ignou
THE PEOPLE'S
UNIVERSITY

Indira Gandhi National Open University
School of Vocational Education and Training

MSEI-021
INTRODUCTION TO INFORMATION SECURITY

Operating System Concepts

4

"शिक्षा मानव को बन्धनों से मुक्त करती है और आज के युग में तो यह लोकतंत्र की भावना का आधार भी है। जन्म तथा अन्य कारणों से उत्पन्न जाति एवं वर्गगत विषमताओं को दूर करते हुए मनुष्य को इन सबसे ऊपर उठाती है।"

— इन्दिरा गांधी

"Education is a liberating force, and in our age it is also a democratising force, cutting across the barries of caste and class, smoothing out inequalities imposed by birth and other circumstances."

— Indira Gandhi

ignou
THE PEOPLE'S
UNIVERSITY
Indira Gandhi National Open University
School of Vocational Education and Training

MSEI-021
Introduction to Information Security

Block

4

OPERATING SYSTEM CONCEPTS

Programme Expert/Design Committee of Post Graduate Diploma in Information Security (PGDIS)

Prof. K.R. Srivathsan
Pro Vice-Chancellor, IGNOU

Mr. B.J. Srinath, Sr. Director & Scientist 'G',CERT-In, Department of Information Technology, Ministry of Communication and Information Technology, Govt of India

Mr. A.S.A Krishnan, Director, Department of Information Technology, Cyber-Laws and E-Security Group, Ministry of Communication and Information Technology, Govt of India

Mr. S. Balasubramony, Dy. Superintendent of Police, CBI, Cyber Crime Investigation Cell, Delhi

Mr. B.V.C. Rao, Technical Director, National Informatics Centre, Ministry of Communication and Information Technology

Prof. M.N. Doja, Professor, Department of Computer Engineering, Jamia Milia Islamia, New Delhi
Dr. D.K. Lobiyal, Associate Professor, School of Computer and Systems Sciences, JNU New Delhi

Mr. Omveer Singh, Scientist, CERT-In Department of Information Technology, Cyber-Laws and E-Security Group, Ministry of Communication and Information Technology Govt of India

Dr. Vivek Mudgil, Director, Eninov Systems Noida

Mr. V.V.Subrahmanyam, Assistant Professor School of Computer and Information Science IGNOU

Mr. Anup Girdhar, CEO, Sedulity Solutions & Technologies, New Delhi

Prof. A.K. Saini, Professor, University School of Management Studies, Guru Gobind Singh Indraprastha University, Delhi

Mr. C.S. Rao, Technical Director in Cyber Security Division, National Informatics Centre, Ministry of Communication and Information Technology

Prof. C.G. Naidu, Director, School of Vocational Education & Training, IGNOU

Prof. Manohar Lal, Director, School of Computer and Information Science, IGNOU

Prof. K. Subramanian, Director, ACIIL, IGNOU
Former Deputy Director General, National Informatics Centre, Ministry of Communication and Information Technology, Govt of India

Prof. K. Elumalai, Director, School of Law, IGNOU

Dr. A. Murali M Rao, Joint Director, Computer Division, IGNOU

Mr. P.V. Suresh, Sr. Assistant Professor, School of Computer and Information Science IGNOU

Ms. Mansi Sharma, Assistant Professor School of Law, IGNOU

Ms. Urshla Kant
Assistant Professor, School of Vocational Education & Training, IGNOU
Programme Coordinator

Block Preparation

Unit Writers

Ms. Urshla Kant
Assistant Professor, School of Vocational Education & Training, IGNOU
(Unit 1, 2, 3 & 4)

Dr. K.Kiran Kumar
Reader, Department of Computer Science
P.G. Center, P.B.S. College
Vijayawada
(Unit 4)

Block Editor

Prof. K.R. Srivathsan,
Pro Vice-Chancellor, IGNOU

Proof Reading

Ms. Urshla Kant
Assistant Professor
School of Vocational Education & Training
IGNOU

Production

Mr. B. Natrajan	Mr. Jitender Sethi	Mr. Hemant Parida
Dy. Registrar (Pub.)	Asstt. Registrar (Pub.)	Proof Reader
MPDD, IGNOU, New Delhi	MPDD, IGNOU, New Delhi	MPDD, IGNOU, New Delhi

August, 2011

ISBN: 978–81-266-5568-7

Further information about the School of Vocational Education and Training and the Indira Gandhi National Open University courses may be obtained from the University's office at Maidan Garhi, New Delhi-110068. or the website of IGNOU www.ignou.ac.in

Printed and published on behalf of the Indira Gandhi National Open University, New Delhi, by the Registrar, MPDD

Laser typeset by Mctronics Printographics, 27/3 Ward No. 1, Opp. Mother Dairy, Mehrauli, New Delhi-30

Printed by : Hi-Tech Graphics, S-39, Okhla Industrial Area, Phase-II, New Delhi-110020

BLOCK INTRODUCTION

Operating System Concepts refers to the process for managing the computer's hardware and software resources. Basically, the operating system serves as the boss or manager and makes sure all the various parts of the computer get what they need. . Such operating systems monitor different programs and users, making sure everything runs smoothly, without interference, despite the fact that numerous devices and programs are used simultaneously. An operating system also has a vital role to play in security. Its job includes preventing unauthorized users from accessing the computer system.

This block introduces many of the methods to secure the operating system such as authentication, system updates, firewalls etc. This block comprises of four units and is designed in the following way;

The **Unit one** introduces operating system. An operating system is software, consisting of programs and data, that runs on computers, manages computer hardware resources, and provides common services for execution of various application software. Operating system is the most important type of system software in a computer system. Without an operating system, a user cannot run an application program on their computer, unless the application program is self booting.

The **Unit two** covers the detailed descriptions of the operating system security. There are various ways for providing security to operating system which is very essential for the proper working of the computer to perform tasks. This unit helps in understanding the possible mechanism for the operating system to function effectively, safely and efficiently.

The **Unit three** explains operating system hardening and controls. It is very important to see over the controls in place for the proper working of operating system. This unit helps in understanding of the controls and hardening process needed for the securing of operating system. It emphasizes on the importance of such mechanism which helps in controlling the operating system to function their tasks properly and effectively.

The **Unit four** covers the detailed descriptions of the Active Directory Controller and SAMBA. An Active Directory structure is a hierarchical framework of objects. The objects fall into two broad categories: resources (e.g. printers) and security principals (user or computer accounts and groups). Security principals are Active Directory objects that are assigned unique security identifiers (SIDs) used to control access and set security. SAMBA is a free software re-implementation, originally developed by Andrew Tridgell, of the SMB/CIFS networking protocol. As of version 3, SAMBA provides file and print services for various Microsoft Windows clients and can integrate with a Windows Server domain, either as a Primary Domain Controller (PDC) or as a domain member. It can also be part of an Active Directory domain.

Hope you benefit from this block.

ACKNOWLEDGEMENT

The material we have used is purely for educational purposes. Every effort has been made to trace the copyright holders of material reproduced in this book. Should any infringement have occurred, the publishers and editors apologize and will be pleased to make the necessary corrections in future editions of this book.

UNIT 1 INTRODUCTION TO OPERATING SYSTEM

Structure

1.0 INTRODUCTION

An operating system (sometimes abbreviated as "OS") is the program that, after being initially loaded into the computer by a boot program, manages all the other programs in a computer. The other programs are called applications or application programs. The application programs make use of the operating system by making requests for services through a defined application program interface (API). In addition, users can interact directly with the operating system through a user interface such as a command language or a graphical user interface (GUI).

1.1 OBJECTIVES

After studying this unit, you should be able to:

- explain operating system;
- explain application of operating system; and
- define the types of operating system.

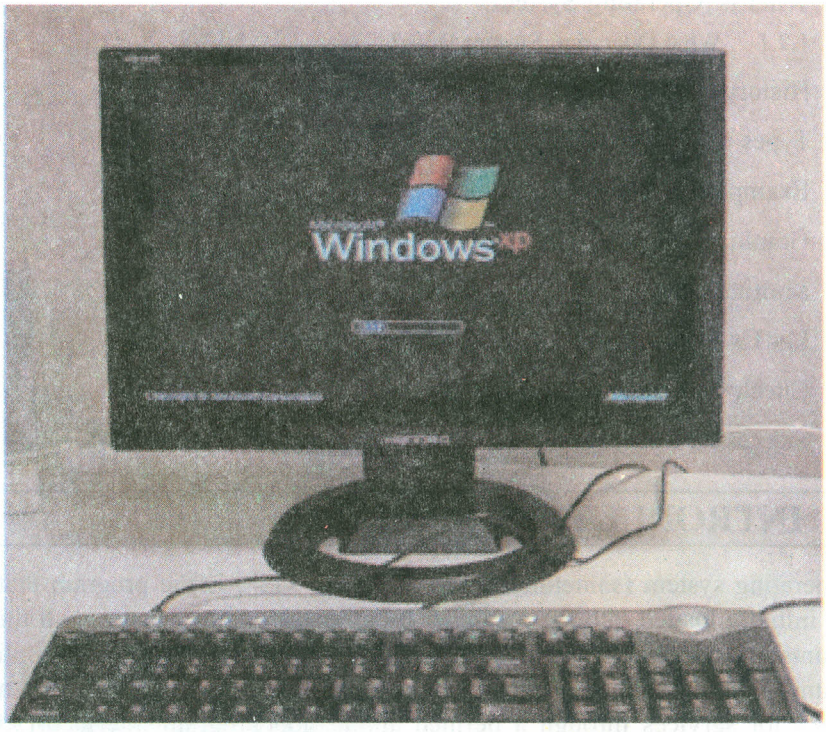
1.2 WHAT IS OPERATING SYSTEM?

An operating system is a program designed to run other programs on a computer. A computer's operating system is its most important program. It is considered the backbone of a computer, managing both software and hardware resources. Operating systems are responsible for everything from the control and allocation of memory to recognizing input from external devices and transmitting output to computer displays. They also manage files on computer shard drives and control peripherals, like printers and scanners.

The operating system of a large computer system has even more work to do. Such operating systems monitor different programs and users, making sure everything runs smoothly, without interference, despite the fact that numerous devices and

programs are used simultaneously. An operating system also has a vital role to play in security. Its job includes preventing unauthorized users from accessing the computer system.

There are multiuser, multiprocessing, multitasking, multithreading and real-time operating systems. A multiuser operating system enables multiple users to run programs simultaneously. This type of operating system may be used for just a few people or hundreds of them. In fact, there are some operating systems that are used to allow thousands of people to run programs at the same time.



A multiprocessing operating system allows a program to run on more than one central processing unit (CPU) at a time. This can come in very handy in some work environments, at schools and even for some home-computing situations. Multitasking operating systems work a little differently; they make it possible to run more than one program at a time. Multithreading operating systems are even more different, allowing varied parts of one program to be used simultaneously.

Real-time operating systems are designed to allow computers to process and respond to input instantly. Usually, general-purpose operating systems, such as disk operating system (DOS), are not considered real time, as they may require seconds or minutes to respond to input. Real-time operating systems are typically used when computers must react to the consistent input of information without delay. For example, real-time operating systems may be used in navigation.

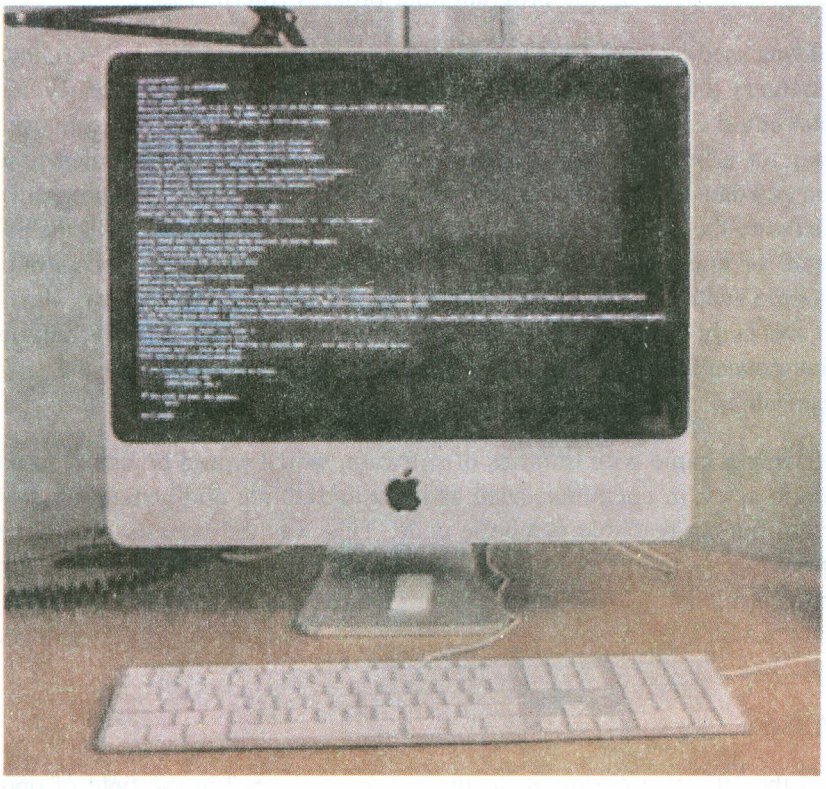
Today's operating systems tend to have graphical user interfaces (GUIs) that employ pointing devices for input. A mouse is an example of such a pointing device, as is a stylus. Commonly used operating systems for IBM-compatible personal computers include Microsoft Windows, Linux and UNIX variations. For Macintosh computers, Mac OS X, Linux, BSD and some Windows variants are commonly used.

