CHAPTER FIVE

Laying the Foundations: Linux As a LAN Server

Linux shines as a server. As a LAN server, you'll most likely use Linux for the following:

- Accessing the Internet across the LAN
- Sharing files
- Sharing printers
- Sending and receiving local e-mail

As a small business intranet server, you can effectively use Linux for the following:

- Hosting a company Web site
- Sharing data from a server-based database

Setting up these applications is covered in the next chapter. First, the foundations must be laid. You should start by *designing* the LAN, taking into account how, and why, it will be used. Next, you need to connect computers physically, and you have to install and configure software, both on the server and on the client machines.

This chapter covers the basics: hardware setup, determining IP addresses, and setting up DHCP and DNS. (If you need an explanation of these terms, read on—they're described in the sections that follow.)

Traditionally, setting up a server, especially a Linux/UNIX server, has been considered the role of a highly trained system administrator. This book takes the view that any reasonably computer-literate employee can set up such a server and can maintain a mixed Linux/Windows LAN for a small firm or workgroup.

Server setup has been made easier by recent improvements in mouse-driven LAN configuration tools, which now extend even to domain name servers. Some configuration, however, still does entail editing certain text files. But text file editing is no more difficult than word processing, and to make the process easier, you can download many configuration files from the Linux Leap Web site (`http://www.linuxleap.org`).

The system administrator is not left out of the equation. This book presents enough detail to be useful as a primer for the administrator new to Linux who desires to understand how it all works, and who may want or need to customize and enlarge his or her LAN for a growing organization.

NOTE *This chapter is mostly concerned with configuration issues, and so, where it is not explicitly stated, it is assumed that the user is logged in as the root user (the system administrator).*

Presentation

Examples aid the learning process by offering practical illustrations to the text. This and following chapters use both a fictional scenario (inspired by a real company in Calgary, Alberta, Canada) and real examples from around the world.

Chapter Five is organized into the following lessons:

- Lesson 5-1. Hardware and Addresses
- Lesson 5-2. Setting Up a Dynamic Server and Clients
- Lesson 5-3. Setting Up a Domain Name Server for the LAN

The Scenario

Development, Ltd., is a small company specializing in project management for domestic (Canadian) health projects, and Canadian and U.S. funded developing world health projects. It does not engage in long-term contracts; rather, it concentrates on producing project viability studies and management reviews. In short, it's a highly specialized firm in a narrow niche market.

Several years of quiet success have brought Development, Ltd., name recognition, and it suddenly finds itself with much more business than it can handle. The owner and CEO has long made do with one part-time and seven full-time employees; she is now faced with adding six new full-time employees and six new computers.

Currently, she has an elderly NT server functioning as a file server and workstation, and several older client machines that she upgraded a year ago to Windows and Office 2000. They are slow, but stable. Her ISP provides her with an ADSL line, five company mailboxes, and hosting for the company's Web site.

Windows XP worries her: She foresees expensive hardware upgrades, and she has a feeling that she's going to be paying for features her employees neither want nor need. She understands that if she upgrades her desktops she should probably upgrade her server, and since she wants to institute an intranet, she's considering BackOffice 2000. However, the license is going to cost almost US$4,000 for five clients and another US$200 for each additional client. She may well need new hardware on top of that. Finally, she wants local e-mail for all employees, but she's run out of the free mailboxes supplied by her ISP.

In short, she has a suspicion that she's locked into a software and hardware model of rigid and regular upgrades, and solutions that, for a small firm like hers, are expensive.

Her new office manager (who is keen to make her mark and has recently read this book) convinces the CEO to try a different approach. Development, Ltd., will buy a preloaded Linux desktop PC for a new server and six new laptops, all preloaded with Linux, for the new employees. Since all the software needed for an intranet server and local e-mail solution comes with Linux, and since StarOffice is downloadable for free, the additional software costs are zero. In addition, the office manager has assured the CEO that those employees currently with Windows machines can keep them (should they want to), and that the new Linux desktops will be compatible.

Figures 5-1 and 5-2 show the present and future setup at Development, Ltd.

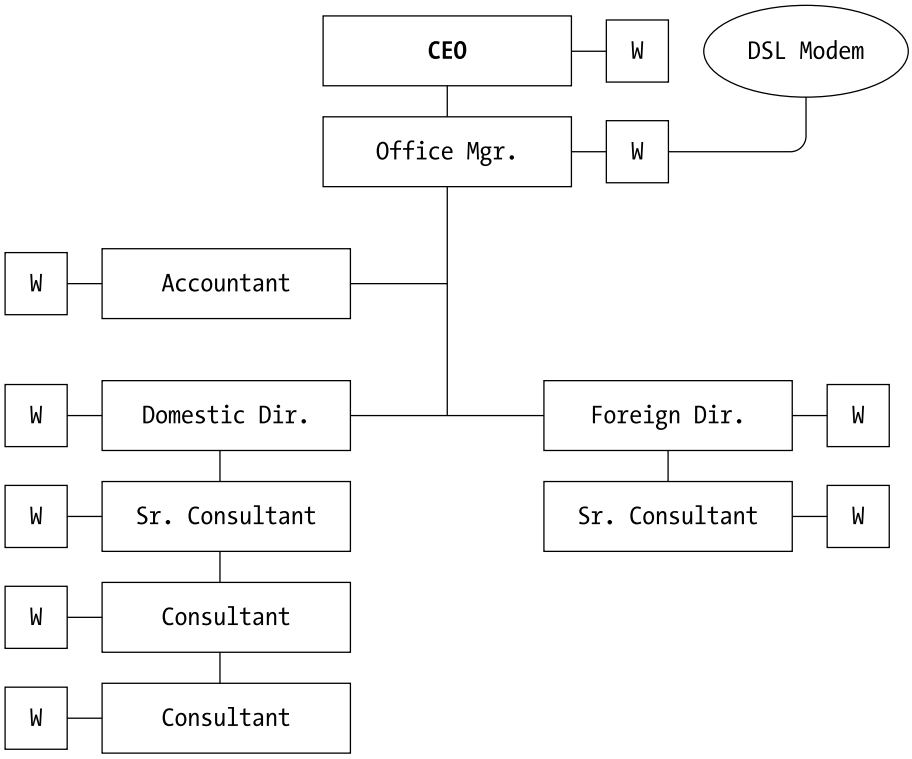

Figure 5-1. The old setup, before expansion

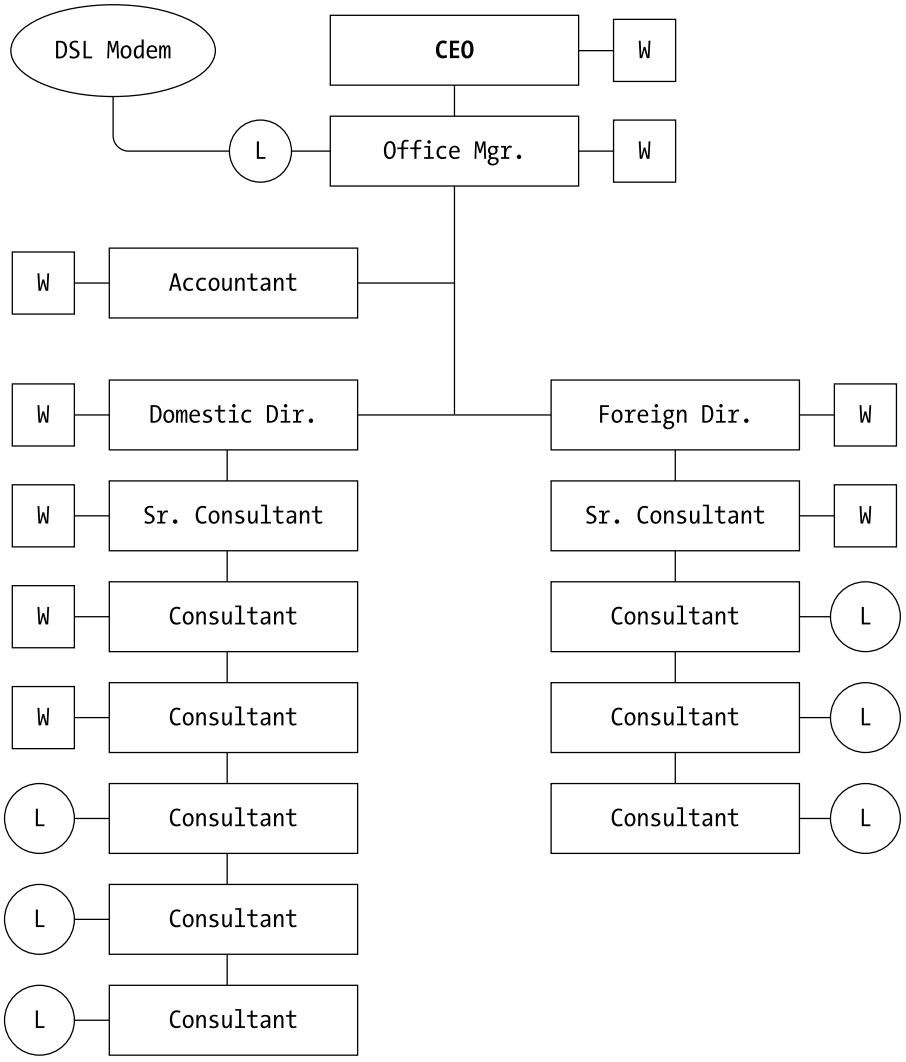

Figure 5-2. The fruits of success: six new employees, six new computers, and a mixed Linux/Windows network

LESSON 5-1.

