

1. How are devices represented in UNIX?

All devices are represented by files called special files that are located in/dev directory. Thus, device files and other files are named and accessed in the same way. A 'regular file' is just an ordinary data file in the disk. A 'block special file' represents a device with characteristics similar to a disk (data transfer in terms of blocks). A 'character special file' represents a device with characteristics similar to a keyboard (data transfer is by stream of bits in sequential order).

2. What is 'inode'?

All UNIX files have its description stored in a structure called 'inode'. The inode contains info about the file-size, its location, time of last access, time of last modification, permission and so on. Directories are also represented as files and have an associated inode. In addition to descriptions about the file, the inode contains pointers to the data blocks of the file. If the file is large, inode has indirect pointer to a block of pointers to additional data blocks (this further aggregates for larger files). A block is typically 8k.

Inode consists of the following fields:

- File owner identifier
- File type
- File access permissions
- File access times
- Number of links
- File size
- Location of the file data

3. Brief about the directory representation in UNIX

A Unix directory is a file containing a correspondence between filenames and inodes. A directory is a special file that the kernel maintains. Only kernel modifies directories, but processes can read directories. The contents of a directory are a list of filename and inode number pairs. When new directories are created, kernel makes two entries named '.' (refers to the directory itself) and '..' (refers to parent directory).

System call for creating directory is mkdir (pathname, mode).

4. What are the Unix system calls for I/O?

- open(pathname,flag,mode) - open file
- creat(pathname,mode) - create file
- close(filedes) - close an open file
- read(filedes,buffer,bytes) - read data from an open file
- write(filedes,buffer,bytes) - write data to an open file
- lseek(filedes,offset,from) - position an open file
- dup(filedes) - duplicate an existing file descriptor
- dup2(oldfd,newfd) - duplicate to a desired file descriptor
- fcntl(filedes,cmd,arg) - change properties of an open file
- ioctl(filedes,request,arg) - change the behaviour of an open file

The difference between fcntl and ioctl is that the former is intended for any open file, while the latter is for device-specific operations.

Naresh.surampudi

## 5. How do you change File Access Permissions?

Every file has following attributes:  
owner's user ID ( 16 bit integer )  
owner's group ID ( 16 bit integer )  
File access mode word

'r w x -r w x- r w x'

(user permission-group permission-others permission)

r-read, w-write, x-execute

To change the access mode, we use `chmod(filename,mode)`.

Example 1:

To change mode of myfile to 'rw-rw-r-' (ie. read, write permission for user - read,write permission for group - only read permission for others) we give the args as:

`chmod(myfile,0664)` .

Each operation is represented by discrete values

'r' is 4

'w' is 2

'x' is 1

Therefore, for 'rw' the value is 6(4+2).

Example 2:

To change mode of myfile to 'rwxr-r-' we give the args as:

`chmod(myfile,0744)`.

## 6. What are links and symbolic links in UNIX file system?

A link is a second name (not a file) for a file. Links can be used to assign more than one name to a file, but cannot be used to assign a directory more than one name or link filenames on different computers.

Symbolic link 'is' a file that only contains the name of another file. Operation on the symbolic link is directed to the file pointed by the it. Both the limitations of links are eliminated in symbolic links.

Commands for linking files are:

Link In `filename1 filename2`

Symbolic link In `-s filename1 filename2`

## 7. What is a FIFO?

FIFO are otherwise called as 'named pipes'. FIFO (first-in-first-out) is a special file which is said to be data transient. Once data is read from named pipe, it cannot be read again. Also, data can be read only in the order written. It is used in interprocess communication where a process writes to one end of the pipe (producer) and the other reads from the other end (consumer).

Naresh.surampudi

8. How do you create special files like named pipes and device files?

The system call `mknod` creates special files in the following sequence.

1. kernel assigns new inode,
2. sets the file type to indicate that the file is a pipe, directory or special file,
3. If it is a device file, it makes the other entries like major, minor device numbers.

For example:

If the device is a disk, major device number refers to the disk controller and minor device number is the disk.

9. Discuss the mount and unmount system calls

The privileged `mount` system call is used to attach a file system to a directory of another file system; the `umount` system call detaches a file system. When you mount another file system on to your directory, you are essentially splicing one directory tree onto a branch in another directory tree. The first argument to `mount` call is the mount point, that is, a directory in the current file naming system. The second argument is the file system to mount to that point. When you insert a cdrom to your unix system's drive, the file system in the cdrom automatically mounts to `/dev/cdrom` in your system.

10. How does the inode map to data block of a file?

Inode has 13 block addresses. The first 10 are direct block addresses of the first 10 data blocks in the file. The 11th address points to a one-level index block. The 12th address points to a two-level (double in-direction) index block. The 13th address points to a three-level (triple in-direction) index block. This provides a very large maximum file size with efficient access to large files, but also small files are accessed directly in one disk read.

11. What is a shell?

A shell is an interactive user interface to an operating system services that allows a user to enter commands as character strings or through a graphical user interface. The shell converts them to system calls to the OS or forks off a process to execute the command. System call results and other information from the OS are presented to the user through an interactive interface. Commonly used shells are `sh`, `csh`, `ks` etc.

12. Brief about the initial process sequence while the system boots up.

While booting, special process called the 'swapper' or 'scheduler' is created with Process-ID 0. The swapper manages memory allocation for processes and influences CPU allocation. The swapper inturn creates 3 children:

- the process dispatcher,
- `vhand` and
- `dbflush`

with IDs 1,2 and 3 respectively.

This is done by executing the file `/etc/init`. Process dispatcher gives birth to the shell. Unix keeps track of all the processes in an internal data structure called the Process Table (listing command is `ps -el`).

13. What are various IDs associated with a process?

Naresh.surampudi

Unix identifies each process with a unique integer called ProcessID. The process that executes the request for creation of a process is called the 'parent process' whose PID is 'Parent Process ID'. Every process is associated with a particular user called the 'owner' who has privileges over the process. The identification for the user is 'UserID'. Owner is the user who executes the process. Process also has 'Effective User ID' which determines the access privileges for accessing resources like files.

- getpid() -process id
- getppid() -parent process id
- getuid() -user id
- geteuid() -effective user id

14. Explain fork() system call.

The `fork()` used to create a new process from an existing process. The new process is called the child process, and the existing process is called the parent. We can tell which is which by checking the return value from `fork()`. The parent gets the child's pid returned to him, but the child gets 0 returned to him.

15. Predict the output of the following program code

```
main()
{
    fork();
    printf("Hello World!");
}
```

Answer:

Hello World!Hello World!

Explanation:

The fork creates a child that is a duplicate of the parent process. The child begins from the fork().All the statements after the call to fork() will be executed twice.(once by the parent process and other by child). The statement before fork() is executed only by the parent process.

16. Predict the output of the following program code

```
main()
{
    fork(); fork(); fork();
    printf("Hello World!");
}
```

Answer:

"Hello World" will be printed 8 times.

Explanation:

$2^n$  times where n is the number of calls to fork()

17. List the system calls used for process management:

Naresh.surampudi  
System calls Description

- fork() To create a new process
- exec() To execute a new program in a process
- wait() To wait until a created process completes its execution
- exit() To exit from a process execution
- getpid() To get a process identifier of the current process
- getppid() To get parent process identifier
- nice() To bias the existing priority of a process
- brk() To increase/decrease the data segment size of a process.

18. How can you get/set an environment variable from a program?

Getting the value of an environment variable is done by using ``getenv()'``. Setting the value of an environment variable is done by using ``putenv()'``.

19. How can a parent and child process communicate?

A parent and child can communicate through any of the normal inter-process communication schemes (pipes, sockets, message queues, shared memory), but also have some special ways to communicate that take advantage of their relationship as a parent and child. One of the most obvious is that the parent can get the exit status of the child.

20. What is a zombie?

When a program forks and the child finishes before the parent, the kernel still keeps some of its information about the child in case the parent might need it - for example, the parent may need to check the child's exit status. To be able to get this information, the parent calls ``wait()'``; In the interval between the child terminating and the parent calling ``wait()'``, the child is said to be a 'zombie' (If you do ``ps'`, the child will have a ``Z'` in its status field to indicate this.)

21. What are the process states in Unix?

As a process executes it changes state according to its circumstances. Unix processes have the following states:

Running : The process is either running or it is ready to run .

Waiting : The process is waiting for an event or for a resource.

Stopped : The process has been stopped, usually by receiving a signal.

Zombie : The process is dead but have not been removed from the process table.

How Linux boots :

As it turns out, there isn't much to the boot process:

1. A boot loader finds the kernel image on the disk, loads it into memory, and starts it.
2. The kernel initializes the devices and its drivers.
3. The kernel mounts the root filesystem.
4. The kernel starts a program called `init`.
5. `init` sets the rest of the processes in motion.
6. The last processes that `init` starts as part of the boot sequence allow you to log in.

Naresh.surampudi

Identifying each stage of the boot process is invaluable in fixing boot problems and understanding the system as a whole. To start, zero in on the boot loader, which is the initial screen or prompt you get after the computer does its power-on self-test, asking which operating system to run. After you make a choice, the boot loader runs the Linux kernel, handing control of the system to the kernel.

There is a detailed discussion of the kernel elsewhere in this book from which this article is excerpted. This article covers the kernel initialization stage, the stage when the kernel prints a bunch of messages about the hardware present on the system. The kernel starts `init` just after it displays a message proclaiming that the kernel has mounted the root filesystem:

```
VFS: Mounted root (ext2 filesystem) readonly.
```

Soon after, you will see a message about `init` starting, followed by system service startup messages, and finally you get a login prompt of some sort.

NOTE On Red Hat Linux, the `init` note is especially obvious, because it "welcomes" you to "Red Hat Linux." All messages thereafter show success or failure in brackets at the right-hand side of the screen.

Most of this chapter deals with `init`, because it is the part of the boot sequence where you have the most control.

### `init`

There is nothing special about `init`. It is a program just like any other on the Linux system, and you'll find it in `/sbin` along with other system binaries. The main purpose of `init` is to start and stop other programs in a particular sequence. All you have to know is how this sequence works.

