3.2 CONCEPT OF DOMAIN

Domain :

"A group of computers that are part of a network and share a common directory database. A domain is administered as a unit with common rules and procedures. Each domain has a unique name."

An Active Directory domain is a collection of computers defined by the administrator of a windows network.

These computers share a common directory database, security policies, and security relationship with other domains. An Active Directory domain provides access to the centralized user accounts

and group accounts maintained by the domain administrator. An Active Directory forest in made up of one or more domains, each of which can span more than one physical location.

A DNS domain is any tree or subtree within the DNS namespace. Although the names for DNS domains often correspond to Active Directory domains, DNS domains should not be confused with Active Directory domains.

Active Directory :

The directory service that stores information about objects on a network and makes this information available to users and network administrators. Active directory gives network users access to permitted resources anywhere on the network using a single logon process. It provides network administrators with an intuitive, hierarchical view of the network and a single point of administration for all network objects.

Domain Name System (DNS) :

A hierarchical, distributed database that contains mappings of DNS domain names to various types of data, such as IP addresses. DNS enables the location of computers and services by user-friendly names, and it also enables the discovery of other information stored in the database.

To join a Domain :

You must be logged on as an administrator or a member of the Administrators group in order to complete this procedure. If your computer is connected to a network, network policy settings might also prevent you from completing this procedure.

1. Open **System** in Control Panel.
2. On the **Computer Name** tab, click **Change**.
3. Under **Member of**, click **Domain**, type the name of the **Domain** you want to join, and then click **OK**.
4. You will be prompted to provide a **User Name** and User **Password** to join the computer to the **Domain**.
5. Click **OK** to close the **System Properties** dialog box. You will be prompted to restart your computer to apply your changes.

Notes :

- To open System, click **Start**, click **Control Panel**, and then double-click **System**.
- You can also use the Network Identification Wizard to join a **domain**. To use the wizard, perform Step 1 above. On the **Computer Name** tab, click **Network ID**, and follow the instructions on your screen.
- It is recommended that you use computer names that are 15 characters or fewer. If your computer has **TCP/IP** networking **protocol** installed, the computer name can be up to 63 characters long but should only contain the numbers 0-9, the letters A-Z and a-z, and hyphens. You can use other characters, but doing so might prevent other users from finding your computer on the network. If your network is using the Microsoft **DNS server**, you can use any characters except periods. If other networking protocols are installed without TCP/IP, the name is limited to 15 characters.
- If you specify a computer name longer than 15 characters and you want longer names to be recognized by the **Active Directory domain**, the **domain** administrator must enable registration of DNS names that are 16 bytes or more.
- If you rename your computer or workgroup when it is disconnected from the network, duplicate computer names might result. Check with your network administrator before renaming your computer.

3.3 WINDOWS ADMINISTRATION TOOLS :

3.3.1 Event Viewer :

Event Viewer maintains logs about program, security, and system events on your computer. You can use Event Viewer to view and manage the event logs, gather information about hardware and software problems, and monitor Windows security events.

To open Event Viewer, click Start, point to Settings, and then click Control Panel. Double-click Administrative Tools, and then double-click Event Viewer.

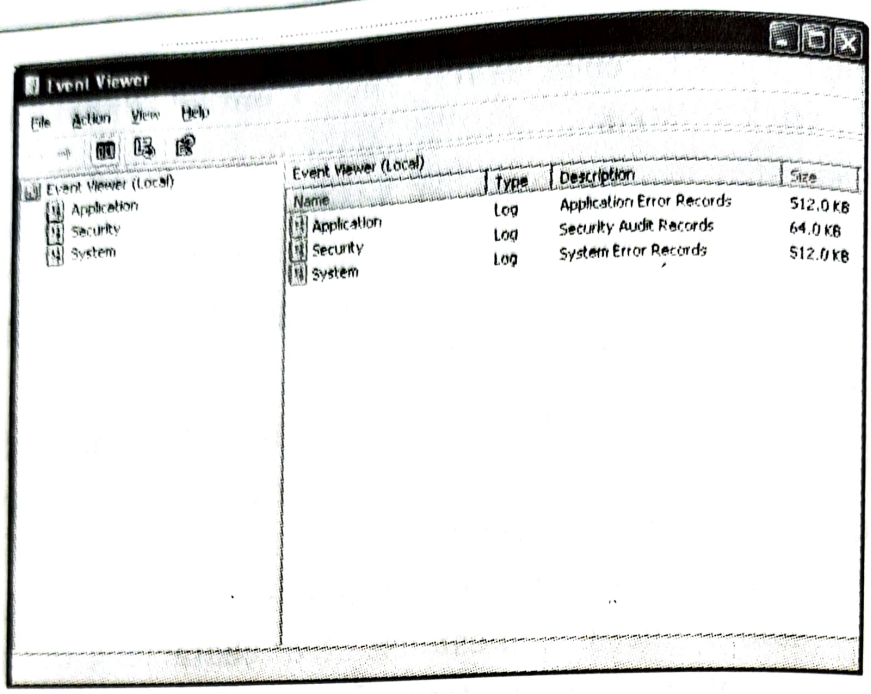

Figure 3.18 Event Viewer

A computer running any version of Windows records events in three kinds of logs :

1. Application log :

The application log contains events logged by applications or programs. For example, a database program might record a file error in the application log. Program developers decide which events to monitor.

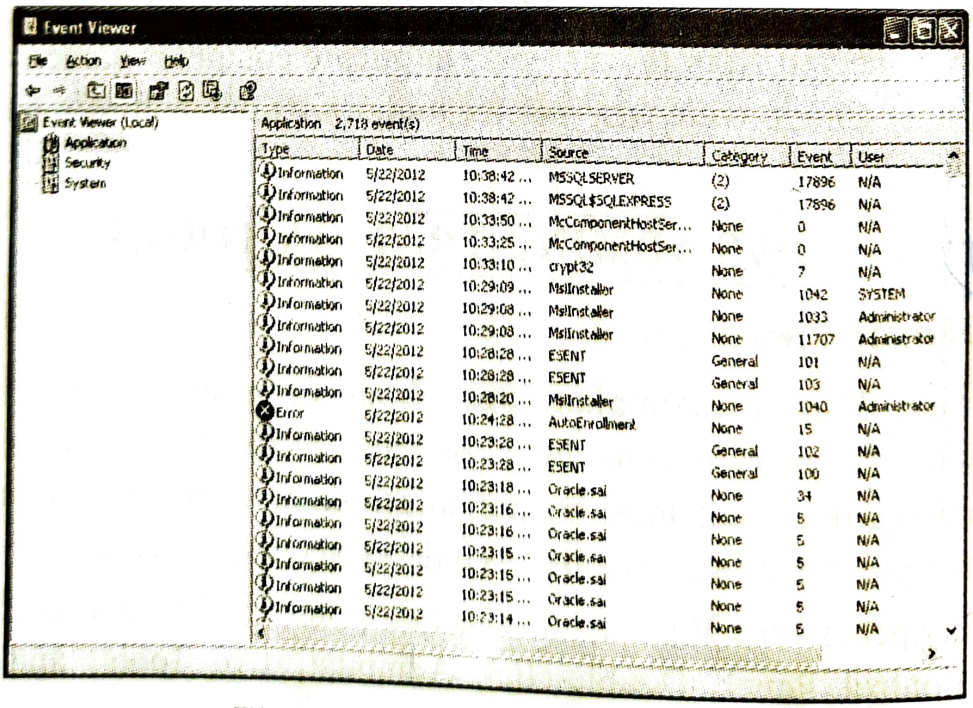

Figure 3.19 Application Log

2. Security log :

The security log records events such as valid and invalid logon attempts, as well as events related to resource use, such as creating, opening, or deleting files or other objects. An administrator can specify what events are recorded in the security log. For example, if you have enabled logon auditing, attempts to log on to the system are recorded in the security log.

