Contents

Introduction

This document provides basic information needed in order to configure your router for routing IP, such as how addresses are broken down and how subnetting works. You learn how to assign each interface on the router an IP address with a unique subnet. There are examples included in order to help tie everything together.

Prerequisites

Requirements

Cisco recommends that you have a basic understanding of binary and decimal numbers.

Components Used

This document is not restricted to specific software and hardware versions.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Additional Information

If definitions are helpful to you, use these vocabulary terms in order to get you started:

- **Address -** The unique number ID assigned to one host or interface in a network.
- **Subnet -** A portion of a network that shares a particular subnet address.
- **Subnet mask -** A 32-bit combination used to describe which portion of an address refers to the subnet and which part refers to the host.
- **Interface -** A network connection.

If you have already received your legitimate address(es) from the Internet Network Information Center (InterNIC), you are ready to begin. If you do not plan to connect to the Internet, Cisco strongly suggests that you use reserved addresses from RFC 1918 ⧉ .

Understand IP Addresses

An IP address is an address used in order to uniquely identify a device on an IP network. The address is made up of 32 binary bits, which can be divisible into a network portion and host portion with the help of a subnet mask. The 32 binary bits are broken into four octets (1 octet = 8 bits). Each octet is converted to decimal and separated by a period (dot). For this reason, an IP address is said to be expressed in dotted decimal format (for example, 172.16.81.100). The value in each octet ranges from 0 to 255 decimal, or 00000000 - 11111111 binary.

Here is how binary octets convert to decimal: The right most bit, or least significant bit, of an octet holds a value of 2^0. The bit just to the left of that holds a value of 2^1. This continues until the left-most bit, or most significant bit, which holds a value of 2^7. So if all binary bits are a one, the decimal equivalent would be 255 as shown here:

```
  1  1  1  1 1 1 1 1
128 64 32 16 8 4 2 1 (128+64+32+16+8+4+2+1=255)
```

Here is a sample octet conversion when not all of the bits are set to 1.

```
 0  1 0 0 0 0 0 1
 0 64 0 0 0 0 0 1 (0+64+0+0+0+0+0+1=65)
```

And this sample shows an IP address represented in both binary and decimal.

```
      10.       1.       23.       19 (decimal)
 00001010.00000001.00010111.00010011 (binary)
```

These octets are broken down to provide an addressing scheme that can accommodate large and small networks. There are five different classes of networks, A to E. This document focuses on classes A to C, since classes D and E are reserved and discussion of them is beyond the scope of this document.

Note: Also note that the terms "Class A, Class B" and so on are used in this document in order to help facilitate the understanding of IP addressing and subnetting. These terms are rarely used in the industry anymore because of the introduction of classless interdomain routing (CIDR).

Given an IP address, its class can be determined from the three high-order bits. Figure 1 shows the significance in the three high order bits and the range of addresses that fall into each class. For informational purposes, Class D and Class E addresses are also shown.

Figure 1

In a Class A address, the first octet is the network portion, so the Class A example in [Figure 1] has a major network address of 1.0.0.0 - 127.255.255.255. Octets 2, 3, and 4 (the next 24 bits) are for the network manager to divide into subnets and hosts as he/she sees fit. Class A addresses are used for networks that have more than 65,536 hosts (actually, up to 16777214 hosts!).

In a Class B address, the first two octets are the network portion, so the Class B example in [Figure 1] has a major network address of 128.0.0.0 - 191.255.255.255. Octets 3 and 4 (16 bits) are for local subnets and hosts. Class B addresses are used for networks that have between 256 and 65534 hosts.

In a Class C address, the first three octets are the network portion. The Class C example in [Figure 1] has a major network address of 192.0.0.0 - 223.255.255.255. Octet 4 (8 bits) is for local subnets and hosts - perfect for networks with less than 254 hosts.

