LECTURE – 20

SECTION -D

SHELL PROGRAMMING

INTRODUCTION

- System administrator
- Roles & responsibilities of a system administrator
- LINUX Operating System
- Applications of LINUX OS

THE SYSTEM ADMINISTRATOR

- System administration task is usually entrusted to a single person – the **system administrator,** also known as the **super user** or **root user.**
- The administrator has vast powers, having access to practically everything.
- The success and stability of any UNIX installation depends, in great measure, on the effectiveness of the system administrator.
- The job of system administration, involves the management of entire system – ranging from maintaining user accounts, security and managing disk space to perform backups.
- The burden is not overwhelming though because UNIX is more easily maintained and well documented than any other systems.

DEFINITION OF SYSTEM ADMINISTRATION

In the context of the OS service provisioning, system administration plays a pivotal role.

This is particularly the case when a system is accessed by multiple users. The primary task of a system administrator is to ensure that the following happens:

- The top management is assured of efficiency in utilization of the system's resources.
- The general user community gets the services which they are seeking.

BOOTING THE SYSTEM

- The **Boot Block** of UNIX file system, containing a small bootstrap program – often referred to as the **Master Boot Record.**
- The program is loaded into memory when the system is booted (hence the name).
- It may in turn load another program from the disk, but eventually it loads kernel into memory.
- However, the bootstrapping program is read in from the boot block of only one file system (the main one, called the root file system).

MAINTAINING USER ACCOUNTS: USER MANAGEMENT

- The term user in UNIX is not meant to be only a person; it can represent a project or an application as well.
- A group of users performing a similar functions may use the username to use the system.
- It thus quite common to have usernames like marketing, accounts etc. For the creation and maintenance of user accounts.
- UNIX provides three commands – **useradd** , **usermod** , **userdel.**
- When opening a user account you've to associate the user with a group.
- A group usually has more than one member with a different set of privileges. People working on a common project should be able to read one another's files, which is possible only if they belong to the same group.

LOGGING IN WITH USERNAME & PASSWORD

- Unix is security-conscious and can be used only by those persons who maintain an account with the computer system. The list of accounts is maintained separately in the computer.
- Because the system can be used by many users, someone has to be given charge of administration of the system. This person is known as the system administrator and he is the person who grant you the permission to use the system.
- He opens an account with a name for your use, and gives you a secret code that you have to enter when the system prompts you for it.
- If you are running UNIX on your desktop, then remember that you are the administrator of the machine.

LOGGING IN WITH USERNAME & PASSWORD

- The login prompt indicates that the terminal is available for someone to log in (i.e. .connect to the machine). This message also indicates that the previous user has logged out (i.e. finished her work and disconnected)
- Suppose you have an account named “kumar” , enter this string at the prompt. Then press *[Enter]* key after the string :

login: kumar *[Enter]*

Password: ****** *[Enter*]** *Entry not displayed*

- If you make mistake while typing, simply press [Enter] one or two times until the login prompt reappears on the screen. Be sure to terminate your responses with [Enter] to make system “see” the input you’ve entered.

- The string you entered at the prompt (login:) is known as variously as your login name, user id, or user name.
- The secret code you entered at the next prompt (Password:) is known as password.
- If you entered them incorrectly, the system flashes the following message:

```
Login incorrect
login:
Password:
```

- Note: As soon as you log in, a program called the shell starts to run at your terminal, and keeps running until you terminate the session. As discussed before, the shell is your system's *command interpreter*. It displays the prompt and accepts all your input from the keyboard.

Creating a user involves defining the following parameters:

- A user identification number (UID) and username.
- A group identification number (GID) and group name.
- The home directory.
- The login shell.
- The password

(1) ADDING A GROUP

Command used: **$ groupadd**

- **If the user is to be placed in a new group, an entry for the group has to be created first in /etc/group.**
- **A user always has one primary group and may also have one or more supplementary groups.**

- **This file contains all of the named groups of the system, and a few lines of this file reveal the structure:**

```
root : x : 0 : root
bin : x : 1 : root, bin ,daemon
lp : x : 7 :
uucp : x : 14 : uucp, fax, root, fnet, sumit
users : x : 100 : henry, oracle, image, enquiry
```

- **Each line contains four colon-delimited fields. Lets focus our attention on the group named users shown in the first filed.**
- **The second field once represented the group password but is hardly used today; it is either blank or an x.**
- **The third field shows the user's GID (here, 100). The last field contains a list of comma-delimited usernames (here, henry, oracle, image, enquiry)**

- Note that primary group for a user is shown in /etc/passwd.
- To create a new group , dba, with a GID of 241 you have to use the groupadd command:

```
$ groupadd –g 241 dba
```

241 is the GID for dba

The command places this entry in /etc/group which you can also insert manually.

```
dba : x : 241 :
```

(2) ADDING A USER

Command used : $ useradd

- The **useradd** command adds new users to the system. All parameters related to the user have to be provided in the command line itself:

```
$ useradd – u 210 – g dba – c “THE DBMS”  - d /home/ oracle – s /bin/ ksh – m
  oracle
```

- It quietly creates the user oracle with a UID of 210 and group name dba. The home directory is /home/ oracle, and the user will use the Korn shell.
- The – m option ensures that the home created if it doesn’t already exist and copies a sample .profile and .kshrc to the user’s home directory.
- The line useradd creates in /etc/passwd is shown below:

```
oracle : x : 210 : 241 : THE RDBMS :/home/oracle: / bin/ksh
```

usermod and userdel : Modifying and Removing Users:

- **usermod** is used for modifying some of the parameters set with **useradd.** Users sometimes need to change their login shell, and the following command line sets Bash as the login shell for the user oracle :

```
$ usermod – s /bin/bash oracle
```

- **Users are removed from the system with userdel. The following command removes the user oracle from the system :**

```
$ userdel oracle
```

Roles And Functions of System Administrator

This is not an exhaustive list, yet it represents most of the tasks which system administrators perform:

1. **System startup and shutdown:** the basic steps required to start and to stop operations in a Unix operational environment.
2. **Opening and closing user accounts:** In Unix an administrator is both a user and a super-user. Usually, an administrator has to switch to the super-user mode with root privileges to open or close user accounts.
3. **Helping users to set up their working environment:** Unix allows any user to customize his working environment.