There are a few different variations, but most Linux distributions use the System V style discussed here. Some distributions use a simpler version that resembles the BSD `init`, but you are unlikely to encounter this.

### Runlevels

At any given time on a Linux system, a certain base set of processes is running. This state of the machine is called its runlevel, and it is denoted with a number from 0 through 6. The system spends most of its time in a single runlevel. However, when you shut the machine down, `init` switches to a different runlevel in order to terminate the system services in an orderly fashion and to tell the kernel to stop. Yet another runlevel is for single-user mode, discussed later.

The easiest way to get a handle on runlevels is to examine the `init` configuration file, `/etc/inittab`. Look for a line like the following:

```
id:5:initdefault:
```

This line means that the default runlevel on the system is 5. All lines in the `inittab` file take this form, with four fields separated by colons occurring in the following order:

Naresh.surampudi

- # A unique identifier (a short string, such as id in the preceding example)
- The applicable runlevel number(s)
- The action that init should take (in the preceding example, the action is to set the default runlevel to 5)
- A command to execute (optional)

There is no command to execute in the preceding `initdefault` example because a command doesn't make sense in the context of setting the default runlevel. Look a little further down in `inittab`, until you see a line like this:

```
l5:5:wait:/etc/rc.d/rc 5
```

This line triggers most of the system configuration and services through the `rc*.d` and `init.d` directories. You can see that `init` is set to execute a command called `/etc/rc.d/rc 5` when in runlevel 5. The `wait` action tells when and how `init` runs the command: run `rc 5` once when entering runlevel 5, and then wait for this command to finish before doing anything else.

There are several different actions in addition to `initdefault` and `wait`, especially pertaining to power management, and the `inittab(5)` manual page tells you all about them. The ones that you're most likely to encounter are explained in the following sections.

`respawn`

The `respawn` action causes `init` to run the command that follows, and if the command finishes executing, to run it again. You're likely to see something similar to this line in your `inittab` file:

```
1:2345:respawn:/sbin/mingetty tty1
```

The `getty` programs provide login prompts. The preceding line is for the first virtual console (`/dev/tty1`), the one you see when you press `ALT-F1` or `CONTROL-ALT-F1`. The `respawn` action brings the login prompt back after you log out.

`ctrlaltdel`

The `ctrlaltdel` action controls what the system does when you press `CONTROL-ALT-DELETE` on a virtual console. On most systems, this is some sort of reboot command using the `shutdown` command.

`sysinit`

The `sysinit` action is the very first thing that `init` should run when it starts up, before entering any runlevels.

How processes in runlevels start

You are now ready to learn how `init` starts the system services, just before it lets you log in. Recall this `inittab` line from earlier:

```
l5:5:wait:/etc/rc.d/rc 5
```

Naresh.surampudi

This small line triggers many other programs. rc stands for run commands, and you will hear people refer to the commands as scripts, programs, or services. So, where are these commands, anyway?

For runlevel 5, in this example, the commands are probably either in /etc/rc.d/rc5.d or /etc/rc5.d. Runlevel 1 uses rc1.d, runlevel 2 uses rc2.d, and so on. You might find the following items in the rc5.d directory:

```
S10syslogd      S20ppp          S99gpm
S12kernel.d    S25netstd_nfs  S99httpd
S15netstd_init S30netstd_misc S99rmnologin
S18netbase     S45pcmcia      S99sshd
S20acct        S89atd
S20logoutd     S89cron
```

The rc 5 command starts programs in this runlevel directory by running the following commands:

```
S10syslogd start
S12kernel.d start
S15netstd_init start
S18netbase start
...
S99sshd start
```

Notice the start argument in each command. The S in a command name means that the command should run in start mode, and the number (00 through 99) determines where in the sequence rc starts the command.

The rc\*.d commands are usually shell scripts that start programs in /sbin or /usr/sbin. Normally, you can figure out what one of the commands actually does by looking at the script with less or another pager program.

You can start one of these services by hand. For example, if you want to start the httpd Web server program manually, run S99httpd start. Similarly, if you ever need to kill one of the services when the machine is on, you can run the command in the rc\*.d directory with the stop argument (S99httpd stop, for instance).

Some rc\*.d directories contain commands that start with K (for "kill," or stop mode). In this case, rc runs the command with the stop argument instead of start. You are most likely to encounter K commands in runlevels that shut the system down.

#### Adding and removing services

If you want to add, delete, or modify services in the rc\*.d directories, you need to take a closer look at the files inside. A long listing reveals a structure like this:

```
lrwxrwxrwx . . . S10syslogd -> ../init.d/syslogd
lrwxrwxrwx . . . S12kernel.d -> ../init.d/kernel.d
lrwxrwxrwx . . . S15netstd_init -> ../init.d/netstd_init
```



```
Naresh.surampudi
lrwxrwxrwx . . . S18netbase -> ../init.d/netbase
...
```

The commands in an rc\*.d directory are actually symbolic links to files in an init.d directory, usually in /etc or /etc/rc.d. Linux distributions contain these links so that they can use the same startup scripts for all runlevels. This convention is by no means a requirement, but it often makes organization a little easier.

To prevent one of the commands in the init.d directory from running in a particular runlevel, you might think of removing the symbolic link in the appropriate rc\*.d directory. This does work, but if you make a mistake and ever need to put the link back in place, you might have trouble remembering the exact name of the link. Therefore, you shouldn't remove links in the rc\*.d directories, but rather, add an underscore (\_) to the beginning of the link name like this:

```
mv S99httpd _S99httpd
```

At boot time, rc ignores \_S99httpd because it doesn't start with S or K. Furthermore, the original name is still obvious, and you have quick access to the command if you're in a pinch and need to start it by hand.

To add a service, you must create a script like the others in the init.d directory and then make a symbolic link in the correct rc\*.d directory. The easiest way to write a script is to examine the scripts already in init.d, make a copy of one that you understand, and modify the copy.

When adding a service, make sure that you choose an appropriate place in the boot sequence to start the service. If the service starts too soon, it may not work, due to a dependency on some other service. For non-essential services, most systems administrators prefer numbers in the 90s, after most of the services that came with the system.

Linux distributions usually come with a command to enable and disable services in the rc\*.d directories. For example, in Debian, the command is update-rc.d, and in Red Hat Linux, the command is chkconfig. Graphical user interfaces are also available. Using these programs helps keep the startup directories consistent and helps with upgrades.

**HINT:** One of the most common Linux installation problems is an improperly configured XFree86 server that flicks on and off, making the system unusable on console. To stop this behavior, boot into single-user mode and alter your runlevel or runlevel services. Look for something containing xdm, gdm, or kdm in your rc\*.d directories, or your /etc/inittab.

## Controlling init

Occasionally, you need to give init a little kick to tell it to switch runlevels, to re-read the inittab file, or just to shut down the system. Because init is always the first process on a system, its process ID is always 1.

Naresh.surampudi

You can control init with telinit. For example, if you want to switch to runlevel 3, use this command:

```
telinit 3
```

When switching runlevels, init tries to kill off any processes that aren't in the inittab file for the new runlevel. Therefore, you should be careful about changing runlevels.

When you need to add or remove respawning jobs or make any other change to the inittab file, you must tell init about the change and cause it to re-read the file. Some people use `kill -HUP 1` to tell init to do this. This traditional method works on most versions of Unix, as long as you type it correctly. However, you can also run this telinit command:

```
telinit q
```

You can also use `telinit s` to switch to single-user mode.

### Shutting down

init also controls how the system shuts down and reboots. The proper way to shut down a Linux machine is to use the shutdown command.

There are two basic ways to use shutdown. If you halt the system, it shuts the machine down and keeps it down. To make the machine halt immediately, use this command:

```
shutdown -h now
```

On most modern machines with reasonably recent versions of Linux, a halt cuts the power to the machine. You can also reboot the machine. For a reboot, use `-r` instead of `-h`.

The shutdown process takes several seconds. You should never reset or power off a machine during this stage.

In the preceding example, `now` is the time to shut down. This argument is mandatory, but there are many ways of specifying it. If you want the machine to go down sometime in the future, one way is to use `+n`, where `n` is the number of minutes shutdown should wait before doing its work. For other options, look at the `shutdown(8)` manual page.

To make the system reboot in 10 minutes, run this command:

```
shutdown -r +10
```

On Linux, shutdown notifies anyone logged on that the machine is going down, but it does little real work. If you specify a time other than `now`, shutdown creates a file called `/etc/nologin`. When this file is present, the system prohibits logins by anyone except the superuser.

When system shutdown time finally arrives, shutdown tells init to switch to runlevel 0 for a halt and runlevel 6 for a reboot. When init enters runlevel 0 or 6, all of the

Naresh.surampudi

following takes place, which you can verify by looking at the scripts inside rc0.d and rc6.d:

1. init kills every process that it can (as it would when switching to any other runlevel).

- The initial rc0.d/rc6.d commands run, locking system files into place and making other preparations for shutdown.
- The next rc0.d/rc6.d commands unmount all filesystems other than the root.
- Further rc0.d/rc6.d commands remount the root filesystem read-only.
- Still more rc0.d/rc6.d commands write all buffered data out to the filesystem with the sync program.
- The final rc0.d/rc6.d commands tell the kernel to reboot or stop with the reboot, halt, or poweroff program.

The reboot and halt programs behave differently for each runlevel, potentially causing confusion. By default, these programs call shutdown with the -r or -h options, but if the system is already at the halt or reboot runlevel, the programs tell the kernel to shut itself off immediately. If you really want to shut your machine down in a hurry (disregarding any possible damage from a disorderly shutdown), use the -f option.

