

Net Masks and IP Addresses

Professor Don Colton

Brigham Young University Hawaii

1 Introduction

IPv4 is currently the main addressing method on the Internet. Students who plan to use networking skills in their future careers should have some technical skill with this addressing method.

This tutorial introduces classful addressing and CIDR (classless) addressing, and explains in detail how to calculate subnets and address ranges.

2 Network, Subnet, Host

The first fundamental fact is that IPv4 addresses consist of 32 bits. These bits are divided into two or three parts, with the first part identifying the network and the last part identifying the host. The middle part, if any, identifies the CIDR subnet.

CIDR stands for Classless Inter-Domain Routing, and has the objective to get around the limitations of only three network sizes (Class A, B, and C) by measuring network sizes as some number of bits. This gives a lot more options.

Telephone numbers work much the same way. In Laie, my office phone number is (808) 293-3478. The (808) portion identifies my phone as being in Hawaii. The 293 portion identifies my phone as being in or near Laie. The 3478 portion identifies my exact phone line. We can say that (808) 293 is the network address, and 3478 is the host address.

(808-293) is the network. (3478) is the host.

BYUH actually controls the entire 3000 series of host addresses in this telephone network. We could call that a subnet. It is common for organizations to control a block of 10, 100, 1000, or more phone numbers.

(808-293) is the network. (3) is the subnet. (478) is the host.

3 32 Bits

With my telephone number, things are organized by “button pushes.” On the Internet, things are organized by bits. Take the following string of bits.

1011 1010 0100 1110 0101 0010 0000 0111

Generally we write IPv4 addresses in base 10. The first eight bits (1011 1010) are called the first octet. They translate to 186. The second eight bits (0100 1110) are called the second octet. They translate to 78. The third eight bits (0101 0010) are called the third octet. They translate to 82. The fourth eight bits (0000 0111) are called the fourth octet. They translate to 7. Generally we would write this address in “dotted-quad” notation, with each octet in base 10, like this.

186.78.82.7

4 Binary Numbers

The binary number system uses only two digits: zero and one. Often instead of calling them digits, they are called bits, which is short for bi(nary digi)ts. With this restriction, instead of counting 1, 2, 3, 4, 5, ... we count 1, 10, 11, 100, 101, ... It is a fully-functional numbering system, able to represent even numbers like π (3.1415) as 11.001 (with a bunch more digits). It takes about three times as many bits to express a number in binary as it does in base 10.

Data transmission on the Internet is generally thought of in terms of binary numbers because the underlying physical phenomena can be comfortably described in that way. Because the underlying nature is generally dealt with in binary, it is important for networking practitioners to also work in binary when appropriate.

4.1 Divide and Shift

Divide and Shift is one easy method to convert a number from base 10 to binary. Divide it by two re-

peatedly. Each time you divide, take the remainder and add it to the front of the binary number you are constructing. Here for example we will convert 2005 (the year this tutorial was written).

Taking 2005, we divide by two to get 1002.5. But let's keep it to whole numbers. Dividing we get 1002 with a remainder of 1. We copy the remainder to the front of the binary number we are building.

2005		our starting number
1002	1	divide and copy
501	01	divide and copy
250	101	continue
125	0101	continue
62	1 0101	continue
31	01 0101	continue
15	101 0101	continue
7	1101 0101	continue
3	1 1101 0101	continue
1	11 1101 0101	continue
0	111 1101 0101	continue
0	0111 1101 0101	okay, stop

We could keep putting zeroes at the front, but there is no point.

2005 in base 10 becomes 111,1101,0101 in base 2.

To convert a number from base 2 to base 10, do the same thing in reverse. Double the base 10 number and add the first bit of the base 2 number. We will convert our second octet from above.

0	0100 1110	our starting number
0	100 1110	double and add zero
1	00 1110	double and add one
2	0 1110	double and add zero
4	1110	double and add zero
9	110	double and add one
19	10	double and add one
39	0	double and add one
78		double and add zero

0100,1110 in base 2 becomes 78 in base 10.