HARDWARE AND ADDRESSES

This lesson covers hardware connections and computer addresses in a mixed Linux/Windows LAN.

Hardware Connections

IN BRIEF

1. Install two NICs into the Linux server, one for the incoming Internet connection and the other for the workgroup hub or switch.

2. Connect the Linux and Windows workstations to the hub or switch with Ethernet cable.

3. Connect a printer (or printers) to either the server or workstations, designating those machines as printer servers.

4. With increasing numbers of workstations, cascade the hubs or replace them with a switch.

Figure 5-3 presents a diagram of a simple SOHO Linux/Windows LAN.

EXPLAINED

Ethernet is the universal type of network for everything from the SOHO to the enterprise. It's simple to set up, the necessary hardware is inexpensive, and the result is fast communications between computers.

To set up an Ethernet network, you need

- A hub or switch, which acts as the central connection point for all the computers

- Network interface cards (NICs), one connected to every computer

- Ethernet cable to connect the computers to the hub(s) or switch

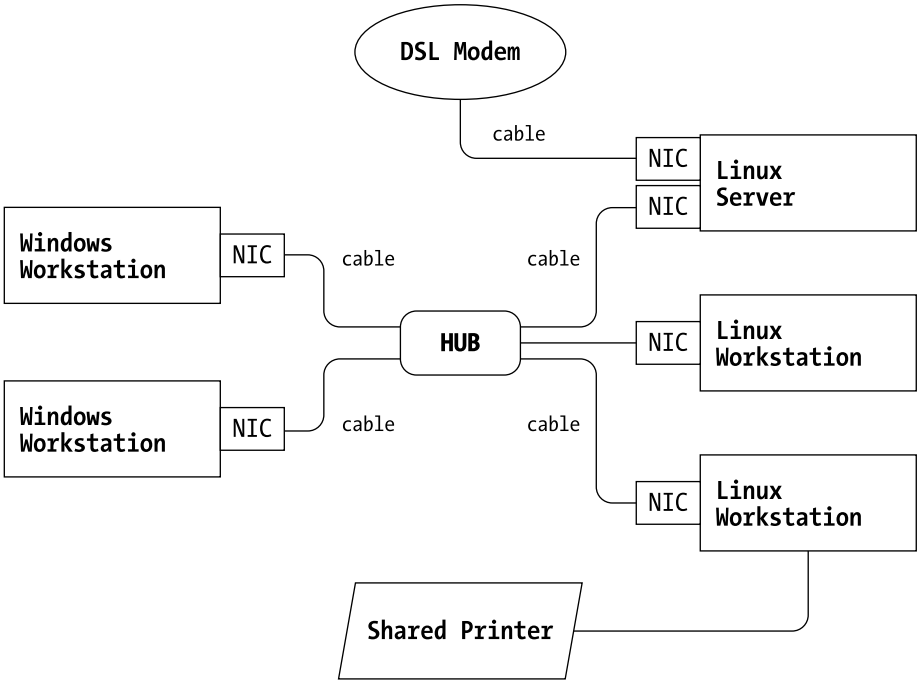

Figure 5-3. Three Linux workstations (one doing double duty as an Internet server, another as a print server) and two Windows workstations, all tied together with a hub

Keeping in mind the example in Figure 5-3, let's start from the outside and move in. Every small business LAN will be connected to the Internet. There are several ways to make such a connection, but a likely method is through an ISP-provided ADSL modem. Service speeds for such lines are now very fast; small business ADSLs can be capable of up to 4Mbps (this speed represents downloading—uploading will be much slower). Standard Ethernet cable would connect the ISP's modem or router to the Linux server's second NIC.

In the example in Figure 5-3, note that the topmost Linux workstation has two NICs, one of which is connected to the ADSL modem. This workstation acts as an Internet server to the rest of the LAN, eliminating the need for a router (Chapter Four contains connection and configuration details for the server's ADSL line).

Your ISP will probably provide the modem for the incoming Internet line. However, you'll be responsible for providing all the hardware for the LAN itself. Before shopping around, it's a good idea to know exactly what specifications you should be looking for.

- *If possible, you should buy all the hubs, switches, and NICs from the same manufacturer,* and they should be certified by the manufacturer to work in both a Linux and Windows environment. This does not mean that you have to buy expensive hardware—you can find quite inexpensive hardware to fit this specification.

- *The hub or switch.* You should buy a 100Mbps (Fast Ethernet) hub or switch. Hubs are cheaper; switches are faster.

- *The NIC.* It's easy to install a network interface card into a computer, but it's even easier to buy a computer with one preinstalled at the factory. If you have to buy a NIC, make sure that it fits into your computer—you should have a spare PCI slot. Like the hub, the NIC should be 100Mbps. Laptops will need a PC Ethernet card; there are many available.

- *Avoid hardware needing USB ports.* Linux has only scrappy support for USB at this time.

- *The cabling.* Buy standard category 5 twisted pair Ethernet cable. You can buy this cable either with or without connector jacks attached.

Hubs or Switches?

The difference between a hub and a switch is that the different ports of a hub all share the same bandwidth, while the different ports of a switch are assigned their own. What this means practically is that hubs produce slower connections on heavily loaded networks. What's worse, the slowdown will not necessarily have a linear relationship with the amount of traffic; it may get much slower quite suddenly, as traffic appears to increase only moderately.

On the other hand, a hub with only a light and occasionally a moderate load, particularly a 100Mbps hub, can function quite well.

The benefits of a hub over a switch are cost and ease of use. A switch with the same number of ports and similar basic specifications as a hub from the same manufacturer will typically cost two to three times as much. And with a hub, you simply plug in the Ethernet cabling, whereas with many switches there's a certain amount of initial configuration.

Both hubs and switches come in a bewildering variety of sizes (number of ports) and costs. You can find very cheap but perfectly capable hubs with only five to eight ports for US$50 to US$60. A low-end eight-port switch can be had for

US$80. Moving up the ladder, you can buy a 24-port hub from a major manufacturer for about US$500, while a 24-port switch from the same company will run about twice as much. All the examples given are for units capable of 100Mbps. You can mix and cascade switches and hubs (to increase the size of the LAN), although cascading will slow the LAN down, as well as increase the number of connections.

Linux Compatibility

Fortunately, most major (and many minor) brands and models of NIC and associated hardware are now Linux compatible. However, if you have any doubts, and you have trouble finding a company representative, consult the Linux Hardware Database (you can find a link to it at `http://www.linuxleap.org`).

Cabling

Unless you're handy with hand tools and willing to go to some trouble, it's best to buy the cable with connectors attached. This means some advance planning: You need to have a wiring plan before you buy the wire. Ethernet cable can be bought in many different lengths, and some stores will sell custom lengths and attach the connectors for you. Note that there is a limit of 100 meters (328 feet) to any single connection.

Making the Connection

Ideally, the NICs were already in the computers when you loaded Linux. However, if they weren't, turn off the machines and install the NICs and the PC Ethernet cards, if you have laptops. After all the NICs and cards are installed, connect them with the Ethernet cables to the hub or switch.

If the hardware is Linux compatible, it should be detected automatically during a reboot. You can check on this quite simply by using the ifconfig command, as follows:

1. Open a terminal and type **ifconfig**.
2. Press Enter.

This should produce a display something like the one shown in Figure 5-4.

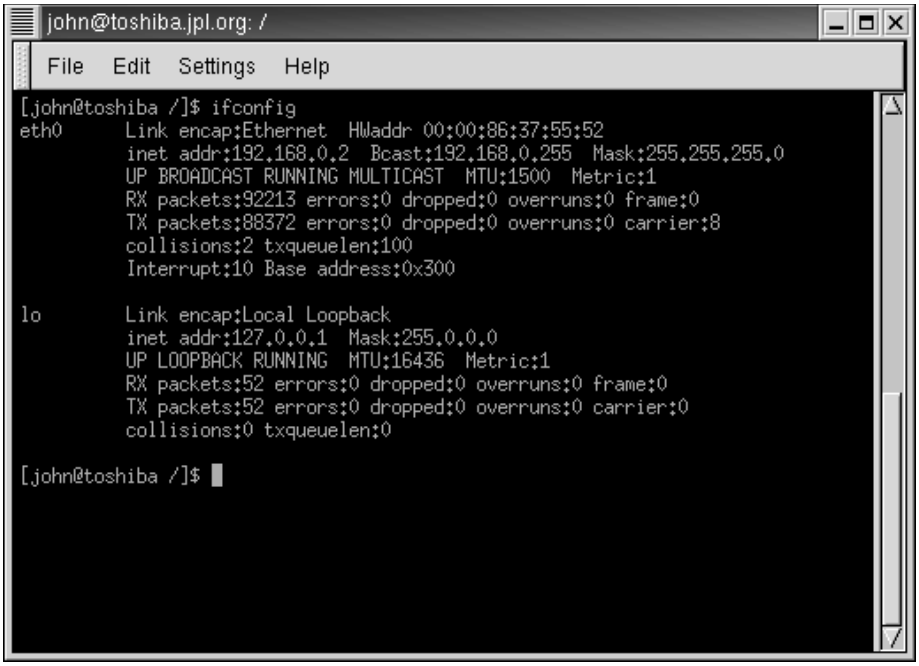

Figure 5-4. The network interface card showing as eth0

On older and less compatible hardware, special steps may be necessary to enable Linux to detect NICs. Chapter Four presents details on configuring new hardware in Linux.

