حسین سیلانی
طراح و توسعه دهنده و SysAdmin

لینوکس را فارسی یادبگیرید
www.learninghive.ir
www.seilany.ir

40 روز 40 دستور لینوکسی

40 DAYS 40 Commands

1st Edition

معرفی کوتاه کتاب:

این کتاب یک مرور سریع از دستورات مهم، رایج در محیط لینوکس است. این کتاب 40 دستور را شامل می‌شود و به ساختار، شرح، گزینه‌ها و مثال‌های دستورات می‌پردازد. همچنین، مثال‌هایی با توضیحات برای رفع ابهامات نیز ارائه شده است.دستورات قابل استفاده برای اکثر توزیع های لینوکسی است.

منبع فارسی دستورات لینوکس

شرح دستورات
ساختار دستورات
گزینه های دستورات
مثال های کامل از دستورات

۴۰ روز ۴۰ دستور لینوکس

40 days, 40 Linux Commands

حسین سیلانی

ویرایش اول

۱۴۰۵

انتشارات یافته

https://yaftehpub.ir

سخنی از نویسنده

با سلام و احترام به دوست گرامی،

از اینکه این کتاب را مورد توجه خود قرار داده‌اید و در این مسیر همراه من بوده‌اید، صمیمانه سپاسگزارم. امیدوارم این اثر برای علاقه‌مندان به این حوزه مفید و ثمربخش واقع شود. همچنین از شما تقاضا دارم که نواقص و لغزش‌های احتمالی کتاب را نادیده نگیرید و با ارائه نظرات و پیشنهادات ارزشمند خود، در ارتقای کیفیت آن یاری‌ام رسانید. لازم به ذکر است که این کتاب، یکی از صد کتاب لینوکسی من(بسیاری ازآنها درمرحله ترجمه یا نگارش هستند) است که در راستای پروژه‌های متن باز شخصی‌ام و با هدف طراحی و توسعه توزیع‌های لینوکس به نگارش درآمده است.

https://predator-os.ir/
https://predator-os.ir/#Little-Psycho
https://emperor-os.ir/

تصمیم دارم تولید محتوای فارسی در زمینه‌های لینوکس و متن باز را در قالب‌های مختلفی مانند کتابچه، کتاب، فلش کارت، صوت و فیلم‌های تصویری در قالب یک سایت ارائه کنم. هدف من از این کار، ایجاد یکی از بزرگترین مرجع‌های آموزش فارسی در این حوزه است.

https://learninghive.ir/

جهت پیگیری آدرس سایت و اطلاع از آخرین اخبار و محتواهای آموزشی، می‌توانید از طریق راه‌های زیر با من در ارتباط باشید:

- Telegram channel: @Linuxtnt
- Telegram ID: @seilany
- Gmail : h.seilany@gmail.com
- https://seilany.ir

تقدیم به تو

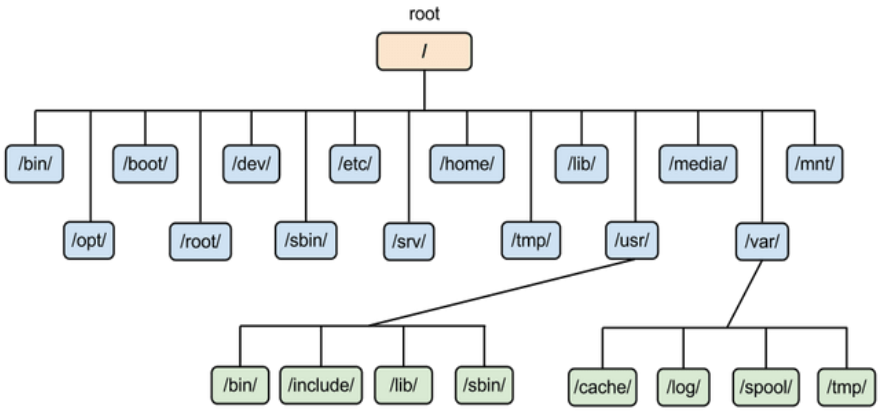
یگانه فرزانه دلم

فهرست مطالب

روز اول

سازمان سیستم فایل

فایل‌ها در یک سیستم لینوکس در ساختار فهرست سلسله مراتبی[۱] قرار می گیرند. این بدان معنی است که آنها در یک الگوی درختی[۲] فهرست (فهرست نامیده شده در سایر سیستم ها) سازماندهی شده اند، که ممکن است حاوی فایل ها و سایر فهرست ها باشد. اولین فهرست در سیستم فایل فهرست ریشه[۳] نامیده می‌شود. فهرست ریشه حاوی فایل ها و زیر شاخه ها[۴] هستند. اکثر محیط های گرافیکی[۵] امروز شامل یک برنامه مدیریت فایل برای مشاهده و دستکاری محتویات سیستم فایل می باشد. اغلب سیستم فایل را به صورت زیرمی بینید:



File System Hierarchy

یکی از تفاوت های مهم بین سایرسیستم عامل و سیستم عامل های مشابه یونیکس مانند لینوکس این است که لینوکس مفهوم حروف درایو[۶] را استفاده نمی‌کند. در حالی که حروف[۷] درایو سیستم فایل را به یک سری درخت[۸] های مختلف تقسیم می‌کند (یکی برای هر درایو)، لینوکس همیشه یک درخت تنها دارد. دستگاه های مختلف ذخیره سازی ممکن است شاخه های مختلف درخت داشته باشند، اما همیشه یک درخت تنها وجود دارد. در سیستم عامل لینوکس مفهوم ساختار درختی[۹] بیان می شود. زیرا لینوکس برپایه یونیکس بوده و مفهوم درخت از آنجا گرفته شده است و دیگر خبری از درایو نیست.ساختار فایل به صورت درختی وارونه است که ریشه این درخت به نام root است.این ریشه در بالاترین بخش درخت قرار دارد و بقیه بخش ها زیر مجموعه آن می باشد . هرچیزی که فایل نباشد یک پردازش[۱۰] به شمار می آید. توجه داشته باشید که کرنل لینوکس در تمامی پارتیشن ها توزیع شده و در یک بخش وجود ندارد.

---

۱ Hierarchical
۲ tree
۳ root
۴ sudirectory
۵ GUI(graphic user interface)
۶ drive
۷ drive letter
۸ tree structure
9 Trees (Data Structures)
۱۰ process

ساختار فهرست ها

فهرست‌ها در لینوکس به صورت سلسله‌مراتبی و ساختاردار سازماندهی می‌شوند. در لینوکس، فهرست‌ها با استفاده از دو نوع عنصر اصلی ساخته می‌شوند:

1 . **فایل‌ها :(Files)** در لینوکس، هر چیزی از پوشه‌ها گرفته تا دستگاه‌های ورودی و خروجی به عنوان فایل شناخته می‌شود. فایل‌ها می‌توانند اطلاعات متنی، تصویری، صوتی، برنامه‌ها و غیره باشند.

2 . **پوشه‌ها :(Directories)** پوشه‌ها به عنوان محلی برای سازماندهی و ذخیره فایل‌ها در سیستم عامل لینوکس عمل می‌کنند. هر پوشه می‌تواند شامل فایل‌ها و یا زیرپوشه‌های دیگری باشد.

ساختار فهرست‌ها در لینوکس به صورت درختی و سلسله‌مراتبی است، به این صورت که یک پوشه می‌تواند شامل فایل‌ها و/یا زیرپوشه‌های دیگری باشد. این امر امکان می‌دهد که اطلاعات بر روی سیستم به شکل سازماندهی شده‌ای ذخیره شوند و دسترسی به آن‌ها سهل‌تر باشد. هر فهرست در لینوکس یک مسیر **(Path)** دارد که نشان‌دهنده موقعیت آن در ساختار فهرست سلسله‌مراتبی است. به عنوان مثال، `/home/user/Documents` نشانگر فهرست Documents درون پوشه user درون پوشه home است.

به طور خلاصه، ساختار فهرست در لینوکس با استفاده از فایل‌ها و پوشه‌ها به صورت درختی و سلسله‌مراتبی سازماندهی می‌شود که این ساختار امکان مدیریت و دسترسی آسان به اطلاعات را فراهم می‌کند.

/ bin :

همه برنامه های باینری[۱] اجرایی مورد نیاز در هنگام بوت شدن، تعمیر و فایل های مورد نیاز برای اجرا به حالت تک کاربره[۲] و سایر دستورات مهم، یعنی cat، du، df، tar، rpm، wc،history، و غیره در این فهرست است.

/ boot :

فایلهای مهمی در طول فرایند بوت شدن، از جمله هسته linux نگهداری می کند.

/ dev :

شامل فایل های دستگاه[۳] برای تمام دستگاه های سخت افزاری دستگاه مانند cdrom، cpuو غیره است.

/ etc :

شامل فایل های پیکربندی برنامه[۴]، راه اندازی ، خاموش کردن ، شروع ، توقف اسکریپت

/ home

فهرست خانگی کاربران[۵]. هر بار که یک کاربر جدید ایجاد می شود، یک فهرست به نام کاربر در فهرست home ایجاد می شود که حاوی فهرست های دیگر مانند desktop ، download و غیره است.

/ lib :

فهرست Lib[۶] شامل ماژول های[۷] هسته[۸] مورد نیاز برای راه اندازی system و اجرای دستورات در سیستم فایل root است.

/lost + found : این فهرست در هنگام نصب سیستم عامل ایجاد شده است، برای بازیابی فایلهایی که ممکن است به علت خاموش شدن غیر منتظره سیستم خراب شده اند، مفید است.

/ media :

---

۱ binary-دودویی
۲ single user
۳ device
۴ configures
۵ home dir
۶ library
۷ modules
۸ kernel

فهرست موقت برای دستگاه های قابل حمل[۱] مانند media / cdrom ایجاد می شود.

/ mnt :

فهرست موقت برای مونت شدن[۲] دستگاه ها به سیستم فایل است.

/ opt :

اختیاری به صورت انتخاب مختصر استشامل نرم افزار نرم افزار شخص ثالث.VIZ، جاواو غیره

/ proc/system

مجازی و شبه فایل که حاوی اطلاعاتی در مورد فرآیند در حال اجرا با خاص Process-id aka pid است.

/ root :

این فهرست home کاربر root است و هرگز نباید با ' /'اشتباه گرفته شود.

/ run :

این فهرست تنها راه حل مناسبی برای مشکل زودهنگام درایو است.

/ sbin :

حاوی برنامه های اجرایی باینری، مورد نیاز توسط مدیر system ، برای تعمیر و نگهداری. viz.

/ srv :

سرویس[۳] به صورت " srv " مخفف است.این فهرست حاوی فایل های مربوط به سرویس دهنده و سرویس خاص است.

/ sys :

توزیع های[۴] مدرن linux شامل یک فهرست /sys به عنوان یک سیستم فایل مجازی[۵] است که اجازه می دهد تا دستگاه های متصل به system را اصلاح کند.

/ tmp :

فهرست موقت[۶] system، قابل دسترسی توسط کاربر ریشه است و فایل های موقت را تا بوت بعدی برای کاربر و system ذخیره می کند.

/ usr :

حاوی مستندات[۷] ، کد منبع[۸] است. معمولاً حاوی بیشترین سهم داده در یک سیستم است. از این رو، این یکی از مهمترین فهرست ها در سیستم است زیرا شامل تمام باینری های کاربر ، اسناد آنها، کتابخانه ها ، پرونده های هدر، و غیره است.

/ var :

مخفف متغیر[۹] است .محتویات این فهرست همواره افزایش پیدا می کند.این فهرست حاوی پرونده های ورودی ، قفل ها، temp است.

---

۱ portable devices
۲ mount
۳ service
۴ distro
۵ virtual file system
۶ temporary
۷ documents
۸ source code
۹ variable

بررسی فایل مهم سیستم فایل

linux یک system پیچیده است که نیاز به یک راه پیچیده تر و کارآمد برای شروع ، متوقف کردن ، نگهداری بر خلاف ویندوز دارد.فایل های پیکربندی[۱] به خوبی برای این امر تعریف شده اند، فایل های باینری ، صفحات man ، فایل های اطلاعات و غیره برای هر پردازش در linux وجود دارد.

/ boot / vmlinuz : فایل کرنل linux است.

/dev / hda : فایل دستگاه برای اولین IDE HDD هارد دیسک است.

/dev / hdc : فایل معمولی برای IDE Cdrom است.

/dev / null : در برخی موارد خروج زباله به /dev / null هدایت می شود، به طوری که برای همیشه از بین می رود.

/ etc / bashrc : حاوی پیش فرض های system و نامهای مستعار استفاده شده توسط shell bash

/etc / crontab : یک اسکریپت پوسته برای اجرای دستورات مشخص در یک زمان از پیش تعریف شده است.

/etc / exports : اطلاعات سیستم فایل موجود در شبکه.

/etc / fstab : اطلاعات دیسک درایو و نقطه اتصال[۲] آنها .

/ etc / group : اطلاعات گروه امنیتی.

/ etc/grub.conf : فایل پیکربندی bootloader گراب[۳].

/etc/init.d : سرویس شروع اسکریپت.

/etc/lilo.conf : پرونده پیکربندی lilo bootloader

/ etc / hosts : اطلاعات آدرس ip و نام hostمربوطه.

/etc/hosts.allow : لیستی از hostها مجاز به دسترسی به خدمات در دستگاه[۴] محلی هستند.

/etc/host.deny : لیستی از میزبآنها یی که دسترسی به خدمات در دستگاه محلی را رد میکنند.

/etc / inittab : روند init و تعامل آن در سطوح اجرایی سیستم

/etc / issue : اجازه می دهد تا پیام پیش از ورود را ویرایش کنید.

/ etc/modules.conf : فایل های پیکربندی برای ماژول های system

/motd : motd / etc مخفف پیام روز است ، کاربران پیام را پس از ورود به ترمینالهای tty دریافت می کنند.

/etc / mtab : در حال حاضر بلوک اطلاعات را نصب کرده است.

/etc / passwd : حاوی رمز عبور از کاربران system در یک فایل سایه، یک پیاده سازی امنیتی است.

/ etc / printcap : اطلاعات چاپگر

/etc / profile : مقادیر پیش فرض پروفایل ها

/etc/profile.d : اسکریپت کاربردی، اجرا شده پس از ورود.

/ etc/rc.d : اطلاعات در مورد اسکریپت مشخصی در زمان اجرا.

/etc/rc.d/init.d : اسکریپت اولیه راه اندازی سطوح اجرا.

/etc/resolv.conf : سرورهای نام دامنه (DNS) که توسط system استفاده می شود.

/etc / securetty : فهرست ترمینال، که در ورود به آن امکان پذیر است.

/etc / skel : محتویات دایرکتوری کاربر جدید را ایجاد و تنظیم می کند.

/etc / termcap : یک فایل ASCIIکه رفتار ترمینال ، کنسول و چاپگر را تعریف می کند.

---

[۱] configure file

[۲] mount point

[۳] grub

[۴] local host

/etc / X11 : فایل های پیکربندیX-window

/ usr / bin : دستورات اجرایی کاربران

/ usr / include : شامل شامل فایل هایی است که توسط برنامه Cاستفاده می شود.

/ usr / share : فهرست فایل های اشتراک گذاشته شده از فایل های man ، فایل های اطلاعات

/usr / lib : فایلهای کتابخانه برنامه ها

/usr / sbin : دستورات کاربر برای مدیریت system

/proc / cpuinfo : اطلاعات CPU

/ proc / filesystems : اطلاعات سیستم فایل جاری رو نگه میدارد.

/ proc / interrupts : درمورد وقفه های فعلی که در حال حاضر مورد استفاده قرار می گیرند.

/proc / ioports : تمام آدرس های ورودی/ خروجی مورد استفاده توسط دستگاه ها در سرور است.

/proc / meminfo : اطلاعات استفاده از memory

/proc / modules : ماژول های مورد استفاده هسته.

/proc / mount : اطلاعات دستگاه های مونت شده

/proc / stat : آمار دقیقی از وضعیت کنونی سیستم

/proc / swaps : اطلاعات حافظه موقت swap

/var / log / lastlog : اطلاعات از آخرین روند بوت شدن سیستم.

/ var / log / messages : اطلاعات پیام های تولید شده daemon syslogدر هنگام بوت شدن.

/var / log / wtmp : لیست زمان ورود و مدت زمان هر کاربر در system نگهداری میکند.

فهرست ریشه

فهرست ریشه دارای فهرست‌های زیرین زیادی می باشد که به طور کلی آنها را معرفی می کنیم :

/bin : user binaries

در این فهرست بیشتردستورهای مدیریتی کاربر سیستم که قابل دسترسی برای همه کاربران و ازجمله فایل‌های باینری قابل اجرا هستند، قرارمی گیرد.

/sbin :system binaries

این فهرست همانند فهرست قبلی شامل دستورهای مدیریتی کاربر سیستم و هم چنین دستورهای مدیریتی مدیر سیستم را شامل می شود.

/etc:configuration files

این فهرست مثل سیستم عصبی بدن عمل می کند.کلیه تنظیمات سیستم که در قالب فایل وجود دارد دراین فهرست نگهداری و ذخیره می شوند. فهرست‌های زیادی به همراه فایل‌های تنظیمات در این مکان وجود دارد که به طور چکیده به معرفی آنها می پردازیم:

فهرست‌های موجود در /etc

/etc/x11 : تمامی تنظیمات مربوط به پنجره موسوم به x window که تنظیمات رابط گرافیکی کاربران است در این فهرست ذخیره می شود.

/etc/apt/sources.list

محل ذخیره لیست منابع مربوط به سرورهای بروزرسانی سیستم عامل و مخازن می باشد.

/dev: device files

فهرستی برای نگهداری فایل‌های سخت افزاری و وسیله‌های متصل به سیستم است.

/proc: process information

این فهرست محل نگهداری اطلاعات مربوط به پردازش‌های در حال اجرا و مخفی‌سیستم است.هم چنین اطلاعات منابع سیستم نیزدراین فهرست موجود است.

/var : variable files

این فهرست معمولا جداگانه ایجاد می‌شود و محل نگهداری اطلاعاتی هست که مرتبا درحال تغییر و بروزرسانی هستند و به عبارتی متغییرهستند.

فهرست‌های موجود در این مسیر:

- /var/backups
- /var/cache
- /var/crash
- /var/games
- /var/log
- /var/mail
- /var/spool

/tmp: temporary files

این فهرست برای نگهداری فایل‌های موقت[1] سیستم و کاربران است و معمولا فایل‌های موجود در آن،پس از راه اندازی مجدد[2] سیستم پاک می شوند.

/usr: user programs

دراین فهرست اطلاعاتی مربوط به کتابخانه ها،سندها، راهنماها،سورس کدها، برنامه‌های نصب شده و غیره می باشند.

/home : home directories

این فهرست محل نگهداری تمامی اطلاعات شخصی[3] کاربرانی است که به سیستم عاملloginکرده اند.مثلا

/home/ali

/boot: boot loader files

محل نگهداری فایل‌های بوت سیستم است.فایل‌هایی که سیستم نیازمند آن‌ها برای راه اندازی سیستم است و همچنین فایل‌هایی که در زمان بوت به آنها نیازدارد

/lib : system libraries

محل نگهداری فایل‌های کتابخانه برنامه‌ها است.این فایل ها مثل ماژول‌های کرنل، کتابخانه‌های اشتراکی(مثل کتابخانه‌های زبان c) که نیاز است به همراه سیستم بارگذاری شوند در این محل قرارمی‌گیرند.

/opt: optional add-on applications

محل نگهداری برنامه‌هایadd-on که برنامه‌های کمکی برای دیگر نرم افزارها هستند و همیشه به صورت پیش فرض نصب نیستند.مثل staroffice یا kget.

---

[1]Temporary
[2] Reboot
[3]Privacy

/mnt: mount directory

محل نگهداری نقطه‌های متصل شده[۱] به طورموقت به سیستم مثل فلش دیسک یایک هاردخارجی[۲] است.

/media : removable media devices

این فهرست برای محل نگهداری وسایل جانبی که قابلیت ذخیره سازی[۳]را دارند، مثل فلش دیسک ها، دیسک گردان ها و غیره می باشد.

/srv: service data

محل نگهداری سرویس‌های که مربوط به سرورها می باشد و معمولا در پس زمینه[۴] در حال اجرا هستند.

سایر سیستم فایل ها

lost+found/

برای هرپارتیشن در سیستم یک فهرست برای آن درفهرستlost+found ساخته می شود. در صورتی که اطلاعات اصلی دراین مسیر موجود باشد عمل بازیابی صورت گرفته و سیستم اجرا می شود و در غیر این صورت کاربر باید برنامه ویا سیستم خود را دوباره نصب و تنظیم نمایید.

/sys

در توزیع‌های جدید لینوکس از این فهرست برای نگهداری سیستم فایل مجازی استفاد می شود.

/boot/vmlinuz

محلی برای نگهداری فایل‌های‌کرنل لینوکس

/etc/fstab

محل نگهداری هارد دیسک‌هایی که به سیستم متصل شده اند، قرار دارد.

/etc/default/grub.conf

محل نگداری فایل تنظیمات راه انداز بوت سیستم :grub

/etc/lilo.conf

محل نگداری فایل تنظیمات راه انداز بوت سیستم :lilo

/etc/init.d

محل نگداری اسکریپت‌هایstart/stop سرویس ها

/etc/hosts

محل نگهداری آدرس‌های هایipهاست‌های مجازی و دیگرآدرس‌هایip

/etc/passwd

محل نگهداری پسوردهای کاربران سیستم

/etc/printcap

محل ذخیره اطلاعت مربوط به پرینترهای موجود برروی سیستم عامل

/etc/x11

محل ذخیره تنظیمات پنجره‌هایx . تنظیمات رابط‌های گرافیکی سیستم

/etc/resolv.conf

محل ذخیره تنظیمات شبکه مربوط به دامین ها و سرور.Domain Name Servers (DNS)

---

[۱] Mount

[۲] External

[۳] Removable Devices

[۴] Background

/usr/share

محل ذخیره فهرست‌های اشتراک گذاشته شده در میان سیستم ها

/proc/cpuinfo

محل ذخیره اطلاعت مربوط به CPU

/proc/filesystems

محل ذخیره اطلاعات مربوط به سیستم فایل که در حال حاضر استفاده می شود.

/proc/interrupts

محل ذخیره اطلاعات درباره رخ دادهای سیستم

/proc/ioports

محل ذخیره تمام اطلاعات آدرس پورت‌های ورودی و خروجی که توسط سیستم استفاده می شوند.

/proc/meminfo

محلی برای ذخیره میزان حافظه استفاده شده توسط سیستم

/proc/modules

محل ذخیره ماژول‌های کرنل

/proc/mount

محلی برای ذخیره اطلاعات مربوط به وسایل و رسانه‌های متصل شده به سیستم

/proc/stat

محل ذخیره چکیده ای از وضعیت سیستم در قالب آمار

/proc/swaps

محل ذخیره اطلاعات مربوط به حافظه مجازی سیستم

/version

محلی برای نگهداری اطلاعات مربوط به نسخه لینوکس

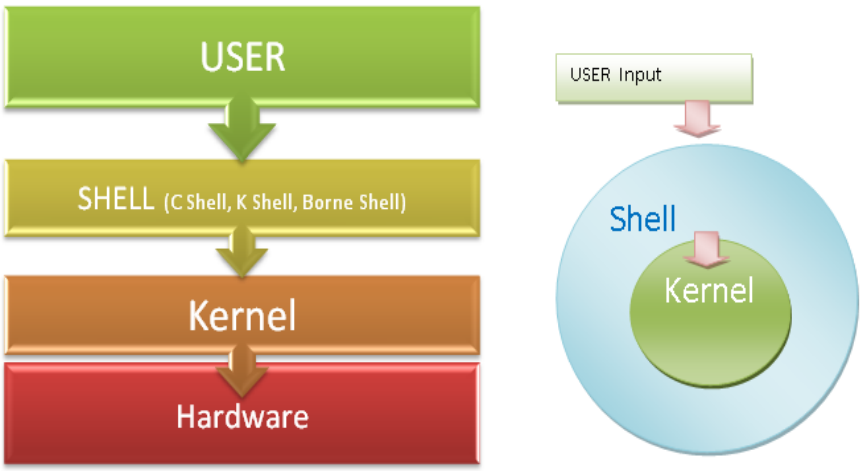
/var/log

محل ذخیره رخ دادها و وقایع سیستم

روز دوم

پوسته–شل چیست؟

به عبارت ساده، پوسته[1] یک برنامه است که دستورات را از صفحه کلید گرفته و آنها را به کرنل سیستم عامل می دهد تا اجرا شود. در زمان های قدیم، این تنها رابط کاربری موجود در سیستم یونیکس مانند لینوکس بود. امروزه ما رابطهای کاربری گرافیکی (GUI) را علاوه بر رابطهای خط دستور (CLI) مانند پوسته داریم. در اکثر سیستم های لینوکس یک برنامه به نام bash که برای Bourne Again Sheell نامیده می شود، نسخه پیشرفته یونیکس پوسته اصلی sh است که توسط Steve Bourne نوشته شده است. به عنوان برنامه پوسته عمل می کند. علاوه برbash ، دیگر برنامه های پوسته وجود دارد که می توانند در یک سیستم لینوکس نصب شوند. اینها عبارتند از:ksh ، tcshوzsh .



ترمینال در تصویر بالا ، قسمت user قرار میگیرد که به شل متصل است و شل به کرنل.

ترمینال چیست؟

برنامه ای است که یک پنجره را باز می کند و به اجازه می دهد با پوسته ارتباط برقرار کنید. تعدادی از شبیه سازهای ترمینال مختلف می‌توانید استفاده کنید. خط فرمان Linux یک رابط متنی برای رایانه است. اغلب به عنوان پوسته ، ترمینال ، کنسول ، سریع یا نام های مختلف دیگر نامیده می‌شود، استفاده از آن پیچیده و گیج کننده است. با این وجود توانایی کپی و paste دستورات از یک وب سایت ، همراه با قدرت و انعطاف پذیری خط فرمان را دارد.

[1] shell

انواع ترمینالها

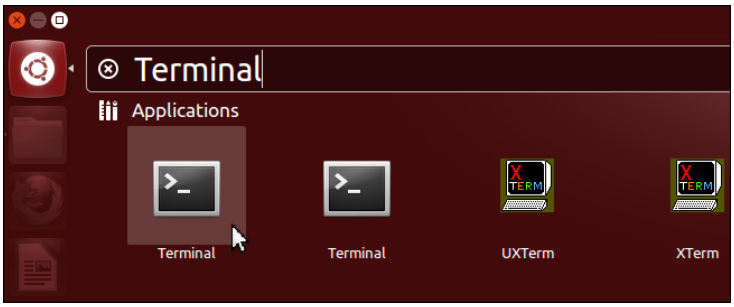
اکثر توزیعهای لینوکس دارای ترمینالهای[۱] پیش فرض خود هستند.به عبارت دیگر، شبیه ساز ترمینال این توانایی را دارد که یک ماشین گنگ را مانند یک کامپیوتر مشتری متصل به سرور به نظر برساند. شبیه ساز ترمینال به کاربر نهایی[۲] اجازه می‌دهد تا به کنسول[۳] و همچنین برنامه های کاربردی آن مانند رابط کاربری متنی و رابط خط فرمان[۴] دسترسی داشته باشد. ممکن است تعداد زیادی شبیه ساز

بازکردن ترمینال

در اکثر توزیع‌های لینوکسی دارای مدیر پنجره[۵] هستند که یک راه برای راه اندازی یک ترمینال از طریق منو دارند. میتوانید از طریق لیست برنامه ها ببینید که آیا چیزی شبیه یک شبیه ساز ترمینال وجود دارد؟به عنوان مثلا اگر یک کاربر KDE هستید، برنامه ترمینال "konsole"نام دارد، در Gnome آن را "gnome-terminal" نامگذاری می کند .یا در میزهای xfce آن را xfce4termianl می نامند. در حالی که تعدادی از شبیه سازهای مختلف ترمینال وجود دارد، همه آنها کار یکسانی را انجام می دهند. یعنی آنها به یک پوسته[۶] دسترسی دارند.

روش اول:

همانگونه که در بالا گفته شد، میتوانید بافشردن کلیدsuper key در سیستم عامل ویندوز به کلید ویندوز اطلاق میشود) منوی برنامه ها را بازکنید و عبارت ترمینال را تایپ نمایید. تابرنامه ترمینال همان میزکاری شما اجرا شود. به عنوان مثال در توزیع اوبونتو:



روش دوم:

اکثر سیستم های لینوکس از میانبر صفحه کلید زیر به عنوان کلید میانبر پیش فرض برای بازکردن آن استفاده می کنند:

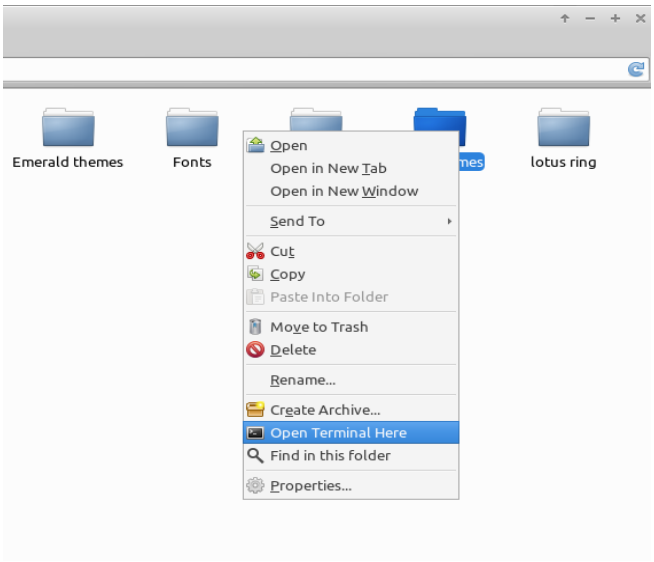
Ctrl+Alt+T

روش سوم:

در بسیاری از توزیع های لینوکسی با راست کلیک کردن روی میزکار[۷] یا داخل هر پنجره ای که باز است،گزینه ای به نام open terminal hereوجود دارد.

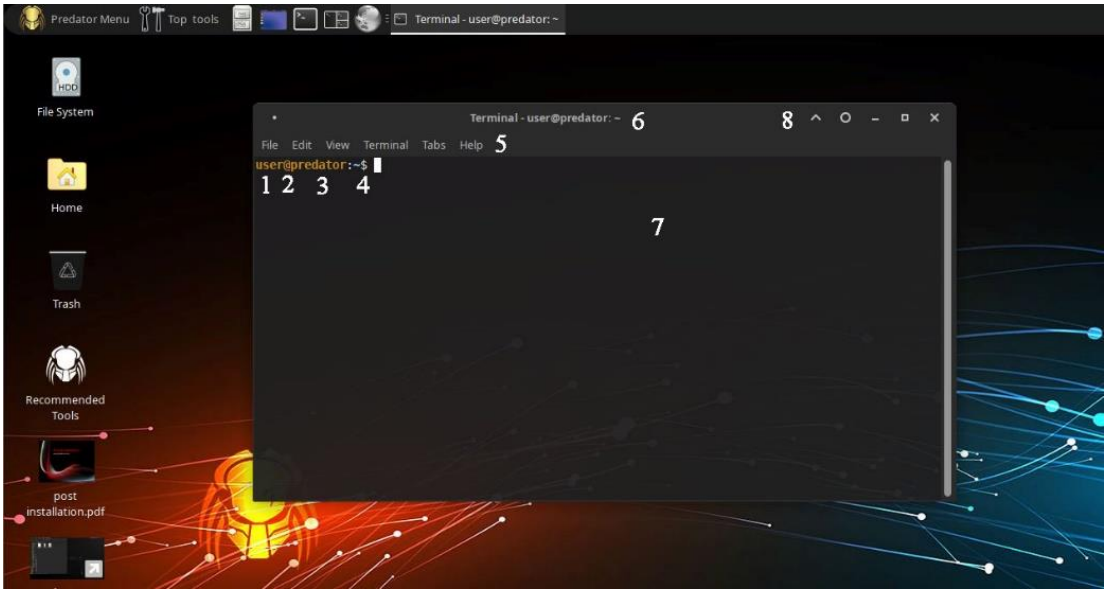
---

۱ Terminal Emulators for Linux
۲ End user
۳ console
۴ Command line
۵ file manager
۶ session shell
۷ desktop



اجزای پنجره ترمینال

پنجره ی ترمینال را بازکنید. در ابتدا قسمت هایی را به صورت پیش فرض و همیشگی در آن مشاهده خواهید کرد. مشخصاتی مانند:



۱.نام کاربر جاری[۱]

۲.علامت @ جداکننده نام کاربری و نام سیستم شما

۳. نام سیستم شما[۲]

۴.علامت $ یا علامت #. مکانی که دستورات خود را تایپ مینمایید.

علامت $ نشان دهنده کاربر عالی فعال است.و دسترسی شما کاربر نرمال است.

علامت # نشان دهنده کاربر ریشه فعال است.

۵. نوار منو

۶. نوار عنوان ترمینال که قابل تغییر هم می باشد.(ازطریق تنظیمات ترمینال)

۷.بدنه ترمینال.فضایی که خروجی دستورات مشاهده میشود.

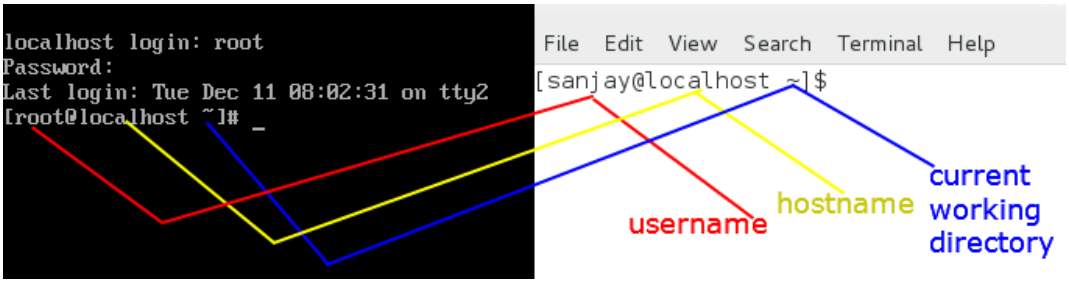
---

[۱] useraccount

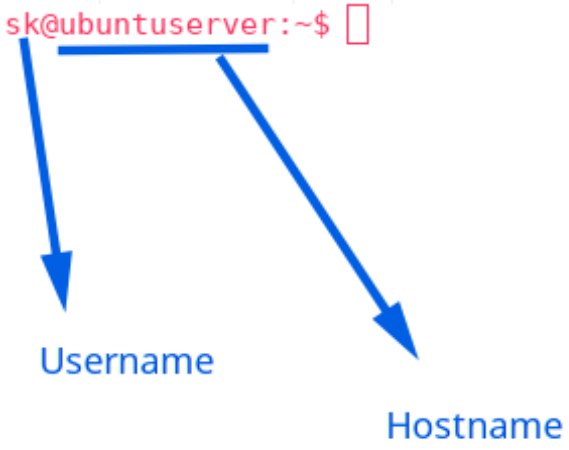
[۲] hostname

۸. دکمه های کنترل پنجره ترمینال

درابتدا که ترمینال خود را راه اندازی می کنید ، ممکنه با یک پنجره نسبتاً کسل کننده همراه باشید که کمی متن عجیب و غریب در بالا دارد ، بسته به سیستم لینوکس شما ممکن است رنگها یکسان نباشد ، و متن احتمالاً چیز دیگری را بیان می کند ، اما طرح کلی یک پنجره با یک متن متن بزرگ (عمدتا خالی) باید مشابه مواردی که ذکر کرده ام، باشد. اول اینکه وقتی دستور را تایپ می کنید از آن به عنوان prompt یادمی شود، یعنی درمقابل علامت $ یا #

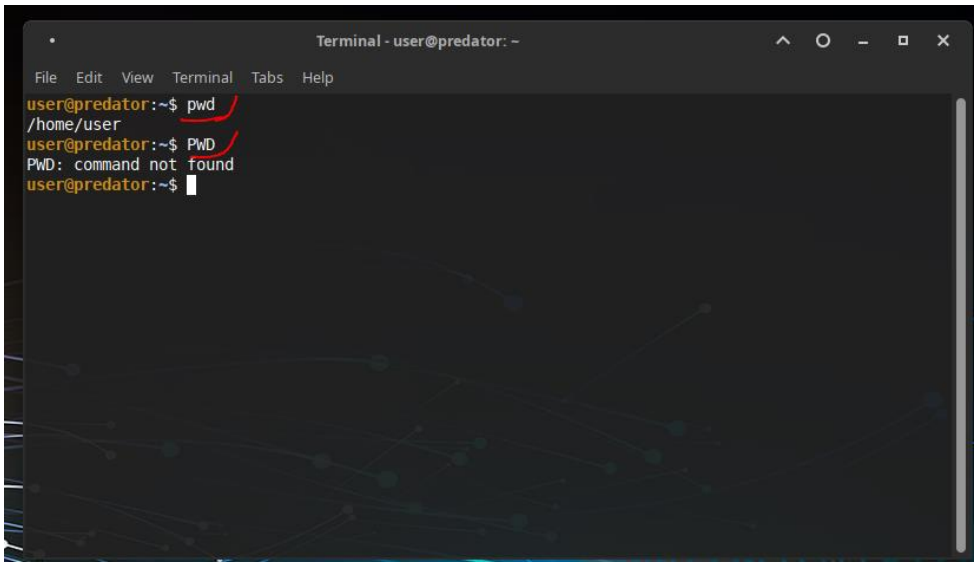


دومین چیزی که باید درک کنید این است که وقتی یک دستور را اجرا می کنید ، هر خروجی که تولید می کند ، مستقیماً در ترمینال چاپ و نمایش داده می‌شود، پس از اتمام آن یک پیام دیگر به شما نشان داده می شود. برخی از دستورات می توانند متن زیادی را چاپ کنند، برخی دیگر بی صدا کار می کنند و به هیچ وجه خروجی نخواهند داشت. اگر دستور را اجرا کردید و بلافاصله دستور دیگری ظاهر شد ، نگران نشوید ، زیرا این معمولاً به معنی موفقیت آمیز بودن دستور است.یا منتظر وارد کردن مقداری از شما می‌باشد.