Network Masks

A network mask helps you know which portion of the address identifies the network and which portion of the address identifies the node. Class A, B, and C networks have default masks, also known as natural masks, as shown here:

```
Class A: 255.0.0.0
Class B: 255.255.0.0
Class C: 255.255.255.0
```

An IP address on a Class A network that has not been subnetted would have an address/mask pair similar to: 8.20.15.1 255.0.0.0. In order to see how the mask helps you identify the network and node parts of the address, convert the address and mask to binary numbers.

```
8.20.15.1 = 00001000.00010100.00001111.00000001
255.0.0.0 = 11111111.00000000.00000000.00000000
```

Once you have the address and the mask represented in binary, then identification of the network and host ID is easier. Any address bits which have corresponding mask bits set to 1 represent the network ID. Any address bits that have corresponding mask bits set to 0 represent the node ID.

```
8.20.15.1 = 00001000.00010100.00001111.00000001
255.0.0.0 = 11111111.00000000.00000000.00000000
            ----------------------------------
              net id |      host id

netid =  00001000 = 8
hostid = 00010100.00001111.00000001 = 20.15.1
```

Understand Subnetting

Subnetting allows you to create multiple logical networks that exist within a single Class A, B, or C network. If you do not subnet, you are only able to use one network from your Class A, B, or C network, which is unrealistic.

Each data link on a network must have a unique network ID, with every node on that link being a member of the same network. If you break a major network (Class A, B, or C) into smaller subnetworks, it allows you to create a network of interconnecting subnetworks. Each data link on this network would then have a unique network/subnetwork ID. Any device, or gateway, that connects *n* networks/subnetworks has *n* distinct IP addresses, one for each network / subnetwork that it interconnects.

In order to subnet a network, extend the natural mask with some of the bits from the host ID portion of the address in order to create a subnetwork ID. For example, given a Class C network of 204.17.5.0 which has a natural mask of 255.255.255.0, you can create subnets in this manner:

```
204.17.5.0 -      11001100.00010001.00000101.00000000
255.255.255.224 - 11111111.11111111.11111111.11100000
                  -------------------------|sub|----
```

By extending the mask to be 255.255.255.224, you have taken three bits (indicated by "sub") from the original host portion of the address and used them to make subnets. With these three bits, it is possible to create eight subnets. With the remaining five host ID bits, each subnet can have up to 32 host addresses, 30 of which can actually be assigned to a device *since host ids of all zeros or all ones are not allowed* (it is very important to remember this). So, with this in mind, these subnets have been created.

```
204.17.5.0 255.255.255.224      host address range 1 to 30
204.17.5.32 255.255.255.224     host address range 33 to 62
204.17.5.64 255.255.255.224     host address range 65 to 94
204.17.5.96 255.255.255.224     host address range 97 to 126
204.17.5.128 255.255.255.224    host address range 129 to 158
204.17.5.160 255.255.255.224    host address range 161 to 190
204.17.5.192 255.255.255.224    host address range 193 to 222
204.17.5.224 255.255.255.224    host address range 225 to 254
```

> **Note**: There are two ways to denote these masks. First, since you use three bits more than the "natural" Class C mask, you can denote these addresses as having a 3-bit subnet mask. Or, secondly, the mask of 255.255.255.224 can also be denoted as /27 as there are 27 bits that are set in the mask. This second method is used with CIDR. With this method, one of these networks can be described with the notation prefix/length. For example, 204.17.5.32/27 denotes the network 204.17.5.32 255.255.255.224. When appropriate, the prefix/length notation is used to denote the mask throughout the rest of this document.

The network subnetting scheme in this section allows for eight subnets, and the network might appear as:

Figure 2
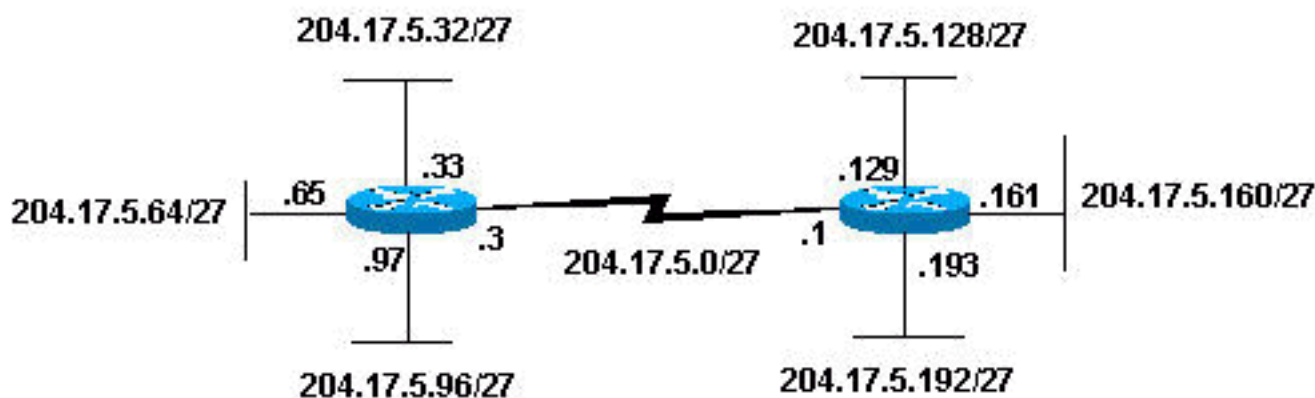
Notice that each of the routers in Figure 2 is attached to four subnetworks, one subnetwork is common to both routers. Also, each router has an IP address for each subnetwork to which it is attached. Each subnetwork could potentially support up to 30 host addresses.