Computer (IP) Addresses

IN BRIEF

1. Decide on a domain name for your company LAN and hostnames for each computer.

2. Decide on a range of IP addresses. An easy and safe method is to double the maximum expected size of your LAN. One of the standard address ranges for a LAN starts at 192.168.0.1. Therefore, a range of computer addresses for a 20-computer LAN expected to grow to a maximum of 40 would be, when doubled, 192.168.0.1 to 192.168.0.80. This allows for a total of 80 computers on the LAN.

3. Assign each computer on the LAN its own IP address.

EXPLAINED

Names

Each computer on a LAN needs its own unique name and address. Names are divided into hostnames and domain names. Each machine has its own hostname, whereas the entire LAN—and probably the company—will share a domain name. In addition, the LAN's domain name will probably be the same as the company's Web domain name. This is a straightforward convention that promotes simplicity and easy configuration.

Some examples of host and domain names include the following:

ceo.dev.org

ibm1.dev.org

mariann.dev.org

In all these examples, the domain name is dev.org. The hostname examples represent three different ways of deriving hostnames: the first by job function; the second by computer make, model, or number; and the third by the user's name.

NOTE *The dev.org domain name is an example only, for the fictional company Development, Ltd. In fact, such a firm would almost certainly find that all its most obvious choices have already been taken, and it would be forced to devise some abbreviation that would be euphonious, easy to remember, clearly connected to the firm—and available.*

Addresses

People communicate with computers by names, but computers communicate with themselves via Internet Protocol (IP) addresses. It follows that each computer has both kinds of addresses, which are unique to that machine.

IP addresses are written in groups of four decimal numbers, each number being in the range of 0 to 255. The four numbers are separated from each other by dots. Two random examples are

192.168.0.23
142.57.108.143

Special groups of addresses have been reserved for LANs that are not directly connected to the Internet—as in this case, where each computer is indirectly connected through an Internet server (see Chapter Six). A typical small business

would use the IP addresses in one of these groups (called a class C private network), which run from 192.168.0.1 to 192.168.255.255.

As an example, in a very small LAN of only five machines, the computer's IP addresses could be as follows:

Computer 1	192.168.0.1
Computer 2	192.168.0.2
Computer 3	192.168.0.3
Computer 4	192.168.0.4
Computer 5	192.168.0.5

The addresses need not start at 1—they could just as easily be 24, 25, 26, 27, and 28. However, it's a good idea to lay as simple and transparent a network foundation as possible.

The computers on a network share several other addresses, which are mathematically determined by their IP addresses. To give some examples, all the IP addresses in the range beginning with 192.168.0.X (such as the five previously listed addresses) would typically share the following other addresses:

Netmask: 255.255.255.0

Network: 192.168.0.0

Broadcast: 192.168.0.255

These are typical addresses for use in a class C private network and are used in this book.

Server Assigned or Client Assigned Addresses?

Should the server be in charge of handing out names and IP addresses to client machines, or should each client be in charge of identifying itself?

On a very small LAN, it's easier if every client identifies itself. In such a case, every computer on the local network, including the server, usually carries a record of every other computer's address. On Linux machines, these records are kept in the /etc/hosts file; on Windows PCs they're kept in the lmhosts or hosts file.

As the LAN grows, it becomes more and more work to keep all of these files on all computers up-to-date. Much of this work can be eliminated by putting the server in charge of assigning addresses to the clients. This is done by running a Dynamic Host Configuration Protocol (DHCP) server.

Each method has its advantages and disadvantages, as shown in Table 5-1.

Table 5-1. Client Assigned versus Server Assigned Addresses

METHOD	ADVANTAGES	DISADVANTAGES
Client assigned	Simplicity. Only hosts or lmhosts files need to be configured, and they're very simple to write. There is no complex server configuration.	Escalating maintenance. Every time a new computer is added to the LAN, configuration files on the server and any other computer it needs to communicate with must be updated. As the LAN or LANs grow, this can become rather impractical.
Server assigned	New computers can be added at any time with little or no configuration necessary. "Guest" computers, for temporary workers, visiting consultants, or clients, can connect immediately and have Internet access through the LAN.	Initial setup on the server is more complex and time consuming.

NOTE *This book suggests using a server assigned system using DHCP. Many consultants will set up DHCP as a matter of course, because that is the method they've found over time to produce the best results with the least maintenance overhead. It's more flexible, with much less work in the long haul, and server setup is made easier with the latest GUI setup tools and with the prewritten configuration files available for download from the Linux Leap Web site.*

The Scenario

Development, Ltd., has decided on dev.org for their domain name. Hostnames are assigned to each computer based on the place of each employee in the business.

With the latest expansion, the company has 16 computers, and the owner has no intention of letting the company grow larger than that. But to be on the safe side, she has chosen a large pool of addresses (considering the size of the company) to be prepared for any contingency. The company will have 40 IP addresses, for a class C private network, ranging from 192.168.0.1 to 192.168.0.40.

The server will be given the first address in the range: 192.168.0.1. The management group follows, starting with the CEO at 192.168.0.2, the domestic consultants, and finally the foreign consultants.

The result will look like Figure 5-5. The method of coming up with computer hostnames is clear (i.e., the CEO's computer is ceo.dev.org).

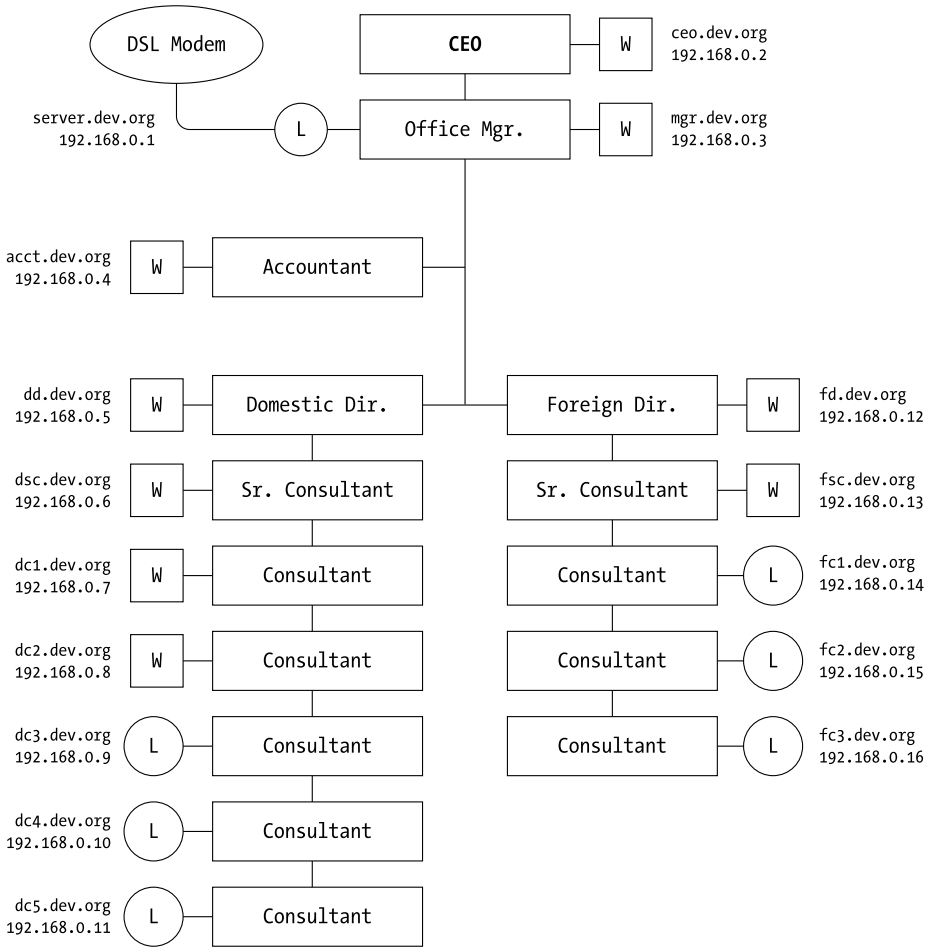

Figure 5-5. Hostnames and static IP addresses

Lesson 5-1 Review

This lesson was devoted to hardware and IP addresses. You learned

- The basics of connecting computers into a LAN with Ethernet cable, NICs, and hubs or switches; the specifications to demand; and purchasing tips

- The difference between a hub and a switch, the advantages and disadvantages of both, and some typical price ranges
- The use of the ifconfig command to check whether Linux has detected your NIC or PC Ethernet card
- Methods of deriving host and domain names
- The most common range of IP addresses to use for a small LAN
- The advantages and disadvantages of client assigned and server assigned addresses

LESSON 5-2.