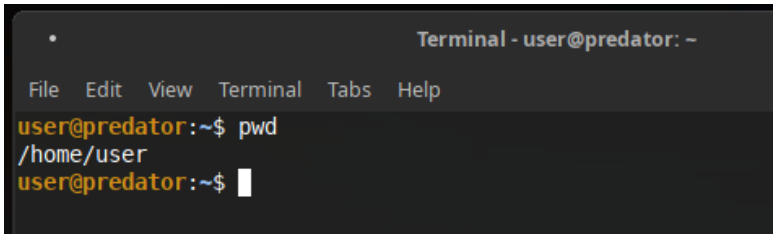


حساسیت دستورات در ترمینال

هنگام تایپ کردن در خط فرمان ترمینال، به موارد بزرگ یا کوچک تایپ کردن دستورات و مقادیر ورودی دقت کنید. تایپ کردن PWD به جای pwd خطایی ایجاد می کند، اما گاهی اوقات مورد اشتباه ممکن است منجر به اجرای دستور شود ، اما آنچه را که انتظار داشتید انجام ندهد.

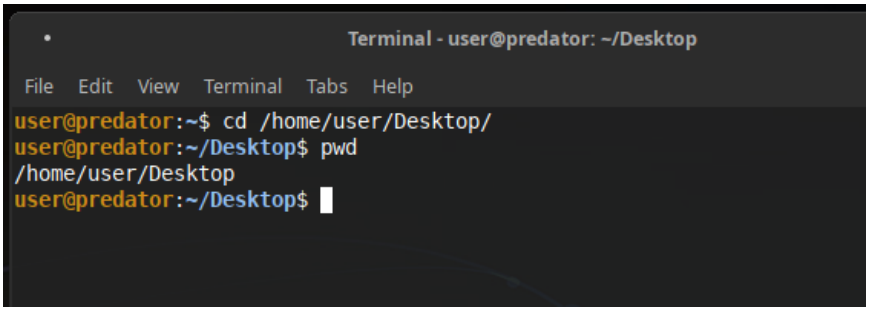


یک مفهوم دیگر که باید از آن اطلاع داشته باشید، این است که شل یک مکان پیش فرض دارد که در آن هرگونه عملیات انجام می شود.به این معنی که بعد از باز کردن ترمینال، مسیر پیش فرضی را دارد.که به آن فهرست کاری فعال[1] گفته میشود. اگر می خواهید فایل یا فهرست جدید ایجاد کنید ، پرونده های موجود را مشاهده کنید یا حتی آنها را حذف کنید ، پوسته فرض خواهد کرد که شما آنها را در فهرست کار فعلی جستجو می کنید مگر اینکه مسیر دیگری را تعیین کنید. بنابراین کاملاً مهم است که در هر زمان مشخص در مورد پوشه ای که پوسته در آن قرار دارد ، استفاده کنید ، بعد از همه اینها ، حذف پرونده ها از پوشه اشتباه می تواند فاجعه بار باشد. اگر نیاز به دانستن مسیر جاری و کاری شل در ترمینال دارید دستور pwd دقیقاً به شما می گوید که فهرست کار فعلی شما چیست.



تغییر مسیر جاری شل در ترمینال

به منظور تغییر مسیر در ترمینال جاری باز شده از دستور cd که مخفف change directiry است، صورت می‌گیرد.

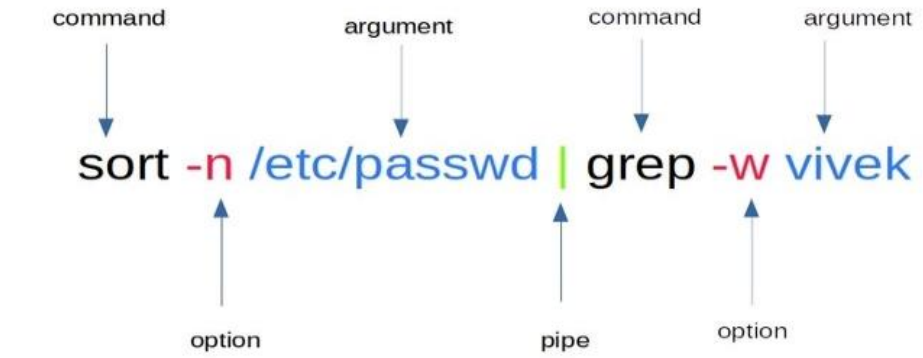


در ابتدای کار با دستورات در ترمینال سعی کنید مسیرجاری و کاری خود را با pwd مشاهده نمایید.

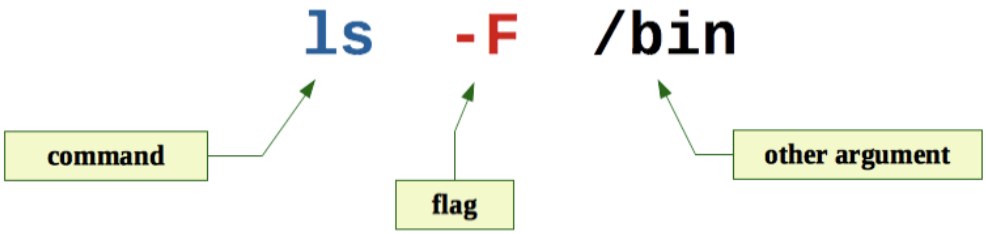
[1] Proccesing working directory

آناتومی دستور

در تصویر زیرساختار یک دستور لینوکسی را مشاهده میکنید:



یا به صورت زیر:



یک دستور در سیستم عامل لینوکس از چند بخش تشکیل شده است که هر بخش نقش وظیفه‌ای خاصی دارد. ساختار یک دستور عموماً به صورت زیر است:

دستور اصلی (:(Commandاین بخش اصلی دستور است که عملیات خاصی را انجام می‌دهد. به عنوان مثال، ls برای لیست کردن فایل‌ها و پوشه‌ها، mkdir برای ایجاد یک پوشه جدید و غیره.

گزینه‌ها یا آرگومان‌ها (:(Options/Argumentsگاهی اوقات دستورات نیاز به گزینه‌ها یا آرگومان‌های اضافی دارند تا عملکرد آنها تغییر کند یا بهتر کند. این گزینه‌ها با استفاده از خط فرمان وارد می‌شوند. به عنوان مثال، ls -l برای نمایش جزئیات بیشتر در لیست فایل‌ها و پوشه‌ها.

پارامترها (:(Parametersبرخی از دستورات ممکن است نیاز به ورودی‌های اضافی داشته باشند که به عنوان پارامترها شناخته می‌شوند. این پارامترها می‌توانند اطلاعات مربوط به فایل‌ها، پوشه‌ها، یا سایر موارد مربوط به دستور باشند.

پایان خط فرمان (:(Command Terminationهر دستور در خط فرمان با فشار دادن دکمه Enter اجرا می‌شود و بازخورد مربوط به اجرای دستور به کاربر نمایش داده می‌شود.

یک دستور لینوکس دارای option های مختلفی میباشد که خروجی دستور را تغییر میدهد. مثلا در دستور ls وقتی option را a– قرار دهید، در خروجی فایل های مخفی نمایش داده میشود.

```
$ls –a
```

درمواردی به این option گاهی flag و گاهی switch نیز گفته میشود. هر دستور ممکنه است دارای پارامترهای ورودی نیز باشه که argument گفته میشود. مثلا در دستور زیر مسیر داده شده یک نوع آرگومان mount می باشد.

```
$ls –a /usr/bin/
```

روز سوم

۱.دستور chsh

تغییر پوسته پیش فرض

از دستور chsh در لینوکس برای تغییر شل (شل سیستم فعلی) استفاده می شود. همانگونه که گفته شد، شل یک رابط کاربری تعاملی با یک سیستم عامل است و می توان آن را یک لایه خارجی از سیستم عامل دانست. پوسته bash یکی از پرکاربردترین پوسته های ورود به سیستم در لینوکس است. این دستور به کاربر اجازه می دهد پوسته را از پوسته فعلی تغییر دهد. کاربر ریشه می تواند شل با دستور chsh که مخفف change shell است، تغییر دهد.

ساختار دستور:

```
chsh [-s shell] [-R] [-h] [username]
```

پس از پیداکردن شل های نصب شده، از دستور زیر شل خود را به شل مدنظر تغییر دهید:

```
$chsh –s /bin/fish
```

مثال های دستور:

```
chsh -s /bin/zsh ## change bash to ksh ##
```

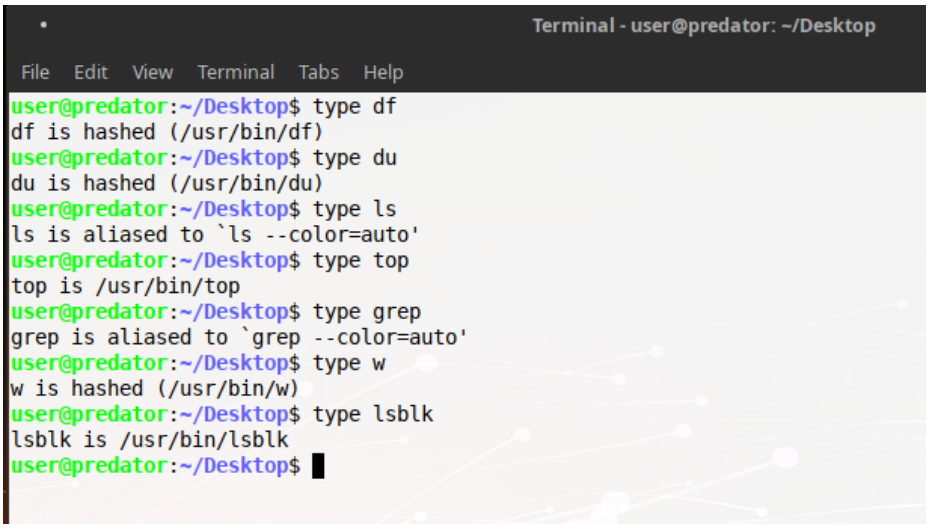
گزینه های دستور:

- **s** برای تنظیم پوسته به عنوان پوسته ورود به سیستم شما استفاده می شود.

دستور type

دستور type می‌گوید که نام کامل یک دستور داده شده چیست؟

۱. **شل[۱] ساخته شده**
۲. **کلمه کلیدی شل**
۳. **نام مستعار[۲]**



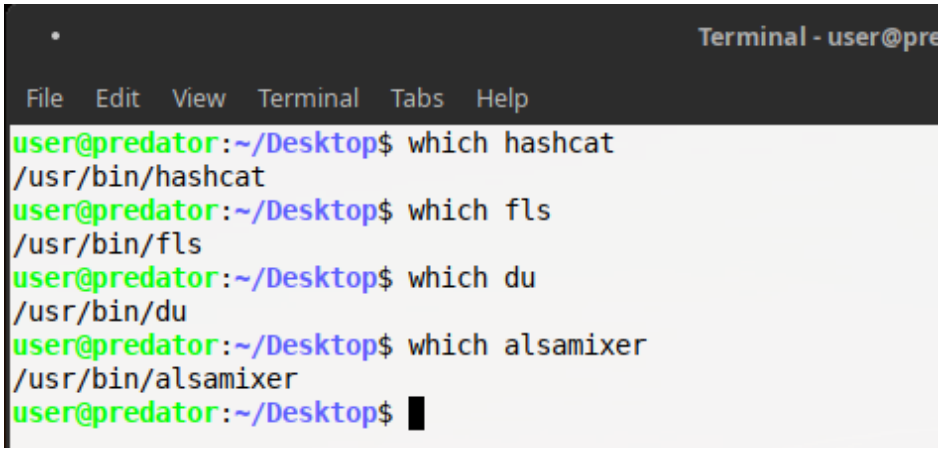
برای مشاهده تمام نام های مستعار ایجاد شده در system خود از دستور نام alias استفاده کنید:

---

[۱] shell
[۲] alias

۲.دستور which

به یک دستور کمک می کند، مسیر دستور مطلق را به صورت زیر چاپ می کند:



دستور which در سیستم‌های عامل مبتنی بر یونیکس (UNIX) و لینوکس برای پیدا کردن مسیر (path) فایل اجرایی یا برنامه استفاده می‌شود. بطور کلی، which به شما اجازه می‌دهد تا ببینید که یک فایل اجرایی یا برنامه خاص در کدام مسیر سیستم قرار دارد. این دستور بیشتر در محیط خط دستوراستفاده می‌شود.

ساختار دستور:

```
$which [OPTIONS] COMMAND
```

در اینجا، COMMAND نشان‌دهنده نام فایل اجرایی یا برنامه است که می‌خواهید مسیر آن را پیدا کنید.

مثال های دستور:

پیدا کردن مسیر یک فایل اجرایی:

```
$which gcc
```

این دستور مسیر فایل اجرایی gcc را در سیستم پیدا می‌کند و مسیر آن را نمایش می‌دهد.

پیدا کردن مسیر یک برنامه:

```
$which python
```

این دستور مسیر برنامه python را در سیستم پیدا می‌کند و مسیر آن را نمایش می‌دهد.

پیدا کردن مسیر یک برنامه با نام یکسان:

```
$which -a java
```

در اینجا، با استفاده از گزینه a- ، which تمام مسیرهای برنامه java را که با همین نام در سیستم وجود دارند، نمایش می‌دهد.

دستور which بسیار مفید است زمانی که بخواهید بفهمید که یک فایل اجرایی یا برنامه خاص در کدام مسیر سیستم قرار دارد. با استفاده از این دستور، می‌توانید مسیر صحیح را برای اجرای فایل‌ها و برنامه‌ها تعیین کنید.

روز چهارم

۳.دستور env

دستور env در لینوکس/یونیکس برای چاپ لیستی از متغیرهای محیط فعلی یا اجرای یک برنامه در یک محیط سفارشی بدون تغییر محیط فعلی استفاده می شود. دستور env در لینوکس برای نمایش یا تنظیم متغیرهای محیطی (environment variables) استفاده می‌شود. این دستور به کاربر اجازه می‌دهد تا متغیرهای محیطی را مشاهده کند، تغییر دهد یا مقدار جدیدی برای آن‌ها تعیین کند.

گزینه های دستور:

- i– یا –- ignore-environment: این گزینه باعث می‌شود که متغیرهای محیطی فعلی نادیده گرفته شوند و یک محیط پاک و جدید برای اجرای دستور مشخص شود.
- u VAR=value– یا –- unset=VAR: با استفاده از این گزینه می‌توانید یک متغیر محیطی مشخص را حذف کنید یا مقدار جدیدی به آن تعیین کنید.
- C– یا –- chdir=DIR: این گزینه مشخص می‌کند که دستور env باید در دایرکتوری مشخص‌شده اجرا شود و متغیرهای محیطی مربوط به آن دایرکتوری را نمایش دهد.
- S– یا –- split-string: این گزینه باعث می‌شود که مقادیر متغیرهای محیطی که با کاراکتر فاصله جدا شده‌اند به عنوان جداکننده متغیرها در نظر گرفته شوند.

ساختار دستور:

```
$ env [OPTION]... [-] [NAME=VALUE]... [COMMAND [ARG]...]
```

مثال های دستور:

مجموعه متغیرهای محیط فعلی را چاپ میکند.

```
$ env
```

دستوری را با محیط خالی اجرا میکند

```
$ env -i command_name
```

حذف یک متغیر از محیط محلی

```
$ env -u variable_name
```

هر خروجی را با NULL پایان میدهد.

```
$  env -0
```

شروع با یک محیط خالی گزینه i :

```
$env -i
```

پایان دادن به خطوط خروجی با NUL به جای خطوط جدید :

```
$env --null
```

حذف یک متغیر محیطی

```
$env --unset=VAR_NAME
```

تغییر فهرست کاری به یک فهرست خاص

```
$env --chdir=/path/to/directory
```

فعال کردن حالت دیباگ

```
$env --debug
```

نمایش پیام راهنما

```
$env --help
```

نمایش اطلاعات نسخه

```
$env --version
```

۴.دستور printenv

printenv مقادیر متغییرهای محیطی مشخص شده را چاپ می‌کند. متغیر(ها) . اگر VARIABLE مشخص نشده است، نام و جفت مقدار همه را چاپ میکند.

ساختار دستور

```
$ printenv [OPTION]... PATTERN...
```

گزینه های دستور

-0 --null به جای اینکه هر خط خروجی را چاپ کند ان را با 0 بایت پایان میدهد

help -- یک پیام راهنما را نمایش میدهد و از آن خارج میشود.

مثال های دستور

۱. نمایش مقادیر تمام متغیرهای محیطی

```
$ printenv
```

۲. فهرست اصلی کاربر فعلی .

```
$ printenv $HOME
```

۳. برای استفاده از گزینه خط دستور null -- به عنوان کاراکتر پایانی بین ورودی های خروجی.

```
$ printenv -- null $SHELL $HOME
```

توجه: به طور پیش فرض، printenv دستور از newline به عنوان کاراکتر پایانی بین ورودی های خروجی استفاده می‌کند.

۵.دستور cd

برای تغییر فهرست جاری از دستور cd استفاده می کنید. برای انجام این کار، cdرا تایپ کنید و پس از نام مسیر[۱] فهرست مورد نظر را وارد نمایید. نام مسیر، مسیری است که در طول شاخه های درخت می گیرید تا به فهرست که می خواهید دسترسی پیدا کنید. نام ها را می توان در یکی از دو روش مختلف مشخص کرد.نام مسیرها به صورت مطلق یا نام آنها به صورت نسبی[۲]، بیایید ابتدا با نام مطلق[۳] نگاهی بیاندازیم.

```
$ cd [directory_name]
```

نام محلی مطلق با فهرست ریشه شروع می شود و پس از ریشه توسط شاخه ها تا مسیر فهرست دلخواه ادامه می یابد. به عنوان مثال ، یک فهرست در سیستم وجود دارد که اکثر برنامه ها نصب شده اند. نام فهرست bin /usr/ است. این بدان معنی است که از فهرست ریشه (که با نماد اسلش / نشان داده میشود) یک فهرست به نام "usr" وجود دارد که حاوی فهرست به نام "bin" است. این مسیر یک فهرست مطلق است.

```
$ cd /usr/bin
$ pwd
/usr/bin
```

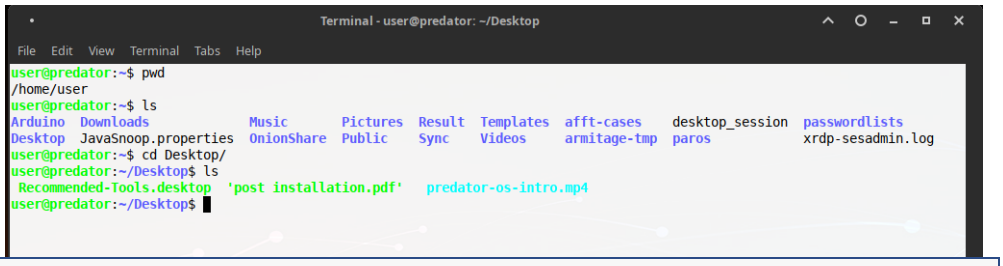
مسیر جاری bin را نشان می دهد. با دستور ls محتوای درون فهرست را مشاهده میتوان کرد.

---

[۱] path
[۲] releative path
[۳] absolute path

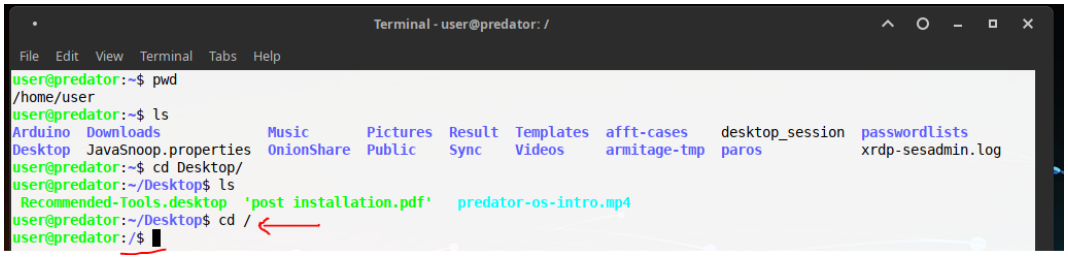
```
$ ls
 lwp-request
2to3                lwp-rget
2to3-2.6              lxterm
a2p                 lz
aalib-config          lzcat
aconnect            lzma
acpi_fakekey          lzmadec
acpi_listen         lzmainfo
add-apt-repository    m17n-db
addpart             magnifier
```

علامت نقطه در دستور cd به فهرست جاری[1] اشاره می کند و علامت ".." به فهرست قبلی از فهرست جاری اشاره می کند. هم اکنون به چگونگی کارکرد آن می پردازیم. اجازه دهید فهرست کاری را دوباره به usr / bin / تغییر دهیم:




```
cd /
```

این دستور برای تغییر فهرست به فهرست ریشه استفاده می شود ، فهرست ریشه اولین فهرست در سلسله مراتب سیستم فایل شما است.



در بالا ، / نماینده فهرست ریشه است.

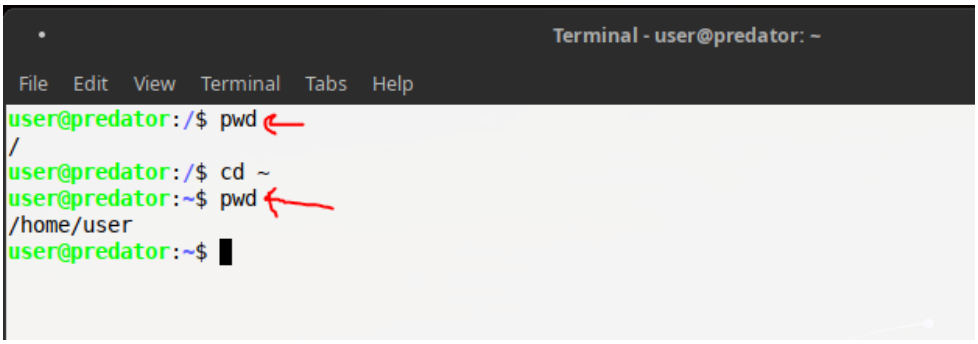
cd dir_1 / dir_2 / dir_3: از این دستور برای جابجایی درون فهرست از فهرست استفاده می شود.

```
$ cd dir_1 / dir_2 / dir_3
$cd /home/ali/Desktop/mp3
```

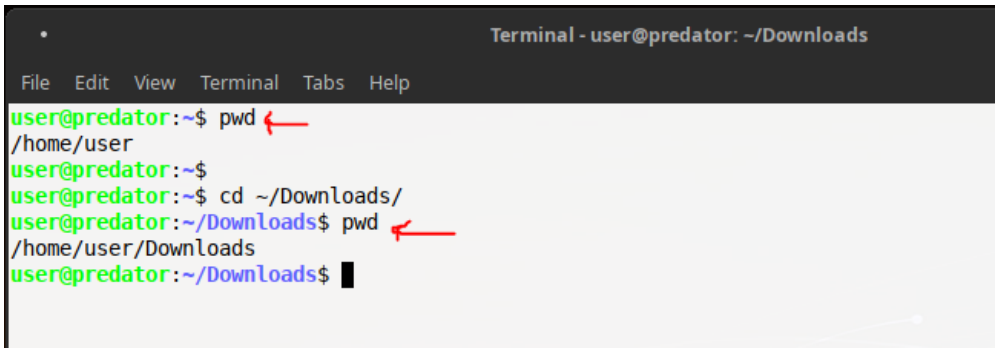
```
cd ~
```

این دستور به فهرست کاربر جاری اشاره دارد.

[1] current dir



به عنوان مثال میتوانید با دستور زیر به فهرست download از هر مسیری که باشید وارد شوید:

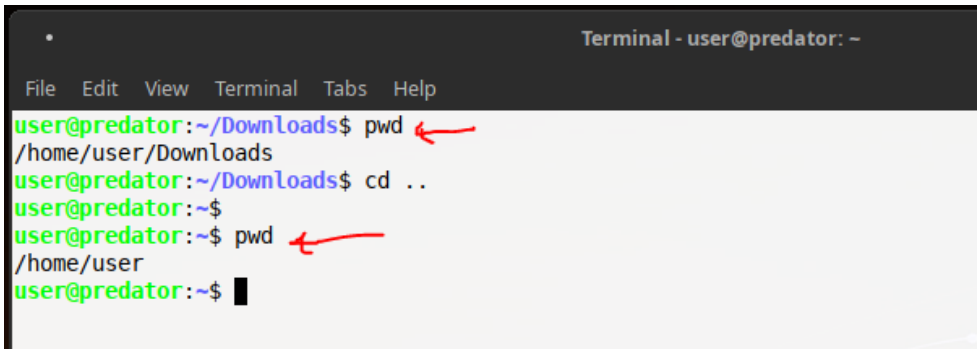


یا به مسیر فهرست کاربر جاری:

```
$cd  ~username
```

```
cd ..
```

این دستور مسیر جاری را به فهرست قبلی برمی گرداند.



روز پنجم

۶. دستور clear

دستور اجازه می دهد صفحه نمایش ترمینال را پاک[۱] کنید، به سادگی تایپ کنید.

```
$ clear
```

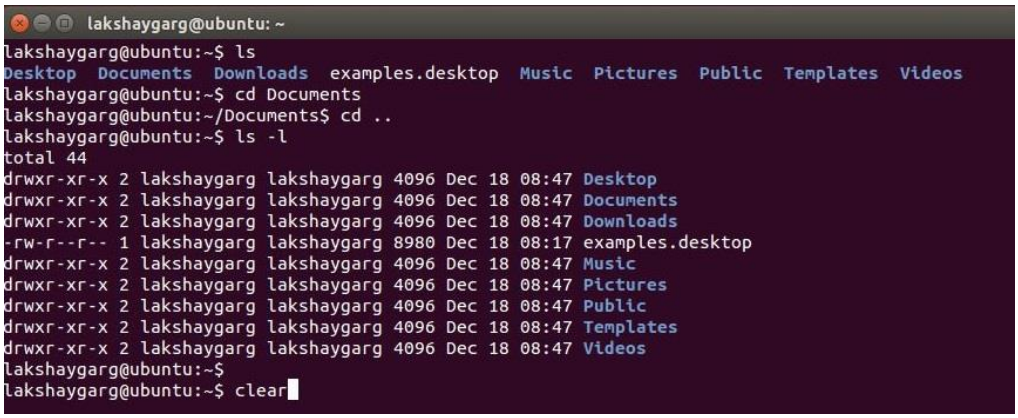
clearیک دستور استاندارد سیستم عامل رایانه یونیکس است که برای پاک کردن صفحه ترمینال استفاده می شود. این دستور ابتدا به دنبال یک نوع ترمینال در محیط است و پس از آن ، پایگاه داده **terminfo** را برای نحوه پاک کردن صفحه پیدامی‌کند و این دستورپارامترهای خط فرمان موجود را نادیده میگیرد. همچنین ، دستور **clear** هیچ گزینهی ندارد و تقریباً مشابه دستور **cls** در تعدادی از سیستم عامل های دیگر است.

ساختار دستور:

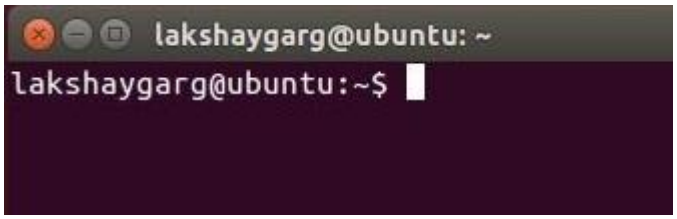
```
$clear
```

مثال های دستور:

ترمینال قبل از اجرای دستور clear :



ترمینال پس از اجرای دستور clear:



میانبر دستور clear

```
ctrl + l
```

(کلید ctrl و سپس L کوچک را بزنید)

[۱] clear

دستورات پیمایش

سیستم فایل های مدرن فهرست درختی دارند، جایی که یک فهرست یک فهرست root است که ما آن را "parent" می نامیم. system های Unix فقط یک فهرست root ای با نام \ دارند . هنگام کار در یک سیستم فایل ، کاربر همیشه در داخل یک فهرست کار می کند که ما آن را فهرست فعلی یا فهرست کاری[1] می نامیم .

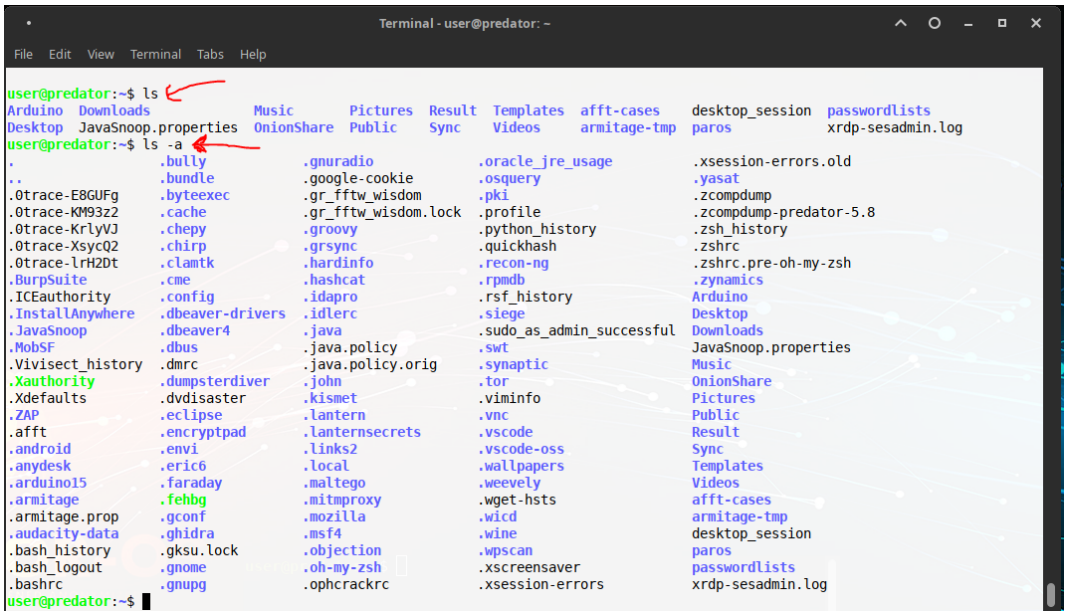
فهرست جاری کاربر را با pwd چاپ کنید :

```
$ pwd
/ home / hossein
```

لیست محتویات این فهرست (فایل ها و / یا فهرست های فرزند و غیره) را با ls :

```
$ ls
```

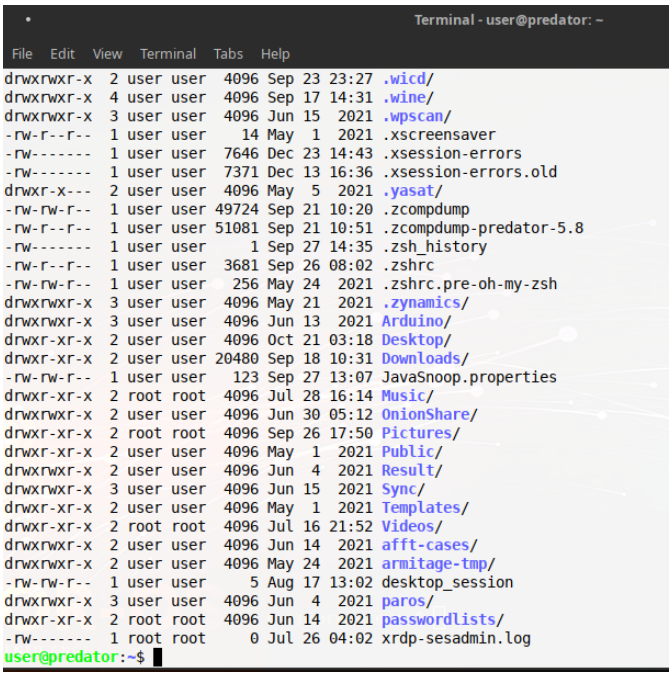
نمایش فایل های cache ("با علامت نقطه")



نمایش جزئیات فایل

```
$ ls -l
```

[1] working or current directory



ترکیب چندین گزینه

```
ls –l -a
```

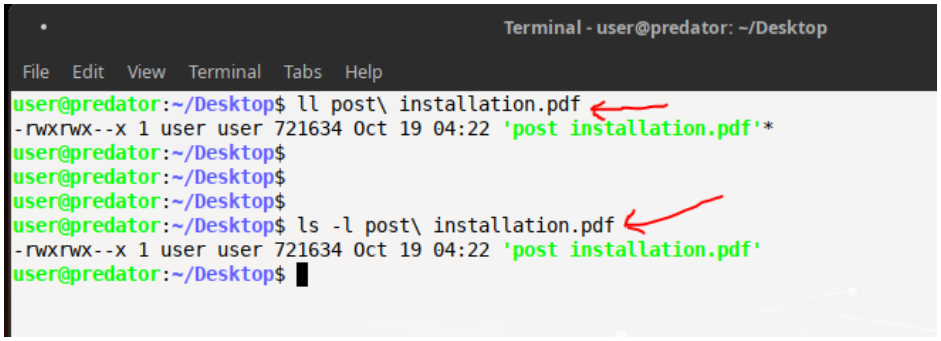
۷.دستور ls

برای فهرست محتویات یک فهرست استفاده می شود. این را می توان در تعدادی از روش های مختلف استفاده می شود. در اینجا چند نمونه است:

دستور	خروجی
ls	فهرست فایل ها را در فهرست کاری نشان می دهد
ls / bin	فهرست فایل ها را در فهرست bin / نشان می دهد
ls -l	فایلهای موجود در فهرست کاری را در قالب نوع دسترسی فهرست میکند
ls -l / etc / bin	فهرست فایل ها در فهرست bin / و فهرست etc / در فرمت دسترسی نشان می دهد
ls -la .	لیست تمام فایل ها (حتی آنها یی که اسامی که معمولا پنهان هستند) در فهرست والدین نشان داده می شوند

نگاهی دقیق تر به فرمت خروجی دستور ls

اگر از گزینه – l با ls استفاده می کنید، لیستی را که حاوی اطلاعات فراوانی در مورد فایل مربوط است، دریافت خواهید کرد. در اینجا یک مثال داریم:



خروجی شامل اطلاعات فایل از قبیل: نام فایل، زمان ویرایش فایل، اندازه فایل، نام گروه، مالکیت فایل، مجوزهای فایل .

