
Understanding the Inter-Networking Protocol

- or -

What do those numbers mean?

Like *way* unfinished.

Second draft of April 18 1999 with thanks to the folks at the [appleshareip](#) list. pdf version of this document is at [tcp.pdf](#)

©Edward Irvine 1999.

Introduction

The Inter-Networking Protocol (IP for short) is now the *lingua-franca* of computer networking.

IP started out as an improved way to move email over a small network of mini-computers and mainframes. This network grew into “**The Internet**” that we know today.

The Internet's popularity and IP's inherent design strengths have resulted in IP being used more and more in Local Area Networks (LAN's) and personal computers as well. Microsoft, Apple and Novell are all quietly moving away from their “in-house” protocols in favour of IP. If you do anything at all with computer networks, even on the smallest scale, sooner or later you are going to have to deal with IP.

There are several comprehensive texts on IP, however these are written with computer professionals in mind. This document is different. For one thing, it is free. For another, it is aimed at Novice networkers - people who might already have the skills to set up a personal computer on their own, who have used a computer on a network, and who now feel that it is time to find out about what IP is about. In fact, I am a high school teacher, and this document was written with school teachers in mind.

This document will explain the fundamentals of how IP works and what those numbers in your computer's control panels mean.

Please note that this is not intended as a definitive reference: To explain concepts, I make a lot of simplifications (perhaps oversimplifications).

One more important point: IP is often called “**TCP/IP**”; or just plain “**TCP**” by people who ought to know better. Technically this is not quite correct, although it probably doesn't matter. TCP is short for “Transmission Control Protocol” and an internal detail of IP which you don't need to worry about.

What IP is good for:

Figure 1, below, shows two Local Area Networks connected together using a telephone line and modems.

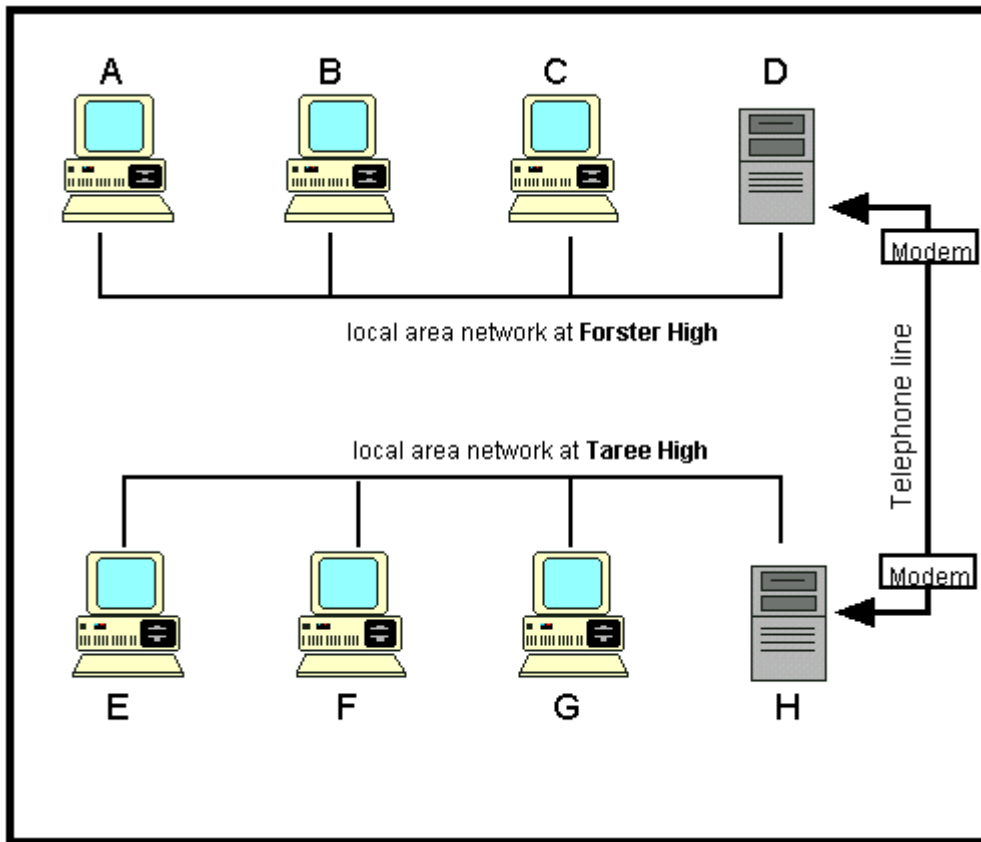


Figure 1: Two LANs joined together.

