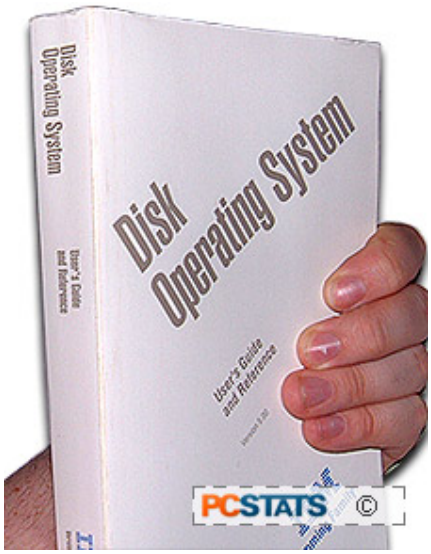
Beginners Guides: WindowsXP Command Prompt

Unlock the enigma that is DOS, and get a handle on the command prompt in Windows XP. PCstats covers the basics, and some commands you ought to know. - Version 2.1.0

Bookmark this PCstats guide for future reference.



Back in the heyday of text-based operating systems like Unix and DOS, the command prompt *was* the operating system, and not knowing how to use it meant that you could not really use your system without someone experienced in computing holding your hand. This was the era of computing clubs, when getting to grips with your computer required some serious devotion and learning.

When the most successful graphical operating systems, the various Apple Mac and Windows incarnations came along, they triggered a massive increase in the popularity of computers by simplifying the whole process of using them. No longer did users have to memorize reams of commands in order to simply configure and navigate around their computers.

On the other hand, icon based graphic interfaces tend to reduce the flexibility of the underlying operating systems considerably, especially since they are designed to be accessible to the novice user. The various Windows operating systems are instantly familiar to just about any citizen of a developed country between the ages of 10 and 35, not to mention anyone who has worked in an office within the last 10 years, but how many people know how to properly configure and customize Windows XP? Judging by the proliferation of computer service ads in every city, not many.

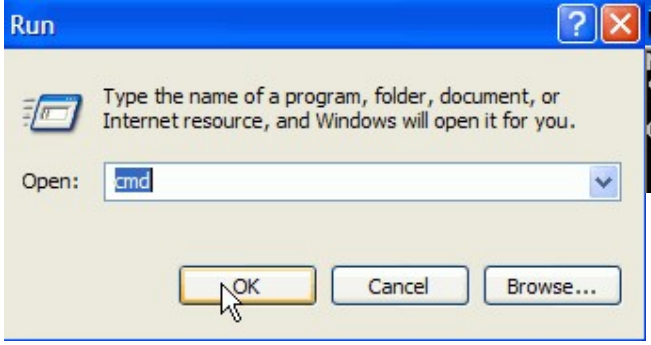
The command prompt, a purely text-driven interface, is still present at the core of Windows XP, and it contains many options that are not otherwise accessible without third-party software, but which most 'experts' could not live without. The mighty PING command being one simple example.

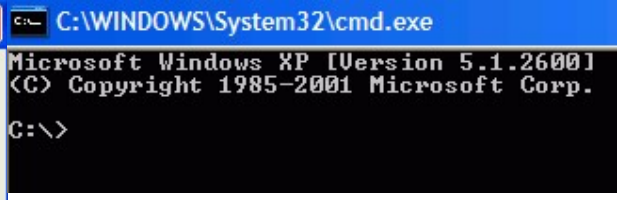
The road to becoming truly familiar with your windows XP system (and saving on service calls) involves learning many of these commands, which in turn requires at least a passing familiarity with using and navigating the command prompt. Since comprehensive manuals are no longer included with computers to detail all these commands, it's up to you to find the answers on your own. Which is where PCstats comes in.

In this PCstats Guide, we will show you how to access and use the Windows XP/2000 command prompt and reveal some useful commands for administering your system with it. If nothing else, you'll come out of this article with the ability to navigate the command prompt, which may save you endless frustration somewhere down the road.

We've split this guide into two parts; the first part deals with opening the command prompt, understanding its environment and learning the basic navigational commands to get around in its DOS-like environment. If you've used the command prompt much in the past, or DOS for that matter, you can safely ignore this section and skip ahead to the goods in the second half of this guide.

