

**Operating Systems – Fall 2003
Laboratory Assignment Number 1
Assigned August 6, 2003
Due August 8, 2003
100 Points**

Name: **SOLUTION**

Instructions:

Answer the following questions in the space provided. For assistance you can use the Linux Documentation Project Website <http://www.tldp.org>. You may discuss with each other during the lab. However, the submitted assignment must be completely your own work.

Question 1:

Using the Linux help system of manual pages invoked using the man command find out the function of the following Linux commands:

<i>Command</i>	<i>Function</i>	<i>Equivalent Windows NT/XP Command</i>	<i>Location of Executable File (In Linux)</i>
ps	ps stands for “process status”. This command is used to list the processes currently running on the system.	Right click on the Task Bar and choose Task Manager. Task Manager utility application allows you to list processes running on a Windows NT/XP system.	/bin/ps
kill	The kill command allows you terminate a running process on a Linux/UNIX system. It also allows you to send a signal to the process. Signals can include those to terminate a process, those to stop and restart a process with the objective that it re-reads its startup configuration which may have changed.	Right click on the Task Bar and choose Task Manager. Task Manager utility application allows you to list the programs and processes running on a Windows NT/XP system.	/bin/kill

SOLUTION

<i>Command</i>	<i>Function</i>	<i>Equivalent Windows NT/XP Command</i>	<i>Location of Executable File (In Linux)</i>
echo	echo is not a separate command but rather a built-in shell command. echo simply prints the string provided to it as an argument. For example “echo SHELL” will print SHELL on a line. However, “echo \$SHELL” will print the value of the SHELL environment variable.	echo	/bin/bash
free	free prints the amount of free memory on the system.	mem command or Task Manager utility application can be used in Windows NT/XP	/usr/bin/free
cat	cat simply prints a file on screen. It is a very basic pager utility. It can also be used to create text files by a clever use of input redirection. More advanced pagers on a modern UNIX system include “more” and “less”.	type	/bin/cat
chmod	chmod is used to change the permissions on a file.	Right clicking on a file and going into Properties option in the context menu allows one to access file security which can be used to set permissions.	/bin/chmod
touch	touch is used to update the last access time of a file. In case the file does not exist touch creates a zero size file by that name.	None.	/bin/touch

SOLUTION

<i>Command</i>	<i>Function</i>	<i>Equivalent Windows NT/XP Command</i>	<i>Location of Executable File (In Linux)</i>
ls	ls is used to perform a directory listing.	dir	/bin/ls
bash	bash is the command interpreter or shell most commonly used on Linux systems. Bash stands for “Bourne Again Shell”. It aims to be compatible with an earlier traditional UNIX shell called the Bourne Shell which was installed as /bin/sh.	cmd	/bin/bash
cp	cp is used to copy a file.	copy	/bin/cp

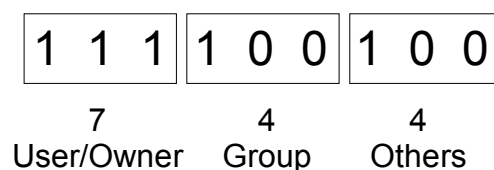
Question 2:

What types of file permissions are supported in Linux? How are they specified?

Linux supports three file permissions which are Read (r), Write (w) and Execute (x). These permissions are supported for the following three entities: User (u), Group members (g) or Others (o). “user” means the owner of the file and “others” means everyone else.

The chmod command can be used to change the permissions. For example if a file named /bin/ls needs to have the User i.e., the file owner to be able to read, write and execute then the chmod command would be “chmod u+rwx /bin/ls”.

Alternatively the permissions can be specified in a numerical format where each of the three entities' permission is represented as a three bit value encoded in decimal. For example if the owner is to be able to read, write and execute a file and every one else is to be able to only read the file then the permission bits are 111100100 = 744. This is shown in the figure below:



Questions 3:

What can file owner and everyone do with a file that has permissions set to:

544: file owner can do 101 = read and execute; everyone can do 100 = read

700: file owner can do 111 = read, write and execute; everyone can do 000 = no access

666: file owner can do 110 = read and write; everyone can do 110 = read and write

711: file owner can do 111 = read, write and execute; everyone can do 001 = execute

555: file owner can do 101 = read and execute; everyone can do 000 = no access

Question 4:

What is the difference between file permissions under UNIX/Linux and access control lists under Windows NT/2000/XP?