This brings up an interesting point. The more host bits you use for a subnet mask, the more

subnets you have available. However, the more subnets available, the less host addresses available per subnet. For example, a Class C network of 204.17.5.0 and a mask of 255.255.255.224 (/27) allows you to have eight subnets, each with 32 host addresses (30 of which could be assigned to devices). If you use a mask of 255.255.255.240 (/28), the break down is:

```
204.17.5.0 -       11001100.00010001.00000101.00000000
255.255.255.240 - 11111111.11111111.11111111.11110000
                  -------------------------|sub |---
```

Since you now have four bits to make subnets with, you only have four bits left for host addresses. So in this case you can have up to 16 subnets, each of which can have up to 16 host addresses (14 of which can be assigned to devices).

Take a look at how a Class B network might be subnetted. If you have network 172.16.0.0 ,then you know that its natural mask is 255.255.0.0 or 172.16.0.0/16. Extending the mask to anything beyond 255.255.0.0 means you are subnetting. You can quickly see that you have the ability to create a lot more subnets than with the Class C network. If you use a mask of 255.255.248.0 (/21), how many subnets and hosts per subnet does this allow for?

```
172.16.0.0  -   10101100.00010000.00000000.00000000
255.255.248.0 - 11111111.11111111.11111000.00000000
                ----------------| sub |-----------
```

You use five bits from the original host bits for subnets. This allows you to have 32 subnets (2^5). After using the five bits for subnetting, you are left with 11 bits for host addresses. This allows each subnet so have 2048 host addresses (2^{11}), 2046 of which could be assigned to devices.

> **Note**: In the past, there were limitations to the use of a subnet 0 (all subnet bits are set to zero) and all ones subnet (all subnet bits set to one). Some devices would not allow the use of these subnets. Cisco Systems devices allow the use of these subnets when the **ip subnet zero** command is configured.

Examples

Sample Exercise 1

Now that you have an understanding of subnetting, put this knowledge to use. In this example, you are given two address / mask combinations, written with the prefix/length notation, which have been assigned to two devices. Your task is to determine if these devices are on the same subnet or different subnets. You can use the address and mask of each device in order to determine to which subnet each address belongs.

```
DeviceA: 172.16.17.30/20
DeviceB: 172.16.28.15/20
```
Determine the Subnet for DeviceA:

```
172.16.17.30  -   10101100.00010000.00010001.00011110
255.255.240.0 -   11111111.11111111.11110000.00000000
                  ----------------| sub|------------
subnet =          10101100.00010000.00010000.00000000 = 172.16.16.0
```

Looking at the address bits that have a corresponding mask bit set to one, and setting all the other address bits to zero (this is equivalent to performing a logical "AND" between the mask and address), shows you to which subnet this address belongs. In this case, DeviceA belongs to subnet 172.16.16.0.

Determine the Subnet for DeviceB:

```
172.16.28.15   -   10101100.00010000.00011100.00001111
255.255.240.0  -   11111111.11111111.11110000.00000000
                   ----------------| sub|-----------
subnet =           10101100.00010000.00010000.00000000 = 172.16.16.0
```

From these determinations, DeviceA and DeviceB have addresses that are part of the same subnet.

Sample Exercise 2

Given the Class C network of 204.15.5.0/24, subnet the network in order to create the network in Figure 3 with the host requirements shown.