IP allows computer A to communicate with computers E, F, G, and H, as well as computers B, C, and D. This is where the term “**Inter Networking**” comes from. Appletalk and Novell's IPX, the other two common networking protocols, can also work across a dozen or so networks, as well as within networks, just like IP. However, they can't cope when hundreds or thousands of networks joined together in the spider-web design that the Internet has become. The ability to link thousands of networks is what makes IP stand out from the others.

Can you see how computers D and H might be called “Link Computers”? Well, the correct term is actually a “**router**” or “**gateway**”. Setting up a router is strictly a job for the computer professional, and well outside the scope of this document. For one thing, these computers need to run advanced operating systems like Unix, Linux, Novell Netware or Cisco's NOS.

Fortunately, setting up a Macintosh or Windows-95 computer is much easier than setting up a gateway. Before we do that, we are going to examine IP to understand a little about how it works.

Now, note the two LANs in figure 1 are not connected to the Internet. They *could* be, of course, but they don't have to be to use IP. The efficiency of IP (perhaps marginally better than IPX, but almost twice as efficient as Appletalk) is reason enough to use IP alone, Internet or not.

These days the link between the two networks shown above is often something much faster than an ordinary telephone line, but telephone lines (and 300 baud modems!) are what the designers of IP had in

mind.

Now have a look at figure 2. It shows many LANs joined together in a web like fashion - a sort of “minature internet”. As before, this minature internet *may* be connected to the real Internet, but it doesn't have to be for people to use web browsers, send email, and share files and printers.

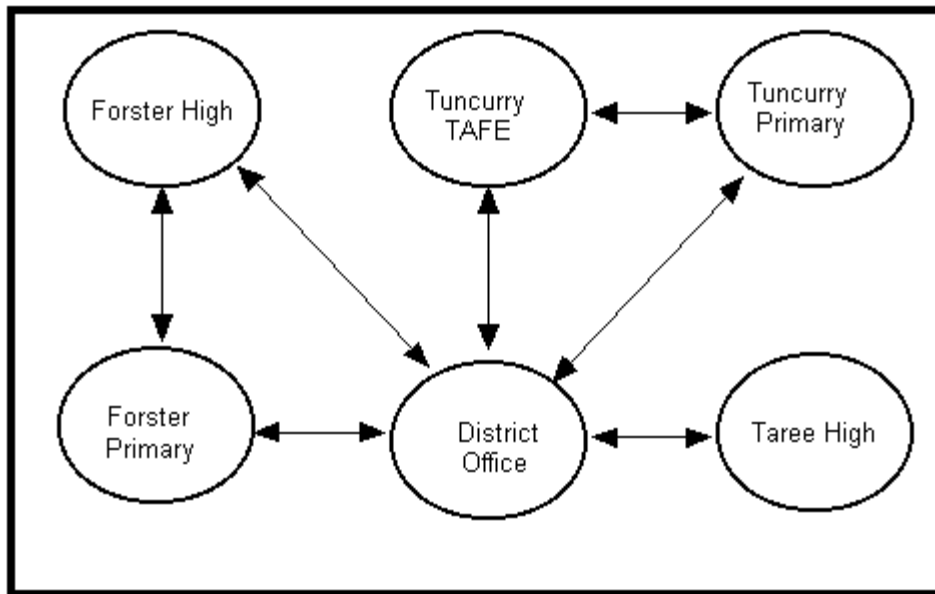


Figure 2: A network of LANs.

A “web like” design has several advantages you may not have thought of: Firstly, it means a “network of networks” (of which the Internet is only one example) can grow organically, with only minimal supervision from the authorities. Secondly, there are a lot of redundant links. If one or two links break, communication can still occur, although it will take a little longer.

The power to link many networks together comes at a price to the individual user: Setting up your personal computer to use IP usually requires you to type some numbers into your computer's control panels. This is not the case with Appletalk, which was designed to set itself up automatically.