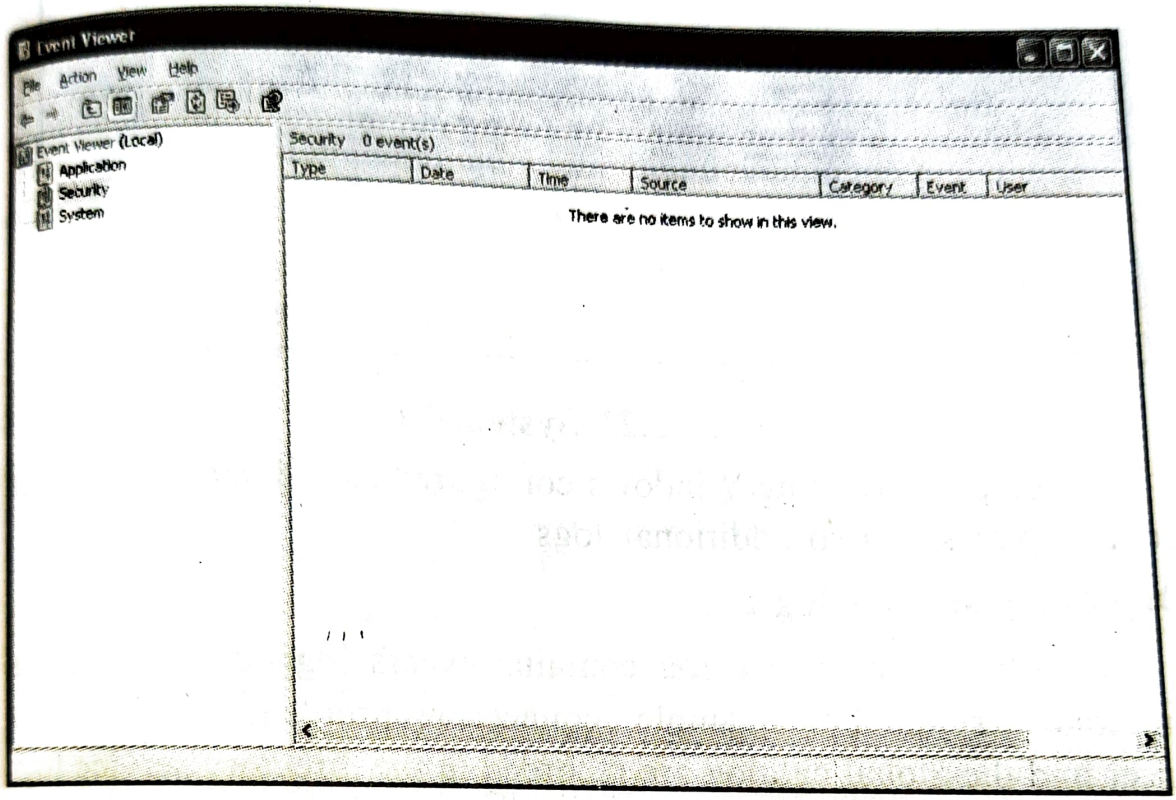

Figure 3.20 Security Log

3. System log:

The system log contains events logged by Windows system components. For example, the failure of a driver or other system component to load during startup is recorded in the system log. The event types logged by system components are predetermined by Windows.

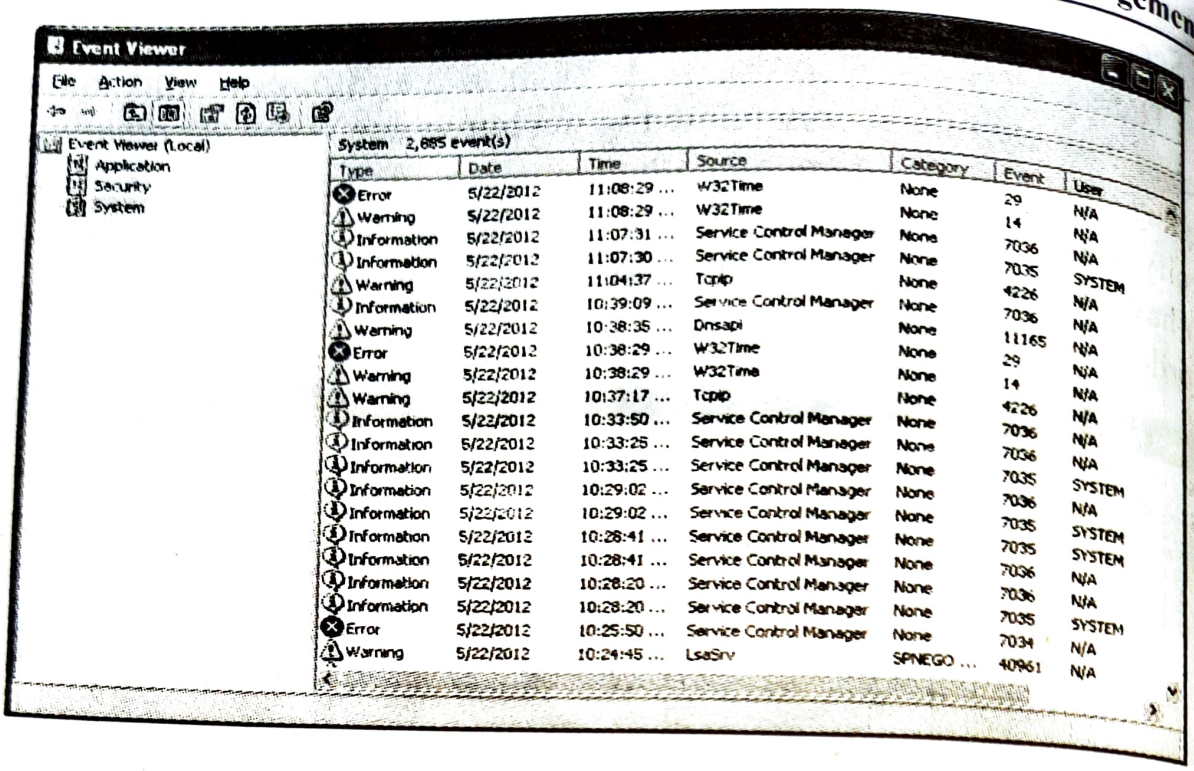

Figure 3.21 System Log

A computer running Windows configured as a domain controller records events in two additional logs :

Directory service log :

The directory service log contains events logged by Windows directory service. For example, connection problems between the server and the global catalog are recorded in the directory service log.