SETTING UP A DYNAMIC SERVER AND CLIENTS

This lesson shows how to set up a Dynamic Host Configuration Protocol (DHCP) server that will provide both static and dynamic IP addresses to clients on the network. This is the first, and most fundamental, software configuration job you need to accomplish to get the LAN up and running. System administrators will recognize this lesson as enabling TCP/IP networking, which is the foundation for all subsequent network connections.

A DHCP server can be configured to always assign a fixed (static) IP address to a certain computer, or it can be configured to assign one of several addresses chosen from a pool (dynamic) to any computer that requests one. Dynamically assigned addresses represent the lowest long-term maintenance solution, but can be a significant amount of work to set up for Linux client machines permanently on the LAN (see Appendix B for an explanation and instructions). Therefore, this book suggests a mixed approach for the small business or workgroup: Each fixed computer on the LAN is given its own fixed IP address (assigned by the server), while "guest" computers are assigned addresses from a pool.

This scheme ensures a maximum of stability and intercommunication within the LAN, while allowing guest computers to log onto the Internet with minimal client and no server configuration.

This lesson covers server configuration first, and then Linux client and Windows client machines.

Server Configuration

IN BRIEF

1. Use the Network Configurator tool in GNOME to configure the server's hostname, local IP address, and local Ethernet interface.
2. Configure Linux to start the DHCP server upon computer start-up by checking the dhcpd option in the System Services menu of the Setup tool.
3. Download the dhcpd, dhcpd.conf, and dhcpd.leases files from `http://www.linuxleap.org`.
4. Copy the files into their correct directories:
   - /etc/init.d/dhcpd
   - /etc/dhcpd.conf
   - /var/lib/dhcp/dhcpd.leases
5. If needed, edit the dhcpd file to insert the correct Ethernet interface.
6. Determine the MAC addresses of Linux clients with ifconfig, of Windows 98 clients with winipcfg, and of Windows 2000 clients with ipconfig.
7. Edit the dhcpd.conf file, inserting your own client hostnames, MAC addresses, and IP addresses.
8. Start the server with the "./etc/init.d/dhcpd start" command.

EXPLAINED

Configuring Addresses, Hostnames, and Ethernet Interfaces

Before anything else, you should configure the server's hostname and IP addresses, and the local Ethernet interface.

Open the Network Configurator in GNOME by clicking the main menu, navigating to Programs ➤ System ➤ Network Configuration, and entering the root password if necessary. Click the DNS tab and enter the server's hostname and

domain in the appropriate boxes. Enter the domain name alone in the DNS Search Path box, and finally, add the server's local IP address in the Primary DNS text box. The result for the scenario company, Development, Ltd., would look like Figure 5-6. The DNS section of the Network Configurator writes the /etc/resolv.conf file for those who want to edit this setting manually.

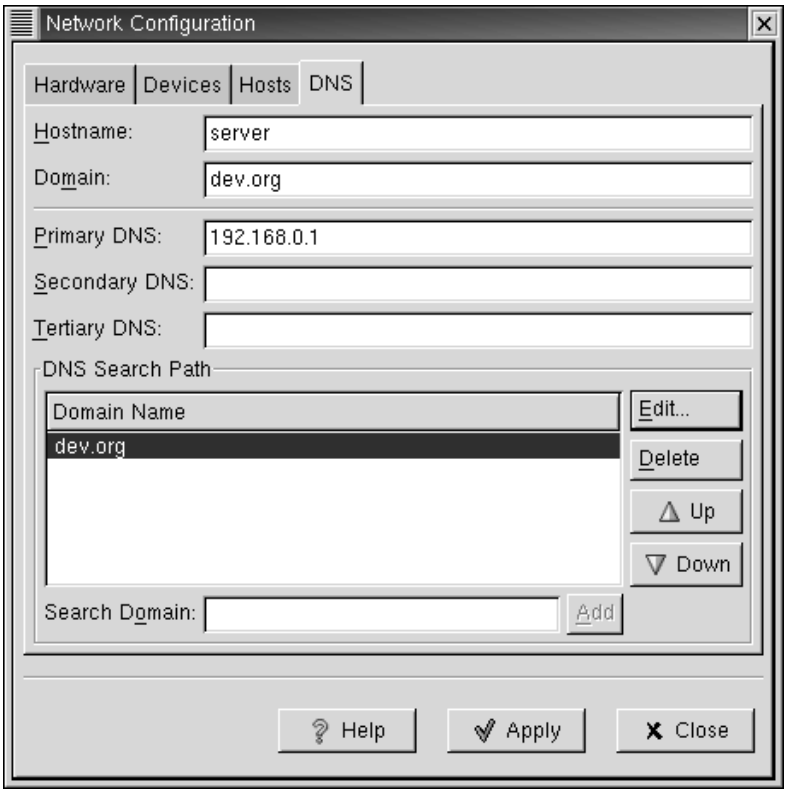

Figure 5-6. Server setup in the DNS window of Network Configurator

Now click the Hosts tab. The /etc/hosts file will serve as a backup identification for the computer on the network. Clicking the Add and Edit buttons allows the user to enter the server's hostname and local IP address. Only two entries should appears: The server itself and the localhost entry, which is automatic, always appear and always must appear. Figure 5-7 shows how this would look on the scenario company's server.

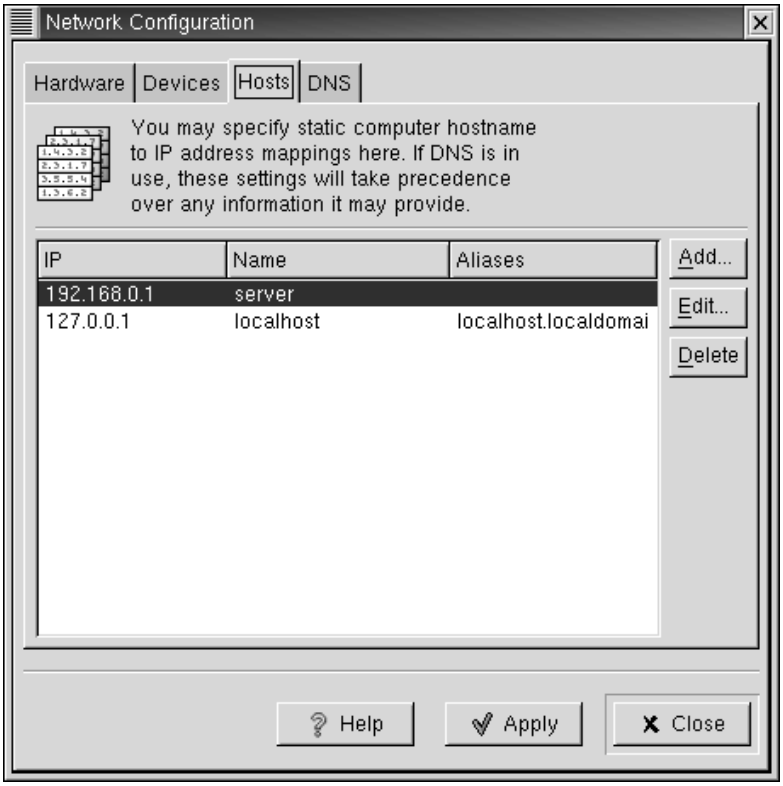

Figure 5-7. The Development, Ltd., server's /etc/hosts file in Network Configurator

Finally, you must configure the local Ethernet interface. On a small business Linux server, there are likely to be two Ethernet interfaces: one for the local network and one for the connection to the ISP. The interfaces will appear as follows:

eth0	The first interface
eth1	The second interface

Click the Devices tab on the Network Configurator. The interface for the local network should have been configured during installation, but if it wasn't, simply add or edit the interface by clicking the appropriate button. Clicking the Edit button brings up the Ethernet Device screen; under the General tab, check the "Activate device when computer starts" box; under the Protocols tab, select TCP/IP; and then click the Edit button. In the TCP/IP Settings screen, select the TCP/IP tab and fill in the information in the Manual IP Address Settings area. This will be the IP address of the server—192.168.0.1, in the example—the standard class C subnet mask, 255.255.255.0, and the default gateway address (the same as the IP address). Figure 5-8 shows the Devices, Protocols, and TCP/IP screens.