Fortunately, when you set up your PC or Mac for IP, you don't have to leave Appletalk or IPX behind. Windows and Macs can run all three protocols at the same time, if need be. That's why a Macintosh user can print to a Laserwriter (which speaks AppleTalk) while browsing the Internet (which speaks IP).

How computers talk - the very basics.

Computers exchange information using little packets of data. Each packet is usually 1500 bytes long and takes around one-thousandth of a second to transmit. Each packet has the sender's address and the destination address on the outside just like a postal envelope.



Figure 3: A stream of data is sent in packets.

The sending computer divides the stream of data (it could be a file, a web page, or a print job) it is transmitting into little packets. The receiving computer simply reassembles the packets in the correct order, requesting that damaged or missing packets be re-transmitted.

Built into IP are agreed methods to deal with packets that get lost, packets that are arriving too fast for the sending computer to deal with, and other sundry details.

If this sounds to you like the postal system, it's no accident. The designers of IP had email firmly in mind, and legend has it that the basic design was sketched out on the back of an envelope in a cafe.

IP addresses.

For IP to work, two things are needed:

1. Every computer has to have its own number to identify itself. This number is called the **IP address**. You can think of an IP address as similar to your telephone number or postal address.
2. All IP addresses on a particular LAN must start with the same numbers. For instance, I know that if I live in Taree, my telephone number must start with a "65". If you live in Tamworth, your telephone number must start with a "67". At Forster High School, all IP addresses start with the numbers 10.13.175. (Note that this is a private network unreachable from the Internet).

In most cases, some sort of computer networking person will give you an IP address, or range of addresses to use on your computers. Just remember, each computer must have it's own IP address.

Sometimes, a server computer has been set up to "lend" or hand out IP addresses to personal computers when they start up by running a program called a "**DHCP**" or "**BootP**" server. If this is the case, lucky you! You probably don't need to type anything into your network control panels, and this document is perhaps academic!

This "lending" of IP numbers is used extensively by the internet service providers you might dial up from home. After they confirm your password, they send back an IP address for your computer to use for the duration of your session. Internet service providers don't do this out of the goodness of their hearts, mind you. They often have many more customers than IP numbers.

How IP addresses are written

IP addresses are usually written as a sequence of **four numbers** separated by periods. This is called the “dotted decimal” notation. Each number can be between 0 and 255. However, the last number ought to be between 1 and 254 (special significance is attached to IP numbers that end with 255 or 0). Inside your computer, an IP address stored as a 32 bit (4 byte) integer. Dotted decimal notation is just an easy way for humans to write an IP address without having to know the binary or hexadecimal numbering that computer programmers use.

Now have a look at Table 1 to see some examples of IP addresses.

IP number	Valid?
200.100.50.25	yes
1.1.1.1	yes
1.1.1.0	no
1.1.1.255	no
1.1.255.1	yes
1.1.900.1	no
6.6.6.6.6.6	no
2.bottles.of.wine	Oui? Non!

Table 1: Dotted decimal notation

Several ranges of IP addresses have been set aside for networks that will never be connected **directly** to the internet. See Table 2. Packets from these addresses will be dropped by the first router they pass through on the internet, should they “escape” from a private network.

First in range	Second in range	Last in range
10.0.0.1	10.0.0.2	10.255.255.254
172.16.0.1	172.16.0.2	172.16.255.254
192.168.0.1	192.168.0.2	192.168.255.254

Table 2: IP numbers for private networks

How computers talk - a little more advanced.

Let's say you live at 12 Fred St, Frederickton, NSW, Australia, and you've got fifty parcels to deliver next door to 14 Fred St. You'd just do the delivery yourself, right? Now, if those parcels had to go to 14 Fred St, Long Beach, California instead, you'd give them to the postmaster (probably by putting them in the nearest post box) and let the postal system do the delivery. The postmaster might not know where Long Beach is, but he probably knows that anything for the United States should be forwarded to Sydney. When the parcels get to the Sydney postmaster, she probably knows that any parcels for California should be forwarded to Los Angeles. And so on.

In networking terms, your own street, where you deliver packets yourself, is your Local Area Network. Packets for every other destination get delivered to the postmaster, which network engineers call a “**router**” or “**gateway**”. In figure 1, the routers are computers D and H. It is possible, but unusual, for an average office LAN to have more than one router on it.

