UNIT 1 INTRODUCTION TO NETWORK ADMINISTRATION

Structure **Page Nos.**

1.0 INTRODUCTION

Computer network is a telecommunications network that connects a collection of computers to allow communication and data exchange between systems, software applications, and users. The computers that are involved in the network that originate, route and terminate the data are called nodes. The interconnection of computers is accomplished with a combination of cable or wireless media and networking hardware. Two devices are said to be networked when a process in one device is able to exchange information with a process in another device. Networks may be classified by various characteristics, such as the media used to transmit signals, the communications protocols used to organize network traffic, network scale, network topology and organizational scope. The best-known computer network is the Internet.

Communication protocols define the rules and data formats for exchanging information in a computer network. Well-known communications protocols include Ethernet, a hardware and link layer standard that is widely used for local area networks, and the Internet protocol suite (TCP/IP), which defines a set of protocols for communication between multiple networks, for host-to-host data transfer, and for application-specific data transmission formats. Protocols provide the basis for network programming.

1.1 OBJECTIVES

After going through this unit, you will be able to:

- know the roles and responsibilities of a Network Administrator;
- know about network client and its purpose;
- understand LINUX system initialization; and
- understand remote system administration and available tools.

1.2 ROLES AND RESPONSIBILITIES OF NETWORK ADMINISTRATOR

A Network Administrator is an individual, who is responsible for configuring, commissioning and maintenance of network infrastructure and services. It also includes the computer hardware and software systems that make up a data network. In an organization, Network Administrator generally don't typically get involved directly with users, instead focus upon configuring, monitoring and maintenance of network components within organization's LAN/WAN infrastructure. Depending on the organization and its size, the Network Administrator may also involve in design and deployment of computer networks.

Roles of a Network Administrator

The roles of a Network Administrator include activities and tasks to be performed such as configuring, commissioning and maintenance of various network devices-routers, switches, VPN gateways, security devices-Firewall and IDS/IPS , creation of De Militarized Zones (DMZ) , IP addresses allocation & management. It also includes configuring and commissioning of various network services/protocols-DHCP, DNS, FTP, HTTP, NFS, etc.

Apart from roles, the Network Administrator is also responsible to

- Ensure data network connectivity
- Network monitoring and management
- Testing the network for breaches, if any
- Keeping an eye out for needed updates
- Update Access Control Lists (ACLs) time to time to regulate network traffic
- Security controls enforcement
- Preparing and implementation of security policy and standards

1.3 LINUX AND TCP/IP INTERNETWORKING CONCEPTS

Linux is a Unix-like computer operating system assembled under the model of free and open source software development and distribution. Linux was originally developed as a free operating system for Intel x86-based personal computers. It is a leading operating system and supports different computer hardware platforms like other operating systems. Linux also runs on embedded systems (a devices where the operating system is typically built into the firmware and highly tailored to the system) such as mobile phones, tablet computers, network routers, building automation controls, televisions, video game consoles and. The Android system, which is in wide use on mobile devices is built on the Linux kernel.

Typically Linux is packaged in a format known as a Linux distribution for desktop and server use. Some popular mainstream Linux distributions include Debian (and its derivatives such as Ubuntu and Linux Mint), Red Hat Enterprise Linux (and its derivatives such as Fedora) , and openSUSE (and its commercial derivative SUSE Linux Enterprise Server).

Linux is the most popular network operating system (NOS) runs on a server and enables the server to manage data, users, groups, security, applications, and other networking functions. It runs based on a client/server architecture in which a server enables multiple clients to share resources. Linux allows shared file and printer access

among multiple computers in a network, typically a local area network (LAN), a private network or to other networks. Linux well supports to configure and commissioning of various network servers and services such as proxy servers, Domain name systems, Mail servers, Web servers, etc that are to be accessed through internet.

Internetworking with TCP/IP

Internetworking is the practice of connecting a computer network with other networks through the use of network gateways that provide a common method of routing data packets between the networks. The most notable example of internetworking is the Internet, which a network of networks based on many underlying hardware technologies, but unified by an internetworking protocol standard, referred to as the Internet Protocol Suite and known as TCP/IP.