نام فایل

نام فایل یا فهرست.

زمان اصلاح

آخرین بار فایل تغییر کرده و اگر آخرین ویرایش گذشته، بیش از شش ماه رخ داده باشد، تاریخ و سال نمایش داده می شود. در غیر این صورت، زمان روز نمایش داده می شود.

اندازه

اندازه فایل در حالت بایت.

گروه

نام گروهی که مجوزهای فایل علاوه بر صاحب پرونده[۱] دارد.

صاحب

نام کاربری که فایل را مالکیت دارد.

مجوزهای پرونده

مجوزهای دسترسی[۲] به پرونده را نشان میدهد.

```
-rw  --  --  --- 1 bshotts bshotts 576 Apr 17 1998 weather.txt
drwxr-xr-x 6 bshotts bshotts 1024 9 ۱۹۹۹ اکتبر web_page
```

اولین کاراکتر مشخصه این است که نوعی فایل است. "-" یک فایل regular (عادی) را نشان می دهد.

"d" نشان می دهد که یک فهرست است. مجموعه دوم از سه کاراکتر، خواندنread، نوشتنwriteو اجرای execute مربوط به دسترسی صاحب فایل است.

مثال های دستور:

در اینجا، ls -l (-l)یک کاراکتر است، نه یک (فایل یا فهرست، اندازه، تاریخ و زمان اصلاح شده، نام فایل یا پوشه و صاحب فایل و مجوز آن را نشان می دهد.

```
# ls -l
```

```
total 176
-rw-r--r--. 1 root root   683 Aug 19 09:59 0001.pcap
-rw-------. 1 root root  1586 Jul 31 02:17 anaconda-ks.cfg
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Desktop
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Documents
drwxr-xr-x. 4 root root  4096 Aug 16 02:55 Downloads
-rw-r--r--. 1 root root 21262 Aug 12 12:42 fbcmd_update.php
-rw-r--r--. 1 root root 46701 Jul 31 09:58 index.html
-rw-r--r--. 1 root root 48867 Jul 31 02:17 install.log
-rw-r--r--. 1 root root 11439 Jul 31 02:13 install.log.syslog
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Music
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Pictures
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Public
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Templates
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Videos
```

---

[۱] owner

[۲] permission

مشاهده فایل های مخفی در لینوکس

فهرست همه فایل ها از جمله فایل های مخفی که با "." شروع می شود.

```
# ls -a
```

```
.                .bashrc         Documents        .gconfd
install.log      .nautilus       .pulse-cookie
..               .cache          Downloads        .gnome2
install.log      .netstat.swp    .recently-used.xbel
0001.pcap        .config         .elinks          .gnome2_private
.kde             .opera          .spice-vdagent
anaconda-ks.cfg  .cshrc          .esd_auth          .gtk-bookmarks
.libreoffice      Pictures       .tcshrc
.bash_history    .dbus           .fbcmd             .gvfs
```

فهرست فایل ها با فرمت خوانا توسط انسان

با ترکیبی از گزینه lh-، اندازه ها را در قالب قابل خواندن توسط انسان نشان می دهد.

```
# ls -lh
```

```
total 176K
-rw-r--r--. 1 root root  683 Aug 19 09:59 0001.pcap
-rw-------. 1 root root 1.6K Jul 31 02:17 anaconda-ks.cfg
drwxr-xr-x. 2 root root 4.0K Jul 31 02:48 Desktop
drwxr-xr-x. 2 root root 4.0K Jul 31 02:48 Documents
drwxr-xr-x. 4 root root 4.0K Aug 16 02:55 Downloads
-rw-r--r--. 1 root root  21K Aug 12 12:42 fbcmd_update.php
-rw-r--r--. 1 root root  46K Jul 31 09:58 index.html
-rw-r--r--. 1 root root  48K Jul 31 02:17 install.log
-rw-r--r--. 1 root root  12K Jul 31 02:13 install.log.syslog
drwxr-xr-x. 2 root root 4.0K Jul 31 02:48 Music
drwxr-xr-x. 2 root root 4.0K Jul 31 02:48 Pictures
drwxr-xr-x. 2 root root 4.0K Jul 31 02:48 Public
drwxr-xr-x. 2 root root 4.0K Jul 31 02:48 Templates
```

فهرست فایل ها و فهرست ها با نویسه "/" در پایان

استفاده از گزینه F- با دستور ls، کاراکتر '/' را در انتهای هر فهرست اضافه می کند.

```
# ls -F
```

```
0001.pcap        Desktop/    Downloads/        index.html
install.log.syslog  Pictures/  Templates/
onda-ks.cfg  Documents/  fbcmd_update.php  install.log
Music/              Public/    Videos/
```

فهرست کردن فایل ها به ترتیب معکوس در لینوکس

دستور زیر با گزینه ls -r فایل ها و فهرست ها را به ترتیب معکوس نمایش می دهد.

```
# ls -r
```

```
Videos      Public     Music                install.log
fbcmd_update.php  Documents  anaconda-ks.cfg
Templates  Pictures  install.log.syslog  index.html
Downloads          Desktop    0001.pcap
```

فهرست فهرست های فرعی را در لینوکس به صورت بازگشتی

گزینه ls -R درخت های فهرست فهرست بسیار طولانی را فهرست می کند. نمونه ای از خروجی دستور را ببینید.

```
# ls -R
```

```
total 1384
-rw-------. 1 root     root      33408 Aug  8 17:25 anaconda.log
-rw-------. 1 root     root      30508 Aug  8 17:25 anaconda.program.log

./httpd:
total 132
```

فهرست کردن فایل ها و فهرست ها به ترتیب معکوس در لینوکس

ترکیبی از ltr– آخرین ویرایش فایل یا تاریخ فهرست را نشان می دهد.

```
# ls -ltr
```

```
Total 176
-rw-r--r--. 1 root root 11439 Jul 31 02:13 install.log.syslog
-rw-r--r--. 1 root root 48867 Jul 31 02:17 install.log
-rw-------. 1 root root  1586 Jul 31 02:17 anaconda-ks.cfg
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Desktop
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Videos
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Templates
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Public
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Pictures
```

مرتب سازی فایل ها بر اساس اندازه فایل در لینوکس

با ترکیبی از lS– اندازه فایل را به ترتیب نمایش می دهد، ابتدا در اندازه بزرگ نمایش داده می شود.

```
# ls -lS
```

```
-rw-r--r--. 1 root root 48867 Jul 31 02:17 install.log
-rw-r--r--. 1 root root 46701 Jul 31 09:58 index.html
-rw-r--r--. 1 root root 21262 Aug 12 12:42 fbcmd_update.php
-rw-r--r--. 1 root root 11439 Jul 31 02:13 install.log.syslog
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Desktop
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Documents
drwxr-xr-x. 4 root root  4096 Aug 16 02:55 Downloads
drwxr-xr-x. 2 root root  4096 Jul 31 02:48 Music
```

نمایش شماره Inode فایل یا فهرست

می‌توانیم تعدادی اعداد را که قبل از نام فایل/فهرست چاپ شده‌اند، ببینیم. با گزینه های i– فایل/فهرست را با یک عدد inode لیست می کند.

```
# ls -i
```

```
2112 0001.pcap        23610 Documents         23793 index.html
23611 Music            23597 Templates         23564 anaconda-ks.cfg
23595 Downloads        22 install.log         23612 Pictures  23613 Videos
23594 Desktop          23585 fbcmd_update.php     35 install.log.syslog
23601 Public
```

بررسی نسخه دستور ls

نسخه دستور ls را بررسی کنید.

```
# ls --version
```

نمایش UID و GID فایل ها

برای نمایش UID و GID فایل ها و فهرست ها. از گزینه n– با دستور ls استفاده کنید.

```
# ls -n
```

```
total 36
drxr-xr-x. 2 500 500 4096 Aug  2 01:52 Downloads
drwxr-xr-x. 2 500 500 4096 Aug  2 01:52 Music
drwxr-xr-x. 2 500 500 4096 Aug  2 01:52 Pictures
```

دستور ls در یک نگاه

گزینه ls	توضیحات
ls -a	در لینوکس، فایل‌های مخفی با نماد . (نقطه) شروع می‌شوند و در فهرست معمولی قابل مشاهده نیستند. دستور (ls -a) لیست کامل فهرست فعلی را با اعمال فایل‌های مخفی نشان می‌دهد.
ls -l	لیست را به صورت فرمت لیست بلند نمایش می‌دهد.
ls -lh	این دستور اندازه فایل‌ها را به صورت خوانا برای انسان نمایش می‌دهد. اندازه فایل بسیار دشوار است وقتی به صورت بایت نمایش داده می‌شود. دستور (ls -lh) اطلاعات را به صورت مگابایت، گیگابایت، ترابایت و غیره نمایش می‌دهد.
ls -lhS	اگر می‌خواهید فایل‌های خود را به ترتیب نزولی (بزرگترین در بالا) بر اساس اندازه‌شان نمایش دهید، می‌توانید از دستور (ls -lhS) استفاده کنید.
ls -l --block-size=[SIZE]	برای نمایش فایل‌ها در یک فرمت اندازه خاص استفاده می‌شود. در اینجا، در [SIZE] می‌توانید اندازه را بر اساس نیاز خود تعیین کنید.
ls -d */	برای نمایش تنها زیرفهرست‌ها استفاده می‌شود.
ls -lG یا ls -g	با این دستور می‌توانید ستون اطلاعات گروه و مالک را حذف کنید.
ls -n	برای چاپ شناسه گروه و شناسه مالک به جای نام‌هایشان استفاده می‌شود.
ls --color=[VALUE]	این دستور برای چاپ لیست به رنگی یا بی‌رنگ استفاده می‌شود.
ls -li	این دستور شماره شاخص را چاپ می‌کند اگر فایل در ستون اول باشد.
ls -p	برای شناسایی آسان فهرست با علامت خط (/) نشان می‌دهد.
ls -r	برای چاپ لیست به ترتیب برعکس استفاده می‌شود.
ls -R	محتوای زیرفهرست‌ها را نیز نمایش می‌دهد.
ls -lX	فایل‌ها با انتهای یکسان را در لیست گروه‌بندی می‌کند.
ls -lt	با نمایش فایل‌های اخیراً تغییر یافته در بالای لیست، لیست را مرتب می‌کند.
ls ~	محتوای فهرست خانه را نشان می‌دهد.
ls ../	محتوای فهرست والدین را نشان می‌دهد.
ls --version	ورژن دستور ls را بررسی می‌کند.

روز ششم

۸.دستور cd

تغییر به یک فهرست دیگر با cd

در لینوکس دستور (cdتغییر فهرست) یکی از مهمترین و پرکاربردترین دستورات برای مبتدیان و همچنین مدیران سیستم است. برای ادمین‌های سرور cdتنها راه برای پیمایش به فهرست برای بررسی گزارش، اجرای یک برنامه و برای هر کار دیگری است. برای مبتدیان این یکی از دستورات اولیه است که با آن دست های خود را کثیف می کنند.

```
$ cd TEST
```

```
cd  A
pwd
A
cd ..
```

برای تغییر به فهرست parent" استفاده می شود:

```
# cd..
/ home / hossein / TEST
```

cd ~ یا فقط cd برای " تغییر مسیر به به فهرست اصلی است.

```
cd~
```

نکته:

cd ~ کاربر به معنای " cd به فهرست خانگی کاربر است

می توانید چند فهرست را با cd .. /..تغییر دهید

```
cd ../..
```

بازگشت به آخرین فهرست با cd–

```
cd -
```

چیزهایی که ما در خط دستور وارد میکنیم ، دستورات نامیده میشوند و آنها همواره برخی از کد ماشین[۱] را که در جایی از رایانه ذخیره شده اند اجرا می کنند. گاهی اوقات این کد دستگاه یک دستور linux داخلی است، گاهی اوقات این برنامه است. گاهی اوقات، ما می خواهیم یک دستور را درست بعد از دیگری اجرا کنیم. برای انجام این کار می توانیم از ؛ (سمیکالن[۲]) استفاده کنیم.

[۱] machine code

[۲] semicolon

در بالا، semicolon به این معنی است که من برای اولین بار (ls) فهرست محتویات فهرست کار را لیست کرده و سپس (pwd) محل فهرست جاری خود را چاپ می کنم.

مثال های دستور

۱.از فهرست فعلی به /usr/local تغییر مسیر دهید.

```
ali@predator-os  : ~$ cd /usr/local
ali @ predator-os :/usr/local$
```

۲. با استفاده از مسیر مطلق، از فهرست فعلی به /usr/local/lib تغییر دهید.

```
ali@predator-os  :/usr/local$ cd /usr/local/lib

ali@predator-os  :/usr/local/lib$
```

۳. با استفاده از مسیر نسبی، از فهرست کاری فعلی به /usr/local/lib تغییر دهید.

```
mypc@predator-os  :/usr/local$ cd lib

mypc@predator-os  :/usr/local/lib$
```

۴. به فهرست قبلی که قبلاً در آن کار می کردید، برگردید.

```
mypc@predator-os  :/usr/local/lib$ cd -

/usr/local
mypc@predator-os  :/usr/local$
```

فهرست فعلی را به فهرست والد تغییر دهید.

```
mypc@predator-os  :/usr/local/lib$ cd ..

mypc@predator-os  :/usr/local$
```

۵.نمایش آخرین فهرست کاری از جایی که ما حرکت کردیم (از سوئیچ '–' استفاده کنید) همانطور که نشان داده شده است.

```
mypc@predator-os  :/usr/local$ cd --
/home/mypc
```

۶.دو فهرست را از جایی که اکنون هستید به بالا ببرید.

```
mypc@predator-os  :/usr/local$ cd ../ ../

mypc@predator-os  :/usr$
```

۷. از هر کجا به فهرست اصلی کاربران بروید.

```
mypc@predator-os  :/usr/local$ cd ~

mypc@predator-os  :~$
یا
mypc@predator-os  :/usr/local$ cd
mypc@predator-os  :~$
```

۸.فهرست کاری را به فهرست کاری فعلی تغییر دهید (به نظر می رسد در حالت کلی هیچ استفاده ای از آن وجود ندارد).

```
mypc@predator-os  :cd ~/Downloads
```

۹. فهرست فعلی شما "/usr/local/lib/python3.4/dist-packages/" است، آن را در یک دستور با بالا رفتن در فهرست به "/home/mypc/Desktop/"تغییر دهید. '/' سپس با استفاده از مسیر مطلق.

```
mypc@predator-os :/usr/local/lib/python3.4/dist-packages$ cd ../../../../../home/mypc/Desktop/

mypc@predator-os :~/Desktop$
```

۱۰. از فهرست کاری فعلی به /var/www/html بدون تایپ کامل با استفاده از TAB تغییر دهید.

```
mypc@predator-os :/var/www$ cd /v<TAB>/w<TAB>/h<TAB>

mypc@predator-os :/var/www/html$
```

۱۱. از فهرست کاری فعلی خود به /etc/v___ _ بروید، اوه !شما نام فهرست را فراموش کرده اید و قرار نیست از TAB استفاده کنید.

```
mypc@predator-os :~$ cd /etc/v*

mypc@predator-os :/etc/vbox$
```

توجه : فقط در صورتی که فقط یک فهرست وجود داشته باشد که با " **v** " شروع می شود، به " **vbox** " منتقل می شود. اگر بیش از یک فهرست که با ' **v** ' شروع می شود وجود داشته باشد، و هیچ معیار بیشتری در خط فرمان ارائه نشده باشد، به اولین فهرست که با ' **v** ' شروع می شود، بر اساس حروف الفبا به عنوان حضور آنها در فرهنگ لغت استاندارد منتقل می شود.

۱۲. بدون استفاده از **TAB**

باید مسیر کامل را وارد نمایید :

```
mypc@predator-os : cd  /home/ali
mypc@predator-os :~$
```

۱۳. Pushd و Popd در لینوکس چیست؟

Pushdو popd دستورات لینوکس در bash و پوسته خاصی هستند که مکان فهرست کاری فعلی را در حافظه ذخیره می کنند و به ترتیب از حافظه به عنوان فهرست کاری فعلی و همچنین فهرست تغییرات به فهرست می آورند.

```
mypc@predator-os :~$ pushd  /var/www/html
/var/www/html
mypc@predator-os :/var/www/html$
```

دستور بالا مکان فعلی را در حافظه ذخیره می کند و به فهرست درخواستی تغییر می دهد. به محض اینکه popd فعال می شود، مکان فهرست ذخیره شده را از حافظه دریافت می کند و آن را به فهرست فعلی تبدیل می کند.

```
mypc@predator-os :/var/www/html$ popd
~
mypc@predator-os :~$
```

۱۴. به فهرست حاوی فاصله با استفاده از علامت اسلش= \ تغییر مسیردهید.

```
mypc@predator-os :~$ cd  test\predator-os /

mypc@ predator-os :~/test predator-os $

یا
```

```
mypc@predator-os :~$ cd 'test predator-os '
 mypc@predator-os :~/test predator-os $

 یا

mypc@predator-os :~$ cd "test predator-os "/
 mypc@predator-os :~/test predator-os $
```

۱۵. از فهرست کاری فعلی به Downloads تغییر مسیردهید و تمام تنظیمات آن را یکجا فهرست کنید.

```
mypc@predator-os :/usr$ cd ~/downloads && ls
```

۹.دستور TLDR

یکی از روشهای رایج ترین و قابل اطمینان برای دریافت راهنما در سیستم های یونیکس از طریق صفحه های شخصی است. صفحات Manمستندات استاندارد برای هر یونیکس مانند سیستم هستند و آنها را به کتابچه های آنلاین برای برنامه ها، توابع، کتابخانه ها، تماس های سیستم، استانداردهای رسمی و کنوانسیون ها، فرمت های فایل و غیره مطابقت دارد.

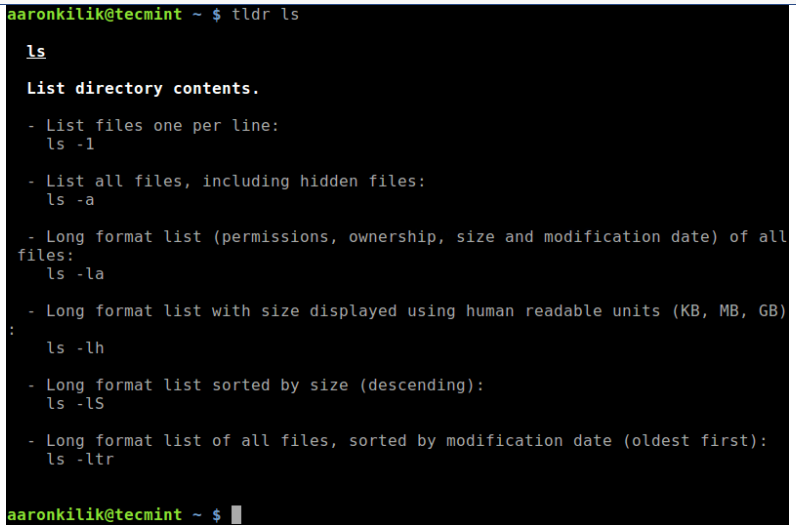
TLDRبه معنای Too Long; Didn't Read خلاصه ای از دستورات در سیستم عامل های مختلف از جمله لینوکس است.

محتوای صفحات TLDR به طور آشکار تحت مجوز MIT مجاز است.

```
Install TLDR
$  sudo npm install -g tldr
$  sudo snap install tldr
$ sudo dnf install tldr
$ sudo apt install tldr
```

مثال:

```
$  tldr ls
```



دستور tldr یک ابزار خط فرمان است که برای نمایش خلاصه‌های ساده و قابل فهم از دستورات خط فرمان استفاده می‌شود. نام tldr مخفف عبارت " "Too Long; Didn't Readاست و هدف آن ارائه توضیح مختصر و ساده از دستورات خط فرمان است تا کاربران بتوانند به سرعت و با سادگی درک کنند چگونه از هر دستور استفاده کنند. با اجرای دستور tldrبه همراه نام دستور خط فرمانی که می‌خواهید راهنمایی درباره آن را ببینید، می‌توانید توضیحات خلاصه و مثال‌های کاربردی را در مورد آن دستور ببینید. این توضیحات

معمولاً به صورت مختصر، ساده و قابل درک برای کاربران عادی ارائه می‌شود. برای استفاده از tldr، باید ابتدا آن را در سیستم خود نصب کنید. این ابزار در برخی توزیع‌های لینوکس و سیستم‌عامل macOS وجود دارد و می‌توانید آن را از طریق یک مدیر بسته نصب کنید.

بعد از نصب، می‌توانید tldrرا با استفاده از دستور زیر فراخوانی کنید:

نام دستور tldr

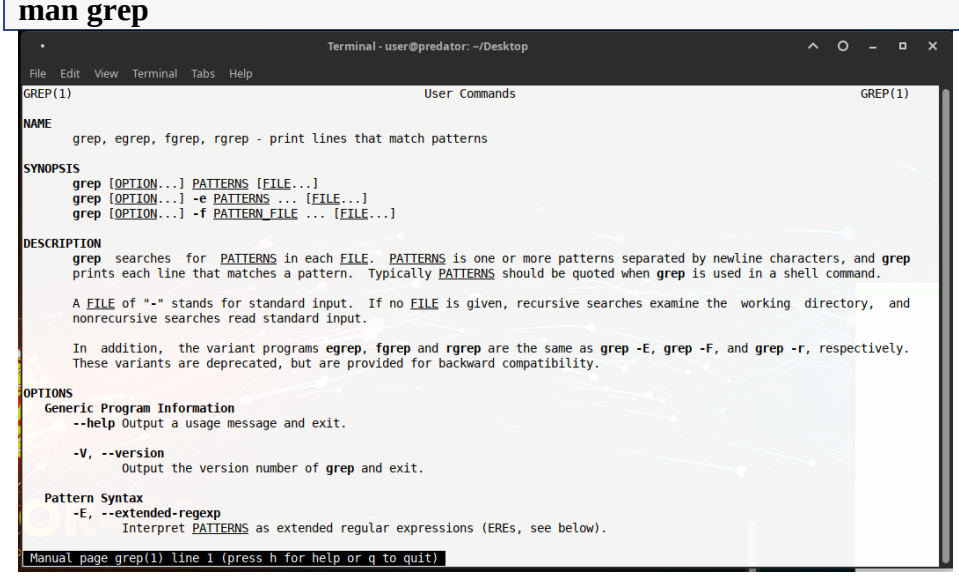
بعد از اجرای این دستور، خلاصه و مثال‌هایی از استفاده صحیح دستور مورد نظر را خواهید دید. این راهنما می‌کند تا به سرعت از دستورات خط فرمان استفاده کنید و به سادگی و با فهم بیشتر با آنها کار کنید.

روز هفتم

۱۰.دستور man

دستور man قبل از هر دستور manual آن را نشان می دهد. man یک اصطلاح کوتاه برای صفحه دستی است. در سیستم عامل های یونیکس مانند لینوکس، manرابطی برای مشاهده راهنمای مرجع سیستم است. کاربر می تواند با تایپ man و سپس یک فاصله و سپس آرگومان، درخواست نمایش یک صفحه man را بدهد. در اینجا آرگومان آن می تواند یک دستور، ابزار یا تابع باشد. یک صفحه دستی مرتبط با هر یک از این آرگومان ها نمایش داده می شود. اگر شماره بخش را در دستور ارائه کنید، به man هدایت می‌شود تا شماره بخش راهنما را بررسی کند و صفحه آن بخش نمایش داده می‌شود. و اگر نه، به طور پیش فرض صفحه اول را نمایش می دهد و باید تمام بخش ها را به صورت از پیش تعریف شده مرور کنید. manیک کتابچه راهنمای داخلی برای استفاده از دستورات لینوکس است. به کاربران اجازه می دهد تا به راهنمای مرجع یک ابزار یا دستوری برای اجرا در ترمینال دسترسی داشته باشند. دستور man حاوی توضیحات فرمان، مثال‌ها، پرچم‌ها، گزینه‌های قابل اجرا و سایر عناصر اطلاعاتی است.

man grep



man ls

این دستور تمام اطلاعات مربوط به فرمان 'ls'را همانطور که در تصویر نشان داده شده است نمایش می دهد. به‌طور پیش‌فرض، فرمان manهر جزء موجود کتابچه راهنمای کاربر را جستجو می‌کند و مطابقت اولیه را نمایش می‌دهد (حتی زمانی که صفحه در بسیاری از بخش‌ها در دسترس است). با دادن شماره بخش، دستور man برای جستجو در یک بخش خاص را نشان می دهد.

۹ بخش دستی به شرح زیر فهرست شده و توضیح داده شده است:

دستورات عمومی: دستوراتی را که در ترمینال استفاده می شود را نشان می دهد.

توابع کتابخانه: توابع موجود در کتابخانه های برنامه را نشان می دهد.

فراخوانی سیستم: اینها توابع ارائه شده توسط هسته را نشان می دهند.

فایل های ویژه: دستگاه ها معمولا در فهرست / dev و درایورهای مربوطه یافت می شوند.

بازی ها: توضیحات دستوری را نشان می دهد که نقل قول های پایگاه داده را نشان می دهد.

فرمت ها و قراردادهای فایل: فرمت های فایل مانند etc/passwd را نشان می دهد.

متفرقه: توضیحات مختلف، مانند پارامترهای بوت، قراردادها و بسته‌های ماکرو و موارد دیگر.

روال کرنل: اطلاعات مربوط به عملیات داخلی هسته را نمایش می دهد.
دستورات مدیریت سیستم: اکثر دستورات برای root محفوظ هستند.

چگونه از دستور man در لینوکس استفاده کنیم؟

دستور man را تایپ کنید و به دنبال آن نام فرمان لینوکس که مایلیم صفحه man را در صفحه ترمینال ببینیم، وارد کنید.
نتیجه طولانی است. ما می توانیم از چرخ اسکرول ماوس، کلیدهای بالا و پایین یا کلیدهای PgUp و PgDn برای کار از طریق آن استفاده کنیم. می‌توانیم روی دکمه H کلیک کنیم تا بخش راهنما و یک جدول کلید احتمالی را پیدا کنیم تا پس از اجرای دستور man در خروجی حرکت کنیم.
برای خروج می توانیم دکمه Q را فشار دهیم.
خروجی فرمان عنوان موجود صفحه man را برای دستور نامگذاری شده نشان می دهد.

در زیر لیست سرفصل ها وجود دارد:

Name : نام فرمان را نشان می دهد.
خلاصه: نحو دستور را نشان می دهد.
مثال‌ها: در این عنوان، مثال‌های زیادی استفاده از دستور را نشان می‌دهند.
پیکربندی: اطلاعات پیکربندی یک دستگاه را نشان می دهد.
پیش فرض ها: آنها عملکرد پیش فرض فرمان و نحوه کنترل آنها را نشان می دهند.
گزینه های دستور: آنها لیستی از پرچم ها و گزینه هایی را که دستور می گیرد نشان می دهند.
وضعیت خروج: فهرستی از مقادیر وضعیت خروج ممکن برای یک فرمان را نشان می دهد.
فایل ها: آنها یک لیست فایل را نشان می دهند که توسط دستور استفاده می شود.
Environment: شرح و لیستی از متغیرهای محیطی که بر دستور تأثیر می گذارند را نمایش می دهد.
همچنین ببینید: این دستورات مربوط به موضوع مشخص شده را نشان می دهد.
تاریخچه: تاریخچه توسعه فرمان را نشان می دهد.
نویسندگان: این عنوان شخصی را نشان می دهد که دستور را نوشته و مدیریت کرده است.
یادداشت ها: این عنوان یادداشت های بسیاری از جمله وابستگی ها، مجوزهای مورد نیاز و غیره را نشان می دهد.
اشکالات: این عنوان مشکلات شناخته شده در نسخه برنامه را نشان می دهد.
مهم: صفحات ممکن است شامل تعداد کمتر یا بیشتر عنوان با تکیه بر محتوای صفحه مرد باشند.

نحوه نمایش خروجی

manخروجی خود را از طریق پیجر نمایش می دهد. پیجر برنامه‌ای است که خروجی‌های خود را در یک زمان پر از صفحه نمایش می‌دهد، به این معنی که کل متن به یکباره ظاهر نمی‌شود و گزینه‌ای برای اسکرول کردن صفحه وجود ندارد.
دو نقطه در پایین انتهای صفحه روی صفحه را نشان می دهد. برای رفتن به صفحه بعدی می توانید از space bar یا f و برای رفتن به صفحه عقب می توانید از ‘b’ استفاده کنید. برای خروج از صفحه روی صفحه از ‘q’ استفاده کنید و به برنامه پوسته هدایت خواهید شد. و برای راهنما، ‘h’را فشار دهید.

بخش ها در صفحه man

صفحه man به بخش های مختلفی تقسیم می شود. هر بخش بر اساس موضوع خاصی تقسیم شده است.صفحات man دارای یک عدد در داخل پرانتز بعد از دستور هستند. این اعداد نشان دهنده شماره بخش است. در تصویر بالا، LS(1)را در بالا مشاهده می کنید که نشان می دهد از بخش ۱ است. همانطور که قبلاً گفتم، اگر شماره بخش خاصی را ذکر کنید، صفحه مرد شما به آن بخش هدایت می شود. در غیر این صورت باید کل بخش ها را به صورت از پیش تعریف شده طی کنید تا به قسمت مورد نظر خود بروید.

بخش ها:

- برنامه های اجرایی و دستورات پوسته
- تماس های سیستمی
- تماس های کتابخانه
- فایل های خاص
- فرمت ها و قراردادهای فایل
- بازی ها
- متفرقه
- دستورات مدیریت سیستم
- روتین های هسته
- Tcl/Tk (یک زبان برنامه نویسی

گزینه های دستور

- man –aw :نمایش تمام بخش‌های موجود یک دستور.
- man –a : نمایش همه صفحات man یک دستور.
- man -k (apropos) نمایش لیستی از نتایج در صفحه man که شامل یک کلمه کلیدی هستند.
- man -f, whatis : نمایش توضیحات از صفحه دستوری اگر موجود باشد.
- man whereis : استفاده برای تعیین مکان یک صفحه man.
- بدون گزینه: نمایش کامل راهنمای دستور.
- man Section-num : نمایش فقط یک بخش خاص از راهنما.
- man –w : نمایش مکانی که صفحه man یک دستور مشخص در آن قرار دارد.
- man –I : اجرای دستور به صورت حساس به بزرگی/کوچکی حروف.
- man --default, -D : بازنشانی رفتار دستور man به حالت پیش‌فرض.
- man --debug, -d : چاپ جزئیات اشکال‌زدایی.
- man --config-file=file, -C file : استفاده از فایل برای پیکربندی کاربر.
- man --warnings[=warnings] : فعال‌سازی هشدارها با groff.
- man --global-apropos, -K : جستجوی خشن در هر صفحه man.
- man --local-file, -l : نمایش فایل‌های راهنمای محلی.
- man --location, --path, --where, -w : نشان دادن مکان فایل nroff منبع.

مثال‌های دستور:

```
man -aw ls
```

نمایش تمام بخش‌های موجود برای دستور ls.

```
man -k search
```

نمایش لیستی از نتایج مرتبط با کلمه کلیدی "search".

```
man -f grep
```

نمایش توضیحات در مورد دستور grep.

```
man whereis tar :
```

تعیین مکان دستور tar.

```
man ls
```

نمایش کامل راهنمای دستور ls.

```
man 5 passwd
```

نمایش بخش مربوط به دستور passwd از بخش ۵.

```
man -w printf
```

نمایش مکان فایل man برای دستور printf.

```
man -I grep
```

اجرای دستور grep با حساسیت به بزرگی/کوچکی حروف.

```
man --default ls
```

بازنشانی رفتار دستور ls به حالت پیش‌فرض.

```
man --debug cat
```

چاپ جزئیات اشکال‌زدایی برای دستور cat.

۱۱.دستور mkdir

از دستور mkdirبرای ایجاد فهرست‌ها با تنظیمات مختلف و عملکردهای مختلف استفاده کنید.حالا بیایید با یک مثال ساده به شما نشان دهم که چگونه از دستور mkdirبه همراه این گزینه‌ها استفاده کنید:

۱. برای استفاده از گزینه mیا mode=MODE برای تنظیم حالت فایل (سطوح دسترسی) می‌توانید از دستور زیر استفاده کنید:

```
mkdir -m 755 my_directory
```

این دستور یک فهرست به نام my_directoryبا سطوح دسترسی ۷۵۵ ایجاد می‌کند.

۲. برای استفاده از گزینه – pیا –– parentsبرای ایجاد فهرست‌های والدین به صورت خودکار می‌توانید از دستور زیر استفاده کنید:

```
mkdir -p parent_dir/child_dir
```

این دستور parent_dirرا ایجاد می‌کند و در داخل آن فهرست child_dirرا ایجاد می‌کند، حتی اگر parent_dirوجود داشته باشد.

۳. برای استفاده از گزینه – vیا –– verboseبرای چاپ پیام برای هر فهرست ایجاد شده، می‌توانید از دستور زیر استفاده کنید:

```
mkdir -v dir1 dir2 dir3
```

این دستور سه فهرست به نام dir1، dir2و dir3 ایجاد می‌کند و برای هرکدام پیامی چاپ می‌کند.

۱۲.دستور mktemp

دستور mktemp در لینوکس برای ایجاد و نمایش یک فایل یا فهرست موقت با نام یکتا استفاده می‌شود. این دستور امکان ایجاد فایل‌ها یا فهرست‌های موقت با نام‌های منحصر به فرد و امن را فراهم می‌کند.

ساختار دستور:

```
mktemp [OPTION]... [TEMPLATE]
```

گزینه‌های دستور:

- d- یا --directory: ایجاد یک فهرست موقت به جای یک فایل.
- u- یا --dry-run: نمایش نام فایل یا فهرست موقت بدون ایجاد آن.
- p template- یا --tmpdir=template: تعیین مسیر موقت برای ایجاد فایل یا فهرست.

مثال‌های دستور:

ایجاد یک فایل موقت:

```
mktemp
```

ایجاد یک فهرست موقت:

```
mktemp -d
```

نمایش نام فایل موقت بدون ایجاد آن:

```
mktemp -u
```

تعیین مسیر موقت برای ایجاد فایل یا فهرست:

```
mktemp --tmpdir=/path/to/directory
```

استفاده از الگوی خاص برای ایجاد فایل یا فهرست موقت:

```
mktemp mytempfile_XXXXXX
```

۱۳.دستور rmdir

دستور rmdir به حذف / حذف فهرست های خالی به صورت زیر کمک می کند.

```
$ rmdir / backup / all
```

دستور rmdir در سیستم عامل‌های مبتنی بر یونیکس (UNIX) و لینوکس برای حذف یک فهرست خالی استفاده می‌شود. این دستور تنها قادر به حذف فهرست‌هایی است که خالی هستند و دارای هیچ فایل یا فهرست دیگری نیستند. برای استفاده از دستور rmdir، شما باید نام فهرست را به عنوان ورودی به آن بدهید.