Because personal computers aren't as smart as postmasters they need a foolproof mathematical method of working out if they should attempt to deliver a packet locally (because the address is on the local LAN) or give it to the router (because the address is not on the local LAN). This is where the **netmask** comes in.

In dotted decimal notation, the most common netmask is 255 . 255 . 255 . 0 . Without getting bogged down in mathematics, this netmask says “If a computer has the first **three** IP numbers the same as mine, then it must be on the same LAN as me and packets should be sent directly to that computer and not to the router”. A netmask of 255 . 255 . 0 . 0 says which “If a computer has the first **two** IP numbers the same as mine, then it must be on the same LAN as me and packets should be sent directly to that computer and not to the router.

Some examples are now in order.

My netmask	My IP address	Recipient's IP address	Delivery method
255 . 255 . 255 . 0	100 . 100 . 100 . 100	100 . 100 . 100 . 1	Local.
255 . 255 . 255 . 0	100 . 100 . 100 . 100	100 . 100 . 100 . 2	Local.
255 . 255 . 255 . 0	100 . 100 . 100 . 100	100 . 100 . 99 . 1	Give to router.
255 . 255 . 255 . 0	100 . 100 . 100 . 100	100 . 99 . 100 . 1	Give to router
255 . 255 . 255 . 0	100 . 100 . 100 . 100	99 . 98 . 97 . 100	Give to router
255 . 0 . 0 . 0	100 . 100 . 100 . 100	100 . 99 . 100 . 1	Local (why?)
255 . 255 . 0 . 0	100 . 100 . 100 . 100	100 . 100 . 99 . 1	Local (why?)

Table 3: Should a packet be sent to the router or delivered directly?

So, that's the theory of IP addresses, netmasks, and routers. Before we attack your computer's control panels, we have one more thing to cover - the Domain Name Service, or DNS for short.

Domain Name Service (DNS)

We all know that computers like to work with numbers, and humans prefer **names**. The designers of the internet realised this, and have set up a system to give names to computers on the internet. A sort of computerised telephone converts names back into IP addresses so computers can do the real work.

Let's look at how computers are named first.

Computers (in networking jargon, an individual computer is called a “**host**”) are given a name much like houses are given names in England. For instance,

Coombe House,
Burnhams Lane,
Big Bookaham,
East Surrey,
United Kingdom.

Translated into a **hostname** for a computer, the above address might look something like so:

`coombehouse.burnhamslane.bigbookham.eastsurrey.uk`

Of course, an address like:

`coombehouse.burnhamslane.bigbookham.eastsurrey.au`

points to a different house in Australia (and thus in an utterly different street and town, for that matter).

Now we'll look at how these names are changed into IP addresses.

A **DNS Server** is a computer somewhere that can change a **hostname** into an **IP address**, and vice versa. It holds a database similar to a telephone book. It also knows the address of other DNS servers it can query if it doesn't have an entry for address you are looking for. If you think about it, this is much like what a telephone operator does if you ring up directory assistance and ask for a number in Pogo-Pogo land.



Figure 4: Connecting via the Internet.

When you ask your web browser to connect to `www.yahoo.com`, your computer asks a DNS server to look up the IP address of `www.yahoo.com`. The DNS server will tell your computer “*Success! I have found out the IP address of www.yahoo.com. It is 204.71.200.68*”. Your computer then uses that IP address to contact yahoo.

It is often wise to point your computer at several DNS servers for redundancy purposes. Your network administrator should always give you at least two DNS numbers to type in.

In a nutshell, every computer must have an IP address. Computers are usually given names as well, but only for human consumption. A DNS server converts the names back to IP addresses so networking can function.

Whole books have been written about DNS, and setting up a DNS server correctly is a real handful for the uninitiated. All most users ever need to know about DNS what the numbers are and where to type them in.

Of course, if you want to make up names for your own computers, you need to get to know whoever it is that administers your nearest DNS server. (A few bottles of beer may be helpful...)

The control panels

First, I'll do a quick run-down of how to configure Windows-95, and then I'll do Macintosh. I'm not going to cover Windows-3.x as it's just too much of a pain. Sorry.

An important point: This demo covers what you might do if you were adding a computer to a local area network, **not** what you would do to set your computer up to dial in to the internet. Your Internet Service Provider ought to have some instructions for doing that.

