



Linux System Administration

Execute Commands with the Bash Shell

Summer Training June 2024

Pushpendra Kumar Pateriya

Lovely Professional University

✉ pushpendra.mnnit@gmail.com

🌐 [pushpendrapateriya](https://github.com/pushpendrapateriya)

🐦 [@ppsgsits](https://twitter.com/ppsgsits)

Objective(s)

- Execute basic shell commands effectively.
- Save time when running commands from a shell prompt with Bash shortcuts.

Basic Command Syntax

- The GNU Bourne-Again Shell (bash) interprets user-typed commands. Commands consist of three parts:
 - **Command:** Commands are program names installed on the system.
 - **Options** (usually prefixed with a hyphen or double hyphen)
 - **Arguments**
- Execute commands by pressing Enter. Output is displayed before the next shell prompt.

Command Line Prompt

```
[user@host ~]$ whoami  
user  
[user@host ~]$
```

We can type more than one command on a single line

To type multiple commands on a single line, use the semicolon (;) as a separator. This special character allows bash to interpret and execute each command sequentially, displaying the output of both commands before the next shell prompt.

The following example shows how to combine two commands (command1 and command2) on the command line.

Command Line Prompt

```
[user@host ~]$ command1 ; command2  
command1 output  
command2 output  
[user@host ~]$
```

Write Simple Commands

date Command:

The date command displays the current date and time. The superuser or a privileged user can also use the date command to set the system clock. Use the plus sign (+) as an argument to specify a format string for the date command.

Command Line Prompt

```
[user@host ~]$ date  
Sun Feb 27 08:32:42 PM EST 2022  
[user@host ~]$ date +%R  
20:33  
[user@host ~]$ date +%x  
02/27/2022
```

passwd Command

- The passwd command changes the current user's password.
- First, enter the original password, then the new password, which must be strong (including lowercase, uppercase, numbers, and symbols).
- Superusers can change other users' passwords.

Command Line Prompt

```
[user@host ~]$ passwd
Changing password for user user.
Current password:  old_password
New password:  new_password
Retype new password:  new_password
passwd:  all authentication tokens updated successfully.
```

file Command

- Linux does not require file name extensions to classify files by type.
- The file command scans the compiled header of a file for a 2-digit magic number and displays its type.
- Text files are recognized because they are not compiled.

Command Line Prompt

```
[user@host ~]$ file /etc/passwd
```

```
/etc/passwd: ASCII text
```

```
[user@host ~]$ file /bin/passwd
```

```
/bin/passwd: setuid ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=a467cb9c8fa7306d4 for GNU/Linux 3.2.0, stripped
```

```
[user@host ~]$ file /home
```

```
/home: directory
```

View the Contents of Files using 'cat' Command

- The cat command is often used in Linux. Use this command to create single or multiple files, view the contents of files, concatenate the contents from various files, and redirect contents of the file to a terminal or to files.
- The following example shows how to view the contents of the /etc/passwd file:

Command Line Prompt

```
[user@host ~]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
...output omitted...
```


Display content of multiple files using cat command

To display the contents of multiple files, add the file names to the cat command as arguments:

Command Line Prompt

```
[user@host ~]$ cat file1 file2  
Hello World!!  
Introduction to Linux commands.
```

Display content of longer files

Some files are long and might need more space to be displayed than the terminal provides. The `cat` command does not display the contents of a file as pages. The `less` command displays one page of a file at a time and you can scroll at your leisure.

Use the `less` command to page forward and backward through longer files than can fit on one terminal window. Use the UpArrow key and the DownArrow key to scroll up and down. Press `q` to exit the command.

The `head` and `tail` commands display the beginning and the end of a file, respectively. By default, these commands display 10 lines of the file, but they both have a `-n` option to specify a different number of lines.

Display content of longer files using 'head' command

Command Line Prompt

```
[user@host ~]$ head /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
```

Display content of longer files using 'tail' command

Following command displays last 3 line of /etc/passwd file.

Command Line Prompt

```
[user@host ~]$ tail -n 3 /etc/passwd
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:980:978::/run/gnome-initial-
setup/:/sbin/nologin
dnsmasq:x:979:977:Dnsmasq DHCP and DNS server:/var
```

wc command

The `wc` command counts lines, words, and characters in a file. Use the `-l`, `-w`, or `-c` options to display only the given number of lines, words, or characters, respectively.

Command Outputs

```
[user@host ~]$ wc /etc/passwd
```

```
41 98 2338 /etc/passwd
```

```
[user@host ~]$ wc -l /etc/passwd ; wc -l /etc/group
```

```
41 /etc/passwd
```

```
63 /etc/group
```

```
[user@host ~]$ wc -c /etc/group /etc/hosts
```

```
883 /etc/group
```

```
114 /etc/hosts
```

```
997 total
```

Understand Tab Completion

With tab completion, users can quickly complete commands or file names after typing enough at the prompt to make it unique. If the typed characters are not unique, then pressing the Tab key twice displays all commands that begin with the typed characters.

Command Outputs

```
[user@host ~]$ pasTab+Tab1
passwd      paste      pasuspender
[user@host ~]$ passTab2
[user@host ~]$ passwd
Changing password for user user.
Current password:
```

1. Press Tab twice.
2. Press Tab once.

Understand Tab Completion (continued)

- Tab completion helps to complete file names when typing them as arguments to commands.
- Press Tab to complete as much of the file name as possible.
- Pressing Tab a second time causes the shell to list all files that the current pattern matches.
- Type additional characters until the name is unique, and then use tab completion to complete the command.

Command Outputs

```
[user@host ~]$ ls /etc/pasTab1  
[user@host ~]$ ls /etc/passwdTab2  
passwd      passwd-
```

1. Press Tab once.
2. Press Tab once.

Completion of the option name with minimal typing using tab

It is difficult to remember all the options supported by a command. Utilize tab completion to easily remember and complete options.