Figure 3
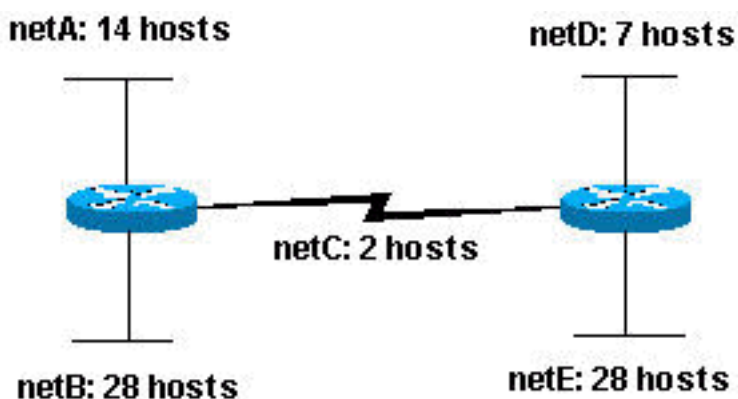
Looking at the network shown in Figure 3, you can see that you are required to create five subnets. The largest subnet must support 28 host addresses. Is this possible with a Class C network? and if so, then how?

You can start by looking at the subnet requirement. In order to create the five needed subnets you would need to use three bits from the Class C host bits. Two bits would only allow you four subnets (2^2).

Since you need three subnet bits, that leaves you with five bits for the host portion of the address. How many hosts does this support? $2^5 = 32$ (30 usable). This meets the requirement.

Therefore you have determined that it is possible to create this network with a Class C network. An example of how you might assign the subnetworks is:

```
netA: 204.15.5.0/27       host address range 1 to 30
netB: 204.15.5.32/27      host address range 33 to 62
netC: 204.15.5.64/27      host address range 65 to 94
netD: 204.15.5.96/27      host address range 97 to 126
netE: 204.15.5.128/27     host address range 129 to 158
```

VLSM Example

In all of the previous examples of subnetting, notice that the same subnet mask was applied for all the subnets. This means that each subnet has the same number of available host addresses. You can need this in some cases, but, in most cases, having the same subnet mask for all subnets ends up wasting address space. For example, in the Sample Exercise 2 section, a class C

network was split into eight equal-size subnets; however, each subnet did not utilize all available host addresses, which results in wasted address space. Figure 4 illustrates this wasted address space.
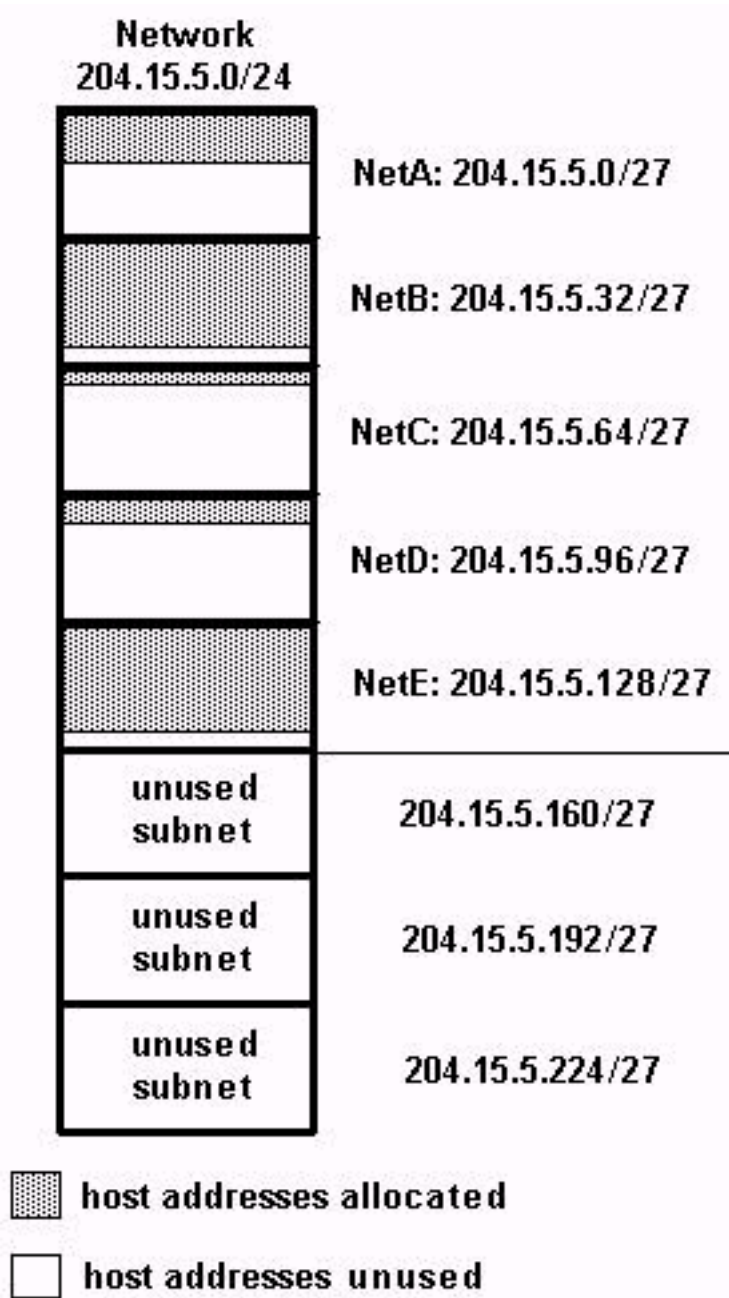
Figure 4

