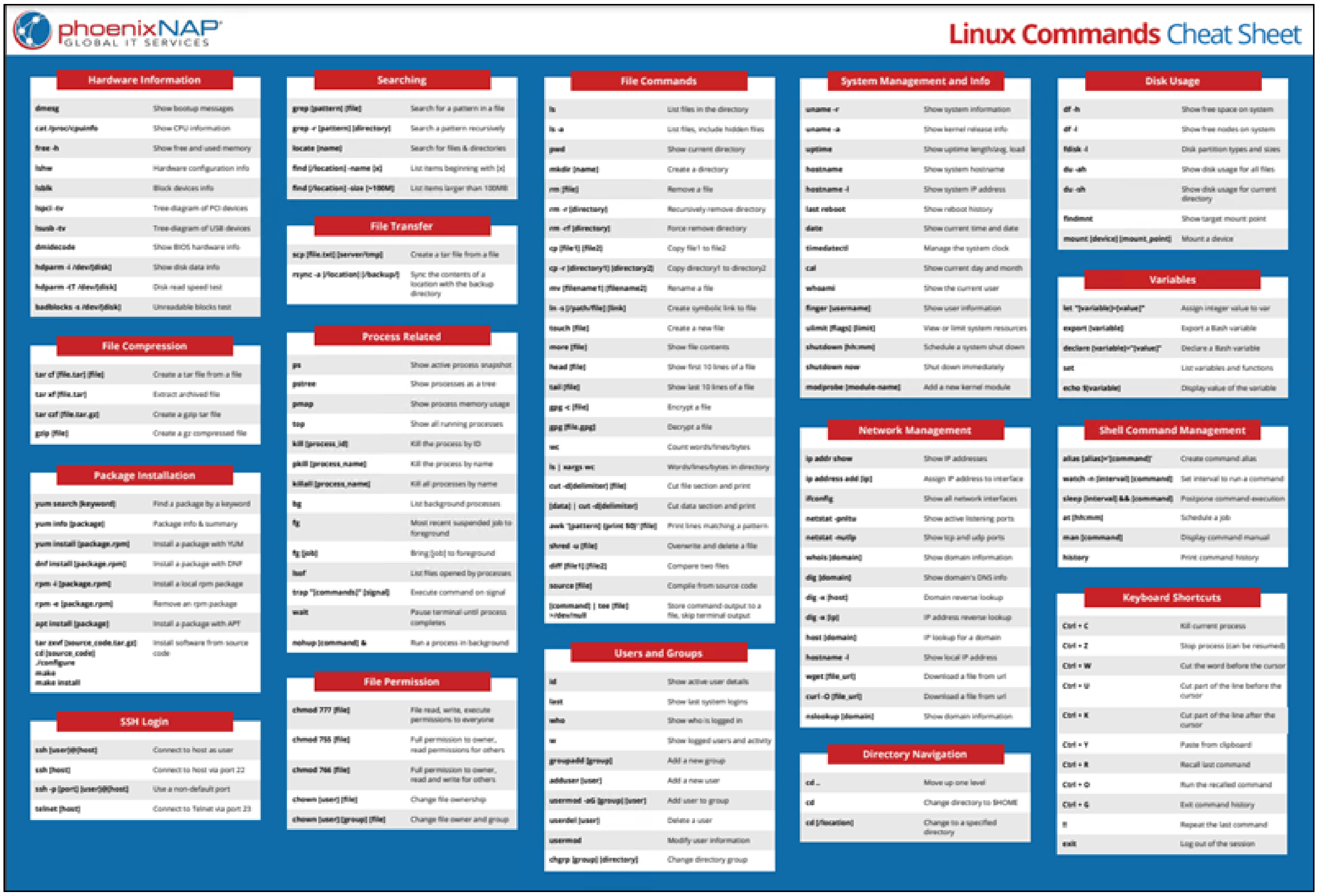

phoenixNAP GLOBAL IT SERVICES

Linux Commands Cheat Sheet

Hardware Information

dmesg	Show bootup messages
cat /proc/cpuinfo	Show CPU information
free -h	Show free and used memory
lshw	Hardware configuration info
lsblk	Block devices info
lspci -tv	Tree diagram of PCI devices
lsusb -tv	Tree diagram of USB devices
dmidecode	Show BIOS hardware info
hdparm -i /dev/[disk]	Show disk data info
hdparm -tT /dev/[disk]	Disk read speed test
badblocks -s /dev/[disk]	Unreadable blocks test