1. Q. How do you list files in a directory?

A. ls - list directory contents  
ls -l (use a long listing format)

2. Q. How do you list all files in a directory, including the hidden files?

A. ls -a (-a, do not hide entries starting with .)

3. Q. How do you find out all processes that are currently running?

A. ps -f (-f does full-format listing.)

4. Q. How do you find out the processes that are currently running or a particular user?

A. ps -au Myname (-u by effective user ID (supports names)) (a - all users)

5. Q. How do you kill a process?

A. kill -9 8 (process\_id 8) or kill -9 %7 (job number 7)  
kill -9 -1 (Kill all processes you can kill.)  
killall - kill processes by name most (useful - killall java)

6. Q. What would you use to view contents of the file?

A. less filename  
cat filename  
pg filename  
pr filename  
more filename  
most useful is command: tail file\_name - you can see the end of the log file.

7. Q. What would you use to edit contents of the file?

A. vi screen editor or jedit, nedit or ex line editor

8. Q. What would you use to view contents of a large error log file?

A. tail -10 file\_name (last 10 rows)

Naresh.surampudi

9. Q. How do you log in to a remote Unix box?

A. Using telnet server\_name or ssh -l ( ssh - OpenSSH SSH client (remote login program))

10.Q. How do you get help on a UNIX terminal?

A. man command\_name  
info command\_name (more information)

11.Q. How do you list contents of a directory including all of its subdirectories, providing full details and sorted by modification time?

A. ls -lac  
-a all entries  
-c by time

12.Q. How do you create a symbolic link to a file (give some reasons of doing so)?

A. ln ../file1 Link\_name

Links create pointers to the actual files, without duplicating the contents of the files. That is, a link is a way of providing another name to the same file. There are two types of links to a file:Hard link, Symbolic (or soft) link;

13.Q. What is a filesystem?

A. Sum of all directories called file system.  
A file system is the primary means of file storage in UNIX.  
File systems are made of inodes and superblocks.

14.Q. How do you get its usage (a filesystem)?

A. By storing and manipulate files.

15.Q. How do you check the sizes of all users' home directories (one command)?

A. du -s  
df

The du command summarizes disk usage by directory. It recurses through all subdirectories and shows disk usage by each subdirectory with a final total at the end.

Q. in current directory

A. ls -ps (p- directory; s - size)

16.Q. How do you check for processes started by user 'pat'?

A. ps -fu pat (-f -full\_format u -user\_name )

17.Q. How do you start a job on background?

A. bg %4 (job 4)

18 Q. What utility would you use to replace a string '2001' for '2002' in a text file?

A. Grep, Kde( works on Linux and Unix)

19. Q. What utility would you use to cut off the first column in a text file?

A. awk, kde

Naresh.surampudi

20. Q. How to copy file into directory?

A. `cp /tmp/file_name .` (dot mean in the current directory)

21. Q. How to remove directory with files?

A. `rm -rf directory_name`

22. Q. What is the difference between internal and external commands?

A. Internal commands are stored in the; same level as the operating system while external commands are stored on the hard disk among the other utility programs.

23. Q. List the three main parts of an operating system command:

A. The three main parts are the command, options and arguments.

24. Q. What is the difference between an argument and an option (or switch)?

A. An argument is what the command should act on: it could be a filename, directory or name. An option is specified when you want to request additional information over and above the basic information each command supplies.

25. Q. What is the purpose of online help?

A. Online help provides information on each operating system command, the syntax, the options, the arguments with descriptive information.

26. Q. Name two forms of security.

A. Two forms of security are Passwords and File Security with permissions specified.

27. Q. What command do you type to find help about the command who?

A. `$ man who`

28. Q. What is the difference between home directory and working directory?

A. Home directory is the directory you begin at when you log into the system. Working directory can be anywhere on the system and it is where you are currently working.

29. Q. Which directory is closer to the top of the file system tree, parent directory or current directory?

A. The parent directory is above the current directory, so it is closer to the root or top of the file system.

30. Q. Given the following pathname:

`$ /business/acctg/payable/supplier/april`

a) If you were in the directory called `acctg`, what would be the relative pathname name for the file called `april`?

b) What would be the absolute pathname for `april`?

A.

a) `$ payable/supplier/april`

b) `$ /business/acctg/payable/supplier/april`

31. Q. Suppose your directory had the following files:

`help.1 help.2 help.3 help.4 help.O1 help.O2`

`aid.O1 aid.O2 aid.O3 back.1 back.2 back.3`

a) What is the command to list all files ending in 2?

b) What is the command to list all files starting in aid?

Naresh.surampudi

c) What is the command to list all "help" files with one character extension?

A.

- a) ls \*2
- b) ls aid.\*
- c) ls help.?

32. Q. What are two subtle differences in using the more and the pg commands?

A. With the more command you display another screenful by pressing the spacebar, with pg you press the return key.

The more command returns you automatically to the UNIX shell when completed, while pg waits until you press return.

33. Q. When is it better to use the more command rather than cat command?

A. It is sometimes better to use the more command when you are viewing a file that will display over one screen.

34. Q. What are two functions the move mv command can carry out?

A. The mv command moves files and can also be used to rename a file or directory.

35. Q. Name two methods you could use to rename a file.

A. Two methods that could be used:

- a. use the mv command
- b. copy the file and give it a new name and then remove the original file if no longer needed.

36. The soccer league consists of boy and girl teams. The boy file names begin with B, the girl teams begin with G. All of these files are in one directory called "soccer", which is your current directory:

Bteam.abc Bteam.OO1 Bteam.OO2 Bteam.OO4  
Gteam.win Gteam.OO1 Gteam.OO2 Gteam.OO3

Write the commands to do the following:

- a) rename the file Bteam.abc to Bteam.OO3.
- b) erase the file Gteam. win after you have viewed the contents of the file
- c) make a directory for the boy team files called "boys", and one for the girl team files called " girls"
- d) move all the boy teams into the "boys" directory
- e) move all the girl teams into the "girls" directory
- f) make a new file called Gteam.OO4 that is identical to Gteam.OO1
- g) make a new file called Gteam.OO5 that is identical to Bteam.OO2

A.

- a) mv Bteam.abc Bteam.OO3.
- b) cat Gteam.win -or- more Gteam.win  
rm Gteam. win
- c) mkdir boys  
mkdir girls
- d) mv Bteam\* boys
- e) mv Gteam\* girls
- f) cd girls  
cp Gteam.OO1 Gteam.OO4

g) There are several ways to do this. Remember that we are currently in the directory

/soccer/girls.

cp ../boys/Bteam.OO2 Gteam.OO5

or

Naresh.surampudi  
cd ../boys  
cp Bteam.OO2 ../girls/Gteam.OO5

37. Q. Draw a picture of the final directory structure for the "soccer" directory, showing all the files and directories.

38. Q. What metacharacter is used to do the following:  
1.1 Move up one level higher in the directory tree structure  
1.2 Specify all the files ending in .txt  
1.3 Specify one character  
1.4 Redirect input from a file  
1.5 Redirect the output and append it to a file

A.  
1. 1.1 double-dot or ..  
1.2 asterisk or \*  
1.3 question or ?  
1.4 double greater than sign: >>  
1.5 the less than sign or <

39. Q. List all the files beginning with A

A. To list all the files beginning with A command: ls A\*

40. Q. Which of the quoting or escape characters allows the dollar sign (\$) to retain its special meaning?

A. The double quote (") allows the dollar sign (\$) to retain its special meaning. Both the backslash (\) and single quote (') would remove the special meaning of the dollar sign.

41. Q. What is a faster way to do the same command?

mv file0.txt newdir  
mv file1.txt newdir  
mv file2.txt newdir  
mv file3.txt newdir

A. A shortcut method would be: mv file?.txt newdir

42. Q. List two ways to create a new file:

A.  
a. Copy a file to make a new file.  
b. Use the output operator e.g. ls -l > newfile.txt

43. Q. What is the difference between > and >> operators?

A. The operator > either overwrites the existing file (WITHOUT WARNING) or creates a new file.

The operator >> either adds the new contents to the end of an existing file or creates a new file.

44. Write the command to do the following:

44.1 Redirect the output from the directory listing to a printer.  
44.2 Add the file efg.txt to the end of the file abc.txt.  
44.3 The file testdata feeds information into the file called program  
44.4 Observe the contents of the file called xyz.txt using MORE.  
44.5 Observe a directory listing that is four screens long.

Naresh.surampudi

A.

44.1 ls > lpr

44.2 cat efg.txt >> abc.txt

44.3 program < testdata

44.4 more < xyz.txt

44.5 ls > dirsave | more

45. Q. How do you estimate file space usage

A. Use du command (Summarize disk usage of each FILE, recursively for directories.) Good to use arguments du -hs (-h, --human-readable print sizes in human readable format (e.g., 1K 234M 2G) (-s, --summarize display only a total for each argument)

46. Q. How can you see all mounted drives?

A. mount -l

47. Q. How can you find a path to the file in the system?

A. locate file\_name (locate - list files in databases that match a pattern)

48. Q. What Linux HotKeys do you know?

A. Ctrl-Alt-F1 Exit to command prompt

Ctrl-Alt-F7 or F8 Takes you back to KDE desktop from command prompt

Ctrl-Alt-Backspace Restart XWindows

Ctrl-Alt-D Show desktop

49. Q. What can you tell about the tar Command?

A. The tar program is an immensely useful archiving utility. It can combine an entire directory tree into one large file suitable for transferring or compression.

50. Q. What types of files you know?

A. Files come in eight flavors:

Normal files

Directories

Hard links

Symbolic links

Sockets

Named pipes

Character devices

Block devices

51. Q. How to copy files from on PC to another on the same network

A. Use the following command: scp yur\_file you\_login@your\_IP

example: copy .conf file from your PC to alex computer-

scp /etc/X11/xorg.conf alex@10.0.10.169:

52. Q. Please describe information below:

```
-rw-rw-r-- 1 dotpc dotpc 102 Jul 18 2003 file.buf
drwxr-xr-x 9 dotpc dotpc 4096 Oct 21 09:34 bin
lrwxrwxrwx 1 dotpc dotpc 20 Mar 21 15:00 client -> client-2.9.5
drwxrwxr-x 11 dotpc dotpc 4096 Sep 2 2005 client-2.8.9
drwxrwxr-x 7 dotpc dotpc 4096 Dec 14 12:13 data
drwxr-xr-x 12 dotpc dotpc 4096 Oct 21 09:41 docs
```



Naresh.surampudi

```
drwxr-xr-x  5 dotpc  dotpc      4096 Dec  7 14:22 etc
drwxr-xr-x 11 dotpc  dotpc      4096 Mar 21 15:54 client-2.9.5
-rw-r--r--  1 dotpc  dotpc     644836 Mar 22 09:53 client-2.9.5.tar.gz
```

A. This is a result of command `$ls -l`  
we have two files, 6 directories and one link to client-2.9.5 directory.  
There is number of files in every directory, size and data of last change.

