

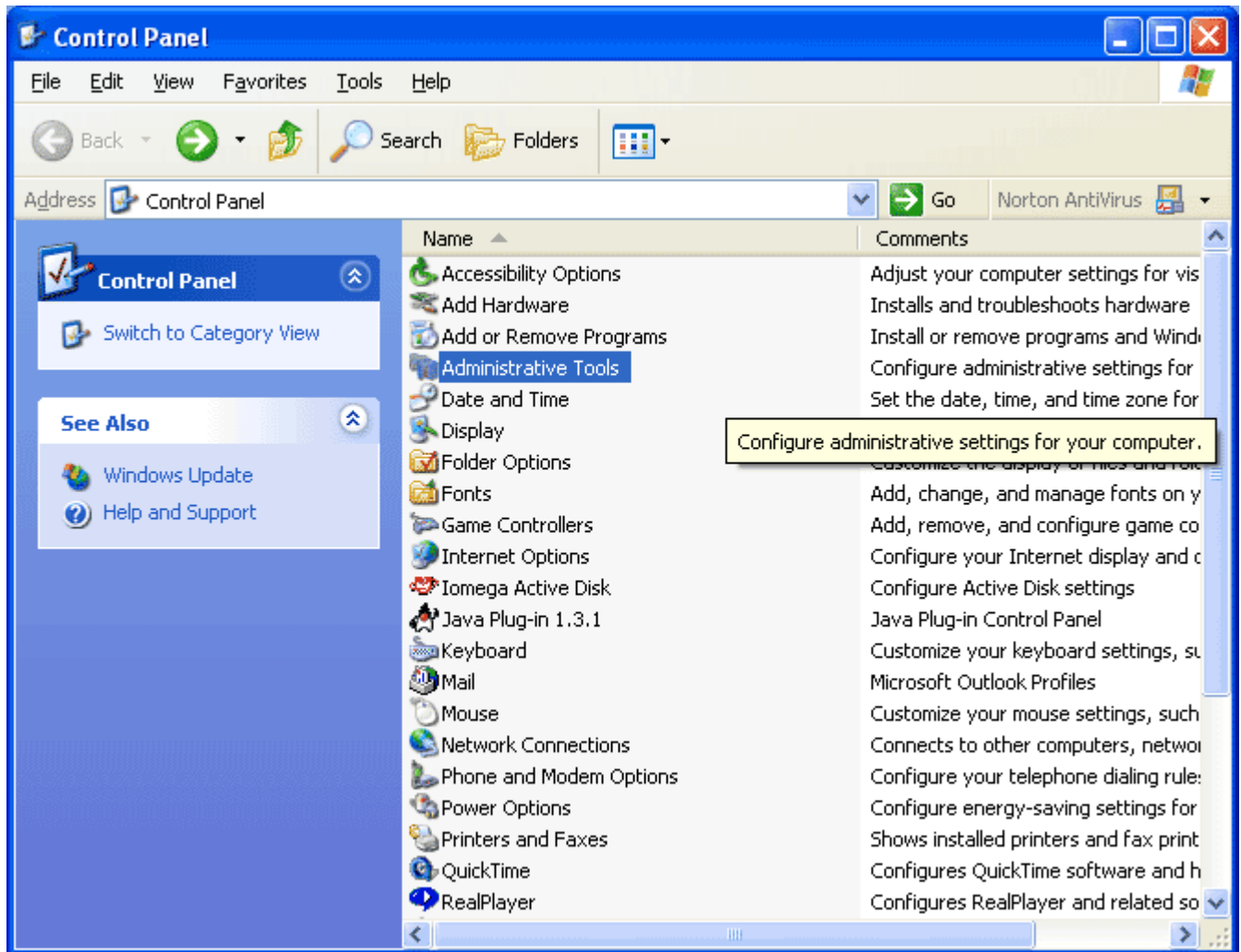
Installing IIS on Windows XP

1. Start
2. Go to Control Panel
3. Go to Add or Remove Programs
4. Go to Add/Remove Windows Components
5. At the Windows Component panel, select the Internet Information Services (IIS) checkbox, and then hit Next.
6. You will probably be prompted for the original Windows XP Professional CD. Insert the CD and continue.
7. Once the installation has completed, continue on to the next section.