The second section of PCstats Guide to the Windows XP Command Prompt covers a myriad of useful commands and functions you can carry out with the command prompt, and your





keyboard. Note that this article is not intended as a comprehensive command prompt manual, but rather as a jumping off point for your own explorations.

Keyboard's ready? Let's begin!

Part 1: Entering and using the Windows XP command prompt

Before we get to the various interesting things you can do with the command prompt, we should first make sure that everyone is on the same level regarding how to actually open and navigate it. With modern Windows PCs, it's quite possible that many users have never even seen a command prompt, let alone learned how to use one.

Entering the Windows XP command prompt

The easiest way to get to the command prompt is by opening up the run command and typing 'cmd'. To do this: Open the start menu and click 'run.' In the text box, type 'CMD'.

This will open the command prompt, starting you off at the C:\documents and settings\(user name)> prompt. To get to the root of the C: drive, type

cd..

and press enter, then repeat this same step again. You will now be looking at the C:\> prompt, meaning the root of your main hard drive, just as if you'd opened 'my computer' and double clicked on drive (C:\).

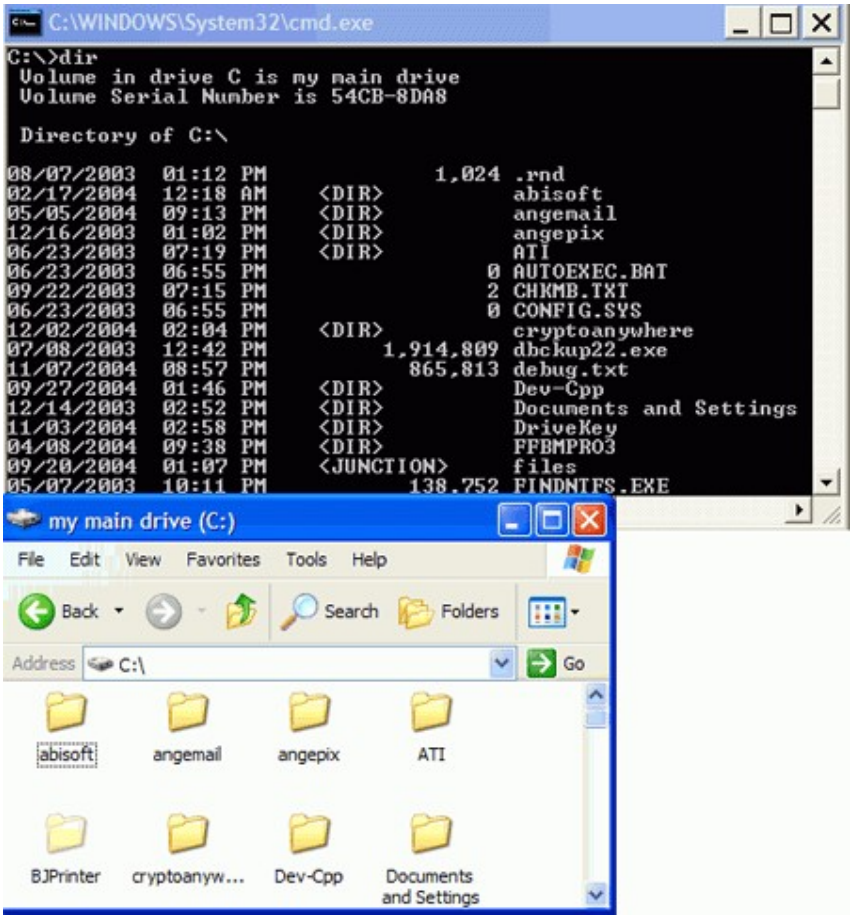
Now let's look at how to get around in the command prompt's DOS-like environment.

Navigating the Command Prompt

Like the Windows graphical environment you are used to, the command prompt uses drives and directories (folders) to organize data. Each logical drive (C:\, D:\, etc.) has it's own entry here, and contains its own set of directories and files.

The command prompt window will place you at'C:\>' by default, meaning you are looking at the logical 'C:\' drive, generally the first hard disk on your computer and the one on which Windows is installed.

As an experiment, go to 'my computer' and open your c: drive in a window. Now at the command prompt, type 'DIR'.