4.2 Powers of Two

Powers of Two is another easy method to convert a number from base 10 to binary. We start with a table of powers of two. We subtract the largest power repeatedly. Then we add up the results. Again we will convert 2005.

1	1
2	10
4	100
8	1000
16	1 0000
32	10 0000
64	100 0000
128	1000 0000
256	1 0000 0000
512	10 0000 0000
1024	100 0000 0000
2048	1000 0000 0000

2005		our starting number
1024	100 0000 0000	largest power of two
981		after we subtract
512	10 0000 0000	largest power of two
469		after we subtract
256	1 0000 0000	largest power of two
213		after we subtract
128	1000 0000	largest power of two
85		after we subtract
64	100 0000	largest power of two
21		after we subtract
16	1 0000	largest power of two
5		after we subtract
4	100	largest power of two
1		after we subtract
1	1	largest power of two
0		done

1024	100 0000 0000
512	10 0000 0000
256	1 0000 0000
128	1000 0000
64	100 0000
16	1 0000
4	100
1	1
2005	111 1101 0101

Similarly we can add up the powers of two to get the answer in base 10.

78	0100 1110
2	10
4	100
8	1000
64	100 0000

$64 + 8 + 4 + 2 = 78$, our answer.

Students should be able to convert numbers using either method, and should be confident with at least one method.

4.3 Numbers to Memorize

To speed up calculations, it is helpful to memorize the sequence of powers of two, at least through 1024. When I say memorize the sequence, I don't mean that you remember that 64 is the sixth number, but rather that 64 comes after 32 and before 128. When we learn the alphabet, we often do not know that T is the 20th letter, but we know it is part of the sequence "Q R S T U V W."

It is also helpful to know the powers of two subtracted from 256, which are also the negative powers of two; a series of all ones followed by a series of all zeroes.

Negative Powers of Two

255	1111 1111	256-1
254	1111 1110	256-2
252	1111 1100	256-4
248	1111 1000	256-8
240	1111 0000	256-16
224	1110 0000	256-32
192	1100 0000	256-64
128	1000 0000	256-128

There are four numbers that are particularly important. They are 0^* , 0^*1 , 1^* , and 1^*0 , with any number of bits.

0^*	... 0000 0000 0000	all zeroes
0^*1	... 0000 0000 0001	zeroes and a one
1^*	... 1111 1111 1111	all ones
1^*0	... 1111 1111 1110	ones and a zero

5 Class A, B, C

We were working with an IP address:

1011 1010 0100 1110 0101 0010 0000 0111
or 186.78.82.7

By looking at the first three bits (101), we can tell that this is a Class B address. Class A addresses start with (0). Class B addresses start with (10). Class C addresses start with (110). Although this is easy to remember, it is probably easier to memorize the corresponding address range boundaries: 128, 192, and 224 (see the negative powers of two above).

Memorize the following facts:

For a Class A address, the first eight bits identify the network. The last twenty-four bits identify the subnet (if any) and host. In base 10, the addresses range from 0 to 127 in the first octet, but 0 and 127 themselves are reserved for special uses.

For a Class B address, the first sixteen bits identify the network. The last sixteen bits identify the subnet (if any) and host. In base 10, the addresses range from 128 to 191 in the first octet.

For a Class C address, the first twenty-four bits identify the network. The last eight bits identify the subnet (if any) and host. In base 10, the addresses range from 192 to 223 in the first octet.

6 Facts to Know

You must determine how many bits are involved with the network, the subnet, and the host. These always add up to 32.

The number of bits in the network portion is 8 for Class A, 16 for Class B, and 24 for Class C. Memorize it.

In the description that follows (n:h) means that n is the value in the network portion of the 32 bits, and h is the value in the host portion. (n:s:h) means that n is the value in the network portion of the 32 bits, s is the value in the subnet portion, and h is the value in the host portion.