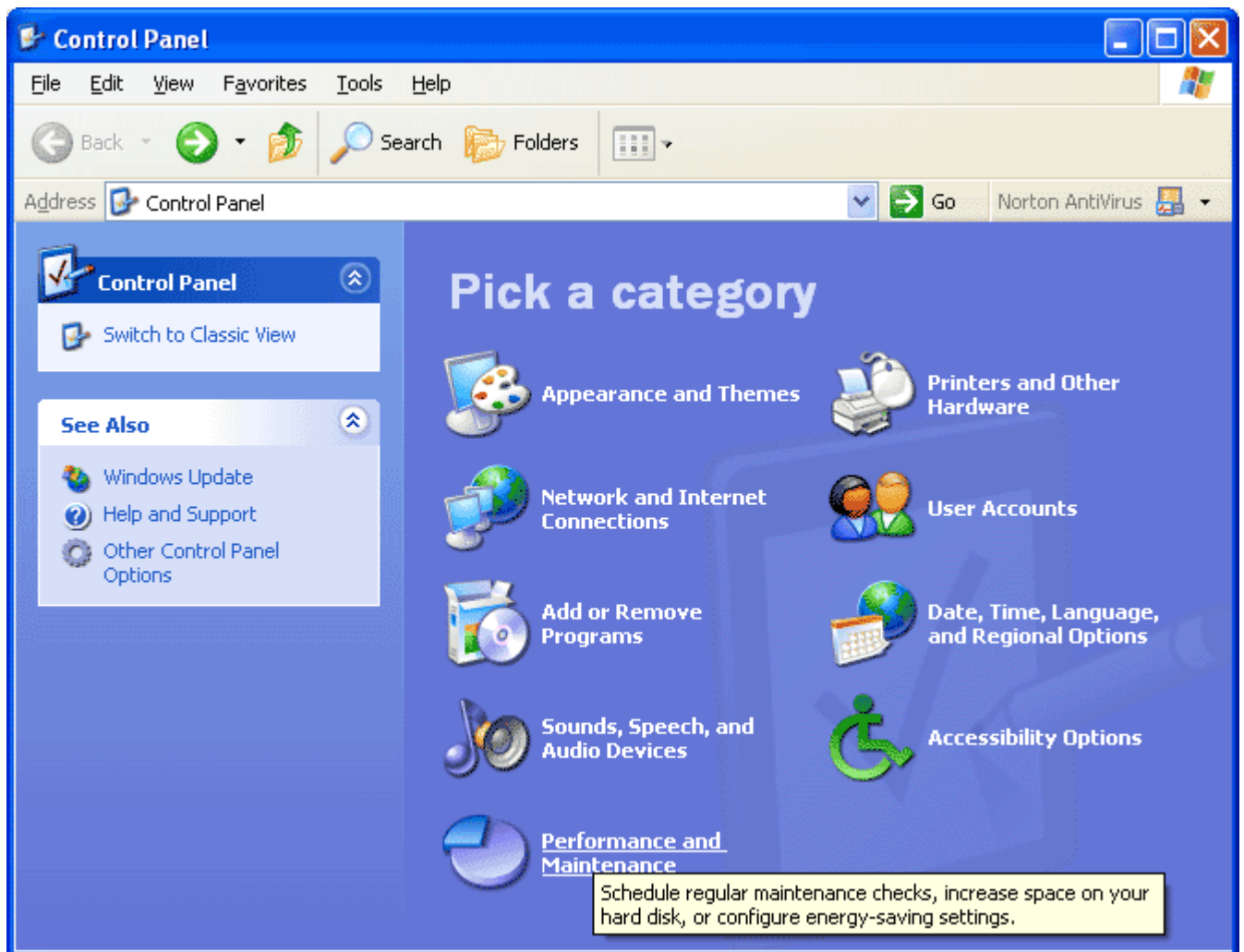
Configuring IIS for a Default Web Site

Your first web site-Control Panel

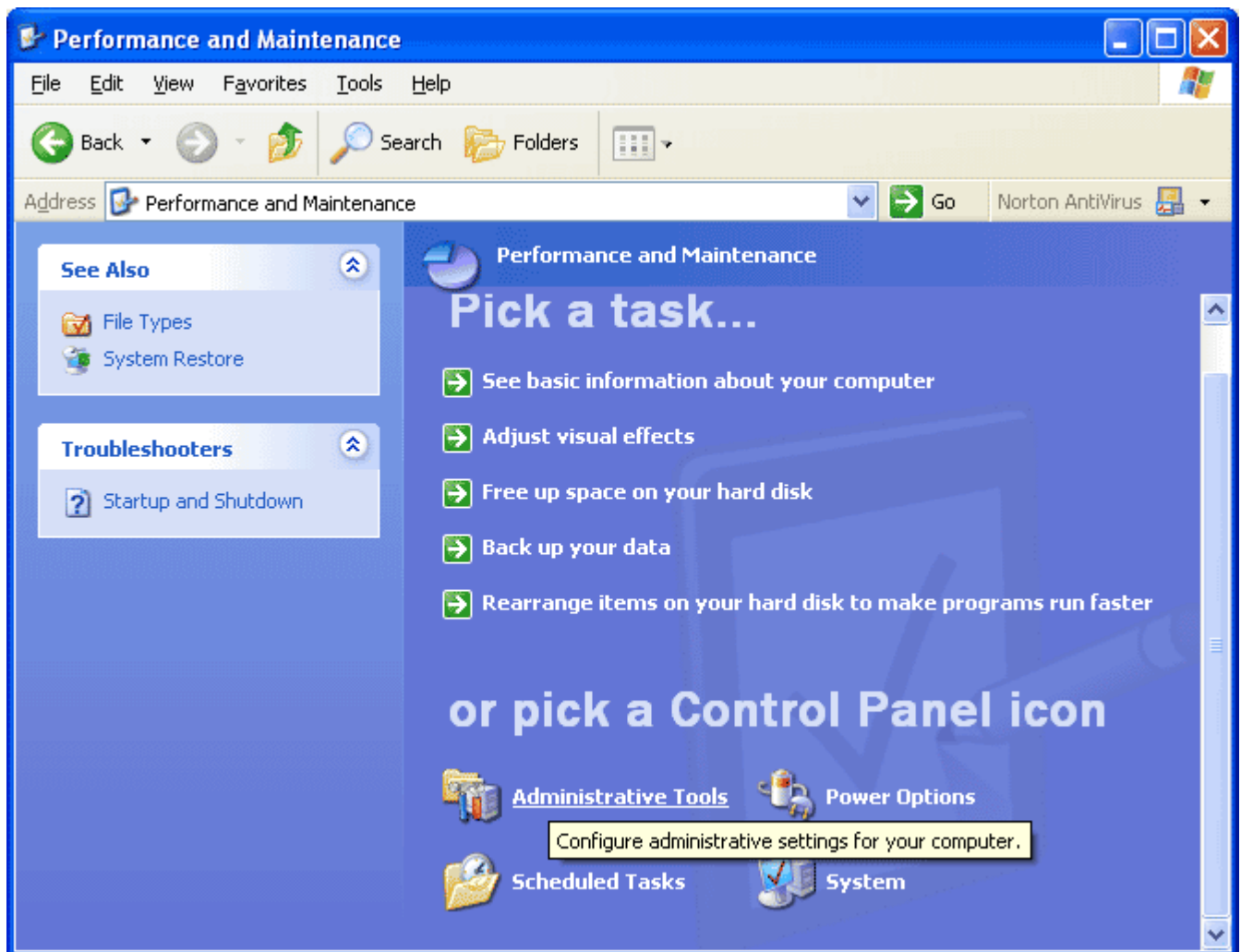
Make certain that you have logged on to the computer as an administrator. Depending on your version of Windows, the initial panels will appear a bit differently. First, you need to access the control panel. Do this via **Start, Settings, Control Panel**. Depending on your operating system, you will see one of two choices. If you are using Windows 2000 or the **Classic view** in Windows XP Professional, you will see the control panel as below. You are interested in the **Administrative Tools** link.



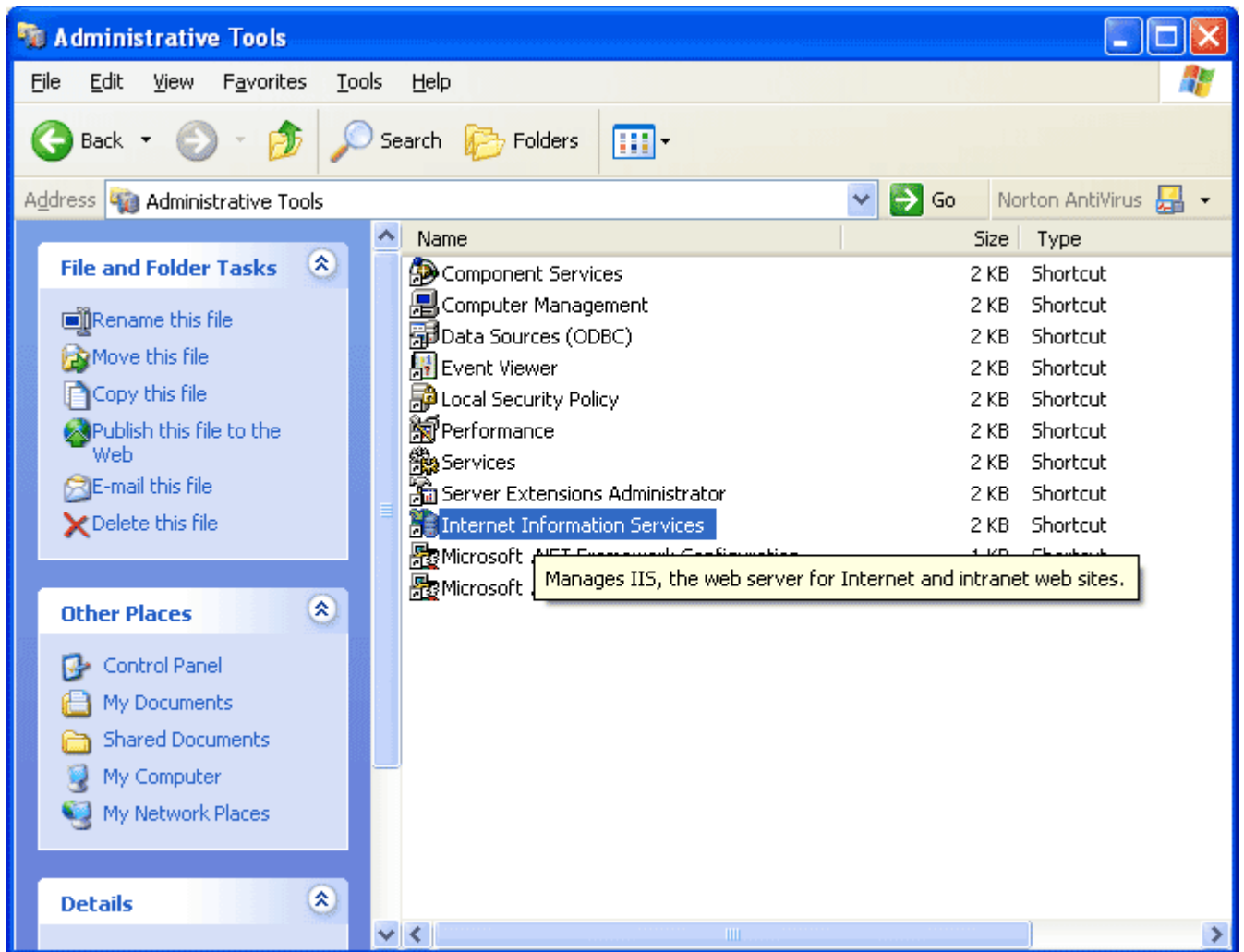
If you are using Windows XP Professional, you may be in **Category View** when you access the control panel. In this case, you will need to select **Performance and Maintenance**. That is shown below.



Within the **Performance and Maintenance** option, you then need to select **Administrative Tools** (see below).

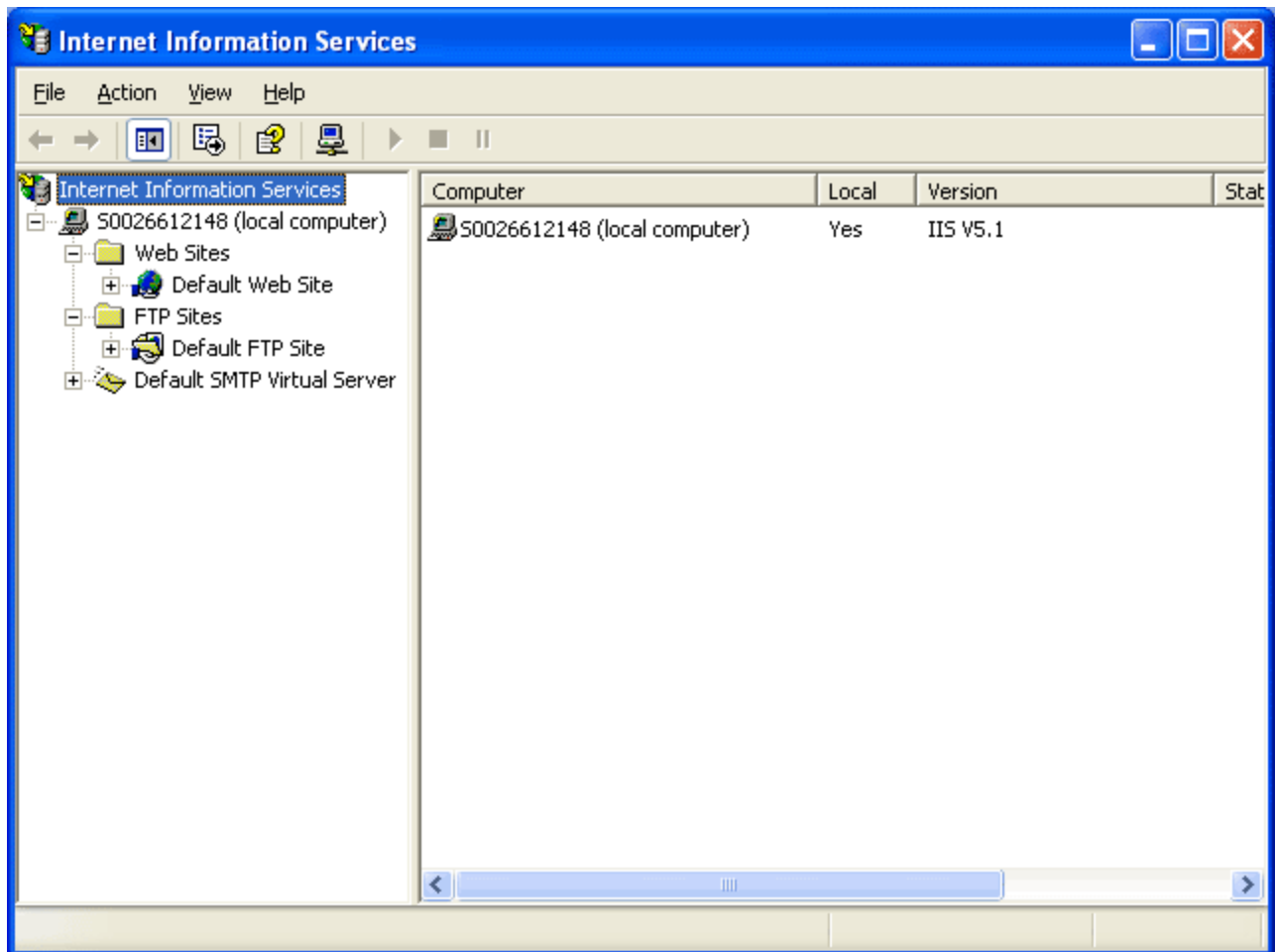


In either scenario, you next need to select **Internet Information Services** (or **Internet Services Manager**) from the list of choices. In this case, the view is now the same for both Win XP and Win 2K.