File Replication service log :

The File Replication service log contains events logged by Windows File Replication service. For example, file replication failures and events that occur while domain controllers are being updated with information about sysvol changes are recorded in the file replication log.

A computer running Windows configured as a Domain Name System (DNS) server records events in an additional log :

DNS server log :

The DNS server log contains events logged by Windows DNS service. Events associated with resolving DNS names to Internet Protocol (IP) addresses are recorded in this log.

Notes :

- The Event Log service starts automatically when you start Windows.
- All users can view application and system logs. Security logs are accessible only to system administrators.
- By default, security logging is turned off. To enable security logging, use Group Policy to set the Audit policy. The administrator can also set auditing policies in the registry that cause the system to halt when the security log is full.

The Event description :

- The format and contents of the event description vary, depending on the event type. The description is often the most useful piece of information, indicating what happened or the significance of the event.
- The event logs record five types of events :

Event type	Description
Error	A significant problem, such as loss of data or loss of functionality. For example, if a service fails to load during startup, an Error will be logged.
Warning	An event that is not necessarily significant, but may indicate a possible future problem. For example, when disk space is low, a Warning will be logged.
Information	An event that describes the successful operation of an application, driver, or service. For example, when a network driver loads successfully, an Information event will be logged.
Success Audit	An audited security access attempt that succeeds. For example, a user's successful attempt to log on the system will be logged as a Success Audit event.
Failure Audit	An audited security access attempt that fails. For example, if a user tries to access a network drive and fails, the attempt will be logged as a Failure Audit event.

Event Properties :

The following table lists the common event properties.

Property Name	Description
Source	The software that logged the event, which can be either a program name such as "SQL Server," or a component of the system or of a large program such as a driver name. For example, "Elnkii" indicates an Ether Link II driver.
EventID	A number identifying the particular event type. The first line of the description usually contains the name of the event type. For example, 6005 is the ID of the event that occurs when the Event log service is started. The first line of the description of such an event is "The Event log service was started." The Event ID and the Source can be used by product support representatives to troubleshoot system problems.
Level	A classification of the event severity: Error, Information, or Warning in the system and application logs; Success Audit or Failure Audit in the security log. In the Event Viewer normal list view, these are represented by a symbol.
User	The user name of the user on whose behalf the event occurred. This name is the client ID if the event was actually caused by a server process, or the primary ID if impersonation is not taking place. Where applicable, a security log entry contains both the primary and impersonation IDs. Impersonation occurs when one process to take on the security attributes of another.
OpCode	Contains a numeric value that identifies the activity or a point within an activity that the application was performing when it raised the event. For example, initialization or closing.

Logged	The date and local time that the event occurred.
Task Category	Used to represent a subcomponent or activity of the event publisher.
Keywords	A set of categories or tags that can be used to filter or search on events. Examples include "Network," "Security," or "Resource not found."
Computer	The name of the computer on which the event occurred.

3.3.2 Computer Management :

You can use Computer Management to manage local and remote computers.

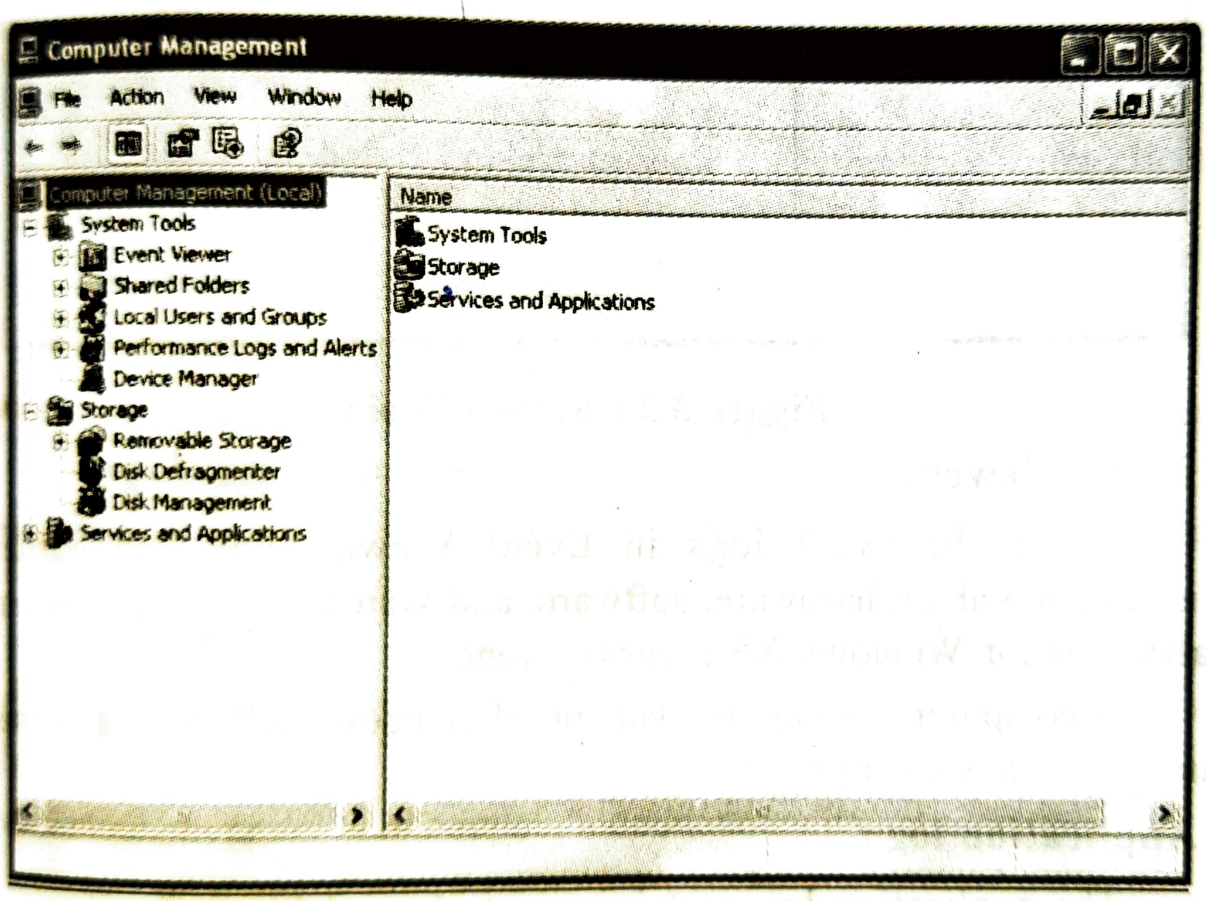

Figure 3.22 Computer Management

3.3.2.1 System Tools :

System Tools overview :

System Tools is the first item in the ***Computer Management console tree***. You can use the default tools, Event Viewer, Shared

Folders. Local Users and Groups. Performance Logs and Alerts, and Device Manager, to manage system events and performance on the target computer.