مثال‌های دستور:

حذف یک فهرست خالی به نام directory:

```
rmdir directory
```

حذف چندین فهرست خالی با نام‌های dir1، dir2 و dir3:

```
rmdir dir1 dir2 dir3
```

حذف یک فهرست خالی و نمایش پیام تأیید قبل از حذف:

```
rmdir -i directory
```

حذف یک فهرست خالی و نمایش پیام‌های خطا در صورت بروز مشکل:

```
rmdir -v directory
```

حذف چندین فهرست خالی در یک دستور:

```
rmdir dir1 dir2 dir3
```

توجه داشته باشید که دستور rmdir تنها برای حذف فهرست‌های خالی قابل استفاده است. در صورتی که فهرست دارای فایل‌ها یا فهرست‌های دیگر باشد، از دستور rm -r استفاده کنید تا فهرست و تمامی محتویات آن را حذف کنید.

دستور rm

دستورrm برای حذف فایل ها یا فهرست ها به صورت زیر استفاده می شود.

```
$ rm file1
$ rm -rf my-directory
```

لطفا توجه داشته باشید که قبل از استفاده از این دستور، باید با دقت عمل کرده و از اطلاعات خود پشتیبانی تهیه کنید، زیرا حذف فایل‌ها و فهرست‌ها غیرقابل بازیابی است.

گزینه های دستور:

حذف یک فایل با نام file.txt:

```
rm file.txt
```

حذف چندین فایل با نام‌های file1.txt، file2.txt و file3.txt:

```
rm file1.txt file2.txt file3.txt
```

حذف یک فهرست به نام directory و تمامی محتویات آن:

```
rm -r directory
```

حذف فایل‌ها بدون نیاز به تأیید کاربر:

```
rm -f file.txt
```

حذف فهرست‌ها و فایل‌ها با نمایش پیام راهنما قبل از حذف هر فایل یا فهرست:

```
rm -i file.txt
```

حذف فهرست‌ها و فایل‌ها به صورت صامت (بدون نمایش پیام خطا):

```
rm -f -silent file.txt
```

حذف فهرست‌ها و فایل‌ها با استفاده از الگوهای خاص:

```
rm *.txt
```

این دستور تمام فایل‌های با پسوند . txt را در فهرست فعلی حذف می‌کند.

حذف یک فهرست و تمامی محتویات آن در یک مسیر خاص:

```
rm -r /path/to/directory
```

در این مثال، فهرست directory در مسیر / path/to/ و تمامی محتویاتش حذف می‌شود.

روز هشتم

۱۴.دستور scp

دستورscp را قادر می سازد به طور مثال فایل های ایمن را در بین hostها در یک شبکه کپی کنید. دراین دستور حرف s مخفف secure میباشد.

```
$ scp ~ / names.txt root@192.168.56.10 : /root/names.txt
```

دستور scpدر سیستم عامل‌های مبتنی بر یونیکس (UNIX) و لینوکس برای انتقال فایل‌ها و فهرست‌ها بین دو سیستم از طریق شبکه استفاده می‌شود. scp مخفف "Secure Copy" است و برای انتقال امن فایل‌ها از طریق پروتکل SSH (Secure Shell) استفاده می‌کند.

ساختار دستور:

```
scp [OPTIONS] SOURCE DESTINATION
```

در اینجا، SOURCEنشان‌دهنده مبدا فایل یا فهرست است و DESTINATIONنشان‌دهنده مقصد انتقال است.

برخی از استفاده‌های رایج scpعبارتند از:

انتقال یک فایل از سیستم محلی به یک سرور از طریقSSH :

```
scp file.txt user@server:/path/to/destination
```

انتقال یک فایل از یک سرور به سیستم محلی:

```
scp user@server:/path/to/file.txt /local/path/destination
```

انتقال یک فهرست از سیستم محلی به یک سرور:

```
scp -r directory user@server:/path/to/destination
```

انتقال یک فهرست از یک سرور به سیستم محلی:

```
scp -r user@server:/path/to/directory /local/path/destination
```

انتقال فایل‌ها و فهرست‌ها با استفاده از پورت خاص:

```
scp -P port file.txt user@server:/path/to/destination
```

انتقال چندین فایل با استفاده از نماد جایگزینی ویژه:

```
scp file*.txt user@server:/path/to/destination
```

انتقال فایل به دسته‌ای از سرورها:

```
scp file.txt user1@server1:/path/to/destination1 user2@server2:/path/to/destination2
```

این مثال‌ها تعدادی از رایج‌ترین استفاده‌ها از دستور scpرا نشان می‌دهند. با استفاده از این دستور و تنظیمات مختلف، می‌توانید فایل‌ها و فهرست‌ها را بین سیستم‌ها به صورت امن منتقل کنید.

۱۵.دستور rsync

rsync احتمالاً یکی از پرکاربردترین دستورات است. آنجا. برای کپی امن فایل ها از یک سرور به سرور دیگر از طریق SSH استفاده می شود. در مقایسه با دستور scp که کار مشابهی را انجام می‌دهد، rsync انتقال را بسیار سریع‌تر می‌کند و در صورت وقفه، می‌توانید فرآیند انتقال را بازیابی/ازسرگیری کنید.

ساختار دستور:

```
Local:  rsync [OPTION...] SRC... [DEST]
Access via remote shell:
  Pull: rsync [OPTION...] [USER@]HOST:SRC... [DEST]
  Push: rsync [OPTION...] SRC... [USER@]HOST:DEST
Access via rsync daemon:
  Pull: rsync [OPTION...] [USER@]HOST::SRC... [DEST]
        rsync [OPTION...] rsync://[USER@]HOST[:PORT]/SRC... [DEST]
  Push: rsync [OPTION...] SRC... [USER@]HOST::DEST
        rsync [OPTION...] SRC... rsync://[USER@]HOST[:PORT]/DEST
```

```
$ rsync -avz user@your-remote-server.com:/home/user/dir/
```

دستور بالا تمام فایل ها و فهرست ها را از پوشه فعلی سرور شما به سرور راه دور کپی می‌کند.

گزینه های دستور:

- a– به طور پیش فرض، تمام مجوزها، مالکیت، زمان‌های دسترسی و پیوندها حفظ می‌شوند.
- b– از الگوریتم انتقال دلتا برای بهینه‌سازی فرآیند همگام‌سازی استفاده می‌کند.
- c– فقط در صورت بروز تغییر در فایل‌ها، آنها را همگام‌سازی می‌کند.
- e– از یک برنامه خاص برای انتقال داده‌ها استفاده می‌کند.
- f– از حذف فایل‌های اضافی در مقصد جلوگیری می‌کند.
- h– راهنمای دستور را نمایش می‌دهد.
- i– قبل از هر گونه تغییر، از کاربر تأیید می‌گیرد.
- n– فقط فرآیند همگام‌سازی را شبیه‌سازی می‌کند، بدون اینکه هیچ تغییری ایجاد کند.
- p– مجوزها، مالکیت و زمان‌های دسترسی را حفظ می‌کند.
- q– خروجی دستور را محدود می‌کند.
- r– دایرکتوری‌ها را به طور بازگشتی همگام‌سازی می‌کند.
- t– فقط فایل‌های متنی را همگام‌سازی می‌کند.
- u– فقط فایل‌هایی را که در مقصد جدیدتر هستند، همگام‌سازی می‌کند.
- v– جزئیات مربوط به فرآیند همگام‌سازی را نمایش می‌دهد.
- z– از الگوریتم فشرده‌سازی برای بهینه‌سازی فرآیند همگام‌سازی استفاده می‌کند.

اگر سرویس SSH روی سرور راه دور روی پورت استاندارد ۲۲ اجرا نمی شود ، می توانید از rsync با یک پورت SSH ویژه استفاده کنید:

```
$ rsync -avz user@your-remote server.com:/home/user/dir/
```

مثال های دستور:

مثال‌هایی برای گزینه‌های دستور rsync:

–a:

```
rsync -a /home/user/Documents /backup/Documents
```

این دستور تمام فایل‌ها و دایرکتوری‌های موجود در دایرکتوری Documents را به دایرکتوری backup/Documents همگام‌سازی می‌کند.

-b:

```
rsync -b /home/user/Desktop /remote/Desktop
```

این دستور فقط فایل‌هایی را که در دایرکتوری Desktop تغییر کرده‌اند، به دایرکتوری remote/Desktop همگام‌سازی می‌کند.

-c:

```
rsync -c /home/user/Pictures /archive/Pictures
```

این دستور فقط در صورتی که فایل‌ها در دایرکتوری Pictures جدیدتر باشند، آنها را به دایرکتوری archive/Pictures همگام‌سازی می‌کند.

-e:

```
rsync -e ssh user@remotehost:/home/user/Music /local/Music
```

این دستور از طریق SSH برای همگام‌سازی فایل‌ها از دایرکتوری /home/user/Music در سیستم راه دور با نام remotehost به دایرکتوری /local/Music در سیستم محلی استفاده می‌کند.

-f:

```
rsync -f /home/user/Videos /media/Videos
```

این دستور تمام فایل‌ها و دایرکتوری‌های موجود در دایرکتوری Videos را به دایرکتوری media/Videos همگام‌سازی می‌کند، حتی اگر فایل‌هایی در دایرکتوری مقصد وجود داشته باشند که در دایرکتوری منبع وجود ندارند.

-h:

```
rsync -h
```

این دستور راهنمای دستور rsync را نمایش می‌دهد.

-i:

```
rsync -i /etc/passwd /etc/passwd.bak
```

قبل از همگام‌سازی فایل /etc/passwd با /etc/passwd.bak، از کاربر تأیید می‌گیرد.

-n:

```
rsync -n /home/user/Downloads /cloud/Downloads
```

فرآیند همگام‌سازی را شبیه‌سازی می‌کند، بدون اینکه هیچ تغییری ایجاد کند.

-p:

```
rsync -p /home/user/Projects /external/Projects
```

مجوزها، مالکیت و زمان‌های دسترسی را در حین همگام‌سازی فایل‌ها از دایرکتوری /home/user/Projects به /external/Projects حفظ می‌کند.

-q:

```
rsync -q /var/log /backup/log
```

خروجی دستور را فقط به اطلاعات ضروری محدود می‌کند.

-r:

```
rsync -r /home/user/Website /public_html
```

دایرکتوری Website را به طور بازگشتی به دایرکتوری public_html همگام‌سازی می‌کند.

–t:

```
rsync -t /home/user/Documents/text /remote/Documents/text
```

فقط فایل‌های متنی را در دایرکتوری Documents/text به دایرکتوری remote/Documents/text همگام‌سازی می‌کند.

–u:

```
rsync -u /home/user/Music /mp3/Music
```

فقط فایل‌هایی را که در دایرکتوری Music جدیدتر از فایل‌های موجود در دایرکتوری mp3/Music هستند، همگام‌سازی می‌کند.

–v:

```
rsync -v /home/user/Pictures /gallery/Pictures
```

جزئیات مربوط به فرآیند همگام‌سازی فایل‌ها از دایرکتوری /home/user/Pictures به /gallery/Pictures را نمایش می‌دهد.

–z:

```
rsync -z /home/user/Videos /archive/Videos
```

از الگوریتم فشرده‌سازی برای بهینه‌سازی فرآیند همگام‌سازی فایل‌ها از دایرکتوری /home/user/Videos به /archive/Videos استفاده می‌کند.

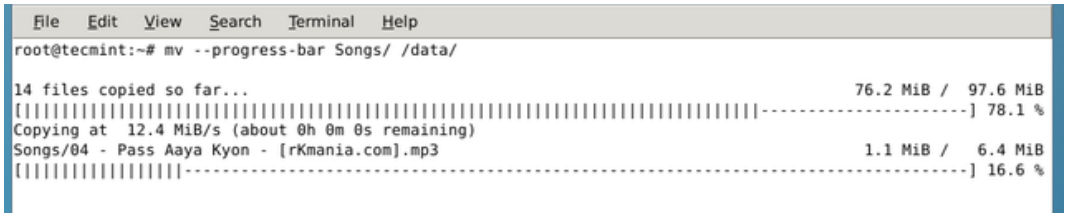
۱۶.دستور mv

دستور mv مخفف move برای انتقال فایل به مکان دیگر است. در اینجا نمونه ای از دستور mvبا صفحه نمایش است.

```
# mv --progress-bar /mp3/* /data/
```

یا

```
# mv –g /mp3 /data /
```



روز نهم

۱۷.دستور file

دستور file در سیستم‌های Unix/Linux استفاده می‌شود تا نوع یا محتوای یک فایل را تشخیص دهد. با اجرای دستور file بر روی یک فایل، سیستم تلاش می‌کند تا نوع فایل (مثلاً متنی، تصویر، فایل اجرایی و غیره) را شناسایی کند و اطلاعات مربوط به آن را نمایش دهد.

ساختار دستور:

```
file [نام_فایل]
```

گزینه های دستور:

- b– نام فایل را از خروجی حذف می‌کند.
- c– نوع فایل را به صورت خلاصه نمایش می‌دهد.
- e– اطلاعات مربوط به رمزگذاری فایل را نمایش می‌دهد.
- f– فایلی را که می‌خواهید نوع آن را تشخیص دهید، مشخص می‌کند.
- h– راهنمای دستور را نمایش می‌دهد.
- i– اطلاعات مربوط به نوع MIME فایل را نمایش می‌دهد.
- m– اطلاعات مربوط به جادوی فایل را نمایش می‌دهد.
- n– تعداد خطوطی که باید از ابتدای فایل خوانده شوند را مشخص می‌کند.
- p– اطلاعات مربوط به نوع فایل را به صورت تفصیلی نمایش می‌دهد.
- s– اندازه فایل را نمایش می‌دهد.
- v– جزئیات مربوط به فرآیند تشخیص نوع فایل را نمایش می‌دهد.
- z– اطلاعات مربوط به فشرده‌سازی فایل را نمایش می‌دهد.

مثال های دستور:

۱. **file filename:** این دستور برای نمایش نوع فایل مورد نظر استفاده می‌شود.

```
file example.txt
```

۲. **file -b filename:** این دستور برای نمایش نوع فایل مورد نظر بدون اطلاعات اضافی استفاده می‌شود.

```
file -b example.txt
```

۳. **file -i filename:** این دستور برای نمایش نوع MIME (Multipurpose Internet Mail Extensions) فایل مورد نظر استفاده می‌شود.

```
file -i example.txt
```

۴. **file -z filename:** این دستور برای نمایش نوع فایل مورد نظر به همراه اطلاعات فشرده‌سازی استفاده می‌شود.

```
file -z example.zip
```

۵. **file -L filename:** این دستور برای نمایش نوع فایل مورد نظر به همراه لینک‌های سمبولیک استفاده می‌شود.

```
file -L example_symlink
```

مثال های بیشتر:

تشخیص نوع فایل بدون نام:

```
file file.txt
```

خروجی:

```
file.txt: ASCII text
```

تشخیص نوع فایل به صورت خلاصه:

```
file -c file.txt
```

خروجی:

text

نمایش اطلاعات مربوط به رمزگذاری فایل:

```
file -e file.txt
```

خروجی:

```
file.txt: UTF-8 Unicode text
```

تشخیص نوع فایل با نام:

```
file -f /path/to/file.txt
```

خروجی:

/path/to/file.txt: ASCII text

نمایش راهنمای دستور:

```
file -h
```

نمایش اطلاعات مربوط به نوع MIME فایل:

```
file -i file.txt
```

خروجی:

```
file.txt: text/plain
```

نمایش اطلاعات مربوط به جادوی فایل:

```
file -m file.txt
```

خروجی:

۰ ۲۰ ۷۳ ۶۹ ۲۰ ۷۳ ۶۹ ۶۸ ۵۴ :۰۰۰۰۰۰۰۰a 00 00 00 00 00 00 00

۰۰ ۰۰ ۰۰ ۰۰ ۰۰ ۰۰ ۰۰ ۰۰ ۰۰ ۰۰ ۰۰ ۰۰ ۰۰ ۰۰ ۰۰ ۰۰ :۰۰۰۰۰۰۱۰

نمایش ۱۰ خط اول فایل:

```
file -n 10 file.txt
```

خروجی:

This is a text file.
It has multiple lines.

نمایش اطلاعات مربوط به نوع فایل به صورت تفصیلی:

```
file -p file.txt
```

خروجی:

file.txt: ASCII text, with CRLF line terminators

نمایش اندازه فایل:

```
file -s file.txt
```

خروجی:

۱۲۳bytes

نمایش جزئیات مربوط به فرآیند تشخیص نوع فایل:

```
file -v file.txt
```

خروجی:

```
file: trying to detect file type...
file: from `file.txt':
file: trying to find magic...
file: found 1 magic matches:
file: `file.txt': ASCII text
```

نمایش اطلاعات مربوط به فشرده‌سازی فایل:

```
file -z file.txt
```

خروجی:

```
file.txt: uncompressed
```

این مثال‌ها تعدادی از استفاده‌های معمول دستور fileدر لینوکس هستند. این دستورات همراه با گزینه‌های مختلف می‌توانند به شما کمک کنند تا اطلاعات مفیدی درباره فایل‌های مختلف در سیستم خود دریافت کنید.

۱۸. دستور cksum

cksum فرمانی در سیستم‌عاملهای شبه یونیکس است جهت تولید سرجمع یا چکسام برای فایلها یا جریان‌های اطلاعاتی (مانند ورودی و خروجی‌های استاندارد)است. فرمان Cksum فایل یا فایلهایی را بعنوان آرگومان دریافت می‌کند و از آنها یک چک‌سام، کد افزونگی چرخشی (CRC) و تعداد بایتهای موجود را محاسبه می‌کند. در خروجی این فرمان، چک‌سام، تعداد بایتها و در انتها نام فایل نوشته خواهد شد. فرمان cksum می‌تواند برای بررسی کردن برابری دو فایل در کامپیوترهای مختلف استفاده شود. مثلاً زمانی که بخواهیم دو فایل را که توسط یک مسیر انتقال پرنویز کپی شده‌اند با هم مقایسه کنیم و مطمئن شویم که کاملاً مشابه همدیگرند و عملیات کپی به‌درستی انجام شده است.

ساختار دستور:

```
$ cksum   FILE]]
$ cksum   [OPTION]
```

مجموع بررسی CRC و تعداد بایت هر فایل را چاپ میکند. دستور "cksum" برای محاسبه و بررسی مقادیر کنترلی (checksum) فایل‌ها استفاده می‌شود. در زیر چند مثال برای استفاده از گزینه‌های مختلف این دستور آورده شده است:

انتخاب نوع الگوریتم مقدار کنترلی(digest) :

```
$cksum -a sha256 file.txt
```

تولید مقدار کنترلی به صورت base64 به جای هگزادسیمال:

```
$cksum --base64 file.txt
```

بررسی مقادیر کنترلی از فایل‌ها:

```
$cksum -c checksums.txt file1.txt file2.txt
```

تولید مقدار کنترلی به طول مشخص (به بیت):

```
$cksum -l 256 file.txt
```

تولید مقدار کنترلی به صورت دودویی (به جای هگزادسیمال):

```
$cksum --raw file.txt
```

تولید مقدار کنترلی به سبک BSD پیش‌فرض

```
$cksum --tag file.txt
```

تولید مقدار کنترلی به سبک معکوس بدون نوعdigest :

```
$cksum --untagged file.txt
```

پایان هر خط خروجی را با NUL نهnewline و غیرفعال کردن escaping نام فایل:

```
$cksum -z file.txt
```

علاوه بر این، برای بررسی مقادیر کنترلی، می‌توانید از گزینه‌های زیر استفاده کنید:

عدم خطا یا گزارش برای فایل‌هایی که موجود نیستند:

```
$cksum --ignore-missing checksums.txt
```

عدم چاپ OK برای هر فایل که با موفقیت بررسی شده است:

```
$cksum --quiet checksums.txt
```

عدم نمایش هیچ مقداری و استفاده از کد وضعیت برای نشان دادن موفقیت:

```
$cksum --status checksums.txt
```

خروجی غیرصحیح برای خطوط مقدار کنترلی نادرست:

```
$cksum --strict checksums.txt
```

هشدار در مورد خطوط مقدار کنترلی به صورت نادرست:

```
$cksum --warn checksums.txt
```

نشان دادن اینکه کدام پیاده‌سازی استفاده شده است:

```
$cksum --debug file.txt
```

نمایش راهنما و خروج:

```
$cksum --help
```

نمایش اطلاعات نسخه و خروج:

```
$cksum --version
```

لطفاً توجه داشته باشید که مسیر و نام فایل‌ها و checksums.txt در مثال‌ها باید با موارد واقعی جایگزین شوند. همچنین، مقادیر کنترلی و خروجی‌ها برای هر مثال ممکن است متفاوت باشند و بسته به الگوریتم و ورودی‌های استفاده شده، متفاوت باشند.

روز دهم

۱۹.دستور md5sum

md5sum یک برنامه کامپیوتری است که هش های MD5 128 بیتی را محاسبه و تأیید می کند، همانطور که در RFC 1321 توضیح داده شده است. هش MD5 به عنوان اثر انگشت دیجیتال فشرده یک فایل عمل می کند. مانند همه این الگوریتم‌های هش‌سازی، از نظر تئوری تعداد نامحدودی از فایل‌ها وجود دارد که دارای هر هش MD5 هستند. با این حال، بسیار بعید است که هر دو فایل غیر یکسان در دنیای واقعی دارای هش MD5 یکسان باشند، مگر اینکه به طور خاص برای این کار ایجاد شده باشند. الگوریتم MD5 زیربنایی دیگر امن تلقی نمی شود. بنابراین، در حالی که md5sum برای شناسایی فایل‌های شناخته شده در موقعیت‌هایی که مربوط به امنیت نیستند، مناسب است، اگر احتمال دستکاری عمدی و مخرب فایل‌ها وجود داشته باشد، نباید به آن اعتماد کرد. در مورد دوم، استفاده از ابزار هش جدیدتر مانند sha256sum توصیه می شود. معمولاً یافتن دو فایل مجزا که منجر به رشته های یکسان می شوند بسیار دشوار است. بنابراین، می‌توانید از md5sum برای بررسی یکپارچگی داده‌های دیجیتال با تعیین اینکه فایل یا ISOای که دانلود کردهاید یک کپی بیت به بیت از فایل راه دور یا ISO استفاده کنید.

دستورزیر یک مقدار هش برای فایل به شرح زیر تولید:

```
$ md5sum group.csv

bc527343c7ffc103111f3a694b004e2f   group.csv
```

هنگامی که سعی می کنید محتوای فایل را با حذف خط اول تغییر دهید `root:x:0`و سپس دستور را برای بار دوم اجرا کنید، سعی کنید مقدار هش را ببینند:

```
$ md5sum group.csv

46798b5cfca45c46a84b7419f8b74735   group.csv
```

متوجه خواهید شد که مقدار هش اکنون تغییر کرده است، که نشان می دهد محتویات فایل تغییر کرده است.

حالا خط اول فایل را برگردانید `root:x:0`و نام آن را به **group_file.txt** تغییر دهید و دستور زیر را اجرا کنید تا مقدار هش آن دوباره ایجاد شود:

```
$ md5sum group_list.txt

bc527343c7ffc103111f3a694b004e2f   group_list.txt
```

از خروجی بالا، حتی زمانی که فایل با محتوای اصلی آن تغییر نام داده است، مقدار هش همچنان یکسان است.

md5 sums : فقط با محتوای فایل به جای نام فایل تایید/کار می کند.

فایل group_list.txtتکراری از group.csvاست ، بنابراین سعی کنید مقدار هش فایل ها را همزمان به صورت زیر تولید کنید. خواهید دید که هر دو دارای مقادیر هش مساوی هستند، زیرا محتوای آنها دقیقاً یکسان است.

```
$ md5sum group_list.txt groups.csv

bc527343c7ffc103111f3a694b004e2f   group_list.txt
bc527343c7ffc103111f3a694b004e2f   group.csv
```

می توانید مقدار هش یک فایل(ها) را به یک فایل متنی هدایت کنید و ذخیره کنید، آنها را با دیگران به اشتراک بگذارید. برای دو فایل بالا، می توانید دستور زیر را صادر کنید تا مقادیر هش تولید شده را به یک فایل متنی برای استفاده بعدی هدایت کنید:

```
$ md5sum groups_list.txt groups.csv > myfiles.md5
```

برای بررسی اینکه فایل‌ها از زمانی که checksum را ایجاد کرده‌اید تغییرنکرده‌اند، دستور بعدی را اجرا کنید. شما باید بتوانید نام هر فایل را همراه با "OK"مشاهده کنید .

-cیا --checkگزینه مقادیر MD5 فایل ها را بررسی کنید.

```
$ md5sum -c myfiles.md5
```

به یاد داشته باشید که پس از ایجاد چک‌سوم، نمی‌توانید نام فایل‌ها را تغییر دهید وگرنه وقتی می‌خواهید فایل‌ها را با نام‌های جدید تأیید کنید، با خطای فایل یا فهرستی وجود ندارد مواجه می‌شوید.

برای مثال:

```
$ mv group_list.txt new.txt
$ mv group.csv file.txt
$ md5sum -c myfiles.md5
```

این مفهوم همچنین برای رشته ها به طور یکسان کار می کند، در دستورات زیر، -nبه این معنی است که خط جدید انتهایی را خروجی نکنید:

```
$ echo -n "Predator-os  How-Tos" | md5sum -

afc7cb02baab440a6e64de1a5b0d0f1b -
$ echo -n "Predator-os  How-To" | md5sum -

65136cb527bff5ed8615bd1959b0a248 -
```

۲۰.دستور sha256sum

sha256sumیک دستور در لینوکس است که برای محاسبه مقدار SHA-256 یک فایل استفاده می‌شود. به عنوان مثال، فرض کنید می‌خواهید فایلی به نام example.txt را با استفاده از sha256sum بررسی کنید و مشخص شود آیا مقدار SHA-256 آن با یک مقدار خاص مطابقت دارد یا خیر. برای این منظور، ابتدا یک مقدار SHA-256 معتبر برای فایل example.txt تولید می‌کنیم. سپس با استفاده از گزینه -c یا --check وارد مرحله بررسی مطابقت می‌شویم.

۱. تولید مقدار SHA-256 برای فایل:

```
  echo -n "Hello, World!" | sha256sum
```

خروجی دستور:

4c4fcd2f5b8e4f1b8f7f8056b9c6a1e0cb5f55e7a5a5e8c3df3b7d1f3b8c3a5d

۲. بررسی مطابقت فایل با مقدار SHA-256:

```
 sha256sum -c <<< "4c4fcd2f5b8e4f1b8f7f8056b9c6a1e0cb5f55e7a5a5e8c3df3b7d1f3b8c3a5d
example.txt"
```

example.txt: OK

۲۱.دستور sha512sum

به عنوان مثال، فرض کنید می‌خواهید فایلی به نام example.txt را با استفاده از sha512sum بررسی کنید و مشخص شود آیا مقدار SHA-512 آن با یک مقدار خاص مطابقت دارد یا خیر. برای این منظور، ابتدا یک مقدار SHA-512 معتبر برای فایل example.txt تولید می‌کنیم. سپس با استفاده از گزینه c– یا check–– وارد مرحله بررسی مطابقت می‌شویم.

تولید مقدار SHA-512 برای فایل:

```
echo -n “Hello, World!” | sha512sum
```

خروجی دستور:

۲ef7bde608ce5404e97d5f042f95f89f1c232871a467dd5f9178568e9d9ec4c7c7d1e3f3a759b3e4d45c
16d3edba7f3552dd9a78e6f7a739b6ac3b3f146c29b5 -

بررسی مطابقت فایل با مقدار SHA-512:

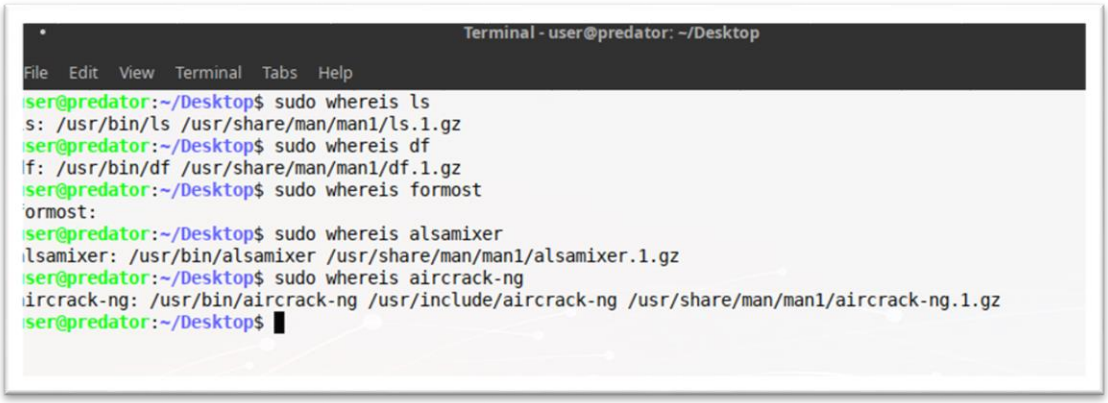
```
sha512sum -c
<<<“2ef7bde608ce5404e97d5f042f95f89f1c232871a467dd5f9178568e9d9ec4c7c7d1e3f3a759b3e4d45c1
6d3edba7f3552dd9a78e6f7a739b6ac3b3f146c29b5  example.txt”
```

خروجی دستور:

example.txt: OK

مشاهده محل اجرا دستورات

دستور whereis فایل های باینری، منبع و فایل های صفحه دستی را برای نام دستور به صورت زیر ارائه می کند:



bash دارای دو ویژگی بزرگ است که به کمک می کند دستورات کامل و دوباره اجرا کنید، به سادگی قسمت اول یک دستور را تایپ کنید، کلید <tab> را بزنید و ترمینال حدس میزند چه کاری انجام میخواهید انجام دهید.

```
ls <ENTER>
apro  <TAB>
```

با دستور tab دستور کامل می شود.

روز یازدهم

۲۲.مدیریت بسته ها

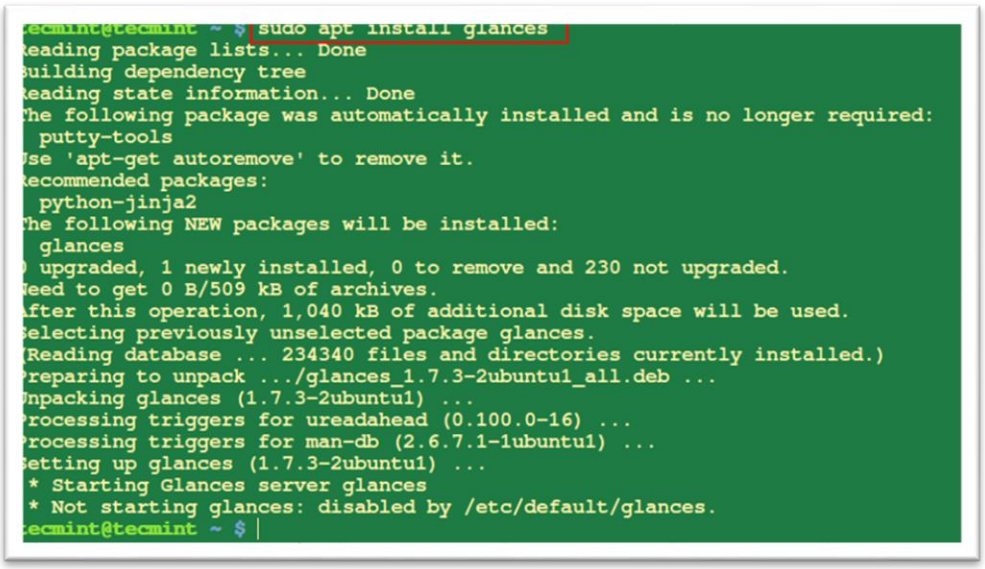
یکی از مهم ترین مواردی است که تحت system linux / سرور اداره می شود مدیریت بسته با استفاده از ابزارهای مختلف مدیریت بسته است. توزیع های مختلف linux برنامه های کاربردی را در یک بسته پیش ساخته شده که فایل های باینری، فایل های پیکربندی و همچنین اطلاعات مربوط به وابستگی های برنامه را شامل می شوند، نصب می کند. ابزار مدیریت بسته به مدیران سرور به روش های مختلف از جمله:

۱. دانلود و نصب نرم افزار
۲. نرم افزار را از مخزن استاندارد نصب میکند
۳. بروز اوری تمام نرم افزار نصب شده
۴. نگه داشتن اطلاعات در مورد نرم افزار نصب شده

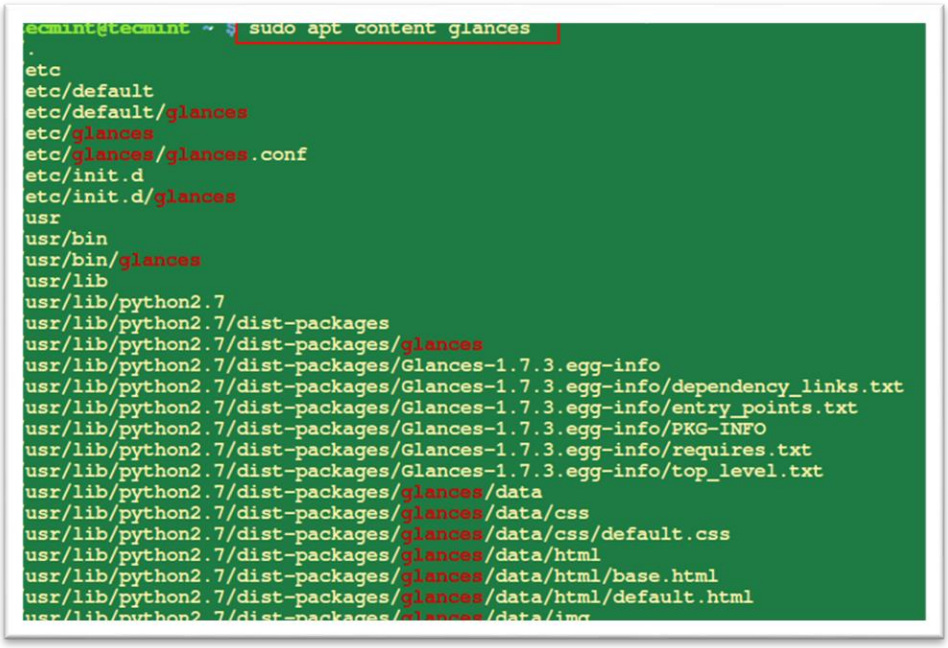
مدیربسته apt

می توانید یک بسته را به صورت زیر تنظیم ونصب کنید، با نام یک بسته واحد را مشخص کنید یا بسته های زیادی را با یک لیست می توانید نصب کنید.

```
$ sudo apt install glances
```



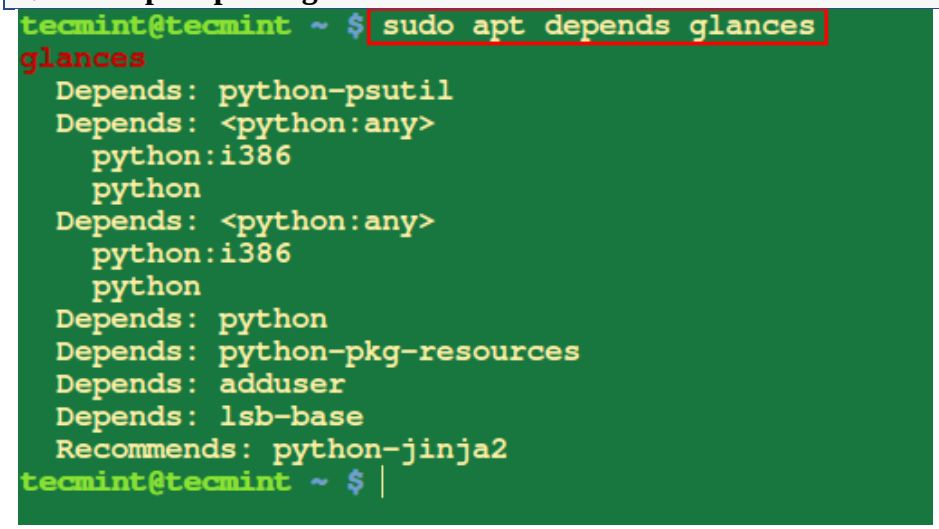
دستور زیرکمک خواهد کرد که لیستی از تمام فایل هایی که در بسته ای به نام glancesوجود دارد .