53. Q. If you would like to run two commands in sequence what operators you can use?

A. ; or && the difference is:  
if you separate commands with ; second command will be run automatically.  
if you separate commands with && second command will be run only in the case the first was run successfully.

54. Q. How you will uncompress the file?

A. Use tar command (The GNU version of the tar archiving utility):  
`tar -zxvf file_name.tar.gz`

55. Q. How do you execute a program or script, my\_script in your current directory?

A. `./my_script`

56. Q. How to find current time configuration in the file my\_new.cfg

A. `grep time my_new.cfg`  
Grep searches the named input files (or standard input if no files are named, or the file name - is given) for lines containing a match to the given pattern.

Q. What does grep() stand for?

A. General Regular Expression Parser.

57. Q. What does the top command display?

A. Top provides an ongoing look at processor activity in real time. It displays a listing of the most CPU-intensive tasks on the system, and can provide an interactive interface for manipulating processes. (q is to quit)

58. Q. How can you find configuration on linux?

A. by using `/sbin/ifconfig`  
If no arguments are given, ifconfig displays the status of the currently active interfaces. If a single interface argument is given, it displays the status of the given interface only; if a single `-a` argument is given, it displays the status of all interfaces, even those that are down. Otherwise, it configures an interface.

59. Q. How to find difference in two configuration files on the same server?

A. Use diff command that is compare files line by line  
`diff -u /usr/home/my_project1/etc/ABC.conf /usr/home/my_project2/etc/ABC.conf`

60. Q. What is the best way to see the end of a logfile.log file?

A. Use tail command - output the last part of files  
`tail -n file_name` ( the last N lines, instead of the last 10 as default)

Naresh.surampudi

61. Q. Please write a loop for removing all files in the current directory that contains a word 'log'

A. for i in \*log\*; do rm \$i; done

62. Question: How to switch to a previously used directory?

Answer: cd -

---

1. Q.How many VI editor modes do you know?

A.Three modes -

Command mode: letters or sequence of letters interactively command vi.

Insert mode: Text is inserted.

Command line mode: enter this mode by typing ":" and entry command line at the foot of the screen.

2. Q. How can you terminate VI session?

A.

Use command: ZZ that is save changes and quit.

Use command line: ":wq" that is write changes and quit.

Use command line: ":q!" to ignore changes and quit.

3. Q. How can you copy lines into the buffer in command mode?

A.

yy - copy a single line defined by current cursor position

3yy - copy 3 lines. Current line and two lines below it.

What is LILO?

LILO stands for Linux boot loader. It will load the MBR, master boot record, into the memory, and tell the system which partition and hard drive to boot from.

What is the main advantage of creating links to a file instead of copies of the file?

A: The main advantage is not really that it saves disk space (though it does that too) but, rather, that a change of permissions on the file is applied to all the link access points. The link will show permissions of lrwxrwxrwx but that is for the link itself and not the access to the file to which the link points. Thus if you want to change the permissions for a command, such as su, you only have to do it on the original. With copies you have to find all of the copies and change permission on each of the copies.

Write a command to find all of the files which have been accessed within the last 30 days.

```
find / -type f -atime -30 > December.files
```

This command will find all the files under root, which is '/', with file type is file. '-atime -30' will give all the files accessed less than 30 days ago. And the output will put into a file call December.files.

What is the most graceful way to get to run level single user mode?

A: The most graceful way is to use the command init s.

If you want to shut everything down before going to single user mode then do init 0

Naresh.surampudi

first and from the ok prompt do a boot -s.

What does the following command line produce? Explain each aspect of this line.

```
$ (date ; ps -ef | awk '{print $1}' | sort | uniq | wc -l ) >> Activity.log
```

A: First let's dissect the line: The date gives the date and time as the first command of the line, this is followed by the a list of all running processes in long form with UIDs listed first, this is the ps -ef. These are fed into the awk which filters out all but the UIDs; these UIDs are piped into sort for no discernible reason and then onto uniq (now we see the reason for the sort - uniq only works on sorted data - if the list is A, B, A, then A, B, A will be the output of uniq, but if it's A, A, B then A, B is the output) which produces only one copy of each UID.

These UIDs are fed into wc -l which counts the lines - in this case the number of distinct UIDs running processes on the system. Finally the results of these two commands, the date and the wc -l, are appended to the file "Activity.log". Now to answer the question as to what this command line produces. This writes the date and time into the file Activity.log together with the number of distinct users who have processes running on the system at that time. If the file already exists, then these items are appended to the file, otherwise the file is created.

How would you make the following SQL statement run faster? SELECT \* FROM TABLEA WHERE COL1='A' AND COL2='B';

Make sure that COL1 and COL2 have indexes.

Find out which condition will return less values and use that as the first conditional.

What is Data Mining

Data mining is the process of sifting through extremely large amounts of Data to find trends or relevant information.

Name the Seven layers in the OSI Model.

Application, Presentation, Session, Transport, Network, Data Link, Physical

What is one way to view a unix network share on a Windows computer, within explorer

NFS, The Unix computer can be running a NFS Server Daemon.

How would you find all the processes running on your computer.

Unix, is ps -ef or ps -aux depending on version.

What is DHCP

DHCP is a way to dynamically assign IP address to computers. Dynamic Host Configuration Protocol

What is HTTP Tunneling

HTTP Tunneling is a security method that encrypts packets traveling through the internet. Only the intended recipient should be able to decrypt the packets. Can be used to Create Virtual Private Networks. (VPN)

Naresh.surampudi

You have 9 identical looking balls, however one ball is heavier than the others. You have two chances to use a balance. How do you find out which ball is the heaviest?

Split into groups of three, randomly choose two groups and use balance on them. If one group is heavier, then discard the other 6 balls. If the two groups are the same weight. The heavier ball must be in the group that was not on the scale. Now randomly choose two balls and test on balance. If they are the same weight, the heaviest ball is on one that was not tested. Else the heaviest ball is already known from the balance.

\* To display a list of all manual pages containing the keyword "date", what command would you type?

\* What command will display the first several lines of a file called "junk"?

== Users and permissions practicum ==

\* Rig it so everything in the folder gets deleted tonight at 10pm. Every night at 10pm.

== Local security ==

\* How do you feel about `sudo`?

\* What's the difference between `telnet` and `ssh`? What's a good use for each?

\* How do you ensure your users have hard-to-guess passwords?

== Filesystem ==

\* What is the difference between a symbolic and hard link? When would you use each?

\* I have a file named `-fr`. How do I get rid of it?

\* Why did I just ask that question?

\* To partition or not? How?

\* What are RAID 0, 1, 5, 0+1? What level would you use for a web server and why? A database server?

== `/etc` ==

\* `ls -l /etc`. What is all this stuff?

\* You added a line to `/etc/aliases`, but it doesn't seem to be working. Why?

\* You've created a `zope` user to run Zope under. How do you secure it so someone doesn't guess its password, log in with it, and mess with stuff?

\* Bring up `/etc/passwd`. What is all this junk?

\* What are shadow passwords?

== Processes ==

\* How many processes are running on your machine right now?

== Shells ==

\* Name as many shells as you can.

\* What's your favorite shell? Why?

\* Write a shell script to append "snork" to the file "test" but only if "test" already exists.

\* A user performed a `cd; chmod 644 .` before logging out. What problem occurs when he logs in the next time, and what level of privilege is required to correct the problem?

== Startup ==

\* Describe the boot process of your favorite Linux in as much detail as you can.

\* What are runlevels?

== Social ==

\* Describe an experience you had with a difficult user.

\* How do you keep up with current tools and practices?

\* How did you document your work at your last job so someone else could pick up where you left off?

Naresh.surampudi

== Totally miscellaneous ==

\* When debugging a core in gdb, what does the command `bt` give: core memory, heap usage, or calling stack?

\* A user complains the web site is slow. What do you do?

== Apache ==

\* How do you rig Apache to start up on boot?

\* Apache doesn't start up on boot, and the thing above checks out okay. How do you track down the problem?

\*To display a list of all manual pages containing the keyword "date", what command would you type?

Code:

```
man -k date
```

```
man -f date
```

[Linux / UNIX: Getting help with man page](#)

\* What command will display the first several lines of a file called "junk"?

Code:

```
head junk
```

```
man head
```

== Users and permissions practicum ==

\* Rig it so everything in the folder gets deleted tonight at 10pm. Every night at 10pm.

Set cronjob, see [How do I add jobs to cron under Linux or UNIX oses?](#)

== Local security ==

\* How do you feel about `sudo`?

sudo allows a permitted user to execute a command as the superuser or another user. sudo is much better than su and you don't have to share root password with other users/admin.

[Linux sudo Configuration](#)

\* What's the difference between `telnet` and `ssh`? What's a good use for each? TELNET, by default, does not encrypt any data sent over the connection (including password, and so it is often practical to eavesdrop on the communications and use the password later for malicious purposes;

SSH by default encrypt password and traffic. SSH is recommended for all use.

\* How do you ensure your users have hard-to-guess passwords?

Set password policy, see

[Howto: Protect account against a password cracking attack](#)

[Linux check passwords against a dictionary attack](#)

[Linux Password Cracking: Explain unshadow and john commands \( john the ripper tool \)](#)

== Filesystem ==

\* What is the difference between a symbolic and hard link? When would you use each?