File Compression

tar cf [file.tar] [file]	Create a tar file from a file
tar xf [file.tar]	Extract archived file
tar czf [file.tar.gz]	Create a gzip tar file
gzip [file]	Create a gz compressed file

Package Installation

yum search [keyword]	Find a package by a keyword
yum info [package]	Package info & summary
yum install [package.rpm]	Install a package with YUM
dnf install [package.rpm]	Install a package with DNF
rpm -i [package.rpm]	Install a local rpm package
rpm -e [package.rpm]	Remove an rpm package
apt install [package]	Install a package with APT
tar zxvf [source_code.tar.gz] cd [source_code] ./configure make make install	Install software from source code

SSH Login

ssh [user]@[host]	Connect to host as user
ssh [host]	Connect to host via port 22
ssh -p [port] [user]@[host]	Use a non-default port
telnet [host]	Connect to Telnet via port 23

Searching

grep [pattern] [file]	Search for a pattern in a file
grep -r [pattern] [directory]	Search a pattern recursively
locate [name]	Search for files & directories
find [/location] -name [x]	List items beginning with [x]
find [/location] -size [+100M]	List items larger than 100MB

File Transfer

scp [file.txt] [server/tmp]	Create a tar file from a file
rsync -a [/location] [/backup/]	Sync the contents of a location with the backup directory

Process Related

ps	Show active process snapshot
pstree	Show processes as a tree
pmap	Show process memory usage
top	Show all running processes
kill [process_id]	Kill the process by ID
pkill [process_name]	Kill the process by name
killall [process_name]	Kill all processes by name
bg	List background processes
fg	Most recent suspended job to foreground
fg [job]	Bring [job] to foreground
lsof	List files opened by processes
trap "[commands]" [signal]	Execute command on signal
wait	Pause terminal until process completes
nohup [command] &	Run a process in background

File Permission

chmod 777 [file]	File read, write, execute permissions to everyone
chmod 755 [file]	Full permission to owner, read permissions for others
chmod 766 [file]	Full permission to owner, read and write for others
chown [user] [file]	Change file ownership
chown [user]:[group] [file]	Change file owner and group

File Commands

ls	List files in the directory
ls -a	List files, include hidden files
pwd	Show current directory
mkdir [name]	Create a directory
rm [file]	Remove a file
rm -r [directory]	Recursively remove directory
rm -rf [directory]	Force remove directory
cp [file1] [file2]	Copy file1 to file2
cp -r [directory1] [directory2]	Copy directory1 to directory2
mv [filename1] [filename2]	Rename a file
ln -s [/path/file] [link]	Create symbolic link to file
touch [file]	Create a new file
more [file]	Show file contents
head [file]	Show first 10 lines of a file
tail [file]	Show last 10 lines of a file
gpg -c [file]	Encrypt a file
gpg [file.gpg]	Decrypt a file
wc	Count words/lines/bytes
ls | xargs wc	Words/lines/bytes in directory
cut -d[delimiter] [file]	Cut file section and print
[data] | cut -d[delimiter]	Cut data section and print
awk '[pattern] {print $0}' [file]	Print lines matching a pattern
shred -u [file]	Overwrite and delete a file
diff [file1] [file2]	Compare two files
source [file]	Compile from source code
[command] | tee [file] >/dev/null	Store command output to a file, skip terminal output