Figure 4 illustrates that of the subnets that are being used, NetA, NetC, and NetD have a lot of unused host address space. It is possible that this was a deliberate design accounting for future growth, but in many cases this is just wasted address space due to the fact that the same subnet mask is used for all the subnets.

Variable Length Subnet Masks (VLSM) allows you to use different masks for each subnet, thereby using address space efficiently.

VLSM Example

Given the same network and requirements as in Sample Exercise 2 develop a subnetting scheme with the use of VLSM, given:

```
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
```

Determine what mask allows the required number of hosts.

```
netA: requires a /28 (255.255.255.240) mask to support 14 hosts
netB: requires a /27 (255.255.255.224) mask to support 28 hosts
netC: requires a /30 (255.255.255.252) mask to support 2 hosts
netD*: requires a /28 (255.255.255.240) mask to support 7 hosts
netE: requires a /27 (255.255.255.224) mask to support 28 hosts

* a /29 (255.255.255.248) would only allow 6 usable host addresses
  therefore netD requires a /28 mask.
```

The easiest way to assign the subnets is to assign the largest first. For example, you can assign in this manner:

```
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
```

This can be graphically represented as shown in Figure 5:
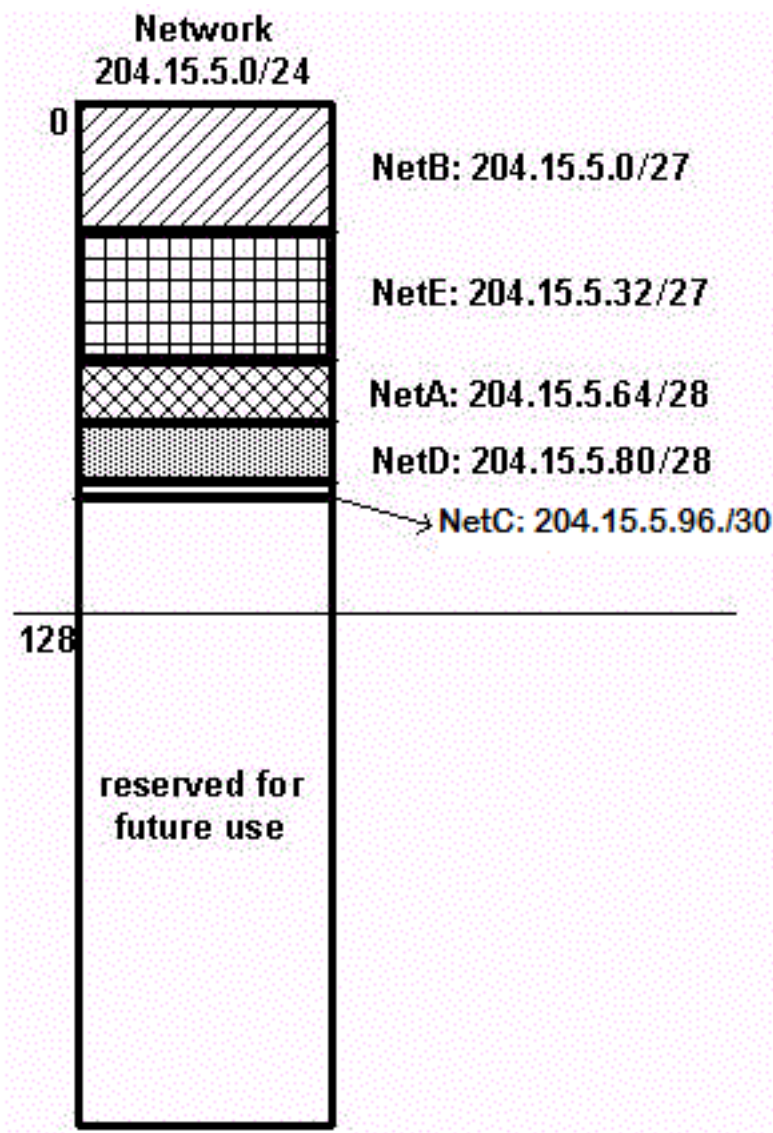
Figure 5