TCP/IP (Transmission Control Protocol (TCP) and the Internet Protocol (IP)) is a networking model and provides end-to-end connectivity specifying how data should be formatted, addressed, transmitted, routed and received at the destination. It has four abstraction layers which are used to sort all related protocols according to the scope of networking involved.

The Figure 1 shows different networks connected to internet.

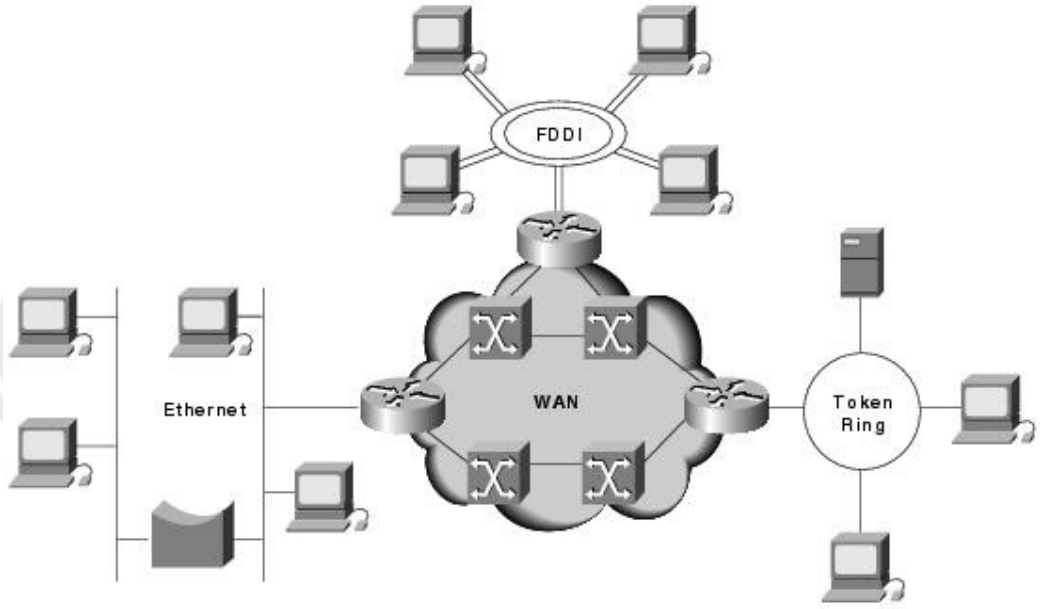

Figure 1: Different networks connected to Internet

How TCP/IP Works

TCP/IP for IP version 4 (IPv4) is a networking protocol suite that uses to communicate over the internet with other computers. It interacts with naming services like Domain Name System (DNS) and security technologies, such as IPsec for secure transfer of IP packets between computers.

Linux provides extensive support for the Transmission Control Protocol/Internet Protocol (TCP/IP) suite, as both a protocol and a set of services for connectivity and management of IP internetworks. Knowledge of the basic concepts of TCP/IP is an absolute requirement for the proper understanding of the configuration, deployment, and troubleshooting of IP-based Linux server.

TCP/IP Protocol Architecture

TCP/IP protocols map to a four-layer conceptual model. The four layers are Application, Transport, Internet, and Network Interface. Each layer corresponds to one or more layers of the seven-layer Open Systems Interconnection (OSI) model.

Figure 2 shows the TCP/IP protocol architecture.