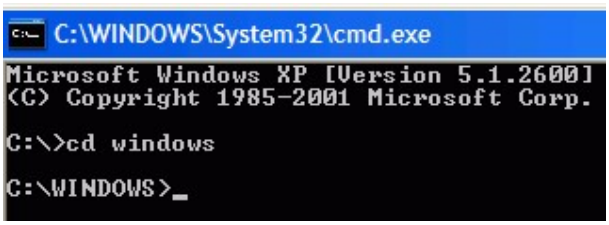
As you can see, the contents of both windows are the same, though the order will be slightly different since Explorer puts folders before files, while the directory (DIR) command simply lists all contents alphabetically.

The 'DIR' command lists the contents of the folder or drive you are currently at in the command prompt. To get a more useful listing of the files and folders in your current directory, use the 'DIR /d' or 'DIR /p' switches. The former displays the list in three columns, fitting more info on a single screen, while the latter pauses at the end of each screen of information, waiting for you to press a key before it continues scrolling. Note how the contents of the 'DIR' command are the same files shown in the Explorer window on your desktop.

Moving Between Folders

To navigate between folders in the command prompt, you can use the 'CD' command. 'CD' stands for 'Change Directory' and it works as you'd think it would. To get to the 'windows' folder from the C:\> prompt;

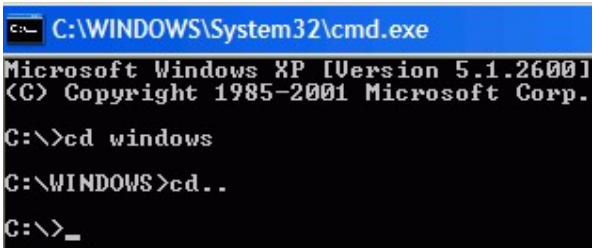
type: *cd windows*



Then, hit the enter key. Your prompt will now read 'C:\WINDOWS>' indicating you are in the 'windows' folder on the 'C:\' drive. Typing 'DIR' now will get you a list of the files in the windows folder.

To go 'back' to the parent folder or drive of the folder you are currently in;

type: *cd..*



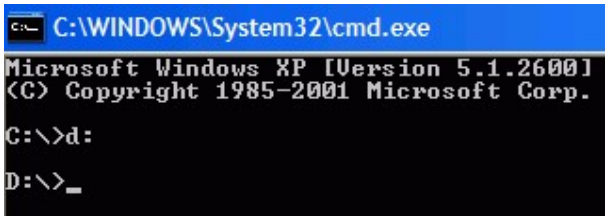
This will return you to the C:\> prompt.

Note that you can navigate multiple directories at a time by typing in the full path after the 'CD' command. Let's say you wanted to reach the C:\windows\system32\drivers folder. The easiest way to do this from the C:\> prompt would be to;

type: *cd windows\system32\drivers*

That command will navigate directly to the folder you are trying to reach. This brings us to changing drives, which is handled differently from changing folders. Instead of using the 'CD' command, you can simply type the drive letter like this;

type: *d:*

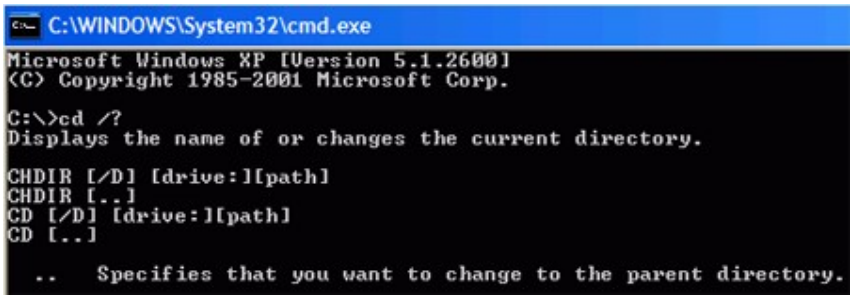


By typing the drive letter you automatically move to your most recent location in that drive. Now use the 'CD' command to navigate to the desired folder on the new drive.

