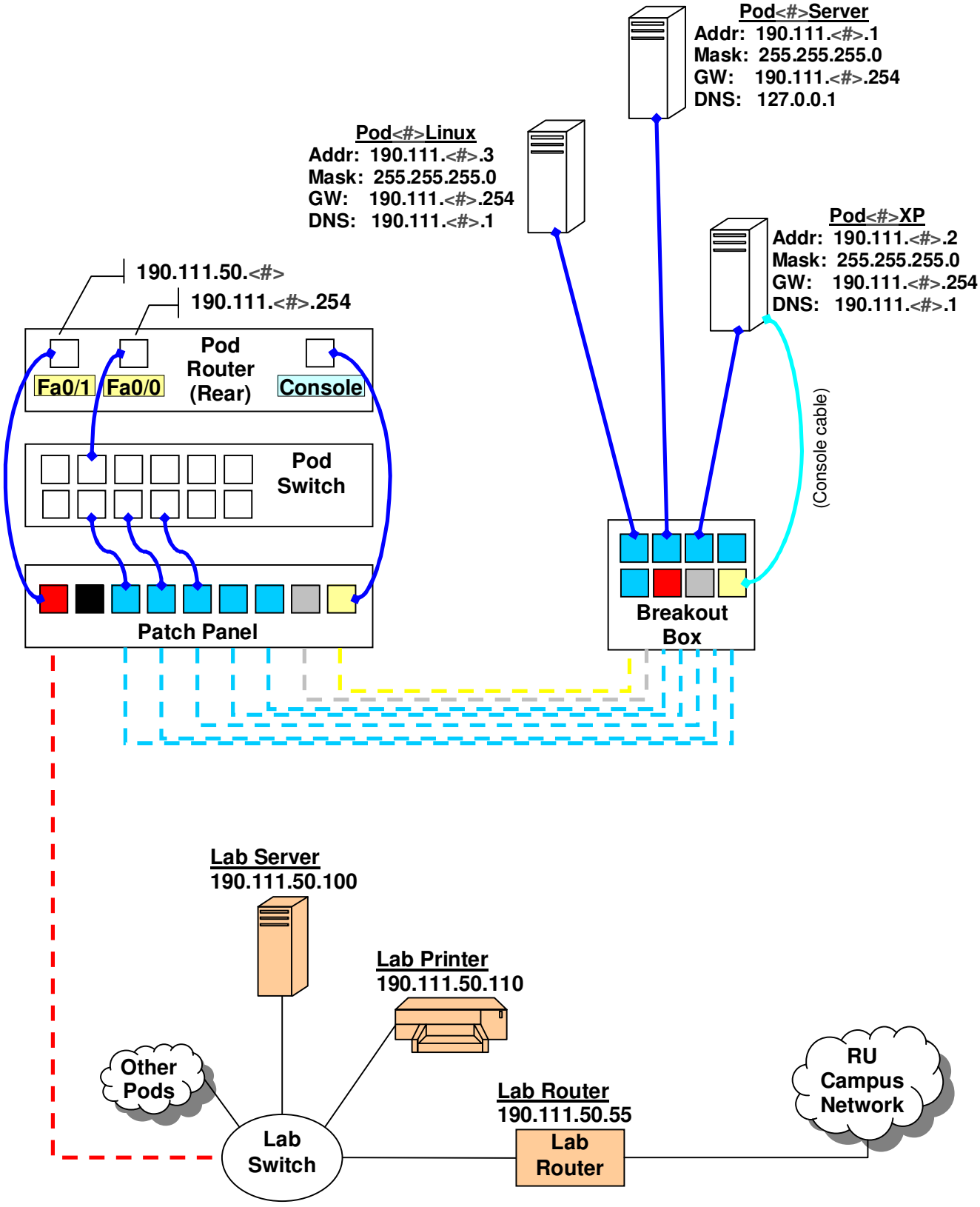
Networks Lab Pod Diagram

<#> = your pod number. (e.g., pod<#> = pod1)



Cabling the Network (see pod diagram)

Pod Cabling

Each pod has three computers that must be connected by Ethernet cable to that pod's switch in the router room. This is made possible by the breakout box positioned at the corner of each pod. **The jacks on the breakout box are connected to the patch panel in the router room by cables that run beneath the floor**.

1. Connect each machine's Ethernet port to the breakout box.
    a. Connect a standard Ethernet cable from the network card of each machine to the breakout box in the corner of the desk. Start with the first (top left-most) jack on the breakout box.
    b. If your machine has two Ethernet ports, try the top-most port first. If the operating system fails to detect a connection, try the other port.
2. Connect the serial console cable to the XP machine
    a. The console cable is the flat, light-blue cable with the serial connector on one end.
    b. Connect the serial side of the cable to the serial port on the Windows XP machine.
    c. Connect the other end of the console cable to the yellow jack on the breakout box.

Switch and Router Cabling

The racks in the router room contain a switch, router, and patch-panel for each pod. To be able to communicate with each other, each computer and the router must be connected to the switch. To communicate with other pods and the Internet, the router must be connected to the lab backbone network.

1. Connect each system in the pod to the switch.
    a. Connect the first three blue jacks on the patch panel to the switch.
    b. Note: all of the ports on the switch are the same. You may use the ports of your choice.
2. Connect the router to the pod network and the backbone network.
    a. Connect the Fa0/0 port on the back of the router to a port on the switch.
    b. Connect the Fa0/1 port to the red jack on the patch panel.
3. Use a standard cable to connect the console port on the router to the yellow jack on the patch panel.

Installing Windows XP and Server 2003

Startup

Before starting the installation, you must select the correct machine for the operating system. Each machine has a small green label on the top with a license key for Windows XP Pro. One machine has a white label with the key for Windows 2003. Although any machine can be used for Windows XP, you should use the Dell GX620 model for Windows because the GX620 is not compatible with Fedora Core 3 Linux.

1. Select the system on which to install Windows. For Windows 2003, use the system labeled with the white "Win2003 Key" label. For Windows XP use a GX620 if it is available.
2. Insert the Windows installation CD and reboot the machine.
3. Press Enter to boot from the CD when prompted (you will have only a few seconds).
4. Do not select the options to install additional drivers or run a recovery tool.
5. When prompted to **Setup Windows**, press Enter.

Partitioning and Formatting

At the disk partitioning screen you will be shown a list of the existing disk partitions. You will need to completely remove all existing partitions and create a new one. After partitioning and formatting you will be asked to restart the machine.

1. Select and delete each partition.
2. Create a new partition and use the default size that it displays (the whole disk).
3. Select the new partition and press Enter to install Windows.
4. When prompted to format the drive, select **NTFS (Quick) for both XP and Server 2003**.

