IP Sub−Networking Mini−Howto

Table of Contents

IP Sub−Networking Mini−Howto

Robert Hart, `hartr@interweft.com.au`

v1.0, 31 March 1997

This document describes why and how to subnetwork an IP network − that is using a single A, B or C Class network number to function correctly on several interconnected networks.

7.Routing

1.Copyright

This document is directly supported by InterWeft IT Consultants (Melbourne, Australia).

The latest version of this document is available at the InterWeft WWW site at InterWeft IT Consultants and from The Linux Documentation Project.

2.Introduction

With available IP network numbers rapidly becoming an endangered species, efficient use of this increasingly scarce resource is important.

This document describes how to split a single IP network number up so that it can be used on several different networks.

This document concentrates on C Class IP network numbers − but the principles apply to A and B class networks as well.

2.1 Other sources of information

There are a number of other sources of information that are of relevance for both detailed and background information on IP numbers. Those recommended by the author are:−

- The Linux Network Administrators Guide.
- The Linux System Administration Guide.
- TCP/IP Network Administration by Craig Hunt, published by O'Reilly and Associates.

3.The Anatomy of IP numbers

Before diving into the delight of sub−networking, we need to establish some IP number basics.

3.1 IP numbers belong to Interfaces − NOT hosts!

First of all, let's clear up a basic cause of misunderstanding − IP numbers are **not** assigned to hosts. IP numbers are assigned to network interfaces on hosts.

Eh − what's that?

Whilst many (if not most) computers on an IP network will possess a single network interface (and have a single IP number as a consequence), this is not the only way things happen. Computers and other devices can have several (if not many) network interfaces − and each interface has its own IP number.

So a device with 6 active interfaces (such as a router) will have 6 IP numbers − one for each interface to each network to which it is connected. The reason for this becomes clear when we look at an IP network!

Despite this, most people refer to *host addresses* when referring to an IP number. Just remember, this is simply shorthand for *the IP number of this particular interface on this host*. Many (if not the majority) of devices on the Internet have only a single interface and thus a single IP number.

3.2 IP Numbers as "Dotted Quads"

In the current (IPv4) implementation of IP numbers, IP numbers consist of 4 (8 bit) bytes − giving a total of 32 bits of available information. This results in numbers that are rather large (even when written in decimal notation). So for readability (and organisational reasons) IP numbers are usually written in the 'dotted quad' format. The IP number

```
192.168.1.24
```

is an example of this − 4 (decimal) numbers separated by (.) dots.

As each one of the four numbers is the decimal representation of an 8 bit byte, each of the 4 numbers can range from 0 to 255 (that is take on 256 unique values − remember, zero is a value too).

In addition, part of the IP number of a host identifies the network on which the host resides, the remaining 'bits' of the IP number identify the host (oops − network interface) itself. Exactly how many bits are used by the network ID and how many are available to identify hosts (interfaces) on that network is determined by the network 'class'.

3.3 Classes of IP Networks

There are three classes of IP numbers

- Class A IP network numbers use the leftmost 8 bits (the leftmost of the dotted quads) to identify the network, leaving 24 bits (the remaining three dotted quads) to identify host interfaces on that network. Class A addresses **always** have the leftmost bit of the leftmost byte a zero − that is a decimal value of 0 to 127 for the first dotted quad. So there are a maximum of 128 class A network numbers available, with each one containing up to 33,554,430 possible interfaces.
  However, the networks 0.0.0.0 (known as the default route) and 127.0.0.0 (the loop back network) have special meanings and are not available for use to identify networks. So there are only 126 *available* A class network numbers.
- Class B IP network numbers use the leftmost 16 bits (the leftmost two dotted quads) to identify the network, leaving 16 bits (the last two dotted quads) to identify host interfaces. Class B addresses always have the leftmost 2 bits of the leftmost byte set to 1 0. This leaves 14 bits left to specify the network address giving 32767 available B class networks. B Class networks thus have a range of 128 to 191 for the first of the dotted quads, with each network containing up to 32,766 possible interfaces.
- Class C IP network numbers use the leftmost 24 bits (the leftmost three bytes) to identify the network, leaving 8 bits (the rightmost byte) to identify host interfaces. Class C addresses always start with the leftmost 3 bits set to 1 1 0 or a range of 192 to 255 for the leftmost dotted quad. There are thus 4,194,303 available C class network numbers, each containing 254 interfaces. (C Class networks with the first byte greater than 223 are however reserved and unavailable for use).