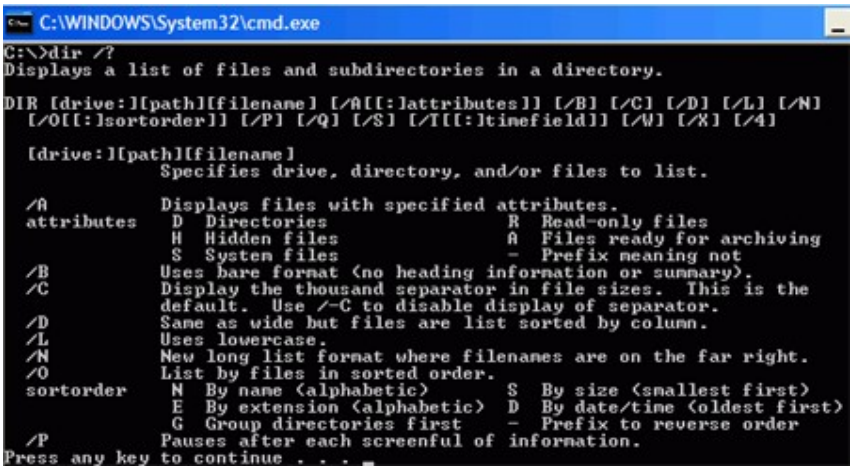
Switches and Command Help

Instructions are available for every command prompt command, and can be accessed simply by typing the command followed by ' /?'. For example, to obtain a help listing for the CD command,

type: *cd /?*



Note there is a space between the command and the switch. These help files will give you a quick blurb on the command's purpose, a guide to the proper syntax for using it, and a list of switches and other options that can be used with the command.



Switches are optional extra settings that can be used with command prompt commands to make them behave differently. For example, by default, the DIR command will list the contents of your current location in one continuous scrolling list. If you are in a directory that contains many files, like C:\windows\system32, this is not very useful. By adding the /p switch, you can make the list pause after every full screen of information is displayed, making the results easier to browse. In a similar fashion, 'DIR /w' will cause the folder contents to be listed in several columns, so you can fit more on a single screen. Switches come between the command itself and the target, like this;

type: *dir /p windows\system32*

More than one switch can be used with a command, like this;

type: *dir /s /w /p windows\system32*

Use the help (/?) switch for each command to find more useful switches and options!

Creating and Deleting Folders and Files

You can create and delete folders within the command prompt by using the make directory 'MKDIR' and remove directory 'RMDIR' commands. Making a directory is as simple as typing: *MKDIR (directory name)*

And to remove an empty directory type: RMDIR (directory name)

Note that like all command prompt commands, these depend on your current location in the command prompt. So if you were at C:\> and typed 'MKDIR myfolder', you would have

created 'C:\myfolder'. If you were at 'D:\mydata>' and typed the same command, you would create 'D:\mydata\myfolder'. You can speed things up by adding paths to the command, as we did with 'CD' above.

type: *MKDIR windows\system32\drivers\creative*

From the C:\> prompt will create the 'creative' directory in 'windows\system32\drivers'. Note that when you use the MKDIR command in this way, any folders in your command that do not exist will be created. For example, if you typed: *MKDIR windows\mike\mikedrivers* the 'mike' and 'mikedrivers' directories would both be created.

The DEL command can be used to delete files within the directories you create. Typing 'del (directory name)' will delete all files in that directory, while typing 'del (filename)' will delete that file. You can add directories to the path of the DEL command, for example:

type: *DEL windows\mikedir\mikedrivers\mike1.exe*

This would delete the 'mike1.exe' file within the directory 'windows\mikedir\mikedrivers\'.

Running Programs Within the Command Prompt

The second part of PCstats Guide to the Windows XP Command Prompt will deal with several command prompt utilities that can make dealing with Windows much easier, especially when networking. To run a utility in the command prompt, you need only type its name. For example; MKDIR, DEL and CD are all little programs that you have already run from the command prompt. Generally you need to be in the same directory or folder as the command file (.EXE, .COM or .BAT) for an application you want to run when you type its name, but the command prompt uses a system taken from previous versions of DOS (and older OS) to ensure that this is not always necessary.

A 'path' statement is automatically loaded with the command prompt, telling it to always look in certain locations on the disk when the user types in a command. For example, without this path statement, when you type: *DEL windows\system32\mikefile.txt* the command would only work if you were currently in the folder containing the DEL command file.

With the path statement, the system knows to check certain directories for command files each time the user types something. As long as it finds the DEL command somewhere within these directories, the entry above will work.

Command Redirectors

The use of redirectors on command prompt commands is not something the average user really needs to know much about (despite screams of protest from assorted Linux enthusiasts), so we will not explore this area in much detail. However, there are some command filters which can benefit any user.