Installation Options

After some time, you will be prompted to set various options for Windows. You will be asked to set your computer's name. The Windows 2003 server should be named **Pod<#>Server**, the XP machine should be **Pod<#>XP**. For example, if you are in Pod 1 then your computers should be named "Pod1Server" and "Pod1XP." You will also be prompted for a password for the local Administrator account, use **pod<#>pod<#>** (e.g. pod1pod1).

1. Regional Settings: No change is required, click Next.
2. Name and Organization: Set both to **RU**
3. License Key: Use the key on the small green label (XP) or the large white label (2003)
4. License Mode (2003 only): No change is required, click Next.
5. Computer Name: **Pod<#>Server** or **Pod<#>XP**
6. Administrator Password: **pod<#>pod<#>**
7. Date and Time: You must set the date and time accurately or you may run into problems later when working with Active Directory. Do not forget to set the time zone to Eastern.
8. Network Settings (2003 only): Select Typical.
9. Workgroup/Domain (2003 only): No change is required, click Next.
10. Select No/Finished if asked to check for updates.
11. Create a user (XP only): name = **XPuser**, password = **pod<#>pod<#>**.

Installing Drivers

After Windows is installed, the drivers for the network and graphics card will need to be installed. The "Network Lab Drivers" CD has drivers for each system model used in the lab. The model numbers are printed on the front of the machine around the power button.

1. Find the folder on the Drivers CD that matches the model of your machine.
2. Double click on each of the files inside of that folder to install the drivers.
3. If asked to reboot, say no until all of the drivers are installed.
4. If you are told that a version of the driver is already installed, install the driver anyway.

Configuring Network Settings

Each pod will use IP addresses assigned from the 190.111.<pod #>.0 network. For example, Pod 1's 2003 server will be 190.111.1.1, XP will be 190.111.1.2. Although it is not setup yet, the Windows 2003 server will be the DNS server for your pod. To make the Windows 2003 server use itself for DNS, you can use the loopback address (127.0.0.1) for its DNS server. Your pod's router will be the default gateway for the network.

1. Open **Start->Control Panel** and click **Switch to classic view**.
2. Open **Network Connections->Local Area Connection**
3. Set the IP address and network settings.
    a. Click **Properties**.
    b. Double-click on **Internet Protocol (TCP/IP)** at the bottom of the list.
    c. IP address: **190.111.<pod #>.1** (Windows 2003) or **190.111.<pod #>.2** (Windows XP).
    d. Subnet mask: **255.255.255.0**. (do not use the default subnet mask of 255.255.0.0)
    e. Default gateway: **190.111.<pod #>.254**.
    f. DNS server: **127.0.0.1** (Windows 2003) or **190.111.<pod #>.1** (Windows XP).
4. Disable the firewall (XP only):
    a. Open **Start->Control Panel->Security Center->Windows Firewall**.
    b. Select **Off**.

Installing Fedora Core 3

Startup

Since Linux does not require a license key, there are no machines labeled for it. You may use any machine except the GX620 models, which are not compatible with Fedora Core 3.

1. Insert **Disc 1** into the CD drive and reboot the machine.
2. Press Enter at the **boot:** prompt to continue with a graphical installation.
3. Select **Skip** when prompted to perform the media test.
4. Click next when asked for a language and keyboard mapping.
5. Select **Custom** when asked for the installation type.

Disk Setup

When you get to the disk partition setup menu, you will be given the choice to let the installer automatically partition for you or to manually partition with Disk Druid. For most people, automatic partitioning should be fine. You will need to delete all existing partitions before installing Fedora. After the disk is partitioned, continue on to Installation Options.

1. Select **Automatically partition**.
2. Select **Remove all partitions on this system** and uncheck **Review the partitions created**.
3. Continue on to the installation options section.

Installation Options

After the disk has been setup, you will be presented with a series of menus to set the basic options for Linux. When you get to the networking menu, you will be asked for a hostname. The hostname is the computer's name and domain. You should set the hostname to **Pod<#>Linux.pod<#>.edu.** For example, if you are pod 2 then your hostname should be "Pod2Linux.pod2.edu." You will also be prompted for a password for the root account. The root user is the administrative user in Linux. Set the password to **pod<#>pod<#>** (e.g. pod2pod2):

1. Boot Loader: No change required, click Next.
2. Network Settings: Set the hostname to **pod<#>Linux.pod<#>.edu**.
3. Security Settings: Disable the firewall and SELinux (ignore the warning).
4. Language: No change required, click Next.
5. Time Zone: No change required, click Next.
6. Root Password: Set to **pod<#>pod<#>**.

7. Packages: Scroll to the bottom and select the **everything** option
8. Click through the confirmation, welcome, and license agreement screens.
9. Date and Time: set the time if needed.
10. Display settings: Should be auto-detected but may wish to set the resolution to 1024x786.
11. System user (non-administrative user): username=**pod<#>user**, password=**pod<#>pod<#>**
12. Additional CDs: None needed, click Next.

Linux Basics

After booting into Linux, the login screen appears. You should login with a username of **root** and a password of **pod<#>pod<#>**. Root is the administrative account on UNIX. If you login as any other user, you will not have permission to make changes to the system.

To access the command prompt, you will need to run a terminal program. You can right-click on the desktop and select Terminal or click on **Applications->System Tools->Terminal** to open a terminal. At the prompt you can type the name of the command and press Enter to run it. If the command takes an argument, such as a filename, or an option, such as –a, there should be spaces between the name, options, and arguments. The following are basic Linux commands:

- **cd <directory>**: change your current location to <directory>. (A directory is the same as a folder.)
- **ls**: list the files contained within the current directory.
- **cp <source files> <destination>**: copy files from <source> to <destination>.
- **rm <files>**: delete <files>.
- **gedit <filename>**: edit <filename> with a notepad-like text editor.

Configuring Network Settings

As with the Windows systems, the Linux machine will be assigned an IP address from the 190.111.<pod #>.0 network and will use the Windows 2003 server for DNS and the router as its default gateway. If you are unable to open any applications after changing the network settings, you may have to logout and log back in (as root). Also, some of the computers in the lab have two network interfaces. If you are unable to access the network from Linux, try plugging the network cable into the second network card.

1. Open **Applications->System Settings->Network**.
2. Set the IP address, subnet mask, and default gateway.
    a. Double-click on **eth0** in the list of devices in the **Devices** tab.
    b. Select **Statically Assign IP** addresses.
    c. IP address: **190.111.<pod #>.3**.
    d. Subnet mask: **255.255.255.0**.
    e. Default Gateway: **190.111.<pod #>.254**.
3. Set the DNS Server.
    a. Click the **DNS** tab.
    b. Set the primary server to **190.111.<pod #>.1**.
    c. Verify that the hostname is **Pod<#>Linux.pod<#>.edu**.
4. Click **Activate** to enable the eth0 interface.
   (Note: if you are unable to open a terminal after doing this, click **Actions->Logout** and re-login as root.)