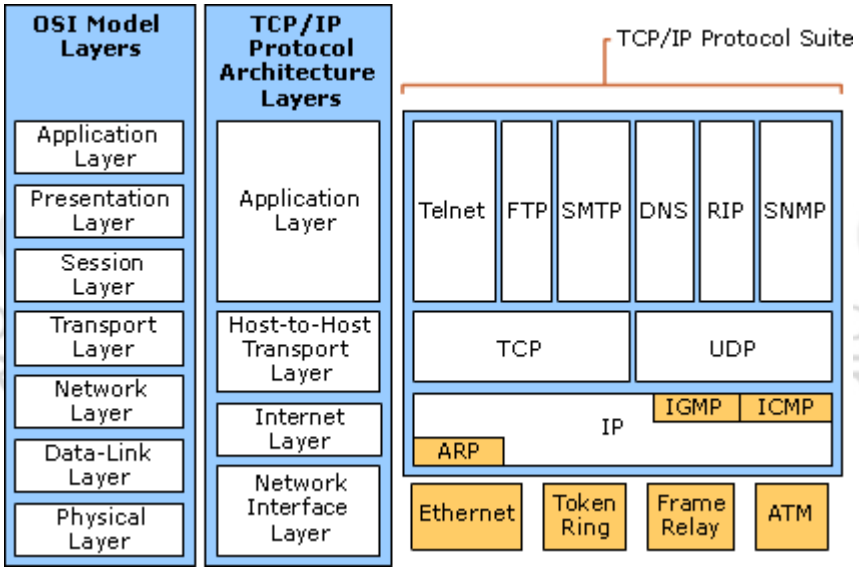

Figure 2: TCP/IP Protocol Architecture

Network Interface Layer

The Network Interface layer also called the Network Access layer that handles placing TCP/IP packets on the network medium and receiving TCP/IP packets off the network medium. TCP/IP was designed to be independent of the network access method, frame format, and medium. In this fashion, TCP/IP can be used to connect differing network types and these include local area network (LAN) media such as Ethernet and Token Ring and also WAN technologies such as X.25 and Frame Relay. The network interface layer encompasses the data link and physical layers of the OSI model.

Internet Layer

The Internet layer handles addressing, packaging, and routing functions. The core protocols of the Internet layer are IP, ARP, ICMP, and IGMP.

- The Internet Protocol (IP) is a routing protocol that handles IP addressing, routing, and the fragmentation and reassembly of packets.
- The Address Resolution Protocol (ARP) handles resolution of an Internet layer address to a Network Interface layer address, such as a hardware address.
- The Internet Control Message Protocol (ICMP) handles providing diagnostic functions and reporting errors due to the unsuccessful delivery of IP packets.
- The Internet Group Management Protocol (IGMP) handles management of IP multicast group membership.

The Internet layer is similar to the Network layer of the OSI model.

Transport Layer

The Transport layer handles and provides session and datagram communication services to Application layer. The core protocols of the Transport layer are Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP).

- TCP provides a one-to-one, connection-oriented, reliable communications service. TCP handles the establishment of a TCP connection, the sequencing and acknowledgment of packets sent, and the recovery of packets lost during transmission.
- UDP provides a one-to-one or one-to-many, connectionless, unreliable communications service. UDP is used when the amount of data to be transmitted is small (such as data that fits into a single packet), when you do not want the overhead of establishing a TCP connection, or when the applications or upper layer protocols provide reliable delivery.

The TCP/IP Transport layer is similar to the Transport layer of OSI model.

Application Layer

The application layer provides services for an application program to ensure that effective communication with another application program in a network is possible. The application layer is not the application itself that is doing the communication, but with various application layer protocols.

The most widely known Application layer protocols are those used for the exchange of user information:

- The Hypertext Transfer Protocol (HTTP) is used to transfer files that make up the Web pages of the World Wide Web.
- The File Transfer Protocol (FTP) is used for interactive file transfer.
- The Simple Mail Transfer Protocol (SMTP) is used for the transfer of mail messages and attachments.
- Telnet, a terminal emulation protocol, is used for logging on remotely to network hosts.

Additionally, the following Application layer protocols help facilitate the use and management of TCP/IP networks:

- The Domain Name System (DNS) is used to resolve a host name to an IP address.
- The Routing Information Protocol (RIP) is a routing protocol that routers use to exchange routing information on an IP internetwork.
- The Simple Network Management Protocol (SNMP) is used between a network management console and network devices (routers, bridges, intelligent hubs) to collect and exchange network management information.

The TCP/IP Application layer encompasses the responsibilities of the Session, Presentation, and Application layers of OSI model.