The network mask is always (-1:0). The final meaning varies depending on whether it is a Class A, B, or C address.

A	1111 1111 0000 0000 0000 0000 0000 0000
B	1111 1111 1111 1111 0000 0000 0000 0000
C	1111 1111 1111 1111 1111 1111 0000 0000

In dotted quad notation, that translates to:

A	255.0.0.0
B	255.255.0.0
C	255.255.255.0

The number of IP addresses in the network is always 2^h , where h is the number of bits in the host portion of the address.

The number of host addresses in the network is always 2^h-2 , where h is the number of bits in the host portion of the address. Two IP addresses (0^* and 1^*) are reserved and cannot be used for hosts.

formula	description
(1*:0*)	network mask
(cc:0*)	network address
(0*:cc)	host component
(cc:0*1)	first host address
(cc:1*0)	last host address
(cc:1*)	broadcast address

The total number of bits in the subnet is given by the subnet mask, or by the number of one-bits in the subnet mask. For example, /21 has 21 one-bits, including the network bits (8 for Class A, etc.). The number in the subnet portion is “total minus network.”

The number of subnets in the network is always 2^s , where s is the number of bits in the subnet portion of the address.

The number of “valid” subnets in the network is always $2^s - 2$, where s is the number of bits in the subnet portion of the address. Two subnets (0 and -1) are reserved, but are widely used anyway.

formula	description
(1*:1*:0*)	subnet mask
(cc:0*:0*)	first invalid subnet
(cc:0*1:0*)	first valid subnet
(cc:1*0:0*)	last valid subnet
(cc:1*:0*)	last invalid subnet
(cc:cc:0*)	subnet address
(0*:0*:cc)	host portion of address
(cc:cc:1*)	broadcast address
(cc:cc:0*1)	first host address
(cc:cc:1*0)	last host address

7 Translating to Dotted Quad

In the above tables, only five numbers appear. They are:

cc: copy the original bits in that part of the address.

0*: all zeroes.

0*1: all zeroes with a one at the end.

1*: all ones.

1*0: all ones with a zero at the end.

Let’s translate the last valid subnet. According to the table, it is described as (cc:1*0:0*). We know what 1*0 and 0* mean. We need to know what the original bits are, and how many bits in each part of the address.

We will use our original address of 186.78.82.7, 1011 1010 0100 1110 0101 0010 0000 0111.

For a Class B address with a /22 subnet mask, the bit division is 16:6:10, meaning 16 bits in the network portion, 6 bits in the subnet portion, and 10 bits in the host portion.

Hence (cc:1*0:0*) means 16 bits from the original address (1011 1010 0100 1110), then 6 bits of 1*0 (1111 10), then 10 bits of 0* (00 0000 0000). The result is:

cc : 1*0 : 0*

1011 1010 0100 1110 : 1111 10 : 00 0000 0000

or in binary dotted quad:

1011 1010 . 0100 1110 . 1111 10 00 . 0000 0000

In dotted quad, the last valid /22 subnet is:

186.78.248.0

Now let’s change the subnet mask to /20 and do the same problem again.

For a Class B address with a /20 subnet mask, the bit division is 16:4:12, meaning 16 bits in the network portion, 4 bits in the subnet portion, and 12 bits in the host portion.

Hence (cc:1*0:0*) means 16 bits from the original address (1011 1010 0100 1110), then 4 bits of 1*0 (1110), then 12 bits of 0* (0000 0000 0000). The result is:

cc : 1*0 : 0*

1011 1010 0100 1110 : 1110 : 0000 0000 0000

or in binary dotted quad:

1011 1010 . 0100 1110 . 1110 0000 . 0000 0000

In dotted quad, the last valid /20 subnet is:

186.78.224.0

8 Practical Shortcuts

In most cases, the problems you are asked to solve are given in base 10 instead of binary. If you remember the tables above, all problems are trivial in binary. But converting from base 10 and back again can be tiresome and error-prone.