In summary:

```
Network class   Usable range of first byte values (decimal)
      A                 1 to 126
      B               128 to 191
      C               192 to 254
```

There are also special addresses that are reserved for 'unconnected' networks − that is networks that use IP but are not connected to the Internet, These addresses are:−

- One A Class Network
  10.0.0.0
- 16 B Class Networks
  172.16.0.0 − 172.31.0.0
- 256 C Class Networks 192.168.0.0 − 192.168.255.0

You will note that this document uses these sequences throughout to avoid confusion with 'real' networks and hosts.

3.4 Network numbers, interface addresses and broadcast addresses

IP numbers can have three possible meanings:−

- the address of an IP network (a group of IP devices sharing common access to a transmission medium − such as all being on the same Ethernet segment). A network number will always have the interface (host) bits of the address space set to 0 (unless the network is sub−networked − as we shall see);
- the broadcast address of an IP network (the address used to 'talk', simultaneously, to all devices in an IP network). Broadcast addresses for a network always have the interface (host) bits of the the address space set to 1 (unless the network is sub−networked − again, as we shall see).
- the address of an interface (such as an Ethernet card or PPP interface on a host, router, print server etc). These addresses can have any value in the host bits **except** all zero or all 1 − because with the host bits all 0, the address is a network address and with the host bits all 1 the address is the broadcast address.

In summary and to clarify things

```
For an A class network...
(one byte of network address space followed by three bytes of host
address space)

        10.0.0.0 is an A Class  network number  because all the host
                bits of the address space are 0
        10.0.1.0 is a host address on this network
        10.255.255.255.255 is the broadcast address of this network
                because all the host bits of the address space are 1

For a B class network...
(two bytes of network address space followed by two bytes of host
address space)

        172.17.0.0 is a B Class network number
        172.17.0.1 is a host address on this network
        172.17.255.255 is the network broadcast address

For a C Class network...
(three bytes of network address space followed by one byte of host
address space)

        192.168.3.0 is a C Class network number
        192.168.3.42 is a host address on this network
        192.168.3.255 is the network broadcast address
```

Almost all IP network numbers remaining available for allocation at present are C Class addresses.

3.5 The network mask

The network mask is more properly called the subnetwork mask. However, it is generally referred to as the network mask.

It is the network mask and its implications on how IP addresses are interpreted *locally* on an IP network segment that concerns us most here, as this determines what (if any) sub−networking occurs.

The standard (sub−) network mask is all the network bits in an address set to '1' and all the host bits set to '0'. This means that the standard network masks for the three classes of networks are:−

- A Class network mask: 255.0.0.0
- B Class network mask: 255.255.0.0
- C Class network mask: 255.255.255.0

There are two important things to remember about the network mask:−

- The network mask affects only the **local** interpretation of **local** IP numbers (where local means on this particular network segment);
- The network mask is **not** an IP number − it is used to modify how local IP numbers are interpreted locally.

4.What are subnets?

A subnet is a way of taking a single IP network address and **locally** splitting it up so that this single network IP address can actually be used on several interconnected local networks. Remember, a single IP network number can only be used on a single network.

The important word here is **locally**: as far as the world outside the machines and physical networks covered by the sub−netted IP network are concerned, nothing whatsoever has changed − it is still just a single IP network. This is important − sub−networking is a **local** configuration and is invisible to the rest of the world.

5.Why subnetwork?

The reasons behind sub−networking date back to the early specification of IP − where just a few sites were running on Class A network numbers, which allow for millions of connected hosts.

It is obviously a huge traffic and administration problem if all IP computers at a large site need to be connected to the same network: trying to manage such a huge beast would be a nightmare and the network would (almost certainly) collapse under the load of its own traffic (saturate).

Enter sub−networking: the A class IP network address can be split up to allow its distribution across several (if not many) separate networks. The management of each separate network can easily be delegated as well.

This allows small, manageable networks to be established − quite possibly using different networking technologies. Remember, you cannot mix Ethernet, Token Ring, FDDI, ATM etc on the same physical network − they can be interconnected, however!