[How to: Linux / UNIX create soft link with ln command](#)

Naresh.surampudi

[Understanding UNIX / Linux symbolic \(soft\) and hard links](#)

\* I have a file named ``-fr``. How do I get rid of it?

Code:

```
rm -- -fr
rm \-rf
```

[How to: Linux / UNIX Delete or Remove Files With Inode Number](#)

\* Why did I just ask that question?

For testing UNIX concepts and command line args.

\* To partition or not? How?

Sure. See [The importance of Linux partitions](#)

\* What are RAID 0, 1, 5, 0+1? What level would you use for a web server and why?  
A database server?

See level @ [What are the different RAID levels for Linux / UNIX and Windows Server?](#)

More about RAID - [Can RAID Act As The Reliable BACKUP Solution For Linux / UNIX / Windows Server?](#)

[Linux Check The Health of Adaptec RAID array](#)

== ``/etc`` ==

\* ``ls -l /etc``. What is all this stuff?

See, [Linux / UNIX - Display the permissions of a file](#)

\* You added a line to ``/etc/aliases``, but it doesn't seem to be working. Why?  
Restart sendmail so that file get updated. The program "newaliases" must be run after this file is updated for any changes to show through to sendmail / postfix.

Code:

```
newaliases
```

\* You've created a ``zope`` user to run Zope under. How do you secure it so someone doesn't guess its password, log in with it, and mess with stuff?  
Deny login access to zope and set shell to `/sbin/nologin`. There are other ways too...

\* Bring up ``/etc/passwd``. What is all this junk?

See, [Understanding /etc/passwd file format](#)

\* What are shadow passwords?

`/etc/shadow` file stores actual password in encrypted format for user's account with additional properties related to user password

[Understanding /etc/ shadow file](#)

== Processes ==

\* How many processes are running on your machine right now?

Code:

```
top
atop
ps -e
```

Naresh.surampudi

```
ps aux
ps aux | wc -l
man ps
```

== Shells ==

\* Name as many shells as you can.

Bourne shell (sh)

Almquist shell (ash)

Debian Almquist shell (dash)

Bourne-Again shell (bash)

Friendly interactive shell (fish)

Korn shell (ksh)

C shell (csh)

TENEX C shell (tcsh)

Es shell (e)

esh (Unix) Easy Shell

rc shell (rc) - shell for Plan 9 and Unix

runscript The initial shell interpreter used to process startup scripts in Gentoo

scsh (Scheme Shell)

Stand-alone Shell (sash)

Z shell (zsh)

\* What's your favorite shell? Why?

bash - it rocks and feature rich.

\* Write a shell script to append "snork" to the file "test" but only if "test" already exists.

Code:

```
[ -f test ] && echo "snork" >> test ||:
```

\* A user performed a `cd; chmod 644 .` before logging out. What problem occurs when he logs in the next time, and what level of privilege is required to correct the problem?

User will not be able to login. A root user can fix this problem by restoring permission

chmod perm /home/user

. is current directory

.. parent directory

== Startup ==

\* Describe the boot process of your favorite Linux in as much detail as you can.

See redhat or any other distro doc

\* What are runlevels?

The term runlevel refers to a mode of operation in one of the computer operating systems that implement Unix System V-style initialization. Conventionally, seven runlevels exist, numbered from zero to six, though up to ten, from zero to nine, may be used.

Code:

```
man init
man runlevel
```

== Social ==

\* Describe an experience you had with a difficult user.

\* How do you keep up with current tools and practices?

Naresh.surampudi

\* How did you document your work at your last job so someone else could pick up where you left off?

Use our social skillz

== Totally miscellaneous ==

\* When debugging a core in gdb, what does the command `bt` give: core memory, heap usage, or calling stack?

Code:

```
man gdb
Read gdb page
```

\* A user complains the web site is slow. What do you do?  
Ask user to upgrade internet connection. If using windows ask to reboot windows ..  
LOL just kidding, google for slow apache problem. There could be zillions of causes

== Apache ==

\* How do you rig Apache to start up on boot?

Code:

```
chkconfig httpd on
```

\* Apache doesn't start up on boot, and the thing above checks out okay. How do you track down the problem?

Code:

```
chkconfig httpd on
httpd -t
service httpd on
netstat -tulpn | grep 80
tail -f /var/log/httpd/access_log
tail -f /var/log/httpd/error_log
```

What is the default Window system / Windows manager used in Linux?

Code:

```
X.org
```

What command is used to list the contents of directory?

Code:

```
ls
ls -l
```

What command is used to list the top 10 files / directories size wise?

Code:

```
for X in $(du -s * | sort -nr | cut -f 2); do du -hs $X ; done
```

What command is used to display a list of currently running processes?

Code:

```
ps
top
pstree
```



Naresh.surampudi

```
pgrep  
/proc file system
```

What is a login shell?

A program get executed when a user logs into UNIX box. E.g. bash, sh, ksh, csh

What is UID?

User identification number which is assigned to each UNIX / Linux user; it may or may not be unique (unique number is recommended to avoid security related

issue . UID and user relationship defined in /etc/passwd file.

Code:

```
man id  
man users  
man groups
```

What GID? What is EUID? What SUID? What is RUID?

[Linux > More on USER ID, Password, and Group management](#)

[Understanding /etc/passwd file format](#)

[Understanding /etc/ shadow file](#)

Explain Unix User security concept

Permissions - chmod and chown

User groups - group management - user management

Read su, sudo man page

What PID?

Process identification number; use ps command to see PID. It is a number used by Unix kernels and Windows operating systems to identify a process.

Explain process ID zero and process ID 1?

All the idle task has process ID zero, and never exits.

The init process, with process ID 1, which does nothing but wait around for its child processes to die. Usually started for /etc/inittab

Explain wheel group usage along with an example?

Code:

```
man su
```

What command is used to check a file system for errors?

Code:

```
fck  
fck.ext3  
fck.nfs  
fck.ext2  
fck.vfat  
fck.reiserfs  
fck.msos
```

Is Linux / UNIX file system case sensitive? Give one example

Yes, test.txt and TEST.txt are two different files

What file contains the list of drives that are mounted at boot?

Naresh.surampudi

/etc/fstab - Linux / Other UNIX version

/etc/vfstab - Solaris UNIX

Explain the usage of the fourth field in /etc/fstab?

It is formatted as a comma separated list of options. Read mount command man page for all the option.

What is /etc/inittab file? In what file is the default run level defined?

System V init examines the '/etc/inittab' file for an 'initdefault' entry, which tells init whether there is a default runlevel. init is the program on Unix that spawns all other processes. It runs as a daemon and typically has PID 1.

Code:

```
man init
cat /etc/inittab
```

Common runlevel values on RHEL

Code:

```
0. Halt
1. Single user mode
6. Reboot
3. Default text
5. Default GUI
```

To check the current runlevel:

Code:

```
who -r
runlevel
```

What command is used to get help about command? What command is used to read manual page for a given command?

Code:

```
info command-name
man command-name
command-name -h
command-name --help
```

What command form or symbol used to redirect output to a file?

Use the > symbol

Code:

```
command-name > output.txt
```

Use the < symbol to read input from a file

What is ssh? Specify ssh command syntax to execute command over a TCP/IP network?

SSH is Application layer protocol which allows data to be exchanged over a secure channel between two computers.

[Sshh - Wikipedia, the free encyclopedia](#)

Code:

Naresh.surampudi

```
ssh user@remote.box command-name
```

Explain steps for password less login? How do you set-up SSH with DSA / RSA public key authentication?

[Howto Linux / UNIX setup SSH with DSA public key authentication \(password less login\)](#)

Explain BSD-style init?

Read rc man page. There are no runlevels; the 'rc' file determines how init is to be run.

[Runlevel - Wikipedia, the free encyclopedia](#)

Explain Sys-V style init?

Read init man page

[Runlevel - Wikipedia, the free encyclopedia](#)

Explain Unix philosophy

[Unix philosophy description by The Linux Information Project \(LINFO\)](#)

What is POSIX? Name 2 POSIX-oriented operating systems?

Portable Operating System Interface is the collective name of a family of related standards specified by the IEEE to define the application programming interface (API). HP-UX, Solaris, AIX etc

Linux and BSD not certified as POSIX but conform to standard.

Explain Raw device and command s to configure Raw device

Block device file that allows accessing a storage device such as a hard drive directly. For example /dev/hda. Use commands

Code:

```
mknod
fdisk
mkfs
mkfs.ext3
```

Explain File descriptor

[File descriptor - Wikipedia, the free encyclopedia](#)

Explain Unix file types

Directory

Pipes

Fifo

Symbolic link

Named pipe

Socket

Device file

Door

Regular file

Explain inode, superblock and hard links

[Understanding UNIX / Linux file system](#)

Explain Unix domain socket

[Unix Sockets](#)

MySQL and many programs uses domain socket to make client / server

Naresh.surampudi  
communication. Usually fast as compare to TCP/IP

Explain UNIX software pipeline concept  
See shell or bash man page

Code:

```
cat /etc/passwd | grep username  
mount | grep cdrom
```

Explain XYZ Unix daemons  
Where XYZ can be any one of the following:

Code:

```
init  
httpd  
dhcpd  
lpd  
nfsd  
ntpd  
syslogd  
ypbind  
ftpd  
telnetd  
sshd  
named
```

Explain udev in Kernel 2.6?

Explain Process management and related commands

Explain Memory management and related commands

What is Open Group standards?

Specify special usage for each one of the following file

/dev/null - Send unwanted output

/dev/random - Random number generation

/dev/zero - Cache or Destroy data on a partition - dd if=/dev/zero of=/dev/sda98

What is SELinux?

Write a command to find all of the files which have been accessed within the last 10 days.

What is LILO?

What is Grub?

Explain the difference between LILO and Grub

What is NFS?

What is NAMED?

What is MySQLD?

Naresh.surampudi

What is mysql?

What is CVS?

Why You Shouldn't Use the root Login for everyday work?

Describe the default partition scheme in Redhat Linux?