پیدا کردن فایل های نصب شده با apt

دستور زیرکمک می کند تا اطلاعات درباره وابستگی های خاص بسته ای که مشخص می کنید نمایش داده شود.

```
$ sudo apt depends glances
```



جستجو برای بسته با apt

گزینه جستجو برای نام بسته داده شده جستجو می کند و تمام بسته های متناسب را نشان می دهد.

```
$ sudo apt search apache2
```

```
tecmint@tecmint ~ $ sudo apt search apache2
ih   apache2                          - Apache HTTP Server
p    apache2:i386                     - Apache HTTP Server
v    apache2-api-20120211             -
v    apache2-api-20120211:i386        -
i A  apache2-bin                      - Apache HTTP Server (binary files and modul
p    apache2-bin:i386                 - Apache HTTP Server (binary files and modul
i A  apache2-data                     - Apache HTTP Server (common files)
v    apache2-data:i386                -
p    apache2-dbg                      - Apache debugging symbols
p    apache2-dbg:i386                 - Apache debugging symbols
p    apache2-dev                      - Apache HTTP Server (development headers)
p    apache2-dev:i386                 - Apache HTTP Server (development headers)
p    apache2-doc                      - Apache HTTP Server (on-site documentation)
p    apache2-mpm-event                - transitional event MPM package for apache2
p    apache2-mpm-event:i386           - transitional event MPM package for apache2
p    apache2-mpm-itk                  - transitional itk MPM package for apache2
p    apache2-mpm-itk:i386             - transitional itk MPM package for apache2
p    apache2-mpm-prefork              - transitional prefork MPM package for apach
p    apache2-mpm-prefork:i386         - transitional prefork MPM package for apach
p    apache2-mpm-worker               - transitional worker MPM package for apache
p    apache2-mpm-worker:i386          - transitional worker MPM package for apache
v    apache2-prefork-dev              -
v    apache2-prefork-dev:i386         -
p    apache2-suexec                   - transitional package for apache2-suexec-pr
p    apache2-suexec:i386              - transitional package for apache2-suexec-pr
p    apache2-suexec-custom            - Apache HTTP Server configurable suexec pro
```

مشاهده اطلاعات مربوط به بسته با apt

این کمک می کند که اطلاعات مربوط به بسته یا بسته ها را نمایش دهید، دستور زیر را با مشخص کردن تمام بسته هایی که می خواهید اطلاعات را نمایش دهید، اجرا کنید.

```
$ sudo apt show firefox
```

```
tecmint@tecmint ~ $ sudo apt show firefox
Package: firefox
State: installed
Automatically installed: no
Version: 43.0+linuxmint1+rosa
Priority: optional
Section: web
Maintainer: Ubuntu Mozilla Team <ubuntu-mozillateam@lists.ubuntu.com>
Architecture: amd64
Uncompressed Size: 106 M
Depends: lsb-release, libasound2 (>= 1.0.16), libatk1.0-0 (>= 1.12.4), libc6 (>=
         2.17), libcairo2 (>= 1.2.4), libdbus-1-3 (>= 1.0.2), libdbus-glib-1-2
         (>= 0.78), libfontconfig1 (>= 2.9.0), libfreetype6 (>= 2.2.1), libgcc1
         (>= 1:4.1.1), libgdk-pixbuf2.0-0 (>= 2.22.0), libglib2.0-0 (>= 2.31.8),
         libgtk2.0-0 (>= 2.24.0), libpango-1.0-0 (>= 1.22.0),
         libpangocairo-1.0-0 (>= 1.14.0), libstartup-notification0 (>= 0.8),
         libstdc++6 (>= 4.6), libx11-6, libxcomposite1 (>= 1:0.3-1), libxdamage1
         (>= 1:1.1), libxext6, libxfixes3, libxrender1, libxt6
Recommends: xul-ext-ubufox, libcanberra0, libdbusmenu-glib4, libdbusmenu-gtk4
Suggests: ttf-lyx
Conflicts: firefox
Replaces: kubuntu-firefox-installer, kubuntu-firefox-installer
Provides: gnome-www-browser, iceweasel, www-browser
Description: Safe and easy web browser from Mozilla
 Firefox delivers safe, easy web browsing. A familiar user interface, enhanced
 security features including protection from online identity theft, and
 integrated search let you get the most out of the web.
```

تأیید بسته برای هر وابستگی ناقص با apt

گاهی اوقات درطول نصب بسته، ممکن است اشتباهات مربوط به وابستگی بسته را ایجاد شودبرای ایجاد وابستگی[1] های یک بسته دستور زیر را واردنماید.

[1] dependency

```
tecmint@tecmint ~ $ sudo apt check firefox
Reading package lists... Done
Building dependency tree
Reading state information... Done
tecmint@tecmint ~ $
```

نمایش بسته های از دست رفته با apt

```
$ sudo apt recommends apache2
```

```
tecmint@tecmint ~ $ sudo apt recommends apache2

No missing recommended packages were found for apache2
```

نمایش نسخه بسته نصب شده با apt

```
$ sudo apt version firefox
```

```
tecmint@tecmint ~ $ sudo apt version firefox
43.0+linuxmint1+rosa
tecmint@tecmint ~ $ sudo apt version apache2
2.4.7-1ubuntu4.8
tecmint@tecmint ~ $ sudo apt version perl
5.18.2-2ubuntu1
tecmint@tecmint ~ $
```

به روز رسانی بسته های سیستم با apt

این کمک می کند تا لیستی از بسته های مخازن مختلف موجود در system خود را بارگیری کنید و آنها را به روز رسانی کنید، زمانی که نسخه های جدید بسته ها و وابستگی های آنها وجود دارد.

```
$ sudo apt update
```

```
tecmint@tecmint ~ $ sudo apt update
Hit http://download.virtualbox.org trusty InRelease
Hit http://download.virtualbox.org trusty/contrib amd64 Packages
Hit http://download.virtualbox.org trusty/contrib i386 Packages
Ign http://download.virtualbox.org trusty/contrib Translation-en_IN
Ign http://download.virtualbox.org trusty/contrib Translation-en
Ign http://dl.google.com stable InRelease
Hit http://dl.google.com stable Release.gpg
Hit http://dl.google.com stable Release
Hit http://dl.google.com stable/main amd64 Packages
Get:1 http://security.ubuntu.com trusty-security InRelease [65.9 kB]
Ign http://archive.ubuntu.com trusty InRelease
Hit http://ppa.launchpad.net trusty InRelease
Ign http://packages.linuxmint.com rosa InRelease
Ign http://extra.linuxmint.com rosa InRelease
Get:2 http://archive.ubuntu.com trusty-updates InRelease [65.9 kB]
Hit http://ppa.launchpad.net trusty InRelease
Hit http://packages.linuxmint.com rosa Release.gpg
Hit http://extra.linuxmint.com rosa Release.gpg
Hit http://ppa.launchpad.net trusty InRelease
Get:3 http://security.ubuntu.com trusty-security/main amd64 Packages [454 kB]
```

ارتقا دادن سیستم با apt

این کمک می کند تا نسخه های جدیدی از تمام بسته های موجود در system خود را بروزرسانی کنید.

```
$ sudo apt upgrade
```

```
tecmint@tecmint ~ $ sudo apt upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following package was automatically installed and is no longer required:
  libvncclient0
Use 'apt-get autoremove' to remove it.
The following packages have been kept back:
  apache2 apache2-bin apache2-data libdrm-dev libdrm-intel1 libdrm-intel1:i386
  libdrm-nouveau2 libdrm-nouveau2:i386 libdrm-radeon1 libdrm-radeon1:i386
  libdrm2 libdrm2:i386
The following packages will be upgraded:
  apache2-utils apt-transport-https apt-utils atril atril-common base-files
  bind9-host ca-certificates caja caja-common coreutils cpio cpp-4.8 curl
  dnsutils ecryptfs-utils eom eom-common ffmpeg firefox firefox-locale-en
  flashplugin-installer g++-4.8 gcc-4.8 gcc-4.8-base gcc-4.8-base:i386
  gcc-4.9-base gcc-4.9-base:i386 gir1.2-caja gir1.2-gtk-2.0 gir1.2-ibus-1.0
  gir1.2-javascriptcoregtk-3.0 gir1.2-mate-panel gir1.2-webkit-3.0 git-man
  glib-networking glib-networking:i386 glib-networking-common
  glib-networking-services google-chrome-stable gtk2-engines-pixbuf
  gtk2-engines-pixbuf:i386 hexchat hexchat-common ibus-gtk:i386 ifupdown
```

حذف بسته نصب شده با apt

هنگام نصب یک بسته جدید در system خود، وابستگی ها نیز نصب شده و از برخی از کتابخانه های system با بسته های دیگر استفاده می کنند.پس از حذف این بسته خاص، وابستگی ها در system باقی خواهند ماند، بنابراین برای حذف آنها از autoremove به صورت زیر استفاده می شود:

```
$ sudo apt autoremove
```

```
tecmint@tecmint ~ $ sudo apt autoremove
Reading package lists... Done
Building dependency tree
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 229 not upgraded.
tecmint@tecmint ~ $
```

پاک کردن مخزن قدیمی بسته های دانلود شده با apt

گزینه 'clean' یا 'autoclean' تمام مخزن محلی قدیمی فایل های بسته دانلود شده را حذف می کند.

```
$ sudo autoclean
```

یا

```
$ sudo apt clean
```

```
tecmint@tecmint ~ $ sudo apt autoclean
Reading package lists... Done
Building dependency tree
Reading state information... Done
tecmint@tecmint ~ $ sudo apt clean
tecmint@tecmint ~ $
```

حذف بسته ها با فایل های پیکربندی آن با apt

برای حذف کامل بسته ها و وابستگی ها ان از گزینه purge استفاده میکنیم.

```
tecmint@tecmint ~ $ sudo apt purge glances
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  glances*
0 upgraded, 0 newly installed, 1 to remove and 229 not upgraded.
After this operation, 1,040 kB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 234394 files and directories currently installed.)
Removing glances (1.7.3-2ubuntu1) ...
 * Stopping Glances server glances
 * PID file not found
Purging configuration files for glances (1.7.3-2ubuntu1) ...
Processing triggers for man-db (2.6.7.1-1ubuntu1) ...
tecmint@tecmint ~ $
```

نصب بسته ها با پسوند .deb با apt

برای نصب یک فایل .deb دستور زیر را با نام فایل به عنوان یک توضیح به صورت زیر اجرا کنید:

```
$ sudo apt deb atom-amd64.deb
```

```
tecmint@tecmint ~ $ sudo apt deb atom-amd64.deb
Selecting previously unselected package atom.
(Reading database ... 234336 files and directories currently installed.)
Preparing to unpack atom-amd64.deb ...
Unpacking atom (1.6.2) ...
Setting up atom (1.6.2) ...
Processing triggers for mime-support (3.54ubuntu1.1) ...
Processing triggers for desktop-file-utils (0.22-1ubuntu1) ...
tecmint@tecmint ~ $
```

راهنمای دستورapt

دستور زیر تمام گزینه های را با شرح آن در مورد **نحوه** استفاده از APTدر system لیست می کند.

```
tecmint@tecmint ~ $ apt help
apt
Usage: apt command [options]
       apt help command [options]

Commands:
add-repository  - Add entries to apt sources.list
autoclean       - Erase old downloaded archive files
autoremove      - Remove automatically all unused packages
build           - Build binary or source packages from sources
build-dep       - Configure build-dependencies for source packages
changelog       - View a package's changelog
check           - Verify that there are no broken dependencies
clean           - Erase downloaded archive files
contains        - List packages containing a file
content         - List files contained in a package
deb             - Install a .deb package
depends         - Show raw dependency information for a package
dist-upgrade    - Perform an upgrade, possibly installing and removing packages
download        - Download the .deb file for a package
dselect-upgrade - Follow dselect selections
held            - List all held packages
help            - Show help for a command
hold            - Hold a package
install         - Install/upgrade packages
policy          - Show policy settings
purge           - Remove packages and their configuration files
recommends      - List missing recommended packages for a particular package
rdepends        - Show reverse dependency information for a package
reinstall       - Download and (possibly) reinstall a currently installed package
remove          - Remove packages
search          - Search for a package by name and/or expression
show            - Display detailed information about a package
source          - Download source archives
sources         - Edit /etc/apt/sources.list with nano
unhold          - Unhold a package
update          - Download lists of new/upgradable packages
upgrade         - Perform a safe upgrade
version         - Show the installed version of a package
                  This apt has Super Cow Powers
tecmint@tecmint ~ $
```

۲۳. دستور apt-cache

دستور apt-cache برای جستجوی بسته نرم افزاری مورد استفاده قرار می گیرد.به عبارت ساده، این ابزار برای جستجو بسته های نرم افزاری مورد استفاده قرار می گیرد، اطلاعات بسته ها را جمع آوری می کند و همچنین برای جستجو برای بسته های موجود برای نصب در system های مبتنی بر دبیان یا ubuntu مورد استفاده قرار می گیرد.

لیست بسته های نصب شده موجود باapt

برای لیست تمام بسته های موجود، دستور زیر را تایپ کنید.

```
$ apt-cache pkgnames
esseract-ocr-epo
pipenightdreams
mumudvb
libsvm-java
libmrpt-hmtslam0.9
libboost-timer1.50-dev
kcm-touchpad
g ++ - 4.5-multilib
```

مشاهده نام و جزئیات بسته با apt

برای جستجو نام بسته و با توضیحات قبل از نصب، از گزینه search استفاده کنید.با استفاده از جستجو با apt-cache لیستی از بسته ها با توضیحات کوتاه نمایش داده می شود.بگذارید بگوییم دوست دارید توضیحات بسته vsftpdرا بیابید، سپس دستور می شود:

```
$ apt-cache search vsftpd
```

برای جستجو و فهرست کردن تمام بسته هایی که از vsftpdشروع می شوند، می توانید از دستور زیر استفاده کنید.

```
$ apt-cache pkgnames vsftpd
vsttpd
```

بررسی Cache بسته ها با apt

دستور stats مربوط به cache memory را نمایش می دهد.به عنوان مثال ، دستور زیر نمایش داده خواهد شد Packets مجموع تعداد بسته های موجود در کش است.

```
$ apt-cache stats
```

ارتقا یک بسته نرم افزاری با apt

دستور ‘ upgrade ‘ برای ارتقاء تمام بسته های نصب شده در system استفاده می شود. با این حال، اگر می خواهید ارتقا دهید، از اینکه آیا بسته های نرم افزاری برای تکمیل وابستگی ها اضافه می شوند یا حذف می شوند، از “دستور dist-upgrade ”استفاده کنید.

```
$ sudo apt dist-upgrade
```

نصب چند بسته همزمان با apt

می توانید نام چند بسته را همراه با دستور اضافه کنید تا چندین بسته را در یک زمان نصب کنید.به عنوان مثال ، دستور زیر بسته های “ nethogs “و “ goaccess “ را نصب خواهد کرد.

```
$ sudo apt install nethogs goaccess
```

نصب چندبسته با Wildcard با apt

با استفاده از عبارت منظم می توانید چندین بسته با یک رشته اضافه کنید.برای مثال ، ما برای استفاده از چندین بسته ی که شامل رشته “ * name * “هستند استفاده می کنیم.

```
$ sudo apt  install ‘* name *’
```

با استفاده از دستور زیر ‘ -no-upgrade ‘ بسته های نصب شده از بروزرسانی جلوگیری می کند.

```
$ sudo apt install packageName  --no-upgrade
```

ارتقا یک بسته خاص با apt

دستور “ only-upgrade “ بسته های جدیدی را نصب نمی کند، اما فقط بسته های نصب شده را ارتقا می دهد و بسته های جدید را غیر فعال می کند.

```
$ sudo apt install packageName  -only-upgrade
```

نصب یک نسخه خاص با apt

به سادگی با استفاده از ‘ = ‘ با نام پکیج و اضافه کردن نسخه دلخواه.

```
$ sudo apt install vsftpd = 2.3.5-3ubuntu1
```

دانلود یک بسته بدون نصب با apt

برای دانلودکد منبع بسته خاص، از گزینه ‘ -download-only source ‘ با نام ’package-name‘ به عنوان نشان داده شده استفاده کنید.

```
$ sudo apt -download-only  source vsftpd
```

دانلود سورس یک بسته با apt

برای دانلود و باز کردن کد منبع یک بسته به یک فهرست خاص، دستور زیر را تایپ کنید.

```
$ sudo apt  source vsftpd
```

روز دوازدهم

۲۴. مدیربسته dnf

مقدمه ای بر دستور DNF

DNF (Dandified Yum)یک ابزار مدیریت بسته در توزیع‌های لینوکس مبتنی بر RPM مانندRHEL ، CentOS، Rocky Linux، OELو Fedora است. می تواند بسته های نرم افزاری روی سیستم را نصب، به روز رسانی یا حذف کند. این نسخه نسل بعدی مدیریت بسته yum در نظر گرفته می شود. در نظر گرفته شده است که yum را با بهبود ویژگی هایی مانند حل وابستگی و استفاده از حافظه جایگزین کند. این اولین بار در فدورا ۱۸ معرفی شد و از زمان فدورا ۲۲ مدیر بسته پیش فرض فدورا بوده است.

هر دو dnf و yum برای نصب، حذف یا به‌روزرسانی بسته‌ها در سیستم مبتنی بر RPM استفاده می‌شوند. DNF جایگزینی به روز شده برای yum به دلیل مشکلات حل نشده در YUM است. مشکلات عبارتند از عملکرد ضعیف، استفاده از حافظه بالا و وضوح وابستگی کند. DNF از کتابخانه 'libsolv' برای حل وابستگی ها استفاده می کند، در حالی که yum از API عمومی استفاده می کند. dnf به زبان هایC ، C++و Python نوشته شده است که باعث عملکرد بهتر و استفاده کم از حافظه می شود. از آنجایی که yumفقط در پایتون نوشته می شود، عملکرد پایینی دارد.

نحوه عملکردdnf

DNFیک ابزار مدیریت بسته است که برای نصب/به روز رسانی/کاهش/حذف بسته ها به مخازن متکی است. بنابراین شما باید یک مخزن داشته باشید تا بتوانید از dnf استفاده کنید. به طور پیش فرض، dnfاز فایل پیکربندی جهانی در / etc/dnf/dnf.confو تمام فایل های مخزن (.repo) موجود در / etc/yum.repos.dاستفاده می کند .

.repoفایل شامل نام مخزن، مسیر مخزن، چک GPG و پارامتر را فعال کنید. بر اساس این پارامتر، dnfاز مخزن برای بسته های موجود پرس و جو می کند.

مثال های دستور:

اکثر dnfدستورات مشابه دستورات yum هستند. برای اجرای دستور dnf به امتیاز root نیاز است. ممکن است dnf از قبل در سیستم شما نصب نشده باشد

نصب مدیر بسته dnf

```
# yum install dnf
```

شما قادر خواهید بود dnfدستور را در ترمینال خود اجرا کنید. در زیر پر استفاده ترین dnfدستورات برای مدیریت بسته ها و وابستگی های آنها در سیستم های مبتنی بر RPM آورده شده است.

نصب بسته و وابستگی های با dnf

می توانید از این دستور برای نصب آخرین نسخه یک بسته و وابستگی های آن استفاده کنید.

```
# dnf install package_name
```

حذف یک بسته و وابستگی ها با dnf

برای حذف بسته های نصب شده و وابستگی های آن می توانید از دستور زیر استفاده کنید.

```
# dnf remove package_name
```

خروجی نمونه:

```
[root@rockylinux ~]# # dnf remove iotop
```

بروزرسانی یک بسته و وابستگی ها با dnf

این دستور به شما امکان می دهد تا بسته های نصب شده و وابستگی های آنها را به آخرین نسخه ها به روز کنید.

```
# dnf update package_name
```

شما می توانید از این دستور بدون هیچ آرگومان برای به روز رسانی هر بسته در سیستم استفاده کنید.

```
# dnf update
```

ارتقاء یک بسته و وابستگی ها با dnf

این دستور عملکرد مشابهی مانند updateدستور دارد که یک بسته و وابستگی های آن را به جدیدترین نسخه ارتقا می دهد.

```
# dnf update package_name
```

خروجی دستور

```
[root@rockylinux ~]# dnf upgrade rsyslog
```

استفاده از این دستور بدون بسته به عنوان آرگومان، تمام بسته های نصب شده روی سیستم را ارتقا می دهد.

```
#  dnf upgrade
```

بررسی به روز رسانی های موجود با dnf

می توانید به روز رسانی های موجود برای بسته های نصب شده و وابستگی های آنها را در سیستم خود بررسی کنید.

```
#dnf check-update
```

خروجی دستور

```
[root@rockylinux ~]# dnf check-update
Last metadata expiration check: 0:02:29 ago on Sat 18 Sep 2021 04:49:30 PM IST.

NetworkManager.x86_64                       1:1.30.0-10.el8_4                   baseos
NetworkManager-adsl.x86_64                  1:1.30.0-10.el8_4                   baseos
NetworkManager-bluetooth.x86_64             1:1.30.0-10.el8_4                   baseos
NetworkManager-config-server.noarch         1:1.30.0-10.el8_4                   baseos
NetworkManager-libnm.x86_64                 1:1.30.0-10.el8_4                   baseos
NetworkManager-team.x86_64                  1:1.30.0-10.el8_4                   baseos
NetworkManager-tui.x86_64                   1:1.30.0-10.el8_4                   baseos
NetworkManager-wifi.x86_64                  1:1.30.0-10.el8_4                   baseos
NetworkManager-wwan.x86_64                  1:1.30.0-10.el8_4                   baseos
bpftool.x86_64                              4.18.0-305.19.1.el8_4               baseos
buildah.x86_64                              1.21.4-1.module+el8.4.0+643+525e162a appstream
cockpit-podman.noarch                       32-2.module+el8.4.0+643+525e162a    appstream
```

لیست کردن تمام بسته ها با dnf

این دستور همه بسته‌ها (نصب شده و در دسترس برای نصب) را برای توزیع لینوکس مبتنی بر RPM فهرست می‌کند.

```
# dnf lsit
```

دریافت تنها لیست بسته های نصب شده با dnf

برای دریافت لیست تنها بسته های نصب شده روی سیستم می توانید از دستور زیر استفاده کنید.

```
# dnf list installed
```

خروجی دستور

```
[root@rockylinux ~]# dnf list installed
Installed Packages
GConf2.x86_64                                3.2.6-22.el8                    @AppStream
ModemManager.x86_64                          1.10.8-2.el8                    @anaconda
ModemManager-glib.x86_64                     1.10.8-2.el8                    @anaconda
NetworkManager.x86_64                        1:1.30.0-7.el8                  @anaconda
NetworkManager-adsl.x86_64                   1:1.30.0-7.el8                  @anaconda
NetworkManager-bluetooth.x86_64              1:1.30.0-7.el8                  @anaconda
NetworkManager-config-server.noarch          1:1.30.0-7.el8                  @anaconda
NetworkManager-libnm.x86_64                  1:1.30.0-7.el8                  @anaconda
NetworkManager-team.x86_64                   1:1.30.0-7.el8                  @anaconda
NetworkManager-tui.x86_64                    1:1.30.0-7.el8                  @anaconda
NetworkManager-wifi.x86_64                   1:1.30.0-7.el8                  @anaconda
NetworkManager-wwan.x86_64                   1:1.30.0-7.el8                  @anaconda
PackageKit.x86_64                            1.1.12-6.el8                    @AppStream
```

جستجوی یک بسته در مخازن موجود با dnf

این دستور به شما امکان می دهد با استفاده از کلمه یک بسته خاص را جستجو کنید. اگر کلمه با نام یا خلاصه بسته مطابقت داشته باشد، نتیجه را چاپ می کند.

```
# dnf search [package_name]
```

خروجی دستور:

```
[root@rockylinux ~]# dnf search trace
Last metadata expiration check: 0:07:39 ago on Sat 18 Sep 2021 04:49:30 PM IST.
================================== Name & Summary Matched: trace ==================================
boost-stacktrace.i686 : Run-time component of boost stacktrace library
boost-stacktrace.x86_64 : Run-time component of boost stacktrace library
crash-trace-command.x86_64 : Trace extension module for the crash utility
libbabeltrace.i686 : Common Trace Format Babel Tower
libbabeltrace.x86_64 : Common Trace Format Babel Tower
pcp-pmda-bpftrace.x86_64 : Performance Co-Pilot (PCP) metrics from bpftrace scripts
python3-tracer.noarch : Common files for tracer
trace-cmd.x86_64 : A user interface to Ftrace
tracer-common.noarch : Common files for tracer
traceroute.x86_64 : Traces the route taken by packets over an IPv4/IPv6 network
======================================== Name Matched: trace ========================================
blktrace.x86_64 : Utilities for performing block layer IO tracing in the Linux kernel
bpftrace.x86_64 : High-level tracing language for Linux eBPF
```

دریافت لیست بسته های خاص با dnf

شما همچنین می توانید لیست یک بسته خاص را با ارسال نام بسته به عنوان آرگومان دریافت کنید. اما مطمئن شوید که نام دقیق بسته را دارید وگرنه این دستور کار نخواهد کرد و ممکن است برگردد

Error: No matching Packages to list

```
# dnf list package_name
```

خروجی نمونه:

```
[root@rockylinux ~]# dnf list wireshark
Last metadata expiration check: 0:11:32 ago on Sat 18 Sep 2021 04:49:30 PM IST.
Available Packages
wireshark.x86_64                        1:2.6.2-12.el8                          appstream
[root@rockylinux ~]#
[root@rockylinux ~]#
[root@rockylinux ~]# dnf list openssh-server
Last metadata expiration check: 0:11:59 ago on Sat 18 Sep 2021 04:49:30 PM IST.
Installed Packages
openssh-server.x86_64                   8.0p1-6.el8_4.2                         @anaconda
[root@rockylinux ~]#
```

اطلاعات بسته را با دستور dnf دریافت کنید با dnf

با این دستور می توانید اطلاعات دقیق هر بسته موجود در مخازن را مشاهده کنید. این دستور نام دقیق بسته را انتظار دارد.

```
# dnf info [Package_name]
```

خروجی دستور:

```
[root@rockylinux ~]# dnf info vim-common
Last metadata expiration check: 0:13:51 ago on Sat 18 Sep 2021 04:49:30 PM IST.
Installed Packages
Name         : vim-common
Epoch        : 2
Version      : 8.0.1763
Release      : 15.el8
Architecture : x86_64
Size         : 27 M
Source       : vim-8.0.1763-15.el8.src.rpm
Repository   : @System
From repo    : AppStream
Summary      : The common files needed by any version of the VIM editor
URL          : http://www.vim.org/
License      : Vim and MIT
Description  : VIM (VIsual editor iMproved) is an updated and improved version of the
             : vi editor.  Vi was the first real screen-based editor for UNIX, and is
             : still very popular.  VIM improves on vi by adding new features:
             : multiple windows, multi-level undo, block highlighting and more.  The
             : vim-common package contains files which every VIM binary will need in
             : order to run.
```

نصب مجدد بسته و وابستگی ها با dnf

گاهی اوقات، ممکن است هنگام استفاده از یک بسته خاص، خطاهایی رخ دهد. به جای حذف آن و نصب مجدد، می توانید مستقیماً یک بسته و وابستگی های آن را مجدداً نصب کنید.

```
# dnf reinstall [package_name]
```

دانلود بسته به جای نصب با dnf

همچنین می توانید بسته را به جای نصب در سیستم دانلود کنید. برای انجام این کار، می توانید از --downloadonlyگزینه استفاده کنید و می توانید فهرست دانلود را با استفاده از --downloaddir=<DIR_PATH>گزینه مشخص کنید.

```
# dnf install package_name --downloadonly --downloaddir=DIR_PATH
```

خروجی دستور

```
[root@rockylinux ~]#
[root@rockylinux ~]# dnf -q -y install iotop --downloadonly --downloaddir=/home/deepak
[root@rockylinux ~]#
[root@rockylinux ~]# ls -l /home/deepak/iotop-0.6-16.el8.noarch.rpm
-rw-r--r--. 1 root root 66728 Sep 18 17:04 /home/deepak/iotop-0.6-16.el8.noarch.rpm
[root@rockylinux ~]#
[root@rockylinux ~]# rpm -q iotop
package iotop is not installed
[root@rockylinux ~]#
```

لیست کردن مخازن فعال فعلی با dnf

برای چاپ لیست تمام مخازن فعال می توانید از دستور زیر استفاده کنید.

```
# dnf repolist
```

یا

```
# dnf repolist enabled
```

خروجی دستور

```
[root@rockylinux ~]# dnf repolist
repo id                                    repo name
appstream                                  Rocky Linux 8 - AppStream
baseos                                     Rocky Linux 8 - BaseOS
extras                                     Rocky Linux 8 - Extras
plus                                       Rocky Linux 8 - Plus
```

لیست کردن مخازن غیرفعال فعلی با dnf

برای دریافت لیست تمامی مخازن غیرفعال شده می توانید از گزینه disabled استفاده کنید.

```
# dnf repolist disabled
```

خروجی دستور

```
[root@rockylinux ~]# dnf repolist disabled
repo id                                    repo name
appstream-source                           Rocky Linux 8 - AppStream - Source
baseos-source                              Rocky Linux 8 - BaseOS - Source
devel                                      Rocky Linux 8 - Devel WARNING! FOR BUILDROOT USE ONLY!
extras-source                              Rocky Linux 8 - Extras - Source
ha                                         Rocky Linux 8 - HighAvailability
ha-source                                  Rocky Linux 8 - High Availability - Source
media-appstream                            Rocky Linux 8 - Media - AppStream
media-baseos                               Rocky Linux 8 - Media - BaseOS
```

لیست کردن تمام مخازن با dnf

همچنین می توانید تمام مخازنی که روی سیستم شما پیکربندی شده اند را فهرست کنید.

```
# dnf repolist all
```

خروجی دستور

```
[root@rockylinux ~]# dnf repolist all
repo id                     repo name                                          status
appstream                   Rocky Linux 8 - AppStream                          enabled
appstream-source            Rocky Linux 8 - AppStream - Source                 disabled
baseos                      Rocky Linux 8 - BaseOS                             enabled
baseos-source               Rocky Linux 8 - BaseOS - Source                    disabled
devel                       Rocky Linux 8 - Devel WARNING! FOR BUILDROOT USE ONLY! disabled
extras                      Rocky Linux 8 - Extras                             enabled
extras-source               Rocky Linux 8 - Extras - Source                    disabled
ha                          Rocky Linux 8 - HighAvailability                   disabled
ha-source                   Rocky Linux 8 - High Availability - Source         disabled
media-appstream             Rocky Linux 8 - Media - AppStream                  disabled
media-baseos                Rocky Linux 8 - Media - BaseOS                     disabled
plus                        Rocky Linux 8 - Plus                               enabled
plus-source                 Rocky Linux 8 - Plus - Source                      disabled
```

فعال کردن مخزن خاص با dnf

با استفاده از --enablerepoگزینه می توانید بسته ای را از یک مخزن خاص نصب کنید. همچنین می تواند یک مخزن غیرفعال را برای نصب یک بسته فعال کند.

```
# dnf install package_name --enablerepo=repo_name
```

یا شما می توانید تمام مخازن را غیرفعال کنید و تنها از یک مخزن استفاده کنید:

```
# dnf install package_name -disablerepo=* --enablerepo=repo_name
```

به عنوان مثال ، من `vsftpd` بسته `appstream` را فقط با استفاده از repoنصب می کنم و تمام مخزن های دیگر را غیرفعال می کنم.

غیرفعال کردن مخزن خاص با dnf

--disablerepoبه شما اجازه می دهد تا مخزن را برای اجرای زمان اجرا غیرفعال کنید. dnf را برای نصب بسته ای از آن مخزن محدود می کند. اگر بسته در سایر مخازن فعال موجود نباشد، ممکن است نتوانید بسته را نصب کنید.

```
# yum install package_name --disablerepo=repo_name
```

پیدا کردن فایل خاص یک بسته با dnf

با استفاده از این دستور می توانید متوجه شوید که کدام بسته ها حاوی یک فایل خاص هستند.

```
# dnf provides file_name
```

یا

```
# dnf whatprovides file_name
```

```
[root@rockylinux ~]# dnf provides python3
Last metadata expiration check: 0:14:51 ago on Sat 18 Sep 2021 05:04:05 PM IST.
python36-3.6.8-2.module+el8.3.0+120+426d8baf.x86_64 : Interpreter of the Python programming language
Repo        : @System
Matched from:
Provide    : python3 = 3.6.8-2.module+el8.3.0+120+426d8baf

python36-3.6.8-2.module+el8.3.0+120+426d8baf.x86_64 : Interpreter of the Python programming language
Repo        : appstream
Matched from:
Provide    : python3 = 3.6.8-2.module+el8.3.0+120+426d8baf
```

همانطور که می بینید، نام RPM را در اختیار ما قرار می دهد که مسئول نصب باینری python3 است. ممکن است چند فایل دیگر با همین نام وجود داشته باشد. برای دریافت لیست کامل فایل ها با نام python3، می توانید از regex با دستور dnf استفاده کنید.

```
[root@rockylinux ~]# dnf provides */python3
Last metadata expiration check: 0:16:08 ago on Sat 18 Sep 2021 05:04:05 PM IST.
bash-completion-1:2.7-5.el8.noarch : Programmable completion for Bash
Repo        : @System
Matched from:
Filename    : /usr/share/bash-completion/completions/python3

bash-completion-1:2.7-5.el8.noarch : Programmable completion for Bash
Repo        : baseos
Matched from:
Filename    : /usr/share/bash-completion/completions/python3

freeradius-3.0.20-3.module+el8.4.0+576+2d493411.x86_64 : High-performance and highly configurable
                                                       : free RADIUS server
Repo        : appstream
Matched from:
Filename    : /etc/raddb/mods-available/python3
```

پیدا کردن وابستگی های بسته با dnf

شما می توانید از deplist استفاده کنید .دستور فهرست کردن وابستگی های بسته و بسته هایی که آنها را ارائه می دهد.

```
# dnf deplist package_name
```

به عنوان مثال ، برای دریافت لیستی از rpm های وابسته مورد نیاز برای نصب `nmap` بسته، می توانیم از دستور زیر استفاده کنیم:

```
nmap dnf deplist
```

خروجی دستور:

```
[root@rockylinux ~]# dnf deplist nmap
Last metadata expiration check: 0:19:46 ago on Sat 18 Sep 2021 05:04:05 PM IST.
package: nmap-2:7.70-5.el8.x86_64
  dependency: libc.so.6(GLIBC_2.15)(64bit)
   provider: glibc-2.28-151.el8.x86_64
  dependency: libcrypto.so.1.1()(64bit)
   provider: openssl-libs-1:1.1.1g-15.el8_3.x86_64
  dependency: libcrypto.so.1.1(OPENSSL_1_1_0)(64bit)
   provider: openssl-libs-1:1.1.1g-15.el8_3.x86_64
  dependency: libdl.so.2()(64bit)
   provider: glibc-2.28-151.el8.x86_64
  dependency: libdl.so.2(GLIBC_2.2.5)(64bit)
   provider: glibc-2.28-151.el8.x86_64
  dependency: libgcc_s.so.1()(64bit)
   provider: libgcc-8.4.1-1.el8.x86_64
  dependency: libgcc_s.so.1(GCC_3.0)(64bit)
```

روز سیزدهم

۲۵.دستور head

head / tail / cat / less

چند خط اول یک فایل را نشان می دهد. گزینه n– تعداد خطوط دلخواهی را نشان می دهد (به طور پیش فرض ۱۰ است). دستورhead ده خط اول نام هر فایلی را می خواند. دستور اصلی دستور head به صورت زیر است:

head [options] [file(s)]

برای مثال ، دستور زیر ده خط اول فایل با نام /etc/passwdرا نمایش می دهد.