1.2.1 What Operating System does?

Booting is the process that occurs when you press the power button to turn your computer on. At the end of that process, the operating system loads. From this point, the operating system begins to do its job of controlling the way in which the computer functions. The operating system is responsible for managing the

computer's hardware and software resources. Basically, the operating system serves as the boss or manager and makes sure all the various parts of the computer get what they need.

When you use your personal computer, you may work on a Word document, print an e-mail and have your Internet browser open for web surfing, all at the same time. These three programs need attention from the central processing unit (CPU) to do whatever task that you, the user, are telling it to do. These programs need memory and storage and need to be able to send messages to devices such as the mouse and the printer to accomplish these tasks. The operating system is responsible for handling these areas, as well as processor and network management.



An operating system performs these services for applications:

- In a multitasking operating system where multiple programs can be running at the same time, the operating system determines which applications should run in what order and how much time should be allowed for each application before giving another application a turn.
- It manages the sharing of internal memory among multiple applications.
- It handles input and output to and from attached hardware devices, such as hard disks, printers and dial-up ports.
- It sends messages to each application or interactive user (or to a system operator) about the status of operation and any errors that may have occurred.
- It can offload the management of what are called batch jobs (for example, printing) so that the initiating application is freed from this work.
- On computers that can provide parallel processing, an operating system can manage how to divide the program so that it runs on more than one processor at a time.

  All major computer platforms (hardware and software) require and sometimes include an operating system. Linux, Windows 2000, VMS, OS/400, AIX and z/OS are all examples of operating systems.

1.3 HISTORY OF OPERATING SYSTEM

In the 1940s, the earliest electronic digital systems had no operating systems. Electronic systems of this time were so primitive compared to those of today that instructions were often entered into the system one bit at a time on rows of mechanical switches or by jumper wires on plug boards. These were special-purpose systems that, for example, generated ballistics tables for the military or controlled the printing of payroll checks from data on punched paper cards. After programmable general purpose computers were invented, machine languages (consisting of strings of the binary digits 0 and 1 on punched paper tape) were introduced that speed up the programming process (Stern, 1981).

OS/360 was used on most IBM mainframe computers beginning in 1966, including the computers that helped NASA put a man on the moon. In the early 1950s, a computer could execute only one program at a time. Each user had sole use of the computer for a limited period of time and would arrive at a scheduled time with program and data on punched paper cards and/or punched tape. The program would be loaded into the machine and the machine would be set to work until the program completed or crashed. Programs could generally be debugged via a front panel using toggle switches and panel lights. It is said that Alan Turing was a master of this on the early Manchester Mark 1 machine and he was already deriving the primitive conception of an operating system from the principles of the Universal Turing machine.

Later machines came with libraries of software, which would be linked to a user's program to assist in operations such as input and output and generating computer code from human-readable symbolic code. This was the genesis of the modern-day operating system. However, machines still ran a single job at a time. At Cambridge University in England the job queue was at one time a washing line from which tapes were hung with different colored clothes-pegs to indicate job-priority.

Mainframes

Through the 1950s, many major features were pioneered in the field of operating systems, including batch processing, input/output interrupt, buffering, multitasking, spooling, runtime libraries, link-loading and programs for sorting records in files. These features were included or not included in application software at the option of application programmers, rather than in a separate operating system used by all applications. In 1959 the SHARE Operating System was released as an integrated utility for the IBM 704 and later in the 709 and 7090 mainframes.

During the 1960s, IBM's OS/360 introduced the concept of a single OS spanning an entire product line, which was crucial for the success of the System/360 machines. IBM's current mainframe operating systems are distant descendants of this original system and applications written for OS/360 can still be run on modern machines. In the mid 70's, MVS, a descendant of OS/360, offered the first implementation of using RAM as a transparent cache for data.

OS/360 also pioneered the concept that the operating system keeps track of all of the system resources that are used, including program and data space allocation in main memory and file space in secondary storage and file locking during update. When the process is terminated for any reason, all of these resources are re-claimed by the operating system.

The alternative CP-67 system for the S/360-67 started a whole line of IBM operating systems focused on the concept of virtual machines. Other operating systems used on IBM S/360 series mainframes included systems developed by IBM: COS/360 (Compatabililty Operating System), DOS/360 (Disk Operating System), TSS/360

(Time Sharing System), TOS/360 (Tape Operating System), BOS/360 (Basic Operating System) and ACP (Airline Control Program), as well as a few non-IBM systems: MTS (Michigan Terminal System), MUSIC (Multi-User System for Interactive Computing) and ORVYL (Stanford Timesharing System).

Control Data Corporation developed the SCOPE operating system in the 1960s, for batch processing. In cooperation with the University of Minnesota, the KRONOS and later the NOS operating systems were developed during the 1970s, which supported simultaneous batch and timesharing use. Like many commercial timesharing systems, its interface was an extension of the Dartmouth BASIC operating systems, one of the pioneering efforts in timesharing and programming languages. In the late 1970s, Control Data and the University of Illinois developed the PLATO operating system, which used plasma panel displays and long-distance time sharing networks. Plato was remarkably innovative for its time, featuring real-time chat and multi-user graphical games. Burroughs Corporation introduced the B5000 in 1961 with the MCP, (Master Control Program) operating system. The B5000 was a stack machine designed to exclusively support high-level languages with no machine language or assembler and indeed the MCP was the first OS to be written exclusively in a high-level language ESPOL, a dialect of ALGOL.

MCP also introduced many other ground-breaking innovations, such as being the first commercial implementation of virtual memory. During development of the AS400, IBM made an approach to Burroughs to licence MCP to run on the AS400 hardware. This proposal was declined by Burroughs management to protect its existing hardware production. MCP is still in use today in the Unisys ClearPath/ MCP line of computers.

UNIVAC, the first commercial computer manufacturer, produced a series of EXEC operating systems. Like all early main-frame systems, this was a batch-oriented system that managed magnetic drums, disks, card readers and line printers. In the 1970s, UNIVAC produced the Real-Time Basic (RTB) system to support large-scale time sharing, also patterned after the Dartmouth BC system.

General Electric and MIT developed General Electric Comprehensive Operating Supervisor (GECOS), which introduced the concept of ringed security privilege levels. After acquisition by Honeywell it was renamed to General Comprehensive Operating System (GCOS). Digital Equipment Corporation developed many operating systems for its various computer lines, including TOPS-10 and TOPS-20 time sharing systems for the 36-bit PDP-10 class systems. Prior to the widespread use of UNIX, TOPS-10 was a particularly popular system in universities and in the early ARPANET community.

In the late 1960s through the late 1970s, several hardware capabilities evolved that allowed similar or ported software to run on more than one system. Early systems had utilized microprogramming to implement features on their systems in order to permit different underlying architecture to appear to be the same as others in a series. In fact most 360s after the 360/40 (except the 360/165 and 360/168) were microprogrammed implementations. But soon other means of achieving application compatibility were proven to be more significant.

The enormous investment in software for these systems made since 1960s caused most of the original computer manufacturers to continue to develop compatible operating systems along with the hardware. The notable supported mainframe operating systems include:

- **Burroughs MCP** – B5000, 1961 to Unisys Clearpath/MCP, present.
- **IBM OS/360** – IBM System/360, 1966 to IBM z/OS, present.

- **IBM CP-67** – IBM System/360, 1967 to IBM z/VM, present.
- **UNIVAC EXEC 8** – UNIVAC 1108, 1967, to OS 2200 Unisys Clearpath Dorado, present.

Microcomputers

PC-DOS was an early personal computer OS that featured a command line interface. Mac OS by Apple Computer became the first widespread OS to feature a graphical user interface. Many of its features such as windows and icons would later become commonplace in GUIs.

The first microcomputers did not have the capacity or need for the elaborate operating systems that had been developed for mainframes and minis; minimalistic operating systems were developed, often loaded from ROM and known as Monitors. One notable early disk-based operating system was CP/M, which was supported on many early microcomputers and was closely imitated in MS-DOS, which became wildly popular as the operating system chosen for the IBM PC (IBM's version of it was called IBM DOS or PC DOS), its successors making Microsoft. In the 80's Apple Computer Inc. (now Apple Inc.) abandoned its popular Apple II series of microcomputers to introduce the Apple Macintosh computer with an innovative Graphical User Interface (GUI) to the Mac OS.

The introduction of the Intel 80386 CPU chip with 32-bit architecture and paging capabilities, provided personal computers with the ability to run multitasking operating systems like those of earlier minicomputers and mainframes. Microsoft responded to this progress by hiring Dave Cutler, who had developed the VMS operating system for Digital Equipment Corporation. He would lead the development of the Windows NT operating system, which continues to serve as the basis for Microsoft's operating systems line. Steve Jobs, a co-founder of Apple Inc., started NeXT Computer Inc., which developed the UNIX-like NEXTSTEP operating system. NEXTSTEP would later be acquired by Apple Inc. and used, along with code from FreeBSD as the core of Mac OS X.

The GNU project was started by activist and programmer Richard Stallman with the goal of a complete free software replacement to the proprietary UNIX operating system. While the project was highly successful in duplicating the functionality of various parts of UNIX, development of the GNU Hurd kernel proved to be unproductive. In 1991, Finnish computer science student Linus Torvalds, with cooperation from volunteers collaborating over the Internet, released the first version of the Linux kernel. It was soon merged with the GNU user space components and system software to form a complete operating system. Since then, the combination of the two major components has usually been referred to as simply "Linux" by the software industry, a naming convention that Stallman and the Free Software Foundation remain opposed to, preferring the name GNU/Linux. The Berkeley Software Distribution, known as BSD, is the UNIX derivative distributed by the University of California, Berkeley, starting in the 1970s. Freely distributed and ported to many minicomputers, it eventually also gained a following for use on PCs, mainly as FreeBSD, NetBSD and OpenBSD.

1.4 TYPES OF OPERATING SYSTEM

1) **Real-time**

A real-time operating system is a multitasking operating system that aims at executing real-time applications. Real-time operating systems often use specialized scheduling algorithms so that they can achieve a deterministic nature of behavior. The main object of real-time operating systems is their quick and predictable response to events. They have an event-driven or time-sharing design and often

aspects of both. An event-driven system switches between tasks based on their priorities or external events while time-sharing operating systems switch tasks based on clock interrupts.

2) **Multi-user vs. Single-user**

A multi-user operating system allows multiple users to access a computer system concurrently. Time-sharing system can be classified as multi-user systems as they enable a multiple user access to a computer through the sharing of time. Single-user operating systems, as opposed to a multi-user operating system, are usable by a single user at a time. Being able to have multiple accounts on a Windows operating system does not make it a multi-user system. Rather, only the network administrator is the real user. But for a UNIX-like operating system, it is possible for two users to login at a time and this capability of the OS makes it a multi-user operating system.

3) **Multi-tasking vs. Single-tasking**

When a single program is allowed to run at a time, the system is grouped under a single-tasking system, while in case the operating system allows the execution of multiple tasks at one time, it is classified as a multi-tasking operating system. Multi-tasking can be of two types namely, pre-emptive or co-operative. In pre-emptive multitasking, the operating system slices the CPU time and dedicates one slot to each of the programs. UNIX-like operating systems such as Solaris and Linux support pre-emptive multitasking. Cooperative multitasking is achieved by relying on each process to give time to the other processes in a defined manner. MS Windows prior to Windows 95 used to support cooperative multitasking.

4) **Distributed**

A distributed operating system manages a group of independent computers and makes them appear to be a single computer. The development of networked computers that could be linked and communicate with each other, gave rise to distributed computing. Distributed computations are carried out on more than one machine. When computers in a group work in cooperation, they make a distributed system.

5) **Embedded**

Embedded operating systems are designed to be used in embedded computer systems. They are designed to operate on small machines like PDAs with less autonomy. They are able to operate with a limited number of resources. They are very compact and extremely efficient by design. Windows CE and Minix 3 are some examples of embedded operating systems.

Check Your Progress 1

Note: a) Space is given below for writing your answers.

b) Compare your answers with the one given at the end of the Unit.

1) Explain operating system.

.....

.....

.....

.....

.....

2) What is the use of operating system?

.....

.....

.....

.....

.....

3) Provide the types of operating system.

.....

.....

.....

.....

.....

1.5 EXAMPLES OF OPERATING SYSTEM

1) **UNIX and UNIX-like operating systems**

Ken Thompson wrote B, mainly based on BCPL, which he used to write UNIX, based on his experience in the MULTICS project. B was replaced by C and UNIX developed into a large, complex family of inter-related operating systems which have been influential in every modern operating system (see History). The UNIX-like family is a diverse group of operating systems, with several major sub-categories including System V, BSD and GNU/Linux. The name "UNIX" is a trademark of The Open Group which licenses it for use with any operating system that has been shown to conform to their definitions. "UNIX-like" is commonly used to refer to the large set of operating systems which resemble the original UNIX.

UNIX-like systems run on a wide variety of machine architectures. They are used heavily for servers in business, as well as workstations in academic and engineering environments. Free UNIX variants, such as GNU/Linux and BSD, are popular in these areas.

Some UNIX variants like HP's HP-UX and IBM's AIX are designed to run only on that vendor's hardware. Others, such as Solaris, can run on multiple types of hardware, including x86 servers and PCs. Apple's Mac OS X, a hybrid kernel-based BSD variant derived from NeXTSTEP, Mach and FreeBSD, has replaced Apple's earlier (non-UNIX) Mac OS.

UNIX interoperability was sought by establishing the POSIX standard. The POSIX standard can be applied to any operating system, although it was originally created for various UNIX variants.

2) **BSD and its descendants**

The first server for the World Wide Web ran on NeXTSTEP, based on BSD. A subgroup of the UNIX family is the Berkeley Software Distribution family, which includes FreeBSD, NetBSD and OpenBSD. These operating systems are most commonly found on webservers, although they can also function as a personal computer OS. The Internet owes much of its existence to BSD, as many of the protocols now commonly used by computers to connect, send and receive data

over a network were widely implemented and refined in BSD. The World Wide Web was also first demonstrated on a number of computers running an OS based on BSD called NextStep.