Another important point: I am **not** covering the set up of a web browser, or setting up your computer for peer-to-peer networking, connecting to a Novell or Windows fileserver, as these are way outside the scope of this document.

When I do Mac and FreeBSD & Linux, I'll fork these three sections off into separate pages

Windows 95/98

Open up the **networking** control panel. You should see something like so:

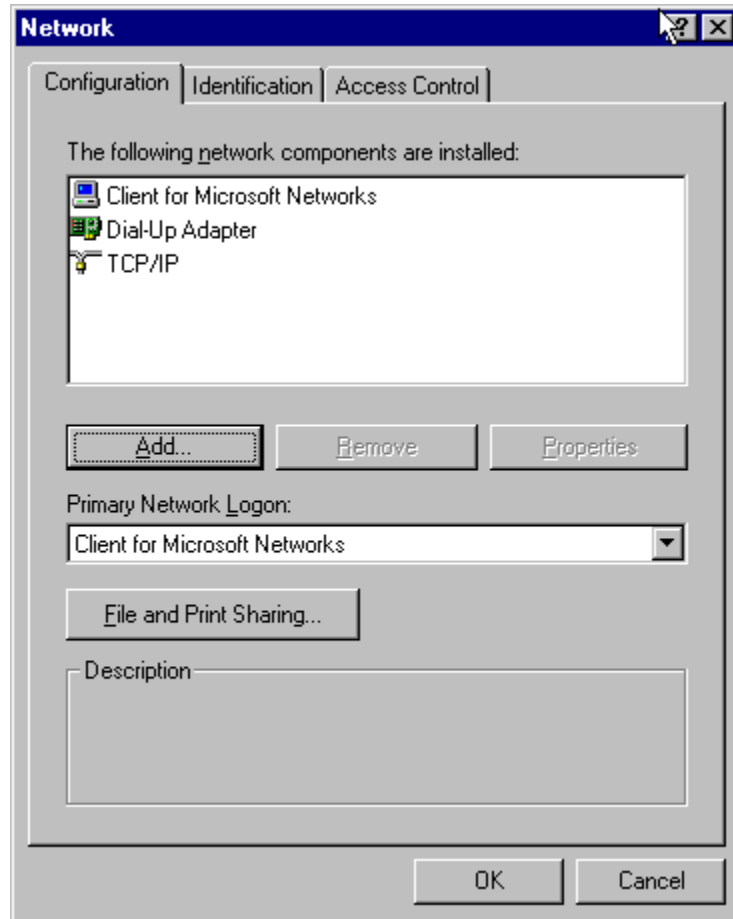


Figure 5: Networking control panel

You may see some other things there, like IPX, Client for Novell Networks, network cards and the like, but check to make sure that “Client for Microsoft Networks” is there, and also “TCP/IP”. If either are missing you will have to use the “Add” button to add them. Hint: TCP/IP is found in the *Protocol --> Microsoft* section.

Once you have TCP/IP showing up, double click on it. We want the “I.P. Address” tab first. You should now see something like so:

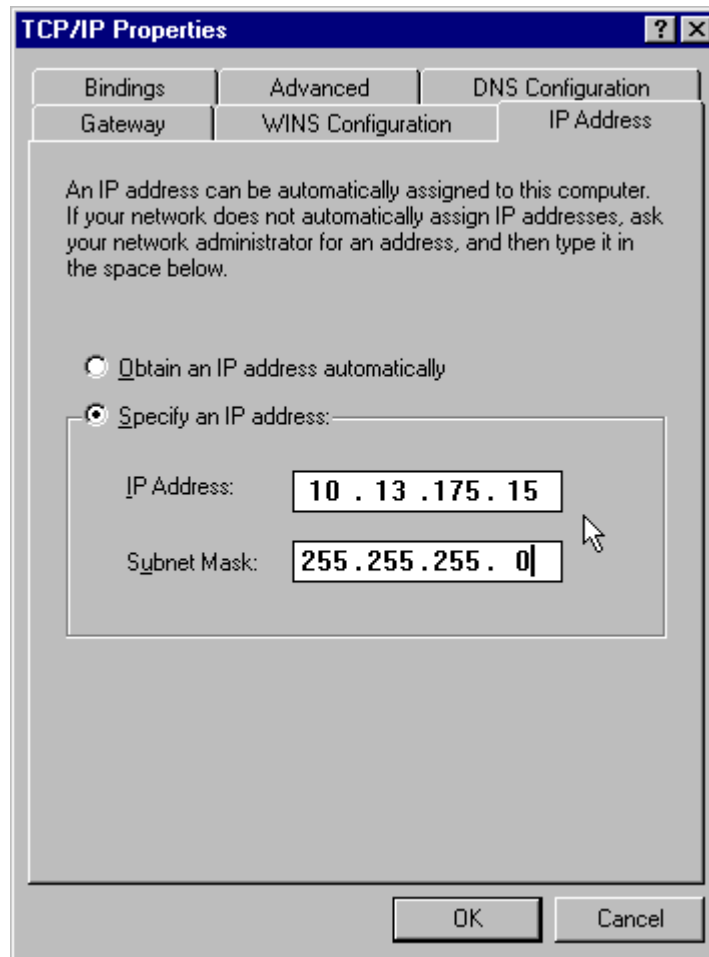


Figure 6: TCP/IP control panel - IP address.

It's now pretty simple. Just type in the IP address of your computer and the netmask. Don't press OK yet! Go to the "Gateway" tab next.

After clicking on the "Gateway" tab, you should see this. Just type in the IP address of the gateway computer and click the "Add" button.

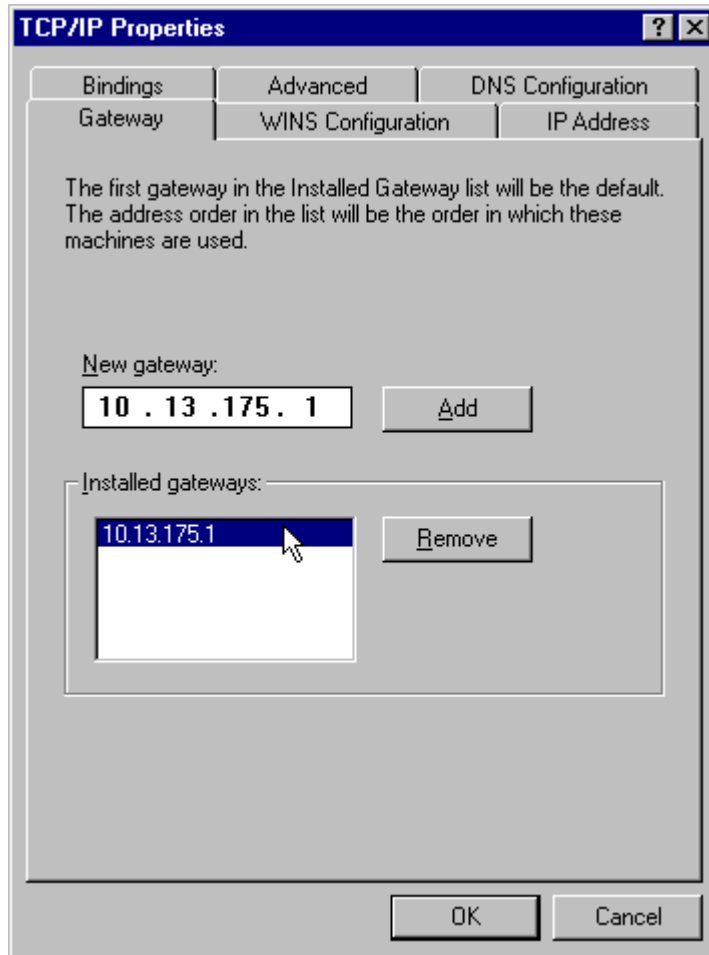


Figure 7: TCP/IP control panel - gateway address.

It is possible that your network administrator has given you several gateway addresses. That's O.K., add them all. If you are on a private network with no gateway, just type in 0 . 0 . 0 . 0. Once you're done adding the gateway(s), there is one more entry to make: Don't press OK! Click on the "DNS" tab.

After clicking on the DNS tab, you should see the following screen, which is a little more involved.