Describe the default partition scheme in Solaris? What is the slice number?

Describe all default mount point?

What is boot block?

What is logical block?

Describe the process for adding a new hard disk to UNIX box?

Describe the process for adding a new hard disk to Linux box?

Describe the process for adding a new hard disk to Linux LVM to grow /home?

Explain one major difference between a regular file system and a journaling file system?

Define JFS

Define UFS

How do you lock and unlock user account / password?

Describe RPM and command to install / remove / update Linux system?

Explain difference between rpm and up2date command.

Explain difference between rpm and apt-get command.

Explain difference between rpm and yum command.

Describe usage for pkgadd, pkginfo and pkgchk command

How do you find files on UNIX or Linux system?

Explain /etc/rc3.d

Explain ntsysv or chkconfig command

How do you get rid of process if kill PID is not working for you?

What is the purpose of the command?

grep

sed

awk

ifconfig

netstat

df

Naresh.surampudi

du

prvtoc

fdisk -l

umaks

getfacl

setfacl

sudo

fsck

probe-scsi

vmstat

Explain LVM

- 1) What is a superblock ?
- 2) What is a parity bit?
- 3) What is an inod?
- 4) Explain top command ?
- 5) How to disable the root login in SSH ?
- 6) use of sysctl command ?
- 7) LVM how to ?
- 8) Different RAID levels ?

What are the services required for nfs, apache(http) and NIS?

What is the best way to check the status of any service?

What do you mean by parity in RAID and which RAID is useful now a days?

Explain Linux Boot process especially kernel and initrd.

Why we do have two commands useradd and adduser when their functionality is same?

Can we have two apache servers having diff versions?

Q: How do you display your running kernel version? (Solaris, AIX, Linux)

A: Linux # uname -r , Solaris # showrev

Q: Which command do you use to display a table of running processes? (Solaris, AIX, Linux)

A: Linux # ps -ef and top , Solaris # prstat

Q: Which file do you modify to configure a domain name resolver? (Solaris, AIX, Linux)

A: Linux # /etc/resolv.conf , Solaris # /etc/resolv.conf

Q: Which file contains a list of locally defined hostnames and corresponding IP addresses? (Solaris, AIX, Linux)

A: Linux # /etc/hosts , Solaris # /etc/hosts and linked file /etc/inet/hosts

Q: How do you display a routing table? (Solaris, AIX, Linux)

A: Linux # ip route show or #netstat -nr or #route -n and Solaris # netstat -nr and #route -n

Q: Which command would you use to view partitions and their sizes on Solaris?

A: # df -kh

Q: Which OpenBoot command would you use to print/view OpenBoot environment

Naresh.surampudi

variables on a SUN server?

A: #printenv

Q: What does "ypwhich" command do? (Solaris, AIX, Linux)

A: # Will display NIS server to which client is connected to and which NIS Server is master for particular map specified with this command

Q: which command would you use to create an OS user on Solaris and Linux?

A: Linux # useradd and Solaris #useradd

Q: Which file contains passwords for local users on Solaris, Linux and on AIX?

A: Linux #/etc/shadow and Solaris # /etc/shadow

Q: Which command would you use to list partitions on Linux?

A: Linux # mount -l or # df -kh

Q: Which command/commands would you use to manage installed packages on RedHat Linux?

A: Linux # rpm

Q: What is the default port for SSH server?

A: 22

Q: Which command/commands would you use to manage installed packages on Solaris?

A: #pkginfo #pkgrm # pkgadd #pkgchk

Q: What command would you use to install an OS patch on Solaris?

A: #showrev -p and #patchadd -p

Q: Which Veritas command would you use to display a list of Veritas volumes?

A: # vxprint

Q: Which Veritas command would you use to display a list of disks on a system?

A: # vxdx list

Q: What is the main system configuration utility in AIX?

A:

Q: Which file has a list of filesystems to be mounted at boot time on Solaris, Linux and AIX?

A: Linux # /etc/fstab and Solaris #/etc/vfstab

Q: Which Veritas Cluster Server command would you use to display the status of a cluster and it's resources?

A:

Q: Which command would you use to rename a disk for VMware Guest virtual machine on ESX server 3 storage volume?

A:

Q: Which command would use on VMware ESX 3 server to display virtual switch configuration?

A:

Naresh.surampudi

Q: Which Veritas Cluster Server command would you use to display the status of a cluster and its resources?

A: clustat and to manage the cluster configuration use clusvcadm

Q: Which command would you use to rename a disk for VMware Guest virtual machine on ESX server 3 storage volume?

A: the best way is clone vm to different datastore or in the same datastore with different name

vmkfstools -i \\vmfs\old\_vm.vmdk \\vmfs\new\_vm.vmdk  
this will take care of it all

Q: Which command would use on VMware ESX 3 server to display virtual switch configuration?

A: esxcfg-vswitch -l or user esxcfg-vswitch -help to see all options

Access common commands quicker ?ps -ef | grep -i \$@

Alternative for top command ?prstat -a

Change to a directory, which is having very long name ?

cd CDMA\_3X\_GEN\*Here original directory name is .  
.CDMA\_3X\_GENERATION\_DATAAnswer:.

Delete blank lines in a file ?

cat sample.txt | grep -v ?^\$? > new\_sample.txt

Display disk usage in Kilobytes ?

du -k

Display Ethernet Address arp table ?

arp -a

Display the all files recursively with path under current directory ?

find . -depth -print

Display the Disk Usage of file sizes under each directory in currentDirectory ?

du -k \* | sort .nr (or) du .k . | sort -nr

Display the files in the directory by file size ?

ls .ltr | sort .nr .k 5

Display the inter-process communication facility status ?

Ipcs



Naresh.surampudi

11. Display the last newly appending lines of a file during appending data to the same file by some processes ?

tail -f Debug.log Here tail shows the newly appended data into Debug.log by some processes/user.

12. Display the no. of active established connections to localhost ?

netstat -a | grep EST

13. Display the page size of memory ?

pagesize -a

14. Display the parent/child tree of a process ?

ptree Example: ptree 1267

15. Display the processes current open files ?

pfiles Example: pfiles 1267

16. Display the processes, which are running under your username ?

ps .aef | grep Singh Here, Singh is the username.

17. Display the state of interfaces used for TCP/IP traffic ?

netstat -i

18. Display the top most process utilizing most CPU ?

top .b 1

19. Display top ten largest files/directories ?

du -sk \* | sort -nr | head

20. Explain difference between IPC mechanisms

ipc mechanisms are mainly 5 types 1.pipes:it is related data only send from one pipe output is giving to another pipe input to share resources pipe are used drawback:it is only related process only communicated 2.message queues:message queues are unrelated process are also communicate with message queues drawback:user dont know which process currently works share memory:memory shared in distributed systems some memory wants to share some files that time it is use full semaphores semaphore is integer type and in semaphore resources give coding like negative value means process are wants to use particular resource waiting only and 0 means no process is waiting and 1 means one resource is free and sockets:sockets also ipc it is communicate clients and server with socket system calls connection oriented and connection less also PIPE: Only two related (eg: parent & child) processes can be communicated. Data reading would be first in first out manner. Named PIPE or FIFO : Only two processes (can be related or unrelated) can communicate. Data read from FIFO is first in first out manner. Message Queues: Any number of processes can

Naresh.surampudi

read/write from/to the queue. Data can be read selectively. (need not be in FIFO manner) Shared Memory: Part of process's memory is shared to other processes. other processes can read or write into this shared memory area based on the permissions. Accessing Shared memory is faster than any other IPC mechanism as this does not involve any kernel level switching(Shared memory resides on user memory area). Semaphore: Semaphores are used for process synchronisation. This can't be used for bulk data transfer between processes.

21. Explain the layered aspect of a UNIX system. What are the layers? What does it mean to say they are layers?

A UNIX system has essentially three main layers: ? The hardware ? The operating system kernel ? The user-level programs The kernel hides the system's hardware underneath an abstract, high-level programming interface. It is responsible for implementing many of the facilities that users and user-level programs take for granted. The kernel assembles all of the following UNIX concepts from lower-level hardware features: ? Processes (time-sharing, protected address space) ? Signals and semaphores ? Virtual Memory (swapping, paging, and mapping) ? The filesystem (files, directories, namespace) ? Pipes and network connections (inter-process communication)

22. Explain the steps that a shell follows while processing a command.

After the command line is terminated by the key, the shell goes ahead with processing the command line in one or more passes. The sequence is well defined and assumes the following order. Parsing: The shell first breaks up the command line into words, using spaces and the delimiters, unless quoted. All consecutive occurrences of a space or tab are replaced here with a single space. Variable evaluation: All words preceded by a \$ are evaluated as variables, unless quoted or escaped. Command substitution: Any command surrounded by backquotes is executed by the shell which then replaces the standard output of the command into the command line. Wild-card interpretation: The shell finally scans the command line for wild-cards (the characters \*, ?, [, ]). Any word containing a wild-card is replaced by a sorted list of filenames that match the pattern. The list of these filenames then forms the arguments to the command. PATH evaluation: It finally looks for the PATH variable to determine the sequence of directories it has to search in order to hunt for the command.

23. Explain the unix file system.

Basically there are 4 different types of file systems in unix ,they are as follows  
1.Device file 2.Directory fil. 3.FIFO 4.Regular file Unix:-There are two file system t.e  
exe1,exe2 Solaris:-Three types file system a)Disk base file system ufs(unix file system),HSFS(High sierra file system),PCFS,UFS(Universal file system)  
b)NFS(Network file system) c)Pseudo or virtual file system

24. Explain the working of Virtual Memory.

Virtual memory like as a temporary storage are Answer:It consists of page table.In this pages are divided into frames.It is a contiguous memory allocation.It is also called logical memory. Memory contents that appear to be in contiguous addresses, but are actually mapped to different physical memory locations by hardware action of the translation lookaside buffer (TLB) and page tables.