BSD has its roots in UNIX. In 1974, University of California, Berkeley installed its first UNIX system. Over time, students and staff in the computer science department there began adding new programs to make things easier, such as text editors. When Berkely received new VAX computers in 1978 with UNIX installed, the school's undergraduates modified UNIX even more in order to take advantage of the computer's hardware possibilities. The Defense Advanced Research Projects Agency of the US Department of Defense took interest and decided to fund the project. Many schools, corporations and government organizations took notice and started to use Berkeley's version of UNIX instead of the official one distributed by AT&T.

Steve Jobs, upon leaving Apple Inc. in 1985, formed NeXT Inc., a company that manufactured high-end computers running on a variation of BSD called NeXTSTEP. One of these computers was used by Tim Berners-Lee as the first webserver to create the World Wide Web.

Developers like Keith Bostic encouraged the project to replace any non-free code that originated with Bell Labs. Once this was done, however, AT&T sued. Eventually, after two years of legal disputes, the BSD project came out ahead and spawned a number of free derivatives, such as FreeBSD and NetBSD. In this two year wait, GNU and Linux appeared.

3) **Plan 9**

Ken Thompson, Dennis Ritchie and Douglas McIlroy at Bell Labs designed and developed the C programming language to build the operating system UNIX. Programmers at Bell Labs went on to develop Plan 9 and Inferno, which were engineered for modern distributed environments. Plan 9 was designed from the start to be a networked operating system and had graphics built-in, unlike UNIX, which added these features to the design later. It is currently released under the Lucent Public License. Inferno was sold to Vita Nuova Holdings and has been released under a GPL/MIT license.

4) **Linux and GNU**

Linux (or GNU/Linux) is a UNIX-like operating system that can be used on a wide range of devices from supercomputers to wristwatches. The Linux kernel is released under an open source license, so anyone can read and modify its code. It has been modified to run on a large variety of electronics. Although estimates suggest that Linux is used on 1.82 % of all personal computers, it has been widely adopted for use in servers and embedded systems (such as cell phones). Linux has superseded UNIX in most places and is used on the 10 most powerful supercomputers in the world. The Linux kernel is used in some popular distributions, such as Red Hat, Debian, Ubuntu, Linux Mint and Google's Android.

The GNU project is a mass collaboration of programmers who seek to create a completely free and open operating system that was similar to UNIX but with completely original code. It was started in 1983 by Richard Stallman and is responsible for many of the parts of most Linux variants. For this reason, some claim that the combined product of the Linux kernel and the GNU software collection is more correctly called GNU/Linux. Thousands of pieces of software for virtually every operating system are licensed under the GNU General Public License. Meanwhile, the Linux kernel began as a side project of Linus Torvalds, a university student from Finland. In 1991, Torvalds began work on it and posted information about his project on a newsgroup for computer students and programmers. He received a wave of support and volunteers who ended up creating

a full-fledged kernel. Programmers from GNU took notice and members of both projects worked to integrate the finished GNU parts with the Linux kernel in order to create a full-fledged operating system.

5) **Google Chrome OS**

Chrome is an operating system based on the Linux kernel and designed by Google. Chrome targets computer users who spend most of their time on the Internet, it is technically only a web browser with no other applications and relies on Internet applications (or Web apps) used in the web browser to accomplish tasks such as word processing and media viewing.

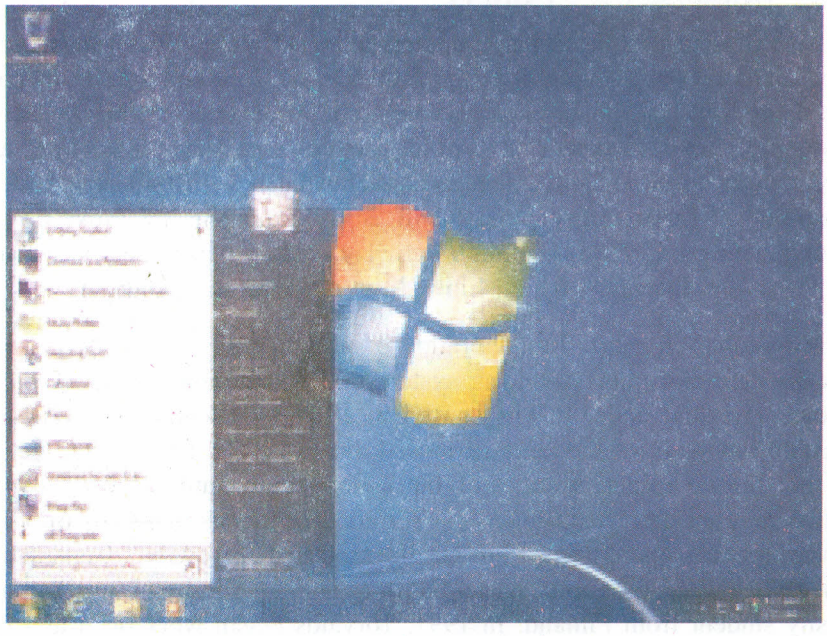
6) **Mac OS X**

Mac OS X is a line of partially proprietary graphical operating systems developed, marketed and sold by Apple Inc., the latest of which is pre-loaded on all currently shipping Macintosh computers. Mac OS X is the successor to the original Mac OS, which had been Apple's primary operating system since 1984. Unlike its predecessor, Mac OS X is a UNIX operating system built on technology that had been developed at NeXT through the second half of the 1980s and up until Apple purchased the company in early 1997.

The operating system was first released in 1999 as Mac OS X Server 1.0, with a desktop-oriented version (Mac OS X v10.0) following in March 2001. Since then, six more distinct "client" and "server" editions of Mac OS X have been released, the most recent being Mac OS X v10.6, which was first made available on August 28, 2009. Releases of Mac OS X are named after big cats; the current version of Mac OS X is "Snow Leopard".

The server edition, Mac OS X Server, is architecturally identical to its desktop counterpart but usually runs on Apple's line of Macintosh server hardware. Mac OS X Server includes work group management and administration software tools that provide simplified access to key network services, including a mail transfer agent, a Samba server, an LDAP server, a domain name server and others. In the upcoming release of Mac OS X v10.7 Lion, all server aspects of Mac OS X Server will be integrated into the client version.

7) **Microsoft Windows**



Microsoft Windows is a family of proprietary operating systems designed by Microsoft Corporation and primarily targeted to Intel architecture based computers.

with an estimated 88.9 percent total usage share on Web connected computers. Currently, the most widely used version of the Windows family is Windows XP, released on October 25, 2001. The newest version is Windows 7 for workstations and Windows Server 2008 R2 for servers.

Microsoft Windows originated in 1985 as an application running on top of MS-DOS, which was the standard operating system shipped on most Intel architecture personal computers at the time. In 1995, Windows 95 was released, combining MS-DOS 7.0 with Windows on the same medium, removing the need of getting a separate MS-DOS license. Keeping much legacy, it could run real-mode MS-DOS and 16 bits Windows 3.x drivers. Windows Me, released in 2000, was the latest version of Windows of the Windows 95 family. Later versions have all been based on the Windows NT kernel. Current versions of Windows run on IA-32 and x86-64 microprocessors, although Windows 8 will support ARM architecture. In the past, Windows NT supported a few non-Intel architectures.

Server editions of Windows are widely used. In recent years, Microsoft has expended significant capital in an effort to promote the use of Windows as a server operating environment. However, Windows' usage on servers is not as widespread as on personal computers, as Windows competes against Linux and BSD for market share.

8) **Other**

Older operating systems which are still used in niche markets include OS/2 from IBM and Microsoft; Mac OS, the non-UNIX precursor to Apple's Mac OS X; BeOS; XTS-300. Some, most notably Haiku, RISC OS, MorphOS, AmigaOS 4 and FreeMint continue to be developed as minority platforms for enthusiast communities and specialist applications. OpenVMS formerly from DEC is still under active development by Hewlett-Packard. Yet other operating systems are used almost exclusively in academia, for operating systems education or to do research on operating system concepts. A typical example of a system that fulfills both roles is MINIX, while for example Singularity is used purely for research.

1.6 GRAPHICAL USER INTERFACE

Most of the modern computer systems support graphical user interfaces (GUI) and often include them. In some computer systems, such as the original implementation of Mac OS, the GUI is integrated into the kernel.

While technically a graphical user interface is not an operating system service, incorporating support for one into the operating system kernel can allow the GUI to be more responsive by reducing the number of context switches required for the GUI to perform its output functions. Other operating systems are modular, separating the graphics subsystem from the kernel and the Operating System. In the 1980s UNIX, VMS and many others had operating systems that were built this way. GNU/Linux and Mac OS X are also built this way. Modern releases of Microsoft Windows such as Windows Vista implement a graphics subsystem that is mostly in user-space; however the graphics drawing routines of versions between Windows NT 4.0 and Windows Server 2003 exist mostly in kernel space. Windows 9x had very little distinction between the interface and the kernel.

Many computer operating systems allow the user to install or create any user interface they desire. The X Window System in conjunction with GNOME or KDE is a commonly found setup on most UNIX and UNIX-like (BSD, GNU/Linux, Solaris) systems. A number of Windows shell replacements have been released for Microsoft Windows, which offer alternatives to the included Windows shell, but the shell itself cannot be separated from Windows.

Numerous UNIX-based GUIs have existed over time, most derived from X11. Competition among the various vendors of UNIX (HP, IBM, Sun) led to much fragmentation, though an effort to standardize in the 1990s to COSE and CDE failed for various reasons and were eventually eclipsed by the widespread adoption of GNOME and KDE. Prior to free software-based toolkits and desktop environments, Motif was the prevalent toolkit/desktop combination (and was the basis upon which CDE was developed).

Graphical user interfaces evolve over time. For example, Windows has modified its user interface almost every time a new major version of Windows is released and the Mac OS GUI changed dramatically with the introduction of Mac OS X in 1999.

1.7 MULTITASKING

Multitasking refers to the running of multiple independent computer programs on the same computer; giving the appearance that it is performing the tasks at the same time. Since most computers can do at most one or two things at one time, this is generally done via time-sharing, which means that each program uses a share of the computer's time to execute.

An operating system kernel contains a piece of software called a scheduler which determines how much time each program will spend executing and in which order execution control should be passed to programs. Control is passed to a process by the kernel, which allows the program access to the CPU and memory. Later, control is returned to the kernel through some mechanism, so that another program may be allowed to use the CPU. This so-called passing of control between the kernel and applications is called a context switch.

An early model which governed the allocation of time to programs was called cooperative multitasking. In this model, when control is passed to a program by the kernel, it may execute for as long as it wants before explicitly returning control to the kernel. This means that a malicious or malfunctioning program may not only prevent any other programs from using the CPU, but it can hang the entire system if it enters an infinite loop.

Modern operating systems extend the concepts of application preemption to device drivers and kernel code, so that the operating system has preemptive control over internal run-times as well.

The philosophy governing preemptive multitasking is that of ensuring that all programs are given regular time on the CPU. This implies that all programs must be limited in how much time they are allowed to spend on the CPU without being interrupted. To accomplish this, modern operating system kernels make use of a timed interrupt. A protected mode timer is set by the kernel which triggers a return to supervisor mode after the specified time has elapsed. (See above sections on Interrupts and Dual Mode Operation.)

On many single user operating systems cooperative multitasking is perfectly adequate, as home computers generally run a small number of well tested programs. Windows NT was the first version of Microsoft Windows which enforced preemptive multitasking, but it didn't reach the home user market until Windows XP (since Windows NT was targeted at professionals).

Check Your Progress 2

Note: a) Space is given below for writing your answers.

b) Compare your answers with the one given at the end of the Unit.

1) Provide the examples of operating system.

.....

.....

.....

.....

2) Explain Graphical User Interface (GUI).

.....

.....

.....

.....

3) What is multitasking?

.....

.....

.....

.....

1.8 LET US SUM UP

An operating system is software, consisting of programs and data, that runs on computers, manages computer hardware resources and provides common services for execution of various application software. Operating system is the most important type of system software in a computer system. Without an operating system, a user cannot run an application program on their computer, unless the application program is self booting.

For hardware functions such as input and output and memory allocation, the operating system acts as an intermediary between application programs and the computer hardware, though the application code is usually executed directly by the hardware and will frequently call the OS or be interrupted by it. Operating systems are found on almost any device that contains a computer-from cellular phones and video game consoles to supercomputers and web servers.

Examples of popular modern operating systems are: BSD, Linux (Ubuntu, Fedora, OpenSuSE, Mandriva, Arch Linux, Debian, Linux mint etc.), Mac OS X, Microsoft Windows and UNIX.

1.9 CHECK YOUR PROGRESS: THE KEY

Check Your Progress 1

1) An operating system is a program designed to run other programs on a computer. A computer's operating system is its most important program. It is considered the backbone of a computer, managing both software and hardware

resources. Operating systems are responsible for everything from the control and allocation of memory to recognizing input from external devices and transmitting output to computer displays. They also manage files on computer hard drives and control peripherals, like printers and scanners.

2)
- In a multitasking operating system where multiple programs can be running at the same time, the operating system determines which applications should run in what order and how much time should be allowed for each application before giving another application a turn. It manages the sharing of internal memory among multiple applications.
- It handles input and output to and from attached hardware devices, such as hard disks, printers and dial-up ports.
- It sends messages to each application or interactive user (or to a system operator) about the status of operation and any errors that may have occurred.
- It can offload the management of what are called batch jobs (for example, printing) so that the initiating application is freed from this work.
- On computers that can provide parallel processing, an operating system can manage how to divide the program so that it runs on more than one processor at a time.

