Various helpful command line commands

System

Check whether a reboot is required:

```
ls /var/run/reboot-required
```

In a bash script, you can use:

```
#!/bin/bash
if [ -f /var/run/reboot-required ]; then
  echo 'reboot required'
fi
```

Copy files between servers

1. Copy between servers with user privileges

   ```
   $ rsync -avz -e 'ssh -p 22' dir user@x.x.x.x:/path/to/dir/
   ```

2. Copy between servers with root privileges

   ```
   $ rsync -avz -e 'ssh -p 22 -i /home/user/.ssh/id_ecdsa' --rsync-
   path="sudo rsync" dir user@x.x.x.x:/path/to/dir/
   ```

   - also enable the user on the target machine to run rsync with root privileges, to do so create file `/etc/sudoers.d/rsync` (line 1) with content (line 2)

     ```
     $ sudo visudo /etc/sudoers.d/rsync
     user ALL = NOPASSWD:/usr/bin/rsync
     ```

3. Copy from cloud

   ```
   $ rsync -avz -e 'ssh -p 22 -i /path/to/identity/file' --info=progress2
   user@x.x.x.x:/path/to/dir/ /local/dir/
   ```

File Management

Read very large text file

```
tail -n 100 ./file-name.txt
```

```
head -n 100 ./file-name.txt
```

Copy all files except some in all sub directories

```
cd ~/movies
find . -type d -exec mkdir ~/copied/{} \;
find . -type f -not -name *.mkv -exec cp {} ~/copied/{} \;
```

Modify file but preserve modified timestamp

```
cp -p <file> <file-with-timestamp>
vim <file>
touch -r <file-with-timestamp> <file>
```

Find all files containing specific text

- Since you have GNU find/xargs, this is a safe way for xargs to read filenames:

  ```
  find . -type f -print0 | xargs -0 grep "some string"
  ```

- If you only want the filenames that have a matching line without showing the matching line:

  ```
  find . -type f -print0 | xargs -0 grep -l "some string"
  ```

Find all files which were modified within the last month from the current directory

```
find . -newermt $(date +%Y-%m-%d -d '1 month ago') -type f -print
```

Find all files modified on the 7th of June, 2006

```
find . -type f -newermt 2007-06-07 ! -newermt 2007-06-08
```

Find all files accessed on the 29th of september, 2008

```
find . -type f -newerat 2008-09-29 ! -newerat 2008-09-30
```

Find files which had their permission changed on the same day

```
find . -type f -newerct 2008-09-29 ! -newerct 2008-09-30
```

Copy all files excluding hardlinks

```
rsync -aAXv /origin /dest
rsync -azAX -H --delete --numeric-ids /path/to/source /path/to/dest
```

Determine size of a folder path

```
du -hs
```

Find size of all sub-directories (only 1 level)

```
find -maxdepth 1 -type d -path ./proc -prune -o -exec du -hs "{}" \;
```

Change permissions for all files in a directory tree

```
find -type f -print0 | xargs -0 chmod 644
```

Copy all files

```
rsync -av <source> <dest>
```

Count all files in a directory (without .files)

```
ls -1 | wc -l
```

Change the name of all files like "source_1111_22.jpg" to "dest_1111_22.jpg"

```
rename 's/source_(\d{4}_\d{2})\.jpg$/dest_$1.jpg/' *
```

If “source” is “sourca” and “sourcb”, do the following:

```
rename 's/\w{6}_(\d{4}_\d{2})\.jpg$/dest_$1.jpg/' *
```

Delete Directory and Files within recursively

```
find -type d -name <dir> -exec rm -rf {} \;
```

Refresh display of the last 15 lines of a file every 2 seconds

```
watch tail -n 15 file.name
```

Zip folder

```
zip -r myarchive.zip dir1 -x dir1/ignoreDir1/**\* dir1/ignoreDir2/**\*
```

Linux rename command

Zip including hidden files

```
zip -r archive.zip * .[^.]*
```

Zip including hidden files

Disk Management

```
# iotop
smartctl -H /dev/sda
blkid /dev/sda1
mount -t cifs -o user=luke //192.168.1.104/share /mnt/linky_share
mdadm --manage /dev/md0 --add /dev/sdb1
```

Diskspace information:

```
df -h
```

VirtualBox

VirtualBox

```
usermod -a -G groupName userName
```

Add physical disk:

```
VBoxManage internalcommands createrawvmdk -filename
/home/bco/.VirtualBox/diskname.vmdk -rawdisk /dev/sda
```

Vim

Find each occurrence of 'foo' (in all lines), and replace it with 'bar'.:

```
:%s/foo/bar/g
```

Find each occurrence of 'foo' (in the current line only), and replace it with 'bar':

```
:s/foo/bar/g
```

Search and Replace

vi

To fix the broken arrow keys in insert mode in vi, change the vi default config as follows:

```
$ vi ~/.exrc
set nocompatible
```

Settings

```
hostname <newname>
vim /etc/hostname
vim /etc/hosts
```

```
vim /usr/share/X11/xorg.conf.d/00-Settings.conf
Section"ServerFlags"
Option "BlankTime" "600"
EndSection
```

Port scan

TCP Port Scan with Nmap

check for root privileges

Don't forget to change the root password. If any user has UID 0 besides root, they shouldn't. Bad idea. To check:

```
grep 'x:0:' /etc/passwd
```

Again, you shouldn't do this but to check if the user is a member of the root group:

```
grep root /etc/group
```

To see if anyone can execute commands as root, check sudoers:

```
cat /etc/sudoers
```

To check for SUID bit, which allows programs to be executed with root privileges:

```
find / -perm -04000
```

Add a service to the init system

Write a script with the LSB tags to be read by the insserv service:

```
#!/bin/sh
### BEGIN INIT INFO
# Provides: myservice
# Required-Start: sshd
# Required-Stop: sshd
# Should-Start:
# Should-Stop:
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: Start myservice
# Description: Start myservice after sshd is started (example)
### END INIT INFO
myservice -options
exit 0
```

Name the script myservice, move it to /etc/init.d/, and run the following command as root:

```
# update.rc-d myservice defaults
```

From:
https://wiki.condrau.com/ - **Bernard's Wiki**

Permanent link:
https://wiki.condrau.com/comp:commands

Last update: **2026/07/19 17:24**