MODULE - V

System Administration

MODULE 5

System Administration

Module Description

This module discusses about routine duties of system administrator like creating and managing user accounts, managing disk space etc. Apart from these responsibilities, a UNIX system can face various problems like file system might crash or a user may delete a supporting secure file. Hence, the system administrator should have a thorough knowledge of every component of UNIX system.

By the end of this module, an administrator can create and manage different accounts; he can create partitions in file system, can create new file system, mount and unmount the file system. He would be in a stage to handle different system issues as he is the super user.

Chapter Table of Contents

Chapter 5.1

Common Tasks

Aim

To understand Linux System Administration

Instructional Objectives

After completing this chapter, you should be able to:

- Identify common administrative tasks performed by the system administrator
- Explain how administrative files, configuration files, and log files can be identified
- List the roles of system administrators
- Demonstrate how to add and delete a user from the group
- Illustrate the process of creating and managing groups as a super user
- Illustrate the process of disabling user accounts

Learning Outcomes

At the end of this chapter, you are expected to:

- Summarise the responsibilities of system administrators
- Differentiate between configuration file and log files
- Handle a user account
- Create and manage groups as a super user
- Amend permission and ownership as a super user
- Explain how to identify the users form the same group
- Demonstrate on how to disable user account

5.1.1 Introduction

Performance of an UNIX installation depends on the effectiveness of system administrator. A user must know some basic administrative functions that he may require to perform anytime. The job of system administration involves managing user accounts, Changing Permissions and Ownerships, Creating and Managing Groups, Modifying Group Attributes, Temporary Disabling of User's Accounts. Let us discuss these roles of system administrator in detail.

5.1.2 Common Administrative Tasks

System administrator is the ultimate authority in UNIX system and this authority comes with several responsibilities. Following are different administrative tasks a system administrator needs to perform:

- Add, change, and delete users, software and hardware
- Carry out routine maintenance activities like backup files and restore them on user's request.
- Ensure system security.
- Monitor system usage with respect to memory space and CPU.
- Provide assistance to users as required.

5.1.3 Identifying Administrative Files, Configuration and Log Files

A system administrator can carry his responsibilities by using system administration shell or by using system administration tools or by using scripts provided by system administration. But discussion of these methods is out of scope of this course. But we will discuss about only those tools that can help us to write system administrative shell scripts. To do so, we need to know where these tools reside. Following figure 5.1 shows the directories that affect system administration:

Directory	Function
/etc	Administrative and operational commands, as well as password and group files reside here
/user/adm	Accounting directories
/usr/lib	Operational logs, commands
/usr/pub	Public directories
/usr/tmp	Temporary Directories

Most of the commands required for system administration reside in /etc.

Self-assessment Questions

1) Most of the commands required for system administration reside in /etc
   a) TRUE b) FALSE

2) Administrative and operational commands, as well as password and group files reside in_____.
   a) a/usr b) /etc
   c) /user/adm d) User/pub

3) Accounting directories reside in_____.
   a) a/usr b) /etc
   c) /user/adm d) User/pub

4) Operational logs reside in
   a) /usr b) /etc
   c) /usr/lib d) User/pub

5.1.4 Role of System Administrator

The system administrator has a special login as root. This account is created at the time of installation. The prompt that appears for root account is #. The prompt $ and % is used by unprivileged users.

The role of the Unix Administrator including the following responsibilities:

- Adding and removing users: In UNIX system, an administrator can create user accounts, he can delete user accounts. Even, the administrator can control the rights of users by using different commands.
- Adding and removing hardware/ software: Everything in UNIX is treated as a file. As we can add and delete files, UNIX administrator can add or remove different hardware and software.
- Performing backups.
- Monitoring the system to ensure correct operation.
- Troubleshooting.
- Documentation.
- Auditing security.
- Helping users.

5.1.5 Managing user accounts

Unix provides following three commands for user account management:

- **useradd,**
- **usermod**
- **userdel**

When we create a new user account, it must be also associated to a user group.

When a new user account is created, following parameters are defined:

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

- The mailbox in /var/mail

Most of these parameters are found in a single file **/etc/passwd**.

Add new group:

groupadd command is used to create a new group. Whenever a new group is created, its entry must be done in **/etc/group. A user is always associated with a primary group and he can also associate with other groups.** The primary group for a user is shown in /etc/passwd.

Suppose, you want to create a new group testgroup, with group ID (i.e. GID) 234, you can use groupadd command as following:

groupadd –g 234 testgroup

as the new group "testgroup" is created, an entry is made in /etc/group as following:

testgroup:x:234

Add new user:

After creating group, you can add user to this newly created group using useradd command.

useradd –u 345 –g testgroup –d /home/test –s /bin/ksh –m test

The above command would create a user with UID 345 and group name testgroup. The home directory /home/test and user will use a korn shell.
The –m option ensures that the home directory is created if it does not exist.

usermode command

usermod is used for modifying some of parameters set with **useradd** command.