Users and Groups

id	Show active user details
last	Show last system logins
who	Show who is logged in
w	Show logged users and activity
groupadd [group]	Add a new group
adduser [user]	Add a new user
usermod -aG [group] [user]	Add user to group
userdel [user]	Delete a user
usermod	Modify user information
chgrp [group] [directory]	Change directory group

System Management and Info

uname -r	Show system information
uname -a	Show kernel release info
uptime	Show uptime length/avg. load
hostname	Show system hostname
hostname -I	Show system IP address
last reboot	Show reboot history
date	Show current time and date
timedatectl	Manage the system clock
cal	Show current day and month
whoami	Show the current user
finger [username]	Show user information
ulimit [flags] [limit]	View or limit system resources
shutdown [hh:mm]	Schedule a system shut down
shutdown now	Shut down immediately
modprobe [module-name]	Add a new kernel module

Network Management

ip addr show	Show IP addresses
ip address add [ip]	Assign IP address to interface
ifconfig	Show all network interfaces
netstat -pnltu	Show active listening ports
netstat -nutlp	Show tcp and udp ports
whois [domain]	Show domain information
dig [domain]	Show domain's DNS info
dig -x [host]	Domain reverse lookup
dig -x [ip]	IP address reverse lookup
host [domain]	IP lookup for a domain
hostname -I	Show local IP address
wget [file_url]	Download a file from url
curl -O [file_url]	Download a file from url
nslookup [domain]	Show domain information

Directory Navigation

cd ..	Move up one level
cd	Change directory to $HOME
cd [/location]	Change to a specified directory

Disk Usage

df -h	Show free space on system
df -i	Show free nodes on system
fdisk -l	Disk partition types and sizes
du -ah	Show disk usage for all files
du -sh	Show disk usage for current directory
findmnt	Show target mount point
mount [device] [mount_point]	Mount a device

Variables

let "[variable]=[value]"	Assign integer value to var
export [variable]	Export a Bash variable
declare [variable]="[value]"	Declare a Bash variable
set	List variables and functions
echo $[variable]	Display value of the variable

Shell Command Management

alias [alias]="[command]"	Create command alias
watch -n [interval] [command]	Set interval to run a command
sleep [interval] && [command]	Postpone command execution
at [hh:mm]	Schedule a job
man [command]	Display command manual
history	Print command history

Keyboard Shortcuts

Ctrl + C	Kill current process
Ctrl + Z	Stop process (can be resumed)
Ctrl + W	Cut the word before the cursor
Ctrl + U	Cut part of the line before the cursor
Ctrl + K	Cut part of the line after the cursor
Ctrl + Y	Paste from clipboard
Ctrl + R	Recall last command
Ctrl + O	Run the recalled command
Ctrl + G	Exit command history
!!	Repeat the last command
exit	Log out of the session

Linux Commands List

The commands found in the downloadable cheat sheet are listed below.

Hardware Information

Show **bootup messages**:

```
dmesg
```

See **CPU information**:

```
cat /proc/cpuinfo
```

Display **free and used memory** with:

```
free -h
```

List **hardware configuration** information:

```
lshw
```

See information about **block devices**:

```
lsblk
```

Show **PCI devices** in a tree-like diagram:

```
lspci -tv
```

Display **USB devices** in a tree-like diagram:

```
lsusb -tv
```

Show **hardware information** from the BIOS:

```
dmidecode
```

Display **disk data** information:

```
hdparm -i /dev/disk
```

Conduct a **read-speed test** on device/disk:

```
hdparm -tT /dev/[device]
```

Test for **unreadable blocks** on device/disk:

```
badblocks -s /dev/[device]
```

Run a disk check on an unmounted disk or partition:

```
fsck [disk-or-partition-location]
```

Searching

Search for a specific pattern in a file with grep:

```
grep [pattern] [file_name]
```

Recursively search for a pattern in a directory:

```
grep -r [pattern] [directory_name]
```

Find all files and directories **related to a particular name**:

```
locate [name]
```

List names that **begin with a specified character** **`[a]`** in a specified location **`[/folder/location]`** by using the **`find`** command:

```
find [/folder/location] -name [a]
```

See **files larger than a specified size** **`[+100M]`** in a folder:

```
find [/folder/location] -size [+100M]
```

Note: Some commands are not recommended to use. Learn about them in our list of dangerous Linux commands.