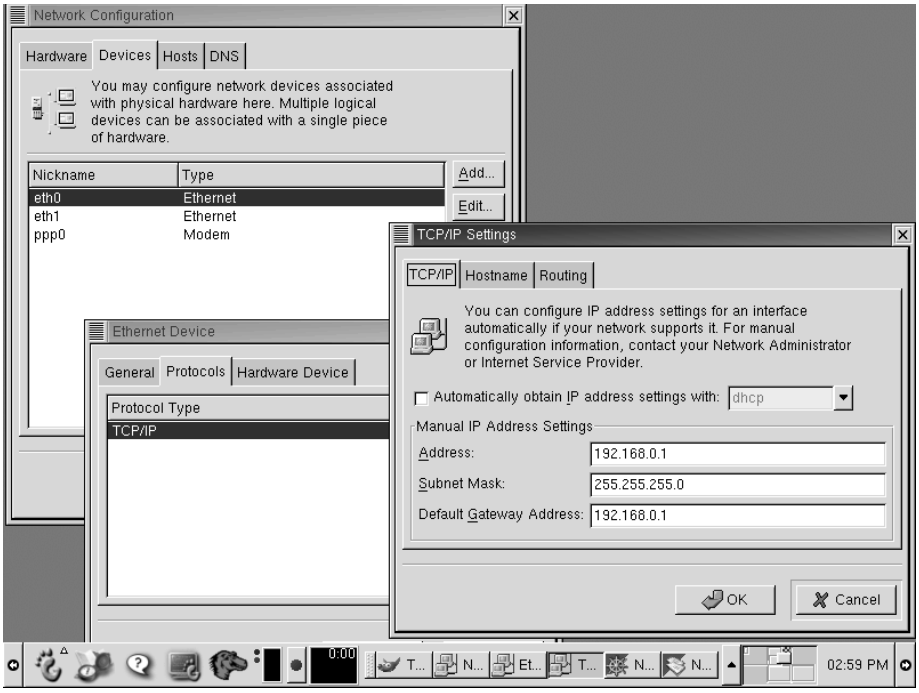

Figure 5-8. Configuring the server's local Ethernet interface

Make a note of your local Ethernet interface; you'll need it a little later.

Configuring the DHCP Server to Start at Boot-Up

The Services part of the Setup tool configures which services are started when the computer boots up. As part of the routine to start the DHCP server, Setup places a start-up file for the server into the /etc/init.d directory. It's necessary to perform this step before copying the identically named file from Linux Leap into that directory; otherwise, the Setup tool will overwrite the file you've just downloaded and copied.

You can start the Setup tool from the main menu by navigating to Programs ➤ System ➤ Text Mode Tool Menu. (Note: You need to be logged in as the root user.) Then open Services by tabbing down to System Services and clicking the Run Tool button.

An alternative way to start the Service tool within GNOME, if you're running as the root user, is to click the terminal icon to open a terminal window, type **ntsysv**, and then press the Enter key.

Make sure the dhcpd box is checked (see Figure 5-9). Tab to the OK button, press the Enter key, and close the terminal window.

Figure 5-9. Configuring the server to start DHCP upon booting

Downloading and Copying the Files

Download from the Linux Leap Web site the three files mentioned previously and copy them to their correct directories on your Linux server:

- /etc/init.d/dhcpd
- /etc/dhcpd.conf
- /var/lib/dhcp/dhcpd.leases

Downloading and Copying Linux Leap Files

The simplest method is to download the files to the directory they belong in on your Linux server. However, that won't be possible if the server's Internet connection is not yet up and running. It may be necessary to copy the files to the server from another machine (possibly a Windows machine) via floppy disks.

Windows-formatted floppy disks work fine in Linux. You can copy files to and from Windows floppies in a Linux terminal using the mcopy command, or you can drag and drop files using either the GMC or Nautilus file manager in GNOME.

One thing to remember: When you use floppy disks within GNOME, you should always unmount the disk before you eject it.

As an example, say you've copied the dhcpd file onto a Windows floppy disk, and you want to transfer the file to the correct directory on your Linux server. You would follow these steps:

1. Insert the floppy disk into the Linux computer.
2. In GNOME, click once on the floppy disk icon on the Panel. This mounts the floppy—the floppy icon will change color and a separate floppy icon will appear on the desktop, indicating that the disk is mounted. Double-clicking the desktop icon opens the file manager, which automatically reads the contents of the floppy disk.
3. Click Tree on the file manager's left panel, find the /etc/init.d directory, and use the mouse to drag and copy the file.
4. Close the file manager and click the floppy disk Panel icon once again. This unmounts the disk.
5. Wait a few moments and then eject the floppy.

Of the three files you download, you may need to edit two. The third file, dhcpd.leases, starts life as a completely empty file and is automatically configured by the program itself. The dynamic server, however, won't start unless this file is present. There are two ways of creating it: You can either download it from the Linux Leap site, as described previously, and copy it into the correct directory, or you can simply type, at a terminal, **touch /var/lib/dhcp/dhcpd.leases** and press the Enter key. This will automatically create an empty dhcpd.leases file.

Editing the dhcpd File

The dhcpd file starts the dynamic server when the computer boots up. There is only one line that you may need to edit. It concerns which Ethernet interface your local network is attached to.

You should have previously made a note of this interface in the section titled "Configuring Addresses, Hostnames, and Ethernet Interfaces." The downloaded dhcpd file is set by default to eth0. This is the first Ethernet interface, and it's the preferred one for the local network.

If, however, you need to edit this setting, open the file with the gedit editor within GNOME by logging in as root, clicking the Start button, and navigating to Programs ➤ Applications ➤ gedit.

Within gedit, click the Open icon and navigate to /etc/init.d/dhcpd. The line you may need to edit is shown in Figure 5-10.

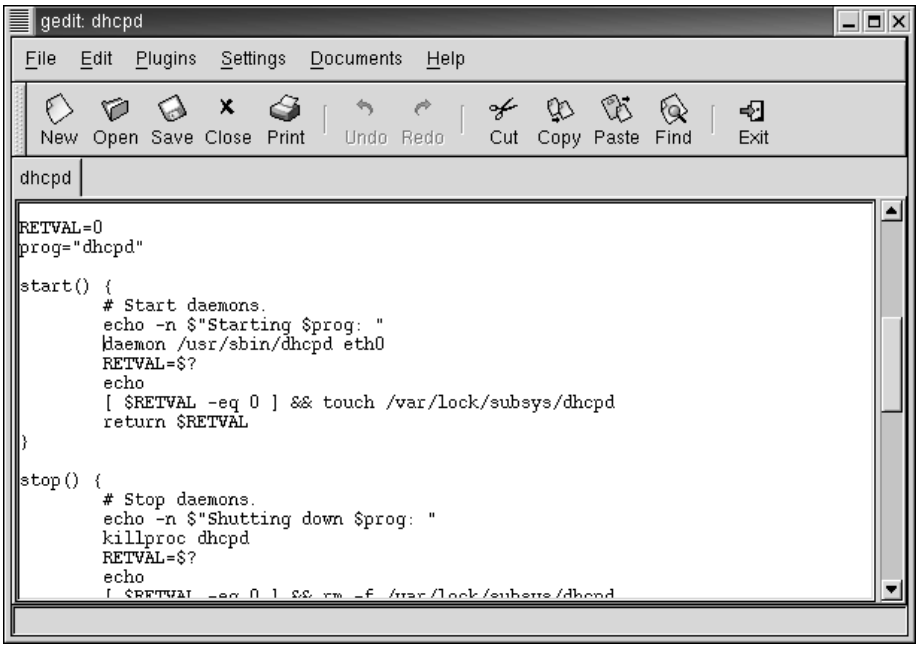

Figure 5-10. Editing the local network interface

Only one small change may need to be made. If your LAN interface is eth1, instead of etho, change the line:

```
daemon /usr/sbin/dhcpd eth0
```

to

```
daemon /usr/sbin/dhcpd eth1
```

Then save the file and exit the editor.

Determining the MAC Addresses on the LAN

The DHCP server assigns static addresses to computers based on each computer's Media Access Control (MAC) address. This is nothing more nor less than the Ethernet card's unique hardware number. Therefore, the system administrator must determine each machine's MAC address. This is quite simple, but it's performed differently on each OS.

In Linux

Open a terminal and type **ifconfig**. The numbers on the line that starts with eth0, and appear after HWaddr, are the MAC address. They are always six pairs of numbers or numbers and letters, separated by colons. An example is 00:00:86:37:55:52. (See Figure 5-4.)

In Windows 2000

Open a command prompt and type **ipconfig /all**. The MAC address is on the line marked Physical Address. An example is 00-80-C6-F9-0B-66 (see Figure 5-11).

In Windows 98

Click the Start button, select Run, and enter **winipcfg**. Look for the Adapter Address under the correct network interface.

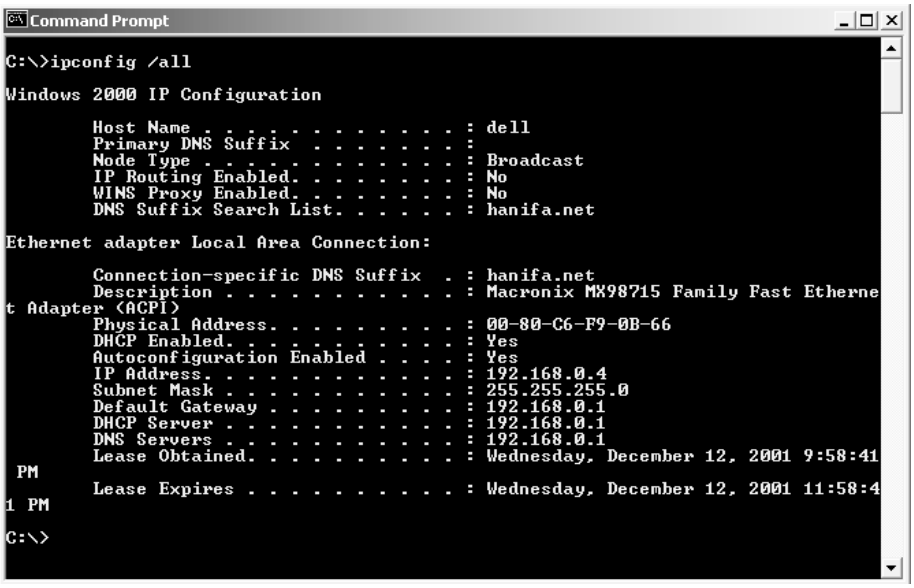

Figure 5-11. Finding Ethernet card addresses in Windows 2000 with ipconfig /all: physical address near middle of screen

Editing the dhcpd.conf File

The main configuration file for the DHCP server is dhcp.conf. It determines the range of addresses to use and which fixed addresses from the range will be assigned to which computers.