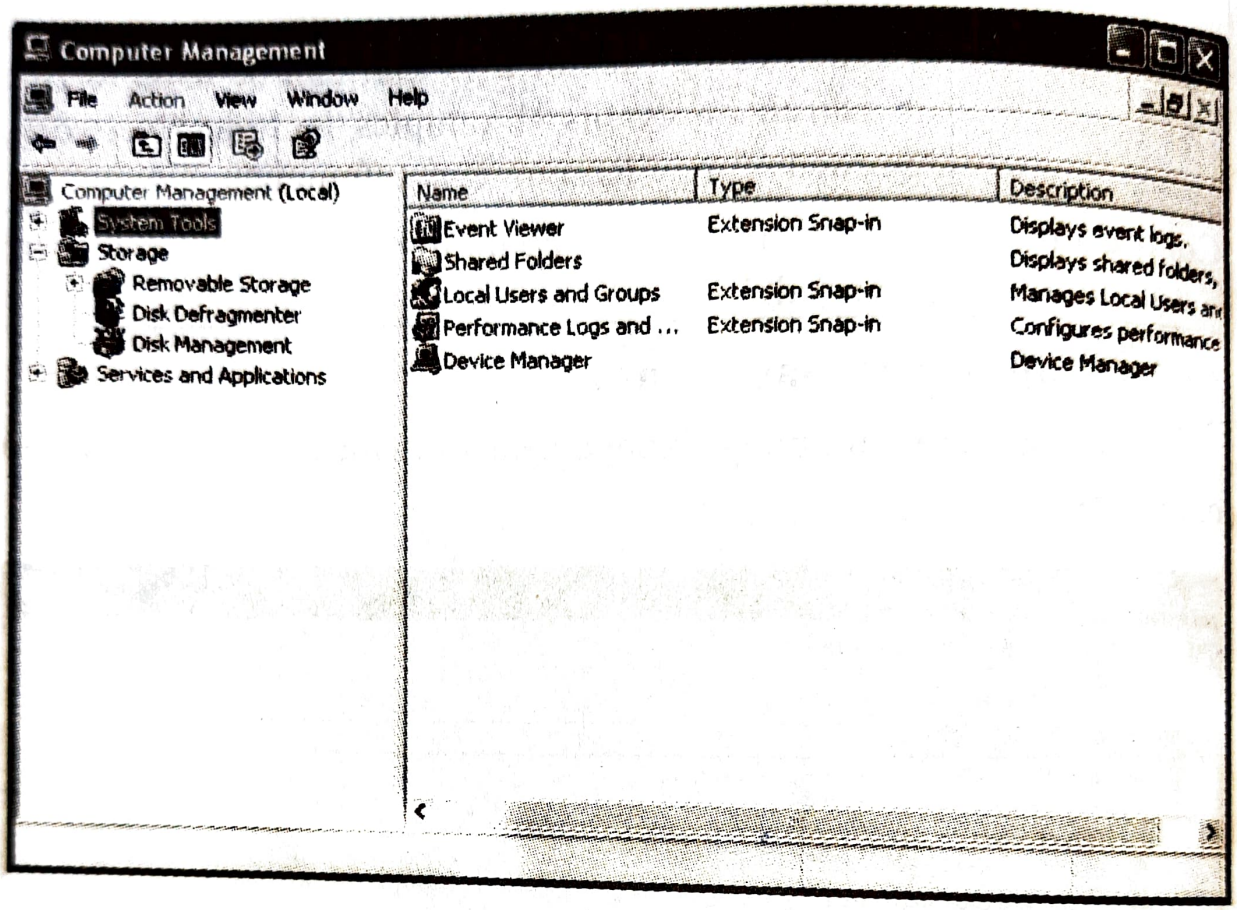

Figure 3.23 System Tools

Event Viewer :

Using the event logs in Event Viewer, you can gather information about hardware, software, and system problems. You can also monitor Windows XP security events.

A computer running any version of Windows XP records events in three kinds of logs:

Application log :

The application log contains events logged by applications or programs. For example, a database program might record a file error in the application log. Program developers decide which events to monitor.

Security log :

The security log records events such as valid and invalid logon attempts, as well as events related to resource use such as creating, opening, or deleting files or other objects. An administrator can specify what events are recorded in the security log. For example, if you have enabled logon auditing, attempts to log on to the system are recorded in the security log.

System log :

The system log contains events logged by Windows XP system components. For example, the failure of a driver or other system component to load during startup is recorded in the system log. The event types logged by system components are predetermined by Windows XP.

Shared Folders :

You can use Shared Folders to view a summary of connections and resource use for local and remote computers.

With Shared Folders, you can :

- Create, view, and set permissions for shared resources.
- View a list of all users who are connected over a network to the computer, and disconnect one or all of them.
- View a list of files that have been opened by remote users, and close one or all of the open files.

The subfolders in Shared Folders contain information, arranged in columns, about all the shared resources, sessions, and open files on the computer.

Create and distribute a certificate trust list (CTL). A certificate trust list is a signed list of root certification authority (CA) certificates that an administrator considers reputable for designated purposes such as client authentication or secure e-mail. For example, if you want to trust a certification authority's certificates for IPSec, but not for client authentication, you can implement that trust relationship with a certificate trust list.

Establish common trusted root certification authorities. You can use this policy setting to make computers and users subject to common root certification authorities (in addition to the ones that they already trust individually). It is not necessary to use this policy setting for certification authorities in a domain, because they are already trusted by all users and computers in the domain. This policy is primarily for establishing trust in a root certification authority that is not a part of your organization.

Add encrypted data recovery agents, and change the encrypted data recovery policy settings. For more information about this policy setting, see Recovering data. For a general overview of the Encrypting File System (EFS).

It is not necessary for you to use these public key policy settings in Group Policy to deploy a public key infrastructure in your organization. However, these settings give you additional flexibility and control when you establish trust in certification authorities, issue certificates to computers, and deploy EFS across a domain.

3.5 WINDOWS MMC AND SNAP-INS

The Microsoft Management Console (MMC) is a tool used to create, save, and open collections of administrative tools, called consoles. Consoles contain items such as snap-ins, extension snap-ins, monitor controls, tasks, wizards, and documentation required to manage many of the hardware, software, and networking components of your Windows system. You can add items to an existing MMC console, or you can create new consoles and configure them to administer a specific system component.