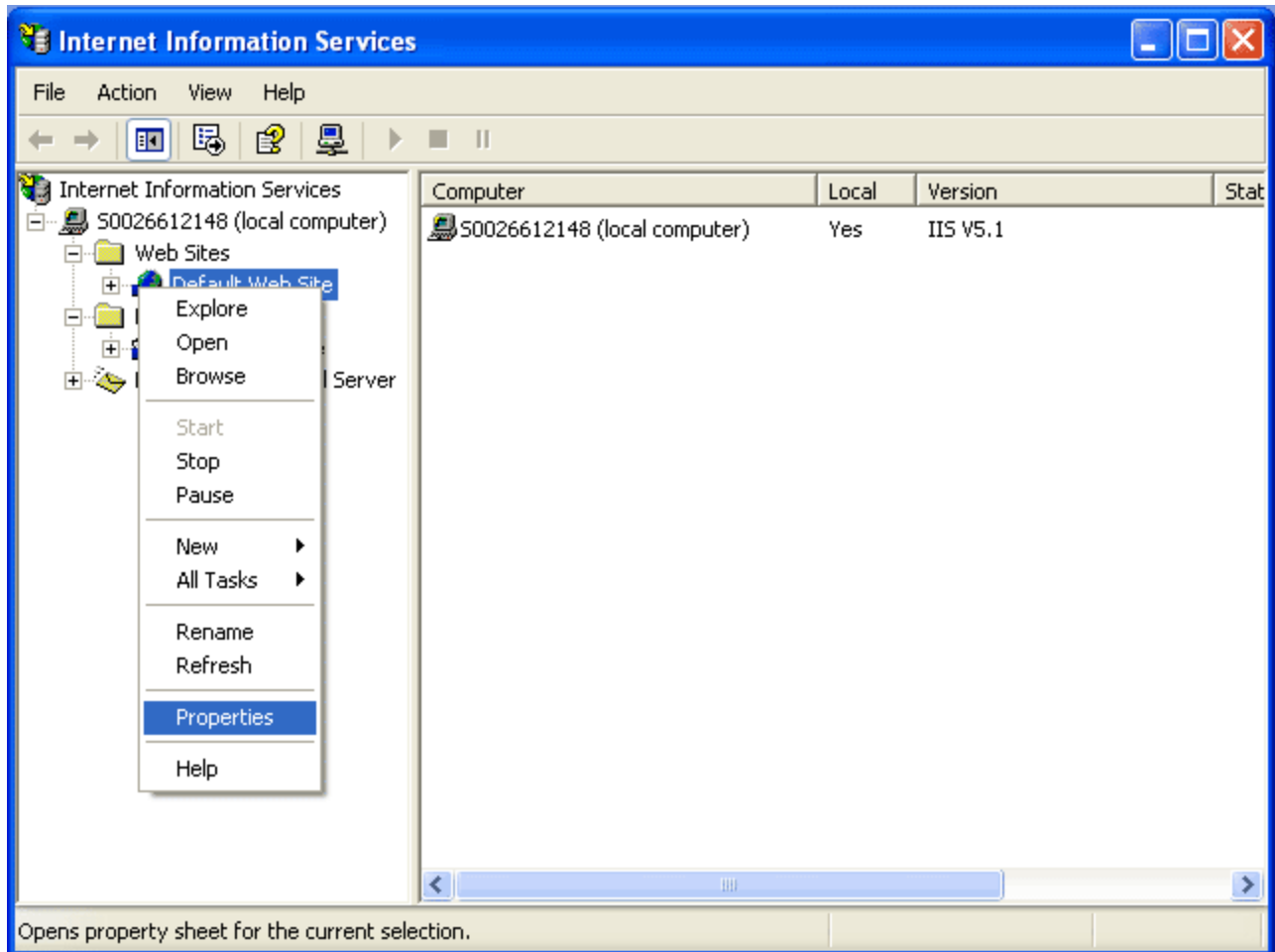


Your first web site - IIS Panel

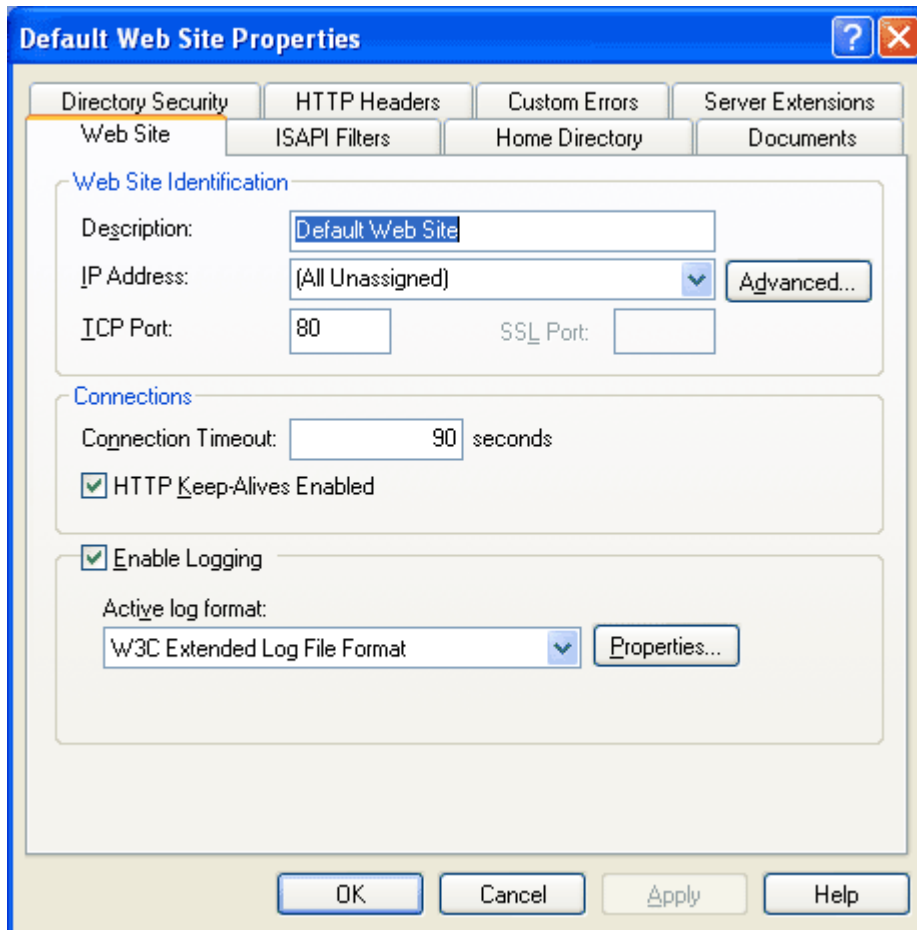
You should now be presented with the IIS management graphical interface. Yes, there is a command line interface, but I prefer the graphical approach. I have partly expanded the view (you can do this by clicking on the (+) icon as you would in Windows Explorer). Note that we are now interested in the **default web site**.



When working with this interface, recall that **right clicking** on items opens a context sensitive menu. In this case, position your cursor over the default web site and right click. Select **Properties** from the list. Yes, you could also use the menu (there is a properties icon - can you find it). Note the image below.



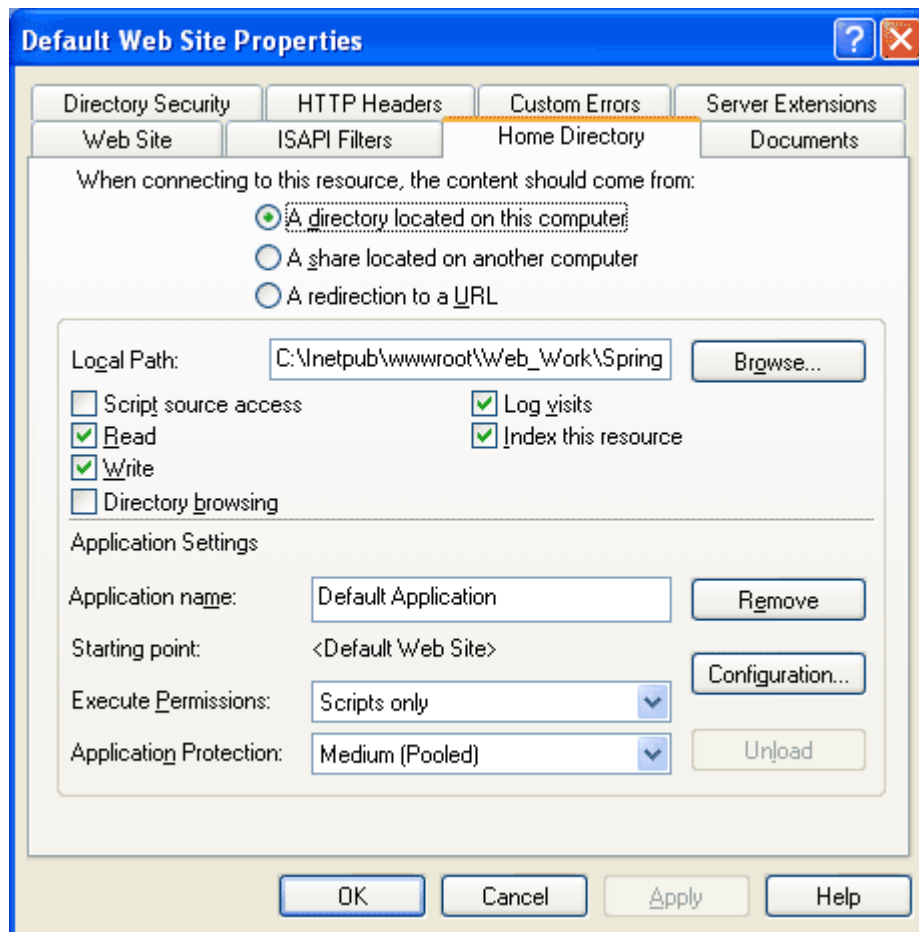
You will next be presented with the **properties panel** (see below). Make certain the **Web site** tab is selected first (it should be the default).