The file as downloaded from Linux Leap is configured for a range of IP addresses starting with 192.168.0.1—the server—and going on to 192.168.0.40. The file has 20 prewritten "host" sections, for computers with static addresses. The host and domain names, the MAC addresses, and the IP addresses need to be supplied for each.

If each of the host sections were used—if the LAN had exactly 20 computers with fixed addresses—and the administrator followed a rigid system of address allocation, the result could be as follows:

192.168.0.1	Server
192.168.0.2 to 192.168.0.20	Range of fixed addresses on the LAN
192.168.0.21 to 192.168.0.40	Range available for dynamic addresses

The range of addresses available for dynamic allocation can be expanded to take advantage of unused addresses in the static range. Thus, in the Development, Ltd., scenario, with only 16 computers in the firm, 23 addresses from the range of .17 to .40 could be available for dynamic allocations to guest machines or other additions to the LAN.

NOTE *Of course, you can modify the prewritten /etc/dhcpd.conf file to suit any range of addresses you like. The IP and other addresses suggested here are typical for a modest class C private network. Note, however, that there are strong relationships between IP, network, netmask, and broadcast addresses. Those readers who are interested in learning more may with profit turn to The Linux Networking HOWTO (*`http://www.linuxports.com/howto/networking/`*).*

The file can be edited in gedit within GNOME. If the standard IP address range is acceptable, only the sections in capital letters and capital Xs need to be edited. (Note: Unused sections should be deleted from the file.) Figure 5-12 shows an example of the first part of this file, edited to reflect the Development, Ltd., scenario. In this example, the company's domain name has been entered, as well as the host and domain names of the first three client computers, their MAC addresses, and their complete IP addresses.

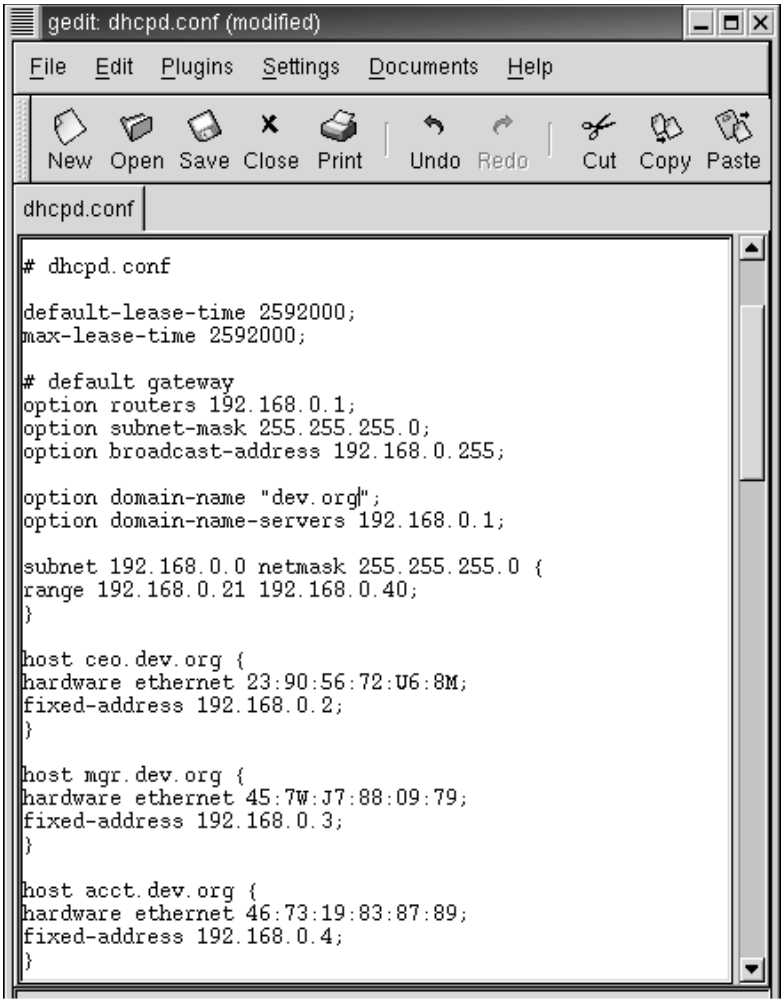

Figure 5-12. Inserting names and MAC and IP addresses in dhcpd.conf

Starting the Server

This is the simplest step of all. Log in at a terminal as root, type **/etc/init.d/dhcpd restart**, and press the Enter key. The server should start up, giving you an OK message.

TIP *It is seldom necessary to reboot a Linux server to start a new or reconfigured program or service. Scripts to start most services are in the /etc/init.d directory. You can start, stop, and restart most servers in the same way as the DHCP server.*

Linux Client Configuration

Although much of the following configuration will already have been done during installation, it's a good idea to do a postinstallation check and update and correct settings as necessary.

IN BRIEF

1. From within GNOME, open the Network Configurator, and from within DNS, enter the client's hostname, primary DNS, and domain.

2. Within Hosts, enter the server's IP address and name.

3. Go to Devices ➤ Protocols ➤ TCP/IP and configure the interface to automatically obtain IP address settings with DHCP.

EXPLAINED

You can completely configure a Linux desktop's network settings from within the Network Configurator program in GNOME.

Entering the Client's Hostname, Primary DNS, and Domain

Open the program by clicking Start and navigating to Programs ➤ System ➤ Network Configuration. Enter the root password if prompted.

Make sure that the client computer's hostname is entered in the DNS window (e.g., ceo). Beneath it, next to Domain, enter the LAN's domain name (e.g., dev.org). Enter the server's IP address next to Primary DNS, and finally, under DNS Search Path enter the domain name again. This will look very similar to Figure 5-6. Enter the server's hostname and IP, and enable DHCP.

Next, under Hosts, enter the local server's hostname and IP address, using the Add and Edit buttons as necessary. This will look exactly the same as Figure 5-7.

Finally, under Interfaces, select the Ethernet interface and click the Edit button to reach the Ethernet Device window. Check the "Activate device when computer starts" button. Then click the Protocols tab and the Edit button to reach the TCP/IP window. Check the "Automatically obtain IP address settings with

DHCP" box. You can leave all other boxes blank. Figure 5-13 shows an example. Note the difference between this TCP/IP window and the one shown in Figure 5-8 for the server.

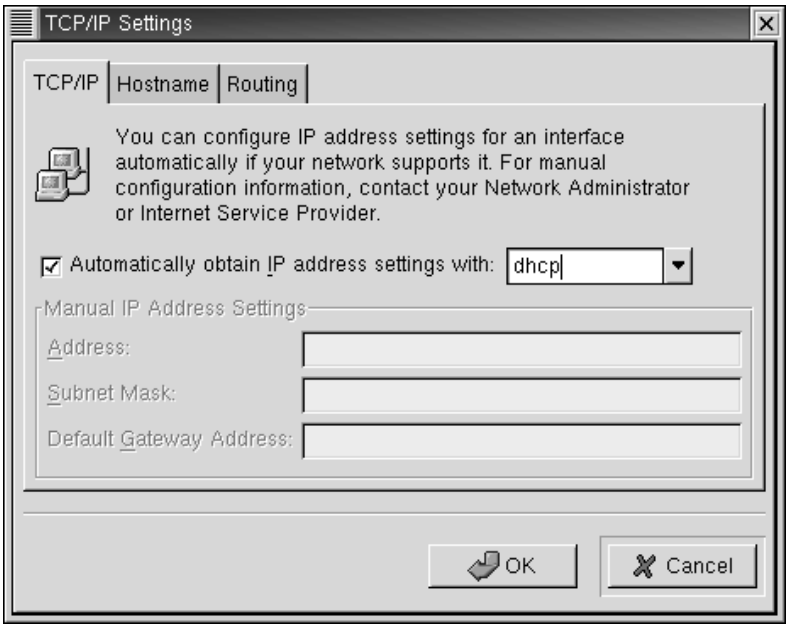

Figure 5-13. Configuring DHCP on the Linux client side

Windows 2000 Client Configuration

Windows 2000 defaults to DHCP, which reduces configuration time in many cases to near zero. Also, Windows configuration is so familiar to so many users that it's covered here only in the "In Brief" section.

Windows 98 is still in common enough use to warrant a separate section.

IN BRIEF

1. Click the Start button and navigate to Settings ➤ Network and Dial-up Connections.

2. Right-click the Local Area Connection icon and choose Properties from the pop-up window.

3. Click Internet Protocol (TCP/IP) and the Properties button. Your screen should now resemble Figure 5-14.

4. As in Figure 5-14, check the "Obtain an IP address automatically" and "Obtain DNS server address automatically" options.