Open MMC :

➠ To open MMC, click **Start**, and then click **Run**. In the **Open** box, type mmc.

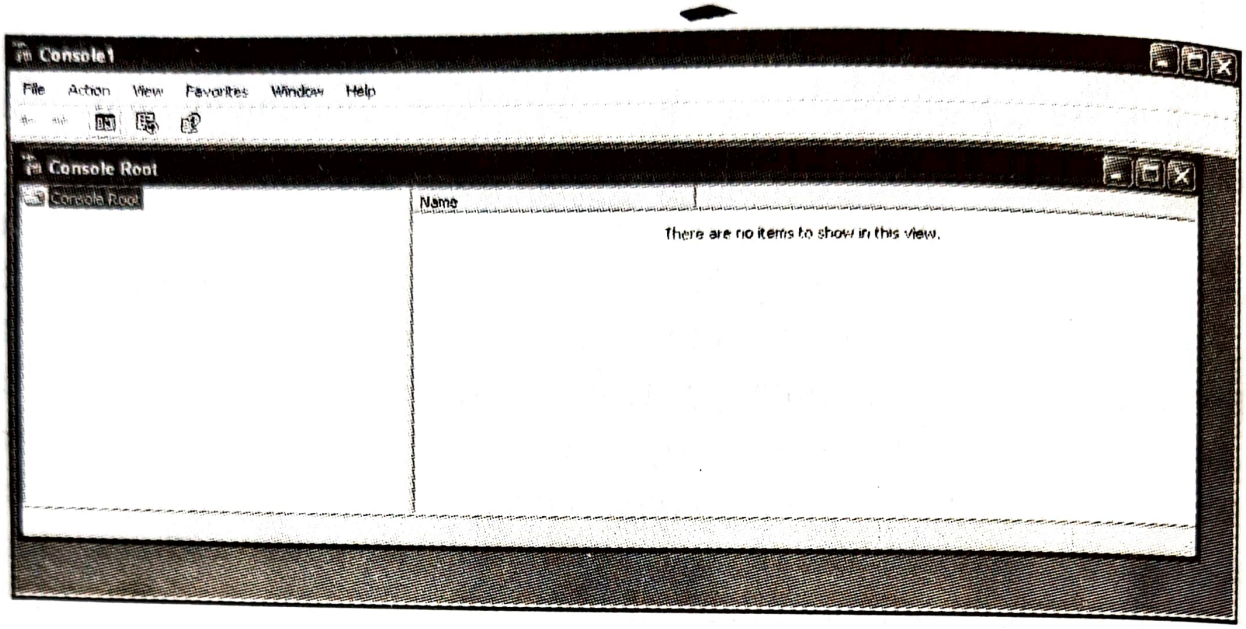

Figure 3.33 MMC

There are two general ways that you can use MMC: in *user mode*, working with existing MMC consoles to administer a system, or in *author mode*, creating new consoles or modifying existing MMC consoles.

Microsoft Management Console (MMC) hosts administrative tools that you can use to administer computers, services, other system components, and networks. You can add one or more of these administrative tools, called snap-ins.

Snap-ins :

A type of tool you can add to a console supported by MMC. A stand-alone snap-in can be added by itself; an extension snap-in can only be added to extend the function of another snap-in.

To add a snap-in

1. Open MMC.
2. On the File menu, click Add/Remove Snap-in.
3. In the Add/Remove Snap-in dialog box, click Add.
4. In the Add Standalone Snap-in dialog box, click the snap-in you want to add to the console, and then click Add.
5. You can add additional snap-ins by repeating steps 2 through 4.

3.6 SYSTEM CONFIGURATION UTILITY (MSCONFIG) :

Description :

- System Configuration is a tool that can help identify problems that might prevent Windows from starting correctly. You can start Windows with common services and startup programs turned off and then turn them back on, one at a time. If a problem doesn't occur when a service is turned off, but does occur when that service is turned on, then the service could be the cause of the problem.
- System Configuration is intended to find and isolate problems, but it's not meant as a startup management program. To permanently remove or turn off programs or services that run at startup

Uses msconfig:

- The more programs you have running on your computer at once, the more likely it is that your computer will either run slowly or even crash.
- Every time you boot your computer "hidden" programs load in the background.
- Some of these hidden programs are essential, but most aren't.
- Turning off some of these hidden programs can significantly increase your computer's performance and reliability.

How to access MSConfig :

Start ==> Run ==> type: msconfig ==> Click 'OK' or hit Enter This will start something called the Microsoft System Configuration Utility aka msconfig.

You will have 6 tabs.

General,

SYSTEM.INI,

WIN.INI,

BOOT.INI,

Services,

Startup.

Each of these tabs has their own purpose.

This launches Microsoft's System Configuration Utility.

Figure 3.34

The next step is as follows :

Write msconfig in Run Window

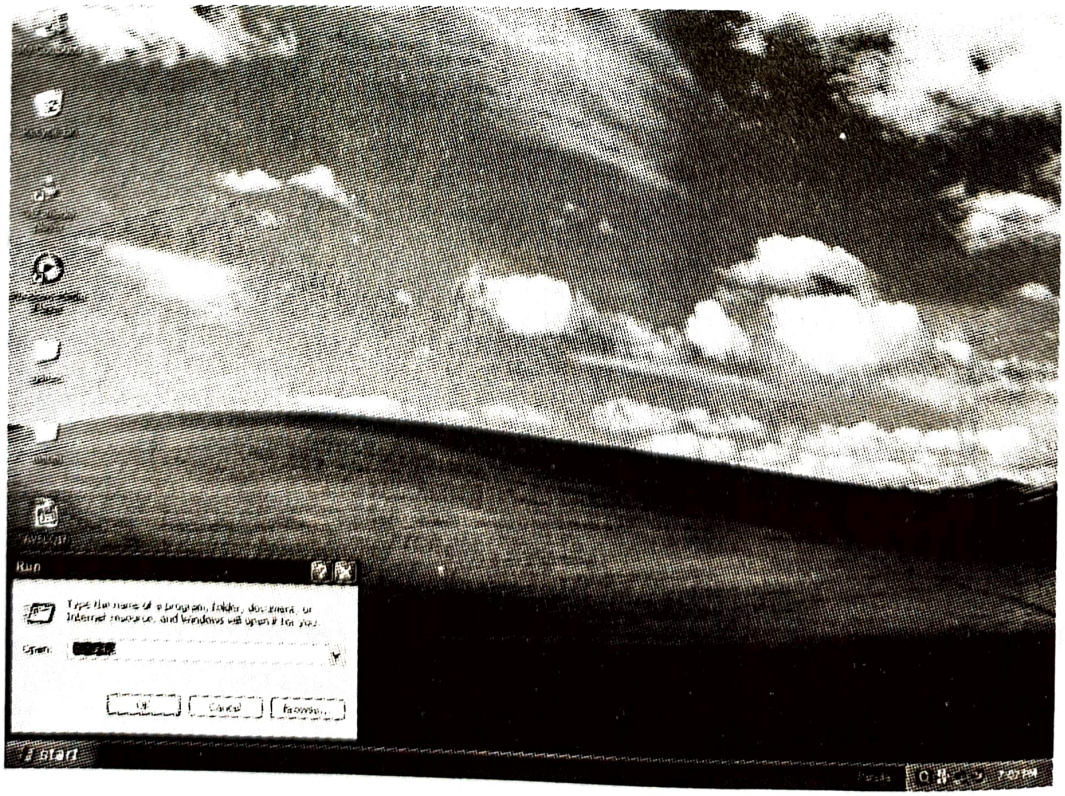

Figure 3.35

General Tab Of msconfig :