Place a description for the web site in the **Description** text box. Yes, it is a good idea to name the web sites (unless you really like the default setting). You may wish to include the IP address for the web site or take the default (**All (Unassigned)**) for now. Since you are just doing this for the lab, it is unlikely that you will have a specific IP address or domain to associate with this site. You may also wish to modify the **TCP Port** as needed. Recall the default port for web sites is 80. Some environments (perhaps Tomcat or JSP) might require 8080 or similar. If this is the case, and you do not specify this properly, individuals will need to access the site by typing `http://server_location:port` instead of just specifying the IP address or server name.

You will note the **Connection timeout** is set to 90 seconds. This is the default. If your site will be running active server pages or another server technology, you may wish to give consideration to this figure. If someone develops a web page that runs a program which is stuck in an infinite loop, this is the only way out. The server will kill any active process after it has been running for the time specified. Ok, let's set it really low. **Don't**, in this case, if you are accessing a database (and making a SQL call), it is quite possible that the database interaction may require seconds to complete. Of course, if the timeout is exceeded, the server will kill the request in mid-transfer. I can't think of a better way to drive a developer nuts. Their code will work, not have any flaws, but will still fail (because of the low timeout setting). The default value was selected after much consideration. However, you may need to change this (after you have been working with the site for a while or have been really irritated by a developer!). Just kidding on the last point...

We next proceed to the **Home Directory** tab. The image below represents this choice.

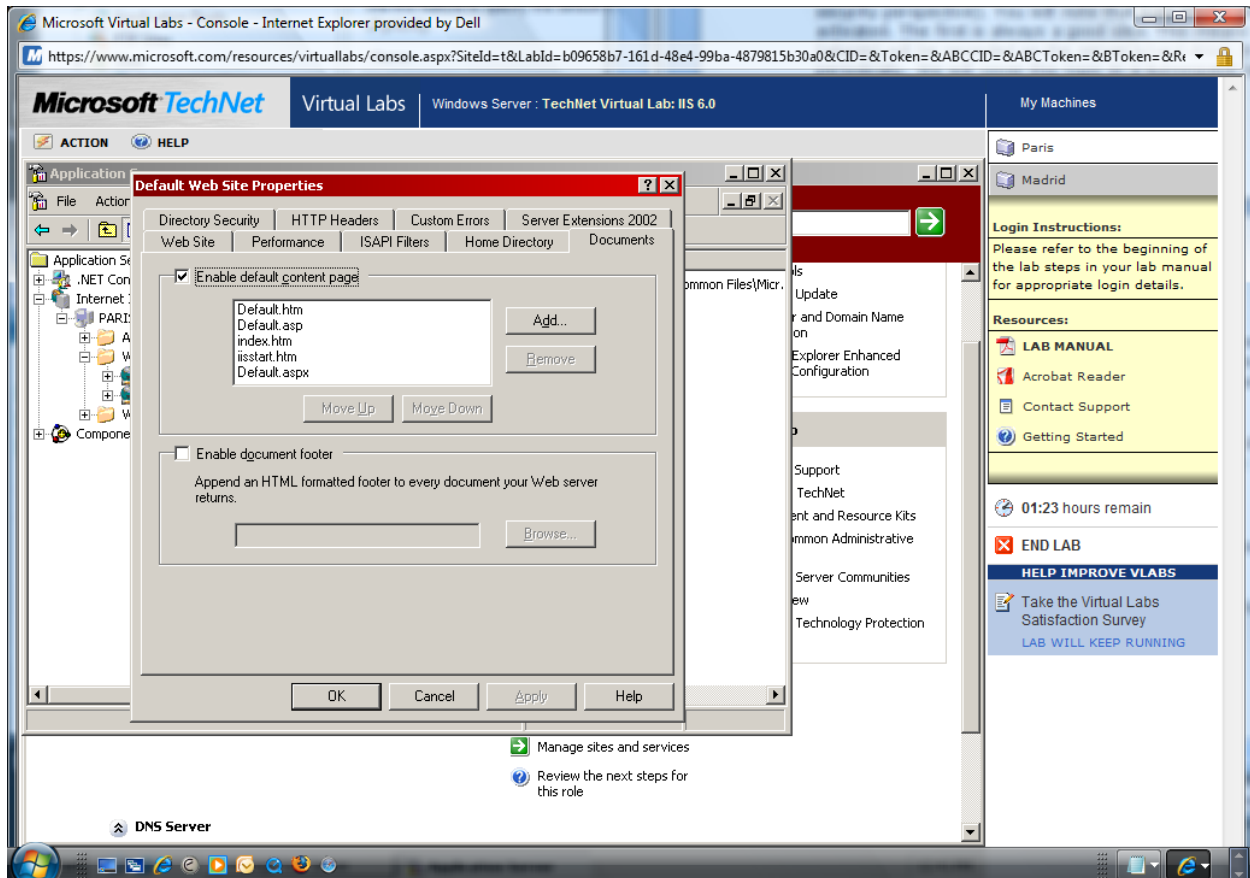


You have several additional options to consider on this tab. First, you must indicate where the server can find files for the web site. Typically, they will be located on the same computer. Therefore, the default **A Directory on this computer** will work just fine. You may need to **Browse** to locate the specific folder (for the lab, create the folder C:\CMWEB290\lastnameinitial and point it there). If your server is part of a server farm, you may select **A share located on another computer**. If your web site needs to point to another site (perhaps you have multiple names and all point to a specific web site), you would select the choice **A redirection to a URL**. For lab purposes, we will only concern ourselves with files located on this computer.

Read access is required for the web site to be available to the public. In the example shown, I have also selected **Write** access. Be very careful about allowing this. It only makes sense if you have an interactive site using a lot of server technology (like ASP). **Script Source Access** is a very dangerous choice. This can expose your site to hacking attempts. For general purpose web sites, do not activate this. **Never, never, never** (did I say **NEVER**) allow **Directory Browsing**. This means that anyone who visits a folder can see all the contents (and click away on them as desired). This is a very dangerous option (from a security perspective). You will note that **Log Visits** and **Index this Resource** are activated. The first is always a good idea. This means that access attempts (successes and failures) will be recorded on the server's hard drive. It is a good idea to examine these periodically. We will cover this topic in a subsequent lab. If you wish to activate search

capabilities (perhaps through .NET or an ASP Component), then you would want to index this resource. **Application Settings** is pertinent if you are working with ASP (and is beyond this course and current topic). When you have finished, click **Apply**, then **OK** to save the changes.

You will also need to add "default.html" to the list of default pages. To do this, click on the Documents Tab to get to a screen that looks like this:



Click on the add button, and put default.html in the box, then click OK.

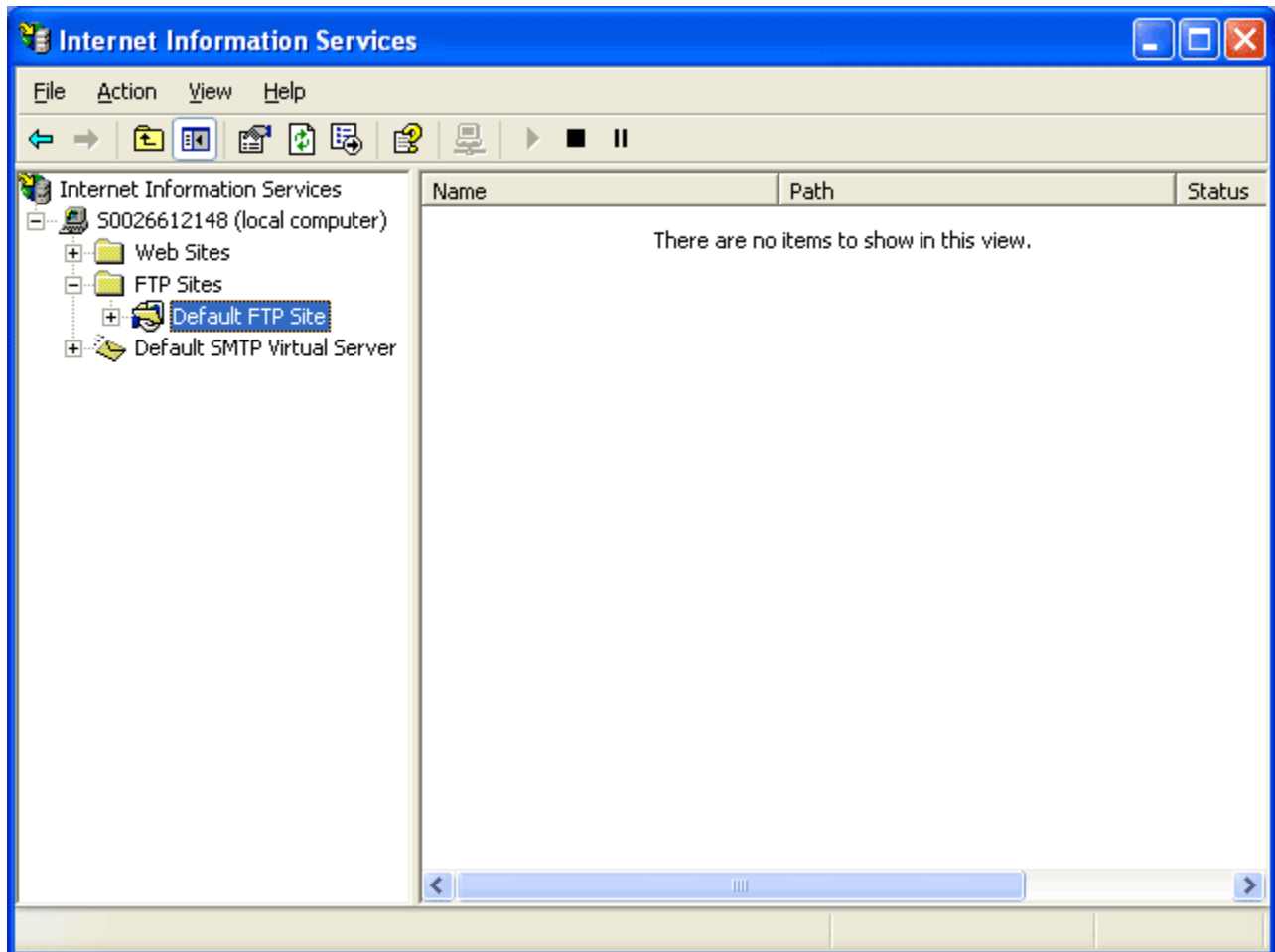
You will need to create an html page called default.html and put it in the folder you defined earlier (c:\CMWEB290\lastnamefirstinitial)