File Commands

List files in the directory:

```
ls
```

List all files (shows hidden files):

```
ls -a
```

Show directory you are currently working in:

```
pwd
```

Create a new directory:

```
mkdir [directory]
```

Remove a file:

```
rm [file_name]
```

Remove a directory recursively:

```
rm -r [directory_name]
```

Recursively remove a directory without requiring confirmation:

```
rm -rf [directory_name]
```

Copy the contents of one file to another file:

```
cp [file_name1] [file_name2]
```

Recursively copy the contents of one file to a second file:

```
cp -r [directory_name1] [directory_name2]
```

Rename `[file_name1]` to **`[file_name2]`** with the command:

```
mv [file_name1] [file_name2]
```

Create a symbolic link to a file:

```
ln -s /path/to/[file_name] [link_name]
```

Create a **new file** using touch:

```
touch [file_name]
```

Show the contents of a file:

```
more [file_name]
```

or use the **`cat`** command:

```
cat [file_name]
```

Append file contents to another file:

```
cat [file_name1] >> [file_name2]
```

Display the **first 10 lines** of a file with head command:

```
head [file_name]
```

Show the **last 10 lines** of a file with tail command:

```
tail [file_name]
```

Encrypt a file:

```
gpg -c [file_name]
```

Decrypt a file:

```
gpg [file_name.gpg]
```

Show the **number of words, lines, and bytes** in a file using wc:

```
wc
```

List number of lines/words/characters in each file in a directory with the xargs command:

```
ls | xargs wc
```

Cut a section of a file and print the result to standard output:

```
cut -d[delimiter] [filename]
```

Cut a section of piped data and print the result to standard output:

```
[data] | cut -d[delimiter]
```

Print all lines matching a pattern in a file:

```
awk '[pattern] {print $0}' [filename]
```

Note: Learn also about gawk command, the GNU version of awk.

Overwrite a file to prevent its recovery, then delete it:

```
shred -u [filename]
```

Compare two files and display differences:

```
diff [file1] [file2]
```

Read and execute the file content in the current shell:

```
source [filename]
```

Sort file contents and print the result in standard output:

```
sort [options] filename
```

Store the command output in a file and skip the terminal output:

```
[command] | tee [filename] >/dev/null
```

Note: Want to read more about file creation? Check out an article about how to create a file in Linux using the command line.
And if you want to find out how to determine the type of a file and its data, read our article about Linux file command.
To view a file's contents one screen at a time read about less command in Linux.

Directory Navigation

Move **up one level** in the directory tree structure:

```
cd ..
```

Change directory **to `$HOME`**:

```
cd
```

Change location to a specified directory:

```
cd /chosen/directory
```

File Compression

Archive an existing file:

```
tar cf [compressed_file.tar] [file_name]
```

Extract an archived file:

```
tar xf [compressed_file.tar]
```

Create a **gzip compressed tar file** by running:

```
tar czf [compressed_file.tar.gz]
```

Compress a file with the **`.gz`** extension:

```
gzip [file_name]
```

Note: For a more comprehensive overview of how to use **`tar`** refer to our guide tar Command in Linux With Examples.