3) Real time, single user, Multi user, embedded, distributed.

Check Your Progress 2

1) Microsoft Windows, Apple Mac OS X and Linux.

2) Most of the modern computer systems support graphical user interfaces (GUI) and often include them. In some computer systems, such as the original implementation of Mac OS, the GUI is integrated into the kernel. While technically a graphical user interface is not an operating system service, incorporating support for one into the operating system kernel can allow the GUI to be more responsive by reducing the number of context switches required for the GUI to perform its output functions.

3) Multitasking refers to the running of multiple independent computer programs on the same computer; giving the appearance that it is performing the tasks at the same time. Since most computers can do at most one or two things at one time, this is generally done via time-sharing, which means that each program uses a share of the computer's time to execute.

1.10 SUGGESTED READINGS

- Operating System Concepts by Abraham Silberchatz.
- Operting System Concepts by Galvin.
- www.depik.com/php/osc.pdf.
- www.ittestpapers.com/operatingsystemconcepts-covers-fundamental-questions-in-os.html.

UNIT 2 OPERATING SYSTEM SECURITY: AN OVERVIEW

Structure

2.0 INTRODUCTION

Computers are becoming largely ubiquitous in today's society. Computers are used in every field of work and entertainment. In the process of modern-day computing, a great deal of personal information is accumulated, processed and exchanged. This has become especially true with the advent of the Internet. Whether sending an e-mail, making a purchase online or just surfing a web page, today's computer user is subject to many privacy and security concerns. So it is with these concerns in mind that it is necessary to have decided to improve the reliability and trustworthiness of the modern computing environment. Operating systems provide the fundamental mechanisms for securing computer processing. Recently, the importance of ensuring such security has become a mainstream issue for all operating systems.

2.1 OBJECTIVES

After studying this unit, you should be able to:

- explain security issues in operating system;
- explain precautions for operating system security; and
- define patch and firewalls.

2.2 OPERATING SYSTEM SECURITY

The goal of computer security is the protection of information stored on the computer system. It is essential for providing security to the virtual information.

Information security is aimed at the following:

1) **Integrity** – The value of all information depends upon its accuracy. If unauthorized changes are made to data, this data loses some or all of its value.

2) **Privacy** – The value of much information depends upon its secrecy.

3) **Availability** – Information must be readily available.

Before you begin using the system, it is helpful to plan and implement security policies. Security policies are very time-consuming to change later, so up-front planning can save a lot of time later.

Various flaws in the operating systems of computers are discovered almost daily. The majority of viruses take advantage of these flaws to infect your computer. Once a virus enters your system, it can potentially cause devastating damage.

2.3 PRECAUTIONS FOR OPERATING SYSTEM SECURITY

To avoid contracting a virus, you should take the following basic precautions:

2.3.1 Authentication

Authentication is any process by which you verify that someone is who they claim they are. This usually involves a username and a password, but can include any other method of demonstrating identity, such as a smart card, retina scan, voice recognition or fingerprints. Authentication is equivalent to showing your drivers license at the ticket counter at the airport. Authentication is the process of obtaining identification credentials such as name and password from a user and validating those credentials against some authority.

2.3.2 Access Control

Access control refers to security features that control who can access resources in the operating system. Applications call access control functions to set who can access specific resources or control access to resources provided by the application. Access control is a system which enables an authority to control access to areas and resources in a given physical facility or computer-based information system. An access control system, within the field of physical security, is generally seen as the second layer in the security of a physical structure.

Access control is, in reality, an everyday phenomenon. A lock on a car door is essentially a form of access control. A PIN on an ATM system at a bank is another means of access control. The possession of access control is of prime importance when persons seek to secure important, confidential or sensitive information and equipment.

Item control or electronic key management is an area within (and possibly integrated with) an access control system which concerns the managing of possession and location of small assets or physical (mechanical) keys.

2.3.3 Security Models

A scheme for specifying and enforcing security policies is a computer security model. A security model may be founded upon a formal model of access rights, a model of computation, a model of distributed computing or no particular theoretical grounding at all. Security models are used in security evaluation, sometimes for proofs of security. The Bell-LaPadula model (BLP) is an important historic milestone in computer security. BLP is a state machine model capturing

confidentiality aspects of access control. Access permissions are defined through an access control matrix and through a partial ordering of security levels. Security policies prevent information flowing downwards from a high security level to a low security level. BLP only considers the information flow that occurs when a subject observes or alters an object.

2.3.4 Patch

A patch is a piece of software designed to fix problems with or update a computer program or its supporting data. This includes fixing security vulnerabilities and other bugs and improving the usability or performance. Though meant to fix problems, poorly designed patches can sometimes introduce new problems.

Patch management is the process of using a strategy and plan of what patches should be applied to which systems at a specified time.

2.3.5 Integrity Checks

Data integrity is data that has a complete or whole structure. All characteristics of the data including business rules, rules for how pieces of data relate dates, definitions and lineage must be correct for data to be complete. It is the process of keeping check on the information shared in the virtual atmosphere. Infact, the functions must ensure integrity before performing on the data. Examples of functions are transforming the data, storing the history, storing the definitions (Metadata) and storing the lineage of the data as it moves from one place to another. The most important aspect of data integrity is to expose the data, the functions and the data's characteristics.

Data that has integrity is identically maintained during any operation (such as transfer, storage or retrieval). Put simply in business terms, data integrity is the assurance that data is consistent, certified and can be reconciled.

In terms of a database data integrity refers to the process of ensuring that a database remains an accurate reflection of the universe of discourse it is modelling or representing. In other words there is a close correspondence between the facts stored in the database and the real world it models.

Database integrity checks are recommended to ensure that the database consistency is intact and if there is a problem with consistency, then it is reported to the appropriate team(s) so that necessary action can be taken to rectify it. This can be done with the help of Database Maintenance Plans.

2.3.6 Software Updates

Make sure that the software on your computer is regularly updated. By doing so, most viruses can be avoided. We recommend setting your computer to check for software updates automatically. This exercise helps in securing the operating system.

2.3.7 Firewall

Basically, a firewall is a barrier to keep destructive forces away from your property. In fact, that's why it's called a firewall. Its job is similar to a physical firewall that keeps a fire from spreading from one area to the next. A firewall is actually a device or program that blocks undesired Internet traffic, including viruses, from accessing your computer. Both Windows and Mac OS X have built-in firewall programs that are easy to set up. By blocking unwanted Internet traffic, a lot of viruses and bugs can be stopped dead in their tracks.

Firewalls make it possible to filter incoming and outgoing traffic that flows through your system. A firewall can use one or more sets of "rules" to inspect the network packets as they come in or go out of your network connections and either allows

the traffic through or blocks it. The rules of a firewall can inspect one or more characteristics of the packets, including but not limited to the protocol type, the source or destination host address and the source or destination port.

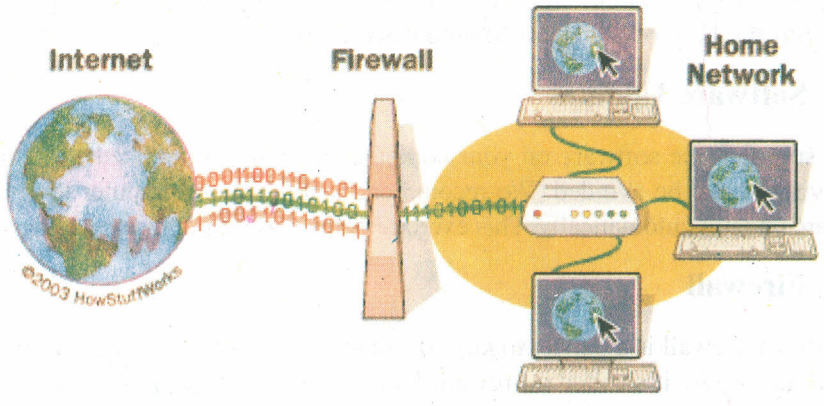
Firewalls can greatly enhance the security of a host or a network. They can be used to do one or more of the following things:

- To protect and insulate the applications, services and machines of your internal network from unwanted traffic coming in from the public Internet.
- To limit or disable access from hosts of the internal network to services of the public Internet.
- To support network address translation (NAT), which allows your internal network to use private IP addresses and share a single connection to the public Internet (either with a single IP address or by a shared pool of automatically assigned public addresses).

The primary purpose of a firewall is to filter traffic. Firewalls inspect packets as they pass through and based on the criteria that the administrator has defined, the firewall allows or denies each packet.

Firewalls block everything that you haven't specifically allowed. Routers with filtering capabilities are a simplified example of a firewall. Administrators often configure them to allow all outbound connections from the internal network, but to block all incoming traffic. So, a user on the internal network would be able to download e-mail without a problem, but an administrator would need to customize the router configuration to connect to your home PC from work by using Remote Desktop. Other applications that might require special firewall configuration are WebCam servers, collaboration software and multiplayer online games.

You use packet filters to instruct a firewall to drop traffic that meets certain criteria. For example, you could create a filter that would drop all ping requests. You can also configure filters with more complex exceptions to a rule. For example, a filter might assist with troubleshooting the firewall by allowing the firewall to respond to ping requests coming from a monitoring station's IP address. By default, Microsoft ISA Server doesn't respond to ping queries on its external interface. You would need to create a packet filter on the ISA Server computer for it to respond to a ping request.



2.3.7.1 Types of Firewalls

There are two main types of firewalls: network firewalls and host-based firewalls.

Network Firewalls

Network firewalls protect an entire network by guarding the perimeter of that network. Network firewalls forward traffic to and from computers on an internal network and filter that traffic based on the criteria the administrator has set. Network

firewalls come in two flavors: hardware firewalls and software firewalls. Hardware-based network firewalls are generally cheaper than software-based network firewalls and are the right choice for home users and many small businesses. Software-based network firewalls often have a larger feature set than hardware-based firewalls and might fit the needs of larger organizations. Software-based firewalls can also run on the same server as other services, such as e-mail and file sharing, allowing small organizations to make better use of existing servers. Network firewalls often include additional features that aren't necessary for host-based firewalls, as described in the following sections. Network firewalls, such as the software-based Microsoft's Internet Security and Acceleration (ISA) Server or the hardware-based Nortel Networks Alteon Switched Firewall System, protect the perimeter of a network by watching traffic that enters and leaves.

Host-Based Firewalls

Host-based firewalls are software firewalls installed on each individual system. Depending on the software you choose, a host-based firewall can offer features beyond those of network firewalls, such as protecting your computer from spyware (a component of some free software that tracks your Web browsing habits) and Trojan horses (a program that claims to do one thing, but does another, malicious thing, such as recording your passwords). If you travel with a laptop, a host-based firewall is a necessity-you need protection wherever you connect to the Internet and your hardware firewall can protect you only at home.

Why would you buy third-party firewall software when Windows XP includes ICF for free? ICF is designed to provide basic intrusion prevention, but doesn't include the rich features of a third-party firewall application. Most third-party firewalls protect you from software that could violate your privacy or allow an attacker to misuse your computer-features not found in ICF. Also, you can install third-party firewall programs on systems that have older versions of Windows. Note that firewall software doesn't replace antivirus software. You should use both.

Popular host-based firewall products include ZoneAlarm, Tiny Personal Firewall, Agnitum Outpost Firewall, Kerio Personal Firewall and Internet Security Systems' BlackICE PC Protection. Most host-based firewall software is available in free or trial versions, so it won't cost you anything to download these packages and determine whether they meet your needs better than ICF.

Host-based firewalls, such as Internet Connection Firewall (ICF included with Windows XP and Windows Server 2003), protect an individual computer regardless of the network it's connected to.

2.3.8 Account Management

Manage the user accounts on your computer, so you can control exactly who can log into your machine. Especially on Windows XP machines, it is easy to accidentally leave your computer wide open to unauthorized users.

2.3.9 Antivirus Software

The prevention against virus is to install anti-virus software and keep the updates current.

Prominent anti-virus software vendors include the following:

1) Mc-Afee
2) Norton Antivirus
3) Quick Heal

4) Kaspersky

5) Trend Micro

6) Avira

7) Avast

8) Panda

9) AVG

Use your antivirus software to scan for viruses as files are being launched. The term "virus" is used to describe self-replicating computer programs that propagate themselves between files on a computer and even between computers. Viruses usually, but not always, do something malicious, such as overwrite files or waste your bandwidth by sending copies of them to everyone in your address book.

Antivirus or anti-virus software is used to prevent, detect and remove computer Viruses, Worms and Trojan horses. It may also prevent and remove adware, spyware and other forms of malware. This page talks about the software used for the prevention and removal of such threats, rather than computer security implemented by software methods.

A variety of strategies are typically employed. Signature-based detection involves searching for known patterns of data within executable code. However, it is possible for a computer to be infected with new malware for which no signature is yet known. To counter such so-called zero-day threats, heuristics can be used. One type of heuristic approach, generic signatures, can identify new viruses or variants of existing viruses by looking for known malicious code or slight variations of such code, in files. Some antivirus software can also predict what a file will do by running it in a sandbox and analyzing what it does to see if it performs any malicious actions.

No matter how useful antivirus software can be, it can sometimes have drawbacks. Antivirus software can impair a computer's performance. Inexperienced users may also have trouble understanding the prompts and decisions that antivirus software presents them with. An incorrect decision may lead to a security breach. If the antivirus software employs heuristic detection, success depends on achieving the right balance between false positives and false negatives. False positives can be as destructive as false negatives. Finally, antivirus software generally runs at the highly trusted kernel level of the operating system, creating a potential avenue of attack.

Identification Methods

There are several methods which antivirus software can use to identify malware.