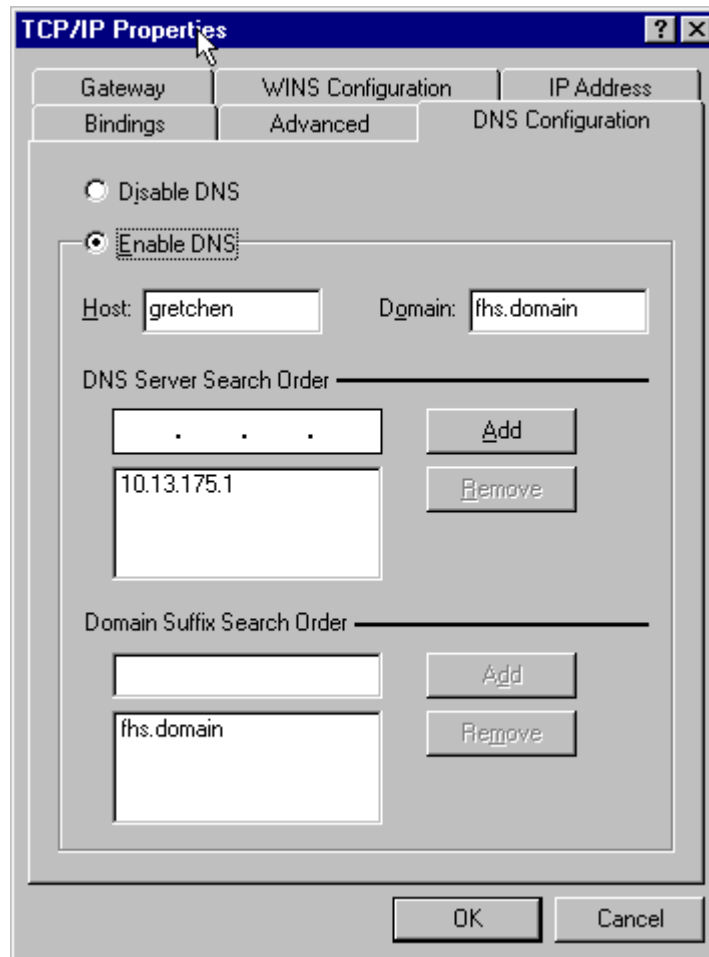


Figure 8: TCP/IP control panel - DNS.

This computer is set up for a hostname of `gretchen.fhs.domain`. There is only one DNS server around, and its IP address is `10.13.175.1`. Your network administrator will give you a hostname and DNS server address.

Click through the tabs once more, checking to make sure all the entries are correct. Note that I haven't covered "WINS", "Advanced", or "Bindings", as these are well outside the scope of this document; You shouldn't have to change anything in there anyhow. **Now** you can press O.K. You want to close all the control panels, (yes, you want to save changes), and the computer should ask you if it can restart. Yes, you want to restart the computer.