Once you have your default.html page saved in that folder, open up internet explorer and type the following address in the address box:

<http://localhost/>

Your first FTP access to the Web Site

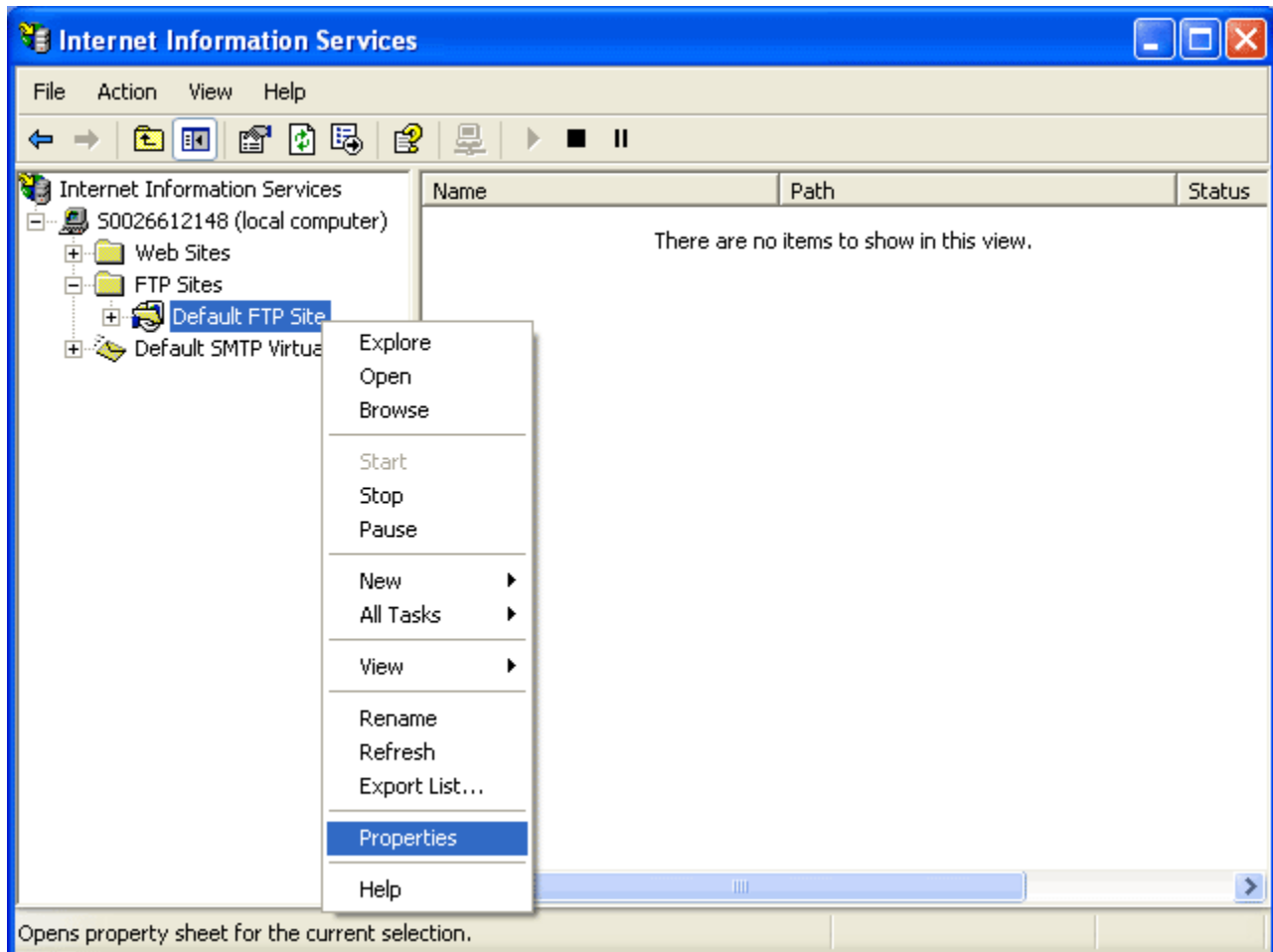
Let's return to the IIS administration interface. We will now focus on the FTP site (instead of the web site which we have just configured). FTP stands for **File Transfer Protocol**. Recall that FTP is the mechanism we use to transfer files and directories to/ from the server. The FTP site acts like a computer (and these sites are sometimes called FTP servers or virtual FTP servers). You use FTP commands to work with files and directories on the remote server (typically located in a secure facility and unavailable for direct access). Examine the image below.



When IIS is installed, you automatically get a default FTP site. This site allows **anonymous** FTP access. This means that you can access the site using the username of **anonymous** and a password of **your email address**. Hey, email@server.com also works just fine as a password. While web sites use a default TCP/IP port of **80**, FTP sites use a default port of **21**. Thus, one could point their browser to the site and use ftp:// instead of http:// to directly access the files/ folders (or at least view them).

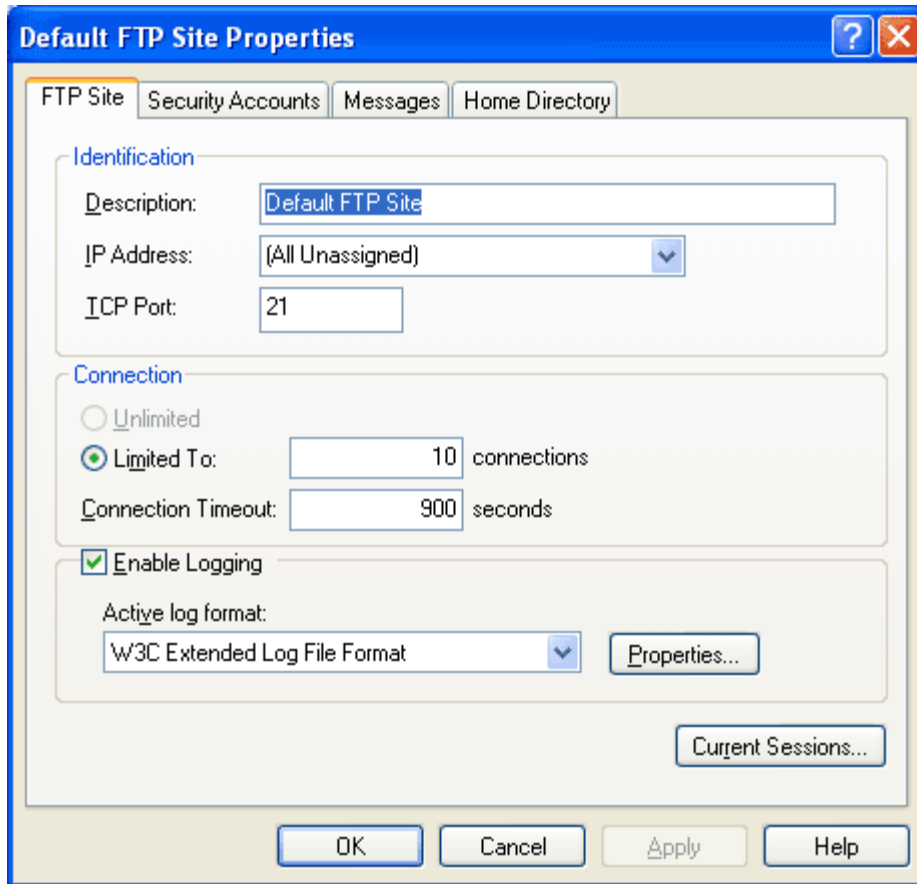
If you are curious, Windows authentication uses the **IUSR_computername** account to provide this anonymous access. Note that anonymous authentication takes precedence even

when you enable Basic Authentication. Ok, enough of the lecture, let's get on with making changes. As with the web site, right click on the FTP site and select properties. Note the image below.

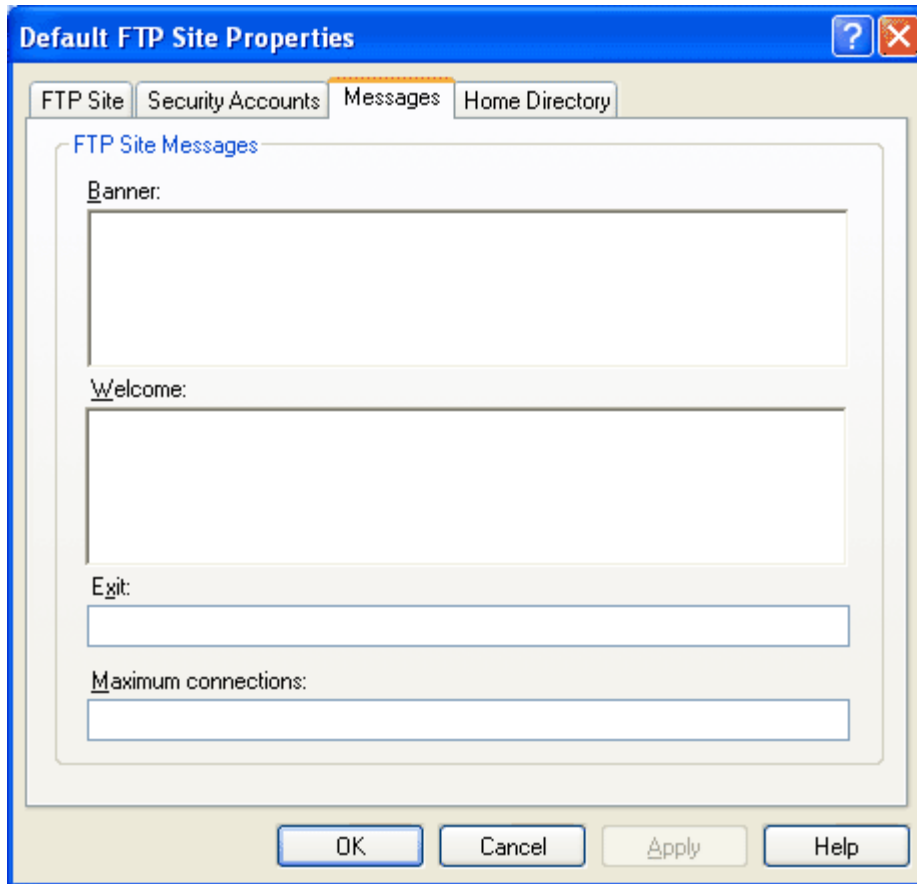


We will now focus on the FTP site properties.