- Signature based detection is the most common method. To identify viruses and other malware, antivirus software compares the contents of a file to a dictionary of virus signatures. Because viruses can embed themselves in existing files, the entire file is searched, not just as a whole, but also in pieces.
- Heuristic-based detection, like malicious activity detection, can be used to identify unknown viruses.
- File emulation is another heuristic approach. File emulation involves executing a program in a virtual environment and logging what actions the program performs. Depending on the actions logged, the antivirus software can determine if the program is malicious or not and then carry out the appropriate disinfection actions.

Antivirus capabilities are a feature of some network and host-based firewalls. Network firewalls might inspect all incoming e-mail traffic for virus-infected attachments and filter them out. Host-based firewalls might change the configuration of the user's e-mail client so that the e-mail client sends all requests through the host-based firewall.

Firewalls are certainly not the only way to protect yourself from viruses and if the firewall you choose doesn't have antivirus features, you'll need to complement it with antivirus software. The best way to protect your organization against viruses is to use a good-quality commercial antivirus package. These scanners examine the files, folders, mail messages and Web pages on your computers, looking for the distinctive patterns of viral code. When the scanner detects something that looks like a virus, it quarantines the suspect object and warns you about what it found.

Check Your Progress 1

Note: a) Space is given below for writing your answers.

b) Compare your answers with the one given at the end of the Unit.

1) Explain security model.

.....

.....

.....

.....

.....

2) What are network firewalls?

.....

.....

.....

.....

.....

3) What is antivirus software?

.....

.....

.....

.....

.....

2.4 LET US SUM UP

There are various ways for providing security to operating system which is very essential for the proper working of the computer to perform tasks. This unit helps in understanding the possible mechanism for the operating system to function effectively, safely and efficiently.

2.5 CHECK YOUR PROGRESS: THE KEY

Check Your Progress 1

1) Security models

A scheme for specifying and enforcing security policies is a computer security model. A security model may be founded upon a formal model of access rights, a model of computation, a model of distributed computing or no particular theoretical grounding at all. Security models are used in security evaluation, sometimes for proofs of security. The Bell-LaPadula model (BLP) is an important historic milestone in computer security. BLP is a state machine model capturing confidentiality aspects of access control. Access permissions are defined through an access control matrix and through a partial ordering of security levels. Security policies prevent information flowing downwards from a high security level to a low security level. BLP only considers the information flow that occurs when a subject observes or alters an object.

2) Network Firewalls

Network firewalls protect an entire network by guarding the perimeter of that network. Network firewalls forward traffic to and from computers on an internal network and filter that traffic based on the criteria the administrator has set. Network firewalls come in two flavors: hardware firewalls and software firewalls. Hardware-based network firewalls are generally cheaper than software-based network firewalls and are the right choice for home users and many small businesses. Software-based network firewalls often have a larger feature set than hardware-based firewalls and might fit the needs of larger organizations. Software-based firewalls can also run on the same server as other services, such as e-mail and file sharing, allowing small organizations to make better use of existing servers. Network firewalls often include additional features that aren't necessary for host-based firewalls, as described in the following sections. Network firewalls, such as the software-based Microsoft's Internet Security and Acceleration (ISA) Server or the hardware-based Nortel Networks Alteon Switched Firewall System, protect the perimeter of a network by watching traffic that enters and leaves.

3) Antivirus Software

Use your antivirus software to scan for viruses as files are being launched. The term "virus" is used to describe self-replicating computer programs that propagate themselves between files on a computer and even between computers. Viruses usually, but not always, do something malicious, such as overwrite files or waste your bandwidth by sending copies of them to everyone in your address book.

Antivirus capabilities are a feature of some network and host-based firewalls. Network firewalls might inspect all incoming e-mail traffic for virus-infected attachments and filter them out. Host-based firewalls might change the configuration of the user's e-mail client so that the e-mail client sends all requests through the host-based firewall.

Firewalls are certainly not the only way to protect yourself from viruses and if the firewall you choose doesn't have antivirus features, you'll need to complement it with antivirus software. The best way to protect your organization against viruses is to use a good-quality commercial antivirus package. These scanners examine the files, folders, mail messages and Web pages on your computers, looking for the distinctive patterns of viral code. When the scanner detects something that looks like a virus, it quarantines the suspect object and warns you about what it found.

2.6 SUGGESTED READINGS

- http://searchsecurity.techtarget.com/resources/operating-system-security.
- http://technet.microsoft.com/en-us/library/cc700820.aspx.
- www.lynuxworks.com/solutions/security.php.

UNIT 3 OPERATING SYSTEM HARDENING AND CONTROLS

Structure

3.0 INTRODUCTION

Every mechanism needs proper controls to work in the specified dimension. Without such controls or tightening framework, operating system cannot function properly and adequately as required. Therefore, it is extremely important to have the hardening system in place for the operating system to perform work rightly.

3.1 OBJECTIVES

After studying this unit, you should be able to:

- explain operating system hardening;
- explain tasks for ensuring operating system hardening; and
- explain hardening process required for application servers.

3.2 OPERATING SYSTEM HARDENING

System hardening identifies the uses of a particular computer such as a Web server, an e-mail or a voice mail server or an Internet server. System hardening also disables or removes all components that are not required. The components allowed on the system are specific to the functions that the system performs. System hardening tightens system security by limiting the number of users, setting password policies and creating access control lists.

Operating system hardening is the process to address security weaknesses in the operation systems by implementing the latest operating system patches, hot fixes and updates as well as follows up the specific procedures and policies to reduce attacks and system down time.

Hardening is a not a one time activity, it is an on going task to mitigate the risk to performing high quality of computing. We have to build-up the secure production server in such a way to remove the unwanted devices, fix up the miss configuration, not allow the default setting, enhancement the current configuration and develop the new system programming and applying new security patches before going to

the production environment. Hardening of the operating system should be support to the high integrity, reliability, availability, privacy, scalability and confidentiality at the lowest level of risk to achieve the highest level of objective (benefits) from the critical IT infrastructure of the organization.

Safeguarding information and protecting the integrity of your network and systems are vital to our business. IT security professionals in many companies have established policies applicable to their entire organization, but it may be up to individual departments that manage the systems to implement security in accordance with these policies. Security professionals recognize the need for flexibility when it comes to implementation, due to the unique requirements of each department.

Hardening of an operating system involves the removal of all non essential tools, utilities and other systems administration options, any of which could be used to ease a hacker's path to your systems. Following this, the hardening process will ensure that all appropriate security features are activated and configured correctly. Again, 'out of the box' systems will likely be set up for ease of access with access to administrator account. Some vendors have now recognized that a market exists for the OS-hardened systems.

Hardening of the operating system includes planning against both accidental and directed attacks, such as the use of fault-tolerant hardware and software solutions. Additionally, it is important to implement an effective system for file-level security, including encrypted file support and secured file system selection that allows for the proper level of access control. For example, Microsoft's New Technology File System (NTFS) allows for file-level access control, whereas most File Allocation Table-based (FAT-based) systems allow for only share-level access control.

It is also imperative to include regular update reviews for all deployed operating systems in order to address newly identified exploits and apply security hotfixes, patches and service packs. Many automated attacks use common vulnerabilities, often ones for which patches and hotfixes are already available. Failure to include planning for application updates on a regular basis, along with update auditing, can result in an unsecure solution that provides an attacker access to additional resources throughout an organization's network.

IP Security (IPSec) and PKI implementations must also be properly configured and updated to maintain key and ticket stores. Some systems may be hardened to include specific levels of access (for example, hardening a system to gain the C2 security rating required by many government deployment scenarios).

Operating system hardening also includes configuring log files and auditing, changing default administrator account names and default passwords and instituting account lockout and password policies to guarantee strong passwords that will be resistant to brute-force attacks.

System hardening must be well defined in the information security guidelines. The process of hardening a system depends on your operating system. You must ensure that you perform the following tasks:

- **Disable Unnecessary Services**

  The default installation can include more services than you need. Disable the services or features that you do not need to make the system more secure and to provide better performance. For more information about Modular Messaging services, see the installation guide for your configuration. For more information about Windows services, contact your Avaya representative for a complete list of Windows services.

- **Patch the System**

  Install all service packs, security patches and hot fixes, especially those that pertain to the security of the system. Once they are installed, validate all the hardening procedures to ensure that the hardening settings are unchanged. Verify that the service packs did not roll back the configuration settings.

- **Configure File System, Directory and Registry Settings**

  Review and enforce access rights to the file system, directory service and registry. Global read and write access to key directories can lead to a security exposure. In most cases, this level of permission is unnecessary.

- **Configure and Tune Logging**

  Configure the system to log more detail and security-relevant information. One of the best ways to learn about attempted and successful security breaches is to monitor system logs regularly.

- **Ensure Physical Security**

  Ensure that the system is physically secure from unauthorized access. Physical security enforces strong security controls and system hardening.

- **Choose Strong Passwords for Administration Accounts**

  Select the passwords for administration accounts according to the specified guidelines. The administrator passwords must be the most closely guarded passwords on the network.

- **Install Virus-Detection Software**

  Use anti-virus products to monitor, identify and secure your systems from viruses and worms.

- **Verify all Security Settings**

  After you configure the security settings on the host, check all the settings to ensure that they are intact. In many operating systems, when you apply security patches and make changes to settings, previously made changes are lost.

3.3 NETWORK HARDENING

Network hardening involves access restrictions to network shares and services, updates to security hardware and software and disabling unnecessary protocol support and services.

Restricting Access to the Network

Firewall and Network Address Translation (NAT) software and hardware solutions provide the first layer of defense against unauthorized access attempts.

Mapping avenues of access is also critical in hardening a network. This process is a part of the site survey that should be performed for any network, especially those that involve public areas where a simple connection through a workstation might link the protected internal network directly to a public broadband connection.

Wireless networks also create significant avenues for unsecure access to a secured network. A user who configures a PC card on her workstation to allow for the synchronization of her compliant wireless PDA may have inadvertently bypassed all security surrounding an organization's network.

If a centralized access control system is used, such as those found in Windows and Novell networks, resource access and restrictions may be assigned to groups and users can be granted membership to those groups. Properly configured access control lists help provide resource access to authorized parties and also limit potential avenues of unauthorized access.

Updating Security Hardware and Software

As with operating system hardening, default configurations and passwords must be changed in network hardware such as routers and managed network devices. Routing hardware must also be maintained in a current state by regularly reviewing applied firmware updates and applying updates that are required for the network configuration and hardware solutions used.

Security software packages need to be updated with as much vigilance as hardware. New tools, better protection and up-to-date virus and attack definition files become available on almost a daily basis. A regular schedule should be identified and followed for proper update procedures for both security hardware and software.

Disabling Unnecessary Protocols and Services

Leaving protocols and services open and unconfigured when they are not necessary for your network can be a dangerous situation. When you install items on your network, we suggest that you do not accept default configurations because the defaults offered may not meet the business and security requirements of your network.

For example, in a homogenous network such as an all-Windows 2000 network, it might be possible to terminate support for AppleTalk, Internetwork Packet Exchange/Sequenced Packet Exchange (IPX/SPX) or other forms of unused network communications protocols. Because you don't have any Macintosh clients or Novell systems on the network, you don't need the services and protocols associated with these systems. By eliminating services such as these, you are closing holes that can potentially be exploited by attackers.

3.4 APPLICATION HARDENING

Each application and service that may be installed within a network must also be considered when planning security for an organization. Applications must be maintained in an updated state through the regular review of hotfixes, patches and service packs. Many applications, such as antivirus software, require regular updates to provide protection against newly emerging threats. Default application-administration accounts, standard passwords and common services installed by default should also be reviewed and changed or disabled as required.

Web Servers

Access restrictions to Internet and intranet Web services may be required to ensure proper authentication for nonpublic sites, whereas anonymous access may be required for other sites. Access control may be accomplished at the operating system or application level, with many sites requiring regular updates of Secure Sockets Layer (SSL) certifications for secured communications.

Regular log review is critical for Web servers to ensure that submitted URL values are not used to exploit unpatched buffer overruns or other forms of common exploits. Many Web servers may also include security add-ins, provided to restrict those URLs that may be meaningfully submitted, filtering out any that do not meet the defined criteria. Microsoft's URL Scan for the Internet Information Services (IIS) Web service is one such filtering add-in.

E-mail Services

E-mail servers require network access to transfer Simple Mail Transfer Protocol (SMTP) traffic. E-mail is often used to transport executable agents, including Trojan horses and other forms of viral software. E-mail servers may require transport through firewall solutions to allow remote Post Office Protocol version 3 (POP3) or Internet Message Access Protocol (IMAP) access or they may require integration with VPN solutions to provide secure connections for remote users. User authentication is also of key importance, especially when e-mail and calendaring solutions allow delegated review and manipulation. Inadequate hardware may be attacked through mail bombs and other types of attacks meant to overwhelm the server's ability to transact e-mail messages.

FTP Servers

File Transfer Protocol (FTP) servers are used to provide file upload and download capabilities to users, whether through anonymous or authenticated connections. Because of limitations in the protocol, unless an encapsulation scheme is used between the client and host systems, the logon and password details are passed in cleartext and may be subject to interception via packet sniffing. Unauthorized parties may also use FTP servers that allow anonymous access to share files of questionable or undesirable content while also consuming network bandwidth and server processing resources.

DNS Servers

DNS servers are responsible for name resolution and may be subject to many forms of attack, including attempts at denial of service (DoS) attacks intended to prevent proper name resolution for key corporate holdings. Hardening DNS server solutions should include planning for redundant hardware and software solutions, along with regular backups to protect against loss of name registrations. Technologies that allow dynamic updates must also include access control and authentication to ensure that registrations are valid.

NNTP Servers

Network News Transfer Protocol (NNTP) servers provide user access to newsgroup posts and share many of the same security considerations that e-mail servers generate. Access control for newsgroups may be somewhat more complex, with moderated groups allowing public anonymous submission with authenticated access required for post approval. Heavily loaded servers may be attacked to perform a denial of service and detailed user account information in public newsgroup posting stores, such as those of the AOL and MSN communities, may be exploited in many ways.

File and Print Servers

User file storage solutions often come under attack when unauthorized access attempts provide avenues for manipulation. Files may be corrupted, modified, deleted or manipulated in many ways. Access control through proper restriction of file and share permissions is necessary, coupled with access auditing and user-authentication schemes to ensure proper access. Removal of default access permissions, such as the automatic granting of allow access to everyone group in Windows systems must be done before network file shares can be secured.

Distributed file system and encrypted file system solutions may require bandwidth planning and proper user authentication to allow even basic access. Security planning for these solutions may also include placing user-access authenticating servers close to the file servers to decrease delays created by authentication traffic.

Print servers also pose several risks, including possible security breaches in the

event that unauthorized parties may access cached print jobs. Denial of service attacks may be used to disrupt normal methods of business. Network connected printers require authentication of access to prevent attackers from generating printed memos, invoices or any other manner of printed materials as desired.

DHCP Servers

DHCP servers share many of the same security problems associated with other network services, such as DNS servers. DHCP servers may be overwhelmed by lease requests if bandwidth and processing resources are insufficient. This can be worsened by the use of DHCP proxy systems relaying lease requests from widely deployed subnets. Scope address pools may also be overcome if lease duration is insufficient and short lease duration may increase request traffic. If the operating system in use does not support DHCP server authentication, attackers may also configure their own DHCP servers within a subnet, taking control of the network settings of clients obtaining leases from the rogue servers. Planning for DHCP security must include regular review of networks for unauthorized DHCP servers.

Data Repositories

Data repositories of any type may require specialized security considerations based on the following:

- The bandwidth and processing resource requirements that are needed to prevent denial of service attacks
- The removal of default password and administration accounts (such as the SQL default "sa" account)
- Security of replication traffic to prevent exposure of access credentials to packet sniffing

Placement of authentication, name resolution and data stores within secured and partially secured zones, such as an organization's DMZ, may require the use of secured VPN connections or the establishment of highly secured bastion hosts. Role-Based Access Control (RBAC) may be used to improve security and the elimination of unneeded connection libraries and character sets may help to alleviate common exploits.

Check Your Progress 1

Note: a) Space is given below for writing your answers.

b) Compare your answers with the one given at the end of the Unit.

1) What task you will perform for ensuring operating system hardening?

.....

.....

.....

.....

2) What are FTP servers?

.....

.....

.....

.....

3) What are data repositories?

.....

.....

.....

.....

3.5 LET US SUM UP

It is very important to see over the controls in place for the proper working of operating system. This unit helps in understanding of the controls and hardening process needed for the securing of operating system. It emphasizes on the importance of such mechanism which helps in controlling the operating system to function their tasks properly and effectively.

3.6 CHECK YOUR PROGRESS: THE KEY

Check Your Progress 1

1) We must perform the following tasks for ensuring operating system hardening:

- **Disable unnecessary services**

  The default installation can include more services than you need. Disable the services or features that you do not need to make the system more secure and to provide better performance. For more information about Modular Messaging services, see the installation guide for your configuration. For more information about Windows services, contact your Avaya representative for a complete list of Windows services.

- **Patch the system**

  Install all service packs, security patches and hot fixes, especially those that pertain to the security of the system. Once they are installed, validate all the hardening procedures to ensure that the hardening settings are unchanged. Verify that the service packs did not roll back the configuration settings.

- **Configure file system, directory and registry settings**

  Review and enforce access rights to the file system, directory service and registry. Global read and write access to key directories can lead to a security exposure. In most cases, this level of permission is unnecessary.

- **Configure and tune logging**

  Configure the system to log more detail and security-relevant information. One of the best ways to learn about attempted and successful security breaches is to monitor system logs regularly.

- **Ensure physical security**

  Ensure that the system is physically secure from unauthorized access. Physical security enforces strong security controls and system hardening.

- **Choose strong passwords for administration accounts**

  Select the passwords for administration accounts according to the specified guidelines. The administrator passwords must be the most closely guarded

passwords on the network.

- **Install virus-detection software**

  Use anti-virus products to monitor, identify and secure your systems from viruses and worms.

- **Verify all security settings**

  After you configure the security settings on the host, check all the settings to ensure that they are intact. In many operating systems, when you apply security patches and make changes to settings, previously made changes are lost.

2) FTP Servers

File Transfer Protocol (FTP) servers are used to provide file upload and download capabilities to users, whether through anonymous or authenticated connections. Because of limitations in the protocol, unless an encapsulation scheme is used between the client and host systems, the logon and password details are passed in cleartext and may be subject to interception via packet sniffing. Unauthorized parties may also use FTP servers that allow anonymous access to share files of questionable or undesirable content while also consuming network bandwidth and server processing resources.

3) Data Repositories

Data repositories of any type may require specialized security considerations based on the following:

- The bandwidth and processing resource requirements that are needed to prevent denial of service attacks
- The removal of default password and administration accounts (such as the SQL default "sa" account)
- Security of replication traffic to prevent exposure of access credentials to packet sniffing

Placement of authentication, name resolution and data stores within secured and partially secured zones, such as an organization's DMZ, may require the use of secured VPN connections or the establishment of highly secured bastion hosts. Role-Based Access Control (RBAC) may be used to improve security and the elimination of unneeded connection libraries and character sets may help to alleviate common exploits.

3.7 SUGGESTED READINGS

- http://learning.infocollections.com/ebook%202/Computer/Cert/Security_Plus_Exam_Cram_2_(SYO-101)/0789729105_ch07lev1sec2.html.
- http://support.avaya.com/elmodocs2/mm/r_4_0_doc/cd_frontend/a_mss_mas/se_syshard.htm.
- http://www.interscience.in/ijcct/IJCCT_Paper7.pdf.

UNIT 4 ADC/SAMBA

Structure

4.0 INTRODUCTION

The structure of the Active Directory Controller includes Forests, trees, domains, Flat-filed, simulated hierarchy and Shadow Groups. The different Structural divisions to improve performance are FSMO Roles, Trust, Adding Users and Computers to the Active Directory Domain and Using Active Directory with Desktop Delivery Controller. It also describes digital forensics. SAMBA provides Windows networking services, on a Unix-like platform. These services range from simple file and printer sharing, to full management of a NT-style domain. All of these services are provided in the SAMBA package, which is itself distributed under the Free Software Foundation's General Public Licence (GPL). SAMBA allows file and print sharing between computers running Windows and computers running UNIX.

4.1 OBJECTIVES

After completion of this unit, you will be able to:

- describe ADC and it's structure; and
- describe SAMBA.

4.2 ACTIVE DIRECTORY CONTROLLER (ADC)

4.2.1 Structure of Active Directory Controller

An Active Directory structure is a hierarchical framework of objects. The objects fall into two broad categories: resources (e.g. printers) and security principals (user

or computer accounts and groups). Security principals are Active Directory objects that are assigned unique security identifiers (SIDs) used to control access and set security.

Each object represents a single entity – whether a user, a computer, a printer or a group - and its attributes. Certain objects can also be containers of other objects. An object is uniquely identified by its name and has a set of attributes – the characteristics and information that the object can contain – defined by a schema, which also determines the kinds of objects that can be stored in Active Directory.

Each attribute object can be used in several different schema class objects. The schema object exists to allow the schema to be extended or modified when necessary. However, because each schema object is integral to the definition of Active Directory objects, deactivating or changing these objects can have serious consequences because it will fundamentally change the structure of Active Directory itself. A schema object, when altered, will automatically propagate through Active Directory and once it is created it can only be deactivated – not deleted. Changing the schema usually requires a fair amount of planning.

A Site object in Active Directory represents a geographic location that hosts networks. Sites contain objects called subnets Sites can be used to assign Group Policy, facilitate the discovery of resources, manage active directory replication and manage network link traffic. Sites can be linked to other Sites. Site-linked objects may be assigned a cost value that represents the speed, reliability, availability or other real property of a physical resource. Site Links may also be assigned a schedule.

4.2.1.1 Forests, Trees and Domains

All objects inside a common directory database are known as a domain. Each domain stores information only about the objects that belong to that domain. A tree consists of a single domain or multiple domains in a contiguous namespace. A forest is a collection of trees and represents the outermost boundary within which users, computers, groups and other objects exist. The Active Directory framework that holds the objects can be viewed at a number of levels. At the top of the structure is the forest. A forest is a collection of multiple trees that share a common global catalog, directory schema, logical structure and directory configuration. The forest, tree and domain are the logical parts in an Active Directory network.

The Active Directory forest contains one or more transitive, trust-linked trees. A tree is a collection of one or more domains and domain trees in a contiguous namespace, again linked in a transitive trust hierarchy. Domains are identified by their DNS name structure, the namespace.

4.2.1.2 Flat-filed, Simulated Hierarchy

The objects held within a domain can be grouped into containers called Organizational Units (OUs). OUs give a domain a hierarchy, ease its administration and can give a resemblance of the structure of the organization in organizational or geographical terms. OUs can contain OUs – indeed, domains are containers in this sense – and can hold multiple nested OUs. Microsoft recommends as few domains as possible in Active Directory and a reliance on OUs to produce structure and improve the implementation of policies and administration. The OU is the common level at which to apply group policies, which are Active Directory objects themselves called Group Policy Objects (GPOs), although policies can also be applied to domains or sites. The OU is the level at which administrative powers are commonly delegated, but granular delegation can be performed on individual objects or attributes as well.

However, Organizational Units are just an abstraction for the administrator and do not function as true containers; the underlying domain operates as if objects were all created in a simple flat-file structure, without any OUs. By contrast, there are other vendor directories such as Novell eDirectory that allow naming attribute duplication across separate OUs. Each user logs in by specifying the context of their account, which is similar to the current working directory of a file system. Context normally operates in relative form: if the login prompt context is "staff-ou.accounts-ou.organization", people with accounts in that specific OU need only type their username "fred". But if the login prompt context were set to be one level higher, at "accounts-ou.organization", people would need to specify the OU within that context: "fred.staff-ou". Context can also be specified in absolute form similar to an absolute directory path by using a leading period: ".fred.staff-ou.accounts-ou.organization", which disregards the current login prompt context.

Novell additionally provides login prompt functionality known as contextless login to permit searching the directory structure via LDAP for all possible matching or similar usernames, making the Novell login process operate similar to Microsoft's flat-file structure that searches the entire domain for accounts regardless of the account's location in the OUs. The concept of account context in the directory does not apply to Active Directory, since object name duplication within a single domain is not permitted to occur in the first place.

Because duplicate usernames cannot exist within separate OUs of a single active directory domain, unique account name generation poses a significant challenge for organizations with hundreds to thousands of users that are part of a generalized mass that can not be easily subdivided into separate domains, such as students in a public school system or university that must be able to login on any computer across the district buildings or campus network.

As the number of users in a domain increases, simple username creation methods such as "first initial, middle initial, last name" will fail due to having so many common names like Smith or Johnson in the collective mass that result in having duplications, such as two JASmith, which requires randomly adding a number to the end (JASmith1) to further differentiate it for one of the two people. At some point of increasingly many users and name duplications, the network IT staff may give up on attempts at making usernames personally memorable and the username simply becomes a serial number 5 to 10 digits long to provide sufficient naming uniqueness within a single domain.

4.2.1.3 Shadow Groups

In Active Directory, organizational units can not be assigned as owners or trustees. Only groups are selectable and members of OUs can not be collectively assigned rights to directory objects.

Unlike Active Directory, Novell eDirectory allows organizational units and all users within the OU to be assigned rights to an object, without having to create shadow groups representing the users in each OU.

It is often useful to associate a collection of users to all share access rights to particular file or secured resource, but with Active Directory it is not possible to choose an OU containing all users that need rights. A user group can be selected to accomplish this, but all users within a particular OU are not automatically made members of a group representing that OU.

Groups can be manually created to duplicate the account membership structure within OUs, but it is an extra step of the account creation process by the administrator to remember all the various groups each new user needs to join. If the administrator forgets this manual step, the users will experience problems until the group memberships are corrected.

To make up for this non-automated deficiency, network administrators can write their own custom scripts which periodically run on the server and use LDAP access commands to add or remove users from groups representing the OUs of the users, known as Shadow Groups. Microsoft refers to shadow groups in the Server 2008 Reference documentation, but does not explain how to create them. Once created, these shadow groups are selectable in place of the OU in the administrative console tools.

The naming of shadow groups is complicated by the fact that OUs can be nested but groups cannot. Groups can only exist in the root of the domain and group names are limited in length so matching the naming of a deeply nested string of OUs for a very large domain is difficult.

Novell e-Directory supports the creation of user groups, but OUs can be natively selected as the assigned owner of a secured resource, so shadow groups are unnecessary.

4.2.2 Structural Divisions to Improve Performance

Active Directory also supports the creation of Sites, which are physical, rather than logical, groupings defined by one or more IP subnets. Sites distinguish between locations connected by low-speed (e.g. WAN, VPN) and high-speed (e.g. LAN) connections. Sites are independent of the domain and OU structure and are common across the entire forest. Sites are used to control network traffic generated by replication and also to refer clients to the nearest domain controllers. Exchange 2007 also uses the site topology for mail routing. Policies can also be applied at the site level.

The actual division of an organization's information infrastructure into a hierarchy of one or more domains and top-level OUs is a key decision. Common models are by business unit, by geographical location, by IT Service or by object type. These models are also often used in combination. OUs should be structured primarily to facilitate administrative delegation and secondarily, to facilitate group policy application.

Although OUs form an administrative boundary, the only true security boundary is the forest itself and an administrator of any domain in the forest must be trusted across all domains in the forest.

Physically the Active Directory information is held on one or more equal peer domain controllers (DCs), replacing the NT PDC/BDC model. Each DC has a copy of the Active Directory; changes on one computer being synchronized (converged) between all the DC computers by multi-master replication. Servers joined to Active Directory that are not domain controllers are called Member Servers.

The Active Directory database is split into different stores or partitions. Microsoft often refers to these partitions as 'naming contexts'. The 'Schema' partition contains the definition of object classes and attributes within the Forest. The 'Configuration' partition contains information on the physical structure and configuration of the forest (such as the site topology). The 'Domain' partition holds all objects created in that domain. The first two partitions replicate to all domain controllers in the Forest. The Domain partition replicates only to Domain Controllers within its domain. A subset of objects in the domain partition is also replicated to domain controllers that are configured as global catalogs.

Unlike earlier versions of Windows, which used NetBIOS to communicate, Active Directory is fully integrated with DNS and TCP/IP-DNS is required. To be fully functional, the DNS server must support SRV resource records or service records.

Active Directory replication is 'pull' rather than 'push'. The Knowledge Consistency Checker (KCC) creates a replication topology of site links using the defined sites to manage traffic. Intrasite replication is frequent and automatic as a result of change notification, which triggers peers to begin a pull replication cycle. Intersite replication intervals are less frequent and do not use change notification by default, although this is configurable and can be made identical to intrasite replication.

A different 'cost' can be given to each link (e.g. DS3, T1, ISDN etc.) and the site link topology will be altered accordingly by the KCC. Replication between domain controllers may occur transitively through several site links on same-protocol site link bridges, if the cost is low, although KCC automatically costs a direct site-to-site link lower than transitive connections. Site-to-site replication can be configured to occur between a bridgehead server in each site, which then replicates the changes to other DCs within the site.

In a multi-domain forest the Active Directory database becomes partitioned. That is, each domain maintains a list of only those objects that belong in that domain. So, for example, a user created in Domain A would be listed only in Domain A's domain controllers. Global catalog (GC) servers are used to provide a global listing of all objects in the Forest. The Global catalog is held on domain controllers configured as global catalog servers. Global Catalog servers replicate to themselves all objects from all domains and hence, provide a global listing of objects in the forest. However, in order to minimize replication traffic and to keep the GC's database small, only selected attributes of each object are replicated. This is called the partial attribute set (PAS). The PAS can be modified by modifying the schema and marking attributes for replication to the GC.

Replication of Active Directory uses Remote Procedure Calls (RPC over IP [RPC/IP]). Between Sites you can also choose to use SMTP for replication, but only for changes in the Schema, Configuration or Partial Attribute Set (Global Catalog) NCs. SMTP cannot be used for replicating the default Domain partition.

The Active Directory database, the directory store, in Windows 2000 Server uses the JET Blue-based Extensible Storage Engine (ESE98), limited to 16 terabytes and 1 billion objects in each domain controller's database. Microsoft has created NTDS databases with more than 2 billion objects. (NT4's Security Account Manager could support no more than 40,000 objects). Called NTDS.DIT, it has two main tables: the data table and the link table. In Windows Server 2003 a third main table was added for security descriptor single instancing.The features of Active Directory may be accessed programmatically via the COM interfaces provided by Active Directory Service Interfaces. Active Directory is a necessary component for many Windows services in an organization such as Exchange, Security.

4.2.2.1 FSMO Roles

Flexible Single Master Operations (FSMO, sometimes pronounced "fizz-mo") roles are also known as operations master roles. Although the AD domain controllers operate in a multi-master model, i.e. updates can occur in multiple places at once, there are several roles that are necessarily single instance:

Role Name	Scope	Description
Schema Master	1 per forest	Controls and handles updates/modifications to the Active Directory schema.
Domain Naming Master	1 per forest	Controls the addition and removal of domains from the forest if present in root domain

PDC Emulator	1 per domain	Provides backwards compatibility for NT4 clients for PDC operations (like password changes). The PDCs also run domain specific processes such as the Security Descriptor Propagator (SDPROP), and is the master time server within the domain.
		It also handles external trusts, the DFS consistency check, holds the most current passwords and manages all GPOs as default server.
RID Master	1 per domain	Allocates pools of unique identifier to domain controllers for use when creating objects
Infrastructure Master	1 per domain/ partition	Synchronizes cross-domain group membership changes. The infrastructure master cannot run on a global catalog server (GCS)(unless all DCs are also GCs, or environment consists of a single domain)

4.2.2.2 Trust

To allow users in one domain to access resources in another, Active Directory uses trusts. Trusts inside a forest are automatically created when domains are created. The forest sets the default boundaries of trust, not the domain and implicit, transitive trust is automatic for all domains within a forest. As well as two way transitive trust, AD trusts can be a shortcut (joins two domains in different trees, transitive, one or two way), forest (transitive, one or two way), realm (transitive or nontransitive, one or two way) or external (nontransitive, one or two way) in order to connect to other forests or non-AD domains.

Trusts in Windows 2000 (native mode)

- **One-way trust** – One domain allows access to users on another domain, but the other domain does not allow access to users on the first domain.
- **Two-way trust** – Two domains allow access to users on both domains.
- **Trusting domain** – The domain that allows access to users from a trusted domain.
- **Trusted domain** – The domain that is trusted; whose users have access to the trusting domain.
- **Transitive trust** – A trust that can extend beyond two domains to other trusted domains in the forest.
- **Intransitive trust** – A one way trust that does not extend beyond two domains.
- **Explicit trust** – A trust that an admin creates. It is not transitive and is one way only.
- **Cross-link trust** – An explicit trust between domains in different trees or in the same tree when a descendant/ancestor (child/parent) relationship does not exist between the two domains.

Windows 2000 Server – supports the following types of trusts:

- Two-way transitive trusts.
- One-way intransitive trusts.

Additional trusts can be created by administrators. These trusts can be:

- Shortcut

Windows Server 2003 offers a new trust type - the forest root trust. This type of trust can be used to connect Windows Server 2003 forests if they are operating at the 2003 forest functional level. Authentication across this type of trust is Kerberos based (as opposed to NTLM). Forest trusts are also transitive for all the domains in the forests that are trusted. Forest trusts, however, are not transitive.

ADAM/AD LDS

Active Directory Application Mode (ADAM) is a light-weight implementation of Active Directory. ADAM is capable of running as a service, on computers running Microsoft Windows Server 2003 or Windows XP Professional. ADAM shares the code base with Active Directory and provides the same functionality as Active Directory, including an identical API, but does not require the creation of domains or domain controllers.

Like Active Directory, ADAM provides a Data Store, which is a hierarchical datastore for storage of directory data, a Directory Service with an LDAP Directory Service Interface. Unlike Active Directory, however, multiple ADAM instances can be run on the same server, with each instance having its own and required by applications making use of the ADAM directory service.

Integrating Unix into Active Directory

Varying levels of interoperability with Active Directory can be achieved on most Unix-like operating systems through standards compliant LDAP clients, but these systems usually lack the automatic interpretation of many attributes associated with Windows components, such as Group Policy and support for one-way trusts.

There are also third-party vendors who offer Active Directory integration for Unix platforms (including UNIX, Linux, Mac OS X and a number of Java- and UNIX-based applications). Some of these vendors include Centrify (DirectControl), Computer Associates (UNAB), CyberSafe Limited (TrustBroker), Likewise Software (Open or Enterprise), Quest Software (Authentication Services) and Thursby Software Systems (ADmitMac). The open source SAMBA software provides a way to interface with Active Directory and join the AD domain to provide authentication and authorization: version 4 (in alpha as of October 2009) can act as a peer Active Directory domain controller. Microsoft is also in this market with their free Microsoft Windows Services for UNIX product.

The schema additions shipped with Windows Server 2003 R2 include attributes that map closely enough to RFC 2307 to be generally usable. The reference implementation of RFC 2307, nss_ldap and pam_ldap provided by PADL.com, contains support for using these attributes directly, provided they have been populated. The default Active Directory schema for group membership complies with the proposed extension, RFC 2307bis. Windows Server 2003 R2 includes a Microsoft Management Console snap-in that creates and edits the attributes.

An alternate option is to use another directory service such as 389 Directory Server (formerly Fedora Directory Server) or Sun Microsystems Sun Java System Directory Server, which can perform a two-way synchronization with Active Directory and thus provide a "deflected" integration with Active Directory as Unix and Linux clients will authenticate to FDS and Windows Clients will authenticate to Active Directory. Another option is to use OpenLDAP with its translucent overlay, which can extend entries in any remote LDAP server with additional attributes stored in a local database. Clients pointed at the local database will see entries containing both the remote and local attributes, while the remote database remains completely untouched.

4.2.2.3 Adding Users and Computers to the Active Directory Domain

After the new Active Directory domain is established, create a user account in that domain to use as an administrative account. When that user is added to the appropriate security groups, use that account to add computers to the domain.

1) To create a new user, follow these steps:

   a) Click **Start**, point to **Administrative Tools** and then click **Active Directory Users and Computers** to start the Active Directory Users and Computers console.

   b) Click the domain name that you created and then expand the contents.

   c) Right-click **Users**, point to **New** and then click **User**.

   d) Type the first name, last name and user logon name of the new user and then click **Next**.

   e) Type a new password, confirm the password and then click to select one of the following check boxes:

   - Users must change password at next logon (recommended for most users)
   - User cannot change password
   - Password never expires
   - Account is disabled

   Click **Next**.

   f) Review the information that you provided and if everything is correct, click Finish.

2) After you create the new user, give this user account membership in a group that permits that user to perform administrative tasks. Because this is a laboratory environment that you are in control of, you can give this user account full administrative access by making it a member of the Schema, Enterprise and Domain administrators groups. To add the account to the Schema, Enterprise and Domain administrators groups, follow these steps:

   - On the Active Directory Users and Computers console, right-click the new account that you created and then click **Properties**.

   a) Click the **Member of** tab and then click **Add**.

   b) In the **Select Groups** dialog box, specify a group and then click **OK** to add the groups that you want to the list.

   c) Repeat the selection process for each group in which the user needs account membership.

   d) Click **OK** to finish.

3) The final step in this process is to add a member server to the domain. This process also applies to workstations. To add a computer to the domain, follow these steps:

   - Log on to the computer that you want to add to the domain.

   a) Right-click **My Computer** and then click **Properties**.

   b) Click the **Computer Name** tab and then click **Change**.

c) In the **Computer Name Changes** dialog box, click **Domain** under **Member of** and then type the domain name. Click **OK**.

d) When you are prompted, type the user name and password of the account that you previously created and then click OK.

A message that welcomes you to the domain is generated.

e) Click **OK** to return to the **Computer Name** tab and then click **OK** to finish.

f) Restart the computer if you are prompted to do so.

4.2.2.4 Using Active Directory with Desktop Delivery Controller

Desktop Delivery Controller uses the services provided by Active Directory. It requires that all computers in a farm are members of a domain, with mutual trusting relationships between the domain used by Desktop Delivery Controller and the domain(s) used by virtual desktops.

Note: If your organizational structure means that you need a deployment where the Desktop Delivery Controller servers are in a separate Active Directory forest from the desktops for your users.It is important to understand how Desktop Delivery Controller uses Active Directory to appreciate the implications for your Active Directory environment.

Desktop Delivery Controller uses Active Directory for two main purposes:

- Active Directory's inbuilt security infrastructure is used by desktops to verify that communications from controllers come from authorized controllers in the appropriate farm. Active Directory's security infrastructure also ensures that the data exchanged by desktops and controllers is confidential. Desktop Delivery Controller uses Active Directory's inbuilt Kerberos infrastructure to guarantee the authenticity and confidentiality of communication. For more information about Kerberos, refer to Microsoft's product documentation.

- Active Directory is optionally used by desktops to discover the controllers that constitute a farm. This means you can add a new controller to a farm without having to reconfigure all desktops in the farm. Instead, desktops determine which controllers are available by referring to information that controllers publish in Active Directory. This feature is available only if the desktops are in the same Active Directory forest as the controllers.

When you create a farm, a corresponding Organizational Unit (OU) must be created in Active Directory if you want desktops to discover the controllers in the farm through Active Directory. The OU can be created in any domain in the forest that contains your computers. As best practice the OU should also contain the delivery controllers in the farm, but this is not enforced or required. A domain administrator with appropriate privileges can create the OU as an empty container. The domain administrator can then delegate administrative authority over the OU to the Desktop Delivery Controller administrator. If the installing administrator has CreateChild permissions on a parent OU, this administrator can also create the farm OU through the Active Directory Configuration wizard during installation. You can use the standard Active Directory Users and Computers MMC snap-in to configure these permissions.

During the Desktop Delivery Controller installation process, a small number of objects that are essential for the operation of the farm are created in the OU.

Note: Only standard Active Directory objects are created and used by Desktop Delivery Controller. It is not necessary to extend the schema.

The set of objects created includes:

- A Controllers security group. The computer account of all controllers in the farm must be a member of this security group. By default, this is done as part of installing Desktop Delivery Controller on a server. Desktops in a farm accept data from controllers only if they are members of this security group.

  Ensure that all controllers have the 'Access this computer from the network' privilege on all virtual desktops running the Virtual Desktop Agent. You can do this by giving the Controllers security group this privilege. If controllers do not have this privilege, virtual desktops will fail to register.

- A Service Connection Point (SCP) object that contains information about the farm, such as the farm's name.

  Note: If you use the Active Directory Users and Computers administrative tool to inspect a farm OU, you may have to enable Advanced Features in the View menu to see SCP objects.

- A container called RegistrationServices, which is created within the farm's OU. This contains one SCP object for each controller in the farm. The SCP is created when Desktop Delivery Controller is installed on a server. Each time the controller starts, it validates the contents of its SCP and updates them if necessary.

If multiple administrators are likely to add and remove controllers after the initial installation is complete, they need permissions to create and delete children on the Registration Services container and Write properties on the Controllers security group. Either the domain administrator or the original installing administrator can grant these permissions and Citrix recommends setting up a security group to do this.

The following points are important to bear in mind when you are using a farm OU with Desktop Delivery Controller:

- Information is written to Active Directory only when installing or uninstalling Desktop Delivery Controller or when a controller starts and needs to update the information in its SCP (for example, because the controller was renamed or because the communication port was changed). By default, the installation routine sets up permissions on the objects in the farm's OU appropriately, giving controllers Write access to their SCP. The contents of the objects in the farm OU are used to establish trust between desktops and controllers. You should ensure that:

  - Only authorized Desktop Delivery Controller administrators can add or remove computers from the Controllers security group, using the security group's access control list (ACL)

  - Only authorized administrators and the respective controller can change the information in the controller's SCP

- Depending on your Active Directory infrastructure, you should be aware of replication and its impact on a Desktop Delivery Controller implementation. Refer to Microsoft's documentation to understand the concepts of replication and associated delays. This is particularly important if you create the farm's OU in a domain that has domain controllers located in multiple Active Directory sites. Depending on the location of desktops, delivery controllers and domain controllers, changes that are made to Active Directory when you are initially creating the OU for the farm, installing or uninstalling controllers or changing controller names or communication ports may not be visible to desktops until that information is replicated to the appropriate domain controller. The

symptoms of such replication delay include desktops that cannot establish contact with controllers and are, therefore, not available for user connections.

- Desktop Delivery Controller uses some of the standard computer object attributes in Active Directory to manage desktops. Depending on your setup, the machine object's fully qualified domain name, as stored in the desktop's Active Directory record, can be included as part of the connection settings that are returned to the user to make a connection. It is, therefore, important to ensure that this information is consistent with information held in your DNS environment.

Check Your Progress 1

Note: a) Space is given below for writing your answers.

b) Compare your answers with the one given at the end of the Unit.

1) Explain active directory structure.

.....

.....

.....

.....

.....

2) List out the partitions of directory information tree.

.....

.....

.....

.....

.....

3) Explain the role of DNS in Active directory.

.....

.....

.....

.....

.....

4) Explain about active directory application mode.

.....

.....

.....

.....

.....

.....

4.3 SAMBA

SAMBA is the standard Windows interoperability suite of programs for Linux and Unix. SAMBA is Free Software licensed under the GNU General Public License, the SAMBA project is a member of the Software Freedom Conservancy. Since 1992, SAMBA has provided secure, stable and fast file and print services for all clients using the SMB/CIFS protocol, such as all versions of DOS and Windows, OS/2, Linux and many others. SAMBA is an important component to seamlessly integrate Linux/Unix Servers and Desktops into Active Directory environments using the winbind daemon.

SAMBA provides Windows networking services, on a Unix-like platform. These services range from simple file and printer sharing, to full management of a NT-style domain. All of these services are provided in the SAMBA package, which is itself distributed under the Free Software Foundation's General Public Licence (GPL).

SAMBA is a free software re-implementation, originally developed by Andrew Tridgell, of the SMB/CIFS networking protocol. As of version 3, SAMBA provides file and print services for various Microsoft Windows clients and can integrate with a Windows Server domain, either as a Primary Domain Controller (PDC) or as a domain member. It can also be part of an Active Directory domain.

SAMBA runs on most Unix and Unix-like systems, such as GNU/Linux, Solaris, AIX and the BSD variants, including Apple's Mac OS X Server (which was added to the Mac OS X client in version 10.2). SAMBA is standard on nearly all distributions of Linux and is commonly included as a basic system service on other Unix-based operating systems as well. SAMBA is released under the GNU General Public License. The name SAMBA comes from SMB (Server Message Block), the name of the standard protocol used by the Microsoft Windows network file system.

SAMBA allows file and print sharing between computers running Windows and computers running Unix. It is an implementation of dozens of services and a dozen protocols, including the NetBIOS over TCP/IP (NBT), SMB, CIFS (an enhanced version of SMB), DCE/RPC or more specifically, MSRPC, the Network Neighborhood suite of protocols, a WINS server also known as a NetBIOS Name Server (NBNS), the NT Domain suite of protocols which includes NT Domain Logons, Secure Accounts Manager (SAM) database, Local Security Authority (LSA) service, NT-style printing service (SPOOLSS), NTLM and more recently Active Directory Logon which involves a modified version of Kerberos and a modified version of LDAP. All these services and protocols are frequently incorrectly referred to as just NetBIOS or SMB. The NetBIOS and WINS protocols are deprecated on Windows.

SAMBA sets up network shares for chosen Unix directories (including all contained subdirectories). These appear to Microsoft Windows users as normal Windows folders accessible via the network. Unix users can either mount the shares directly as part of their file structure using the smbmount command or, alternatively, can use a utility, smbclient (libsmb) installed with SAMBA to read the shares with a similar interface to a standard command line FTP program. Each directory can have different access privileges overlaid osn top of the normal Unix file protections. For example: home directories would have read/write access for all known users, allowing each to access their own files. However they would still not have access to the files of others unless that permission would normally exist. Note that the netlogon share, typically distributed as a read only share from /etc/SAMBA/ netlogon, is the logon directory for user logon scripts.

SAMBA services are implemented as two daemons:

- smbd, which provides the file and printer sharing services and
- nmbd, which provides the NetBIOS-to-IP-address name service. NetBIOS over TCP/IP requires some method for mapping NetBIOS computer names to the IP addresses of a TCP/IP network.

4.3.1 History of SAMBA

SAMBA quietly evolved over the past 12 years from a barely functional prototype, used to communicate between a DOS Pathworks client and a Sun server, into a solid file and print server for Windows clients, maintained by a team of over 30 international developers, 12 of which are active at any one time.

SAMBA 2.0

After years of 1.x and in particular 1.9.x releases, SAMBA 2.0 brought new levels of protocol completeness to the SAMBA project and initial support for becoming a domain member.

SAMBA 2.2

SAMBA 2.2 brought the first implementation of a Domain Controller to SAMBA's stable series and provided a solid domain member platform with the introduction of the winbindd daemon in SAMBA 2.2.3.

SAMBA 3.0

With the introduction of SAMBA 3.0, SAMBA finally used the Unicode character representation when talking to network clients, solving many issues in non-English environments. SAMBA 3.0 also featured a vastly improved domain controller and support for being a client of Active Directory.

4.3.2 SAMBA as a DC

SAMBA 2.2 and in particular SAMBA 3.0 grew to include the ability to be an NT4 compatible domain controller, a functionality that even allows SAMBA to 'take over' an existing Windows network. This has allowed many sites to remove Windows servers entirely from their networks. Because SAMBA 3.0 implements the full requirements of an NT4 DC, it can be used to host some of the legacy parts of the protocol, not yet found in SAMBA4 – in particular, NetBIOS name registration and NETLOGON requests.

Patches have been proposed (and some already accepted) to allow this piece of SAMBA3 infrastructure to handle these roles, in the SAMBA framework.

4.3.3 SAMBA as a Active Directory Domain Member

SAMBA 3.0 releases has the ability to be a member of an Active Directory domain and as such has an implementation of a particular form of AD client. This client uses Kerberos for authentication and used DCE-RPC and LDAP to query user and group information from the DCs.

- **SAMBA 3.0 Active Directory DC Research**

As part of a internship project known as Blue Directory, students and supervisors at IBM's Linux Technology Center spent a lot of time researching the problem space around Active Directory, but as described by McDonough, they kept hitting up against limitations in the available technology. Their research work has been rapidly superseded by the SAMBA effort, but their input showed what would be possible with the proper infrastructure.

- **Heimdal Kerberos**

Heimdal is an Open Source implementation of the Kerberos protocol. Created outside the USA due to export controls on strong encryption, it has been developed independently of the well-known MIT distribution. The Heimdal source code is well tested and quite easy to modify. The presence of the HDB back-end interface is what made Heimdal the clear choice for this integration effort. Another aspect that makes Heimdal a key building block in this effort has been the active participation of key Heimdal developers in our branch of the Heimdal source.

- **HDB Back-end**

Within Heimdal, there is an abstraction layer that separates the password database from the rest of the Kerberos implementation. In the unmodified code, this allows the administrator to select between an LDAP back-end and a simple key-value database. It is this interface that this project will extend, with a new 'ldb' back-end to be provided.

- **Heimdal/SAMBA Integration**

Another feature of current Heimdal snapshots is support for integration with SAMBA 3.0. By using SAMBA 3.0's password entries in the LDAP database, Heimdal snapshots can use the SAMBA NT Password attribute as an arcfour-hmac-md5 Kerberos key. This integration work not only opened up valuable communication channels between SAMBA and Heimdal developers, it provided hands-on experience in hdb module development.

- **Clapd**

Clapd is a simple Connectionless LDAP daemon, written as part of the IBM research effort and designed to answer the basic requests that a Windows client makes over connectionless LDAP. At present, it is functional to the extent required for my domain join test, but has failed for others. It will need to be rewritten and properly integrated into the SAMBA system.

- **BIND**

An Active Directory domain is strongly based on a DNS domain, particularly due to the tight integration between Kerberos and DNS and the fact that this allows a move to a hierarchical name space. No modifications have been required to the BIND software and only the installation of configuration files is required. In the future, changes to BIND will be required to support Microsoft's dynamic DNS update scheme.

Check Your Progress 2

Note: a) Space is given below for writing your answer.

b) Compare your answer with the one given at the end of the Unit.

Explain SAMBA?

.....

.....

.....

.....

.....

.....

4.4 LET US SUM UP

This unit covers the detailed descriptions of the Active Directory Controller and SAMBA. An Active Directory structure is a hierarchical framework of objects. The objects fall into two broad categories: resources (e.g. printers) and security principals (user or computer accounts and groups). Security principals are Active Directory objects that are assigned unique security identifiers (SIDs) used to control access and set security. **SAMBA** is a free software re-implementation, originally developed by Andrew Tridgell, of the SMB/CIFS networking protocol. As of version 3, SAMBA provides file and print services for various Microsoft Windows clients and can integrate with a Windows Server domain, either as a Primary Domain Controller (PDC) or as a domain member. It can also be part of an Active Directory domain.

4.5 CHECK YOUR PROGRESS: THE KEY

Check Your Progress 1

1) The Active Directory structure and storage architecture consists of four parts:

   i) **Active Directory domains and forests:** Forests, domains and organizational units (OUs) make up the core elements of the Active Directory logical structure. A forest defines a single directory and represents a security boundary. Forests contain domains.

   ii) **Domain Name System (DNS) support for Active Directory:** DNS provides a name resolution service for domain controller location and a hierarchical design that Active Directory can use to provide a naming convention that can reflect organizational structure.

   iii) **Schema:** The schema provides object definitions that are used to create the objects that are stored in the directory.

   iv) **Data store:** The data store is the portion of the directory that manages the storage and retrieval of data on each domain controller.

2) Directory Information Tree (DIT) is broken into the following partitions:

   Schema partition – Defines rules for object creation and modification for all objects in the forest. Replicated to all domain controllers in the forest. Replicated to all domain controllers in the forest, it is known as an enterprise partition.

   Configuration partition – Information about the forest directory structure is defined including trees, domains, domain trust relationships and sites (TCP/IP subnet group). Replicated to all domain controllers in the forest, it is known as an enterprise partition.

   Domain partition – Has complete information about all domain objects (Objects that are part of the domain including OUs, groups, users and others). Replicated only to domain controllers in the same domain. Partial domain directory partition - Has a list of all objects in the directory with partial list of attributes for each object.

3) Active Directory is integrated with Domain Naming System (DNS) and requires it to be present to function. DNS is the naming system used for the Internet and on many Intranets. You can use DNS which is built into Windows 2000 and newer or use a third party DNS infrastructure such as BIND if you have it in the environment. It is recommended you use Window's DNS service as it

is integrated into Windows and provides the easiest functionality. AD uses DNS to name domains, computers, servers and locate services. A DNS server maps an object's name to its IP address. In an Active Directory network, it is used not only to find domain names, but also objects and their IP address. It also uses service location records (SRV) to locate services

4) Active Directory Application Mode (ADAM) is a new mode of Active Directory that is designed specifically for directory-enabled applications. ADAM is a Lightweight Directory Access Protocol (LDAP) directory service that runs as a user service, rather than as a system service. You can run ADAM on servers and domain controllers running operating systems in the Windows Server 2003 family (except for Windows Server 2003, Web Edition) and also on client computers running Windows XP Professional. ADAM does not require the deployment of domains or domain controllers. You can run multiple instances of ADAM concurrently on a single computer, with an independently managed schema and independently managed data for each ADAM instance.

Check Your Progress 2

SAMBA is software that can be run on a platform other than Microsoft Windows, for example, UNIX, Linux, IBM System 390, OpenVMS and other operating systems. SAMBA uses the TCP/IP protocol that is installed on the host server. When correctly configured, it allows that host to interact with a Microsoft Windows client or server as if it is a Windows file and print server. SAMBA is an Open Source/Free Software suite that provides seamless file and print services to SMB/ CIFS clients. SAMBA is freely available, unlike other SMB/CIFS implementations and allows for interoperability between Linux/Unix servers and Windows-based clients.

4.6 SUGGESTED READINGS

- http://en.wikipedia.org/wiki/Active_Directory.
- http://en.wikipedia.org/wiki/SAMBA_%28software%29.
- http://www.SAMBA.org/SAMBA/docs/SAMBAIntro.htmlwww.secure computing.com.

NOTE

MPDD-IGNOU/P.O. 1T/September, 2011

ISBN : 978-81-266-5568-7