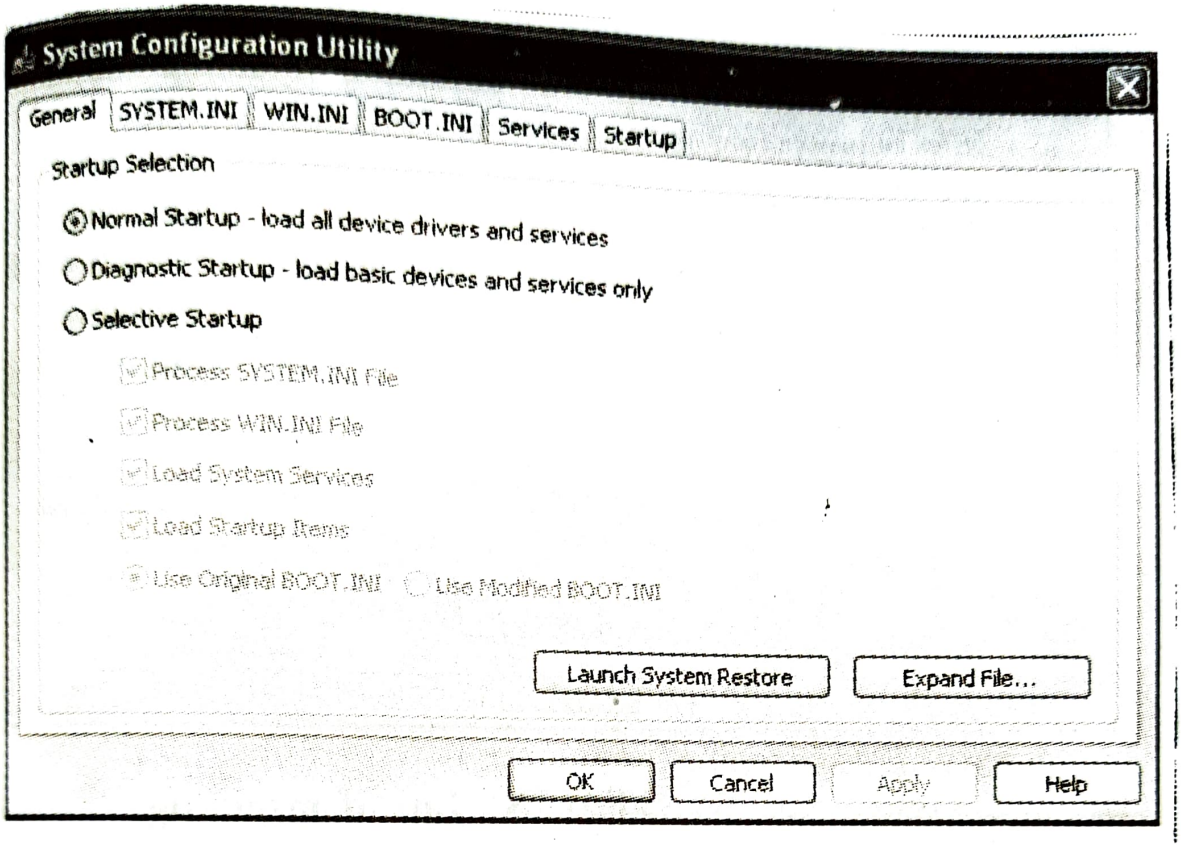

Figure 3.36 General Tab of msconfig

Lists choices for startup configuration modes:

- Normal startup. Starts Windows in the usual manner. Use this mode to start Windows after you're done using the other two modes to troubleshoot the problem.
- Diagnostic startup. Starts Windows with basic services and drivers only. This mode can help rule out basic Windows files as the problem.
- Selective startup. Starts Windows with basic services and drivers and the other services and startup programs that you select.

SYSTEM.INI Tab Of msconfig:

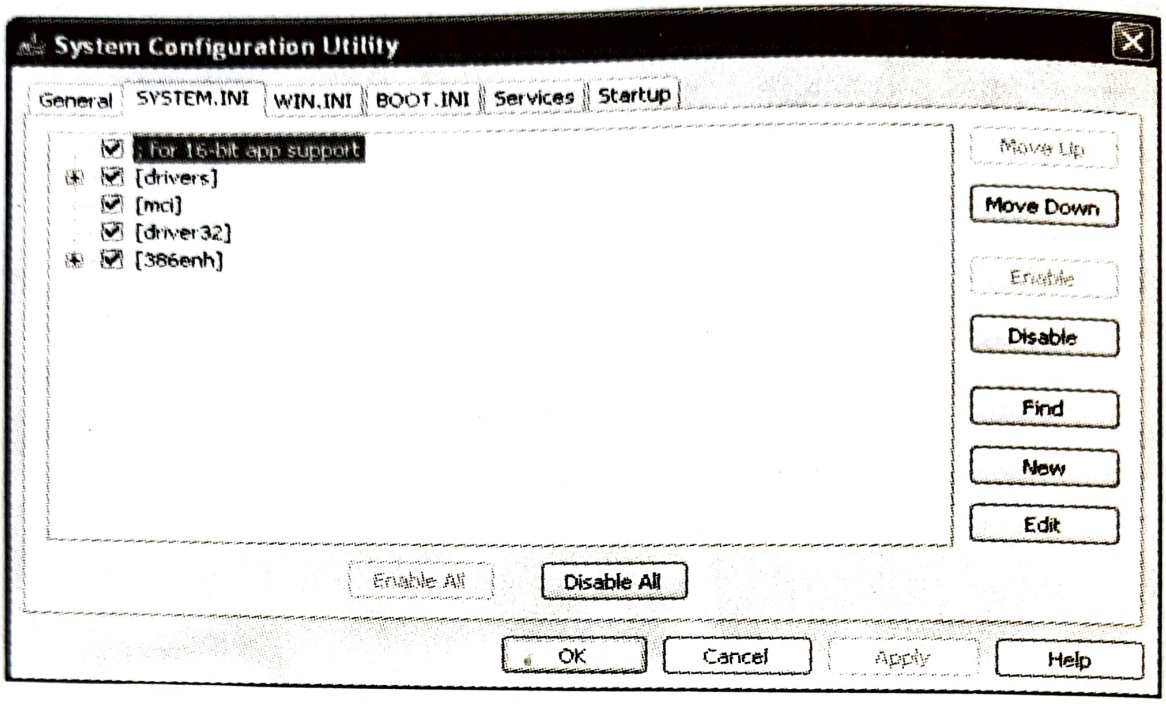

Figure 3.37 SYSTEM.INI Tab of msconfig

- **SYSTEM.INI** was a basic startup file used in early versions of Microsoft Windows to load device drivers and the default Windows shell (Program Manager or Windows Explorer). Many of these settings were honored in Windows 9x, although the files had begun to be phased out in favor of the Windows registry. Windows XP still acknowledges some entries in the **SYSTEM.INI** file, to provide backwards compatibility with older 16-bit applications.
- This is a configuration file that tells the machine what system aspects to load. I highly recommend not touching this tab.
- This file is used to initialize Windows system files. Such as, the fonts, keyboard, language and various other settings. It is located in the Windows directory.