File Transfer

Copy a file to a server directory securely using the Linux scp command:

```
scp [file_name.txt] [server/tmp]
```

Synchronize the contents of a directory **with a backup directory** using the rsync command:

```
rsync -a [/your/directory] [/backup/]
```

Users and Groups

See details about the **active users**:

```
id
```

Show **last system logins**:

```
last
```

Display who is **currently logged into the system** with the who command:

```
who
```

Show which users are **logged in** and **their activity**:

```
w
```

Add a new group by typing:

```
groupadd [group_name]
```

Add a **new user**:

```
adduser [user_name]
```

Add a **user to a group**:

```
usermod -aG [group_name] [user_name]
```

Temporarily **elevate user privileges** to superuser or root using the sudo command:

```
sudo [command_to_be_executed_as_superuser]
```

Delete a user:

```
userdel [user_name]
```

Modify user information with:

```
usermod
```

Change directory group:

```
chgrp [group-name] [directory-name]
```

Note: If you want to learn more about users and groups, take a look at our article on how to add a user to a group in Linux.

Package Installation

List all installed packages with **yum**:

```
yum list installed
```

Find a package by a **related keyword**:

```
yum search [keyword]
```

Show **package information and summary**:

```
yum info [package_name]
```

Install a package using the **YUM package manager**:

```
yum install [package_name.rpm]
```

Install a package using the **DNF package manager**:

```
dnf install [package_name.rpm]
```

Install a package using the **APT package manager**:

```
apt install [package_name]
```

Install an **`.rpm`** package from a local file:

```
rpm -i  [package_name.rpm]
```

Remove an **`.rpm`** package:

```
rpm -e [package_name.rpm]
```

Install software from **source code**:

```
tar zxvf [source_code.tar.gz]
cd [source_code]
./configure
make
make install
```

Process Related

See a **snapshot of active processes**:

```
ps
```

Show **processes in a tree-like diagram**:

```
pstree
```

Display a **memory usage map** of processes:

```
pmap
```

See all running processes:

```
top
```

Terminate a Linux process under a **given ID**:

```
kill [process_id]
```

Terminate a process under a **specific name**:

```
pkill [proc_name]
```

Terminate all processes **labelled “proc”**:

```
killall [proc_name]
```

List and resume stopped jobs in the background:

```
bg
```

Bring the most **recently suspended job to the foreground**:

```
fg
```

Bring a **particular job to the foreground**:

```
fg [job]
```

List **files opened by running processes** with lsof command:

```
lsof
```

Catch a system error signal in a shell script:

```
trap "[commands-to-execute-on-trapping]" [signal]
```

Pause terminal or a Bash script until a running process is completed:

```
wait
```

Run a Linux process in the background:

```
nohup [command] &
```

Note: If you want to learn more about shell jobs, how to terminate jobs or keep them running after you log off, check out our article on how to use disown command.

System Management and Information

Show **system information** via uname command:

```
uname -r
```

See kernel release information:

```
uname -a
```

Display **how long the system has been running**, including load average:

```
uptime
```

See system **hostname**:

```
hostname
```

Show the **IP address** of the system:

```
hostname -i
```

List system **reboot history**:

```
last reboot
```

See current time and date:

```
date
```

Query and **change the system clock** with:

```
timedatectl
```

Show current **calendar** (month and day):

```
cal
```

List logged in users:

```
w
```

See which **user you are using**:

```
whoami
```

Show **information about a particular user**:

```
finger [username]
```

View or limit system resource amounts:

```
ulimit [flags] [limit]
```

Schedule a system shutdown:

```
shutdown [hh:mm]
```

Shut Down the system immediately:

```
shutdown now
```

Add a new kernel module:

```
modprobe [module-name]
```

Disk Usage

You can use the df and du commands to check disk space in Linux.

See **free and used space** on mounted systems:

```
df -h
```

Show **free inodes** on mounted filesystems:

```
df -i
```

Display **disk partitions, sizes, and types** with the command:

```
fdisk -l
```

See **disk usage** for all files and directory:

```
du -ah
```

Show **disk usage of the directory** you are currently in:

```
du -sh
```

Display **target mount point** for all filesystem:

```
findmnt
```

Mount a device:

```
mount [device_path] [mount_point]
```

SSH Login

Connect to host as user:

```
ssh user@host
```

Securely **connect to host via SSH** default port 22:

```
ssh host
```

Connect to host **using a particular port**:

```
ssh -p [port] user@host
```

Connect to host **via telnet default port 23**:

```
telnet host
```

Note: For a detailed explanation of SSH Linux Commands, refer to our 19 Common SSH Commands in Linux tutorial.