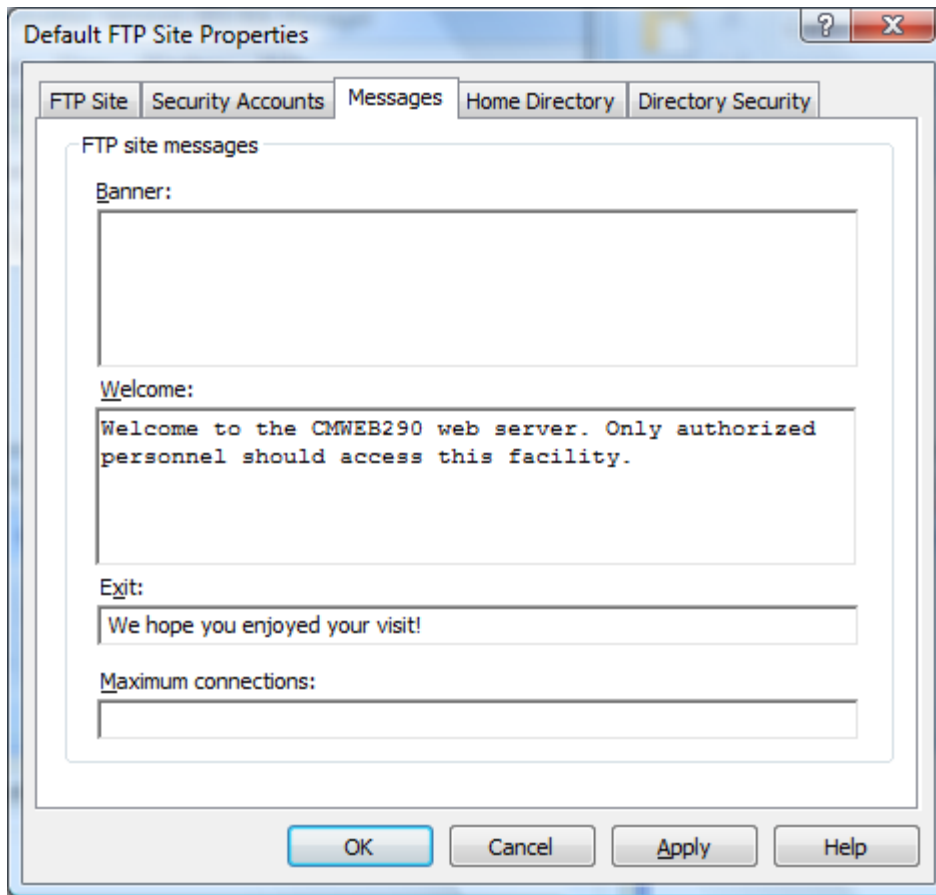
FTP Properties



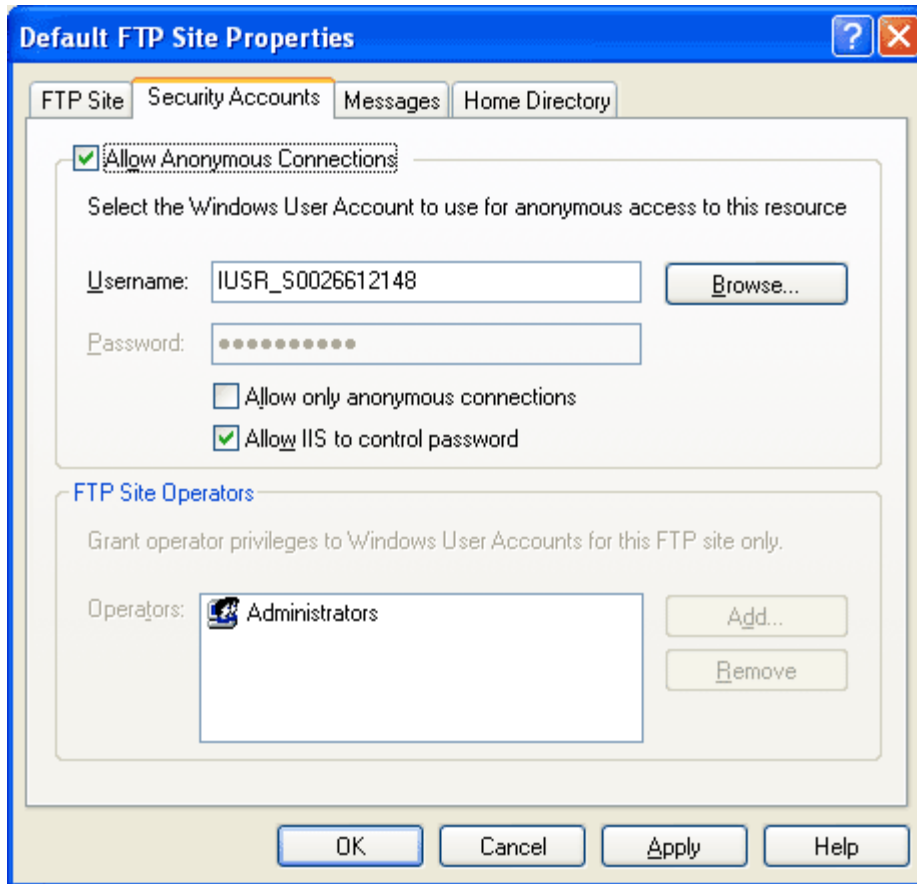
Note the first tab (**FTP Site**) is where we direct our initial attention. As with the web site, you probably want to change the name of the default FTP site to something more meaningful. Leave the IP address as **All unassigned** for this lab. In an actual business setting, you may want to configure that to meet custom requirements. You can also control the number of concurrent connections (10 is a good start) and the timeout (900 seconds is a good start). Depending on the power of your server, the number of potential users, etc., you may wish to change the number of concurrent users. For now, we will accept the defaults. Click the **Messages** tab.



I recommend setting some informational messages to welcome visitors to the FTP site and update them when they leave. Note that I placed a rather generic message in these boxes on the image below. You may wish to do likewise. You can also specify the maximum number of connections (if desired). Compare setting this value and the value on the FTP site and observe the differences. Obviously, try out low numbers. If you are using a tool like Cute FTP, FTP Explorer (or the paid version of WS-FTP), you can establish numerous concurrent connections to the server (helps when transferring a large number of files at once). This feature can act like a throttle to keep the server from being overwhelmed by concurrent file transfers.

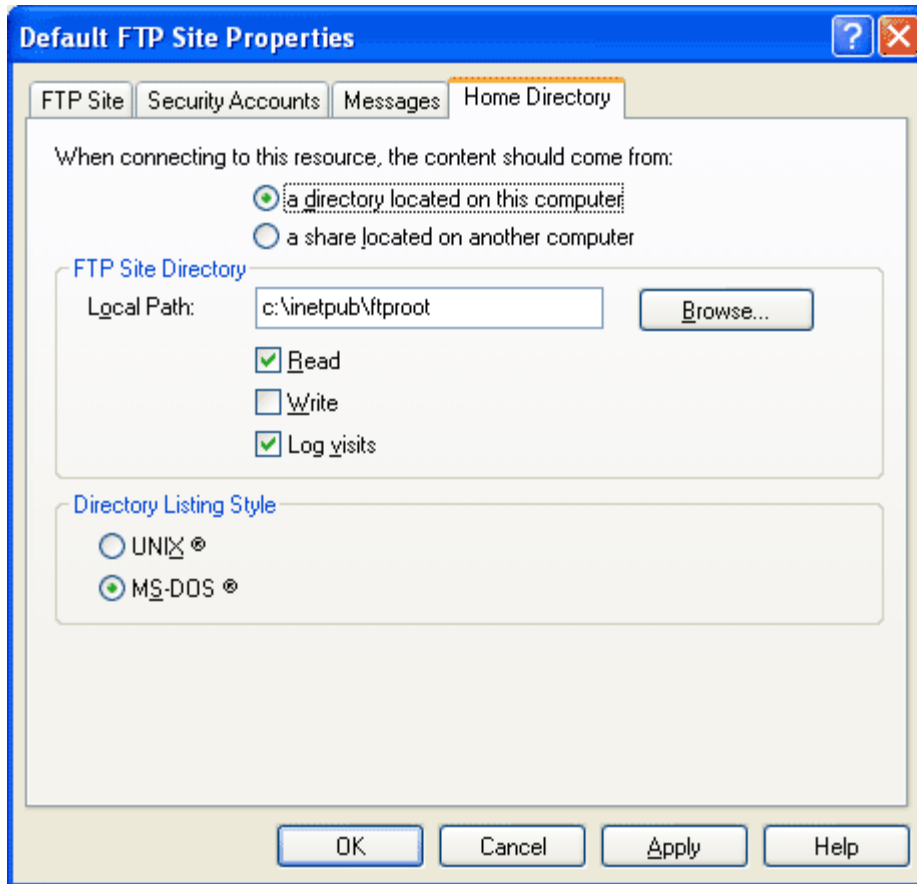


We next focus on the security settings. Click on the **Security Accounts** tab and you will see a screen similar to the one shown below.



