Linux Operating System and Programming

TOPIC# 3

General Purpose Utilities

The general-purpose utilities have diverse functions and are broadly divided into two categories:

- They act handy accessories that can perform calculations or handle your mail.
- They tell the state the system is in and even manipulate it.

cal: The Calendar

$ cal [[month] year]

It displays the calendar of the current month. When this command is used with arguments, the month is optional but the year is not.

Example:

$ cal

This displays calendar of current month of current year.

$ cal 08 1993

This displays calendar of August month of year 1993.

$ cal 1993

This displays calendar of the year 1993.

date: Displaying the system date

It displays the date and time of system.

$ date

This command can also be used with suitable format specifiers. Each format is preceded by the + symbol, followed by the % operator and a single character describing the format.

Format	Description
+%m	Prints only month
+%h	Prints only name of the month
+%d	Prints the day of the month (1 to 31)
+%y	Prints the last two digits of the year
+%H	Prints the hours
+%M	Prints the minutes
+%S	Prints the seconds

+%D	Prints the date in mm/dd/yy format
+%T	Prints the time in hh:mm:ss format

One can use multiple format specifiers at a time. For that, they must be enclosed with single or double quotes and there must be a single + symbol before it.

echo: displaying a message

This command can be used to:

- To display a message
- To evaluate shell variables

$echo HelloWorld

An escape sequence is generally a two character-string beginning with a \(backslash).
Escape sequences used by **echo** and **printf**

\a	bell
\b	Backspace
\c	No newline (cursor in same line)
\f	formfeed
\n	Newline
\r	carriage return
\t	Tab
\v	Vertical tab
\\	Backslash
\on	ASCII character represented by the octal value n, where n can't exceed 0377 (decimal value 255)

Escape sequences are not directory supported by bash echo command and hence you may use echo -e option. You may also, try using csh shell. To make echo behave in the normal manner, place the statement "shopt -s xpg_echo" in your rc file. Probably ~/.bashrc

Escape sequences are example where system need to assing a special meaning to a character.

For example in html,

```
& becomes &amp;
< becomes &lt;
> becomes &gt;
```

If you wish that escape sequences to be part of the message, you may put one extra backward slash.

i.e. echo “Studnets, \\n is an example of escape sequence for newline.”

printf : an alternative to echo

This command is used to display a message.
Example
$ printf “hello\tworld\n”
hello world

There are some **placeholders** that can be used with printf.

%s	String
%d	Decimal integer
%o	Octal integer
%f	Floating point number
%x	Hexadecimal integer

Examples:
$ printf “The shell is %s\n” $SHELL
The shell is /usr/bin/bash

$ printf “The value of 225 is %o in octal and %x in hexadecimal\n” 255 255
The value of 255 is 377 in octal and ff in hexadecimal

bc: The calculator

UNIX provides two types of calculators- a graphical object (the **xcalc** command) that looks like one and the text-based **bc** command.

Example:
$ bc
23 + 5
28
[ctrl + d]
$

bc performs only integer computations and truncates the decimal portion that it sees.
Example
$ bc

```
10/3
3
```

To enable floating- point computations, set scale to the number of digits of precision before keying in the expression.

```
scale=2
17/7
2.42
```

bc command can also be used to convert numbers from one base to another.

```
ibase=2
110
6
```
Here, input base is 2 (binary) and output base is 10 (decimal)

```
obase=2
14
1110
```
Here, input base is 10 (decimal) and output base is 2 (binary)

script: recording your session

Script makes a typescript of everything printed on your terminal. If the argument file is given, script saves all dialogue in file if no file name is given; the typescript is saved in the file typescript. The script ends when the forked shell exits (a control-D to exit the Bourne shell (sh (1)) and exit, logout or control-d (if ignored of is not set) for the C-shell, csh (1)).

```
$ script
Script started, file is typescript
$ _

$ exit
Script done, file is typescript
$
```

passwd: Changing your password

passwd expects the user to respond three times. First, it prompts for the old password. Next, it checks whether you have entered a valid password, and if you have, it then prompts for the new password. Enter the new password. Finally, passwd asks you to reenter the new password.

who: who are the users?

UNIX maintains an account of all users who are logged on to the system. The **who** command displays an informative listing of these users.

```
$ who
```
This displays usernames (user-ids), device names, date and time of logging in and also shows the machine name from where the user has logged in.

$ who am i
This displays the information of current user of the system and not all the users.

$ whoami
This displays the username of the current user of the system.

uname: knowing your machine's characteristics

This command displays certain features of the operating system running on the machine.

$ uname
This displays the name of the operating system.

$ uname –r
This displays the version of the operating system.