How Data Transmitted

Figure 3 shows, how data transmitted from source computer to destination computer.

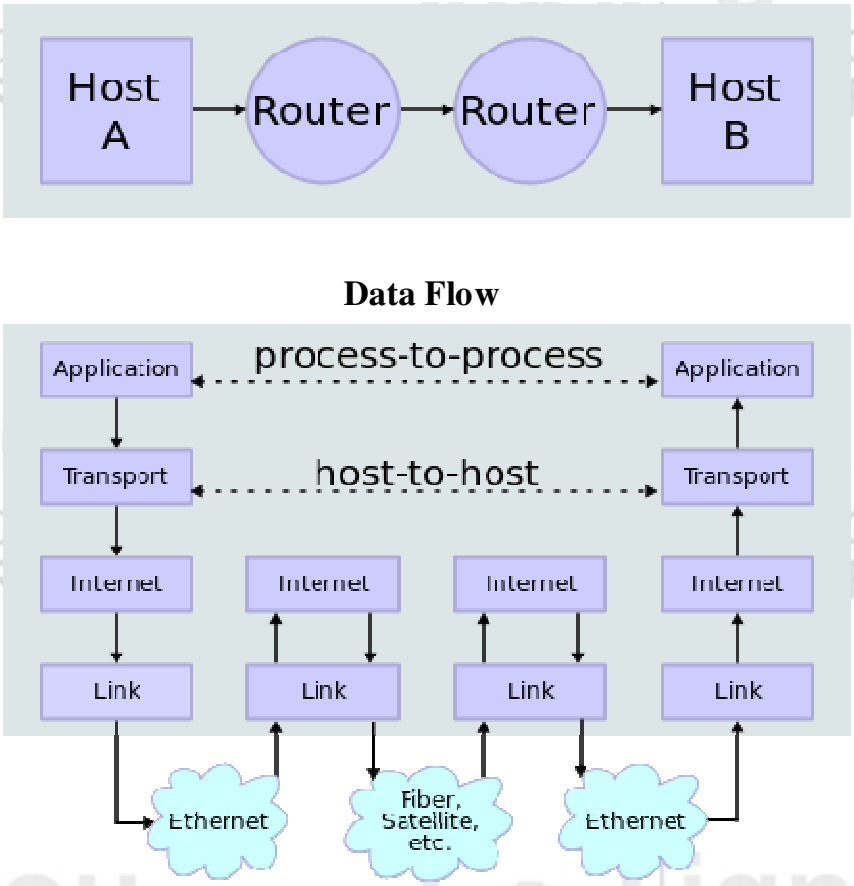

Figure 3: Data transmission between two Hosts

1.4 USING NETWORK CLIENTS

A client is a piece of computer hardware or software or both or a computer program that sends a service request to a server/computer program or accesses a service made available by a server/computer program. A network client is a client that can interact with servers/computer program through network/internet. The term client was first applied to devices that were not capable of running their own stand-alone programs, but could interact with remote computers via a network.

For example, web browsers are clients that connect to web servers and retrieve web pages for display. Email clients retrieve email from mail servers. Online chat uses a variety of clients, which vary depending on the chat protocol being used. Multiplayer video games or online video games may run as a client on each computer. The term "client" may also be applied to computers or devices that run the client software or users that use the client software. Similarly, the devices such as laptops, notebooks, palmtops, tablet PCs, smart phones, and other such devices also called network clients through which services requests can be sent or services can be retrieved to/from servers that are providing services. Figure 4 shows network clients that are being used in a cloud computing model.