Configuring the Router

Accessing the router

To manage the router, you will first need to connect the router's console port to the serial port on the back of the Windows XP workstation. This can be done using the flat, light blue console cable. Once connected, the router's command prompt can be accessed using Hyper Terminal.

On the Windows XP workstation:

1. Connect the flat, light blue console cable to the serial port on the back of the machine.
2. Connect the other end of the cable to the yellow capped socket on the breakout box.
3. Setup a Hyper Terminal session to connect to the router.
    a. Open **Start->All Programs->Accessories->Communications->Hyper Terminal**
    b. Name the new connection **router**.
    c. Select **COM1** on the **Connect To** menu.
    d. Set **Bits per second** to **9600** on the COM1 Properties menu.
    e. Select **File->Save** in the Hyper Terminal window to save your setup.
4. Press Enter after Hyper Terminal connects. You should see a prompt from the router.

Note: to re-open the router Hyper Terminal session, click **Start->All Programs->Accessories->Communications->Hyper Terminal->Hyper Terminal->router.**

Router command modes

Cisco routers operate in several different modes. Each mode allows only certain commands. The prompt indicates the mode of the router. When you first login to the router, it will be in user mode and the prompt will show *Router>*. To have access to most commands, you must enter privileged mode. To enter configuration commands, you must be in config mode (note that there are several sub-config modes). You cannot use configuration commands in privileged mode or non-configuration commands in config mode.

Use these commands to switch between modes:

- **enable**: Enter privileged mode (enable mode). The prompt will be *Router#*.
- **config term**: Enter global config mode. The prompt will be *Router (config)#*.
- **exit**: exit the last mode entered.

Initial Setup

The first time you connect to the router, you might be asked to enter setup mode, type no. If you are asked to terminate auto-install, type yes. You will need to set the router's name, the privileged mode password (enable secret), and the user mode password (console line password).

1. **enable**
2. **config term**
3. **hostname pod<#>router**
4. **enable secret pod<#>pod<#>**
5. **line con 0**
    a. **password pod<#>pod<#>**
    b. **login**
6. **exit**

Setting IP addresses on the interfaces

The purpose of a router is to forward data between two physically separate networks. Your pod network is connected to one interface (Fa0/0) and the lab backbone network is connected to the other (Fa0/1). When the router receives data packets from a system on your pod's network that are addressed to a system on a different network, it will check its routing table to see if it knows how to reach that other network. If it does, then it will send the data out on the interface that connects to the other network. Because the router talks to systems on two different networks, each interface must have an IP address from the network that is connected to it.

The IP address assigned to the pod network's interface is the address your pod's systems will use as their default gateway. Any time a system wants to send data to a destination that is not on the local network, it will send the data to the default gateway router. The default gateway must then find a path to the destination and forward the data along that route.

From config mode:

1. Assign an IP address to the interface that connects to your pod's network
    a. **interface fa0/0**
    b. **ip address 190.111.<pod #>.254 255.255.255.0**
    c. **no shutdown**
    d. **exit**
2. Assign an IP address to the interface that connects to the lab backbone network
    a. **interface fa0/1**
    b. **ip address 190.111.50.<pod #> 255.255.255.0**
    c. **no shutdown**
    d. **exit**
3. Exit config mode and show the list of interfaces. Verify that fa0/0 and fa0/1 are up.
    a. **exit**
    b. **show ip interface brief**

Running RIP

RIP version 2 is the protocol that exchanges routing information between each of the pod routers and the lab backbone router. RIP works by sending advertisements that list all of the networks that can be reached by going through this router. The other pod routers and the lab backbone router see these advertisements and learn routes to each other's networks. That is, each router will learn that it can send data to any of the other pods by sending the data to that pod's router first. So even though the pods are not directly connected to each other, they are all indirectly connected through the same backbone network and can communicate by routing data over the backbone though the router connected to each other pod.

From config mode (**config term**):

1. Start RIP, set the version to 2, and assign both connected networks to RIP.
    a. **router rip**
    b. **version 2**
    c. **network 190.111.<pod #>.0**
    d. **network 190.111.50.0**
    e. **exit**

2. After a minute or so, check the routing table for any routes learned from RIP (they will have an R in front of them). Note that if no one else has started RIP, there will not be any RIP routes.
    a. **exit**
    b. **show ip route**

Creating a Default Route

Normally a router will only forward data to networks that are in its routing table. That is, networks that are directly connected to one of the router's interfaces, or networks learned from RIP that can be reached by going through a router that is on a directly connected network. When a router connects to the Internet, there are millions of other networks that can be reached. Since the router can't learn so many routes, it uses a default route (a route to **0.0.0.0**) that will match all unknown networks. That is, when the router needs to forward a data packet to a network that it doesn't know how to reach (such as those on the Internet), it will use the default route set by the administrator to send the packet to another router that will know more about where to forward it.

To be able to access the Internet from your pod, you will have to create a default route on your pod's router that will forward packets to the lab backbone router (**190.111.50.55**). The lab backbone router connects the lab network to the campus network and the Internet.

From config mode (**config term**):

1. Create a default route that will send packets to the lab backbone router.
    a. **ip route 0.0.0.0 0.0.0.0 190.111.50.55**
2. Exit config mode and verify that the default route is present in the routing table.
    a. **exit**
    b. **show ip route**

Saving and showing the configuration

After making changes to the running-configuration, you should save it to permanent memory using the command **copy run start**. You can also list the current running-configuration with **show run**.

1. Show the running configuration with **show run**.
2. Save the configuration with **copy run start**

Network Services

Confirming network connectivity

The ping command is the standard tool for testing connectivity between two machines. It works by sending a series of packets (Echo requests) and waiting for responses (Echo replies). If you do not receive a reply to any of the ping requests then check the network settings on the device that is not responding. If the interface is configured correctly then verify that the machine is properly cabled. **If there are two Ethernet interfaces on your machine then try plugging the network cable into the other interface. If you are unable to ping a machine but that machine can ping everything else then verify that its firewall is disabled.**

On each system:

1. Open a command prompt.
    a. In Windows open **Start->Run** and type **cmd**.
    b. In Linux open **Applications->System Tools->Terminal**.
    c. Note: you may use **up arrow** to retrieve the previous command and the left and right arrows to edit the command.
2. Ping the IP address of each machine.
    a. **ping 190.111.<pod #>.1**
    b. **ping 190.111.<pod #>.2**
    c. **ping 190.111.<pod #>.3**
    d. Note that when running ping in Linux you will have to press **Ctrl-C** to stop pinging.
3. Ping both interfaces on your pod's router.
    a. **ping 190.111.<pod #>.254**
    b. **ping 190.111.50.<pod #>**
4. Ping the lab backbone router and server.
    a. **ping 190.111.50.55**
    b. **ping 190.111.50.100**

Active Directory Domain Controller setup

Active Directory domains are used to manage users, computers, and shared resources on a network. The Domain Controller is the Windows server that manages the Active Directory service. Every domain must belong to a Tree, and every tree belongs to a Forest. The tree and forest for your domain will be created automatically when you create your domain. On larger networks, it is possible to have multiple domains in a tree and multiple trees in a forest.

During the installation of Active Directory, you will be asked for the domain name. The domain should be named **pod<#>.edu** (e.g. pod1.edu). Also, the installer will warn that a DNS server is not present and ask if it should configure it automatically. Make sure this option is selected. Active Directory will not function correctly without a DNS server.

On the Windows 2003 server:

1. Insert the Windows 2003 CD and close the installation menu when it appears.
2. Open **Start->Manage Your Server->Add or remove a role**.
3. Select **Custom** Configuration, if asked to select between typical settings and custom.
4. Select **Domain Controller (Active Directory)**.
5. Click **Next** when presented with the Operating System Compatibility information.
6. Select **Domain Controller for new domain**.
7. Select the **Domain in new forest** option.
8. Enter **pod<#>.edu** as the full DNS name of the domain.
9. Accept **POD<#>** as the NetBIOS name.
10. Accept the default locations for the Database and Log folders.
11. Accept the default location for the Shared System Volume.
12. Select **Install and Configure the DNS server** when warned that no DNS server was found.
13. Select **Permissions compatible with Windows 2000 or Windows 2003**.
14. Use **pod<#>pod<#>** for the restore password.
15. After Active Directory and DNS are installed, restart the computer when prompted.

Joining the Active Directory domain

Before it can be managed by Active Directory, the Windows XP workstation must become a member of the domain. When XP attempts to join the domain it will ask for the username and password of a domain administrator. This would be the Administrator account on the Windows 2003 server. After the XP workstation has joined the domain, only users that have been added to Active Directory on the 2003 server will be able to logon to that computer. At this step, the 2003 Administrator is the only Active Directory user. It is important to notice that at the login screen you will have a choice of logging on to the local computer or to your pod's domain. Note that Active Directory only works in Windows so the Linux machine will not be able to join the domain.

On the Windows XP workstation:

1. Join the XP machine to your pod's Active Directory domain
    a. Open **Start->Control Panel->System**.
    b. Select the **Computer Name** tab
    c. Click the **Change** button
    d. Select the option to be a member of a domain and set the domain to **pod<#>.edu**.
    e. When prompted for a domain administrator login use **Administrator** and **pod<#>pod<#>**.
2. Restart the computer when asked.
3. Login as the domain administrator
    a. At the Windows XP login screen, use **Administrator** and **pod<#>pod<#>** for the username and password.
    b. Click **More Options** and select the **POD<#>** domain.
4. Disable the Windows firewall (reactivated by domain controller policy).
    a. Open **Start->Control Panel->Security->Windows Firewall**.
    b. Select **Off**.

Installing AdminPak to manage the Active Directory domain

When a Windows XP machine joins a domain, it becomes possible for the domain administrator to manage the domain and all the services on the domain controller (e.g. DNS, DHCP) from the Windows XP workstation. However, Windows XP does not include most of the management plug-ins under Administrative Tools that are required. AdminPak installs the complete set of management plug-ins. AdminPak can be found on the Windows 2003 CD. Once installed, you will use the tools to add a user to Active Directory for each member of your pod. Make sure you are logged in as Administrator on your pod's domain before installing Adminpak.

On the Windows XP workstation:

1. Verify that you are logged on as a domain Administrator (if unsure, logout and log back in).
    a. Logout by clicking **Start->Log off**.
    b. At the login screen, use **Administrator** and **pod<#>pod<#>** for the username and password.
    c. Click **More Options** and select the **POD<#>** domain.
2. Insert the Windows 2003 CD and close the installation menu when it appears.
3. Open **Start->My Computer** and go to the **I386** folder
4. Find **adminpak.msi**, copy it to the desktop, and double-click on it.
5. Follow the prompts to complete the AdminPak installation.
6. Create a non-administrative user for yourself.
    a. Open **Start->Administrative Tools->Active Directory Users and Computers**.

b. Double-click on **pod<#>.edu** to expand the list.
c. Right-click on the **Users** folder and select **New->User**.
d. Enter your first name in the **First Name** and **User logon name** fields.
e. Use **Pod<#>Pod<#>** for the password (note that the P's must be uppercase).
f. Uncheck the **User must change password at next logon** option.
g. Check the **Password never expires** option.

7. Create user accounts for the other members of your pod.

Configuring DNS service

DNS is the system that looks up hostnames, such as Pod1XP.pod1.edu, and translates them into IP addresses (called forward lookups). Networking devices are only capable of using IP addresses but since it is very difficult for most people to remember an IP address, DNS was created to allow the use of hostnames. Anytime a computer is given a hostname, it must perform a DNS lookup to get the IP address.

Because the DNS server was automatically configured by the Active Directory installer you will not need to add the role manually. You will need to create records to map the name of each computer on your pod to its IP address. You will also need to create a Reverse Zone and records that map IP addresses back to hostnames (PTR records). In addition, your DNS server must forward requests for all unknown domains (like those of websites from the Internet) to the lab DNS server to be looked up.

On the Windows 2003 server:

1. Open **Start->Administrative Tools->DNS** and double-click on **Pod<#>Server**.
2. Create a new Reverse Lookup zone.
   a. Right-click on **Reverse Lookup Zones** and select **New Zone**.
   b. Select Primary Zone.
   c. Use **190.111.<pod #>** as the Network ID.
   d. Accept the zone file name by clicking Next.
   e. Select **Do not allow dynamic updates**.
3. Add a reverse record for your Windows 2003 server to map its IP address back to its hostname.
   a. Right-click on the **190.111.<pod #>.x Subnet** reverse zone created above and select **New Pointer (PTR)**.
   b. Set the last field in the **Host IP number** to **1**.
   c. Set the host name to **Pod<#>Server.pod<#>.edu**.
4. Add a forward and reverse records for the Windows XP workstation.
   a. Double-click on **Forward Lookup Zones**.
   b. Right-click on **pod<#>.edu** and select **New Host (A)**.
   c. Set the hostname to **Pod<#>XP**.
   d. Set the IP address to **190.111.<pod #>.2**.
   e. Select **Create associated pointer (PTR) record**.
5. Add a forward and reverse record the for Linux machine.
   a. Right-click on **pod<#>.edu** and select **New Host (A)**.
   b. Set the hostname to **Pod<#>Linux**.
   c. Set the IP address to **190.111.<pod #>.3**.
   d. Select **Create associated pointer (PTR) record**.
6. Set the DNS server to forward requests for all unknown domains to the network lab server.
   a. Select **Pod<#>Server** in the left panel
   b. Double-click **forwarders** in the right panel.
   c. **All other DNS domains** should be selected in the DNS domains list.

d. Set the **Forwarder's IP address** field to **190.111.50.100** and click **Add**.
e. Leave the forwarders tab open and continue to step 7.

7. For **each pod**, add the pod's domain name to the list and set that pod's server as a forwarder.
   a. Note: **do NOT** include your own pod's domain in the list of domains.
   b. Click the **New** button next to the DNS domains box.
   c. Enter **pod1.edu** in the DNS Domain pop-up dialogue box and click OK.
   d. Select the **forwarder IP_address list** box next to the add button.
   e. Enter **190.111.1.1** (pod1.edu's win2003 server) into the box and click **Add**.
   f. Repeat steps a – e and for **each other pod EXCEPT your own**.
      i. **pod2.edu 190.111.2.1**
      ii. **pod3.edu 190.111.3.1**
      iii. **pod4.edu 190.111.4.1**
      iv. **pod5.edu 190.111.5.1**
      v. **pod6.edu 190.111.6.1**
      vi. **pod7.edu 190.111.7.1**
      vii. **pod8.edu 190.111.8.1**
      viii. **pod9.edu 190.111.9.1**
   g. Click OK when all forwarders are added. If you receive an error, verify that your pod's domain is not in the list of DNS domains. If it is, select your domain and click Remove.

Confirming DNS operation

When the DNS server has been configured properly, you will no longer need to use IP addresses to contact machines on your network. When the hostname of a machine is used, the computer will automatically request the configured DNS server to resolve that name into an IP address. You can test hostname-to-address resolution by pinging the hostname of each machine. To verify reverse DNS resolution (address-to-hostname) you may use nslookup.

1. Ping the host name of each machine to verify forward DNS resolution (name-to-address).
   a. **ping Pod<#>XP.pod<#>.edu**
   b. **ping Pod<#>Server.pod<#>.edu**
   c. **ping Pod<#>Linux.pod<#>.edu**
   d. Note that when running ping in Linux you will have to press **Ctrl-C** to stop pinging.
2. Use nslookup (a DNS record lookup tool) to verify reverse DNS resolution (address-to-name).
   a. **nslookup 190.111.<pod #>.1**
   b. **nslookup 190.111.<pod #>.2**
   c. **nslookup 190.111.<pod #>.3**
3. Attempt to open a web page to verify Internet connectivity.
4. Verify DNS resolution of hostnames from other pods.
   a. **ping pod<other pod #>.edu** (e.g ping pod1.edu, ping pod2.edu)
   b. You should be able to ping the Pod<#>XP, Pod<#>Server, and Pod<#>Linux machines on each pod that has been configured.

Installing the IIS web server

When first installed, the IIS web server has a default website already pre-configured. The default website's home folder is C:\Inetpub\wwwroot. This is where the web server will look for web pages for the website.

When you access a website and don't specify a particular web page, the IIS web server will automatically attempt to display the *index.htm* file, which usually contains the home page for the website (note that all non-Windows web servers use index.html). Although no additional configuration will be required for this task, the web server can be managed from Start->Administrative Tools->Internet Information Services.

On the Windows 2003 server:

1. Insert the Windows 2003 CD and close the installation menu when it appears.
2. Open **Start->Manage Your Server->Add or remove a role** and select **Application Server**.
3. Create a simple web page for your pod.
    a. Open **Start->My Computer->Local Disk (C:)->Inetpub->wwwroot**.
    b. Click **File->New->Text Document**.
    c. Change the name to **index.htm** (ignore the warning about the file extension).
    d. Right-click on **index.htm** and type a message (you may use any HTML tags).

On the Windows XP workstation:

1. Open **Start->Internet Explorer**.
2. Enter **Pod<#>Server.pod<#>.edu.** into the address bar.

Configuring mail service

A typical mail server actually runs two standard services, SMTP and POP3. SMTP is the protocol used to send emails to other computers. When an email is sent, the SMTP server forwards the email to the final recipient's mail server. When a mail server receives an email for a user, it will store that email in the user's mailbox. POP3 is the protocol used to retrieve email from the mailbox on the server.

Once your mail server is setup, you will have to create a POP3 mailbox for each of the user accounts made for the members of your pod. The passwords for their mailboxes will come from their Active Directory accounts. You can then setup Outlook express on both the Windows 2003 and XP machines to exchange emails.

On the Windows 2003 Server:

1. Install the mail server
    a. Insert the Windows 2003 CD and close the installation menu when it appears.
    b. Open **Start->Manage Your Server->Add or remove a role** and select **Mail server**.
    c. Enter **pod<#>.edu** as the **E-mail domain name**.
2. Create a DNS Mail Exchanger (MX) record that points to the mail server for your domain.
    a. Open **Start->Administrative Tools->DNS** and double-click on **Pod<#>Server**.
    b. Double click on **Forward Lookup Zones** then Double click on **Pod<#>.edu**.
    c. Right-click on **Pod<#>.edu** and select **New Mail Exchanger (MX)**.
    d. Enter **Pod<#>Server** into the **Host or child domain** box and click OK.
3. Enable your pod's IP subnet to relay mail to outside addresses.
    a. Open **Start->Administrative Tools->Internet Information Services (IIS) Manager**.
    b. Double-click **Pod<#>Server**.
    c. Right-click **Default SMTP Virtual Server** and select **Properties**.
    d. Select the **Access Tab** and click the **Relay** button near the bottom of the panel.
    e. Click **Add** and select the **Group of computers** option.
    f. Enter **190.111.<pod #>.0** for subnet address and **255.255.255.0** for subnet mask.

4. Create a POP3 mailbox for your user account.
    a. Open **Start->Administrative Tools->POP3**.
    b. Double-click on **Pod<#>Server** in the left panel.
    c. Right-click on **Pod<#>.edu** and select **New->Mailbox**.
    d. Enter the username of the account you created for yourself in the **Name** field.
    e. Uncheck **Create associated user for this mailbox**.
5. Create POP3 mailboxes for the other members of your pod.
    a. Repeat step 4 for each member of your pod.

On both the Windows XP and 2003 machines:

1. Setup Outlook Express (make sure you use different email accounts for XP and 2003).
    a. Click **Start->All Programs->Outlook Express**.
    b. Set **Display Name** to your first name.
    c. Set the **Email** address to **<your account name>@pod<#>.edu**.
    d. Use **Pod<#>Server.pod<#>.edu** for both the incoming and outgoing servers.
    e. Set the account password to **Pod<#>Pod<#>**.
    f. Check **Logon using Secure Password Authentication**.
2. Send an email to your teammate on the other Windows machine.
    a. Click **Create Email** and use **<teammate's name>@pod<#>.edu** for the "To:" address.
3. Check your mailbox on the server for received email.
    a. Click the **Send/Recv** button.
4. Send an email to a member of another pod
    a. Ask the members of a neighboring pod for their email addresses.
    b. Repeat steps 2 and 3 for each email you wish to send.

Enabling Linux FTP and Telnet services

Most of the smaller Internet services in Linux are controlled by a single super server called *xinetd*. Xinetd has a separate configuration file in the **/etc/xinetd.d** directory (folder) for each server program that it handles. To enable the FTP and Telnet services, you must edit the files named **gssftp** and **telnet** and then restart the xinetd server.

On the Linux machine:

1. Open **Applications->System Tools->Terminal** to access the command prompt.
2. Change your current directory to /etc/xinetd.d.
    a. **cd /etc/xinetd.d**
3. Edit the gssftp file, enable FTP service, and remove the special authentication option.
    a. **gedit gssftp**
    b. Change the **disabled = yes** line to **disabled = no**.
    c. Remove the **–a** option from the **server_args** line.
    d. Save and exit.
4. Edit the telnet file and enable Telnet service.
    a. **gedit telnet**
    b. Change the **disabled = yes** line to **disabled = no**.
    c. Save and exit.
5. Restart the xinetd service.
    a. **service xinetd restart**

Once the services are enabled, you should be able to use the command-line telnet client in Windows XP to remotely access the Linux command prompt and the ftp client to transfer files. Both Telnet and FTP require a login. Most services do not allow the root user to login so you will have to login as the non-administrative user created during the installation (**pod<#>user**). Note that the backspace key may not work at the telnet login prompt.

On the Windows XP workstation:

1. Open **Start->Run** and type **cmd** to access the DOS command prompt.
2. Open an FTP session to the Linux machine and upload the C:\Config.sys file.
    a. **ftp Pod<#>Linux**.
    b. User = **pod<#>user**, password = **pod<#>pod<#>**.
    c. **ls** (list the files in the current directory)
    d. **put C:\config.sys** (upload the C:\config.sys file)
    e. **ls** (verify that config.sys was uploaded)
    f. **quit** (close the ftp connection)
3. Open a Telnet session to the Linux machine
    a. **telnet Pod<#>Linux**.
    b. Login = **pod<#>user,** password = **pod<#>pod<#>**.
    c. **ls** (list the files in the current directory)
    d. **exit** (close the telnet connection)

Sharing a folder

A shared folder is simply a regular folder that is being made available to other computers over the network. A share is identified by its *service name*, which is similar to a URL. The service name provides a complete path to the shared folder and has a format similar to *Server-name**Share-name*. Shares can be accessed from Windows using My Computer and from Linux with smbclient. Note that, by default, only administrative users will have permission to access the shared folder. Allowing non-administrative users to access the share is done in a later task.

On the Windows 2003 Server:

1. Create a folder on the desktop named **Shared**
2. Right-click on the folder and select **Properties**.
    a. Click the Sharing tab and select **share this folder**.
3. Open the folder and create a text file named **pod<#>.txt**.
    a. Click **File->New->Text Document**.
    b. Change the name to **pod<#>.txt**.
    c. Double-click on **pod<#>.txt** and type a few words in the file.

On the Windows XP workstation:

1. Open **Start->My Computer**.
2. Type **\\Pod<#>Server\Shared** into the **Address:** bar at the top of the window.

Note that you can also type just **\\Pod<#>Server** in the **Address**: bar to see all of the shared folders offered by the server, or you can browse the whole network by clicking **My Network Places->Entire Network->Microsoft Windows Network**.

Accessing a shared folder from Linux

Linux has several ways of accessing a folder shared from a Windows machine, including a command-line tool called **smbclient**. To access a shared folder with smbclient, you must provide the service name (//server-name/folder-name), a password to access the server, and the name of a user who has access to the share. Once smbclient is connected, it functions very much like an FTP client. It has the commands **get** (download), **put** (upload), and **ls** (list files).

On the Linux machine:

1. Open **Applications->System Tools->Terminal**.
2. Access the shared folder on the Windows server as the Administrator user.
    a. **smbclient //Pod<#>Server/Shared pod<#>pod<#> -U Administrator**
    b. List the files in the remotely shared directory with **ls**.
    c. Download the pod<#>.txt file with **get pod<#>.txt**.
    d. Exit smbclient with **quit**.
    e. Use **ls** to verify that the file was downloaded to the local machine.

Attaching a networked printer

A printer can be attached directly to a network through either a built-in network interface card or an external print server. Before a printer can be used over a network, it must be assigned an IP address, either statically or from a DHCP server. The network lab printer has been statically assigned the address 190.111.50.110. The Windows 2003 server must be configured to send printer data to that IP address. Although the printer is used over the network, Windows terminology describes it as a local printer that is attached to a TCP/IP port.

On Windows 2003 server:

1. Open **Start->Control Panel->Printers and Faxes**.
2. Right-click in an empty part of the window and select **Add Printer**.
3. Select **Local Printer attached to this computer** and uncheck **Automatically detect**.
4. Select **Create a new port** and set the type to **Standard TCP/IP Port**.
5. Enter the IP address **190.111.50.110** in the **Printer Name or IP Address** field.
6. Select **HP** for the manufacturer and **HP Laserjet 2000** for the printer model.
7. Select **Yes** to **using this as the default printer**.
8. Select **Share this printer** and set the Share Name to **Printer**.
9. No location or description information is needed, click Next.
10. Select **Yes** to print a test page.

Using Terminal Services to remotely control another machine

The Terminal server allows a remote computer to take complete control of the Windows 2003 server's desktop. Everything that you would normally see on the server's screen is sent over the network to another computer. The Terminal server is already pre-configured to allow a Remote Desktop connection and requires no further setup. Note that the Remote Desktops administrative tool on the Windows XP workstation was installed by AdminPak.

On the Windows 2003 server:

1. Open **Start->Manage Your Server->Add or remove a role** and select **Terminal server**. (Note that the machine will reboot after the Terminal server is installed.)

On the Windows XP workstation:

1. Open **Start->Control Panel->Administrative Tools->Remote Desktops**.
2. Right-click on **Remote Desktops** in the left panel and select **Add New Connection**.
3. Set the **Server Name** to **Pod<#>Server**.
4. Double-click on **Remote Desktops** to expand the list.
5. Right-click on **Pod<#>Server** and select **Properties**.
6. Select the **Screen Options** tab.
7. Select **Choose desktop size->800x600**.
8. Double-click **Pod<#>Server** to connect to the Windows 2003 desktop and login.

Configuring DHCP on the Windows 2003 server

DHCP is a widely used protocol that lets a computer learn its network configuration (IP address, Subnet mask, Default Gateway, DNS server and domain) from a central server. This is how the majority of computers that connect to a network or the Internet get their IP address. The DHCP server works by using a specified range of IP addresses from the local network as a *scope* or *pool* and then assigning addresses from that range to client computers that send a request for an address.

To make sure that the DHCP server does not try to give out one of the addresses that are already being used (1, 2, 3, and 254), the scope's range should not include every address in the network (1-254). The scope can also be configured to send other network settings, such as the DNS server and default gateway. To test the DHCP server, the Windows XP workstation should be set to obtain its IP address automatically.