A command redirector takes the output or contents of one command or file and sends it to another file or command. That's pretty much it. Simple right? Well no, but we're going to keep it simple by only using a couple of them. For a full explanation/list see the documentation on Microsoft's website. The two redirectors we are concerned about deal with redirecting the output of commands to files. This allows you to conveniently store the results of your commands in text files and spreadsheets for archiving and/or printing.

To output the results of a command to a file, use the '>' sign. For example, to copy the results of your DIR command to a text file for storage;

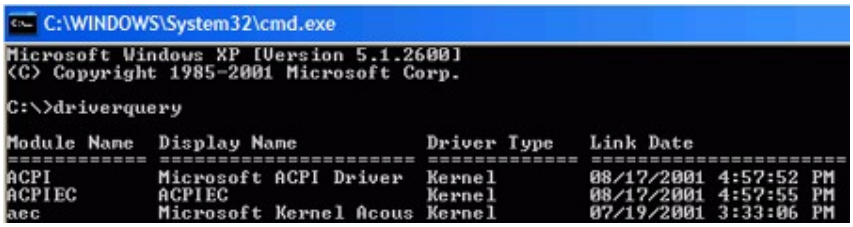
type: *Dir > c:\mydirfile.txt*

This will create a file called 'mydirfile.txt' in your C:\ directory which will contain the results of your DIR command. I think you can see the potential usefulness of this. The '>' operator creates a new file every time it is used. This means that if you use the same file name twice, you will lose the previous data. To append data onto the end of an already created file, use the '>>' redirector.

Part 2: Command Prompt Functions You Need To Know

In the second part of this guide, we will be focusing on some commands and utilities available in the command prompt and nowhere else (or at least nowhere else easily). This is by no means a complete list, just a selection of some of the most useful functions that are unique to the command prompt. Feel free to contact us if you can think of something that should be added to this list, and let us know why.

DRIVERQUERY: Displays a list of the current drivers on your system. This command can operate remotely with the /s switch. It's also the easiest way of listing all installed drivers on a Windows system without the aid of third party software.



The /FO switch can be used to format the output in CSV or table form for saving to a spreadsheet file. See the '/?' help switch for more information.

More Handy CMD Commands

FSUTIL: This is a comprehensive set of file system management tools for XP disguised as a simple command. Using the FSUTIL submenus, you can perform a variety of file system operations, some of which cannot otherwise be done without editing the registry.

Here's some interesting uses for the FSUTIL command and its various sub-commands:

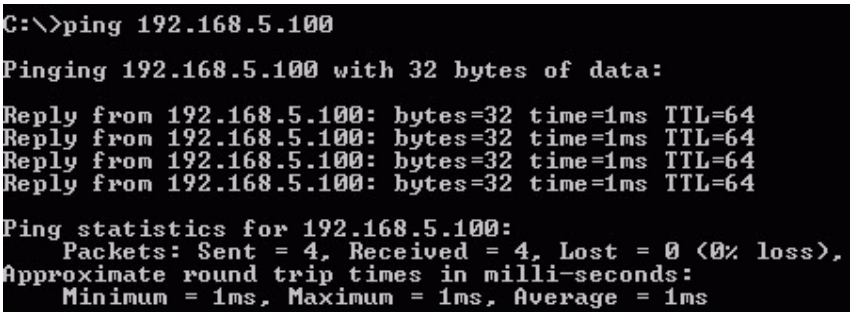
By typing in *'fsutil behavior set disable8dot3 1'* you can disable the automatic supplemental support for DOS 8.3 character filenames that Windows XP uses by default. This can speed up folder access and file creation tasks, and is one of the registry changes PCSTATS revealed in our 101 Tech Tips Guide.

In a similar way, typing *'fsutil behavior set disablelastaccess 1'* prevents Windows XP from automatically updating the 'last accessed' timestamp on files and folders contained in an NTFS-formatted drive each time they are read. This can considerably speed up file and folder browsing, but may mess up your automated data backups. This is another tweak that can only otherwise be done by editing the registry. The *'fsutil fsinfo'* submenu gives you access to a huge amount of detailed information about your drives. Explore it and see what you find.