Other reasons for sub−networking are:−

- Physical site layout can create restrictions (cable run lengths) in terms of the how the physical infrastructure can be connected, requiring multiple networks. Sub−networking allows this to be done in an IP environment using a single IP network number.
  This is in fact now very commonly done by ISPs who wish to give their permanently connected clients with local networks static IP numbers.
- Network traffic is sufficiently high to be causing significant slow downs. By splitting the network up using subnetworks, traffic that is local to a network segment can be kept local − reducing overall traffic and speeding up network connectivity without requiring more actual network bandwidth;
- Security requirements may well dictate that different classes of users do not share the same network − as traffic on a network can always be intercepted by a knowledgeable user. Sub−networking provides a way to keep the marketing department from snooping on the R & D network traffic (or students from snooping on the administration network)!
- You have equipment which uses incompatible networking technologies and need to interconnect them (as mentioned above).

6.How to subnetwork a IP network number

Having decided that you need to subnetwork your IP network number, how do you go about it? The following is an overview of the steps which will then be explained in detail:−

- Set up the physical connectivity (network wiring and network interconnections − such as routers;
- Decide how big/small each subnetwork needs to be in terms of the number of devices that will connect to it − ie how many usable IP numbers are required for each individual segment.
- Calculate the appropriate network mask and network addresses;

* Give each interface on each network its own IP address and the appropriate network mask;
* Set up the routes on the routers and the appropriate gateways, routes and/or default routes on the networked devices;
* Test the system, fix problems and then relax!

For the purpose of this example, we will assume we are sub−networking a single C class network number: 192.168.1.0

This provides for a maximum of 254 connected interfaces (hosts), plus the obligatory network number (192.168.1.0) and broadcast address (192.168.1.255).

6.1 Setting up the physical connectivity

You will need to install the correct cabling infrastructure for all the devices you wish to interconnect designed to meet your physical layout.

You will also need a mechanism to interconnect the various segments together (routers, media converters etc.).

A detailed discussion of this is obviously impossible here. Should you need help, there are network design/installation consultants around who provide this sort of service. Free advice is also available on a number of Usenet news groups (such as comp.os.linux.networking).

6.2 Subnetwork sizing

There is a play off between the number of subnetworks you create and 'wasted' IP numbers.

Every individual IP network has two addresses unusable as interface (host) addresses − the network IP number itself and the broadcast address. When you subnetwork, each subnetwork requires its own, unique IP network number and broadcast address − and these have to be valid addresses from within the range provided by the IP network that you are sub−networking.

So, by sub−networking an IP network into two separate subnetworks, there are now **two** network addresses and **two** broadcast addresses − increasing the 'unusable' interface (host) addresses; creating 4 subnetworks creates **eight** unusable interface (host) addresses and so on.

In fact the smallest usable subnetwork consists of 4 IP numbers:−

* Two usable IP interface numbers − one for the router interface on that network and one for the single host on that network.

* One network number.
* One broadcast address.

Quite why one would want to create such a small network is another question! With only a single host on the network, any network communication must go out to another network. However, the example does serve to show the law of diminishing returns that applies to sub−networking.

In principle, you can only divide your IP network number into 2^n (where n is one less that the number of host bits in your IP network number) equally sized subnetworks (you can subnetwork a subnetwork and combine subnetworks however).

So be realistic about designing your network design − you want the **minimum** number of separate local networks that is consistent with management, physical, equipment and security constraints!

6.3 Calculating the subnetwork mask and network numbers

The network mask is what performs all the **local** magic of dividing an IP network into subnetworks.

The network mask for an un−sub−networked IP network number is simply a dotted quad which has all the 'network bits' of the network number set to '1' and all the host bits set to '0'.

So, for the three classes of IP networks, the standard network masks are:−

* Class A (8 network bits) : 255.0.0.0
* Class B (16 network bits): 255.255.0.0
* Class C (24 network bits): 255.255.255.0

The way sub−networking operates is to *borrow* one or more of the available host bits and make then make interfaces **locally** interpret these borrowed bits as part of the network bits. So to divide a network number into two subnetworks, we would borrow one host bit by setting the appropriate bit in the network mask of the first (normal) host bit to '1'.