 # head /etc/passwd

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
```

اگر بخواهید تعداد خطوط بیشتری نسبت به ده پیش‌فرض بازیابی شود، از گزینه n به همراه یک عدد صحیح که تعداد خطوطی را که باید بازیابی می‌شوند، استفاده می‌شود. به عنوان مثال ، دستور زیر ۳ خط اول فایل را نمایش می دهد.

```
# prints the first 3 lines
$ head-n 3 c
this
file
has
tail
```

دستور head در سیستم عامل‌های Unix و مشتقات آن مانند Linux برای نمایش بخش ابتدایی یک فایل استفاده می‌شود. این دستور به صورت پیش‌فرض اولین ۱۰ خط از یک فایل را نمایش می‌دهد. عمدتا برای مشاهده اطلاعات اولیه یک فایل یا خروجی‌هامورد استفاده قرار می‌گیرد.

مثال های دستور:

۱. نمایش ۵ خط اول از یک فایل:

```
head -n 5 file.txt
```

۲. نمایش ۲۰ خط اول از چند فایل:

```
head -n 20 file1.txt file2.txt
```

۳. نمایش ابتدای یک فایل با نمایش شماره خطها:

```
head -n 10 -v file.txt
```

۴. نمایش ابتدای یک خروجی:

```
ls -l | head
```

۵. نمایش بخشی از یک فایل بر اساس بایت:

```
head -c 100 file.txt
```

۶. نمایش خطوط بدون نمایش خطاهای تکراری:

```
head -q file.txt
```

۷. نمایش ابتدای یک فایل بدون نمایش نام فایل:

```
head -h file.txt
```

۸. نمایش تعداد خطها بدون نمایش پیام اطلاعاتی:

```
head -q -v file.txt
```

۹. نمایش ابتدای یک فایل به همراه نام فایل:

```
head -v -f file.txt
```

۱۰. نمایش ابتدای یک فایل به صورت معکوس:

```
head -n -10 file.txt
```

۲۶.دستور tail

چند خط آخر یک فایل را نمایش می دهد.

```
# prints the end of the file, beginning with the 4th line
$ tail-n +4 c
exactly
six
lines
```

دستور tail یک ابزار است که در سیستم‌های عامل مبتنی بر یونیکس (UNIX) و لینوکس استفاده می‌شود. این دستور برای نمایش آخرین محتوای یک فایل متنی استفاده می‌شود.

ساختار دستور:

```
tail [OPTIONS] [FILE]
```

در اینجا، FILE نشان‌دهنده نام فایلی است که می‌خواهید محتوای آخر آن را ببینید. اگر FILE مشخص نشود، tail به صورت پیش‌فرض محتوای آخر فایل استاندارد ورودی را نمایش می‌دهد. برخی از استفاده‌های رایج tail عبارتند از:

نمایش آخرین n خط از یک فایل:

```
tail -n [NUMBER] [FILE]
```

با استفاده از گزینه - n و تعیین یک عدد مثبت برای [NUMBER] ، tail آخرین NUMBER خط از فایل مورد نظر را نمایش می‌دهد.

نمایش آخرین n بایت از یک فایل:

```
tail -c [NUMBER] [FILE]
```

با استفاده از گزینه – c و تعیین یک عدد مثبت برای [NUMBER] ، tail آخرین NUMBER بایت از فایل مورد نظر را نمایش می‌دهد.

دنبال کردن (follow) تغییرات در یک فایل:

```
tail -f [FILE]
```

با استفاده از گزینه –f، tail به صورت تعاملی محتوای آخرین بخش از فایل را نمایش می‌دهد و تغییرات جدید را به صورت زنده نمایش می‌دهد. این بسیار مفید است زمانی که می‌خواهید تغییرات جدید در یک فایل را به صورت زنده پیگیری کنید، مانند فایل‌های لاگ. دستور tail بسیار مفید است زمانی که می‌خواهید به سرعت آخرین بخش از یک فایل را ببینید، به ویژه در مواردی که فایل بسیار بزرگ است. با استفاده از این دستور، می‌توانید به راحتی به آخرین خطوط یا بایت‌ها دسترسی پیدا کنید و تغییرات جدید را ببینید.

مثال‌های دستور:

نمایش ۵ خط آخر از یک فایل:

```
tail -n 5 file.txt
```

نمایش ۲۰ خط آخر از چند فایل:

```
tail -n 20 file1.txt file2.txt
```

نمایش انتهای یک فایل با نمایش شماره خطها:

```
tail -n 10 -v file.txt
```

نمایش انتهای یک خروجی:

```
ps aux | tail
```

نمایش بخشی از یک فایل بر اساس بایت:

```
tail -c 100 file.txt
```

نمایش خطوط بدون نمایش خطهای تکراری:

```
tail -q file.txt
```

نمایش انتهای یک فایل بدون نمایش نام فایل:

```
tail -h file.txt
```

نمایش تعداد خطهای مشخص از انتهای یک فایل:

```
tail -n +15 file.txt
```

نمایش انتهای یک فایل به همراه نام فایل:

```
tail -v -f file.txt
```

نمایش انتهای یک فایل به صورت معکوس:

```
tail -n -10 file.txt
```

روز چهاردهم

۲۷.دستور cat

محتوای فایل را به خروجی (معمولا ترمینال) می فرستد. می تواند فقط با یک فایل یا چندین فایل استفاده شود و اغلب به سرعت آنها را نشان می دهد.

```
$ cat a.txt
this is a file
```

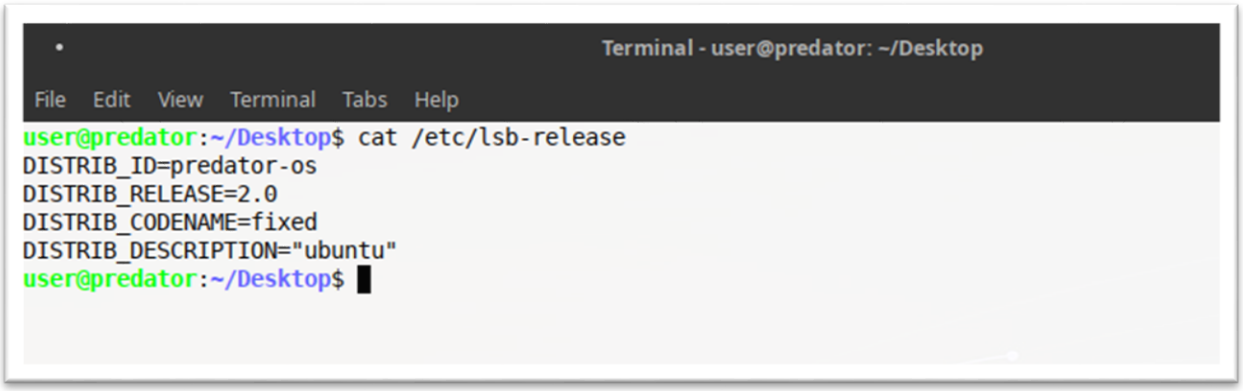
```
cat  a.txt  b.txt
file a
file b
```

دستور Cat الحاق به طور مکرر در Linux استفاده می شود. داده ها را از پرونده می خواند و محتوای آنها را به عنوان خروجی می دهد. این به ما کمک می کند تا پرونده ها را ایجاد ، مشاهده و ادغام کنیم. بنابراین اجازه دهید برخی از دستورات cat که اغلب استفاده می شوند را ببینیم.

مشاهده محتوای یک فایل

بعد از دستور مسیر فایل خود را وارد نمایید.

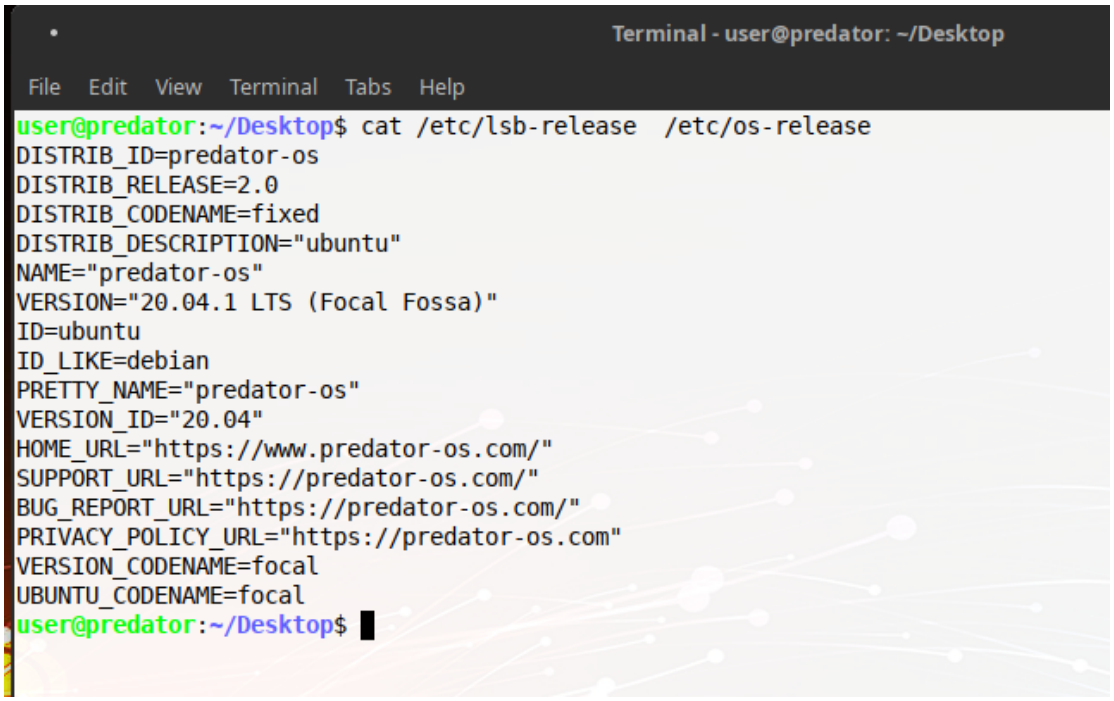
```
cat   /etc/lsb-release
```



مشاهده محتوای چندین فایل

```
$ cat file1 file2
```

خروجی دستور



مشاهده محتویات یک پرونده قبل با شماره خط

```
$cat -n filename
```

خروجی دستور

این محتوا را با شماره خط نشان می دهد

```
cat  -n  ali.txt
```

این مثال است و یک آرایه منحصر به فرد است. دستور زیر یک پرونده ایجاد میکند.

```
$ cat > newfile
```

خروجی دستور

ارسال محتوای یک پرونده را در پرونده دیگر

```
$cat [filename-whose-contents-is-to-be-copied] > [destination-filename]
```

محتوا در پرونده مقصد کپی می شود.

دستور Cat می تواند خطوط خالی تکراری را در دستورخروجی ازبین ببرد

```
$ cat -s ali.txt
```

دستور Cat می تواند محتویات یک فایل را به انتهای پرونده دیگری اضافه کند.

```
$ cat file1 >> file2
```

دستور Cat می تواند با استفاده از دستور tac محتوا را به ترتیب معکوس نمایش دهد.

```
$tac filename
```

دستور Cat می تواند انتهای خط را برجسته کند.

```
$ cat -E “نام پرونده”
$cat -E “filename”
```

اگر می خواهید از گزینه v- ، – Eو T- با هم استفاده کنید ، به جای نوشتن vET- در دستور ، می توانید از گزینه A- خط فرمان استفاده کنید.

```
$ cat -A “نام پرونده”
$cat -A “filename”
```

دستور Cat برای باز کردن فایل های dashed

```
$ cat  - “-dashfile”
```

دستور Cat درصورتی که فایل محتوای زیادی داشته باشد و در ترمینال قرار نگیرد.

```
$cat “filename” | more
```

دستور Cat برای ادغام محتویات چندین پرونده.

```
$ cat “filename1” “filename2” “filename3”> “merged_filename”
```

دستور Cat برای نمایش محتوای تمام فایل های متنی در پوشه.

```
$ cat * .txt.
```

مشاهده محتویات فایل های چندگانه در ترمینال

در مثال زیر، محتویات فایل testو test1 در ترمینال نمایش داده می شود.

```
# cat test test1
```

یک فایل را با دستور cat ایجاد می کند. ما فایلی با نام test2 را با دستور زیر ایجاد خواهیم کرد.

```
# cat  >  test2
```

پس از زدن دستور، متن مورد نظر را تایپ کرده و CTRL + Dرا فشار دهید . متن در فایل test2 نوشته خواهد شد. می توانید محتویات فایل را با دستور catزیر دنبال کنید.

```
# cat  test2
```

از دستور Cat با گزینه های بیشتر و کمتر استفاده کنید اگر فایل با تعداد زیادی از محتوا که در ترمینال خروجی مناسب نیست و صفحه نمایش بسیار سریع حرکت می کند، ما می توانیم با استفاده از پارامترهای بیشتر و کمتر با دستور catبه عنوان مثال بالا استفاده کنیم.

```
# cat song.txt |more
# cat song.txt |less
```

نمایش شماره خط در فایل

با گزینه –n می توانید شماره خطوط یک فایل را در ترمینال خروجی ببینید.

```
# cat   -n  song.txt
```

نمایش $ در پایان فایل

در ادامه میتوانید با گزینه e– علامت " $ " در انتهای خطوط قراردهید.

```
# cat -e test
```

خروجی دستور:

$ سلام به همه، چگونه می توانم انجام دهم؟
$
$ سلام، من خوبم
$ چگونه آموزش خود را ادامه می دهی؟
$

نمایش فاصله بین خطوط جدا شده در فایل

در خروجی پایین، می توانیم بفهمیم فاصله ها با کاراکتر ' ^ ' پر شده است.

```
# cat –T test
```

خروجی دستور:

سلام ^من همه، چطور؟
سلام ^من خوبم
^ ^
بیایید انجام دهیم ^ تمرین را.

نمایش محتوای چندین فایل

در مثال زیر ما دارای 2 فایل test1 وtest2 هستیم و قادر به مشاهده محتویات آن فایل هستیم

cat test1 ؛ cat tset 2

این فایل test1 است.
این فایل test2 است.

استفاده از خروجی استاندارد با اپراتور هدایتگر

ما می توانیم خروجی استاندارد یک فایل را به یک فایل جدید دیگری که با علامت ' > ' نمایش داده می شود، هدایت کنیم. محتویات تست ۱ ، با محتویات فایل تست، رونویسی خواهد شد.

```
# cat test > test1
```

ضمیمه خروجی استاندارد با اپراتور هدایتگر

در فایل جاری با نماد « >> » مقادیر اضافه به فایل داده شده اضافه میشود. در اینجا، محتویات فایل تست در پایان فایل test1 اضافه می شود.

```
# cat test >> test1
```

انتقال استاندارد ورودی با اپراتور هدایتگر

از نام فایل test2 به عنوان ورودی برای یک دستور استفاده می کند و خروجی در ترمینال نشان داده می شود.

```
# cat < test2
```

مسیر فایل های چندگانه در یک فایل

این یک فایل به نام test3ایجاد خواهد کرد و تمام خروجی ها در یک فایل جدید ایجاد شده هدایت می شوند.

```
# cat  test  test1  test2  > test3
```

مرتب سازی محتویات فایل های چندگانه در یک فایل

در این مثال یک فایل تست ۴ ایجاد می شود و خروجی های دستور cat به صورت مرتب در یک فایل تازه ایجاد شده هدایت می شود.

```
# cat   test  test1  test2  test3 |  ordered > test4
```

۲۸.دستور grep

فیلتر grep یک فایل را برای یک الگوی خاص از نویسه ها جستجو می کند و تمام خطوطی را که حاوی آن الگو هستند نمایش می دهد. از الگویی که در پرونده جستجو می شود به عنوان عبارت منظم یاد می شود (grep مخفف عبارت globally search for منظم و چاپ کردن است.

ساختار دستور:

```
grep [options] pattern [files]
```

- cتعداد خطوطی که با یک الگو مطابقت دارند را چاپ می‌کند.
- hخطوط همسان را نمایش می‌دهد، اما نام پرونده‌ها را نمایش نمی‌دهد.
- iبرای تطبیق، بدون توجه به حروف بزرگ و کوچک، عمل می‌کند.
- lفقط لیست نام پرونده‌ها را نمایش می‌دهد.
- nشماره خطها را نمایش دهید.
- vتمام خطوطی که با الگو مطابقت ندارند را چاپ می‌کند.
- f fileالگوها را از پرونده دریافت می‌کند، یکی در هر خط.
- Eالگو را به عنوان یک عبارت منظم گسترده (ERE) در نظر می‌گیرد.
- wتنها کلمات کامل را مطابقت دهید.
- oفقط بخش‌های مطابق یک خط را چاپ می‌کند.

مثال بیشتر:

فایل زیر را به عنوان ورودی در نظر بگیرید.

```
$ cat > mypasswords.txt
```

جستجوی عدم حساسیت به پرونده: گزینه i- امکان جستجوی یک مورد رشته ای را به صورت غیر حساس در پرونده داده فراهم می کند. با کلماتی مانند "UNIX" ، "Unix"، "unix"مطابقت دارد.

```
$ grep -i “UNix”   mypasswords.txt
```

نمایش تعداد تعداد تطبیق ها: می توانیم تعداد خطوطی را که با رشته / الگوی داده شده مطابقت دارد پیدا کنیم

```
$ grep -c “unix”   mypasswords.txt
```

یا

```
$ grep -l “unix” *
```

یا

```
$ grep -l “unix” f1.txt f2.txt f3.xt f4.txt
```

خروجی دستور:

```
mypasswords.txt
```

بررسی کل کلمات در یک پرونده: به طور پیش فرض ، grepبا رشته / الگوی داده شده مطابقت دارد حتی اگر به عنوان زیر رشته در یک فایل پیدا شود. گزینه w- برای grep باعث می شود که فقط با کل کلمات مطابقت داشته باشد.

```
$ grep -w “unix”  mypasswords.txt
```

نمایش فقط الگوی همسان: به طور پیش فرض ، grepکل خطی را که دارای رشته همسان است نشان می دهد. با استفاده از گزینه o- می توانیم grep را طوری نمایش دهیم که فقط رشته همسان را نشان دهد.

```
$ grep -o “unix”  mypasswords.txt
```

نمایش شماره خط هنگام نمایش خروجی با استفاده از :n- grep برای نشان دادن شماره خط پرونده با خط همسان.

```
$ grep -n “unix”  mypasswords.txt
```

معکوس کردن مطابقت الگو: با استفاده از گزینه v- می توانید خطوطی را که با الگوی جستجوی مشخص شده مطابقت ندارند نمایش میدهد.

```
$ grep -v “unix”  mypasswords.txt
```

تطبیق خطوطی که با یک رشته شروع می شوند: الگوی بیان منظم ^ شروع یک خط را مشخص می کند. این را می توان در grepبرای مطابقت با خطوطی که با رشته یا الگوی داده شده شروع می شوند ، استفاده کرد.

```
$ grep “^ unix”  mypasswords.txt
```

مطابقت خطوطی که با یک رشته خاتمه می یابند: الگوی بیان $ منظم انتهای یک خط را مشخص می کند. این را می توان در grepبرای مطابقت با خطوطی که با رشته یا الگوی داده شده مطابقت دارند استفاده کرد.

```
$ grep “os $”  mypasswords.txt
```

بیان را با گزینه e- مشخص می کند. می تواند چندین بار استفاده کند:

```
$ grep –e “Agarwal” –e “Aggarwal” - e “Agrawal”  mypasswords.txt
```

f- گزینه پرونده الگوها را از پرونده می گیرد ، یکی در هر خط.

```
$ cat pattern.txt
```

```
$ grep –f pattern.txt  mypasswords.txt
```

۲۹.دستور exit

از دستور exit در linux برای خروج از پوسته ای که در حال حاضر در آن اجرا می شود استفاده می شود. یک پارامتر دیگر به عنوان [N] طول می کشد و با بازگشت وضعیت N از پوسته خارج می شود . اگر n ارائه نشده باشد ، به سادگی وضعیت آخرین دستوری را که اجرا می شود برمی گرداند.

ساختار دستور:

```
exit [n]
```

گزینه های دستور:

:exitخروج بدون پارامتر

```
File  Edit  View  Search  Terminal  Help
naman@root:~$ exit
```

exit [n]: خروج با پارامتر

```
File  Edit  View  Search  Terminal  Help
naman@root:~$ exit 110
```

پس از فشار دادن enter ، پنجره ترمینال بسته می شود و وضعیت ۱۱۰ را برمی گرداند. وضعیت بازگشت مهم است زیرا گاهی اوقات می توان آنها را برای تشخیص خطا ، هشدارها و اعلان ها ترسیم کرد. به عنوان مثال

- وضعیت بازگشت:"0"به این معنی است که برنامه با موفقیت اجرا شده است.
- "1"به معنای این است که برنامه دارای خطاهای جزئی است.
- exit n:با استفاده از "sudo su" ما به فهرست ریشه می رویم و سپس از فهرست ریشه با وضعیت بازگشت ۵ خارج می شویم. پس از بازگشت ، نحوه نمایش کد وضعیت بازگشت نشان داده می شود.

```
File  Edit  View  Search  Terminal  Help
naman@root:~$ sudo su
[sudo] password for naman:
root@root:/home/naman# exit 5
exit
naman@root:~$ echo $?
5
naman@root:~$
```

echo $?

از این دستور برای دیدن آخرین وضعیت بازگشت استفاده می شود.

exit –help

اطلاعات راهنما را نمایش می دهد.

```
naman@root:~$ exit --help
exit: exit [n]
    Exit the shell.

    Exits the shell with a status of N.  If N is omitted, the exit status
    is that of the last command executed.
naman@root:~$
```

روز پانزدهم

مقایسه فایل ها

به طور معمول، برای مقایسه دو فایل در linux، از diffاستفاده می کنیم. یک ابزار خط دستور یونیکس ساده و اصلی است که تفاوت بین دو فایل کامپیوتری را نشان می دهد، فایل ها را به صورت خطی مقایسه می کند و از آن آسان است که استفاده می شود، با بسیاری از توزیع های linux نصب می شود. سوال این است که چگونه می توانیم بین دو فهرست در linux تفاوت ایجاد کنیم؟در اینجا، ما می خواهیم بدانیم که کدام فایل ها / زیر شاخه ها در دو فهرست مشترک هستند، کسانی که در یک فهرست موجود هستند، اما نه در فهرست.

۳۰.دستور diff

دستورالعمل متعارف برای اجرای diff به شرح زیر است:

```
$ diff [option] ... files
$ diff dir1 dir2
```

به طور پیش فرض، خروجی آن به ترتیب حروف الفبا توسط نام فایل / زیر شاخه به ترتیب نشان داده شده است.در این دستور، q– گزارش را فقط زمانی که پرونده ها متفاوت هستند، گزارش می کند.

```
$ diff -q directory-1 / directory-2 /
```

دستور diff به subdirectories عمل نمیکند، اما می توانیم از گزینه r– برای خواندن subdirectories ها نیز استفاده کنیم.

```
$ diff -qr directory-1 / directory-2 /
```

دستور diff در سیستم عامل‌های Unix و مشتقات آن مانند Linux برای مقایسه محتوای دو فایل متنی به کار می‌رود. این دستور تفاوت‌های بین دو فایل را نمایش می‌دهد و به کاربر امکان می‌دهد تا خطوطی که در یک فایل وجود دارند اما در فایل دیگر نیستند یا برعکس را مشاهده کند.

مثال های دستور:

۱. مقایسه دو فایل و نمایش تفاوت‌ها به صورت خروجی استاندارد:

```
diff file1.txt file2.txt
```

۲. مقایسه دو فایل و نمایش تفاوت‌ها به صورت کامل (تفاوت‌ها و تطابق‌ها):

```
diff -u file1.txt file2.txt
```

۳. مقایسه دو فایل و نمایش تنها خطوط متفاوت:

```
diff -q file1.txt file2.txt
```

۴. مقایسه دو فایل و نمایش تفاوت‌ها به صورت کوتاه:

```
diff -c file1.txt file2.txt
```

۵. مقایسه دو فهرست و نمایش تفاوت‌ها:

```
diff -r directory1 directory2
```

۶. مقایسه دو فایل و ذخیره نتیجه در یک فایل خروجی دستور:

```
diff file1.txt file2.txt > differences.txt
```

۷. مقایسه دو فایل و نمایش تفاوت‌ها به صورت خط به خط:

```
diff -y file1.txt file2.txt
```

۸. مقایسه دو فایل و نمایش تغییرات به صورت خلاصه:

```
diff -e file1.txt file2.txt
```

۹. مقایسه دو فایل و نمایش تفاوت‌ها به صورت کلمه به کلمه:

```
diff -w file1.txt file2.txt
```

۱۰. مقایسه دو فایل و نمایش تفاوت‌ها به صورت باز و بسته:

```
diff -B file1.txt file2.txt
```

۳۱.دستور sed

sed مخفف stream editor است. یک ویرایشگر جریان برای انجام تبدیل های متنی اساسی در یک جریان ورودی (فایل یا ورودی) استفاده می شود. به عنوان **مثال**، می تواند بسیاری از عملکردها را روی فایل ها مانند جستجو، یافتن و جایگزینی، درج یا حذف انجام دهد. در حالی که از برخی جهات شبیه ویرایشگری است که ویرایش‌های اسکریپت‌شده را مجاز می‌کند (مانند ed)، sed تنها با یک عبور از ورودی(ها) کار می‌کند و در نتیجه کارآمدتر است. اما این توانایی sed برای فیلتر کردن متن در خط pipe است که آن را از سایر انواع ویرایشگرها متمایز می‌کند. دستور sed برای جایگزینی یا find and replace است. با استفاده از sed می‌توانید فایل‌ها را حتی بدون باز کردن آن ویرایش میکند، که راه بسیار سریع‌تری برای یافتن و جایگزین کردن چیزی در فایل است. از عبارات منظم اولیه و توسعه یافته پشتیبانی می‌کند که به شما امکان می‌دهد الگوهای پیچیده را مطابقت دهید. اکثر توزیع های لینوکس با گنو ارائه می شوند و sed به طور پیش فرض از قبل نصب شده است. دستور sed یک ابزار قدرتمند در لینوکس و سیستم عامل‌های مشابه است که برای ویرایش و تغییر محتوای خطوط فایل‌ها استفاده می‌شود. این دستور قابلیت جستجو و جایگزینی الگوها در فایل‌ها را دارد و می‌تواند برای انجام تغییرات چندین خط یا پرونده به صورت همزمان استفاده شود.

ساختار دستور:

```
$sed [گزینه‌ها] ‘دستور’ [فایل‌ها]
```

گزینه های دستور:

- e – : استفاده از چندین دستور در یک خط دستور.
- f – : خواندن دستورها از یک فایل.
- i – : ویرایش فایل‌ها در حالت مستقیم (دستورات اصلاحی را در فایل‌ها ذخیره می‌کند).
- n – : عدم چاپ خطوط به صورت خودکار.
- r یا regexp-extended -- : استفاده از عبارات منظم گسترده.

مثال های دستور:

جایگزینی الگو:

```
$sed ‘s/foo/bar/’ filename
```

جایگزینی تمام نمونه‌های الگو در یک خط:

```
$sed ‘s/foo/bar/g’ filename
```

جایگزینی الگو در یک بازه خطی:

```
$sed ‘2,4s/foo/bar/’ filename
```

حذف خطوط با الگوی مشخص:

```
$sed ‘/pattern/d’ filename
```

افزودن یک خط پس از یک الگو:

```
$sed ‘/pattern/a\New Line’ filename
```

افزودن یک خط قبل از یک الگو:

```
$sed ‘/pattern/i\New Line’ filename
```

نمایش خطوط با الگوی مشخص:

```
$sed -n ‘/pattern/p’ filename
```

روز شانزدهم

۳۱.دستور tee

دستور tee در سیستم عامل‌های Unix و مشتقات آن مانند Linux برای خواندن و نوشتن همزمان اطلاعات از یک ورودی (مانند خروجی یک دستور)، آن را به یک یا چند فایل متنی ارسال می‌کند و همچنین اطلاعات را به خروجی استاندارد می‌فرستد.

مثال‌های دستور:

۱. نوشتن خروجی یک دستور به فایل و نمایش آن در خروجی استاندارد:

```
ls -l | tee file.txt
```

۲. اضافه کردن خروجی یک دستور به انتهای یک فایل موجود و نمایش آن در خروجی استاندارد:

```
echo “New line” | tee -a file.txt
```

۳. نوشتن خروجی یک دستور به چند فایل متنی مختلف:

```
ps aux | tee process1.txt process2.txt
```

۴. نوشتن خروجی یک دستور به یک فایل متنی و انتقال آن به یک فایل دیگر:

```
ls -l | tee file1.txt | tee file2.txt
```

۵. استفاده از tee به همراه pipeکشی ((pipingبرای انتقال خروجی یک دستور به فایل:

```
cat file.txt | grep “pattern” | tee filtered_output.txt
```

۶. نوشتن خروجی یک دستور به فایل با دسترسی خواندنی و قابل اجرا:

```
ls -l | tee file.txt
chmod +x file.txt
```

۷. نوشتن خروجی یک دستور به فایل با دسترسی فقط خواندنی:

```
ps aux | tee file.txt
chmod -r file.txt
```

۸. نوشتن خروجی یک دستور به فایل و نمایش پیامی پس از انجام عملیات:

```
ls -l | tee file.txt && echo “Output saved to file.txt”
```

۹. نوشتن خروجی یک دستور به یک فایل و انتقال خروجی به یک دستور دیگر:

```
cat file.txt | tee -a output.txt | grep “pattern”
```

۱۰. نوشتن خروجی یک دستور به فایل و نمایش پیام خطایی در صورت خطا:

```
ls -l | tee file.txt || echo “Error: Failed to save output to file.txt”
```

۳۲.دستور nano

دستور nano به شما امکان ایجاد/ویرایش فایل های متنی را در پنجره می‌دهد.

نصب و راه اندازی:

ویرایشگر متن نانو روی macOS و اکثر توزیع‌های لینوکس از پیش نصب شده است. این برنامه جایگزینی برای vi و vim است. برای بررسی اینکه آیا روی نوع سیستم شما نصب شده است یا خیر دستور زیر را وارد نمایید:

```
$ nano --version
```

اگر نانو را نصب نکرده اید، می‌توانیداز دستور زیر و با استفاده از مدیریت بسته‌ها آن را نصب کنید:

نصب در اوبونتو یا دبیان:

```
$ sudo apt install nano
```

فدورا:

```
$ sudo dnf install nano
```

مثال های دستور:

۱.برای باز کردن یک فایل ، ابتدا nano را تایپ کنید و سپس مسیر فایل را تایپ کنید:

```
$ nano /path/to/filename
```

۲. برای ایجاد یک فایل جدید ، nano و سپس نام فایل را تایپ کنید:

```
$ nano filename
```

۳. برای باز کردن یک فایل را با مکان نما روی یک خط و کاراکتر خاص از دستور زیر استفاده کنید:

```
$ nano +line_number,character_number filename
```

مروری بر برخی از میانبرها و عملکردهای آنها:

توضیحات میانبرها:

- Ctrl + S ذخیره فایل فعلی
- Ctrl + O پیشنهاد نوشتن فایل (“ذخیره به عنوان”) جدید
- Ctrl + X بستن بافر، خروج از نانو
- Ctrl + K خط فعلی را به cutbuffer برش میدهد
- Ctrl + U محتویات cutbuffer را paste میکند
- Alt + 6 خط فعلی را در cutbuffer کپی میکند.
- Alt + U برای Undo کردن آخرین اقدام
- Alt + E انجام مجدد آخرین اقدام لغو شده–redo

روز هفدهم

۳۳.دستور echo

دستور echo به شما امکان می‌دهد خط متن/رشته ای را که به عنوان آرگومان ارسال می شود را نمایش دهید. دستور echo در لینوکس و سیستم‌عامل‌های مشابه استفاده می‌شود. این دستور برای نمایش رشته‌ها (متن) در ترمینال استفاده می‌شود.

ساختار دستور:

```
$echo [رشته‌ها] [گزینه‌ها]
```

در این ساختار، شما می‌توانید از گزینه‌ها و رشته‌ها برای تنظیمات و نمایش متن موردنظر استفاده کنید. البته استفاده از گزینه‌ها و رشته‌ها اختیاری است و می‌توانید تنها با استفاده از echo بدون هیچ پارامتری، یک خط خالی را در ترمینال نمایش دهید.

گزینه های دستور:

- n – این گزینه باعث می‌شود echo عبارت را چاپ کند اما به انتهای خط جدید رفتار نکند. به عبارت دیگر، خط جدیدی که به طور پیش‌فرض پس از چاپ عبارت ایجاد می‌شود، حذف می‌شود.
- e – این گزینه به echo می‌گوید که رشته ورودی را بررسی کند و کاراکترهای خاص را تفسیر کند. برخی از کاراکترهای خاص که توسط این گزینه تفسیر می‌شوند عبارتند از \) nخط جدید)، \) tتب) و \\ (کاراکتر برگشتی).
- E – این گزینه باعث می‌شود echo کاراکترهای خاص را تفسیر نکند و به عنوان رشته‌های عادی چاپ کند.
- به علاوه، می‌توانید رشته‌های متغیر را با دستور echo نمایش دهید. برای این کار، رشته مورد نظر را با $ قبل از نام متغیر محاصره کنید.

مثال های دستور:

```
name=“John”
$echo “My name is $name”
```

در این مثال، با استفاده از دستور echo، رشته “My name is John” در ترمینال نمایش داده می‌شود. دستور echo در کاربردهای مختلف مانند اسکریپت‌نویسی و تنظیمات سیستم بسیار مفید است و می‌توانید از آن برای نمایش و چاپ رشته‌ها در ترمینال استفاده کنید.

یک خط از متن را وارد کنید و در خروجی استاندارد نمایش داده شود.

```
$ echo hosein and linux
```

خروجی متن زیر است:

```
hosein and linux
```

یک متغیر را تایپ کنید و مقدار آن را echo کنید.به عنوان مثال ، اعلام یک متغیر از xو اختصاص مقدار آن 10

```
$ x = 10
```

مقدار خروجی:

```
$ echo x
```

مقدار متغیر x = 10

نکته: گزینه ‘ e- ‘ تمام فضاهای بین خطوط را حذف می کند.

```
$ echo -e “ hosein \ bis \ ba \ bcommunity \ bof \ bLinux \ bNerds”
```

با استفاده از گزینه - ‘ \ n ‘ خط جدید را از جایی که از آن استفاده می شود ایجاد می کند.

```
$ echo -e “ hosein  \ nis \ na \ ncommunity \ nof \ n linux \ n”
```

با استفاده از گزینه ‘ c \ ‘ خط جدید ادامه می دهد.

```
$ echo -e “ hosein  Community  /c  of Linux Nerds “
```

تمامی فایل ها / فهرست را با استفاده از دستور چاپ کنید.