4. **Maintaining user services:** Users require services for printing, mail Web access and chat. We shall deal with mail and chat, where we discuss .rc files and with print services, where we discuss device management and services. These services include spooling of print jobs, provisioning of print quota, etc.

5. **Allocating disk space and re-allocating quotas when the needs grow**: Usually there would be a default allocation. However, in some cases it may be imperative to enhance the allocation.

6. **Installing and maintaining software:** This may require installing software patches from time to time. Most OSs are released with some bugs still present. Often with usage these bugs are identified and patches released. Also, one may have some software installed which satisfies a few of the specialized needs of the user community. As a convention this is installed in the directory /usr/local/bin. The local is an indicator of the local (and therefore a non-standard) nature of software.

7. **Installing new devices and upgrading the configuration:** As a demand on a system grows, additional devices may need to be installed. The system administrator will have to edit configuration files to identify these devices.

8. **Provisioning the mail and internet services:** Users connected to any host shall seek Mail and internet Web access. In addition, almost every machine shall be a resource within a local area network. So for resource too the machine shall have an IP address. In most cases it would be accessible from other machine as well.

9 . **Ensuring security of the system:** The internet makes the task of system administration both interesting and challenging. The administrators need to keep a check on spoofing and misuse

10. Maintaining system logs and profiling the users: A system administrator is required to often determine the usage of resources. This is achieved by analysing system logs. The system logs also help to profile the users. In fact, user profiling helps in identifying security breaches as was explained in the module entitled OS and Security.

11. System accounting: This is usually of interest to the management. Also, it helps system administrators to tune up an operating system to meet the user requirements. This also involves maintaining and analyzing logs of the system operation.

12. Reconfiguring the kernel whenever required: Sometimes when new patches are installed or a new release of the OS is received, then it is imperative to compile the kernel. Linux users often need to do this as new releases and extensions become available.

OVERVIEW OF LINUX OPERATING SYSTEM

- **What is Linux?**

NOTE: you can refer this topic from your POS practical file as well. Its very well explained there.

Linux is, in simplest terms, an operating system. It is the software on a computer that enables applications and the computer operator to access the devices on the computer to perform desired functions. The operating system (OS) relays instructions from an application to, for instance, the computer's processor. The processor performs the instructed task, then sends the results back to the application via the operating system.

CONCEPTUAL STRUCTURE OF THE LINUX KERNEL

So, what does the Linux kernel do? Similar to other operating systems, the Linux kernel keeps track of files on the disk, starts programs and multiplexes processor and other hardware btween them to provide multi-tasking, assigns memory and other resources to various processes, receives packets from and sends packets to the network. The Linux Operating System has two major parts:

1. Shell, which reads the commands entered by the user and translates them in the kernel.
2. Kernel, which interacts directly with the system's hardware.

SPECIFICALLY, THE LINUX KERNEL IS COMPOSED OF :

- ***Process Manager:*** *The Process Manager performs the administration of processes. The process scheduler, specifically, manages the time-slice and priority requirements of Linux processes. It also controls the forking and parent-child relationship of processes. The process manager also takes advantage for hardware with multiple CPUs using symmetric processing support.*
- ***Memory Manager:*** *The memory manager implements virtual memory to allow execution of process(es) that requires more memory than the physical memory.*
- ***Virtual Filesystem:*** *The filesystem manager implements the hierarchical file system with file access control and directory control. One of the important features of Linux is its support for many different filesystems.*
- ***Abstract network services (sockets):*** *Linux and networking are terms that are closely related because Linux itself is often used to support networking requirements of organizations. Linux's networking implementation is modeled on 4.3 BSD such that it supports BSD sockets and a full range of TCP/IP networking.*

APPLICATIONS

- As well as those designed for general purpose use on desktops and servers, distributions may be specialized for different purposes including: computer architecture support, embedded systems, stability, security, localization to a specific region or language, targeting of specific user groups, support for real-time applications, or commitment to a given desktop environment. Furthermore, some distributions deliberately include only free software. Currently, over three hundred distributions are actively developed, with about a dozen distributions being most popular for general-purpose use.[64]
- Linux is a widely ported operating system kernel. The Linux kernel runs on a highly diverse range of computer architectures: in the hand-held ARM-based iPAQ and the mainframe IBM System z9, System z10; in devices ranging from mobile phones to supercomputers.[65] Specialized distributions exist for less mainstream architectures. The ELKS kernel fork can run on Intel 8086 or Intel 80286 16-bit microprocessors, while the µClinux kernel fork may run on systems without a memory management unit. The kernel also runs on architectures that were only ever intended to use a manufacturer-created operating system, such as Macintosh computers (with both PowerPC and Intel processors), PDAs, video game consoles, portable music players, and mobile phones.