SYSTEM.INI file is located in the C:\SYSTEMS directory. Here is the content of SYSTEM.INI on my Windows XP system.

WIN.INI Tab Of msconfig :

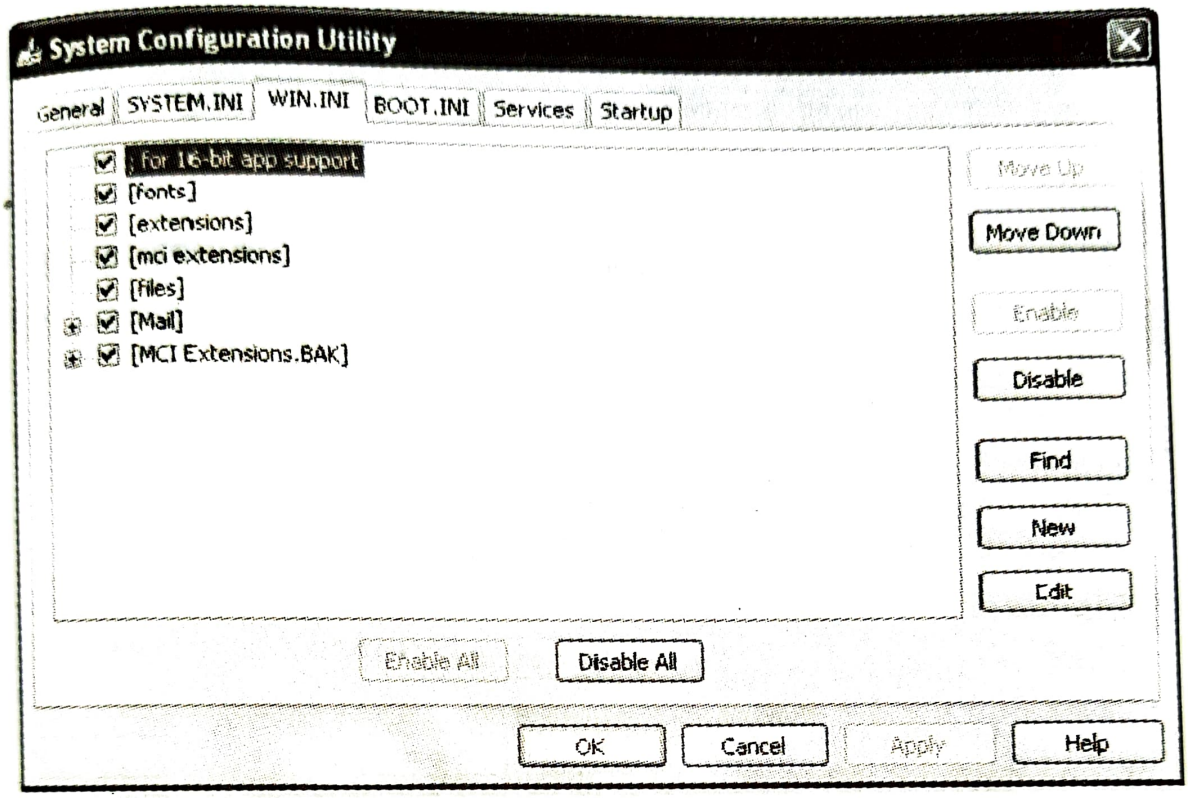

Figure 3.38 WIN.INI Tab of msconfig

- Once again this is a configuration file that tells the machine what system aspects to load. I highly recommend not touching this tab.
- This file is located in the Windows directory, and is used to load various settings when Windows boots. Such as, communications drivers, wallpaper, screen saver, languages, fonts, etc... These are all loaded when the file is initialized. However, if this file becomes corrupt, Windows may not load or if it does, it will have many errors.

AUTOEXEC.BAT :

AUTOEXEC.BAT is a file containing Disk Operating System commands that are executable when the computer is booted (started). The commands in AUTOEXEC.BAT tell the operating system which application programs are to be automatically started, how memory is to be managed, and initialize other settings. Each command in AUTOEXEC.BAT could be typed in manually after the computer is started, but that would take too long. The AUTOEXEC.BAT file is, in fact, a command script that is written beforehand so that it can be automatically executed when the operating system is started. The BAT suffix stands for batch, indicating that this is a file containing a sequence of commands entered from a file rather than interactively by a user.

Usage :

AUTOEXEC.BAT is read upon startup by all versions of DOS, including MS-DOS version 7.x as used in Windows 95 and Windows 98. Windows Me only parses environment variables as part of its attempts to reduce legacy dependencies, but this can be worked around.

Under DOS, the file is executed once the operating system has booted and after the CONFIG.SYS file has been processed. Windows NT and its descendants Windows XP and Windows Vista parse AUTOEXEC.BAT when a user logs on. As with Windows Me, anything other than setting environment variables is ignored. Unlike CONFIG.SYS, the commands in AUTOEXEC.BAT can be entered at the interactive command line interpreter. They are just standard commands that the computer operator wants to be executed automatically whenever the computer is started, and can include other batch files.

AUTOEXEC.BAT is most often used to set environment variables such as keyboard, soundcard, printer, and temporary file locations. It is also used to initiate low level system utilities, such as the following:

- Virus scanners
- Disk caching software - SMARTDRV.EXE from Microsoft the most common
- Mouse drivers
- Keyboard drivers
- CD drivers
- Miscellaneous other drivers

WMI Control :

The WMI (Windows Management Instrumentation) Control is a tool that enables you to configure WMI settings on a remote computer or local computer. Using the WMI Control, you can :

- Authorize users or groups and set permission levels
- Configure error logging

- Back up the repository
- Change the default namespace for scripting
- Connect as a different user

Authorize users or groups and set permission levels :

You can authorize a user or a group to access and perform WMI tasks and services. For each user or group you authorize, you set their permission level for specific namespaces. For example, you can enable a group to manage WMI's Common Information Model (CIM) objects on their local computers.