5. Exit by clicking the OK buttons.

6. Click the Start button again and navigate to Settings ➤ Control Panel.

7. Double-click the System icon, choose the Network Identification tab, and click the Properties button.

8. Type in the computer's hostname in the Computer Name box and its domain name in the Workgroup box.

9. Exit again by clicking the OK buttons.

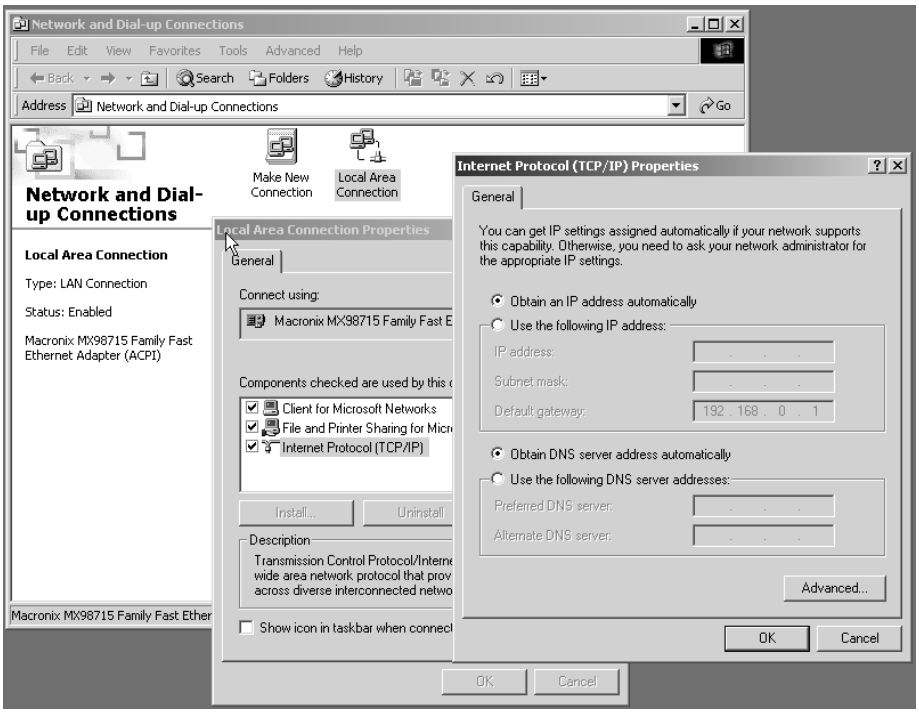

Figure 5-14. Configuring a dynamic address client in Windows 2000

Windows 98 Client Configuration

IN BRIEF

1. Click the Start button and navigate to Settings ➤ Control Panel.
2. Double-click the Network icon. Select the line that begins with "TCP/IP" and continues with a description of your Ethernet network card, and click the Properties button.
3. Check the "Obtain an IP address automatically" option.
4. Exit to the main Network window by clicking the OK button. Choose the Identification tab and enter the computer's hostname in the Computer Name box and its domain name in the Workgroup box.
5. Exit by clicking the OK buttons.

Testing the Connection

At this stage, the client computers can talk to the server only with IP addresses, not names. To test this fundamental level of networking, use the ping command. This small program sends a tiny packet of information that asks for an *echo,* or, in other words, a return receipt. This is a command line–only program, both in Linux and Windows, and all it needs is the IP address of the machine you're trying to ping.

From a terminal in your Linux Server, type **ping *(IP address of client)***. As an example, to test the connection to a client named ceo.dev.org with an IP address of 192.168.0.2, you would type **ping 192.168.0.2** and press the Enter key.

If successful, ping will produce an apparently endless series of connection messages. Stop the program by simultaneously holding down the Ctrl and c keys. If unsuccessful, ping will return a Destination Host Unreachable error message, which may have to be stopped with the Ctrl+c key combination.

TIP *The ping test works exactly the same way in Linux and Windows clients. In Windows, open a command prompt and enter the ping command with the IP address of your server.*

Lesson 5-2 Review

This lesson was devoted to setting up a dynamic IP address server and clients. You learned

- How to configure a Linux DHCP server's Ethernet interface
- How to configure the server to start at boot-up
- Which files need to be downloaded from Linux Leap and where they need to be copied
- Why you may need to edit the dhcpd and dhcpd.conf files, and how to do so
- How to start the server without rebooting
- How to configure a Linux client by entering its server's name and address, and how to configure the client for DHCP
- How to configure DHCP assigned addresses in both Windows 2000 and Windows 98
- How to use the ping command to test network connectivity from either Linux or Windows

LESSON 5-3.

SETTING UP A DOMAIN NAME SYSTEM SERVER FOR THE LAN

Why a domain name system (DNS) server for a small firm's LAN? Because, although it is more work up front, it greatly increases the LAN's flexibility. It enables maximum communication—including e-mail—between all computers on the LAN and makes Web and e-mail serving to the Internet very simple to implement. It also increases LAN stability and reduces maintenance.

What a DNS Server Does

A DNS server resolves computer hostnames to IP addresses (and vice versa). When any computer on the network tries to get in touch with another computer by using its hostname, the DNS server matches the name with the IP address, thus

enabling the communication to take place. The server does not just resolve hostnames on the local LAN: When it's unable to find a computer locally, it directs enquiries to "root servers" on the Internet, and it then "caches" the results in memory, so that when a computer on the LAN needs to resolve the same address again, the correct IP address comes from the local DNS server. This can greatly increase response time.

Thus, a local DNS server acts as a nameserver for the company's domain on the LAN. However, this can also extend to the Internet. In order to be registered, every domain needs two nameservers to match up the domain's IP address with its name. Although a firm does not need to host its own nameservers, it does give the firm more control.

Traditionally, setting up DNS meant a great deal of work with several text configuration files of abstruse design. Fortunately, GNOME's BIND Configuration Tool greatly simplifies this job for small firms.

The files to be configured are several "zone" files in /var/named/ and one general configuration file, /etc/dhcpd.conf. The zone files (all end in ".zone") match local computer names with addresses (forward master zones) and addresses with names (reverse master zones). In addition, one file, /var/named/named.ca, directs queries that can't be resolved locally to the Internet's root servers.

A small firm's administrator does not need to touch one of these files with a text editor. The administrator can configure them all from the BIND Configuration Tool. ("BIND" is an abbreviation for Berkeley Internet Name Domain, the actual name of the server program.) Usage of this tool will be a mystery to the uninitiated. Some scanty help is provided in the Red Hat Customization Guide, which is available on the documentation CD and on the Red Hat Web site. However, short of a lengthy tutorial, the best way to show how the tool works is to give an example. The Development, Ltd., scenario will be used.

Configuring a DNS Server for Development, Ltd.

The following operations take place on the scenario company's Linux server. You can access the configuration tool by clicking the Start button and navigating to Programs ➤ System ➤ DNS Configure DNS.

Configuring the Forward Master Zone for dev.org

To configure the forward master zone for dev.org, click the Add button, check the Forward Master Zone option, and enter the domain name **dev.org**.

This procedure creates the forward master zone file for the local domain. To be of use, every computer on the network must be listed in it. First, however, identify the LAN's nameserver by selecting dev.org under Records and clicking the Edit button.

In the Settings window, click the Add button and type in the fully qualified domain name (FQDN)—the hostname plus the domain name—of your server. In Development, Ltd.'s case, it is **server.dev.org**. Click the OK button, and enter the server's IP address at the bottom of the window—for the scenario, **192.168.0.1** (see Figure 5-15). Click the OK button to return to the main Master Zone window.

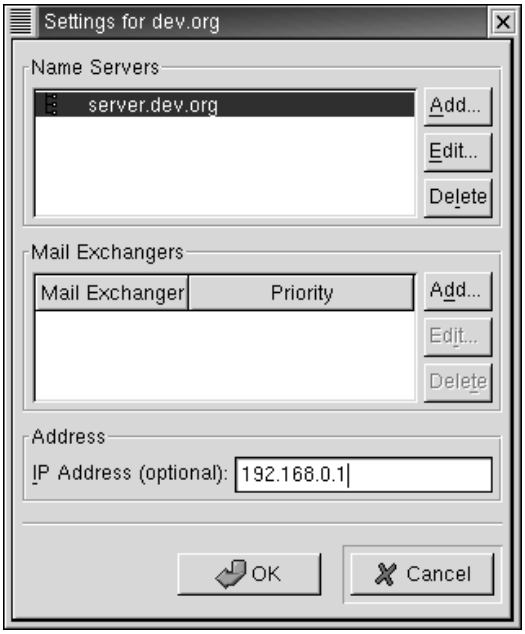

Figure 5-15. Setting up dev.org's nameserver on server.dev.org

Now the tedious part: Each computer has to be entered under Records. Start by clicking the Add button, selecting Host as the type of record resource, and typing in the first client machine's hostname and IP address (see Figure 5-16).