The example below adds the user to the group named as "folks" also changes the shell to ksh. The syntax for the usermod command is

```
# usermod -aG folks -s /bin/ksh someuser
```

The flag that is used by useradd command is utilized by usermod command. The command in above example uses a new flag as –a. This flag tells usermod command to add the user to the group named as folks. If this flag is not specified, then the user someuser will be removed from special people and others group and added to the group named as folks.

Deleting Users and Groups

The groupdel and userdel commands can be used to delete a user or a group.

Deleting a Group

groupdel folks

When the groupdel folks command is used, all references of folks group from the/etc/group file.

grep -c folks /etc/group

Deleting a User

userdel -r someuser

The flag used in above command i.e. –r tells the userdel command to delete the user's home directory as well. This action is not implicit however it should be done if the user is deleted as per requirement and the contents of the home directory for that user are no more required.

Self-assessment Questions

5) The system administrator has a special login as _____
   a) User
   b) Root
   c) Superuser
   d) (b) and (c)

6) The prompt that appear for root account is
   a) $
   b) %
   c) #
   d) @

7) _____ is a command to add new group
   a) Groupadd
   b) Grupadd
   c) Addgrup
   d) Addgroup

8) The primary group for a user is shown in _____
   a) /etc/group
   b) /etc/passwd
   c) /etc
   d) None of these

9) _____ is a command to add new user
   a) Adduser
   b) Useradd
   c) Add_user
   d) None of these

10) The groupdel command will remove all references of the folks group from the/etc/group file
   a) TRUE
   b) FALSE

5.1.6 Changing Permissions and Ownerships

If you want to change permission/ownership of a file, you need to know the characteristics of that file. Using ls –l command, you can display the file attributes and ownership

For example:

$ls –l

-rwxrwxrwx 1 user 1 group1 100 May 1 10:40 filename

In the above output of ls command – the first column displays file types and permissions

eg:

drwxrwxrwx – represents directory

or

-rwxrwxrwx – represents file

As already discussed in Module 2, a file can have three types of permissions

r – read

w-write

x-execute

When a user creates a file, he becomes the owner of that file and get associated with one group.

chmod command

chmod command is used to change the file permission

chmod can be used to change the permissions of following arguments:

arguments	Values	Shortnames used for action
User category	user, group or others	u, g, o
Operation to be performed	assign or remove a permission	+ or -
Type of permission	read, write, execute	r, w, x

Let us try chmod command

Example

- Check the attributes of a file using ls command

$Is –l filename

-rwxrwxrwx 1 user1 group1 100 May 1 10:40 filename

- Now you can remove execute permission for user by using chmod command (hint: use u for user and – for removing the permission) as following:

$ chmod u-x filename

- Use ls command to see change in permission of user

$ls –l filename

-rw-rwxrwx 1 user1 group1 100 May 1 10:40 filename

Let us see another example: To remove execute permission from user, group and others also use following command:

$ chmod ugo-x filename

Verify the action by using the ls command.

$ ls -l

-rw-rw-rw- 1 user1 group1 100 May 1 10:40 filename

Absolute permissions:

Sometimes user want to set all nine permission bits explicitly. Here, chmod can use string of three octal numbers as following:

Read permission – 4 – (octal 100)

Write permission – 2 – (octal 010)

Execute permission – 1 – (octal 001)

The below Table shows different combinations with permissions:

Binary	Octal	Permission
000	0	---
001	1	--x
010	2	-w-
011	3	-wx
100	4	r--
101	5	r-x
110	6	rw-
111	7	rwx

For Example: if you want to assign all permissions (read, write, execute) to user, group and others the you can use the following command:

$ chmod 777 filename

$ ls –l

-rwxrwxrwx 1 user1 group1 100 May 1 10:40 filename

You can change ownership of a file by using chown command as following:

$ chown user2 filename

you can change group owner of a file using chgrp command as following:

$ chgrp group2 filename1

5.1.7 Creating and Managing Groups

A group holds a list of users. Group name and GID are the identifiers of a group. As we have discussed earlier, each user in UNIX belongs to at least one group. A group is identified by a group name and a group identification number (GID).

When a user account is created, as a username get associated with it, a group name is also associated with it.

Listing of group name to GID is available in **/etc/group**. This plain text file consists of four colon-delimited fields. The first field represents the group name, second field represents the encrypted password, GID is the third field and the comma-delimited list of members is the forth field.

A superuser is the root user and able to modify /etc/group using a text editor.

Groups help the system administrator to manage different types of users. Suppose you want to create a group of users who are working on a science project together and no one else should be able to read and modify their files.

Groups can also be used to restrict access to sensitive information or specially licensed applications to a particular set of users.

The Superuser

The system administrator has a special login account as root user which is also known as superuser. Its user ID is 0Every UNIX system comes with a special user in the /etc/passwd file with a UID of 0.

5.1.8 Modifying Group Attributes

Groupmod command is used to modify group attributes. This command is generally used to change group name or group ID (GID). To modify the group attributes you need to login as superuser. The below command will modify the both group name and group ID.