The types of access controls provided by file permissions are very useful. However they are also limited in capability because only three entities can be specified as subjects and only three types of permissions can be defined for each of these entities. In the system of access control list (ACLs) which is supported under Windows NT/XP individual computer users/group and domain users/groups can be given or denied rights of access and the types of access rights are also more fine grained than the simple three levels of read, write and execute supported under traditional UNIX/Linux permissions.

It may also be noted that UNIX/Linux systems also support ACLs as an add-on software which can be downloaded from the Internet and installed. More information about such a package is available at <http://acl.bestbits.at/>.

Question 5:

How do you make a file into a hidden file under Linux? How do you do the same under Windows NT/2000/XP?

If a file name begins with a dot character, i.e., "." then the file is considered a hidden file in Linux. Hidden files do not show up in ordinary file listings done using the ls command. However, using the option "ls -a" hidden files will also be listed.

Question 6:

Where is the Linux Kernel file located? How do you check the version of the Linux Kernel?

It is located in /boot/vmlinux.

SOLUTION

One way to check the kernel version would be to look at the directory listing for the /boot/vmlinuz file and see what file it points to. Generally, the name of the actual kernel file would contain embedded version information.

The second but sure shot way would be to run the command “cat /proc/version”. The output would be the result of a query to running kernel about its version number.

Question 7:

What is a bootloader? Name at least two bootloaders that can be used with Linux?

Bootloader is a small program that is run by the BIOS after it has performed the initial system checks. The Bootloader is responsible for loading the operating system from the hard disk. Two of the bootloaders commonly used with Linux are (1) LILO and (2) GRUB.

Question 8:

Read the manual page for the init command and then list the run levels of operation of the system along with the description of each run level. Why should there be different run levels? Is there something comparable in Windows NT/2000/XP?

The following information is taken from:

<http://www.redhat.com/docs/manuals/linux/RHL-7.2-Manual/ref-guide/s1-init-boot-shutdown-init.html>.

The idea behind operating different services at different runlevels essentially revolves around the fact that different systems can be used in a different ways. Some services cannot be used until the system is in a particular state, or mode, such as ready for more than one user or has networking available.

There are times in which you may want to operate the system at a lower mode, such as fixing disk corruption problems in runlevel 1 so no other users can possibly be on the system or leaving a server in runlevel 3 without an X session running. In these cases, running services that depend upon a higher system mode to function does not make sense because they will not work correctly anyway. By already having each service assigned to start when its particular runlevel is reached, you ensure an orderly start up process and can quickly change the mode of the machine without worrying about which services to manually start or stop.

Generally, Red Hat Linux operates in runlevel 3 — full multi-user mode. The following runlevels are defined in Red Hat Linux:

- 0 — Halt
- 1 — Single-user mode
- 2 — Not used (user-definable)

- 3 — Full multi-user mode
- 4 — Not used (user-definable)
- 5 — Full multi-user mode (with an X-based login screen)
- 6 — Reboot

The default runlevel for a system to boot to and stop is configured in `/etc/inittab`.

In summary, different run levels are useful during situations when some diagnostic or troubleshooting needs to be done. Run levels allow control in what software/programs the system can run. In Windows NT/XP there are also special modes of operation for diagnostic purposes which can be chosen at startup.

Question 9:

Read the file `/etc/passwd` and then comment on its structure. Show the generic structure.

The `/etc/passwd` file contains information about system's user accounts. It is a text file with information about each user beginning on a new line. Information about a user is organized in fields which are separated by colons (:). Each entry consists of the following fields:

```
username:passwd:userid:groupid:full name:home directory:shell:plan
```

Question 10:

What algorithm(s) is(are) used by Linux to store a user's password securely? List them again in the order of security strength.

Traditionally Linux/UNIX has stored users password entries in encrypted form using a variant of the DES algorithm/program called CRYPT. The encrypted passwords were traditionally stored in the `/etc/passwd` file which was world-readable. To enhance the security of passwords they were moved to the file `/etc/shadow` which was only readable by the root user. To make the passwords further secure, instead of storing the encrypted password, the MD5 hash of the password is stored in the `/etc/shadow` file. The listing of password storage methods according to security strength is:

1. MD5 – most secure
2. CRYPT – less secure