

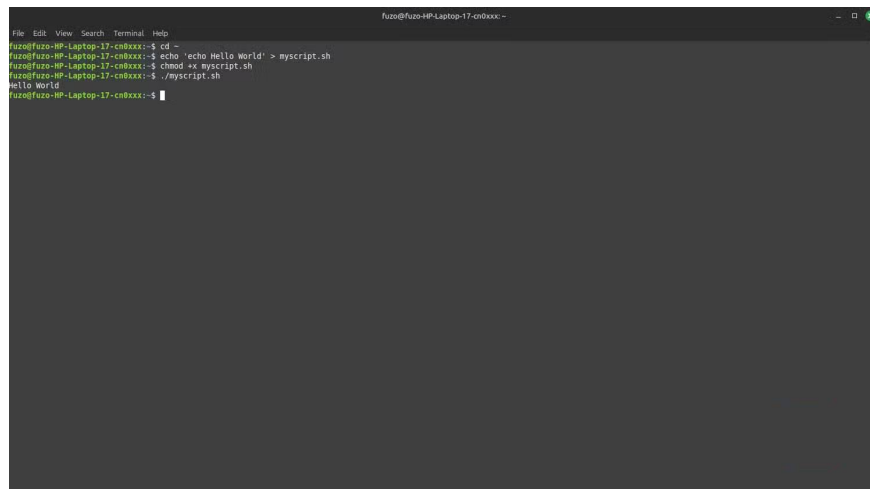
6 Linux permission commands every admin needs to know

You need to master some Linux terminal permission commands and below are the top terminal commands that you should memorize.

Once you understand how Linux handles files—who can read, write, or execute—you'll be a true Linux administrator. There are a number of Linux terminal permission commands you'll need to master, and here are the top terminal commands you should memorize.

chmod

Control who can read, write, or execute files

A terminal window screenshot showing a sequence of commands and their outputs. The user is in a directory named 'myscript.sh'. They run 'echo 'echo Hello World' > myscript.sh' to create a file. Then they run 'chmod +x myscript.sh' to make it executable. Finally, they run './myscript.sh' which outputs 'Hello World'.

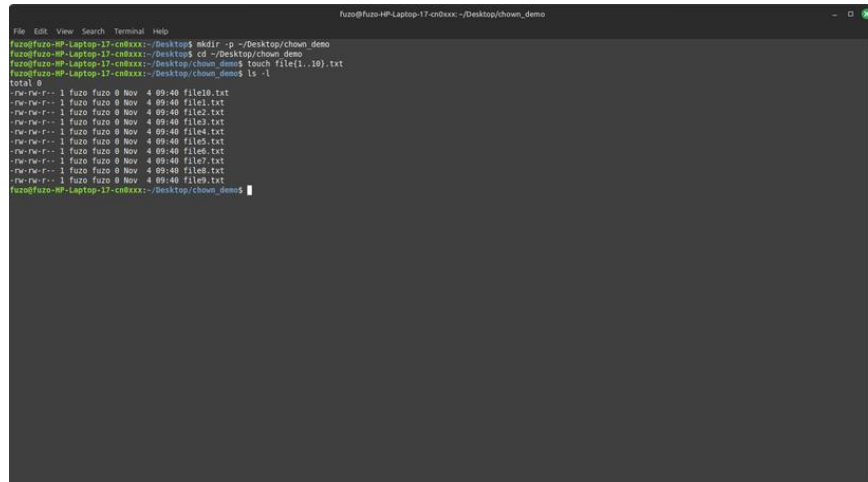
```
fuzo@fuzo-HP-Laptop-17-cn8xxx:~$ cd -
fuzo@fuzo-HP-Laptop-17-cn8xxx:~/myscript.sh$ echo 'echo Hello World' > myscript.sh
fuzo@fuzo-HP-Laptop-17-cn8xxx:~/myscript.sh$ chmod +x myscript.sh
fuzo@fuzo-HP-Laptop-17-cn8xxx:~/myscript.sh$ ./myscript.sh
Hello World
fuzo@fuzo-HP-Laptop-17-cn8xxx:~/myscript.sh$
```

Once you know why a script isn't running, start using the "chmod" command more often. All Linux files have read (r), write (w), and execute (x) permissions. If you run **chmod +x** followed by the script name, the file will be executed as a program.

chmod also allows you to change permissions numerically. The command **chmod 755 filename** gives the owner full access (rwx) and everyone else read and execute (rx). After you encounter your first "Permission denied" message, running this command can become a habit.

chown

Take ownership of files as administrator



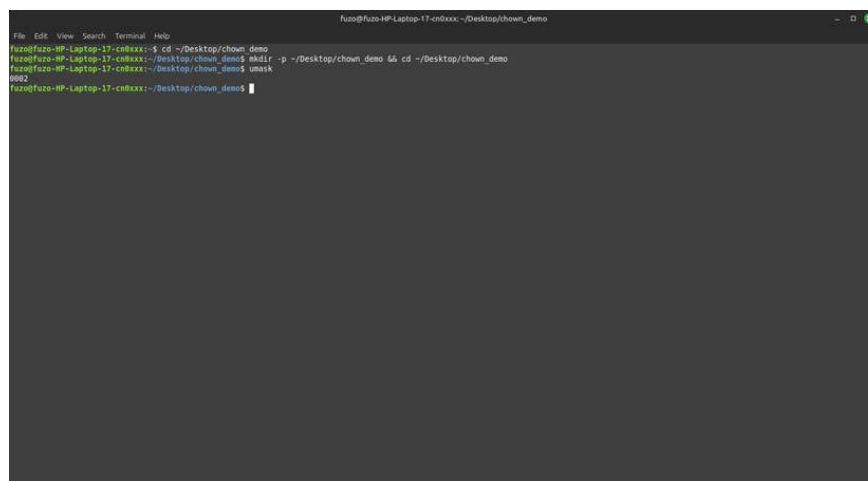
```
fuzo@fuzo-HP-Laptop-17-cdxxxx:~/Desktop/chown_demo$ sudo chown demo
fuzo@fuzo-HP-Laptop-17-cdxxxx:~/Desktop/chown_demo$ cd ~/Desktop/chown_demo
fuzo@fuzo-HP-Laptop-17-cdxxxx:~/Desktop/chown_demo$ touch file{1..10}.txt
fuzo@fuzo-HP-Laptop-17-cdxxxx:~/Desktop/chown_demo$ ls -l
total 0
-rw-rw-r-- 1 fuzo fuzo 0 Nov  4 09:40 file10.txt
-rw-rw-r-- 1 fuzo fuzo 0 Nov  4 09:40 file1.txt
-rw-rw-r-- 1 fuzo fuzo 0 Nov  4 09:40 file2.txt
-rw-rw-r-- 1 fuzo fuzo 0 Nov  4 09:40 file3.txt
-rw-rw-r-- 1 fuzo fuzo 0 Nov  4 09:40 file4.txt
-rw-rw-r-- 1 fuzo fuzo 0 Nov  4 09:40 file5.txt
-rw-rw-r-- 1 fuzo fuzo 0 Nov  4 09:40 file6.txt
-rw-rw-r-- 1 fuzo fuzo 0 Nov  4 09:40 file7.txt
-rw-rw-r-- 1 fuzo fuzo 0 Nov  4 09:40 file8.txt
-rw-rw-r-- 1 fuzo fuzo 0 Nov  4 09:40 file9.txt
fuzo@fuzo-HP-Laptop-17-cdxxxx:~/Desktop/chown_demo$
```

"chown" is a lifesaver when you're copying files between drives and can't access them afterwards. It directly affects permissions by changing the ownership of a file or directory. The basic format is: **sudo chown user:group filename** .

You will need this command after moving important system data or restoring a backup. For example, if you need the web server (Apache) to read a directory, use the command **sudo chown -R www-data:www-data /path/to/project** , where **www-data** is the username and group specific to the Apache web server.

umask

Set default permissions for new files



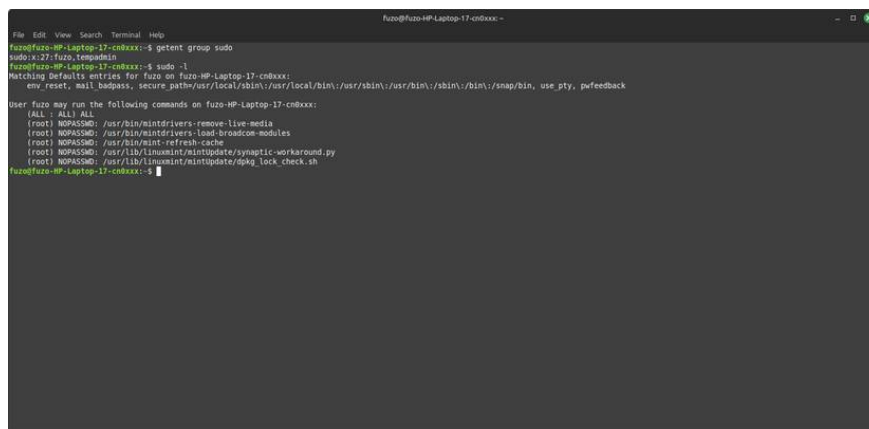
```
fuzo@fuzo-HP-Laptop-17-cdxxxx:~/Desktop/chown_demo$ sudo umask 002
fuzo@fuzo-HP-Laptop-17-cdxxxx:~/Desktop/chown_demo$
```

On Linux, umask determines how new files and directories inherit permissions. Linux uses your current umask value to determine which permissions to mask (remove) from the base permissions. The umask command allows you to view or change this setting for the current session.