The first thing you may notice is that **Anonymous** access is allowed. If you do not wish this, deselect the check box, but make certain that all users who need access have an account set up (with username and password) on the computer. Otherwise, no one will be able to access the facility via FTP. You will note that the bottom part of the screen is grayed out. Deselecting Anonymous access will allow you to control this part of the screen and specify who can access this resource.

We next focus on the **Home Directory** settings. Click on the **Home Directory** tab and you will see a screen similar to the one shown below.



You may notice similarities with the previous web site configuration tab. In addition to specifying the location of the content (and browsing if needed to locate it), you may see that **Write** access is not enabled by default. This makes it rather hard for individuals to upload files to the server via FTP (although it is good security). I also recommend logging all access attempts (and reviewing them periodically). Not doing so is the equivalent of placing a video camera in a bank but never placing a tape in the recorder. Doesn't do much good after a crime has been committed... You can also control the listing style. Unless you are fond of NIX, I recommend leaving this default to MS-DOS. Hey, we are talking about a Windows environment after all.

To test it, open up My Computer, and put the following in the address bar:

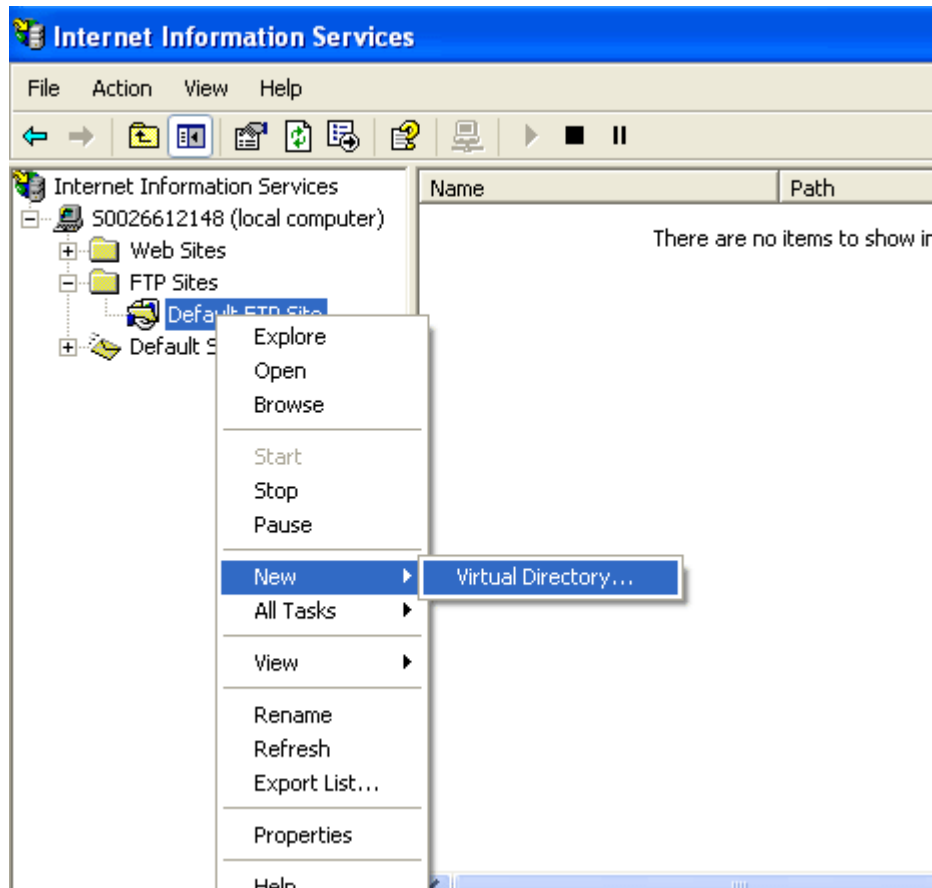
<ftp://theusernameyouuseforyourcomputer:thepasswordyouuseforyourcomputer@localhost>

it should connect to the ftp, and the folder should contain the default.html file you put in that directory previously. See if you can drag another file into the folder; if you can, it is set up properly.

Ok, we now have a single FTP site for everyone to access. What if we want to make multiple sites for different users (perhaps they have their own web environment).

New FTP Directories

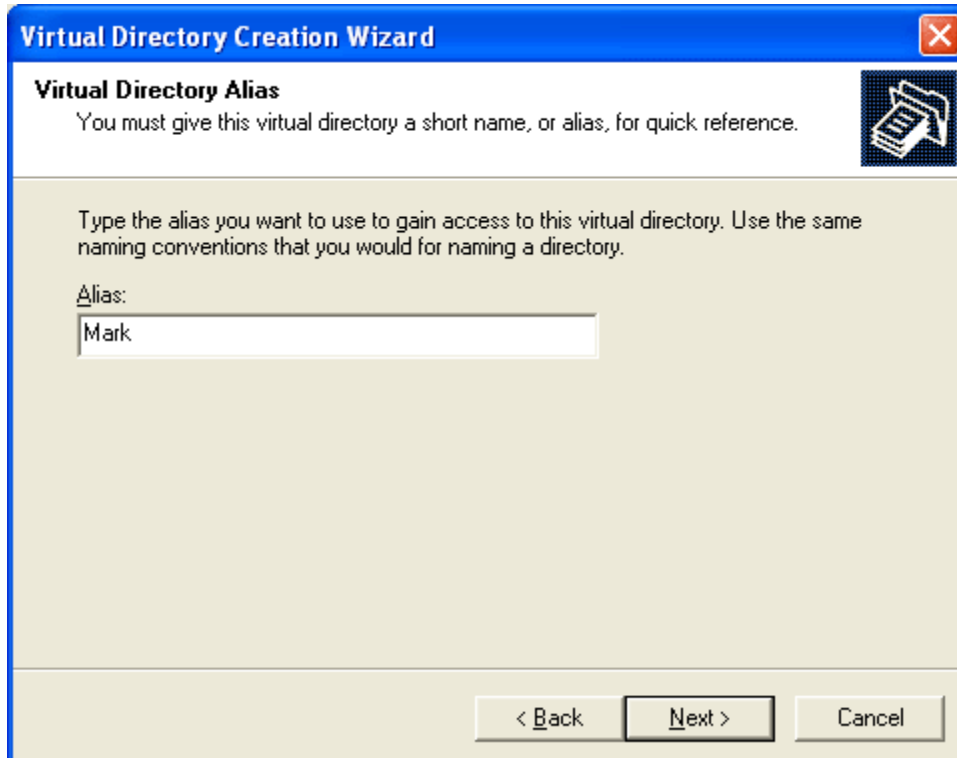
It is probable that you will have more than one individual (or group of individuals) accessing the server. You probably don't want them all to access the same directories nor see all the resources. This is also true of web sites, but we will deal with that in a later tutorial. You need to first access the IIS FTP site and select **New** (to take advantage of the wizard (Virtual Directory dialog). This will allow you to place authenticated users into their own directory when they login.



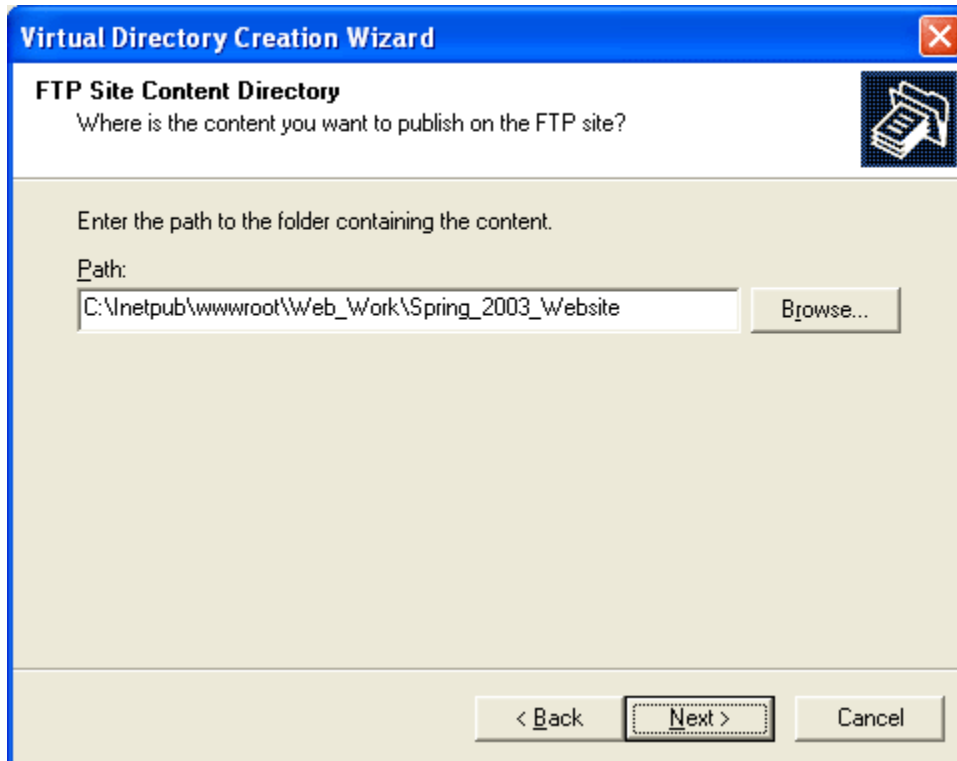
You will then be guided through a number of screens to create a virtual directory. The initial screen is shown below.