Note that when the DHCP server is fully functional, there should be a small green arrow next to the name of the server. If there isn't then you will need to **authorize** the server. To do that, double-click on the server's name in the left panel. There should be a message giving you the option to authorize the server..

On the Windows 2003 server:

1. Open **Start->Manage Your Server->Add or remove a role** and select **DHCP server**.
2. Set the scope name to **Pod<#>Scope**.
3. Use **190.111.<pod #>.10** as the starting IP address and **190.111.<pod #>.50** as the ending address. Set the subnet mask to **255.255.255.0** if it is not already set.
4. No addresses need to be excluded from range used by the scope, click Next.
5. No change is needed to the default address lease time, click Next.
6. Select yes to configure **DHCP options**.
    a. Set the Default Gateway to **190.111.<pod #>.254** and click **Add**.
    b. Set the parent domain to **pod<#>.edu**.
    Set the server IP address to **190.111.<pod #>.1** and click **Add**.
    c. No WINS server is needed, click Next.
7. Activate the Scope.
8. Right-click on the server's name, **Pod<#>Server**, in the left panel and select **Authorize**.

(Note: When the DHCP server is operational there should be a small green arrow next to the server's name. You may need to close and reopen the DHCP configuration tool to see this.)

On the XP workstation:

1. Open **Start->Control Panel->Network Connections->Local Area Connection**.
2. Click **Properties**.
3. Double-click on **Internet Protocol (TCP/IP)** at the bottom of the list.
4. Select **Obtain IP address automatically**.
5. Select **Obtain DNS server address automatically**.
6. Click OK to close the property Window and go to the **Support** tab on the Local Area Connection Status window.
7. Click the **Repair** button to trigger a DHCP request. The status panel should display the new IP address and that it was assigned by DHCP.

Network Security

Analyzing network traffic using tcpdump

Tcpdump is very widely used command-line based application for examining network traffic. It captures every data packet that is sent to or from your computer (including broadcast packets) and outputs a list containing the relevant information about each packet, such as the source and destination addresses and various levels of network protocol information.

By default, the tcpdump command will capture and output every type of packet found. You must use *qualifiers* to limit the capture to specific protocols (TCP, UDP, ARP) and ports (www, telnet, ftp). The qualifier expression is given on the command line when you run the tcpdump command. The table below lists the qualifier expressions for the most common packet types on the lab network.

The exact format of the output for each packet will vary with the selected protocol and port. With the exception of ARP, most of the above protocols have a similar output format. In general, the format will be: (packet type) (source address).(source port) > (destination address).(destination port): (protocol details).

The most complex output is generated by any of the TCP based services. Its protocol data includes TCP flags (S=SYN, P=PSH, F=FIN), the first:last sequence number, the Ack flag, and the acknowledged sequence number. The output for some protocols includes application information. For example, output for a DNS lookup shows the requested DNS record and the server's response.

Commonly used Qualifier Expressions:

Application	Protocol	Ports	Service Name	Qualifier Expression
ping (echo)	ICMP			**icmp**
Web	TCP	80	www	**tcp port 80**
Email	TCP	25	smtp	**tcp port 25**
Telnet	TCP	23	telnet	**tcp port 23**
FTP	TCP	21, 20	ftp, ftp-data	**tcp port 21 and tcp port 20**
DNS	UDP	53	domain	**udp port 53**
DHCP	UDP	67, 68	bootps, bootpc	**udp port 67 and udp port 68**
RIP	UDP	520	router	**udp port 520**
ARP	ARP			**arp**

On the Linux machine:

1. Login as **root**.
2. Open **Applications->System Tools->Terminal** to access the command prompt.
3. Verify that tcpdump is producing output. Find the correct interface number if it is not.
    a. Type **tcpdump –i1** (you should see a continuous stream of output).
    b. Press **Ctrl-C** to exit tcpdump.
    c. If no output was seen, try **tcpdump –i2** or **tcpdump –i3** until you see output. (You may optionally get a list of network interfaces with tcpdump -D.)
4. Run tcpdump with the qualifier (see table above) that captures the type of traffic you want to see.
    a. At the prompt, type **tcpdump –i1 –t qualifier expression**
5. Run a network application that will generate the type of traffic tcpdump is capturing.
    a. For Web traffic, open **Applications->Internet->Firefox** and go to google.com.
    For FTP or Telnet, follow the steps in FTP/Telnet task to connect to your Linux machine.
    For ping traffic, open a terminal and ping one of the machines in your pod.
    For DNS, use the nslookup from a terminal window.
6. Examine the source and destination addresses and ports of the packets listed by tcpdump.
7. Press **Ctrl-C** to exit tcpdump.

Connecting two Active Directory domains using a trust relationship

To connect two domains that are in separate forests, you must establish a trust relationship. A trust will allow the systems in each domain to login to the accounts and access the resources of the other domain. It also allows both domains to be managed from the Administrative Tools of a single system. Once a trust has been established, you should be able to login to an account on the other domain.

It is very important that both Windows servers have as close to the same time as possible. If their clocks are not synchronized, you will get an error when attempting to connect the domains. It is best to synchronize both clocks with an Internet time server, so your Internet connection should be up before attempting this task. If you are unable to use an external time server, it is possible to synchronize the clocks by hand.

On your pod's Windows 2003 server:

1. Synchronize the clocks of the Windows 2003 servers in both pods.
    a. Right-click on the clock in the taskbar and select **Adjust Date/Time**.
    b. Select the **Time Zone** tab and verify that the zone is set to **Eastern**.
    c. Select the **Date and Time** tab and verify that both servers have the same date and time.
2. Create a two-way trust between your pod's domain and the other pod.
    a. Open **Start->Administrative Tools->Active Directory Domains and Trusts**.
    b. Right-click on **pod<your pod #>.edu** in the left panel and select **Properties**.
    c. Select the **Trusts** tab and click **New Trust**.
    d. Enter **pod<other pod #>.edu**.
    e. Select **Two-way trust**.
    f. Click through the confirmations.
3. Repeat these steps on the other pod's Windows 2003 server using your pod's domain and DNS server in the trusts and forwarders setup screens.

On the Windows XP workstation in both pods:

1. Log out by clicking **Start-Log off**.
2. Enter the username of an account in the other pod's domain at the Login screen.
3. Click **More Options** and select the other pod's domain from the list.

Creating a router Access Control List

As a router is forwarding data packets from one network to another, it normally forwards every packet that comes in on every interface. An access control list tells the router to only forward certain packets and not others. An access list consists of a series of entries that either permit or deny a packet. Depending on the type of list, entries can match a packet's source or destination IP address, source or destination port, and protocol. Standard lists (1-99) only match the source IP address. Extended lists (100-199) can match anything.