PING: Quite possibly the most often used command prompt command ever in any version of Windows, considering it can't be accessed from anywhere else. It's the 'killer app' of the command prompt, if you will. The PING command queries a remote (or local) IP address by sending a stream of data to it and listening for a response. If no response is received, you know the intended computer is either not receiving or not responding to the ping. If a

response is received, you know the two systems can communicate over the network. This ability makes it the first tool in any network troubleshooter's arsenal.

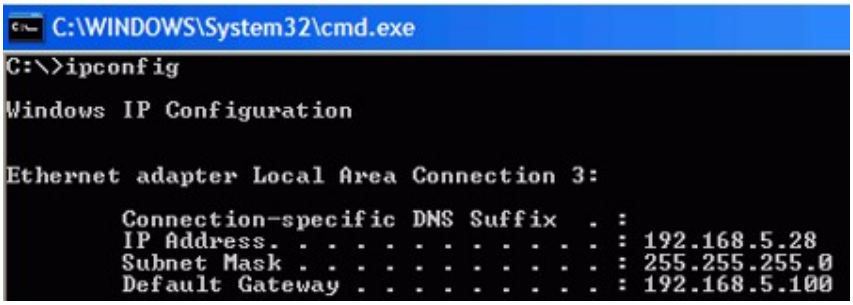
By default, the XP ping command sends 4 groups of data packets at intervals and reports any successful replies. The '-t' switch sends a continuous stream, only halted by pressing CTRL+C or closing the command prompt window.



You can also ping a DNS or URL address, like www.google.com, which will display that site's public IP address if successful.

All About IPConfig

IPCONFIG: This command is used to retrieve information about your system's network setup, or broadband internet connection, and details that are otherwise buried in layers of menus in the GUI. The IPCONFIG command used by itself provides you with a concise list of the network interfaces on your system and their IP addresses.



Using the '/all' switch throws you into a whole new level of detail, showing DNS information as well as the MAC addresses and other information attached to each of your network cards. This information is very hard to come by anywhere else within Windows XP.

Information is not all that the IPCONFIG command provides. The '/release', '/renew', and '/flushDNS' commands serve an essential purpose in Windows XP, performing functions you cannot duplicate elsewhere.

The 'ipconfig /release' and 'ipconfig /renew' commands direct your network adaptor(s) to drop and renew their current IP addresses respectively. This is useful in networks that use DHCP (Dynamic Host Configuration Protocol) to distribute IP addresses, such as with cable modem broadband internet providers. It forces your system to give up its current address and seek a new one from the DHCP server, which is essential when attempting to resolve certain network problems.

The 'ipconfig /flushdns' command serves an equally useful function. DNS ('Domain Name System') is the system that maps IP addresses to Internet addresses like www.pcstats.com (among other things). A full explanation of how DNS works is beyond the scope of this

article, but see PCSTATS Networking Guide here for more details if you're interested in learning.

By default, your computer system keeps a DNS resolver cache which stores the IP address attached to frequently used DNS names (and Internet URLs, which are essentially the same thing). This enables your system to bring up frequently accessed web pages quickly, without the need to first consult a chain of DNS servers on the Internet to find out what IP address is associated with, say www.pcstats.com.

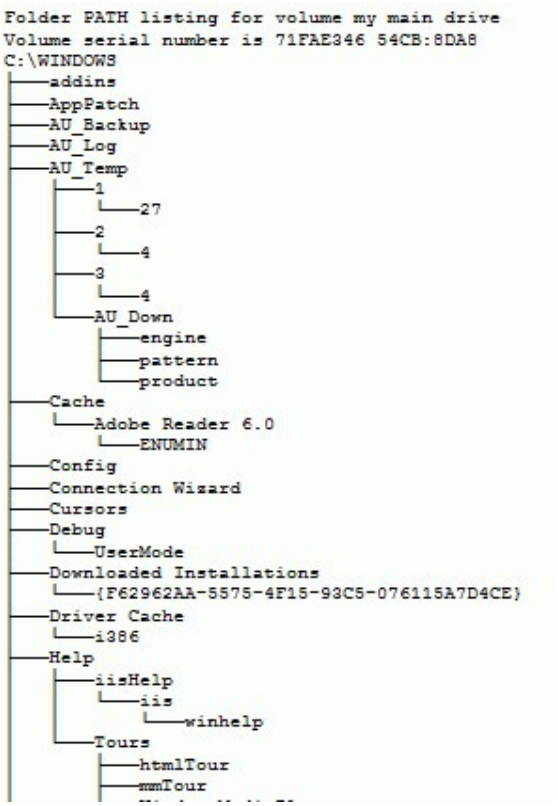
If you are in a network that uses an Internal DNS server as the first point in this chain of servers, that DNS server's IP address is going to be a more or less permanent resident of your DNS cache. So what happens if that server changes or goes down? Even if there is a backup, your system still has the original IP address in its cache, and will check that address first whenever you type in a request for a web page. Obviously, querying a non-existent DNS server is not going to get you far. Unfortunately, even if you change the address of the DNS server to a valid one in your network connection settings, your system will ignore it in favour of the entry in the cache. This can lead to much frustration.