Configure error logging :

You can turn error logging on or off and, if turned on, set it to report errors only (the default) or all actions (verbose). Error logging can help you troubleshoot WMI problems. You can also define a maximum size for log files and their folder location.

Back up the repository :

You can configure the WMI control to back up your repository on a regularly-scheduled basis, or you can do it manually at any time. The repository is the database of objects that can be accessed through WMI. You can also restore a previous version of the repository.

Change the default namespace for scripting :

You can change the default namespace that is targeted in WMI scripts.

Connect as a different user :

You can log on under a different user name to change WMI Control settings. For example, if you have defined an administrative user account on several workstations, you can connect to those workstations under that user name.

Open WMI :

- To open the WMI Control console, click Start, and then click Run. In the Open box, type wmimgmt.msc, and then click OK.

Device Manager :

Device Manager is an extension of the Microsoft Management Console that provides a central and organized view of all the Microsoft Windows recognized hardware installed in a computer.

Device Manager provides you with a graphical view of the hardware that is installed on your computer. You can use Device Manager to update the drivers (or software) for hardware devices, modify hardware settings, and troubleshoot problems.

Use of Device Manager :

Device Manager is used to manage the hardware devices installed in a computer like hard disk drives, keyboards, sound cards, USB devices, and more.

Device Manager can be used for changing hardware configuration options, managing drivers, disabling and enabling hardware, identifying conflicts between hardware devices, and much more.

Think of Device Manager as the master list of hardware that Windows understands. All the hardware on your computer can be configured from this centralized utility.

You can use Device Manager to :

- Determine whether the hardware on your computer is working properly.
- Change hardware configuration settings.
- Identify the device drivers that are loaded for each device, and obtain information about each device driver.
- Change advanced settings and properties for devices.
- Install updated device drivers.
- Disable, enable, and uninstall devices.
- Roll back to the previous version of a driver.
- Print a summary of the devices that are installed on your computer.

You will typically use Device Manager to check the status of

your hardware and update device drivers on your computer. Advanced users who have a thorough understanding of computer hardware might also use Device Manager's diagnostic features to resolve device conflicts and change resource settings.

To get to the device manager in Windows :

1. Right click on the My Computer icon, choose properties, and then click on the Hardware tab and then click on the Device Manager tab.
2. You can select a variety of management options.

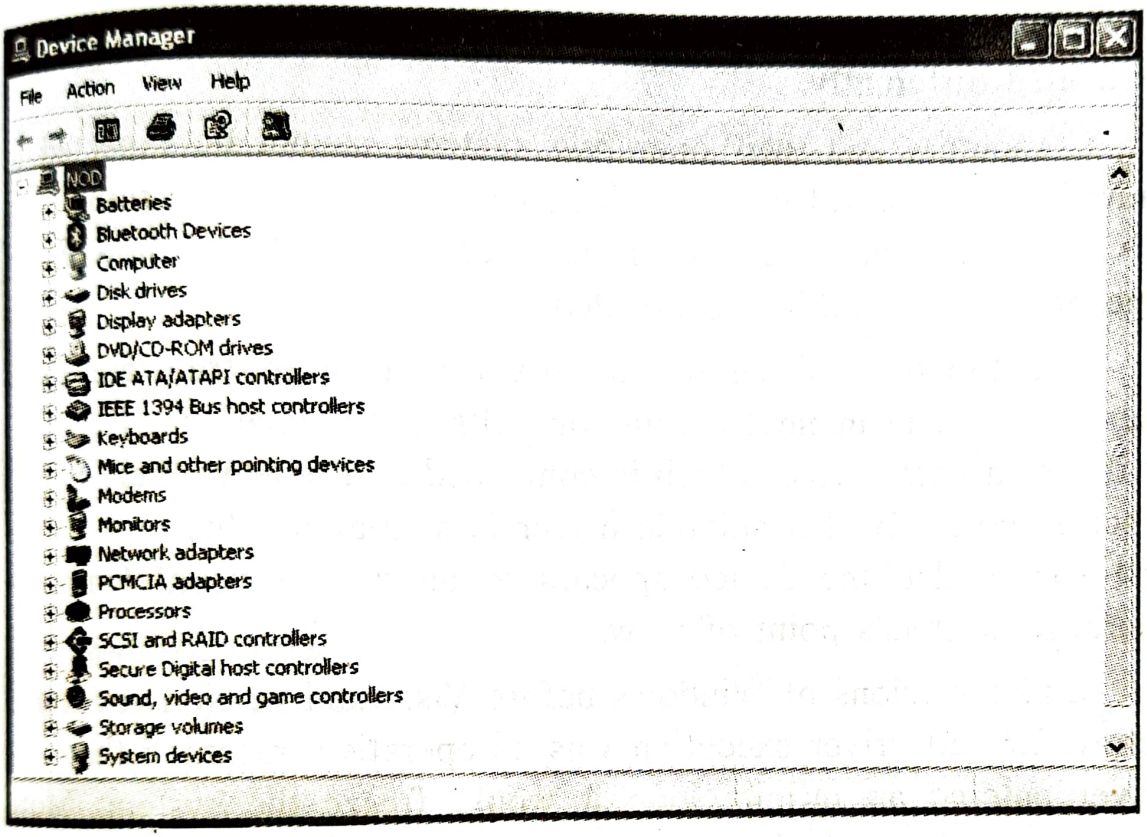

Figure 3.12 Device Manager

Device driver :

A device driver is a specific type of computer software developed to allow interaction with hardware devices.

Typically this constitutes an interface for communicating with the device, through the specific computer bus or communications subsystem that the hardware is connected to, providing commands to and/or receiving data from the device, and on the other end, the requisite interfaces to the operating system and software applications.

It is a specialized hardware-dependent computer program which is also operating system specific that enables another program, typically an operating system or applications software package or computer program running under the operating system kernel, to interact transparently with a hardware device, and usually provides the requisite interrupt handling necessary for any necessary asynchronous time-dependent hardware interfacing needs.

The key design goal of device drivers is abstraction. Every model of hardware (even within the same class of device) is different. Newer models also are released by manufacturers that provide more reliable or better performance and these newer models are often controlled differently.

Computers and their operating systems cannot be expected to know how to control every device, both now and in the future. To solve this problem, operative systems essentially dictate how every type of device should be controlled.

The function of the device driver is then to translate these operative system mandated function calls into device specific calls. In theory a new device, which is controlled in a new manner, should function correctly if a suitable driver is available. This new driver will ensure that the device appears to operate as usual from the operating system's point of view.

Under versions of Windows before Vista and versions of Linux before 2.6, all driver execution was co-operative, meaning that if a driver entered an infinite loop it would freeze the system. More recent revisions of these operating systems incorporate kernel preemption, where the kernel interrupts the driver to give it tasks, and then separates itself from the process until it receives a response from the device driver, or gives it more tasks to do.