First, recognize that without subnetting, you never need to convert to and from binary. The only numbers that matter are cc, 0*, 0*1, 1*, and 1*0. In an eight-bit quad, those are cc, 0, 1, 255, and 254. No conversion is needed.

Second, recognize that even with subnetting, you never need to convert more than one quad to and from binary. If the subnet/host boundary falls in the middle of a quad, go ahead and convert it. The other three quads never need to be converted.

Let’s do the previous problem mostly in base 10. Start with 186.78.82.7/20.

The subnet/host division falls in the third quad.

For last valid subnet, we don't need the original in the third quad, so no need to convert to binary.

The first four bits are 1*0, and the last four bits are 0*, giving us 1110 0000.

1110 0000 converts back to 224.

In dotted quad, the last valid /20 subnet is:

186.78.224.0

If we need the original in the answer, we convert 82 to binary, giving us 0101 0010. Let's calculate the last IP address in the subnet.

cc : cc : 1*0

186 . 78 : (0101) : (1111) . 254

Converting 0101 1111 to base 10, we get 95, for a final answer of:

186.78.95.254

9 Quick Reference

The following page gives all the formulas and tables for quick reference. You may want to memorize all or part of this information.

Powers of Two

1	1
2	10
4	100
8	1000
16	1 0000
32	10 0000
64	100 0000
128	1000 0000
256	1 0000 0000
512	10 0000 0000
1024	100 0000 0000

Negative Powers of Two

255	1111 1111	256-1
254	1111 1110	256-2
252	1111 1100	256-4
248	1111 1000	256-8
240	1111 0000	256-16
224	1110 0000	256-32
192	1100 0000	256-64
128	1000 0000	256-128

Four Important Numbers

There are four numbers that are particularly important. They are 0^* , 0^*1 , 1^* , and 1^*0 , with any number of bits.

0^*	... 0000 0000 0000	all zeroes
0^*1	... 0000 0000 0001	zeroes and a one
1^*	... 1111 1111 1111	all ones
1^*0	... 1111 1111 1110	ones and a zero

Class A, B, C

Class	network bits	range
A	0...	0 to 127
B	10...	128 to 191
C	110...	192 to 223

Network masks:
mask bits

A	1111 1111 0000 0000 0000 0000 0000 0000
B	1111 1111 1111 1111 0000 0000 0000 0000
C	1111 1111 1111 1111 1111 1111 0000 0000

In dotted quad notation, that translates to:

Class	CIDR	mask
A	/8	255.0.0.0
B	/16	255.255.0.0
C	/24	255.255.255.0

The total number of bits in the subnet is given by the subnet mask, or by the number of one-bits

in the subnet mask. For example, /21 has 21 one-bits, including the network bits (8 for Class A, etc.). The number in the subnet portion is “total minus network.”

Address Formulas

The order is network:subnet:host. In some cases there is no subnet.

formula	description
2^h	IP address count
2^h-2	host address count
2^s	subnet count
2^h-2	valid subnet count
$(1^*:0^*:0^*)$	network mask
$(cc:0^*:0^*)$	network address
$(0^*:cc:cc)$	host component
$(cc:0^*:1^*)$	first host address
$(cc:1^*:1^*0)$	last host address
$(cc:1^*:1^*)$	broadcast address
$(1^*:1^*:0)$	subnet mask
$(cc:0^*:0^*)$	first invalid subnet
$(cc:0^*1:0^*)$	first valid subnet
$(cc:1^*0:0^*)$	last valid subnet
$(cc:1^*:0^*)$	last invalid subnet
$(cc:cc:0^*)$	subnet address
$(0^*:0^*:cc)$	host portion of address
$(cc:cc:1^*)$	broadcast address
$(cc:cc:0^*1)$	first host address
$(cc:cc:1^*0)$	last host address