25. Give examples of how memory leaks can occur with c programs

Naresh.surampudi

a memory leak occurs when mem is allocated but never freed . leaks can b caused by using malloc() without using any free(). but leaks can also be caused if a pointer to dynamically allocated memory is delleted overwritten , it can be caused when allocated mem is overwritten accidentally

26. How Connect to a Database in Shell Programming?Please tell me Step by Step

```
To Connect to Oracle Database and Collect the data in to text file from the table.
echo "Enter User Name "read nameecho "Enter passwd "stty -echoread passsstty echoecho
"Enter Detp no"read dnosqlplus -S $name@oracle/$pass< Suppose you are using
db2 and ksh #!/usr/bin/ksh connect to :@ user using -----
If u have profile and catalog then ----- #!/usr/bin/ksh db2
connect to ----- To run a sample proc .. use the below db2
""
```

27. How do u determine disk usage?

The disk usage can be determined by using the command,du.THIS command outputs the number of kilobytes used by each sub-directory.

28. How do you debug a core dump

dbx-core-'exename' If you want to migrate a 32 bit application to a 64 bit os, what all would you check I am assuming the question is - port 32 bit appl to 64 bit os. This can a simple checklist of things one can check for - 1. Int, long will be 64 bits so care should be taken to see if there are any comparisons or allocations that assume a certain fixed amount of Memory 2. short is 32 bit and long long are 128 bit - conversion and assignment issues around mixing ints with longlongs / shorts and ints 3. ptr\* is 64 bit, make sure things are properly aligned (in some OSes where alignment is a concern) 4. if the 32 appl has to run as is in a 64 bit machine then care should be taken to do so(coexistence of 32 and 64 bit application) 5. compilers and makefile issues should be handled

29. How do you find out drive statistics ?

```
iostat -E
```

30. How do you kill the process

```
kill -9
```

31. How do you know that how many commands can be stored in the buffer in one session?

```
type "env" command on the prompt and check HISTORY or BUFFER or HISTSIZE
value
```

32. How do you see the number of processes running in the system

```
ps -ef
```

Naresh.surampudi

33. How does the kernel differentiate device files and ordinary files?

Kernel checks 'type' field in the file's inode structure. device files are of 2 types --- character device file and block device file type field in the file's inode structure b--- block device file c--- character device file

34. How is the command `?$cat file2 ?` different from `?$cat >file2`

The Command `$cat file` in unix is used to display the content of the file and where as command `$cat >> file` is to append the text to the end of the file without overwriting the information of the file. In case if the file does not exist in the directory the command will create a newfile in file system.

35. How many prompts are available in a UNIX system?

Two prompts, PS1 (Primary Prompt), PS2 (Secondary Prompt). Unix/ Linux Supports four Prompts PS1, PS2, PS3, PS4

36. How many prompts are available in Unix / Linux

Unix/ Linux Supports four Prompts PS1, PS2, PS3, PS4 and you can also set them in .profile

37. How much space is used for users in kilobytes ?

`quot -af`

38. How to create a blank file in unix

`touch`

39. How to create null file ?

`cat /dev/null > filename1`

40. how to find free space in unix/linux

`df -h` on linux `free -m` will display free memory in MB '`df`' and '`du`' commands give the information about space. `df -k` the better command is `df -h`

41. How to know the date & time for . when script is executed ?

Add the following script line in shell script. `eval echo "Script is executed at `date`"`  
`>> timeinfo.inf` Here, .timeinfo.inf. contains date & time details ie., when script is executed and history related to execution.

42. How to save man pages to a file ?

`man | col .b > Example : man top | col .b > top_help.txt`

43. How to see your server name

Naresh.surampudi  
hostname

44. how to find free space in unix/linux

on linux free -m will display free memory in MB 'df' and 'du' commands give the information about space. df -k df -h or df -H human readable gives human readable format of free space

45. How to switch to a super user status to gain privileges?

Use 'su' command. The system asks for password and when valid entry is made the user gains super user (admin) privileges.

46. How to terminate a process which is running and the specialty of kill command?

With the help of kill command we can terminate the process. Syntax: kill pid Kill 0 - kills all processes in your system except the login shell.

47. How would you create shared and dynamic libraries

Well shared libraries have 2 types 1) Static 2) Dynamic. u can create library by ar cr -o sharedobj.a file1.o file2.o while file1 and file2 are object files (obj) now put this sharedobj.a into /usr/lib directory

48. How would you remove a semaphore / shared memory whose owner processes have died?

ipcrm -sem id ; for semaphores ipcrm -shm id ; for shared mem

49. If you have a string "one two three", which shell command would you use to extract the strings

```
echo $string | cut -d" " -f1 echo $string | cut -d" " -f2 echo $string | cut -d" " -f3  
echo "one two three" | cut -d" " -f 1,2,3 or echo "one two three" | awk '{print $1 $2 $3}'
```

50. In Unix OS, what is the file server?

The file server is a machine that shares its disk storage and files with other machines on the network.

51. In Unix OS, what is the file server? (Asked by Silicon Magic Corp. people)

The file server is a machine that shares its disk storage and files with other machines on the network.

52. Is 'du' a command? If so, what is its use?

Yes, it stands for 'disk usage'. With the help of this command you can find the disk capacity and free space of the disk. du -k : returns the disk usage in KBs du -k : gives the d

Naresh.surampudi

53. Is it possible to count number char, line in a file; if so, How?

Yes, It is possible to count the words, lines, characters in a file.. In Unix, I used.. \$wc -c filename -> for characters \$wc -l filename -> for lines \$wc -w filename -> for words

54. Is it possible to restrict incoming message?

yes its by using mesg but the command is mesg n (to restrict incoming message) mesg y (to allow message)

55. List some Hot Keys for bash shell ?

Ctrl+l . Clears the Screen. Ctrl+r . Does a search in previously given commands in shell. Ctrl+u - Clears the typing before the hotkey. Ctrl+a . Places cursor at the beginning of the command at shell. Ctrl+e . Places cursor at the end of the command at shell. Ctrl+d . Kills the shell. Ctrl+z . Places the currently running process into background.

56. List the files in current directory sorted by size ?

ls -l | grep ^- | sort -nr

57. List the hidden files in current directory ?

ls -a1 | grep "^\\."

58. Search for a sample string in particular files ?

grep .Debug. \*.conf Here grep uses the string .Debug. to search in all files with extension..conf. under current directory.

59. Set the Display automatically for the current new user ?

export DISPLAY=`eval ?who am i | cut -d "(" -f2 | cut -d ")" -f1`` Here in above command, see single quote, double quote, grave ascent is used. Observe carefully.

60. Show the working directory of a process ?

pwdx Example: pwdx 1267

61. Unix command to concatenate (attach) two strings?

echo \$var1 \$var2

62. What are Shell Variables

shell variables are system environment variables. they include TERM, SHELL, MAIL the output of the shell variable we can see by typing the command \$>echo \$TERM ansi at the prompt.

63. What are shell variables?

Naresh.surampudi

shell variables are system environment variables. they include TERM, SHELL, MAIL the output of the shell variable we can see by typing the command `$>echo $TERM` and at the prompt.

64. What are the difference between Daemons in Unix and service processes in Windows?

both are same. daemons are the background processes in unix. similarly background processes in windows are called as service agents or service processes.

65. What are the differences between Shared and Dynamic libraries

There are two ways in which a library is shared. Static and dynamic In statically **linked** library the code of library is referenced at compile time and the result executable will be bigger. In dynamically linked libraries the code of library is referenced at run time and resulting executable will be smaller. But drawback is that at run time this will need the library to reference the library related symbols.

66. What are the techniques that you use to handle the collisions in hash tables?

We can use two major techniques to handle the collisions. They are open addressing and separate chaining. In open addressing, data items that hash to a full array cell are placed in another cell in the array. In separate chaining, each array element consist of a linked list. All data items hashing to a given array index are inserted in that list.

67. What difference between cmp and diff commands?

cmp gives the 1st byte no and line no of file1 which is to be changed to make filename1 identical to filename2. diff gives the text of filename2 which is different from filename1. we can use directory name in diff, but not in cmp

68. What does the following command do? `cp $ABC/$SUP/xyz.txt .;`

This command will copy the file abc.txt (picked up from the location \$ABC/\$SUP/) to the current directory u are working on. The file created on the current directory will also be of the same name abc.txt

69. What does the following command line produce? Explain each aspect of this line. `$ (date ; ps -ef | awk {print $1}' | sort | uniq | wc -l) >> Activity.log`

First let's dissect the line: The date gives the date and time as the first command of the line, this is followed by the a list of all running processes in long form with UIDs listed first, this is the ps -ef. These are fed into the awk which filters out all but the UIDs; these UIDs are piped into sort for no discernible reason and then onto uniq (now we see the reason for the sort - uniq only works on sorted data - if the list is A, B, A, then A, B, A will be the output of uniq, but if it's A, A, B then A, B is the output) which produces only one copy of each UID. These UIDs are fed into wc -l which counts the lines - in this case the number of distinct UIDs running processes on the system. Finally the results of these two commands, the date and the wc -l, are appended to the file "Activity.log". Now to answer the question as to what this command line produces. This writes the date and time into the file Activity.log together with the number of distinct users who have processes running on the

Naresh.surampudi

system at that time. If the file already exists, then these items are appended to the file, otherwise the file is created.

70. What is a dynamically linked file?

This is confusing because of the use of the word 'file'. A dynamically linked program is one that, when executed, loads shared libraries from /lib or /usr/lib in order to execute. The idea is that most programs use many of the same functions, so include a copy of a common function in \*every\* program on the file system. Instead, the function is placed in a shared library and when the program starts executing, the library is loaded which provides the program access to the function.

71. What is a pipe and give an example?

A pipe is two or more commands separated by pipe char '|'. That tells the shell to arrange for the output of the preceding command to be passed as input to the following command. Example : ls -l | pr The output for a command ls is the standard input of pr. When a sequence of commands are combined using pipe, then it is called pipeline. A much better example is: who|wc -l This redirects the output of the "who" command to the wc -l and thus displays number of users that currently logged on...