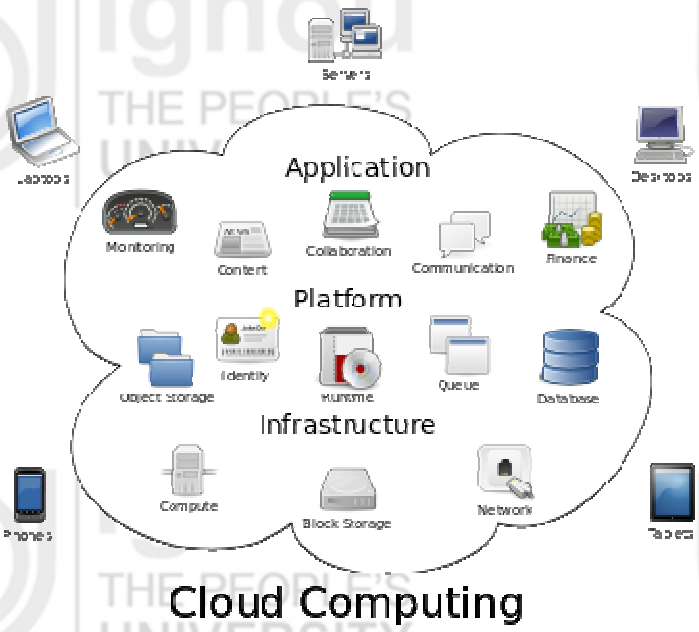

Figure 4: Network clients in a Cloud computing model

1.5 UNDERSTANDING SYSTEM INITIALIZATION

Here system means the combination of the computer hardware and the software (generally it the operating system installed on the computer hardware). Linux system (the computer installed with Linux operating system) initialization means how system gets stared after power is on. The Linux startup process is the process of Linux-operating system initialization. It is in many ways similar to the BSD and other Unix style boot processes, from which it derives.

In Linux, the flow of control during a boot is from BIOS (Basic Input/output System), to boot loader, to kernel. The kernel then starts the scheduler and runs the first program *Init* (which is mostly responsible to run startup scripts for each runlevel), at which point the kernel goes idle unless called externally.

init (short for initialization) is a program for Unix-based computer operating systems that spawns all other processes. It runs as a daemon and typically has PID 1. The *boot loader* starts the kernel and the kernel starts *init*. If some one has to delete *init* without a replacement, the system would encounter a kernel panic on the next reboot.

Overall system initialization process

i) The *BIOS* performs hardware-platform specific startup tasks

ii) Once the hardware is recognized and started correctly, the BIOS loads and executes the partition boot code from the designated boot device, which contains phase 1 of a Linux boot loader. Phase 1 loads phase 2 (the bulk of the boot loader code). Some loaders may use an intermediate phase (known as phase 1.5) to achieve this since modern large disks may not be fully readable without further code.

iii) The *boot loader* often presents the user with a menu of possible boot options. It then loads the operating system, which decompresses into memory, and sets up system functions such as essential hardware and memory paging, before calling 'start_kernel()'.

'start_kernel()' then performs the majority of system setup (interrupts, the rest of memory management, device initialization, drivers, etc.) before spawning separately, the idle process and scheduler, and the Init process (which is executed in user space).

iv) The Init process executes scripts as needed that set up all non-operating system services and structures in order to allow a user environment to be created, and then presents the user with a login screen.

The standard sequence for initializing a Linux system is as follows:

- Power on the System
- Initializing the BIOS
- Bootloader
- Kernel initialization
- Starting from "init"

Initializing the BIOS

- The BIOS (Basic Input/Output System) is the interface between the hardware and software at a very basic level, it provides all the basic instructions used by the operating system.
- The BIOS begins by executing an auto-ignition test (POST), and then it searches for devices.
- After the POST, a boot device is selected from a list that is configurable in the BIOS.
- The BIOS reads and executes the first physical sector of the boot media selected on the system, which is usually contained in the first 512 bytes of hard disk.

Bootloader

The bootloader is usually contained in the first sector of the disk and then read and executed by the BIOS. The storage space that reads the BIOS is not sufficient to contain all the bootloader, but just a part sufficient enough to start the rest of the bootloader, which is usually contained in a configuration file stored elsewhere on the disc. Hence the start is done in two steps:

- Launch via BIOS
- Launch a file under boot

The bootloader is designed to load and run the system kernel. The standard bootloader is GRUB but can also shift to LILO.