Command Outputs

```
[root@host ~]# useradd -Tab+Tab1
-badnames      -gid           -no-log-init   -shell
-base-dir      -groups        -non-unique    -skel
-btrfs-subvolume-home -help         -no-user-group
system
-comment       -home-dir      -password      -uid
-create-home   -inactive      -prefix        -user-
group
-defaults      -key           -root
-expiredate    -no-create-home -selinux-user
```

1. Press Tab twice.

Write a Long Command on Multiple Lines

When commands get long with lots of options, they might wrap to the next line in the command window. To make them easier to read, you can split a long command across multiple lines.

1. Start typing your command on the first line.
2. When you reach the end of the line or want to break the command into multiple lines for readability, insert a backslash (`\`) at the end of the line.
3. Press Enter to move to the next line while continuing the same command.
4. Continue typing the rest of your command on the new line.
5. Repeat steps 2-4 as needed to complete your command.

Write a Long Command on Multiple Lines (continued)

Example

```
[user@host ~]$ head -n 3 \  
/usr/share/dict/words \  
/usr/share/dict/linux.words  
==> /usr/share/dict/words <==  
1080  
10-point  
10th  
==> /usr/share/dict/linux.words <==  
1080  
10-point  
10th
```

The Command History

The history command displays a list of previously executed commands that are prefixed with a command number.

- **View History:** Simply type history in the terminal to see a list of your recently executed commands along with their line numbers.
- **Search History:** Utilize Ctrl + R to search through your command history. As you type, it will auto-fill the most recent command that matches what you've typed so far.
- **Execute Commands by Number:** You can rerun commands from history by typing !n, where n is the line number of the command you want to execute.
- **Execute Last Command:** Use !! to rerun the last command you executed.
- **Execute Previous Command Starting With:** If you want to execute the last command that started with a specific character or string, type !<character> or !<string>.
- **Execute Previous Command with Specific String:** You can also execute the last command containing a specific string by typing !?<string>.
- **Clear History:** If you want to clear your command history, use the command history -C.
- **Ignore Certain Commands:** You can tell history not to remember certain commands by starting them with a space. For example, <space>ls instead of ls.

The Command History

Example

```
[user@host ~]$ history
...output omitted...
23 clear
24 who
25 pwd
26 ls /etc
27 uptime
28 ls -l
29 date
30 history
[user@host ~]$ !ls
ls -l
total 0
drwxr-xr-x.  2 student student 6 Feb 27 19:24 Desktop
...output omitted...
[user@host ~]$ !26
```

Navigate through previous commands

The arrow keys help you move through past commands. UpArrow goes to the previous command, DownArrow goes to the next. LeftArrow and RightArrow let you move around in the current command to edit it.

To quickly insert the last word from a previous command, press Esc+. or Alt+. together. You can keep pressing Alt+. to cycle through earlier commands.

Edit the Command Line

Table: Useful Command-line Editing Shortcuts

Shortcut	Description
Ctrl + A	Move to the beginning of the line
Ctrl + E	Move to the end of the line
Ctrl + U	Cut everything before the cursor
Ctrl + K	Cut everything after the cursor
Ctrl + W	Cut the word before the cursor
Ctrl + Y	Paste the last thing you cut
Ctrl + L	Clear the screen
Ctrl + C	Cancel the current command
Ctrl + D	Logout (exit the shell if no text is entered)
Ctrl + R	Search backward through history for a command
!!	Repeat the last command
!	Last argument from the previous command
!^	First argument from the previous command
Alt + Backspace	Delete the word before the cursor
Alt + F	Move cursor forward one word
Alt + B	Move cursor backward one word

Question

Which Bash command displays the last five lines of the `/var/log/messages` file?

- A. `head -n 10 /var/log/messages`
- B. `tail 10 /var/log/messages`
- C. `tail -n 5 /var/log/messages`
- D. `tail -l 10 /var/log/messages`
- E. `less /var/log/messages`

Answer

C. `tail -n 5 /var/log/messages`

Question

Which Bash shortcut or command separates commands on the same line?

- A. Pressing Tab
- B. history
- C. ;
- D. !string
- E. Pressing Esc+.

Answer

C. ;

Question

Which Bash command is used to change a user's password?

- A. password
- B. pass
- C. passwd
- D. usermod
- E. userpassword

Answer

C. passwd

Question

Which Bash command is used to display the file type?

- A. file
- B. less
- C. cat
- D. history
- E. view

Answer

Answer

A. file

Question

Which Bash shortcut or command is used for completing commands, file names, and options?

- A. ;
- B. !number
- C. history
- D. Pressing Tab
- E. Pressing Esc+.

Answer

D. Pressing Tab

Question

Which Bash shortcut or command re-executes a specific command in the history list?

- A. Pressing Tab
- B. !number
- C. !string
- D. history
- E. Pressing Esc+.

Answer

B. !number

Question

Which Bash shortcut or command jumps to the beginning of the command line?

- A. !number
- B. !string
- C. Pressing Ctrl+LeftArrow
- D. Pressing Ctrl+K
- E. Pressing Ctrl+A

Answer

E. Pressing Ctrl+A

Question

Which Bash shortcut or command displays the list of previously executed commands?

- A. Pressing Tab
- B. !string
- C. !number
- D. history
- E. Pressing Esc+.

Answer

D. history

Question

Which Bash shortcut or command copies the last argument of previous commands?

- A. Pressing Ctrl+K
- B. Pressing Ctrl+A
- C. !number
- D. Pressing Esc+.

Answer

D. Pressing Esc+.

Thanks!

Thank you
for your attention.