$ uname –n
This displays the machine name or domain name of the system if and when connected in a network.

tty: knowing your terminal

It is a teletype command that tell you the filename of the terminal you are using.
$tty
/dev/pts/10

See that as linux every device memory is a file, for example display/screen of terminal, when you run cp file1 <yourttyoutput>, it displays it on the sreen.

mail command using postfix:

http://www.postfix.org/

This utility helps send out mail and also check inbox for a user.
Mostly, this can be used for monitoring shell scripts on a linux server to send status email or other email notification via mail command.

whoami

user1

cat /etc/hosts

127.0.0.1 localhost.localdomain localhos

As root:

yum list | grep postfix

postfix.i386 2:2.3.3-6.el5 installed

postfix-pflogsumm.i386 2:2.3.3-6.el5 base

/etc/init.d/postfix status

master (pid 10147) is running...

mail -s "hi" user1@localhost.localdomain

hi there, this is a test email body.

<Ctrl+d>

Cc: <enter>

To Send file content to mail body, redirect file as input.

mail -s "hi... demo.txt content is embedded as body herewith" user1@localhost.localdomain < demo.txt

To check mail inbox for user1@localhost.localdomain run command mail on the prompt:

mail

Type exit to come out of mail client.

FTP (File Transfer Protocol utility):

There is client-server model involved in this. It is used to transfer files over network. Let's say if administrator already have a ftp site and guest/anonymous access:

ftp command in linux by default act as ftp client. See below output:

[user1@centos ~]$ **ftp ftp.ddu.ac.in**

Connected to ftp.ddu.ac.in.

220 (vsFTPd 2.2.2)

530 Please login with USER and PASS.

530 Please login with USER and PASS.

KERBEROS_V4 rejected as an authentication type

Name (ftp.ddu.ac.in:user1): **anonymous**

331 Please specify the password.

Password:

230 Login successful.

Remote system type is UNIX.

Using binary mode to transfer files.

ftp> pwd

257 "/"

ftp> dir

227 Entering Passive Mode (192,168,1,101,54,88).

150 Here comes the directory listing.

-rw-r--r-- 1 0 0 454656 Jul 07 2012 PUTTY.EXE

-rw-r--r-- 1 0 0 236 Jul 07 2012 boot.ini

-rw-r--r-- 1 0 0 250032 Jul 07 2012 ntldr

drwxr-xr-x 11 0 0 4096 Aug 25 2012 pub

drwxrwxrwx 2 0 0 4096 Aug 01 09:00 uploads

226 Directory send OK.

ftp> help

Commands may be abbreviated. Commands are:

!	cr	mdir	proxy	send
$	delete	mget	sendport	site
account	debug	mkdir	put	size
append	dir	mls	pwd	status
ascii	disconnect	mode	quit	struct
bell	form	modtime	quote	system
binary	get	mput	recv	sunique

bye glob newer reget tenex

case hash nmap rstatus trace

ccc help nlist rhelp type

cd idle ntrans rename user

cdup image open reset umask

chmod lcd passive restart verbose

clear ls private rmdir ?

close macdef prompt runique

cprotect mdelete protect safe

ftp> help get

get receive file

ftp> htlp mget

?Invalid command

ftp> help mget

mget get multiple files

ftp> bye

221 Goodbye.

[user1@centos ~]$

Quoting

Quoting is used to accomplish two goals:

1. To control (i.e., limit) substitutions/values of variables and
2. To perform grouping of words.

i.e

message=”Hi there, how are you?”

```
TITLE="System Information for $HOSTNAME"
```

In this case, the text is surrounded by double quote characters. The reason we use quoting is to group the words together. If we did not use quotes, bash would think all of the words after the first one were additional commands. Try this:

```
[me@linuxbox me]$ TITLE=System Information for $HOSTNAME
```

Single and double quotes

The shell recognizes both single and double quote characters. The following are equivalent:

```
var="this is some text"
var='this is some text'
```

However, **there is an important difference between single and double quotes. Single quotes limit substitution.** As we saw in the previous lesson, you can place variables in double quoted text and the shell still performs substitution. We can see this with the echo command:

```
[me@linuxbox me]$ echo "My host name is $HOSTNAME."
My host name is linuxbox.
```

If we change to single quotes, the behavior changes:

```
[me@linuxbox me]$ echo 'My host name is $HOSTNAME.'
My host name is $HOSTNAME.
```

Curtsey: http://linuxcommand.org/wss0060.php

BACK QUOTE / backtick

There is one more linux symbol ` [ back quote ` -top left character on keyboard under esc, character on key tilt ~], is used in a special manner in linux as below:

Let's say that following searches for pattern 'localhost' and displays file names with path in current directory and subdirectories:

grep -lr localhost .

Now, I wish to open all these files in vi editor then I can do as below:

vi `grep -lr mail . `

Backtick is not a quotation sign, it has a very special meaning. Everything you type between backticks is evaluated (executed) by the shell before the main command (like `grep` in above example), and the *output* of that execution is used by that command, just as if you'd type that output at that place in the command line.

Contributors (CE Department, DDU):

- Prof. Niyati J. Buch
- Prof. Jigar M. Pandya