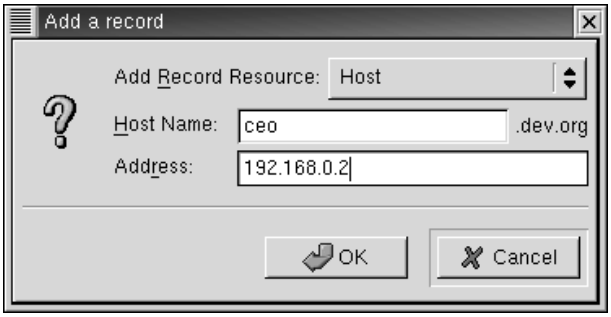

Figure 5-16. Adding a host record—the scenario company's first client name and address

You need to add each client computer in this way. When you've finished, the hostname and IP address of each machine will show up in the Records pane, as shown in Figure 5-17. Exit this window by clicking the OK button.

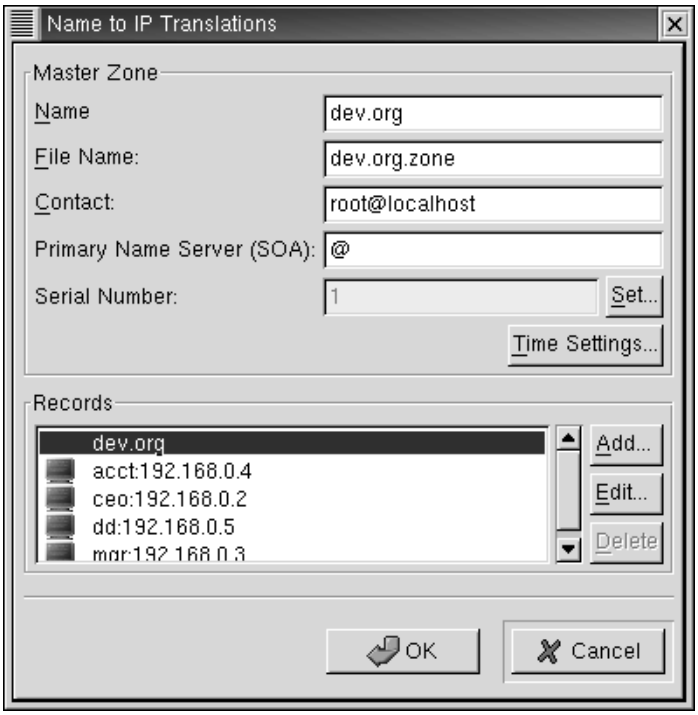

Figure 5-17. Each client appearing under Records

Configuring the Reverse Master Zone for dev.org

Now you must create the reverse master zone file. Go back to the very first window and click the Add button. This time, choose Reverse Master Zone and enter the first three sections (octets) of the server's IP address. For the scenario, the administrator would enter **192.168.0**. Click the OK button.

In the IP to Name Translations window that appears, click the Add button to add the local nameserver. Simply type in the FQDN of the server—in the scenario, this is **server.dev.org.** Note that in this box you must include a period (or "full stop" for British, Irish, and Indian readers) after the domain name.

Next, you must add every computer on the LAN, including the server, to the Reverse Address Table, much as you added them earlier. Click the Add button next to the bottom panel and enter the last "octet" of each machine's IP address, along with its FQDN. Note that here, also, a period must be added to the end of the name.

When you've finished, the window will look something like Figure 5-18.

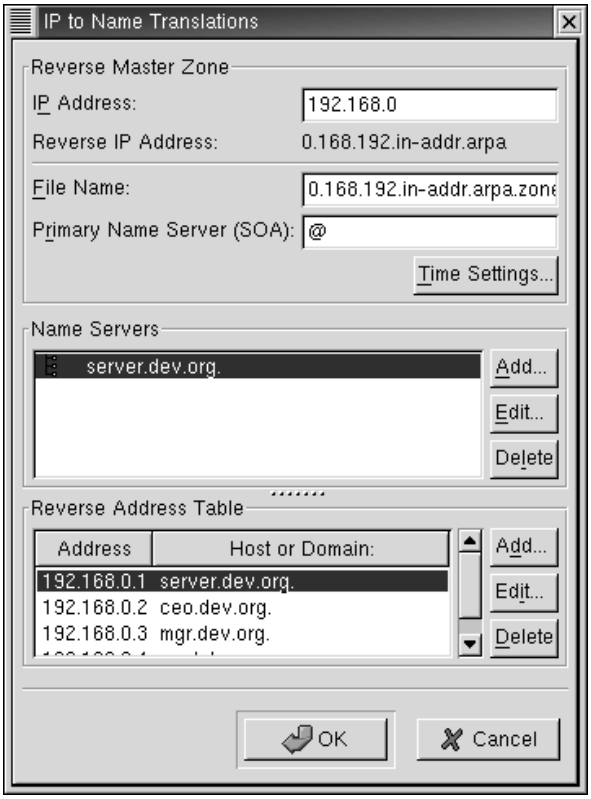

Figure 5-18. The IP to Name Translations window for the scenario. Each machine on the LAN appears in the Reverse Address Table.

Closing the configuration tool by clicking the X in the upper right-hand corner will automatically write all the configuration files to disk.

Client Configuration

The client configuration for DNS is very simple, and it's covered in this section in brief for all three types of client in the scenario.

Linux Clients

IN BRIEF

1. Using the Network Configurator tool, make sure that in the DNS window the local server's IP address is entered in the Primary DNS box and the domain name is entered under DNS Search Path.

2. In the Hosts window, make sure that the server's hostname and IP address are entered.

Windows 2000 Clients

IN BRIEF

1. Click the Start button and navigate to Settings ➤ Network and Dial-up Connections.

2. Right-click Local Area Connection and choose Properties.

3. Select Internet Protocol and click the Properties button.

4. Check the Use the Following DNS Server Addresses option, and enter the local server's IP address.

Windows 98 Clients

IN BRIEF

1. Click the Start button and navigate to Settings ➤ Control Panel.
2. Double-click the Network icon.
3. Under the DNS Configuration tab, check Enable and enter the client machine's hostname and domain.
4. In the same screen, add the local server's IP address under DNS Server Search Order (see Figure 5-19).

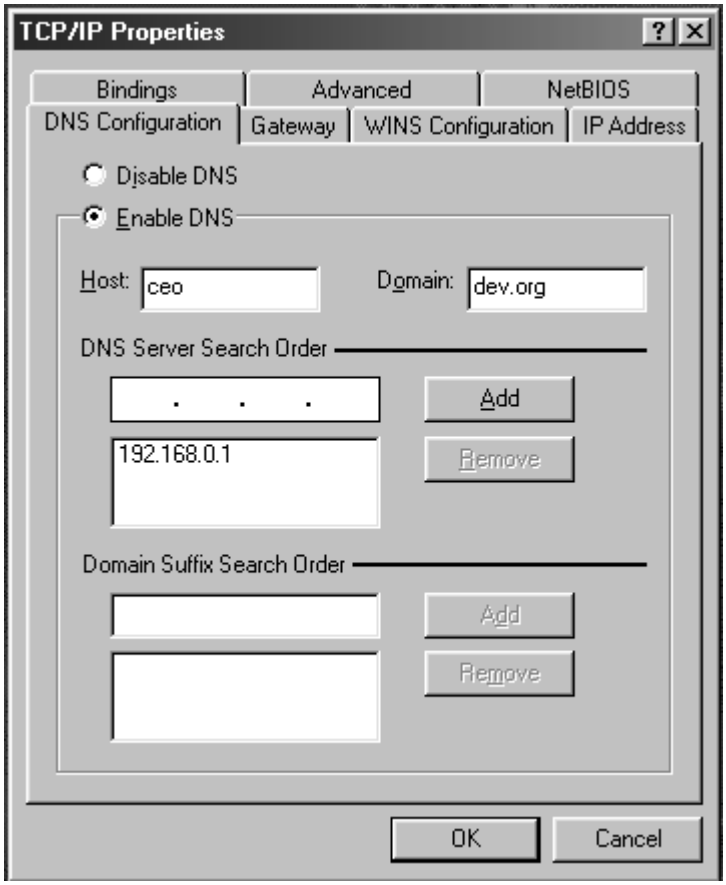

Figure 5-19. Configuring DNS in Windows 98

Starting DNS

You should configure BIND to start at system boot. To do so, start the Setup tool at a terminal and enter the System Services section. Check the "named" entry. To restart BIND at any time, type **/etc/init.d/named restart**.

Testing the Connection

With DNS working, all the computers should be able to resolve each other's names into IP addresses and vice versa. To test this, use the same program you used in the previous lesson, the ping program, but try it using the hostnames of the computers on the LAN.

For instance, from a terminal in your Linux server, type **ping *(hostname of client)***. As an example, to test the connection to a client named ceo.dev.org, you would type **ping ceo** and press the Enter key.

This procedure should work from and to any machine on the LAN, including Linux to Windows and Windows to Linux.

Lesson 5-3 Review

This lesson was devoted to setting up BIND, the domain name server. You learned

- What a name resolver is, and why it needs to be used
- How to use the BIND Configuration Tool to configure the forward and reverse master zone files
- Client configuration for DNS
- How to configure the BIND program to start at system boot-up
- How to restart and test DNS

What's to Come

The next chapter of the book builds on the foundation presented in this chapter. Topics covered are

- Setting up an Internet server for the LAN
- Setting up a file server
- Setting up a print server
- Setting up a local e-mail server
- Setting up a local Web and database server