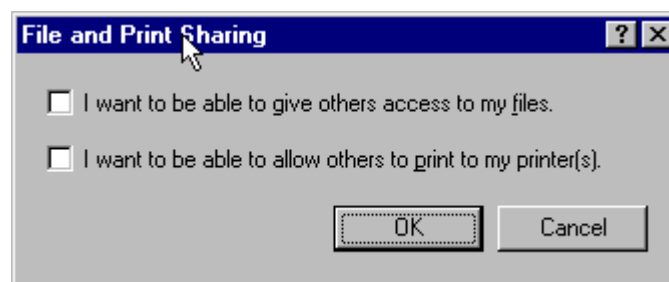


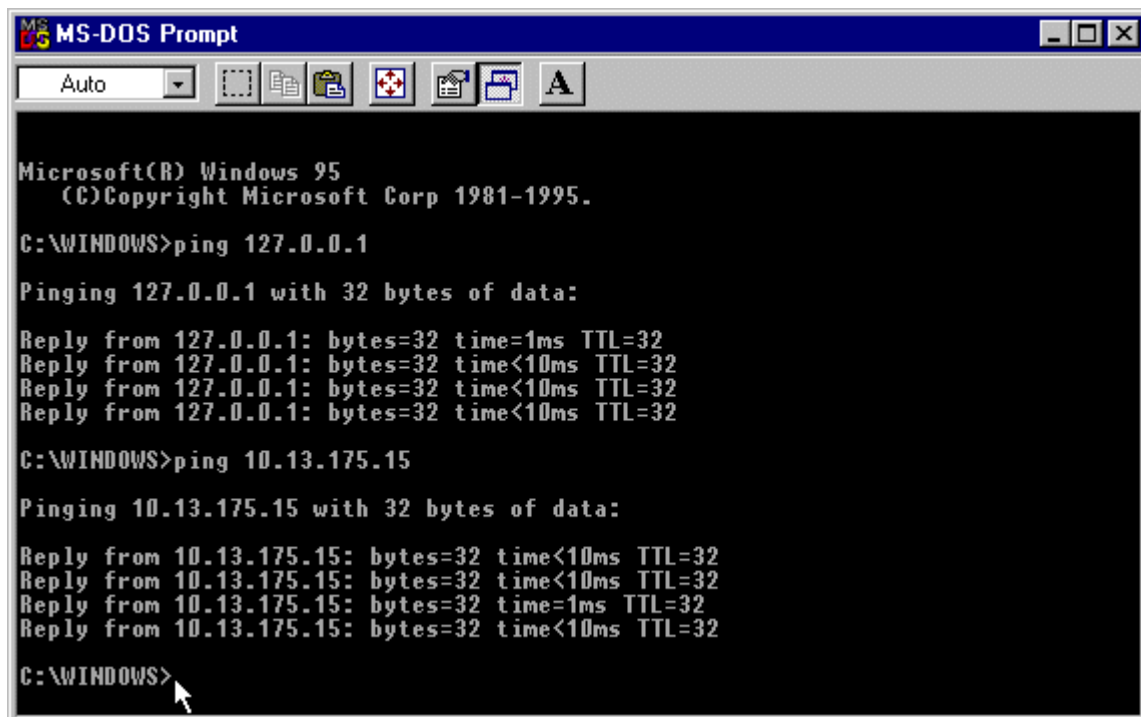
Figure 9: Network control panel. File and Print sharing dialog.

OK. Once your computer has restarted, you need to go back to the networking control panel, click on “File and Print Sharing”, and make sure that the contents of your computer aren't available for the rest of the network to fool around with. You should get a dialog box that looks like figure 9. You probably want both boxes un-ticked.

Testing the connection

There are five things to do to test the connection:

1. Is the **Link Light** on? The link light is a (usually) green light where the ethernet cable plugs into the back of the computer. A link light that is on indicates that everything is plugged in OK at both ends. It also indicates that the cable is probably OK. It sometimes helps if you re-boot after checking the connections.
2. Can you ping yourself using the **Loopback address**? Open an MS-DOS window, and type "ping 127.0.0.1" [return] without the quotes. If you don't get replies, this might mean that the TCP/IP protocol is not present in the network control panel.
3. Can you ping your **own** IP address? In my previous example, this was 10.13.175.15.
4. Can you ping the IP address of other computers on the network that has been set up for IP? A fileservers, the DNS server, and gateway are all good candidates.
5. Can you ping another computer given it's hostname? For instance, "ping www.yahoo.com". This checks to make sure your DNS server entry is OK and your gateway is running correctly.



```
MS-DOS Prompt
Auto
Microsoft(R) Windows 95
(C)Copyright Microsoft Corp 1981-1995.
C:\WINDOWS>ping 127.0.0.1
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time=1ms TTL=32
Reply from 127.0.0.1: bytes=32 time<10ms TTL=32
Reply from 127.0.0.1: bytes=32 time<10ms TTL=32
Reply from 127.0.0.1: bytes=32 time<10ms TTL=32
C:\WINDOWS>ping 10.13.175.15
Pinging 10.13.175.15 with 32 bytes of data:
Reply from 10.13.175.15: bytes=32 time<10ms TTL=32
Reply from 10.13.175.15: bytes=32 time<10ms TTL=32
Reply from 10.13.175.15: bytes=32 time=1ms TTL=32
Reply from 10.13.175.15: bytes=32 time<10ms TTL=32
C:\WINDOWS>
```

Figure 10: Calling the Ping program from MS-DOS.

Macintosh FreeBSD Linux

still to do

Constructive criticism? Be harsh. Note that Mac, FreeBSD and Linux are all on the to do list. Eddie at eirvine@tpgi.com.au. My home page is [here](#) and has a few other odds and ends.