By using the 'ipconfig /flushdns' command, you delete the contents of the DNS resolver cache, meaning that your system will now recheck its settings to see where it should be going to get DNS addresses. Problem solved.

The 'ipconfig /displaydns' command will show you the current contents of your system's DNS resolver cache.

Tree and Netstat

TREE: The TREE command graphically displays the subfolders contained in the folder you point it to, using a tree structure to show the hierarchy. The '/f' switch for the TREE command will also display all the files in each folder. You can send the output of the tree command to a text file by typing: *'tree (your drive and directory) > mytreefile.txt'* and open it in word (with DOS formatting) for a nice printable directory list.



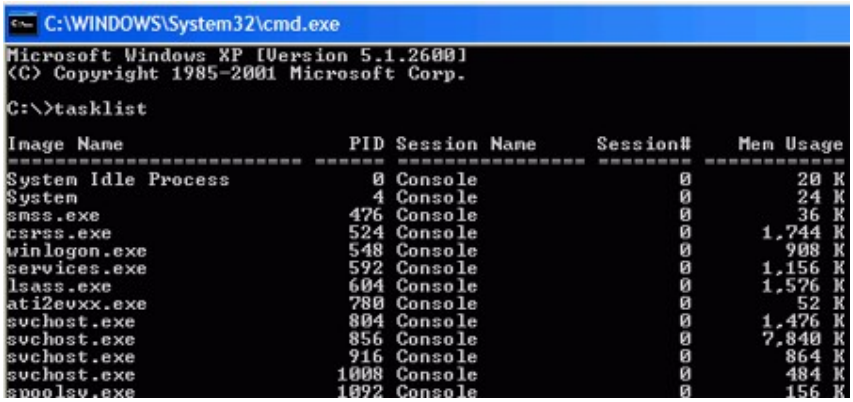
NETSTAT: The NETSTAT command can be used to display a wealth of networking and port information that would otherwise not be available through the Windows interface without the use of third-party software. Its most vital function is the ability to show what ports are communicating data or open and listening for data at any given time.

When used by itself, 'netstat' will display currently active network connections between your system and any other systems on your network or the Internet. The '/a' switch shows all current connections along with all open and listening ports.

The '/o' switch shows the process ID which belongs to each connection. This can be handy for tracking and identifying unwanted software such as spyware on your system. Use the TASKLIST command to match this process ID to the processes currently running on your system.

Tasklist and SystemInfo

TASKLIST: This command will give you a quick list of the processes currently running on your system. It can be configured to show a lot more information than the Task Manager Window, so it's worth mentioning.



There are a lot of options and switches you can use to order this information; two notable ones we use are: *'tasklist /svc'* which will show the services that are running under each process and *'tasklist /v'* which gives a more detailed (verbose) listing for each process.

SYSTEMINFO: This command can be used to display essential information about your computer and its operating system, including all installed patches, Service Packs and Hotfixes. SYSTEMINFO can also be used on remote systems, and you can output the results to a CSV or table for use in Excel or other spreadsheet programs by using the '/FO CSV' or '/FO TABLE' switches in conjunction with the '>' filter which passes the results to a file.

An example would be: *systeminfo /fo csv > c:\systeminfofile.txt*

Which would output your system information to the c:\systeminfo.txt file in CSV (Comma Separated Variable) format, which Excel can turn into a spreadsheet.

PATHPING: This command essentially combines the PING and TRACERT commands into a single, potent network-tracing tool. In its default usage, pathping will first list the number of hops (steps through different routers or network servers) required to reach your destination IP address or URL, then it will ping each step along the way 100 times, giving you a clear idea of the speed of the connection between your system and your target as well as any weak links in the network.