File Permission

Chown command in Linux changes file and directory ownership.

Assign **read, write, and execute permission** to everyone:

```
chmod 777 [file_name]
```

Give **read, write, and execute permission to owner**, and r**ead and execute permission to group and others**:

```
chmod 755 [file_name]
```

Assign **full permission to owner**, and **read and write permission to group and others**:

```
chmod 766 [file_name]
```

Change the **ownership of a file**:

```
chown [user] [file_name]
```

Change the **owner and group ownership of a file**:

```
chown [user]:[group] [file_name]
```

Note: To learn more about how to check and change permissions, refer to our Linux File Permission Tutorial.

Network

List IP addresses and **network interfaces**:

```
ip addr show
```

Assign an **IP address to interface eth0**:

```
ip address add [IP_address]
```

Display **IP addresses of all network interfaces** with:

```
ifconfig
```

See **active (listening) ports** with the netstat command:

```
netstat -pnltu
```

Show **tcp** and **udp ports** and their programs:

```
netstat -nutlp
```

Display more **information about a domain**:

```
whois [domain]
```

Show **DNS information** about a domain using the dig command:

```
dig [domain]
```

Do a **reverse lookup on domain**:

```
dig -x host
```

Do **reverse lookup of an IP address**:

```
dig -x [ip_address]
```

Perform an **IP lookup for a domain**:

```
host [domain]
```

Show the **local IP address**:

```
hostname -I
```

Download a file from a domain using the `wget` command:

```
wget [file_name]
```

Receive information about an internet domain:

```
nslookup [domain-name]
```

Save a remote file to your system using the filename that corresponds to the filename on the server:

```
curl -O [file-url]
```

Variables

Assign an integer value to a variable:

```
let "[variable]=[value]"
```

Export a Bash variable:

```
export [variable-name]
```

Declare a Bash variable:

```
declare [variable-name]= "[value]"
```

List the names of all the shell variables and functions:

```
set
```

Display the value of a variable:

```
echo $[variable-name]
```

Shell Command Management

Create an alias for a command:

```
alias [alias-name]='[command]'
```

Set a custom interval to run a user-defined command:

```
watch -n [interval-in-seconds] [command]
```

Postpone the execution of a command:

```
sleep [time-interval] && [command]
```

Create a job to be executed at a certain time (**Ctrl+D** to exit prompt after you type in the command):

```
at [hh:mm]
```

Display a built-in manual for a command:

```
man [command]
```

Print the history of the commands you used in the terminal:

```
history
```

Linux Keyboard Shortcuts

Kill process running in the terminal:

```
Ctrl + C
```

Stop **current process**:

```
Ctrl + Z
```

The process can be **resumed** in the **foreground** with `fg` or in the **background** with **`bg`**.

Cut **one word before the cursor** and add it to clipboard:

```
Ctrl + W
```

Cut **part of the line before the cursor** and add it to clipboard:

```
Ctrl + U
```

Cut **part of the line after the cursor** and add it to clipboard:

```
Ctrl + K
```

Paste from clipboard:

```
Ctrl + Y
```

Recall last command that matches the provided characters:

```
Ctrl + R
```

Run the previously recalled command:

```
Ctrl + O
```

Exit command history without running a command:

```
Ctrl + G
```

Run the last command again:

```
!!
```

Log out of current session:

```
exit
```

Conclusion

The more you use Linux commands, the better you will get at remembering them. Do not stress about memorizing their syntax; use our cheat sheet.

Whenever in doubt, refer to this helpful guide for the most common Linux commands.

Was this article helpful? Yes No