Access lists also require a wildcard mask for each IP address used. A wildcard mask can be thought of as an inverse subnet mask. If you want an entry to match an entire network that has a mask of 255.255.255.0 then the wildcard mask would be 0.0.0.255. To match a single IP address, you would use a wildcard mask of 0.0.0.0. To have the entry match any possible address, you can use the keyword *any* in place of the address/wildcard-mask combination.

After the list is created, it must be applied to an interface and set to match either inbound or outbound packets. Once in place, the router will compare every packet to each entry in the access list. If no entry matches the packet, there is an *implicit deny* statement at the end of the list. That is, all unmatched packets are automatically denied. Note that packets are only matched if they must go through the router to reach their destination. That is, packets sent between the systems within your pod's network will not go through the router and so will not be filtered by the access list.

You will have to pick a single Linux machine from one of your neighboring pods and create an access entry to block all packets from this machine. The list will also block telnet access from everywhere. Once you have selected the pod and created the list, you can test the list from that pod. If the list is successful, the Linux machine will not be able to ping anything in you pod and all attempts to telnet to your pod should fail. Everything else, should work as normal.

On **your pod's router** in config mode (**config term**):

1. Create an access list entry that blocks telnet access from "any" host outside of your pod.
    a. **access-list 100 deny tcp any 190.111.<your pod #>.0 0.0.0.255 eq telnet**
2. Create an entry to block any ip traffic from **the Linux machine in a neighboring pod** to your pod.
    a. **access-list 100 deny ip 190.111.<other pod #>.3 0.0.0.0 190.111.<your pod #>.0 0.0.0.255**
3. Create an entry to match all other packets and allow them.
    a. **access-list 100 permit ip any any**
4. Apply the list to the fa0/0 interface to match packets going out to the pod's network.
    a. **interface fa0/0**
    b. **ip access-group 100 out**
    c. **exit**

On the **neighboring pod's** XP machine:

1. Verify that you can ping your pod from the Windows XP workstation but cannot telnet.
    a. Open **Start->Run** and type **cmd**.

b. **ping 190.111.<your pod #>.3**
c. **telnet 190.111.<your pod #>.3**

On the **neighboring pod's** Linux machine:

1. Verify that you cannot ping systems in your pod from the Linux machine.
   a. Run **Applications->System Tools->Terminal**.
   b. **ping 190.111.<your pod #>.1**
   c. **ping 190.111.<your pod #>.3**

Restricting permissions of a shared folder and auditing

Security permissions can be set on a folder (or any other object) to control what tasks a user can and cannot perform on that folder. Shared folders have two types of permission settings, *Share Permissions* and *Access Control Lists*. The Share Permissions are set by clicking on the Permissions button in the Sharing tab of the folder's properties. Access lists are managed in the Security tab. It is important to realize that both sets of permissions are used by the server to determine what each user will be permitted to do. The server will always pick the most restrictive interpretation of the settings. For example, if the share permissions grant write access to everyone but the access list denies write access for some users, then the more restrictive deny entry will take precedence over the less restrictive share permissions.

By default, the shared folder created earlier has share permissions that allow all users to open and read files from the folder but write to or delete. It also has a more restrictive access control list that only permits administrators to access the folder. So by default, the shared folder cannot even be opened by non-administrative users. To give all users the ability to read, write and delete files, the share permissions can be set to give *Full Control* permissions to *Everyone*. Also, a new Access Control List entry must be added for the Everyone group that also grants Full Control permission. Once this is done, all non-administrative users should have permission to do anything in the shared folder.

To deny write and delete permission for one specific user, you have to create an access list entry that specifically denies write and delete permission to that user. Delete permission is not included in the basic access list menu so an advanced access list entry must be used. Note that creating an access list for the user that doesn't deny permission but doesn't allow it either is not the same as actually denying permission.

Auditing is a system that allows administrators to track and log when a user makes an attempt (successful or failed) to access system resources or perform certain tasks. In this case, it will be used to monitor each time a specific user is denied permission to write or delete files in the shared folder. Auditing on a folder is configured in the Security tab and works almost exactly like file permissions. Before you can create auditing rules on a folder, you must enable Object Access auditing in the Domain Controller Security Policy (any auditing rules created before enabling the policy will not work). Once auditing has been established, any failed attempts to write or delete files in the shared folder will be logged to the Security log of the Event Viewer. To test auditing on a shared folder, you must login as the restricted user on the Windows XP workstation and attempt to delete or write to the file. The permissions should cause the attempt to fail and the auditing rule should log the failed attempt.

On the Windows 2003 server

1. Right-click on the folder named **Shared** and select **Properties**.
2. Set the share permissions to grant Full Control to all users
   a. Select the **Sharing** tab and click on the **Permissions** button.

   b. Select the **Everyone** entry in the user and groups list.
   c. Select the **Allow** option for **Full Control** permission in the permissions panel.
3. Create an Access Control List entry that grants Full Control to all users.
   a. Select the **Security** tab.
   b. Click the **Add** button below users and groups and type **Everyone**.
   c. Select **Everyone** and check the **Allow** option for **Full Control** permission in the list.
4. Deny write permission to one of the accounts created for the members of your pod.
   a. Click the **Add** button and type in the username of a non-administrative user.
   b. Select the added user and uncheck all **Allow** permissions in the permissions list.
   c. Check the **Deny** option for **Write** permission.
   d. Click the **Advanced** button.
   e. Select the **Deny** entry associated with the added user and click **Edit**.
   f. In the permissions list, check the **Deny** options for **Delete** and **Delete Files and Subfolders**. (Note that you should see several Deny options already checked for write permissions.)
   g. Click OK to return to the Advanced Settings window.
5. Enable auditing for object (file and folder) access in the domain controller policies.
   a. Open **Start->Administrative Tools->Domain Controller Security Policy**.
   b. Double-click **Local Policies->Audit Policy->Audit object access**.
   c. Select **Define this policy** and **Failed**.
6. Create an auditing entry to track failed access attempts made by the user.
   a. Select the **Auditing** tab in the Advanced Settings window.
   b. Click **Add** and type the username of the selected account.
   c. In the object access panel, select the **Failed** option for the six write or delete access methods starting at **Create Files/Write Data** and ending at **Delete**.
   d. Click Apply and OK on all of the windows associated with the folder's properties.
7. Open the Event Viewer Security log, clear all exiting log entries, and watch for a Failed entry.
   a. Open **Start->Administrative Tools->Event Viewer**.
   b. Select **Security** in the left panel.
   c. Right-click on **Security** and select **Clear all events**. Click No when asked to save to a file.
   d. When an access is attempted, right-click on the right panel and select refresh to see it.

On the Windows XP workstation:

1. Click **Start->Log off** to logout then login as the user who has the restricted permissions.
2. Open **Start->My Computer** and type **\\Pod<#>Server\Shared** into the address bar.
3. Right-click on the **pod<#>.txt** file and attempt to delete it (you should see permission denied).
4. Check the Event Viewer Security log on the Windows 2003 server for a failed access event.