```
$ echo *
```

```
103.odt 103.pdf 104.odt 104.pdf 105.odt 105.pdf 106.odt 106.pdf 107.odt 107.pdf 108a.odt 108.odt 108.pdf 109.odt 109.pdf 110b.odt 110. odt 110.pdf 111.odt 111.pdf 112.odt 112.pdf 113.odt linux-headers-3.16.0-customkernel_1_amd64.deb linux-image-3.16.0-customkernel_1_amd64.deb network.jpeg
```

فایلهای یک نوع خاص را چاپ کنید.به عنوان مثال ، فرض کنید میخواهید تمام فایلهای . jpegرا چاپ کنید، از دستور زیر استفاده کنید.

```
$ echo * .jpeg
```

مثال های دستور

۱.برای نمایش خط متن یا رشته ارسال شده به عنوان آرگومان:

```
$ echo Hello There
```

۲. برای نمایش همه فایل ها/پوشه ها مشابه دستور ls :

```
$ echo *
```

۳. برای ذخیره متن در فایلی به نام foo.bar:

```
$ echo “Hello There” > foo.bar
```

۴. برای افزودن متن به فایلی به نام foo.bar:

```
$ “echo Hello There” >> foo.bar
```

معمولاً در اسکریپت‌های پوسته یاهمون شل و فایل‌های دستهای ،برای خروجی متن به صفحه یا یک فایل استفاده می‌شود. e– که همراه آن استفاده می‌شود، تفسیر بک اسلش را امکان‌پذیر می‌کند.

\b	تمام فاصله های بین متن را حذف می‌کند
\ c	برای ادامه بدون ایجاد خط جدید،
\ n	خط جدیدی را از جایی که استفاده می شود ایجاد می‌کند
\ t	horizontal tabایجاد می‌کند
\ r	carriage با مفسر e-‘ backspace’ برمی گردد تا در خروجی بازده بار مشخصی داشته باشد
\ v	vertical tabایجاد می‌کند
\ a	هشدار صوتی برمی‌گردد
– n	خط جدید دنباله دار را حذف می‌کند.

نمایش یک رشته ساده:

```
$echo “Hello, World!”
```

خروجی دستور:

```
Hello, World!
```

نمایش رشته به همراه کاراکتر خاص (مثلاً خط جدید):

```
$echo -e “Line 1\nLine 2”
```

خروجی دستور:

```
Line 1
Line 2
```

نمایش متغیرها:

```
name=“John”
age=25
echo “My name is $name and I am $age years old.”
```

خروجی دستور:

```
My name is John and I am 25 years old.
```

نمایش تاریخ و زمان فعلی:

```
$echo “Today is $(date +%A), $(date +%F) and the time is $(date +%T).”
```

خروجی دستور:

```
Today is Wednesday, 2022-09-28 and the time is 14:30:00.
```

استفاده از گزینه `-n` برای حذف خط جدید:

```
$echo -n “This is a “
$echo “single line.”
```

خروجی دستور:

```
This is a single line.
```

مقادیر **environment variables**

نام متغییر	**توضیح دهید**
BASH_VERSION	شماره نسخه فعلی نمونه پوستهBash
EDITOR	ویرایشگر پیش فرض استفاده شده توسط برخی از دستورات پوسته
GROUPS	عضویت گروهی حساب کاربری
HISTFILE	نام فایل تاریخچه فرمان پوسته کاربر
HISTSIZE	حداکثر تعداد دستورات ذخیره شده در فایل تاریخچه
HOME	نام فهرست اصلی کاربر فعلی
HOSTNAME	نام میزبان سیستم فعلی
LANG	دسته بندی محلی برای پوسته
LC_*	تنظیمات محلی مختلف که LANG را لغو می کنند
LC_ALL	دسته محلی برای پوسته ای که LANG را لغو می کند
LD_LIBRARY_PATH	فهرست فهرست های کتابخانه جدا شده با دو نقطه برای جستجو قبل از جستجو در فهرست های استاندارد کتابخانه
PATH	فهرست فهرست های جدا شده با دو نقطه برای جستجوی دستورات
PS1	رشته درخواست رابط خط فرمان پوسته اولیه
PS2	رشته اعلان رابط خط فرمان پوسته ثانویه
PWD	فهرست کار فعلی حساب کاربر
SHLVL	سطح پوسته فعلی
TZ	منطقه زمانی کاربر، اگر با منطقه زمانی سیستم متفاوت باشد
UID	شماره شناسایی کاربر حساب کاربری

روز هجدهم

مدیریت سیستم فایل،دیسک و پارتیشن ها

۳۴.دستور smartctl

دستور smartctl در سیستم عامل‌های Unix/Linux برای دسترسی و نمایش اطلاعات SMART (Self-Monitoring, Analysis, and Reporting Technology) دیسک‌های سخت (hard driveو SSD) استفاده می‌شود. این دستور به شما امکان می‌دهد تا وضعیت و عملکرد دیسک‌های خود را بررسی کنید و از پیش‌اخطارهای ممکن درباره خرابی یا نقص‌های احتمالی مطلع شوید.

مثال‌های دستور:

نمایش اطلاعات SMART دیسک: این دستور برای نمایش جزئیات و اطلاعات SMART دیسک مورد استفاده قرار می‌گیرد.

```
smartctl -a /dev/sda
```

تست کامل دیسک: این دستور برای انجام یک تست کامل بر روی دیسک و بررسی وضعیت آن استفاده می‌شود.

```
smartctl -t long /dev/sdb
```

نمایش وضعیت خطاهای دیسک: این دستور برای نمایش وضعیت خطاهای دیسک و تعداد بلوک‌های پیش‌اخطاری استفاده می‌شود.

```
smartctl -l error /dev/sdc
```

تنظیم و ذخیره‌سازی تنظیمات SMART: این دستور برای تنظیم و ذخیره‌سازی تنظیمات SMART بر روی دیسک استفاده می‌شود.

```
smartctl -s on -S /dev/sdd
```

تست سریع دیسک: این دستور برای انجام یک تست سریع بر روی دیسک و بررسی وضعیت آن استفاده می‌شود.

```
smartctl -t short /dev/sde
```

۳۵.دستور badblocks

دستور badblocks در سیستم‌عامل‌های Unix/Linux برای تشخیص و شناسایی بلاک‌های خراب یا بد در یک دیسک سخت (هارد دیسک) استفاده می‌شود. این دستور به کاربر این امکان را می‌دهد تا بلاک‌های خراب یا بد را بررسی کرده و اطلاعات مربوط به آن‌ها را دریافت کند.

مثال‌های دستور:

بررسی بلاک‌های خراب در یک دیسک سخت: این دستور برای بررسی بلاک‌های خراب در یک دیسک سخت با استفاده از حالت خواندن نوشتن تصادفی (read-write modeاستفاده می‌شود.

```
badblocks -w /dev/sda
```

بررسی بلاک‌های خراب با استفاده از حالت خواندن نوشتن تصادفی: این دستور برای بررسی بلاک‌های خراب در یک دیسک سخت با استفاده از حالت خواندن نوشتن تصادفی و نمایش اطلاعات جامع در خروجی استفاده می‌شود.

```
badblocks -vw /dev/sdb
```

بررسی و نمایش بلاک‌های خراب در حالت خواندن: این دستور برای بررسی بلاک‌های خراب در یک دیسک سخت با استفاده از حالت خواندن و نمایش اطلاعات جامع در خروجی استفاده می‌شود.

```
badblocks -n /dev/sdc
```

بررسی بلاک‌های خراب و ذخیره نتایج در فایل: این دستور برای بررسی بلاک‌های خراب در یک دیسک سخت و ذخیره نتایج در یک فایل خروجی استفاده می‌شود.

```
badblocks -o badblocks_output.txt /dev/sdd
```

بررسی و تعمیر بلاک‌های خراب در حالت خواندن و نوشتن: این دستور برای بررسی و تعمیر بلاک‌های خراب در یک دیسک سخت با استفاده از حالت خواندن و نوشتن استفاده می‌شود.

```
badblocks -c 10240 -w -s /dev/sde
```

۳۶.دستور dd

دستور dd برای کپی فایل ها، تبدیل و قالب بندی با توجه به گزینه های ارائه شده در خط دستور استفاده می شود.مثال زیر نشان می دهد ایجاد یک دستگاه USB با قابلیت بوت :

```
$ dd if = /home/hosein/kali-linux-3-amd64.iso  of =/dev/sdc1  bs = 512M
```

روز نوزدهم

۳۷.دستور df

در اینترنت مقدار زیادی ابزار برای بررسی استفاده از فضای دیسک در linux پیدا خواهید کرد.با این حال، linux یک ابزار قدرتمند ساخته شده را به نام ‘ df ‘ ساخته است.دستور “ df “ برای “ سیستم فایل دیسک ” است، برای خلاص شدن کامل از در دسترس بودن و استفاده از فضای استفاده از دیسک system در linux system استفاده می شود. با استفاده از پارامتر « h- » با df -h ((آمار فضای دیسک فایلی را در فرمت » قابل خواندن برای انسان[1] نشان می دهد، به این معنی که جزئیات در بایت، مگا بایت و گیگابایت را می دهد. دستور “ df “ اطلاعات نام دستگاه، بلوک های کل، فضای دیسک کل، فضای دیسک مورد استفاده، فضای دیسک در دسترس و نقاط سوپاپ در یک سیستم فایل را نمایش می دهد.

```
$ sudo df
```

هنگام استفاده از لینوکس ممکن است شرایطی پیش بیاید که بخواهید از میزان فضای مصرف شده توسط یک سیستم فایل خاص در سیستم LINUX خود یا میزان فضای موجود در یک فایل سیستم خاص مطلع شوید. اگر هیچ نام پرونده ای به عنوان آرگومان با دستور df منتقل نشود ، آنگاه فضای موجود در تمام سیستم های پرونده نصب شده در حال حاضر را نشان می دهد. این چیزی است که شما ممکن است بخواهید بدانید زیرا دستور **df** قادر به نشان دادن فضای موجود در سیستم فایلهای پیاده نشده است و دلیل این امر این است که برای انجام این کار در برخی از سیستم ها نیاز به دانش بسیار عمیقی در مورد ساختار سیستم فایل پرونده دارد.
به طور پیش فرض ، **df** فضای دیسک را در **1** بلوک **K** نشان می دهد .

ساختار دستور:

```
df [OPTION] ... [FILE] ...
```

فرض کنید فایلی به نام kt.txt دارید و می خواهید از فضای دیسک استفاده شده در سیستم فایل که شامل این فایل است ، بدانید ، بنابراین می توانید از **df** در این حالت استفاده کنید:

```
$ df kt.txt
```

حال ، اگر هیچ نام پرونده ای به t df ندهید ، در این صورت df اطلاعات استفاده از دیسک را برای تمام سیستم های پرونده نصب شده به صورت زیر نشان می دهد:

```
$ df
```

نمایش اطلاعات تمام سیستم فایل استفاده از فضای دیسک

اطلاعات system های فایل ساختگی همراه فضای استفاده از دیسک و استفاده از memory را نمایش می دهد.

```
$ sudo df -a
```

نمایش استفاده از فضای دیسک در فرمت قابل خواندن انسانی

متوجه شده اید که دستورات بالا اطلاعات را در بایت نمایش می‌دهد، که هنوز قابل خواندن نیست، چرا که ما عادت به خواندن اندازه ها در مگابایت، گیگابایت و غیره هستیم زیرا بسیار آسان است که بتوانیم آن را درک کنیم و به یاد داشته باشیم. دستور DFگزینه ای برای نمایش اندازه در فرمت های خوانده شده انسان فراهم می کند با استفاده از ’h-‘ چاپ نتایج در فرمت قابل خواندن انسانی به عنوان مثال :

[1] human read

```
$ sudo df -h
```

```
/ dev / cciss / c0d0p2 75G 23G 49G 32 ٪/
/ dev / cciss / c0d0p5 24G 22G 1.2G 95 ٪/ home
/ dev / cciss / c0d0p3 29G 25G 2.6G 91 ٪/ data
/ dev / cciss / c0d0p1 289M 22M 253M 8 ٪/ boot
```

نمایش اطلاعات / سیستم فایل home

برای مشاهده اطلاعات تنها سیستم فایل /home در فرمت قابل خواندن برای انسانی، از دستور زیر استفاده کنید.

```
$ sudo df -hT / home
```

```
/dev/cciss/c0d0p5 ext3 24G 22G 1.2G 95 ٪home
```

نمایش اطلاعات سیستم فایل درحالت Bytes

برای نمایش تمام اطلاعات سیستم فایل و استفاده در بلوک های 1024بایت ، از گزینه k- به صورت زیر استفاده کنید.

```
$ sudo df -k
```

خروجی دستور

```
/ dev / cciss / c0d0p2 78361192 23187212 51129216 32 ٪/
/ dev / cciss / c0d0p5 24797380 22273432 1243972 95 ٪/ home
/ dev / cciss / c0d0p3 29753588 25503792 2713984 91 ٪/ data
/ dev / cciss / c0d0p1 295561 21531 258770 8 ٪/ boot
tmpfs 257476 0 257476 0 ٪/ dev / shm
```

نمایش اطلاعات سیستم فایل درMB

برای نمایش اطلاعات تمام استفاده از سیستم فایل در MB مگا بایت از گزینه به عنوان " m- " استفاده کنید.

```
$ sudo df -m
```

خروجی دستور

```
/ dev / cciss / c0d0p2 76525 22644 49931 32 ٪/
/ dev / cciss / c0d0p5 24217 21752 1215 95 ٪/ home
/ dev / cciss / c0d0p3 29057 24907 2651 91 ٪/ data
/ dev / cciss / c0d0p1 289 22 253 8 ٪/ boot
tmpfs 252 0 252 0 ٪/ dev / shm
```

نمایش اطلاعات سیستم فایل درGB

برای نمایش اطلاعات تمام آمار سیستم فایل در GBگیگابایت از گزینه به عنوان ' df -h ' استفاده کنید.

```
$ sudo df -h
```

```
/ dev / cciss / c0d0p2 75G 23G 49G 32 ٪/
/ dev / cciss / c0d0p5 24G 22G 1.2G 95 ٪/ home
```

```
/ dev / cciss / c0d0p3 29G 25G 2.6G 91 ٪/ data
/ dev / cciss / c0d0p1 289M 22M 253M 8 ٪/ boot
tmpfs 252M 0 252M 0 ٪/ dev / shm
```

نمایش سیستم فایل Inodes

با استفاده از سوئیچ " i- " اطلاعاتی از تعداد inode ها و درصد آنها برای سیستم فایل نمایش داده خواهد شد.

```
$ sudo df -i
```

نمایش سیستم فایل نوع

اگرتمام خروجی های دستورات بالا را مشاهده می کنید، خواهید دید که هیچ نوع سیستم فایل وجود ندارد که در نتایج ذکر شده باشد.برای بررسی نوع سیستم فایل خود از گزینه ' T ' استفاده کنید.این نوع سیستم فایل را همراه با سایر اطلاعات نمایش می دهد.

```
$ sudo df -T
```

خروجی دستور

```
/ dev / cciss / c0d0p2 ext3 78361192 23188812 51127616 32 ٪/
/ dev / cciss / c0d0p5 ext3 24797380 22273432 1243972 95 ٪home
/ dev / cciss / c0d0p3 ext3 29753588 25503792 2713984 91 ٪/ data
/ dev / cciss / c0d0p1 ext3 295561 21531 258770 8 ٪/ boot
```

تعیین نوع خاص سیستم فایل

اگر می خواهید نوع خاصی از سیستم فایل را نمایش دهید، از گزینه ' t- ' استفاده کنید.به عنوان مثال ، دستور زیر فقط سیستم فایل ext3را نمایش می دهد.

```
$ sudo df -t ext3
```

I

خروجی دستور I

```
/ dev / cciss / c0d0p2 78361192 23190072 51126356 32 ٪/
/ dev / cciss / c0d0p5 24797380 22273432 1243972 95 ٪/ home
/ dev / cciss / c0d0p3 29753588 25503792 2713984 91 ٪/ data
/ dev / cciss / c0d0p1 295561 21531 258770 8 ٪/ boot
```

مثالهای بیشتر

a– : در صورت نیاز به نمایش تمام سیستم های پرونده همراه با آنهایی که اندازه بلوک آنها صفر است ، از گزینه **a-** با df استفاده کنید.

```
$ df -a
```

h– : این امر برای نمایش دستور df در خروجی با فرمت قابل خواندن توسط انسان استفاده می شود.

```
$ df -h kt.txt
```

k– : این اطلاعات و استفاده از سیستم فایل را در بلوک های ۱ K نمایش می دهد.

```
$ df -k kt.txt
```

استفاده از گزینه **total–** برای تولید ستون های اندازه ، استفاده شده و در دسترس در خروجی استفاده می شود.

```
$ df --total
```

T– : با کمک این گزینه می توانید نوع مربوط به سیستم فایل را همانطور که نشان داده شده مشاهده کنید.

```
$ df -T kt.txt
```

t– : این مورد هنگامی مورد استفاده قرار می گیرد که شما بخواهید اطلاعات مربوط به دیسک سیستم های پرونده فقط دارای نوع خاصی باشد.

```
$ df -t squashfs
```

x– : اکنون ، شما همچنین می توانید به df بگویید که اطلاعات استفاده از دیسک را در تمام سیستم های پرونده نمایش دهد ، به جز موارد خاص با کمک گزینه **x-** .

```
$ df -x squashfs
```

i– Using .8 : از این گزینه برای نمایش اطلاعات **inode** در خروجی استفاده می شود.

```
$ df -i kt.txt
```

sync- - : به طور پیش فرض ، دستور df با گزینه **no -sync- -** خروجی تولید می کند که قبل از بازنشر اطلاعات استفاده ، تماس سیستم همگام سازی را انجام نمی دهد. اکنون می توانیم از گزینه sinc- - استفاده کنیم که همگام سازی را به نتیجه می رساند و خروجی کاملاً به روز است.

```
$  df  - -sync
```

l– : وقتی دستور df را اجرا می کنیم ، به طور پیش فرض این سیستم پرونده های نصب شده در خارج از کشور را نشان می دهد که شامل آنهایی هستند که از سرورهای NFS یا Samba خارجی استفاده می کنند. ما می توانیم اطلاعات این سیستم های پرونده خارجی را از خروجی پنهان کنیم با **نحو** گزینه l- که در زیر نشان داده شده است.

```
$ df -l
```

۳۸.دستور du

(Disk Usage) " du "یک دستورالعمل استاندارد linux است که برای بررسی اطلاعات استفاده دیسک از فایل ها و فهرست ها در یک ماشین استفاده می شود.دستور duدارای گزینه های پارامترهای بسیاری است که می تواند برای نتایج در بسیاری از فرمت ها استفاده شود.دستور duهمچنین فایل ها و اندازه های فهرست را به صورت بازگشتی نمایش می دهد.

```
[ root@ hosein  ] # du / home /  hosein
```

```
40 / home /  hosein  / downloads
4 /home/ hosein /.mozilla/plugins
4 /home/ hosein /.mozilla/extensions
12 /home/ hosein /.mozilla
12 /home/ hosein /.ssh
689112 /home/ hosein /Ubuntu-12.10
689360 / home /  hosein
```

خروجی دستور بالا تعداد بلوک های دیسک را در فهرست home/hosein/ همراه با زیر شاخه های آن نمایش می دهد. با استفاده از گزینه " h- " با دستور " du " در "فرم قابل خواندن برای انسان" نتایج حاصل میشود.یعنی می توانید اندازه های Bytes، Kilobytes، Megabytes، Gigabytesو غیره را ببینید

```
[ root@ hosein  ] # du -h / home /  hosein
```

```
40K / home /  hosein  / downloads
```

```
4.0K /home/ hosein /.mozilla/plugins
4.0K /home/ hosein /.mozilla/extensions
12K /home/ hosein /.mozilla
```

برای دریافت خلاصه ای از حجم کل حجم دیسک فهرست از گزینه “ s- ” به صورت زیر استفاده کنید.

```
[ root@ hosein ] # du -sh / home / hosein
```

```
674M / home / hosein
```

استفاده از گزینه “ a- ” با دستور “ du ” استفاده از دیسک از تمام فایل ها و فهرست ها را نمایش می دهد.

```
[ root@ hosein ] # du-a / home / hosein
```

```
4 /home/ hosein /.bash_logout
12 /home/ hosein /downloads/uploadprogress-1.0.3.1.tgz
24 /home/ hosein /downloads/Phpfiles-org.tar.bz2
40 / home / hosein / downloads
12 /home/ hosein /uploadprogress-1.0.3.1.tgz
4 /home/ hosein /.mozilla/plugins
4 /home/ hosein /.mozilla/extensions
12 /home/ hosein /.mozilla
4 /home/ hosein /.bashrc
689108 /home/ hosein /Ubuntu-12.10/ubuntu-12.10-server-i386.iso
689112 /home/ hosein /Ubuntu-12.10
689360 / home / hosein
```

با استفاده از “ a- ” گزینه همراه با “ h- ” استفاده از دیسک از تمام فایل ها و فهرست ها را در فرمت قابل نمایش برای انسان[1] نمایش می دهد.خروجی زیر آسان تر است زیرا فایلی را در کیلوبایت ، مگابایت و غیره نشان می دهد.

```
[ root@ hosein ] # du -ah / home / hosein
```

```
4.0K /home/ hosein /.bash_logout
12K /home/ hosein /downloads/uploadprogress-1.0.3.1.tgz
24K /home/ hosein /downloads/Phpfiles-org.tar.bz2
40K / home / hosein / downloads
12K /home/ hosein /uploadprogress-1.0.3.1.tgz
4.0K /home/ hosein /.mozilla/plugins
4.0K /home/ hosein /.mozilla/extensions
12K /home/ hosein /.mozilla
4.0K /home/ hosein /.bashrc
673M /home/ hosein /Ubuntu-12.10/ubuntu-12.10-server-i386.iso
673M /home/ hosein /Ubuntu-12.10
674M / home / hosein
```

find استفاده از درایو یک درخت فهرست با زیرمجموعه آن در بلوک های کیلوبایت.از “ k- ” استفاده کنید (نمایش اندازه در واحد 1024بایت)

```
[ root@ hosein ] # du-k / home / hosein
40 / home / hosein / downloads
4 /home/ hosein /.mozilla/plugins
4 /home/ hosein /.mozilla/extensions
```

[1] human read

```
12 /home/ hosein /.mozilla
12 /home/ hosein /.ssh
689112 /home/ hosein /Ubuntu-12.10
689360 / home /  hosein
```

برای به دست آوردن خلاصه ای از فضای استفاده از سیستم فایل ها همراه با زیر شاخه های آن در حالت مگابایت تنها از گزینه " mh- "به صورت زیر استفاده کنید.گزینه " m- " بلوک های موجود در واحد MBرا شمارش می کند و " h- " برای فرمت قابل خواندن انسان است.

```
[ root@ hosein  ] # du-mh / home /  hosein
```

```
40K / home /  hosein  / downloads
4.0K /home/ hosein /.mozilla/plugins
4.0K /home/ hosein /.mozilla/extensions
12K /home/ hosein /.mozilla
12K /home/ hosein /.ssh
673M /home/ hosein /Ubuntu-12.10
674M / home /  hosein
```

گزینه " c- " یک فضای استفاده کل دیسک را محاسبه و در آخرین خط را نشان می دهد.

```
[ root@ hosein  ] # du -ch / home /  hosein
```

```
40K / home /  hosein  / downloads
4.0K /home/ hosein /.mozilla/plugins
4.0K /home/ hosein /.mozilla/extensions
12K /home/ hosein /.mozilla
12K /home/ hosein /.ssh
673M /home/ hosein /Ubuntu-12.10
674M / home /  hosein
مجموع ۶۷۴ مگابایت
```

دستور زیر مقدار استفاده تمام فایل ها و فهرست ها را نمایش می دهد، اما فایل هایی را که با الگوی داده شده مطابقت دارد، حذف می کند.دستور زیر محتویات فایل های " txt. " را محاسبه نمی کند.

```
[ root@ hosein  ] # du  -ah  -exclude = "* .txt"
```

```
/ home /  hosein  4.0K /home/ hosein /.bash_logout 12K /home/ hosein /downloads/uploadprogress-1.0.3.1.tgz 24K / home / hosein /downloads/Phpfiles-org.tar.bz2 40K / home /  hosein  / downloads 12K /home/ hosein /uploadprogress-1.0.3.1.tgz 4.0K /home/ hosein /.bash_history 4.0K /home/ hosein /.bash_profile 4.0K
```

استفاده از فضای دیسک را بر اساس زمان نمایش دهید، از گزینه " time- " استفاده کنید، همانطور که در زیر نشان داده شده است.

```
[ root@ hosein  ] # du  -ha -time / home /  hosein
```

```
4.0K 2012-10-12 22:32 /home/ hosein /.bash_logout
12K 2013-01-19 18:48 /home/ hosein /downloads/uploadprogress-1.0.3.1.tgz
24K 2013-01-19 18:48 /home/ hosein /downloads/Phpfiles-org.tar.bz2
```

```
40K 2013-01-19 18:48 / home /  hosein  / downloads
12K 2013-01-19 18:32 /home/ hosein /uploadprogress-1.0.3.1.tgz
4.0K 2012-10-13 00:11 /home/ hosein /.bash_history
4.0K 2012-10-12 22:32 /home/ hosein /.bash_profile
0 2013-01-19 18:32 /home/ hosein /xyz.txt
0 2013-01-19 18:32 /home/ hosein /abc.txt
4.0K 2012-10-12 22:32 /home/ hosein /.mozilla/plugins
4.0K 2012-10-12 22:32 /home/ hosein /.mozilla/extensions
12K 2012-10-12 22:32 /home/ hosein /.mozilla
4.0K 2012-10-12 22:32 /home/ hosein /.bashrc
24K 2013-01-19 18:32 /home/ hosein /Phpfiles-org.tar.bz2
4.0K 2013-01-19 18:32 /home/ hosein /geoipupdate.sh
4.0K 2012-10-12 22:32 /home/ hosein /.zshrc
120K 2013-01-19 18:32 /home/ hosein /goaccess-0.4.2.tar.gz.1
673M 2013-01-19 18:51 /home/ hosein /Ubuntu-12.10/ubuntu-12.10-server-i386.iso
673M 2013-01-19 18:51 /home/ hosein /Ubuntu-12.10
674M 2013-01-19 18:52 / home /  hosein
```

روز بیستم

۳۹.دستور mount

همه پرونده ها در سیستم فایل لینوکس به صورت یک درخت بزرگ ریشه در " / " مرتب شده اند . این پرونده ها را می توان در دستگاه های مختلف بر اساس جدول پارتیشن شما پخش کرد ، در ابتدا فهرست والدین شما به این درخت نصب می شو. از دستور **mount** برای لیست سیستم فایل موجود در دستگاه به ساختار درخت بزرگ که در " / " ریشه دارد استفاده می شود.

ساختار دستور:

```
mount -t type device dir
```

مثال های دستور:

```
mount [-l | -h | -V]
```

```
mount -a [-fFnrsvw] [-t fstype] [-O optlist]
```

```
mount [-fnrsvw] [-o options] device|dir
```

```
mount [-fnrsvw] [-t fstype] [-o options] device dir
```

etc / fstab / معمولاً حاوی اطلاعاتی در مورد اینکه کدام دستگاه باید در کجا نصب شود وجود دارد. بیشتردستگاه‌ها توسط پرونده هایی مانند dev/sda4/ و غیره نشان داده می‌شوند اما برای‌سیستم های پرونده خاص می‌تواند متفاوت باشد .اشکال مختلف تا حدودی مورد بحث قرار می گیرند زیرا محدودیت های خاصی در هسته های مختلف دارد.

مثال های دستور:

- اطلاعات مربوط به سیستم فایل های نصب شده را نمایش می دهد:

```
vivek@vivek-X556UQK:~$ sudo mount -l -t ext4
/dev/sda3 on / type ext4 (rw,relatime,errors=remount-ro,data=ordered)
/dev/sda4 on /media/vivek type ext4 (rw,relatime,data=ordered)
/dev/sda5 on /media/vivek type ext4 (rw,relatime,data=ordered)
vivek@vivek-X556UQK:~$ sudo mount -l -t fuseblk
/dev/sda6 on /media/vivek type fuseblk (rw,relatime,user_id=0,group_id=0,allow_o
ther,blksize=4096)
vivek@vivek-X556UQK:~$
```

- سیستم های پرونده را ماونت می کند:

```
vivek@vivek-X556UQK:~$ sudo mount -l -t fuseblk
/dev/sda6 on /media/vivek type fuseblk (rw,relatime,user_id=0,group_id=0,allow_o
ther,blksize=4096)
vivek@vivek-X556UQK:~$
```

- اطلاعات نسخه را نمایش می دهد:

```
vivek@vivek-X556UQK:~$ sudo mount -V
mount from util-linux 2.27.1 (libmount 2.27.0: selinux, assert, debug)
vivek@vivek-X556UQK:~$
```

- پیاده سازی سیستم های پرونده:

```
vivek@vivek-X556UQK:~$ sudo umount /dev/sda6
vivek@vivek-X556UQK:~$ sudo umount /dev/sda5
vivek@vivek-X556UQK:~$ sudo umount /dev/sda4
vivek@vivek-X556UQK:~$ sudo mount -l -t fuseblk
vivek@vivek-X556UQK:~$ sudo mount -l -t ext4
/dev/sda3 on / type ext4 (rw,relatime,errors=remount-ro,data=ordered)
vivek@vivek-X556UQK:~$
```

مونت کردن cd/dvd

برای نصب cdیا دی وی دی در لینوکس مراحل زیر را دنبال کنید. با قرار دادن دیسک در رایانه خود شروع کنید، سپس مراحل زیر را طی کنید: ابتدا سعی کنید از blkidدستور استفاده کنید تا ببینیددستگاه شما در چه پارتیشنی قراردارد.. معمولاً این مسیر خواهد بود /dev/sr0، اما ممکن است مال شما مسیر دیگری باشد.

بعد، یک نقطه اتصال برای جایی که می‌خواهید سی‌دی را در آن نصب کنید، ایجاد کنید. یا، اگر از قبل جایی در ذهن دارید، فقط یک فهرست خالی در جایی از رایانه خود انتخاب کنید.

```
$ sudo mkdir /mnt/cdrom
```

اکنون می توانیم از دستور mount برای نگاشت فایل دستگاه به فهرست که به تازگی ایجاد کرده ایم استفاده کنیم.

```
$ sudo mount /dev/sr0 /mnt/cdrom
```

CDیا DVD شما اکنون باید در فهرست که در آن نصب شده است قابل دسترسی باشد.

```
$ls /mnt/
$ cd /mnt/cdrom
```

همچنین می توانید با استفاده از mount دستور بدون گزینه های بیشتر تأیید کنید که mount موفقیت آمیز بوده است .

```
$ mount | grep cdrom
```

وقتی کارتان تمام شد، فقط cdرا جدا کرده و از دیسک خارج کنید.

```
$ sudo umount /mnt/cdrom
```

روز بیست و یکم

نحوه پاک کردن Swap Space

اگر میخواهید فاصله Swap را پاک کنید، دستور زیر را اجرا کنید.

```
# swapoff -a && swapon -a
```

۴۰.دستور mkswap

شرح دستور

دستور mkswapدر سیستم‌های لینوکس برای ایجاد یک پارتیشن یا فایل swap استفاده می‌شود. پارتیشن یا فایل swap یک فضای ذخیره‌سازی مجازی است که سیستم عامل از آن برای ذخیره داده‌های موقت استفاده می‌کند، به خصوص زمانی که حافظه اصلی (RAM)پر شده است. این فرایند به عنوان swap یا paging شناخته می‌شود.

ساختار دستور:

```
sudo mkswap <device_or_file>
```

- **sudo:**برای اجرای دستور با دسترسی روت (superuser) استفاده می‌شود.
- **mkswap:**نام خود دستور است.
- **<device_or_file>:**پارتیشن یا فایلی که می‌خواهید به عنوان swap استفاده کنید. مثلاً / dev/sdb1برای یک پارتیشن یا / swapfileبرای یک فایل.

کاربردهای دستور:

- **ایجاد پارتیشن swap:** زمانی که یک پارتیشن جداگانه برای swap در نظر گرفته شده باشد، از این دستور برای فرمت کردن آن به عنوان swap استفاده می‌شود.
- **ایجاد فایل swap:** اگر پارتیشن جداگانه‌ای برای swap وجود نداشته باشد، می‌توان یک فایل ایجاد کرده و آن را به عنوان swapفرمت کرد.
- **تغییر تنظیمات swap:** در برخی موارد، ممکن است نیاز به تغییر تنظیمات یک پارتیشن swap موجود باشد.

گزینه‌های دستور:

- **بدون گزینه:** دستور بدون هیچ گزینه‌ای، پارتیشن یا فایل مشخص شده را به عنوان swap فرمت می‌کند.
- **v, --verbose:** – اطلاعات بیشتری در مورد فرآیند فرمت کردن نمایش می‌دهد.
- **L <label>, --label=<label>:** – یک برچسب برای پارتیشن swap تعیین می‌کند.

مثال‌ها:

ایجاد پارتیشن swap بروی /dev/sdb1:

```
sudo mkswap /dev/sdb1
```

ایجاد فایل swap با نام /swapfile:

```
sudo mkswap /swapfile
```

ایجاد پارتیشن swap با برچسب "myswap":

```
sudo mkswap -L myswap /dev/sdb1
```

فعال کردن پارتیشن یا فایل swap:

پس از ایجاد پارتیشن یا فایل swap ، برای استفاده از آن باید آن را فعال کنید:

```
sudo swapon <device_or_file>
```

غیرفعال کردن پارتیشن یا فایل swap:

برای غیرفعال کردن یک پارتیشن یا فایل swap :

```
sudo swapoff <device_or_file>
```

اضافه کردن پارتیشن swap یا فایل fstab:

برای اینکه پارتیشن swap به طور خودکار در هنگام بوت شدن سیستم فعال شود، باید آن را به فایل /etc/fstab اضافه کنید.
برای مثال:

```
/swapfile swap swap defaults 0 0
```

نکات مهم:

- **اندازه swap:** اندازه swap باید متناسب با میزان رم سیستم و نیازهای برنامه‌ها باشد. معمولاً دو برابر تا چهار برابر اندازه رم توصیه می‌شود.
- **پشتیبانی از swap:** همه سیستم‌های فایل از swap پشتیبانی نمی‌کنند. مطمئن شوید که سیستم فایل پارتیشن یا فایلی که می‌خواهید به عنوان swap استفاده کنید، از swap پشتیبانی می‌کند.
- **ایجاد فایل swap:** برای ایجاد فایل swap ، ابتدا باید یک فایل با اندازه دلخواه ایجاد کنید (مثلاً با استفاده از دستور dd) و سپس آن را با mkswap فرمت کنید.
- **فعال کردن swap در هنگام بوت:** برای فعال کردن خودکار swap در هنگام بوت، باید آن را به فایل /etc/fstab اضافه کنید.

روز بیست و دوم

۴۱.دریافت اطلاعات سخت افزاری در linux

دستور Dmidecode

چگونه می‌توان از دستورDmidecode برای بازیابی اطلاعات سخت افزاری هر linux system استفاده کرد.فرض کنید ما می خواهیم اطلاعات یک system را که نیاز داریم جمع آوری کنیم.مانند memory ، بایوس[۱] و پردازنده و غیره. با استفاده از دستور Dmidecode، ما می خواهیم جزئیات را بدون باز کردن system تشخیص دهیم.دستور Dmidecodeبرای RHEL / CentOS / Fedora / Ubuntu Linuxکار می کند.

به ابزار DmidecodeDMIبرخی می گوید (SMBIOS) که جدولی، برای جمع آوری داده ها و نمایش اطلاعات مفید system مانند اطلاعات سخت افزاری ، شماره سریال و نسخه BIOS، پردازنده و غیره در فرمت قابل خواندن برای انسان است. می توانید از دستور root برای اجرای دستور dmidecode استفاده کنید.

در زیر نمونه خروجی دستور دستور Demidecode است.