72. What is a router? What is a gateway?

Routers are machines that direct a packet through the maze of networks that stand between its source and destination. Normally a router is used for internal networks while a gateway acts a door for the packet to reach the 'outside' of the internal network

73. What is a statically linked file?

hard link (created with ln). Source and dest will have the same inode. Making two different copies causes more disk space due to redundancy.

74. What is AWK

AWK is a complete pattern scanning and processing language, it is most commonly used as a Unix command-line filter to reformat the output of other commands. For example, to print only the second and sixth fields of the date command (the month and year) with a space separating them, at the Unix prompt, you would enter: date | awk '{print \$2 ? ? \$6}'

75. What is CVS? List some useful CVS commands.

CVS is Concurrent Version System. It is the front end to the RCS revision control system which extends the notion of revision control from a collection of files in a single directory to a hierarchical collection of directories consisting of revision controlled files. These directories and files can be combined together to form a software release. There are some useful commands that are being used very often. They are cvs checkout cvs update cvs add cvs remove cvs commit

76. What is difference between Hard and Soft mount?

Hard mount in unix in the normal filesystem mount mainly used to mount local filesystems. The filesystem will be in the mounted state until you unmount it



Naresh.surampudi

manually. Soft mount is an option that is very useful for mounting network filesystems say nfs filesystem so that soft mount will allow automatic unmount if the filesystem is idle for a specified time period Eg. can be used with autofs

77. what is difference between lilo and stub?

STUB is A temporary implementation of part of a program for debugging purposes.LILO (Linux LOader) is a boot loader for Linux.

78. what is Inode?

a unique number associated with each filename. This number is used to look up an entry in the inode table which gives information on the type, size, and location of the file and the userid of the owner of the file.

79. What is LILO?

LILO stands for Linux boot loader. It will load the MBR, master boot record, into the memory, and tell the system which partition and hard drive to boot from.

80. what is Mutex?

Short for mutual exclusion object. In computer programming, a mutex is a program object that allows multiple program threads to share the same resource, such as file access, but not simultaneously. When a program is started, a mutex is created with a unique name. After this stage, any thread that needs the resource must lock the mutex from other threads while it is using the resource. The mutex is set to unlock when the data is no longer needed or the routine is finished.

81. What is NFS? What is its job?

NFS stands for Network File System. NFS enables filesystems physically residing on one computer system to be used by other computers in the network, appearing to users on the remote host as just another local disk.

82. what is Process Control board in UNIX?

I heard about the Process control block. Which will store the Pids of the processes in the system. But Process control board??

83. What is redirection?

Redirection is a feature in Unix where the data from the standard out put or a file,so on.can be redirected i.e divert to a file or a program and vice versa. > -- out put redirection >> -- out put redirectin(appendng at the last) < -- input redirection

84. What is relative path and absolute path.

absolute path is from home directory to user directory.. but relative path is from present working directory to user directory..

85. What is SED?

Naresh.surampudi

SED (which stands for Stream Editor) is a simple but powerful computer program used to apply various pre-specified textual transformations to a sequential stream of text data. Answer: It reads input files line by line, edits each line according to rules specified in its simple language (the sed script), and then outputs the line.

86. What is Semaphore? What is deadlock?

Semaphore is a synchronization tool to solve critical-section problem, can be used to control access to the critical section for a process or thread. The main disadvantage (same of mutual-exclusion) is require busy waiting. It will create problems in a multiprogramming system, where a single CPU is shared among many processes. Busy waiting wastes CPU cycles. Deadlock is a situation when two or more processes are waiting indefinitely for an event that can be caused by only one of the waiting processes. The implementation of a semaphore with a waiting queue may result in this situation.

87. What is stty used for

Sets options for your terminal. Print or change terminal characteristics

88. What is the difference between > and >> redirection operators ?

'>' - redirects the O/P to file by creating it and if the file already exists it takes the liberty to overwrite it without user intimation '>>' - redirects the output to a file by creating it and if the file already exists, it is appended

89. What is the difference between a 'thread' and a 'process'?

Process is a program in execution whereas thread is a separate path of execution in a program. Process & Thread share almost all datastructure except (thread has its own register & stack area ) A process is a collection of virtual memory space, code, data, and system resources. A thread is code that is to be serially executed within a process. A processor executes threads, not processes, so each application has at least one process, and a process always has at least one thread of execution, known as the primary thread. A process can have multiple threads in addition to the primary thread Thread ? is stream of executable code within process. They are light weight process. All thread with in a process share process instruction,code & data segment,open file descriptor,signal handler,userID and GroupID. Thread has its own set of register including program counter,stack pointer

90. What is the difference between binaries in /bin, and /usr/bin?

Under Solaris, there is no difference. /bin is a symbolic link pointing to /usr/bin. Under Linux (RHAS3) /bin is seemingly for standard unix programs like vi, cp, mv, rm which you'd need in a single user environment where as /usr/bin contains programs you'd want for a multiuser environment. Keep in mind that sometimes /usr is a different disk partition and when you start up in single user mode you only have / mounted. The /sbin directories are \*supposed to\* contain statically linked programs. This was morphed into the idea of bin for user programs,/sbin for admin programs.

91. What is the difference between cat and more command?

In Unix, cat command is used to display the contents on to the screen \$cat filename  
more command is used to display a file with pausing. \$more filename

Naresh.surampudi

92. What is the difference between soft link and hard link in unix operating system ?

Soft link can span across file system Hard link cannot span across file system Soft link have different inod numbers ( ls -li) Hard link have same inod number for all files Soft link can link directories and files Hard link can only link files not directories You can create non existing files with the soft link Hardware link fails if the source files dont exists. If you remove the source files, soft link will not work . Remove the source files in hard link , still data will be available if its a hard link. Usage : ln -s source file target file ( soft link)Usage :ln source file target file ( hard link)Send your solaris queries to support.solaris@gmail.com.

93. What is the main advantage of creating links to a file instead of copies of the file?

The main advantage is not really that it saves disk space (though it does that too) but, rather, that a change of permissions on the file is applied to all the link access points. The link will show permissions of lrwxrwxrwx but that is for the link itself and not the access to the file to which the link points. Thus if you want to change the permissions for a command, such as su, you only have to do it on the original. With copies you have to find all of the copies and change permission on each of the copies.

94. What is the major advantage of a hash table?

The major advantage of a hash table is its speed. Because the hash function is to take a range of key values and transform them into index values in such a way that the key values are distributed randomly across all the indices of a hash table.

95. What is the major advantage of a hash table? (Asked by Silicon Magic Corp. people)

The major advantage of a hash table is its speed. Because the hash function is to take a range of key values and transform them into index values in such a way that the key values are distributed randomly across all the indices of a hash table.

96. What is the most graceful way to get to run level single user mode?

The most graceful way is to use the command init s. If you want to shut everything down before going to single user mode then do init 0 first and from the ok prompt do a boot -s.

97. What is the significance of the ?tee? command?

tee - replicate the standard output Syntax:-tee [ -ai ] [ file ... ] -a Append the output to the files rather than overwriting them. -i Ignore interrupts. The tee utility will copy standard input to standard out- put, making a copy in zero or more files. tee will not buffer its output. The options determine if the specified files are overwritten or appended to.

98. what is the unix command used for giving or changing the permissions for files and folders?

Naresh.surampudi

chmod [ -R ] [ -h ] [ -f ] PermissionCode { File | Directory }-f Suppresses all error reporting except invalid permissions and usage statements.-h Suppresses a mode change for the file or directory pointed to by the encountered symbolic link.-R Descends only directories recursively, as specified by the patternFile...|Directory... The -R flag changes the file mode bits of each directory and of all files matching the specified pattern.You can get more info from:  
<http://nersp.cns.ufl.edu/~dicke3/nerspcs/chmod.html> Farhan Anwar Network and Systems Administrator The Aga Khan University Karachi, Pakistan

99. What is the use of `grep` command?

`grep` is a pattern search command. It searches for the pattern, specified in the command line with appropriate option, in a file(s). Syntax : `grep` Example : `grep 99mx mcafile`

100. What is the use of the command "`ls -x chapter[1-5]`"

`ls` stands for list; so it displays the list of the files that starts with 'chapter' with suffix '1' to '5', `chapter1`, `chapter2`, and so on.

101. what is this line in the shell script `do ??? #!/bin/ksh`

To invoke the shell indirectly this line is added as the first line in the file. This particular line invokes Korn shell

102. What is UTP?

UTP ? Unshielded twisted pair 10BASE-T is the preferred Ethernet medium of the 90s. It is based on a star topology and provides a number of advantages over coaxial media: It uses inexpensive, readily available copper phone wire. UTP wire is much easier to install and debug than coax. UTP uses RG-45 connectors, which are cheap and reliable.

103. What is Virtual Memory?

Virtual memory is a technique that allows the execution of processes that may not be completely in memory. A separation of user logical memory from physical memory allows an extremely large virtual memory to be provided for programmers when only a smaller physical memory is available. It is commonly implemented by demand paging. A demand paging system is similar to a paging system with swapping. Processes reside on secondary memory (which is usually a disk). When we want to execute a process, we swap it into memory.

104. What type of scheduling is used in Unix

Multi Level Feedback Queue Scheduling with each queue in round robin

105. where is kernel located in HP unix?

kernel will be middle layer, i.e between user level and hardware level layers.

106. Which command is used to delete all files in the current directory and all its sub-directories?

Naresh.surampudi

rm -fr

107. Write a command to find all of the files which have been accessed within the last 30 days.

`find / -type f -atime -30 > December.files` This command will find all the files under root, which is `/?/?`, with file type is file. `?-atime -30` will give all the files accessed less than 30 days ago. And the output will put into a file call `December.files`.

108. Write a command to kill the last background job?

Kill `#!`

109. Write a shell script to list only the hidden files in the current directories.

`ls -A | grep ^[.]` Works on Linux `ls -a | grep ^[.]` `ls -la | grep -v ^d | awk {'print $9'} | grep ^[.]`

nare700@gmail.com