5.1.9 Temporary Disabling of User's Accounts

There are two methods to disable user account or prevent a user from being able to login:

1. you can lock the user by editing **/etc/passwd** file
2. by directly issuing the **passwd** command with the **-l** switch

In the second case the user can login using another authentication token (e.g. an SSH key).

Method #1:

User need to login as root. In this method you need to edit /etc/passwd. Open /etc/passwd in vi editor by using the following command:

 # vi /etc/passwd

Press Esc i to go in insert mode.

Change the file: find /bin/bash with /bin/nologin

Following is the line before editing:

```
root:x:0:0:root:/root:/bin/bash
```

following is the line after editing:

```
root:x:0:0:root:/root:/bin/nologin
```

now close the vi editor

Self-assessment Questions

11) In the output of ls command – the first column values drwxrwxrwx – represents _____.

a) Directory
b) Root
c) File
d) User

12) a file can have three types of permissions

a) Read, write, open
b) Read, write, execute
c) Read, write, append
d) All of these

13) command used to change file access permissions

a) chmod
b) chown
c) chdir
d) mkdir

14) $ chmod u-x filename

a) Add user execute permission
b) Remove users execute permission
c) Add user read permission
d) Remove users read permission

15) Digital value assigned for read permission is

a) 1
b) 2
c) 3
d) 4

16) User's Primary Group id is listed in which file, at the time of creation of the user

a) /etc/passwd
b) /etc/groups
c) /etc/login
d) /etc/profile

17) User id 0 is

a) An invalid user id
b) The id of the root user
c) The id of a user when the user's account is deleted
d) None of the above

Summary

- The job of system administration involves managing user accounts, Changing Permissions and Ownerships, Creating and Managing Groups, Modifying Group Attributes, Temporary Disabling of User's Accounts.
- System administrator is the ultimate authority in UNIX system and this authority comes with several responsibilities like Add, change, and delete users, software and hardware, Carry out routine maintenance activities like backup files and restore them on user's request, Ensure system security etc.
- The system administrator has a special login as root. This account is created at the time of installation. The prompt that appears for root account is #. The prompt $ and % is used by unprivileged users.
- Unix provides following three commands for user account management: useradd, usermod ,userdel
- When we create a new user account, it must be also associated to a user group.
- groupadd command is used to create a new group. Whenever a new group is created, its entry must be done in /etc/group. A user is always associated with a primary group and he can also associate with other groups. The primary group for a user is shown in /etc/passwd.
- After creating group, you can add user to this newly created group using useradd command.
- A user or group can be deleted with the groupdel and userdel commands.
- If you want to change permission/ownership of a file, you need to know the characteristics of that file. Using ls –l command, you can display the file attributes and ownership. chmod command is used to change the file permission
- A group consists of list of users. Group name and GID are identifiers for a group. As we discussed earlier, each user in UNIX belongs to at least one group. A group is identified by a group name and a group identification number (GID).

Terminal Questions

- List common administrative tasks performed by the system administrator.
- Explain how administrative files, configuration files, and log files can be identified.
- Demonstrate how to add and delete a user from the group as a system administrator.
- Illustrate the process of creating and managing groups as a super user with appropriate example.
- Illustrate the process of disabling user accounts with appropriate example

Answer Keys

Self-assessment Questions	
Question No.	Answer
1	a
2	b
3	c
4	c
5	d
6	c
7	a
8	b
9	b
10	a
11	a
12	b
13	a
14	b
15	d
16	a
17	b

Activity

Activity Type: Online/Offline **Duration: 30 Minutes**

Description:

1) Prepare a presentation with screenshots (15 slides) on managing user account.
2) Name five administrative functions that cannot be performed by a non-privileged user.

Bibliography

e-References

- This website was referred on 3rd May 2016, while developing content for common tasks in Unix http://wpollock.com/AUnix1/SysAdminTasks.htm
- This website was referred on 3rd May 2016, while developing content for common tasks in Unix https://www.washington.edu/R870/
- This website was referred on 3rd May 2016, while developing content for common tasks in Unix http://www.cyberciti.biz/faq/what-is-the-role-of-the-system-administrator/
- This website was referred on 3rd May 2016, while developing content for common tasks in Unix http://www.tutorialspoint.com/unix/unix-user-administration.htm
- This website was referred on 3rd May 2016, while developing content for common tasks in Unix https://www.freebsd.org/doc/handbook/users-synopsis.html

External Resources

- Maurice J. Bach, The Design of Unix Operating System, (2010) Pearson Education
- S. Prata, Advance UNIX, a Programmer's Guide, (2011), BPB Publications, and New Delhi,
- B.W. Kernighan & R. Pike, The UNIX Programming Environment, (2009) Prentice Hall of India.
- Jack Dent Tony Gaddis, Guide to UNIX Using LINUX, (2010) Vikas/ Thomson Pub. House Pvt. Ltd.

Video Links

Topic	Link
The Linux File System	https://www.youtube.com/watch?v=2qQTXp4rBEE
Directory structure of the UNIX file system	https://www.youtube.com/watch?v=PEmi550E7zw

Notes: