

- راهنمای دستورات لینوکس
- عملیات فایل و دایرکتوری 1.
- مجوزهای فایل 2.
- مشاهده و ویرایش فایل‌ها 3.
- اطلاعات سیستم 4.
- شبکه و اتصال به سرور 5.
- مدیریت فرآیندها و تاریخچه دستورات 6.
 - متغیرها و محیط برنامه نویسی شل 7.
- 8. مدیریت بسته‌ها (Package Management)
 - (مانند Ubuntu) Debian برای توزیع‌های مبتنی بر
 - مثال:
 - موجود نباشد به خطا dependency چون از افلاین نصب میکند اگر dpkg : نکته میخورد
 - دانلود پکیج‌ها بدون نصب
 - پاک کردن پکیج‌ها
- 9. دستورات مربوط به ادمین:
 - useradd
 - groupadd
 - passwd
 - usermode
 - groupmod
 - chown
 - chmod
 - chgrp
 - ترکیب با دستورات دیگر
- 10. سایر دستورات مفید:

راهنمای دستورات لینوکس

این راهنما دستورات اساسی لینوکس را به همراه نحوه استفاده آنها به صورت دسته‌بندی شده برای مراجعه سریع ارائه می‌دهد. شامل:

- عملیات فایل و دایرکتوری
- مجوزهای فایل
- مشاهده و ویرایش فایل‌ها
- اطلاعات سیستم

- شبکه و اتصال به سرور
- مدیریت فرآیندها و تاریخچه دستورات
- سایر دستورات مفید

عملیات فایل و دایرکتوری

- نمایش محتویات دایرکتوری: **ls**

```
ls          # دیدن محتویات یک مسیر  
ls -l       # لیست تفصیلی  
ls *.txt    # دیدن همه فایلها با پسوند txt  
ls -ltch    # دیدن فایلها با مخصات بیشتر
```

- فایلها و دایرکتوریهای مخفی: برای نمایش فایلها و دایرکتوریهای مخفی (که با نقطه) استفاده کنید **-a** شروع می‌شوند) از گزینه

```
ls -a
```

- تغییر دایرکتوری: **cd**

```
cd /home/user # دایرکتوری مطلق  
cd new_folder # دایرکتوری نسبی
```

- نمایش دایرکتوری کاری فعلی: **pwd**

```
pwd
```

- command دیدن دستور العمل هر: **man**

```
man ls  
man cd
```

- command دیدن محل ذخیره شدن فایل اجرایی هر: **whereis**

```
whereis ls  
whereis cd
```

- **mkdir**: ایجاد یک دایرکتوری جدید.

```
mkdir new_folder
```

- **rmdir**: حذف یک دایرکتوری خالی.

```
rmdir old_folder
```

- **rm**: حذف فایل‌ها یا دایرکتوری‌ها.

```
rm file.txt
```

- **touch**: ایجاد یک فایل خالی.

```
touch newfile.txt
```

- **stat**: نمایش اطلاعات دقیق یک فایل یا دایرکتوری.

```
stat file.txt
```

2. مجوزهای فایل

- **chmod**: تغییر مجوزهای فایل.

```
chmod 755 script.sh
```

- **chown**: تغییر مالک و گروه فایل.

```
chown user:group file.txt
```

- **chgrp**: تغییر مالکیت گروه یک فایل.

```
chgrp groupname file.txt
```

3. مشاهده و ویرایش فایل‌ها

cat: نمایش محتویات فایل.

شروع می‌شود **file** دیدن همه فایل‌های که نامشان با

```
cat file*
```

- **pipe** جستجو در خروجی دستورات با استفاده از **grep** دستورات مرتبط با

```
ls -l | grep "pattern"
```

- شمارش خطوط، کلمات و کاراکترها در یک فایل (**wc**) دستگاه شمارش کلمات.

```
wc file.txt
```

4. اطلاعات سیستم

- **top**: نمایش فرآیندهای در حال اجرا به صورت زنده.
- **df**: نمایش استفاده از فضای دیسک.

```
df -h
```

- **free**: نمایش استفاده از حافظه.

```
free -m
```

- **uname -a**: نمایش اطلاعات سیستم.

```
uname -a
```

- **hostnamectl**: نمایش یا تنظیم نام میزبان سیستم.

```
hostnamectl
```

- **whoami**: نام کاربری جاری

```
whoami
```

- **who**: نمایش کاربران وارد شده

```
who
```

5. شبکه و اتصال به سرور

- **ssh username@hostname**: SSH اتصال به یک ماشین خارجی با پروتکل.

```
ssh user@hostname
```

- برای دانلود فایل‌ها از اینترنت: **wget** و **curl** دستورات

```
wget http://example.com/file.zip
```

```
curl -O http://example.com/file.zip
```

مدیریت فرآیندها و تاریخچه دستورات 6.

دستورات مدیریت فرآیندها (**bg**, **fg**, **jobs**):

کار پس‌زمینه:

bg

کار پیش‌زمینه:

fg

لیست کارها:

jobs

تاریخچه دستورات (**history**):

history

متغیرها و محیط برنامه نویسی شل 7.

- متغیرها (**variables**):

VAR=value

متغیرهای ویژه (**special variables**):

\$. کد خروج آخرین دستور اجرا شده: \$?

\$\$: شناسه فرآیند جاری.

- تنظیم متغیر محیطی (**export**):

```
export VAR=value
```

- حذف متغیر (**unset**):

```
unset VAR
```

- بارگذاری مجدد تنظیمات شل: **source** استفاده از

```
source ~/.bashrc
```

8. مدیریت بسته‌ها (Package Management)

Debian (مانند Ubuntu) برای توزیع‌های مبتنی بر

source.list: آپدیت لیست ریو‌های در سیستم اوکال

```
apt update
```

در خصوص آن پکیج (**info**) دید اطلاعات

```
apt info package_name
```

آپگرید پکیج‌های موجود

```
apt upgrade
```

پکیج‌های صب شده بروی سیستم

```
apt list --installed
```

- نصب بسته‌ها (**apt**, **apt-get**, **dpkg**)

```
sudo apt install package_name
```

```
sudo apt-get remove package_name
```

: داریم بصورت افلاین **deb**. برای نصب برنامه های که در انتهای آنها پسوند

```
# i = --install  
dpkg -i package.deb
```

مثال:

ابتدا پکیج را دالود می کنیم و سپس آنرا بصورت آفلاین نصب می کنیم

چون از افلاینن نصب میکند اگر dpkg : نکته موجود نباشد به خطا میخورد dependency

```
# Download postgresql  
# curl -o yourname.deb Address  
# download all from Address to -> pg17main.deb  
curl -O pg17main.deb  
https://apt.postgresql.org/pub/repos/apt/pool/main/p/postgresql-  
17/postgresql-17_17.2-1.pgdg%2B1_amd64.deb  
  
#install  
dpkg -i pg17main.deb  
  
# Show All dependency of  
dpkg --contents pg17main.deb
```

دیدن لیست ریو های انلای موجود برای یک پکیج وقتی میخواهیم یک پکیج را پیدا و سپس نسخه مناسب را نصب کنیم

```
apt search --names-only postgre |grep post
```

دانلود پکیج ها بدون نصب

دانلود پکیج و نیازمندی های آن بدون نصب (مثلا سب دانلود کنیم تا فردا نصب کنیم)

```
apt-get install--download-only <package_name>
```

دانلود فقط خود برنامه

```
apt-get download <package_name>
```

پاک کردن پکیج ها

پکیج مورد نظر پاک می شود اما نیازمندی های آن که نصب شده باقی می ماند

```
apt-get remove <package_name>
```

که بخاطر یک پکیج دیگر نصب شده اند و الان دیگر به آنها dependency پاک کردن پکیج های نیازی نیست:

```
apt-get autoremove <package_name>
```

پاک کردن همه نیازمندی های درون سیستم که بلا استفاده هستند

```
apt-get autoremove
```

دستورات مربوط به ادمین: 9.

useradd

با دسترسی ادمین اجرا می شوند sudo دستورات بعد از #

sudo useradd <username> # افزودن کاربر جدید

sudo useradd -m <username> # ساخت هوم دایرکتوری

```
sudo useradd <username> -e 10-10-24 # افزودن کاربر جدید همراه با تاریخ انقضا
sudo useradd <username> -g <groupname> # افزودن کاربر جدید همراه با اضافه کردن به
گروه مشخص
```

groupadd

```
sudo groupadd <groupname>
# ایجاد یک گروه جدید با نام مشخص
sudo groupadd developers

# مشخص GID ایجاد یک گروه جدید با
sudo groupadd -g <GID> <groupname>
sudo groupadd -g 1050 finance

# (کمتر از GID 1000 گروهی با) ایجاد گروه سیستمی
sudo groupadd -r <groupname>
sudo groupadd -r system_group

# groupadd نمایش نسخه دستور
groupadd --version

# /etc/login.defs با توجه به فایل) ایجاد گروه با تنظیمات پیش فرض
sudo groupadd <groupname>

# برای حذف یک گروه
sudo groupdel <groupname>
```

فایل‌های مرتبط:

/etc/group: فایل حاوی لیست گروه‌ها در سیستم.

/etc/login.defs: فایل پیکربندی پیش فرض برای ایجاد کاربران و گروه‌ها.

passwd

برای مدیریت رمز عبور کاربران در سیستم‌های لینوکسی استفاده می‌شود **passwd** دستور

```
passwd
# تغییر رمز عبور کاربر فعلی

# (sudo نیاز به دسترسی ریشه یا) تغییر رمز عبور یک کاربر دیگر
sudo passwd <username>
```

قفل کردن حساب کاربری (غیرفعال کردن رمز عبور کاربر) #

```
sudo passwd -l <username>
```

```
sudo passwd -l john
```

باز کردن قفل حساب کاربری #

```
sudo passwd -u <username>
```

حذف رمز عبور کاربر (امکان ورود بدون رمز عبور) #

```
sudo passwd -d <username>
```

```
sudo passwd -d jane
```

تنظیم تاریخ انقضای رمز عبور کاربر #

```
sudo passwd -e <username>
```

تغییر تعداد روزهای انقضا و اطلاع از انقضای رمز عبور #

```
sudo passwd -n <minimum_days> -x <maximum_days> -w <warning_days> <username>
```

پس از 90 روز منقضی می‌شود و 10 روز قبل از انقضا هشدار داده "john" رمز عبور کاربر (می‌شود).

```
sudo passwd -x 90 -w 10 john
```

مشاهده اطلاعات مربوط به رمز عبور کاربر (فقط توسط کاربر ریشه) #

```
sudo passwd -S <username>
```

- l (Lock) قفل کردن حساب.
- u (Unlock) باز کردن قفل حساب.
- d حذف رمز عبور.
- e مجبور کردن کاربر به تغییر رمز عبور در ورود بعدی.
- n تنظیم تعداد روزهای حداقل برای تغییر رمز عبور.
- x تنظیم تعداد روزهای حداکثر برای انقضای رمز عبور.
- w تعداد روزهای اطلاع‌رسانی قبل از انقضای رمز عبور.
- S نمایش وضعیت رمز عبور کاربر.

usermode

باعث می‌شود که کاربر به یک گروه اضافه شود بدون اینکه از گروه‌های دیگر حذف -aG دستور (append Group) شود.

```
sudo usermod -aG <groupname> <username>
```

افزودن یک کاربر به گروه مشخص

```
# sudo برای استفاده از داکر بدون نیاز به گروه
sudo usermod -aG docker <username>

# تغییر نام کاربری
sudo usermod -l <new_username> <old_username>

# قفل کردن حساب کاربری
sudo usermod -L <username>

# باز کردن قفل حساب کاربری
sudo usermod -U <username>

# کاربر (home directory) تغییر پوشه خانگی
sudo usermod -d /new/home/directory <username>

# تغییر پوشه خانگی و انتقال فایل‌های موجود به پوشه جدید
sudo usermod -d /new/home/directory -m <username>

# کاربر UID تغییر
sudo usermod -u <new_uid> <username>

# گروه اصلی کاربر GID تغییر
sudo usermod -g <new_gid> <username>
```

groupmod

برای تغییر تنظیمات گروه‌های موجود در سیستم‌های لینوکسی استفاده **groupmod** دستور می‌شود.

```
sudo groupmod -n <new_groupname> <old_groupname>
# تغییر نام یک گروه
sudo groupmod -n engineers developers

# یک گروه GID تغییر
sudo groupmod -g <new_GID> <groupname>
sudo groupmod -g 2000 finance

# گروه به‌طور همزمان GID تغییر نام و
sudo groupmod -n <new_groupname> -g <new_GID> <old_groupname>
sudo groupmod -n admins -g 1500 staff
```

chown

فایل‌ها و دایرکتوری‌ها در لینوکس (group) و گروه (owner) برای تغییر مالکیت **chown** دستور استفاده می‌شود.

```
sudo chown <new_owner> <file_or_directory>
# تغییر مالک فایل یا دایرکتوری
sudo chown alice file.txt

# تغییر گروه فایل یا دایرکتوری
sudo chown :<new_group> <file_or_directory>
sudo chown :developers /var/www/

# تغییر همزمان مالک و گروه فایل یا دایرکتوری
sudo chown <new_owner>:<new_group> <file_or_directory>
sudo chown alice:developers project.zip

# تغییر مالک و گروه همه فایل‌ها و دایرکتوری‌های داخل یک مسیر به صورت بازگشتی
sudo chown -R <new_owner>:<new_group> <directory>
sudo chown -R bob:staff /home/bob/

# تغییر فقط مالکیت فایل‌ها (بدون تغییر گروه)
sudo chown --changes <new_owner> <file_or_directory>

# تغییر مالک و گروه یک مسیر خاص و نمایش جزئیات تغییرات
sudo chown --verbose <new_owner>:<new_group> <file_or_directory>
```

chmod

فایل‌ها و دایرکتوری‌ها در (Permissions) برای تغییر مجوزهای دسترسی **chmod** دستور سیستم‌های لینوکسی استفاده می‌شود.

```
ls -l # دیدن مجوز دسترسی ها

chmod [options] <permissions> <file_or_directory>
# تغییر مجوزها به صورت عددی (مثلاً 755)
chmod 755 <file_or_directory>

# تغییر مجوزها به صورت نمادین (مثلاً افزودن مجوز اجرا برای مالک)
chmod u+x <file_or_directory>

# حذف مجوز نوشتن از گروه
chmod g-w <file_or_directory>
```

```
# تنظیم مجوزها برای همه کاربران (مالک، گروه و دیگران)
chmod a+r <file_or_directory>
```

نمادهای مربوط به مجوزها:

سه بخش اصلی مجوزها:

u: مالک (user)

g: گروه (group)

o: دیگران (others)

a: همه (all)

نوع دسترسی:

r:4: خواندن (read)

w:2: نوشتن (write)

x:1: اجرا (execute)

عملیات:

+ افزودن مجوز:

- حذف مجوز:

= تنظیم مجوز به طور دقیق:

مثال:

```
chmod u+x script.sh
```

به فایل مجوز اجرا برای مالک اضافه می‌شود.

```
chmod g-w file.txt
```

مجوز نوشتن از گروه برای فایل حذف می‌شود.

```
chmod 744 file.txt
```

این دستور به مالک مجوز خواندن، نوشتن، و اجرا، و به دیگران فقط مجوز خواندن می‌دهد.

```
chmod u=rwx,g=rx,o=r file.txt
```

این دستور مجوزها را به طور دقیق برای فایل به این شکل تنظیم می‌کند:

(rwx) مالک: خواندن، نوشتن، و اجرا

(rx) گروه: خواندن و اجرا

(r) دیگران: خواندن

فایل‌ها و دایرکتوری‌ها (Group Ownership) در لینوکس برای تغییر گروه مالک **chgrp** دستور استفاده می‌شود.

```
sudo chgrp <groupname> <file_or_directory>
# تغییر گروه یک فایل یا دایرکتوری
sudo chgrp developers file.txt # file
sudo chgrp staff /var/www      # dir

# تغییر گروه فایل‌ها و دایرکتوری‌ها به صورت بازگشتی
sudo chgrp -R <groupname> <directory>
sudo chgrp -R admins /home/project

# نمایش جزئیات تغییر گروه‌ها
sudo chgrp --verbose <groupname> <file_or_directory>
sudo chgrp --verbose users report.txt
```

ترکیب با دستورات دیگر:

تغییر مالک و گروه هم‌زمان: **chown** ترکیب با

```
sudo chgrp developers file.txt
sudo chmod 775 file.txt

sudo chown user:developers file.txt
```

سایر دستورات مفید.10:

برای تأخیر در اجرای دستورات: **sleep** دستور

```
sleep 5 # تأخیر به مدت پنج ثانیه
```

برای مشاهده راهنمای دستورات: **man** دستور

```
man ls # مشاهده راهنمای دستور ls
```

حساسیت کوچکی و بزرگی عبارات را نادیده می‌گیرد **-i** - فلگ

```
# find sajad SAJAD Sajad SaJaD  
find -i sajad
```