Kernel initialization

The kernel in Linux handles all operating system processes, such as memory management, task scheduling, I/O, interprocess communication, and overall system control. This is loaded in two stages - in the first stage the kernel is loaded into memory and decompressed, and a few fundamental functions such as basic memory management are set up. Control is then switched to the main kernel start process. Once the kernel is fully operational, it looks for an init process to run, which sets up a user space and the processes needed for a user environment and ultimate login. The kernel itself is then allowed to go idle, subject to calls from other processes.

The kernel initialization includes:

- The detection and initialization of devices. It means any device drivers compiled into the kernel are called and try to locate their corresponding devices.
- Mounting the root file system in read-only mode
- Loading the initial process "init"

The kernel initialization is very rapid and therefore it is very difficult to follow visually. One can read system generated log file to check what happened during kernel initialization. Generally log file can be stored under */var/log/dmesg*

Initialize "init"

Init (initialization) is the father of all processes. Its primary role is to create processes from a script stored in the file */etc/inittab*. This file usually has entries which cause *init* to spawn gettys on each line that users can log in. It also controls autonomous processes required by any particular system.

- "**init"** is the main process, it will always have a **PID** value: **1**.
- **"init"** reads its configuration from the */etc/inittab* file, that contains the settings for the system at every level of execution.

Run Levels

A run level is a software configuration of the system which allows only a selected group of processes to exist. The processes spawned by *init* for each of these run levels are defined in the /etc/inittab file.

"init" defines the following run levels:

- **Level 0:** Stop (not to be attributed to the initdefault)
- **Level 1, S:** single user mode (only the root user can log). Typically used for maintenance.
- **Level 2:** Multi-user mode without NFS network
- **Level 3:** full mode for multiple users including network
- **Level 4:** User Configurable duplicate but the level 3 by default.
- **Level 5:** X11 (including network)
- **Level 6:** Restart

Runlevel (System V)

The ability to change runlevel offers easy interaction with administrators; this allows to switch between different levels of startup.

Scripts services are in */etc/rc.d/init.d*. Each runlevel correspond to a */etc/rc.d/rcX.d* directory, where **X** is the runlevel.

System Shutdown

On shutdown, *Init* is called to close down all user space functionality in a controlled manner, again via scripted directions, following which Init terminates and the Kernel executes its own shutdown.

To stop the system, use commands like:

#Shutdown -h now

#halt

#poweroff

#init 0

Check Your Progress 1

1. Write the main responsibilities of a Network Administrator.

   ……

   ……

   ……

   ……

   ……

2. What are the various run levels of Linux system?

   ……

   ……

   ……

   ……

   ……

1.6 USER REMOTE ADMINISTRATION SERVICES AND TOOLS

Remote administration is an approach being followed to control either a computer system or a network or an application or all three from a remote location. A remote location may refer to a computer in the next room or one on the other side of the world. Generally, remote administration is essentially adopted when it is difficult or impractical to a person to be physically present and do administration on a system's terminal.

Requirements to perform Remote Administration

Network connectivity is essentially needed to perform remote administration. The network could be either Local Area Network (LAN) connectivity or internet connectivity depending on remote location. It means, any computer with an internet connection or on a LAN can be remotely administered.

For non-malicious administration, the user must install or enable remote administration software/tool on the host system then the user/client can access the host system from another computer using the remote tool.

Common Services for which remote administration is used

Generally, remote administration is needed for user management, file system management, software installation/configuration, network management- Network Security/Firewalls, VPN, Infrastructure Design, Network File Servers, Auto-mounting etc. and kernel optimization/ recompilation.

The following are some of the tasks/ services for which remote administration need to be done:

General

Controlling one's own computer from a remote location (e.g. to access the software on a personal computer from an internet café).

ICT Infrastructure Management

Remote administration essentially needed to administer the ICT infrastructure such as the servers, the routing and switching components, the security devices and other such related.

Shutdown

- Shutting down or rebooting a computer over a network

Accessing Peripherals

- Using a network device, like printer
- Retrieving streaming data, much like a CCTV system

Modifying

- Editing another computer's registry settings
- Modifying system services
- Installing software on another machine
- Modifying logical groups

Viewing

- Remotely assisting others
- Supervising computer or internet usage
- Access to a remote system's "Computer Management" snap-in

Hacking

Computers infected with malware such as Trojans sometimes open back doors into computer systems which allow malicious users to hack into and control the computer. Such users may then add, delete, modify or execute files on the computer to their own ends.

Remote Desktop Solutions for Linux

Most people who are used to a Unix-style environment know that a machine can be reached over the network at the shell level using utilities like telnet or ssh. And some people realize that X Windows output can be redirected back to the client workstation. But many people don't realize that it is easy to use an entire desktop over the network. The following are some of proprietary and open source applications that can be used to achieve this.

SSH (Secure Shell)

Secure Shell (SSH) is a proprietary cryptographic network tool for secure data communication between two networked computers that connects, via a secure channel over an insecure network, a server and a client (running SSH server and SSH client programs, respectively). The protocol specification distinguishes between two major versions that are referred to as SSH-1 and SSH-2.

The best-known application of the tool is for access to shell accounts on Unix-like operating systems-GNU/Linux, OpenBSD, FreeBSD, but it can also be used in a similar fashion for accounts on Windows.

SSH is generally used to log into a remote machine and execute commands. It also supports tunneling, forwarding TCP ports and X11 connections, it can transfer files using the associated SSH file transfer (SFTP) or secure copy (SCP) protocols. SSH uses the client-server model.

SSH is important in cloud computing to solve connectivity problems, avoiding the security issues of exposing a cloud-based virtual machine directly on the Internet. An SSH tunnel can provide a secure path over the Internet, through a firewall to a virtual machine

OpenSSH (OpenBSD Secure Shell)

OpenSSH is a tool providing encrypted communication sessions over a computer network using the SSH protocol. It was created as an open source alternative to the proprietary Secure Shell software suite offered by SSH Communications Security.

Telnet

Telnet is used to connect a remote computer over network. It provides a bidirectional interactive text-oriented communication facility using a virtual terminal connection on internet or local area networks. Telnet provides a command-line interface on a remote host. Most network equipment and operating systems with a TCP/IP stack support a Telnet service for remote configuration (including systems based on Windows NT). Telnet is used to establish a connection to Transmission Control Protocol (TCP) on port number 23, where a Telnet server application (telnetd) is listening.

Experts in computer security, recommend that the use of Telnet for remote logins should be discontinued under all normal circumstances, for the following reasons:

- Telnet, by default, does not encrypt any data sent over the connection (including passwords), and so it is often practical to eavesdrop on the communications and use the password later for malicious purposes; anybody who has access to a router, switch, hub or gateway located on the network between the two hosts where Telnet is being used can intercept the packets passing by and obtain login, password and whatever else is typed with a packet analyzer.
- Most implementations of Telnet have no authentication that would ensure communication is carried out between the two desired hosts and not intercepted in the middle.
- Several vulnerabilities have been discovered over the years in commonly used Telnet daemons.

rlogin

rlogin is an utility for Unix-like computer operating systems that allows users to log in on another host remotely through network, communicating through TCP port 513.

rlogin has several serious security problem- all information, including passwords is transmitted in unencrypted mode. rlogin is vulnerable to interception. Due to serious security problems, rlogin was rarely used across distrusted networks (like the public internet) and even in closed networks.

rsh

The remote shell (rsh) can connect a remote host across a computer network. The remote system to which rsh connects runs the rsh daemon (rshd). The daemon typically uses the well-known Transmission Control Protocol (TCP) port number 514. In security point of view, it is not recommended.

PuTTY

PuTTY is a free and open source terminal emulator application which can act as a client for the SSH, Telnet, rlogin, and raw TCP computing protocols and as a serial console client. The name "PuTTY" has no definitive meaning, though "tty" is the name for a terminal in the Unix tradition, usually held to be short for Teletype. PuTTY was originally written for Microsoft Windows, but it has been ported to various other operating systems as well

VNC (Virtual Network Computing)

VNC is a remote display system which allows the user to view the desktop of a remote machine anywhere on the internet. It can also be directed through SSH for security

Install VNC server on a computer (server) and install client on local PC. Setup is extremely easy and server is very stable. On client side, set the resolution and connect to IP of VNC server.

One can download VNC software from the following URLs:

http://www.realvnc.com/download.html

http://www.tightvnc.com/download.html

http://www.uvnc.com/

FreeNX allows to access desktop from another computer over the internet. One can use this to login graphically to a desktop from a remote location. One example of its use would be to have a FreeNX server set up on home computer, and graphically logging in to the home computer from work computer, using a FreeNX client. One can download FreeNX software from the following URLs:

https://help.ubuntu.com/community/FreeNX

http://ubuntuforums.org/showthread.php?t=97277&highlight=freenx

http://freenx.berlios.de/ (FreeNX homepage)

Wireless Remote Administration

Remote administration software has recently started to appear on wireless devices such as the BlackBerry, Pocket PC, and Palm devices, as well as some mobile phones.

Generally these solutions do not provide the full remote access seen on software such as VNC or Terminal Services, but do allow administrators to perform a variety of tasks, such as rebooting computers, resetting passwords, and viewing system event logs, thus reducing or even eliminating the need for system administrators to carry a laptop or be within reach of the office.

AetherPal and Netop are some of the tools used for full wireless remote access and administration on Smartphone devices.

Disadvantages of Remote Administration

Remote administration has many disadvantages too apart from its advantages. The first and foremost disadvantage is the security. Generally, certain ports to be open at Server level to do remote administration. Due to open ports, the hackers/attackers takes advantage to compromise the system. It is advised that remote administration to be used only in emergency or essential situations only to do administration remotely. In normal situations, it is ideal to block the ports to avoid remote administration.

Check Your Progress 2

1. Why remote administration is needed? Explain.

   ……..

   ……..

   ……..

   ……..

   ……..

2. List the different network clients.

   ……..

   ……..

   ……..

   ……..

   ……..

3. What are the different remote administration tools?

   ……..

   ……..

   ……..

   ……..

   ……..

1.7 SUMMARY

In this unit, different roles and responsibilities of a network administrator are clearly explained The TCP/IP and its role in data transmission from source to destination is made clear. System initialization process and importance of remote administration also covered.

1.8 ANSWERS TO CHECK YOUR PROGRESS

Check Your Progress 1

1) Network Administrator is responsible to

- Ensure data network connectivity

- Network monitoring and management
- Testing the network for breaches, if any
- Keeping an eye out for needed updates
- Update Access Control Lists (ACLs) time to time to regulate network traffic
- Security controls enforcement
- Preparing and implementation of security policy and standards

2) The following are various run levels of a Linux system

Level 0: Stop (not to be attributed to the initdefault)

Level 1, S: single user mode (only the root user can log). Typically used for maintenance.

Level 2: Multi-user mode without NFS network

Level 3: full mode for multiple users including network

Level 4: User Configurable duplicate but the level 3 by default.

Level 5: X11 (including network)

Level 6: Restart

Check Your Progress 2

1) The need for remote administration is for

- User management
- ICT Infrastructure management
- Problem Diagnosis and Troubleshooting
- File system management
- Software installation/configuration
- Network Management- Network Security/Firewalls, VPN, Infrastructure Design, Network File Servers, Auto-mounting etc.

2) The different network clients are

Computers, Notebooks, Palmtops, Tablet PCs, Smart phones and other such related.

3) The following are different remote administration tools

SSH, Telnet, rsh, rlogin, openSSH, VNC, etc

1.9 FURTHER READINGS

1. Computer Networks by Andrew S Tanenbaum , Fifth Edition
2. SA2, Redhat System Administration I & II, Student Workbook
3. Cisco Certified Network Associate Study Guide, Seventh Edition by Todd Lammle
4. Redhat Enterprise Linux System Administration
5. http://en.wikipedia.org/wiki/Internetworking
6. http://en.wikipedia.org/wiki/Remote_administration