C address space that were allocated to them. This resulted in a huge wastage of address space and newer demands for IP address could not be met properly. To overcome this problem, IP subnetting concept was introduced which enables splitting already existing larger networks into smaller networks. Use of subnetting freed up unnecessary allocation of IPv4 addresses and also made network management easier.

Now, let’s see how this subnetting actually works in practice. You know a IPv4 address consists of two portions: a Network part and a Host part. A *class A* address will have 8 bits for its network portion and rest 24 bits for its host portion. What we do in subnetting is we take some bits from the host portion and designate them as ***subnetwork*** portion and the rest of the bits are used for host addressing. Then the number of bits in the subnetwork portion (raised to the power of 2) will define the number of possible subnets. A bit-mask known as **subnet mask** is used to differentiate between the subnetwork and host portion. A subnet mask is a 32-bit mask which contains 1's in its most significant bits equal to the number of bits in the network+subnetwork portion of an address. The remaining bits in the subnet mask are zero. Put in another way, the number of 0's in a subnet mask is equal to the number of hosts in each subnet. A concrete example of IP subnetting is given below.

Let’s consider the network with IP address *172.16.0.0*. This is a *class B* IP address and we know for a class B address, first 16 bits are the Network portion and rest 16 bits are host portion. Now, let’s say, we want to have *2* subnets out of this bigger network. We'll then take 1 bit (as 2^1=2) from 16 bits host portion. Then host portion will have 15 bits in total which means each subnet will have *(2^15)-2* = 32,768-2 = 32,766 hosts. And the subnet mask will be *255.255.128.0*.

172.16.0.0 - 10101100 00010000 00000000 00000000 ⇦ Network Address

255.255.128.0 - 11111111 11111111 10000000 00000000 ⇦ Subnet Mask

Note that, a different notation known as CIDR (Classless Inter-Domain Routing) notation is also used to specify a subnet mask. This is basically the number of bits in network portion (including the subnetwork portion) of an IP address and is written with a / in front of it. So, for our running example, it would be written as 172.16.0.0/17 as there are 17 bits in network portion. For remainder portion of this lab handout, we will use this CIDR notation.

So, continuing on the above example, the two subnets will be *172.16.0.0/17* (subnet A) and *172.16.128.0/17* (subnet B). Each subnet will now function as an independent network. So, at the cost of reducing total number of hosts in a network, we've added another level of network to ease management and reduce IP address wastage. In a similar way, a bigger network can be subnetted to meet the number of host requirement. Now, that we've an understanding of how IP subnetting works, we are ready to understand VLSM (Variable Length Subnet Mask) addressing scheme which is a special form subnetting.

## VLSM:

Though we stated that subnet masking reduces IP address wastage, this is partially true. Subnetting scheme demonstrated above also suffers from IP address wastage albeit in a reduced manner than without subnetting. This is because in case of normal subnetting the same subnet mask is applied to all the subnets which results in same number of hosts in each subnet. This is not an efficient solution as some networks might have number of host requirements that don't *exactly* match the number of hosts obtained after applying same subnet masks in all subnets. All subnets might not utilize all hosts that result in address wastage.

To counter this issue, a subnet is further divided into more subnets and this method is known as variable length subnet masking because each subnet would have a different subnet mask (variable length) unlike what we saw in previous section. Now, let's consider a concrete example.

Suppose, we have been asked to subnet the network with address 204.15.5.0/24 to meet the following host requirements:

- netA: must support 14 hosts
- netB: must support 28 hosts
- netC: must support 2 hosts
- netD: must support 7 hosts
- netE: must support 28 host

The way to create subnets using VLSM technique is to take the largest host requirement first and assign it the appropriate subnet mask. Let's take netB first. It requires 28 hosts, so a subnet mask of /27 would meet its needs. Because /27 leaves 5 bits for host addressing that gives us 2^5 = 32 hosts. Note that, we always need to choose the nearest block that meets the given host requirement. In this case, a subnet mask of /28 would give us 2^4 = 16 host addresses which would fail to meet the given requirement and a mask of /26 would result in address wastage. So, if we follow similar approach as above for the other networks (in descending order of host requirements), we would reach the final subnetting as shown below:

netB: 204.15.5.0/27 host address range 1 to 30

netE: 204.15.5.32/27 host address range 33 to 62

netA: 204.15.5.64/28 host address range 65 to 78 netD: 204.15.5.80/28 host address range 81 to 94

netC: 204.15.5.96/30 host address range 97 to 98

Remember to increment the host portion of the calculated network addresses accordingly. So, following this technique, we can meet any kind of host specification with very little address wastage.

## Telnet and SSH:

Telnet is an application protocol for communication between two end devices which enables virtual access to a remote device. It's a bi-directional client-server protocol i.e., both sides can communicate interactively with one another. The standard TCP port for Telnet is 23. A user can log in to a remote server and interact with it using this protocol. Major drawback of this protocol is that all the communication happens in plain text which enables an attacker to see through any telnet communication by intercepting the packets and makes it possible to capture the password and get access to the remote device. For more on Telnet, you can further read here.

Secure Shell Protocol (SSH) have now become the application protocol of choice for communicating with remote resources as it provides an encrypted channel between the two ends. It also provides remote authentication and allows for remote administration. The standard TCP port for SSH is 22. In order for SSH to work, both the parties must agree on a common encryption technique. Further communication happens based on the agreed encryption technique. You can read more on SSH here.

## 3. Tasks:

I. You need to subnet a network following the given network specification and configure the devices properly following proper IP addressing. The task description is provided in the ***Task-1_subnet-an-ipv4-network*** pdf. You're also given a .pka file for this task.

II. The task description for this task is provided in the ***Task-2_ VLSM*** pdf. You'll have to subnet a given network topology following the network description provided in the pdf. You'll need to create the network topology by yourself as there's no .pka file provided for this task.

III. Create a copy of your completed .pka file from task I and rename it as "**Task3_ZZZ**" where ZZZ denotes last 3 digits of your student ID. Then following the steps in *Basic Telnet and SSH configuration* in section 3, configure ssh in the *CustomerRouter* device. You just need to customize one thing while doing this task. Change the **domain name** of the router to **ZZZ.com** where ZZZ is last 3 digits of your student ID.