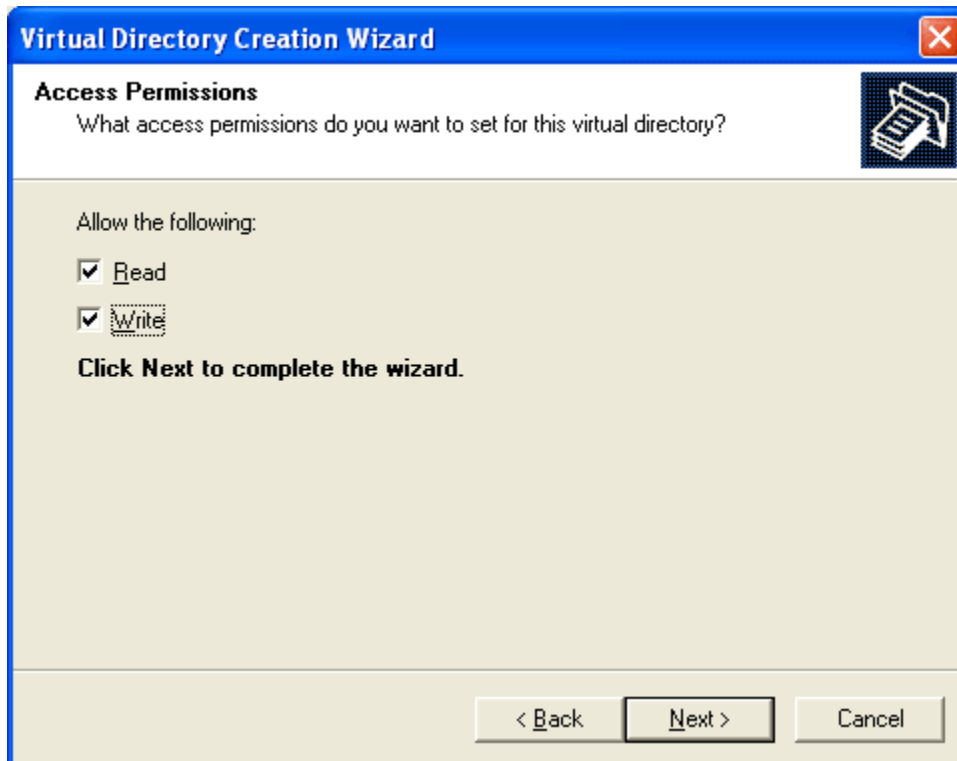
Obviously, click **Next** to proceed. You knew that, didn't you?



You are asked to provide an alias. This name should match that of the visitor (authenticated visitor).



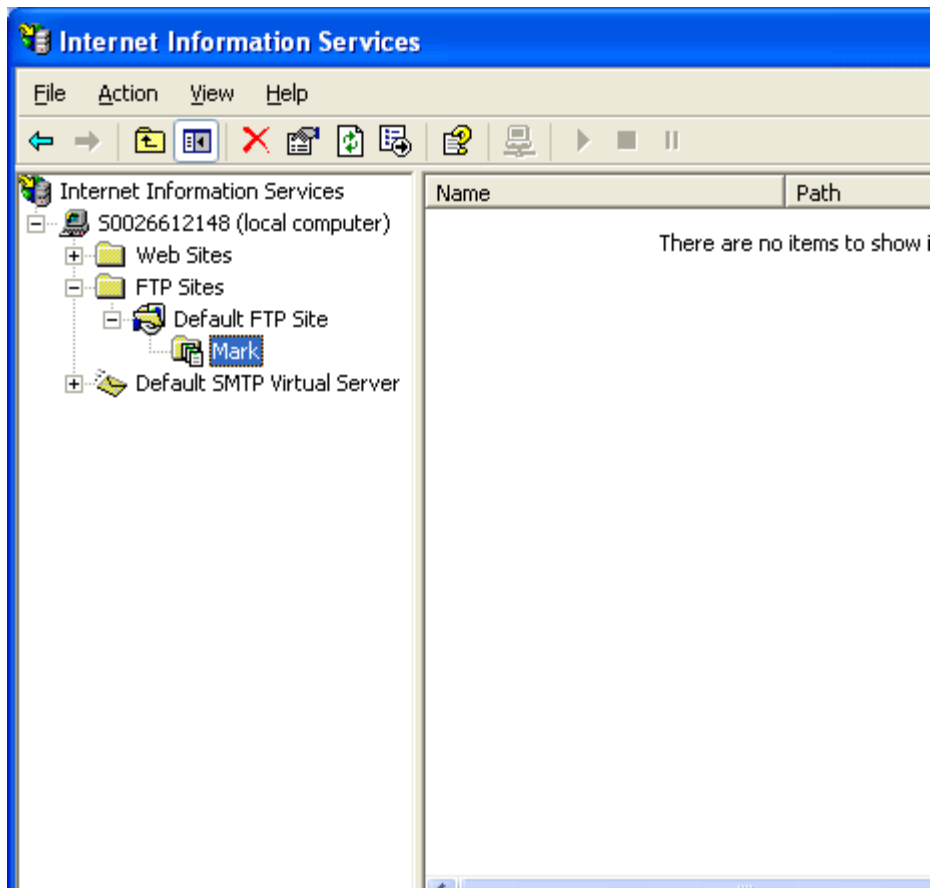
You next select the appropriate path to the contents for this alias. For the lab, you would want to create a directory called "Fred" under c:\inetpub\wwwroot.



You next set access permissions. Don't forget to allow both **Read** and **Write**.

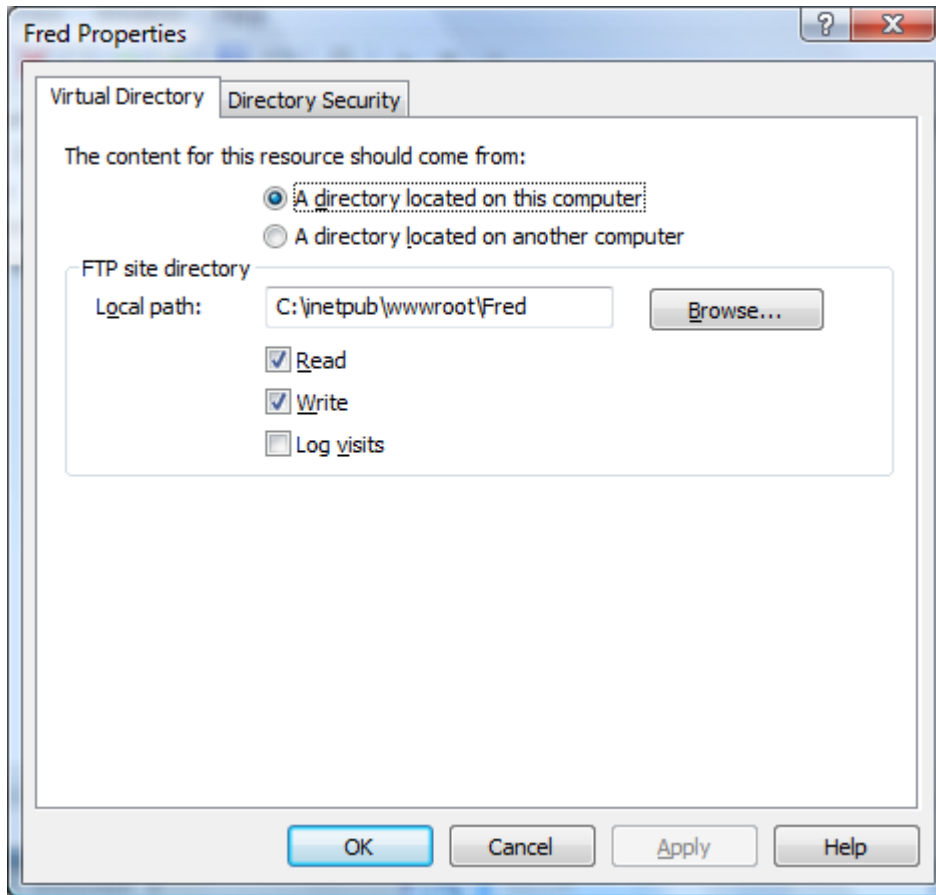


You should receive a **Success** message indicating the virtual directory has been created.



When you return to the IIS interface, you will see the additional virtual directory.

Right-click on the newly-created Fred, select properties, and you should have a screen similar to the following for the lab:



This is the screenshot you will send me for your lab.

To test, create an html page and save it as default.html. In it, put the title as "Fred's page" and in the body put "hi, I'm fred's page".

Then put the following connection string in your "My Computer" address bar:

<ftp://theusernameyouuseforyourcomputer:thepasswordyouuseforyourcomputer@localhost/fred>

Note that the files and folders I see are not the default FTP directory. Instead I am immediately directed to the appropriate location.

When it connects to the directory, drag and drop your fred page into the directory.

Then open up Internet explorer, and put the following in the address bar:

<ftp://localhost/fred/>

it should display default.html in its listing of files. Click on it, and you should see your "hi, I'm fred's page".

Obviously, you could create a separate alias for each user. We won't discuss security as we have already covered a fair amount of material for one tutorial.

Always verify the FTP access matches your web site(s)

This should be obvious (but often isn't). When you have completed your work on a web site and FTP site... verify that you can connect as desired. Try out the userids you created and make certain the views are as desired before passing this information along to the actual users. Try uploading something, creating a directory, deleting something, downloading something. Make certain it works as desired. Otherwise, expect a number of phone calls at odd times of the day and night asking you why something isn't working as desired.

If you want to limit access (so a visitor can only see what is in their directory)...

1. Place the FTP directory under the inetpub/wwwroot directory
2. point the FTP site to the appropriate directory
3. create a second FTP server under the first one (make certain it has the same name as the username of the client)
4. point this second server to the same directory you created in step 2
5. set the file permissions to full access on the second FTP directory and deny all permissions on the root directory

This will prevent the visitor from working outside their virtual directory. Try this out if you want. I don't recommend leaving the FTP sites in this state for the lab as you will probably want to perform other tasks and end up tripping over this sort of security. However, I thought you should be aware of how this is done in IIS.