The common default value of umask is **0022**, and it removes write permissions from "group" and "other users". You can type **umask** to see your current value, and a command like **umask 0027** will change it for the current session. This setting makes new files more secure, often blocking access from other users or anyone outside your group.

sudoers

Grant or restrict security admin rights



```
fuze@fuze-HP-Laptop-17-cn8xxx:~$ getent group sudo
sudo:27:fuze,tepmadm
fuze@fuze-HP-Laptop-17-cn8xxx:~$ sudo -l
Matching Defaults entries for fuze on fuze-HP-Laptop-17-cn8xxx:
env_reset, mail_badmail, secure_path=/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin, use_pty, pwfeedback

User fuze may run the following commands on fuze-HP-Laptop-17-cn8xxx:
  (ALL : ALL) ALL
  (root) NOPASSWD: /usr/bin/mintdrivers-remove-live-media
  (root) NOPASSWD: /usr/bin/mintdrivers-load-broadcom-modules
  (root) NOPASSWD: /usr/bin/mint-refresh-cache
  (root) NOPASSWD: /usr/lib/linuxmint/mintupdate/synaptic-workaround.py
  (root) NOPASSWD: /usr/lib/linuxmint/mintupdate/dpkg_lock_check.sh
fuze@fuze-HP-Laptop-17-cn8xxx:~$
```

Many people might think that you only become a real Linux administrator after successfully editing the sudoers file. The **/etc/sudoers** file determines who has sudo privileges and what commands they can run. As a rule, always edit this file with **sudo visudo** to avoid syntax errors that can get you locked out.

On Linux Mint, if you need to give admin privileges to another user, use the command **sudo usermod -aG sudo username**. This is much cleaner and safer than editing the file manually, although you may need specific command permissions for advanced settings.

groups

See which access group you belong to

```
File Edit View Search Terminal Help
fuzo@fuzo-HP-Laptop-17-cn0xxx:~$ groups
fuzo adm cdrom sudo dip plugdev users lpadmin sambashare
fuzo@fuzo-HP-Laptop-17-cn0xxx:~$ groups fuzo
fuzo : fuzo adm cdrom sudo dip plugdev users lpadmin sambashare
fuzo@fuzo-HP-Laptop-17-cn0xxx:~$ for user in $(cut -f1 -d: /etc/passwd); do echo "$user: $(groups $user)"; done
root: root : root
daemon: daemon : daemon
bin: bin : bin
sys: sys : sys
sync: sync : nogroup
news: news : news
man: man : man
lp: lp : lp
mail: mail : mail
news: news : news
nscp: nscp : nscp
proxy: proxy : proxy
www-data: www-data : www-data
backup: backup : backup
list: list : list
irc: irc : irc
apt: apt : nogroup
nobody: nobody : nogroup
system-network: system-network : system-network
system-timesync: system-timesync : system-timesync
dshppd: dshppd : nogroup
messagebus: messagebus : messagebus
systemd: systemd : systemd
systemd-resolve: systemd-resolve : systemd-resolve
polkitd: polkitd : polkitd
cbsm: cbsm : plugdev
tss: tss : tss
rtkit: rtkit : rtkit
systemd-coredump: systemd-coredump : systemd-coredump
kernoops: kernoops : nogroup
nvidia: nvidia : nvidia
cups-pk-helper: cups-pk-helper : lpadmin
avahi: avahi : avahi
avahi-autoipd: avahi-autoipd : avahi-autoipd
flatpak: flatpak : flatpak
avahi: avahi : avahi
mocoles: mocoles : mocoles
```

Groups on Linux determine what you can access and tell you which group you belong to in the hierarchy. For example, if you run the command **groups afam** , it might return **afam sudo adm lp** , showing you are an admin user with access to the printer.

Running **sudo usermod -aG project afam** can make a big difference if you're new to a project. It allows you to run scripts and edit files with teammates without having to sudo for every command.

id

Identify yourself with the system

```
File Edit View Search Terminal Help
fuzo@fuzo-HP-Laptop-17-cn0xxx:~$ id
uid=1000(fuzo) gid=1000(fuzo) groups=1000(fuzo),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),100(users),105(lpadmin),125(sambashare)
fuzo@fuzo-HP-Laptop-17-cn0xxx:~$ id -u
1000
fuzo@fuzo-HP-Laptop-17-cn0xxx:~$ id -g
1000
fuzo@fuzo-HP-Laptop-17-cn0xxx:~$ id -G
1000 4 24 27 30 46 100 105 125
fuzo@fuzo-HP-Laptop-17-cn0xxx:~$ id demo_user
id: 'demo_user': no such user
fuzo@fuzo-HP-Laptop-17-cn0xxx:~$
```

The id command tells you who you are to the system, which is necessary before you can fix permissions. Simply running the id command will show you your user ID (UID), group ID (GID), and a list of all the groups you belong to. While this command doesn't directly label your user type, the UID can infer which of the four main Linux user types you are.

Rely on this command when debugging access issues. So if a certain share fails to open, you can use the id to see if your user matches the owner or group, confirming whether the "permission denied" issue is due to identity or permissions.

You finished reading the article "**6 Linux permission commands every admin needs to know**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.

