An IP (Internet Protocol) address is a unique identifier for a node or host connection on an IP network. An IP address is a 32 bit binary number usually represented as 4 decimal values, each representing 8 bits, in the range 0 to 255 (known as octets) separated by decimal points. This is known as "dotted decimal" notation.

Example**: 140.179.220.200**

It is sometimes useful to view the values in their binary form.

140 .179 .220 .200

10001100.10110011.11011100.11001000

Every IP address consists of two parts, one identifying the network and one identifying the node. The Class of the address and the subnet mask determine which part belongs to the network address and which part belongs to the node address.

**Address Classes**

There are 5 different address classes. You can determine which class any IP address is in by examining the first 4 bits of the IP address.

* **Class A** addresses begin with **0xxx**, or **1 to 126** decimal.
* **Class B** addresses begin with **10xx**, or **128 to 191** decimal.
* **Class C** addresses begin with **110x**, or **192 to 223** decimal.
* **Class D** addresses begin with **1110**, or **224 to 239** decimal.
* **Class E** addresses begin with **1111**, or **240 to 254** decimal.

Addresses beginning with **01111111**, or **127** decimal, are reserved for loopback and for internal testing on a local machine; [You can test this: you should always be able to ping **127.0.0.1**, which points to yourself] Class D addresses are reserved for multicasting; Class E addresses are reserved for future use. They should not be used for host addresses.

Now we can see how the Class determines, by default, which part of the IP address belongs to the network (N, in blue) and which part belongs to the node (n, in red).

* Class A -- NNNNNNNN.nnnnnnnn.nnnnnnnn.nnnnnnnn
* Class B -- NNNNNNNN.NNNNNNNN.nnnnnnnn.nnnnnnnn
* Class C -- NNNNNNNN.NNNNNNNN.NNNNNNNN.nnnnnnnn

In the example, 140.179.220.200 is a Class B address so by default the Network part of the address (also known as the *Network Address*) is defined by the first two octets (140.179.x.x) and the node part is defined by the last 2 octets (x.x.220.200).

In order to specify the network address for a given IP address, the node section is set to all "0"s. In our example, 140.179.0.0 specifies the network address for 140.179.220.200. When the node section is set to all "1"s, it specifies a broadcast that is sent to all hosts on the network. 140.179.255.255 specifies the example broadcast address. Note that this is true regardless of the length of the node section.

**Private Subnets**

There are three IP network addresses reserved for private networks. The addresses are **10.0.0.0, Subnet Mask 255.0.0.0**, **172.16.0.0, Subnet Mask 255.240.0.0**, and **192.168.0.0, Subnet Mask 255.255.0.0.**These addresses are also notated **10.0.0.0/8**, **172.16.0.0/12**, and **192.168.0.0/16**; this notation will be explained later in this tutorial. They can be used by anyone setting up internal IP networks, such as a lab or home LAN behind a NAT or proxy server or a router. It is always safe to use these because routers on the Internet by default will never forward packets coming from these addresses. These addresses are defined in [RFC 1918](http://www.faqs.org/rfcs/rfc1918.html).

Subnetting an IP Network can be done for a variety of reasons, including organization, use of different physical media (such as Ethernet, FDDI, WAN, etc.), preservation of address space, and security. The most common reason is to control network traffic. In an Ethernet network, all nodes on a segment see all the packets transmitted by all the other nodes on that segment. Performance can be adversely affected under heavy traffic loads, due to collisions and the resulting retransmissions. A router is used to connect IP networks to minimize the amount of traffic each segment must receive.

**Subnet Masking**

Applying a subnet mask to an IP address allows you to identify the network and node parts of the address. The network bits are represented by the 1s in the mask, and the node bits are represented by the 0s. Performing a bitwise [logical AND](http://www.ralphb.net/IPSubnet/logical.html) operation between the IP address and the subnet mask results in the *Network Address* or Number.   
For example, using our test IP address and the default Class B subnet mask, we get:

10001100.10110011.11110000.11001000 140.179.240.200 Class B IP Address

11111111.11111111.00000000.00000000 255.255.000.000 Default Class B Subnet Mask

--------------------------------------------------------

10001100.10110011.00000000.00000000 140.179.000.000 Network Address

Default subnet masks:

* **Class A** - 255.0.0.0 - 11111111.00000000.00000000.00000000
* **Class B** - 255.255.0.0 - 11111111.11111111.00000000.00000000
* **Class C** - 255.255.255.0 - 11111111.11111111.11111111.00000000
* Additional bits can be added to the default subnet mask for a given Class to further subnet, or break down, a network. When a bitwise [logical AND](http://www.ralphb.net/IPSubnet/logical.html) operation is performed between the subnet mask and IP address, the result defines the *Subnet Address* (also called the *Network Address* or *Network Number*). There are some restrictions on the subnet address. Node addresses of all "0"s and all "1"s are reserved for specifying the local network (when a host does not know its network address) and all hosts on the network (broadcast address), respectively. This also applies to subnets. A subnet address cannot be all "0"s or all "1"s. This also implies that a 1 bit subnet mask is not allowed. This restriction is required because older standards enforced this restriction. Recent standards that allow use of these subnets have superseded these standards, but many "legacy" devices do not support the newer standards. If you are operating in a controlled environment, such as a lab, you can safely use these restricted subnets.
* To calculate the number of subnets or nodes, use the formula (2n-2) where n = number of bits in either field, and 2n represents 2 raised to the nth power. Multiplying the number of subnets by the number of nodes available per subnet gives you the total number of nodes available for your class and subnet mask. Also, note that although subnet masks with non-contiguous mask bits are allowed, they are not recommended.
* Example:
* 10001100.10110011.11011100.11001000 140.179.220.200 IP Address
* 11111111.11111111.**111**00000.00000000 255.255.**224**.000 Subnet Mask
* --------------------------------------------------------
* 10001100.10110011.11000000.00000000 140.179.192.000 Subnet Address
* 10001100.10110011.11011111.11111111 140.179.223.255 Broadcast Address
* In this example a **3 bit subnet mask** was used. There are 6 (23-2) subnets available with this size mask (remember that subnets with all 0's and all 1's are not allowed). Each subnet has 8190 (213-2) nodes. Each subnet can have nodes assigned to any address between the Subnet address and the Broadcast address. This gives a total of 49,140 nodes for the entire class B address subnetted this way. Notice that this is less than the 65,534 nodes an unsubnetted class B address would have.
* You can calculate the Subnet Address by performing a bitwise [logical AND](http://www.ralphb.net/IPSubnet/logical.html) operation between the IP address and the subnet mask, then setting all the host bits to **0**s. Similarly, you can calculate the *Broadcast Address* for a subnet by performing the same [logical AND](http://www.ralphb.net/IPSubnet/logical.html) between the IP address and the subnet mask, then setting all the host bits to **1**s. That is how these numbers are derived in the example above.
* Subnetting always reduces the number of possible nodes for a given network. There are complete subnet tables available here for [Class A](http://www.ralphb.net/IPSubnet/class_a.html), [Class B](http://www.ralphb.net/IPSubnet/class_b.html) and [Class C](http://www.ralphb.net/IPSubnet/class_c.html). These tables list all the possible subnet masks for each class, along with calculations of the number of networks, nodes and total hosts for each subnet.
* Here is another, more detailed, example. Say you are assigned a Class C network number of 200.133.175.0 (apologies to anyone who may actually own this domain address). You want to utilize this network across multiple small groups within an organization. You can do this by subnetting that network with a subnet address.
* We will break this network into 14 subnets of 14 nodes each. This will limit us to 196 nodes on the network instead of the 254 we would have without subnetting, but gives us the advantages of traffic isolation and security. To accomplish this, we need to use a subnet mask 4 bits long.   
  Recall that the default Class C subnet mask is
* 255.255.255.0 (11111111.11111111.11111111.00000000 binary)
* Extending this by 4 bits yields a mask of
* 255.255.255.**240** (11111111.11111111.11111111.**1111**0000 binary)
* This gives us 16 possible network numbers, 2 of which cannot be used:

|  |  |  |  |
| --- | --- | --- | --- |
| **Subnet bits** | **Network Number** | **Node Addresses** | **Broadcast Address** |
| 0000 | 200.133.175.0 | Reserved | None |
| 0001 | 200.133.175.16 | .17 thru .30 | 200.133.175.31 |
| 0010 | 200.133.175.32 | .33 thru .46 | 200.133.175.47 |
| 0011 | 200.133.175.48 | .49 thru .62 | 200.133.175.63 |
| 0100 | 200.133.175.64 | .65 thru .78 | 200.133.175.79 |
| 0101 | 200.133.175.80 | .81 thru .94 | 200.133.175.95 |
| 0110 | 200.133.175.96 | .97 thru .110 | 200.133.175.111 |
| 0111 | 200.133.175.112 | .113 thru .126 | 200.133.175.127 |
| 1000 | 200.133.175.128 | .129 thru .142 | 200.133.175.143 |
| 1001 | 200.133.175.144 | .145 thru .158 | 200.133.175.159 |
| 1010 | 200.133.175.160 | .161 thru .174 | 200.133.175.175 |
| 1011 | 200.133.175.176 | .177 thru .190 | 200.133.175.191 |
| 1100 | 200.133.175.192 | .193 thru .206 | 200.133.175.207 |
| 1101 | 200.133.175.208 | .209 thru .222 | 200.133.175.223 |
| 1110 | 200.133.175.224 | .225 thru .238 | 200.133.175.239 |
| 1111 | 200.133.175.240 | Reserved | None |

Now that you understand "classful" IP Subnetting principals, you can forget them ;). The reason is **CIDR** -- **C**lassless **I**nter**D**omain **R**outing. CIDR was invented several years ago to keep the internet from running out of IP addresses. The "classful" system of allocating IP addresses can be very wasteful; anyone who could reasonably show a need for more that 254 host addresses was given a Class B address block of 65533 host addresses. Even more wasteful were companies and organizations that were allocated Class A address blocks, which contain over 16 Million host addresses! Only a tiny percentage of the allocated Class A and Class B address space has ever been actually assigned to a host computer on the Internet.

People realized that addresses could be conserved if the class system was eliminated. By accurately allocating only the amount of address space that was actually needed, the address space crisis could be avoided for many years. This was first proposed in 1992 as a scheme called **Supernetting**. Under supernetting, the classful subnet masks are extended so that a network address and subnet mask could, for example, specify multiple Class C subnets with one address. For example, If I needed about 1000 addresses, I could supernet 4 Class C networks together:

192.60.128.0 (11000000.00111100.10000000.00000000) Class C subnet address

192.60.129.0 (11000000.00111100.10000001.00000000) Class C subnet address

192.60.130.0 (11000000.00111100.10000010.00000000) Class C subnet address

192.60.131.0 (11000000.00111100.10000011.00000000) Class C subnet address

--------------------------------------------------------

192.60.128.0 (11000000.00111100.10000000.00000000) Supernetted Subnet address

255.255.252.0 (11111111.11111111.11111100.00000000) Subnet Mask

192.60.131.255 (11000000.00111100.10000011.11111111) Broadcast address

In this example, the subnet 192.60.128.0 includes all the addresses from 192.60.128.0 to 192.60.131.255. As you can see in the binary representation of the subnet mask, the Network portion of the address is 22 bits long, and the host portion is 10 bits long.

Under CIDR, the subnet mask notation is reduced to a simplified shorthand. Instead of spelling out the bits of the subnet mask, it is simply listed as the number of 1s bits that start the mask. In the above example, instead of writing the address and subnet mask as

192.60.128.0, Subnet Mask 255.255.252.0

the network address would be written simply as:

192.60.128.0/22

which indicates starting address of the network, and number of 1s bits (22) in the network portion of the address. If you look at the subnet mask in binary (**11111111**.**11111111**.**111111**00.00000000), you can easily see how this notation works.

The use of a CIDR notated address is the same as for a Classful address. Classful addresses can easily be written in CIDR notation (Class A = /8, Class B = /16, and Class C = /24)

It is currently almost impossible for an individual or company to be allocated their own IP address blocks. You will simply be told to get them from your ISP. The reason for this is the ever-growing size of the internet routing table. Just 10 years ago, there were less than 5000 network routes in the entire Internet. Today, there are over 100,000. Using CIDR, the biggest ISPs are allocated large chunks of address space (usually with a subnet mask of /19 or even smaller); the ISP's customers (often other, smaller ISPs) are then allocated networks from the big ISP's pool. That way, all the big ISP's customers (and their customers, and so on) are accessible via 1 network route on the Internet. But I digress.

It is expected that CIDR will keep the Internet happily in IP addresses for the next few years at least. After that, IPv6, with 128 bit addresses, will be needed. Under IPv6, even sloppy address allocation would comfortably allow a billion unique IP addresses for every person on earth! The complete and gory details of CIDR are documented in [RFC1519](http://www.faqs.org/rfcs/rfc1519.html), which was released in September of 1993.