For a C Class address, this would result in a netmask of
11111111.11111111.11111111.10000000
or 255.255.255.128

For our C Class network number of 192.168.1.0, these are some of the sub−networking options you have:−

```
No of       No of
subnets     Hosts/net     netmask
```

```
2           126       255.255.255.128 (11111111.11111111.11111111.10000000)
4            62       255.255.255.192 (11111111.11111111.11111111.11000000)
8            30       255.255.255.224 (11111111.11111111.11111111.11100000)
16           14       255.255.255.240 (11111111.11111111.11111111.11110000)
32            6       255.255.255.248 (11111111.11111111.11111111.11111000)
64            2       255.255.255.252 (11111111.11111111.11111111.11111100)
```

In principle, there is absolutely no reason to follow the above way of subnetworking where network mask bits are added from the most significant host bit to the least significant host bit. However, if you do not do it this way, the resulting IP numbers will be in a *very* odd sequence! This makes it extremely difficult for us humans to decide to which subnetwork an IP number belongs as we are not too good at thinking in binary (computers on the other hand are and will use whatever scheme you tell them with equal equanimity).

Having decided on the appropriate netmask, you then need to work out what the various Network and broadcast addresses are − and the IP number range for each of these networks. Again, considering only a C Class IP Network number and listing only the *final* (host part) we have:−

```
Netmask         Subnets Network B'cast  MinIP   MaxIP   Hosts  Total Hosts
--------------------------------------------------------------------------
    128            2        0      127       1     126    126
                          128      255     129     254    126     252

    192            4        0       63       1      62     62
                           64      127      65     126     62
                          128      191     129     190     62
                          192      255     193     254     62     248

    224            8        0       31       1      30     30
                           32       63      33      62     30
                           64       95      65      94     30
                           96      127      97     126     30
                          128      159     129     158     30
                          160      191     161     190     30
                          192      223     193     222     30
                          224      255     225     254     30     240
```

As can be seen, there is a very definite sequence to these numbers, which make them fairly easy to check. The 'downside' of sub−networking is also visible in terms of the reducing total number of available host addresses as the number of subnetworks increases.

With this information, you are now in a position to assign host and network IP numbers and netmasks.

7.Routing

If you are using a Linux PC with two network interfaces to route between two (or more) subnets, you need to have IP Forwarding enabled in your kernel. Do a

```
        cat /proc/ksyms | grep ip_forward
```

You should get back something like...

```
00141364 ip_forward_Rf71ac834
```

If you do not, then you do not have IP−Forwarding enabled in your kernel and you need to recompile and install a new kernel.

For the sake of this example, let us assume that you have decided to subnetwork you C class IP network number 192.168.1.0 into 4 subnets (each of 62 usable interface/host IP numbers). However, two of these subnets are being combined into a larger single network, giving three physical networks.

These are :−

```
Network         Broadcast       Netmask                 Hosts
192.168.1.0     192.168.1.63    255.255.255.192         62
192.168.1.64    192.168.1.127   255.255.255.192         62
182.168.1.128   192.168.1.255   255.255.255.126         124 (see note)
```

Note: the reason the last network has only 124 usable network addresses (not 126 as would be expected from the network mask) is that it is really a 'super net' of two subnetworks. Hosts on the other two networks will interpret 192.168.1.192 as the *network* address of the 'non−existent' subnetwork. Similarly, they will interpret 192.168.1.191 as the broadcast address of the 'non−existent' subnetwork.

So, if you use 192.168.1.191 or 192 as host addresses on the third network, then machines on the two smaller networks will not be able to communicate with them.

This illustrates an important point with subnetworks − the usable addresses are determined by the **SMALLEST** subnetwork in that address space.

7.1 The routing tables

Let us assume that a computer running Linux is acting as a router for this network. It will have three network interfaces to the local LANs and possibly a fourth interface to the Internet (which would be its default route.

Let us assume that the Linux computer uses the lowest available IP address in each subnetwork on its interface to that network. It would configure its network interfaces as

```
Interface       IP Address      Netmask
eth0            192.168.1.1     255.255.255.192
eth1            192.168.1.65    255.255.255.192
eth2            192.168.1.129   255.255.255.128
```

The routing it would establish would be

```
Destination     Gateway         Genmask         Iface
192.168.1.0     0.0.0.0         255.255.255.192 eth0
192.168.1.64    0.0.0.0         255.255.255.192 eth1
192.168.1.128   0.0.0.0         255.255.255.128 eth2
```

On each of the subnetworks, the hosts would be configured with their own IP number and net mask (appropriate for the particular network). Each host would declare the Linux PC as its gateway/router, specifying the Linux PCs IP address for its interface on to that particular network.

Robert Hart Melbourne, Australia March 1997.