illustrates how using VLSM helped save more than half of the address space.

CIDR

Classless Interdomain Routing (CIDR) was introduced in order to improve both address space utilization and routing scalability in the Internet. It was needed because of the rapid growth of the Internet and growth of the IP routing tables held in the Internet routers.

CIDR moves way from the traditional IP classes (Class A, Class B, Class C, and so on). In CIDR , an IP network is represented by a prefix, which is an IP address and some indication of the length of the mask. Length means the number of left-most contiguous mask bits that are set to one. So network 172.16.0.0 255.255.0.0 can be represented as 172.16.0.0/16. CIDR also depicts a more hierarchical Internet architecture, where each domain takes its IP addresses from a higher level. This allows for the summarization of the domains to be done at the higher level. For example, if an ISP owns network 172.16.0.0/16, then the ISP can offer 172.16.1.0/24, 172.16.2.0/24, and so on to customers. Yet, when advertising to other providers, the ISP only needs to advertise 172.16.0.0/16.

For more information on CIDR, see RFC 1518 ↗ and RFC 1519 ↗ .

Appendix

Sample Config

Routers A and B are connected via serial interface.

Router A

```
hostname routera
!
ip routing
!
int e 0
ip address 172.16.50.1 255.255.255.0
!(subnet 50)
int e 1 ip address 172.16.55.1 255.255.255.0
!(subnet 55)
int s 0 ip address 172.16.60.1 255.255.255.0
!(subnet 60) int s 0
ip address 172.16.65.1 255.255.255.0 (subnet 65)
!S 0 connects to router B
router rip
network 172.16.0.0
```

Router B

```
hostname routerb
!
ip routing
!
int e 0
ip address 192.1.10.200 255.255.255.240
!(subnet 192)
int e 1
ip address 192.1.10.66 255.255.255.240
```

```
!(subnet 64)
int s 0
ip address 172.16.65.2 (same subnet as router A's s 0)
!Int s 0 connects to router A
router rip
network 192.1.10.0
network 172.16.0.0
```

Host/Subnet Quantities Table

```
Class B               Effective  Effective
# bits       Mask      Subnets     Hosts
-------  ---------------  ---------  ---------
  1      255.255.128.0        2      32766
  2      255.255.192.0        4      16382
  3      255.255.224.0        8       8190
  4      255.255.240.0       16       4094
  5      255.255.248.0       32       2046
  6      255.255.252.0       64       1022
  7      255.255.254.0      128        510
  8      255.255.255.0      256        254
  9      255.255.255.128    512        126
 10      255.255.255.192   1024         62
 11      255.255.255.224   2048         30
 12      255.255.255.240   4096         14
 13      255.255.255.248   8192          6
 14      255.255.255.252  16384          2

Class C               Effective  Effective
# bits       Mask      Subnets     Hosts
-------  ---------------  ---------  ---------
  1      255.255.255.128      2        126
  2      255.255.255.192      4         62
  3      255.255.255.224      8         30
  4      255.255.255.240     16         14
  5      255.255.255.248     32          6
  6      255.255.255.252     64          2


*Subnet all zeroes and all ones included. These
 might not be supported on some legacy systems.
*Host all zeroes and all ones excluded.
```

Related Information

- **IP Subnet Calculator (registered customers only**
- **IP Routing Protocols Technology Support**
- **Subnet Zero and the All-Ones Subnet**
- **Host and Subnet Quantities**
- **Technical Support & Documentation - Cisco Systems**