```
# dmidecode
```

خروجی دستور :

```
اصلاح نسخه(2.3 -> 2.31) SMBIOS .
SMBIOS 2.3 موجود است
45 سازه اشغال ۱۶۴۲ بایت.
جدول در x000E0010.
دسته x0000، نوع0 DMI ، ۲۰بایت
اطلاعاتBIOS
فروشنده: فونیکس فن آوری های محدود
نسخه: ۶,۰۰
تاریخ انتشار: ۲۰۰۶/۰۶/۱۲
```

[۱] bios

```
آدرس: xE78A0۰
حجم اجرا: ۱۰۰۱۹۲ بایت
ROM حجم: ۶۴ کیلوبایت
مشخصات:
ISA پشتیبانی می شود
PCI پشتیبانی می شود
PC Card (PCMCIA) پشتیبانی می شود
PNP پشتیبانی می شود
APM پشتیبانی می شود
بایوس قابل ارتقا است
سایه BIOS مجاز است
پشتیبانی ESCD در دسترس است
میراث USB پشتیبانی می شود
باتری هوشمند پشتیبانی می شود
مشخصات بوت BIOS پشتیبانی می شود
```

دریافت انواع DMI

اطلاعات سخت افزاری خاص system ارائه می دهد. Dmidecodeبا گزینه های ' t- ' یا ' type- ' و ' Id ' به ما اطلاعات دقیق ارائه می دهد. id اطلاعات ماژول memory را به ما نشان می دهد.

```
$ sudo dmidecode -t 6
```

```
اصلاح نسخه(2.3 -> 2.31) SMBIOS .
SMBIOS 2.3 موجود است
دسته ۰x0009، نوع DMI 6 ، ۱۲بایت
اطلاعات ماژول memory
تعیین سوکت: سوکت RAM ره ۰
اتصالات بانکی: ۱ ۰
سرعت فعلی: ناشناخته است
نوع:EDO DIMM
اندازه install شده: ۱۰۲۴ مگابایت (اتصال تک بانکی)
```

در زیر جزئیات نوع DMIوجود دارد. برای مثال ، برای به دست آوردن اطلاعات کش[1] در system، می توانید دستور زیر را اجرا کنید.

```
$ sudo dmidecode -t cache
# dmidecode 2.11
```

دریافت اطلاعات memory

چگونه اطلاعات memory در system را دریافت کنیم وسیستم چقدر memory پشتیبانی می‌کند؟دستور زیر نشان می دهد که system می تواند حداکثر 4 گیگابایت RAMرا پشتیبانی کند.

```
$ sudo dmidecode -t 16
```

[1] cache memory

دریافت اطلاعات BIOS

برای دریافت اطلاعات BIOSsystem، دستور زیر را با گزینه ’t-‘اجرا کنید.

```
$ sudo dmidecode -t BIOS
```

PNP پشتیبانی می شود
APM پشتیبانی می شود
بایوس قابل ارتقا است
سایه BIOS مجاز است
پشتیبانی ESCD در دسترس است
میراث USB پشتیبانی می شود
باتری هوشمند پشتیبانی می شود
مشخصات بوت BIOS پشتیبانی می شود
.5چگونه تولید کننده، مدل و ره سریال را دریافت کنم؟

برای دریافت اطلاعات مربوط به سازنده ، مدل و شماره سریال system، از دستور زیر استفاده کنید، همانطور که در زیر نشان داده شده است.

```
$ sudo dmidecode -t system
```

همیشه یک تجربه خوب برای دانستن اجزای سخت افزاری linux system در حال اجرا ، این کمک می کند تا با مشکلات سازگاری در هنگام نصب بسته ها، درایورهای system خود، مقابله کنید. بنابراین در این نکات و ترفندها، ما باید به برخی از دستورات مفید نگاه کنیم که می تواند به در استخراج اطلاعات در مورد system عامل linux و اجزای سخت افزاری کمک کند.

روز بیست و سوم

بدست آوردن اطلاعات سیستم

برای شناختن نام system فقط می توانید از دستور unameاستفاده کنید .

```
hosein @ hosein ~ $ uname
```

برای مشاهده نام host شبکه خود، از کلید "n-"با دستور uname استفاده کنید، همانطور که نشان داده شده است.

```
hosein @ hosein ~ $ uname -n
```

برای دریافت اطلاعات مربوط به هسته[1] نسخه، از کلید 'v-'استفاده کنید.

```
hosein @ hosein ~ $ uname -v
# 64-ubuntu SMP Mon Sep 22 21:28:38 UTC 2014
```

برای دریافت اطلاعات در مورد انتشار هسته خود، از کلید 'r-'استفاده کنید.

```
hosein @ hosein ~ $ uname -r
```

```
3.13.0-37 public
```

برای چاپ نام سخت افزار دستگاه خود، از کلید "m-"استفاده کنید:

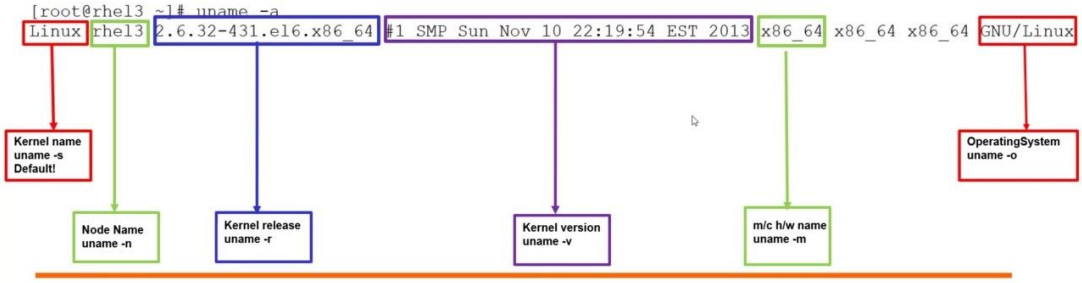
```
hosein @ hosein ~ $ uname-m
```

x86_64

تمام این اطلاعات را می توان در یک بار با اجرای دستور uname -aبه صورت زیر نشان داد.

```
hosein @ hosein ~ $ uname -a
linux hosein .com 3.13.0-37- public ٦٤ # -ubuntu SMP monday ٢٢ november٢١:٢٨:٣٨ UTC
2014 x86_64 x86_64 x86_64 gnu / linux
```

به صورت کلی



بدست آوردن اطلاعات سخت افزاری

در اینجا می‌توانید از ابزارlshw برای جمع آوری اطلاعات وسیع در مورد اجزای سخت افزاری خود مانند پردازنده ، دیسک ، memory ، کنترل کننده هایUSB و غیره استفاده کنید. lshw یک ابزار نسبتا کوچک است و چندین گزینه وجود دارد که می توانید با آن در هنگام استخراج اطلاعات استفاده کنید.اطلاعات ارائه شده توسط lshw جمع آوری شده از فایل های مختلف / proc تشکیل شده است.

توجه داشته باشید: فراموش نکنید که دستور lshwتوسط superuser (root) یا *sudo*کاربر اجرا می شود. می توانید خلاصه ای از اطلاعات سخت افزار خود را با استفاده از گزینه – shortچاپ کنید.

[1] kernel

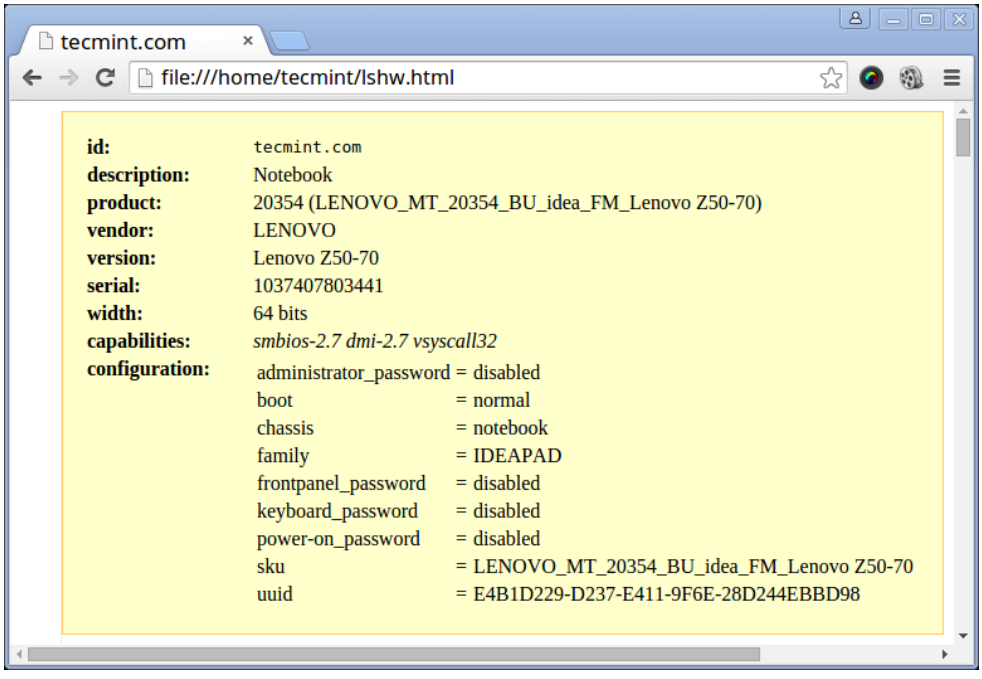
```
hosein @ hosein ~sudo lshw -short
```

بدست آوردن کلاس دستگاه H / W

```
system۲۰۳۵۴ (LENOVO_MT_20354_BU_idea_FM_Lenovo Z50-70)
/ 0 bus Lancer 5A5
/ 0/0 memory۱۲۸ KiB BIOS
/ 0/4 پردازنده اینتل (R) هسته (TM) i5-4210U CPU @ 1.70GHz
/ 0/4 / b memory۳۲ KiB L1 memory cache
/ 0/4 / c memory۲۵۶ KiB L2 memory
/ 0/4 / d memory۳ MiB L3 memory cache
/ 0 / memory memory۳۲ KiB L1
/ 0/12 memory۸ GiB memory system
/ 0/12/0 memory DIMM [empty]
/ 0/12/1 memory DIMM [empty]
/ 0/12/2 memory۸ GiB SODIMM DDR3 ۰٫۶) مگاهرتز ۱۶۰۰ همزمان ns)
/ 0/12/3 memory DIMM [empty]
/ 0/100 کنترل پلیر Haswell-ULT DRAM
/ 0/100/2 نمایشگرVGA مجتمع Haswell-ULT
/ 0/100/3 چند رسانه ای کنترل Haswell-ULT HD صوتی
```

اگر می خواهید خروجی را به عنوان یک فایل HTML ایجاد کنید، می توانید از گزینه html- استفاده کنید.

```
hosein @ hosein ~ $ sudo lshw -html> lshw.html
```



۴۲.بدست آوردن اطلاعات CPU

برای مشاهده اطلاعات مربوط به CPU خود، از دستور lscpu استفاده کنید چون اطلاعات مربوط به architect CPU مانند تعداد CPU ها، هسته ها، مدل CPU family، memory های CPU ، موضوعات و غیره را از sysfsو / proc /cpuinfo نشان می دهد.

```
hosein @ hosein ~ $ lscpu
```

```
architect: x86_64
bit ٦٤ ،protocolCPU: 32 bit
Order Byte: Little Endian
پردازنده (ها): ٤
```

روز بیست و چهارم

۴۳.بدست آوردن اطلاعات مربوط به بلوک ها

دستگاه های بلوکی دستگاه های ذخیره سازی مانند هارد دیسک، درایوهای فلش و غیره است. دستورlsblk برای گزارش اطلاعات مربوط به دستگاه های بلوک به شرح زیر استفاده می شود.

```
hosein @ hosein ~ $ lsblk
```

```
NAME MAJ: MIN MIN SIZE type RO MOUNTPOINT
sda 8: 0 0 931.5G 0
 ├─sda1 8: 1 0 1000M 0
 ├─sda2 8: 2 0 260M 0 part / boot / efi
 ├─sda3 8: 3 0 1000M 0
 ├─sda4 8: 4 0 128M 0
 ├─sda5 8: 5 0 557.1G 0
 ├─sda6 8: 6 0 25G 0
 ├─sda7 8: 7 0 14.7G 0
 ├─sda8 8: 8 0 1M 0
 ├─sda9 8: 9 0 324.5G 0   /
 └─sda10 8:10 0 7.9G 0   [SWAP]
sr0  0 1 1024M 0 rom
```

اگر می خواهید تمام دستگاه های بلوک را در system خود مشاهده کنید ، گزینه – aرا وارد کنید.

```
hosein @ hosein ~ $ lsblk -a
```

```
NAME MAJ: MIN MIN SIZE نوع RO MOUNTPOINT
sda 8: 0 0 931.5G 0
 ├─sda1 8: 1 0 1000M 0
 ├─sda2 8: 2 0 260M 0 part / boot / efi
 ├─sda3 8: 3 0 1000M 0
 ├─sda4 8: 4 0 128M 0
 ├─sda5 8: 5 0 557.1G 0
 ├─sda6 8: 6 0 25G 0
 ├─sda7 8: 7 0 14.7G 0
 ├─sda8 8: 8 0 1M 0
 ├─sda9 8: 9 0 324.5G 0   /
```

۴۴.بدست آوردن اطلاعات پردازنده

دریافت اطلاعات CPU با استفاده از دستورcat

می توانید به سادگی اطلاعات مربوط بهCPU system خود را با مشاهده محتویات فایل /cpuinfo / proc با کمک دستورcat به صورت زیر مشاهده کنید:

```
$ cat / proc / cpuinfo
```

هسته ها را نشان می دهد

```
$ cat / proc / cpuinfo | grep 'core id' #
```

دستور lscpuاطلاعات architect پردازنده را از sysfsو / cpuinfo / proc چاپ می کند همانطور که در زیر نشان داده شده است:

```
$ lscpu
```

معماری پردازنده[۱] را نشان میدهد

```
cpuid
```

دستور cpuidاطلاعات کاملی از CPU جمع آوری شده از دستور CPUID را دریافت می‌کند و همچنین مدل دقیق پردازنده های x86از آن اطلاعات را نمایان می کند. اطمینان حاصل کنید قبل از اجرا آن را نصب کنید.

```
$ sudo apt install cpuid
$ sudo yum install cpuid
$ sudo dnf install cpuid
```

پس از install، cpuidرا اجرا کنید تا اطلاعات مربوط به پردازنده x86 را جمع آوری کنید.

```
$ cpuid
```

[۱] cpu

روز بیست و پنجم

۴۵.دستور lshw

lshw ابزار کمکی برای جمع آوری اطلاعات عمیق در تنظیمات سخت افزاری کامپیوتر است. می توانید از گزینه C– برای انتخاب کلاس سخت افزار، CPU در این مورد استفاده کنید:

```
$ sudo lshw -C CPU
```

۴۶.دستور nproc

دستور nproc برای نشان دادن تعداد پردازنده موجود در رایانه استفاده می شود. دستور nproc در لینوکس با گزینه‌های مختلفی قابل استفاده است. در زیر توضیح و مثال‌هایی برای هر یک از گزینه‌های nproc آورده شده است:

گزینه a– یا all––

این گزینه تعداد هسته‌های پردازنده مجازی (virtual CPU cores)را نشان می‌دهد.

```
nproc -a
```

گزینه i– یا ignore––

این گزینه تعداد هسته‌های پردازنده مجازی را نشان می‌دهد، اما اگر فرایندهای اجرایی محدودیت‌های مرتبط با تعداد هسته‌ها داشته باشند، این محدودیت‌ها را نادیده می‌گیرد.

```
nproc -i
```

گزینه N– یا allcores––

این گزینه تعداد همه هسته‌های پردازنده را نشان می‌دهد، حتی اگر همه آن‌ها فعال نباشند.

```
nproc -N
```

گزینه p– یا phys––

این گزینه تعداد هسته‌های پردازنده فیزیکی را نشان می‌دهد.

```
nproc -p
```

گزینه v– یا virtual––

این گزینه تعداد هسته‌های پردازنده مجازی را نشان می‌دهد.

```
nproc -v
```

نمایش همه هسته‌های پردازنده:

```
nproc -a
```

روز بیست و ششم

۴۷.دستور info

دستور **info** مستندات را در قالب **info** می خواند . این info دقیق را برای یک فرمان در مقایسه با صفحه man به شما می دهد. این صفحات با استفاده از ابزار **texinfo** ساخته می شوند ، به همین دلیل می تواند با صفحات دیگر پیوند برقرار کند ، منوها و ناوبری آسان ایجاد کند. یکی دیگر از ابزارهای مفیدی که می توانید برای کسب اطلاعات بیشتر در مورد دستورات لینوکس از آن استفاده کنید، دستور اطلاعات است. info اسناد را در قالب اطلاعات می خواند (فرمت خاصی که معمولاً از منبع Texinfo تولید می شود). صفحات اطلاعات معمولاً اطلاعات دقیق تری را در مورد یک فرمان نسبت به صفحات مرد مربوطه آن ارائه می دهند. اطلاعات همچنین به پیمایش و پیوند بین صفحات اجازه می‌دهد.

ساختار دستور:

info فهرست منو [...] گزینه [

گزینه های دستور:

- **a، –all** : از همه راهنماهای منطبق استفاده می کند.
- **k، –apropos = STRING** : STRING را در تمام شاخص های همه راهنما جستجو می کند.
- **d، –directory = DIR** : این DIR را به INFOPATH اضافه می کند.
- **f، –file = MANUAL** : این دفترچه راهنما را برای بازدید مشخص می کند.
- **h، –help** : این راهنما را نمایش می دهد و از آن خارج می شود.
- **n، –node = NODENAME** : این گره ها را در پرونده info ی که برای اولین بار بازدید شده مشخص می کند.
- **o، –output = FILE** : گره های انتخاب شده را به FILE منتقل می کند.
- **O، –show-options ،–usage** : به گره گزینه های خط فرمان می رود.
- **v، –variable VAR = VALUE** : مقدار متغیری را به متغیر info ی VAR اختصاص می دهد.
- **–version** info: نسخه را نمایش می دهد و از آن خارج می شود.
- **w، –Where ،–location** : این مکان فیزیکی فایل info را چاپ می کند.

مثال ها:

- **a:** از همه راهنماهای منطبق استفاده می کند و آنها را برای یک دستور خاص نمایش می دهد.

```
info -a cvs
```

```
Node: File names matching 'cvs'

Info File Index
***************

File names that match 'cvs':

* Menu:

*     1: (/usr/share/info/cvs.info.gz).
*     2: (*manpages*)cvs.
```

- **k** این STRING را در تمام شاخص های همه راهنما جستجو می کند و سپس همان را نشان می دهد.

```
CV info  -k
```

```
linux@ubuntu:~$ info -k cvs
"(cvs)Concurrency" -- #cvs.lock, removing
"(cvs)Locks" -- #cvs.lock, technical details
"(cvs)Locks" -- #cvs.pfl, technical details
"(cvs)Backing up" -- #cvs.rfl, and backups
"(cvs)Concurrency" -- #cvs.rfl, removing
"(cvs)Locks" -- #cvs.rfl, technical details
"(cvs)Locks" -- #cvs.tfl
"(cvs)Concurrency" -- #cvs.wfl, removing
"(cvs)Locks" -- #cvs.wfl, technical details
"(cvs)Specifying a repository" -- .bashrc, setting CVSROOT in
"(cvs)Specifying a repository" -- .cshrc, setting CVSROOT in
"(cvs)~/" -- .cvsrc file
"(cvs)Specifying a repository" -- .profile, setting CVSROOT in
"(cvs)Specifying a repository" -- .tcshrc, setting CVSROOT in
"(cvs)Repository" -- /usr/local/cvsroot, as example repository
"(cvs)Working directory storage" -- Base directory, in CVS directory
"(cvs)Working directory storage" -- Baserev file, in CVS directory
"(cvs)Working directory storage" -- Baserev.tmp file, in CVS directory
"(cvs)BUGS" -- Bugs in this manual or CVS
"(cvs)Compatibility" -- Compatibility, between CVS versions
"(cvs)config" -- config, in CVSROOT
"(cvs)What is CVS?" -- Contributors (CVS program)
"(cvs)What is CVS?" -- Credits (CVS program)
"(cvs)Watches Compatibility" -- CVS 1.6, and watches
"(cvs)Structure" -- CVS command structure
"(cvs)CVS in repository" -- CVS directory, in repository
"(cvs)Working directory storage" -- CVS directory, in working directory
"(cvs)Password authentication server" -- CVS passwd file
"(cvs)What is CVS?" -- CVS, history of
"(cvs)What is CVS?" -- CVS, introduction to
"(cvs)Compatibility" -- CVS, versions of
"(cvs)Working directory storage" -- CVS/Base directory
"(cvs)Working directory storage" -- CVS/Baserev file
"(cvs)Working directory storage" -- CVS/Baserev.tmp file
"(cvs)Working directory storage" -- CVS/Entries file
"(cvs)Working directory storage" -- CVS/Entries.Backup file
"(cvs)Working directory storage" -- CVS/Entries.Log file
```

-d : را به INFOPATH اضافه می کند و همچنین همان را نمایش می دهد.

```
info -d cvs
```

```
File: dir,      Node: Top,      This is the top of the INFO tree.

This is the Info main menu (aka directory node).
A few useful Info commands:

  'q' quits;
  'H' lists all Info commands;
  'h' starts the Info tutorial;
  'mTexinfo RET' visits the Texinfo manual, etc.

* Menu:

Basics
* Common options: (coreutils)Common options.
* Coreutils: (coreutils).       Core GNU (file, text, shell) utilities.
* Date input formats: (coreutils)Date input formats.
* Ed: (ed).                     The GNU line editor
* File permissions: (coreutils)File permissions.
                                Access modes.
* Finding files: (find).        Operating on files matching certain criteria.

C++ libraries
* autosprintf: (autosprintf).   Support for printf format strings in C++.

Compression
* Gzip: (gzip).                 General (de)compression of files (lzw).

Development
* SSIP: (ssip).                 Speech Synthesis Interface Protocol.
* Speech Dispatcher: (speech-dispatcher).
                                Speech Dispatcher.

DOS
* Mtools: (mtools).             Mtools: utilities to access DOS disks in Unix.

Editors
-----Info: (dir)Top, 264 lines --Top-----------------------------------------------------------------------------------------------------
```

-O : برای دستور خاصی به گره گزینه های خط فرمان رفته و همان را نمایش می دهد.

```
info -O cvs
```

w– : محل فیزیکی فایل info را چاپ می کند.

```
info -w cvs
```

```
linux@ubuntu:~$ info -w cvs
/usr/share/info/cvs.info.gz
linux@ubuntu:~$
```

توجه داشته باشید:

برای بررسی صفحه دستی info ، از دستور زیر استفاده کنید:

```
man info
```

برای بررسی صفحه راهنمای دستور info ، از دستور زیر استفاده کنید:

```
info -h
```

روز بیست هفتم

۴۸.دستور blkid

blkid یک ابزار خط فرمان در سیستم‌عامل‌های لینوکس است که برای شناسایی و نمایش اطلاعات مربوط به پارتیشن‌ها و دستگاه‌های بلوکی استفاده می‌شود. این اطلاعات شامل UUID ، نوع سیستم فایل، LABEL و سایر مشخصه‌های مهم است که برای مدیریت و mount کردن دستگاه‌ها بسیار مفید است.

شرح دستور

دستور blkid با اسکن دستگاه‌های بلوکی موجود در سیستم، اطلاعات مربوط به هر دستگاه را جمع‌آوری کرده و به صورت قابل خواندن نمایش می‌دهد. این اطلاعات برای ابزارهای دیگر مانند fstab ، udev و سایر ابزارهای مدیریت دستگاه‌ها استفاده می‌شود.

ساختار دستور

```
blkid [options] [DEVICE]
```

- **options:** گزینه‌های مختلفی که رفتار دستور را تغییر می‌دهند.
- **DEVICE:** دستگاه بلوکی خاصی که می‌خواهید اطلاعات آن را نمایش دهید. اگر این گزینه مشخص نشود، blkid تمام دستگاه‌های بلوکی را اسکن می‌کند.

کاربردهای دستور

- **نمایش اطلاعات دستگاه‌های بلوکی:** با اجرای ساده دستور blkid ، اطلاعات کلی تمام دستگاه‌های بلوکی نمایش داده می‌شود.
- **یافتن UUID یک دستگاه:** UUID یک شناسه منحصر به فرد برای هر دستگاه است که در فایل fstab برای mount کردن دستگاه‌ها استفاده می‌شود.
- **یافتن نوع سیستم فایل:** با استفاده از blkid می‌توانید نوع سیستم فایل هر دستگاه را به راحتی تشخیص دهید.
- **یافتن LABEL یک دستگاه:** LABEL نامی است که به یک دستگاه اختصاص داده می‌شود و برای شناسایی آن در محیط گرافیکی یا اسکریپت‌ها استفاده می‌شود.

گزینه‌های مهم دستور

- **o, --output** خروجی را به یک قالب خاص) مانند JSON ، XML) تغییر می‌دهد.
- **s, --show** لیستی از ویژگی‌های خاصی را نمایش می‌دهد) مثلاً TYPE ، LABEL، UUID).
- **L, --label** فقط دستگاه‌هایی را نشان می‌دهد که دارای LABEL مشخصی هستند.
- **U, --UUID** فقط دستگاه‌هایی را نشان می‌دهد که دارای UUID مشخصی هستند.
- **t, --type** فقط دستگاه‌هایی را نشان می‌دهد که دارای نوع سیستم فایل مشخصی هستند.

مثال‌ها:

- **نمایش اطلاعات تمام دستگاه‌های بلوکی:**

```
blkid
```

- **نمایش UUID دستگاه/dev/sda1 :**

```
blkid /dev/sda1
```

- **نمایش فقط نوع سیستم فایل تمام دستگاه‌ها:**

```
blkid -s TYPE
```

- **نمایش اطلاعات دستگاهی که دارای "LABEL "home است:**

```
blkid -L home
```

کاربردهای عملی:

- **ایجاد فایل fstab:** اطلاعاتی که توسط blkid ارائه می‌شود، برای ایجاد فایل fstab که برای mount کردن خودکار دستگاه‌ها در هنگام بوت شدن استفاده می‌شود، بسیار مفید است.
- **اسکریپت‌نویسی:** blkid می‌تواند در اسکریپت‌های مختلف برای خودکارسازی وظایف مربوط به مدیریت دستگاه‌ها استفاده شود.
- **عیب‌یابی:** در مواقعی که مشکلی در ارتباط با دستگاه‌های بلوکی وجود دارد، blkidمی‌تواند برای تشخیص علت مشکل بسیار مفید باشد.

نکته: خروجی دستور blkid معمولاً به صورت زیر است:

```
/dev/sda1: UUID="e2c87a52-a1b8-471c-b69c-6b5758c34938" TYPE="ext4"
/dev/sdb1: LABEL="home" UUID="827db6e5-4b8c-4dba-8b04-2b9bf72a67c0"
TYPE="ext4"
```

در این مثال، UUID، TYPEو LABEL هر دستگاه به وضوح مشخص شده است.

۴۹.بدست آوردن اطلاعات USB

دستور lsusb برای گزارش اطلاعات مربوط به کنترل کننده های USB و تمام دستگاه هایی که به آنها متصل است استفاده می شود.

```
hosein @ hosein ~ $ lsusb
```

```
bus ۰۰۱ device۰۰۲ : ID 8087: 8000 اینتل Corp.
bus ۰۰۱ device۰۰۱ : ID 1d6b: 0002 linux۲,۰ بنیاد hub root
bus ۰۰۳ device۰۰۱ : ID 1d6b: 0003 linux۳,۰ بنیاد hub root
Bus 002 device۰۰۵ : ID 0bda: b728 Realtek Semiconductor Corp.
bus ۰۰۲ device۰۰٤ : ID 5986: 0249 ایسر، وارز
bus ۰۰۲ device۰۰۲ : ID 045e: 00cb Microsoft Corp. Basic Optical Mouse v2.0
bus ۰۰۲ device۰۰۱ : ID 1d6b: 0002 linux پایه ۲,۰ root
```

می توانید از گزینه –vبرای تولید اطلاعات دقیق در مورد هر دستگاه USB استفاده کنید.

```
hosein @ hosein ~ $ lsusb -v
```

۵۰.بدست آوردن اطلاعات دستگاه هایPCI

دستگاه های PCI ممکن است شامل پورت هایUSB ، کارت گرافیک، آداپتورهای شبکه و غیره باشد. ابزارlspci برای تولید اطلاعات مربوط به تمامی کنترل کننده های PCI در system و همچنین دستگاه هایی که به آنها متصل است استفاده می شود.

برای چاپ اطلاعات در مورد دستگاه های PCI دستور زیر را اجرا کنید:

```
hosein @ hosein ~ $ lspci
```

از گزینه –tبرای تولید خروجی در فرمت درخت استفاده کنید.

```
hosein @ hosein ~ $ lspci -t
```

```
- [0000: 00] - + - 00.0
+0.0.0
+0.0.0
+ -14.0
+ -16.0
```

از گزینه –vبرای تولید اطلاعات دقیق در مورد هر دستگاه متصل استفاده کنید.

```
hosein @ hosein ~ $ lspci -v
```

```
00.0 00: پل میزبان: کنترل کننده DRAM های Haswell-ULT شرکت(rev 0b) Intel
زیرsystem: دستگاه3978 Lenovo
گزینه ها: bus master ، devsel fast، latency 0
```

روز بیست و هشتم

۵۱.دستور dmesg

از دستور **dmesg** به عنوان "نمایش پیام" برای بررسی بافر حلقه هسته و چاپ بافر پیام هسته استفاده می شود. خروجی این دستور شامل پیام های تولید شده توسط درایورهای دستگاه است. وقتی کامپیوتر بوت می شود ، پیام های زیادی (ورود به سیستم) هنگام راه اندازی سیستم ایجاد می شود. بنابراین می توانید همه این پیام ها را با استفاده از دستور dmesg بخوانید. محتویات بافر حلقه هسته نیز در پرونده **dmesg / log / var /** ذخیره می شود. دستور dmesg هنگامی که سیستم در هنگام راه اندازی با مشکلی روبرو می شود می تواند مفید باشد ، بنابراین با خواندن محتویات دستور dmesg می توانید واقعاً بفهمید که این مشکل در کجا رخ داده است (زیرا مراحل زیادی در توالی راه اندازی سیستم وجود دارد.

ساختاردستور:

```
dmesg [گزینه ها]
```

گزینه های دستور:

- C – Clear- : بافر حلقه را پاک میکند.
- c –read-clear- : بافر حلقه را پس از چاپ محتوای آن پاک میکند.
- D –console-off- : چاپ پیام برای کنسول را غیرفعال میکند.
- E –console-on- : چاپ پیام برای کنسول را فعال میکند.
- F –file file- : پیام های پرونده داده شده را میخواند.
- h –help- : نمایش متن راهنما.
- k –kernel- : پیام های هسته را چاپ میکند.
- t –notime- : مهر زمانهای هسته را چاپ نمیکند.
- u –userspace- : پیام های فضای کاربر را چاپ میکند.

از آنجا که خروجی **dmesg و** دستوربسیار بزرگ است، بنابراین برای بدست آوردن اطلاعات خاص در خروجی dmesg و است، بهتر است به دستور dmesg با less استفاده نمایید.

```
dmesg |less
```

یا

```
dmesg | grep “ grep “text_to_search”
dmesg | grep “usb”
```

گزینه – tخروجی را با زمان مشخص می کند.

```
dmesg -t
```

مثال های دستور:

۱. گزینه –C یا --clear:

– این گزینه بافر حلقه (ring buffer)پیام‌های کرنل را پاک می‌کند.

```
dmesg -C
```

۲. گزینه –c یا --read-clear:

– این گزینه پیام‌های کرنل را چاپ می‌کند و سپس بافر حلقه را پاک می‌کند.

```
dmesg -c
```

۳. گزینه –D یا --console-off:

– این گزینه چاپ پیام‌ها برای کنسول را غیرفعال می‌کند.

```
dmesg -D
```

۴. گزینه –E یا --console-on:

– این گزینه چاپ پیام‌ها برای کنسول را فعال می‌کند.

```
dmesg -E
```

۵. گزینه –F یا --file file:

– این گزینه پیام‌های موجود در فایل مشخص شده را نمایش می‌دهد.

```
dmesg -F /var/log/dmesg.log
```

۶. گزینه –h یا --help:

– این گزینه راهنمای دستور dmesg را نمایش می‌دهد.

```
dmesg -h
```

۷. گزینه –k یا --kernel:

– این گزینه پیام‌های هسته را چاپ می‌کند.

```
dmesg -k
```

۸. گزینه –t یا --notime:

– این گزینه مهر زمان‌های هسته را در پیام‌ها چاپ نمی‌کند.

```
dmesg -t
```

۹. گزینه –u یا --userspace:

– این گزینه پیام‌های مربوط به فضای کاربر را چاپ می‌کند.

```
dmesg -u
```

۵۲.نمایش system memory

دستور free برای بررسی فضای مورد استفاده و در فضای دسترس از memory فیزیکی و swap memory درحالت KBاستفاده می شود.دستور زیر را در زیر ببینید.

```
# free
```

```
Mem: 1021628 912548 109080 0 120368 655548
```

```
بافر / کش: ۱۳۶۶۳۲ ۸۸۴۹۹۶ + / -
تعویض: ۴۱۹۴۲۹۶ ۰ ۴۱۹۴۲۹۶
```

نمایش memory در حالت Bytes

دستور free با گزینه –b نمایش اندازه memory در Bytes

```
# free -b
```

نمایش memory در Kilo Bytes

دستور free با گزینه –k، اندازه memory را در KB کیلوبایت نشان می دهد.

```
# free-k
```

نمایش memory در مگابایت

برای مشاهده اندازه memory در مگابایت از گزینه به عنوان –m استفاده کنید.

```
# free -m
```

نمایش memory در گیگابایت

با استفاده از گزینه –g با دستور free، اندازه memory در GB گیگابایت نمایش داده می شود.

```
# free -g
```

نمایش مجموع حافظه

```
# free -t
```

غیرفعال کردن بافر

به طور پیش فرض، دستور free “ buffer adjusted “ را نشان میدهد، برای غیر فعال کردن این گزینه –o استفاده نمایید.

```
# free -o
```

نمایش وضعیت memory در زمان های مشخص

گزینه s– مورد استفاده برای بروزرسانی دستور free در فواصل زمانی منظم[1] است.به عنوان مثال ، دستور زیر دستور ree هر 5ثانیه را به روز می کند.

```
# free -s 5
```

شماره نسخه دستور free

گزینه –V، نمایش اطلاعات نسخه دستور free را نمایش می دهد.

```
# free -V
```

[1] period time

روز بیست و نهم

۵۲.دستور ss

شرح دستور

دستور ss (socket statistics)ابزاری برای نمایش اطلاعات مربوط به سوکت‌های شبکه در سیستم لینوکس است. این دستور جایگزین دستور netstatشده است و قابلیت‌های بیشتری را ارائه می‌دهد.

ساختار دستور

```
ss [options]
```

- **options:** گزینه‌های مختلفی برای فیلتر کردن و نمایش اطلاعات سوکت‌ها.

کاربردهای دستور

- نمایش اطلاعات کلی در مورد سوکت‌ها
- نمایش سوکت‌های در حال گوش‌دادن(listening sockets)
- نمایش اتصالات شبکه برقرار شده
- نمایش اطلاعات دقیق درباره سوکت‌ها) مانندUID ، inode، زمان‌سنج‌ها(
- نمایش سوکت‌های حافظه مشترک

گزینه‌های مهم دستور

- **h, --help:** نمایش کمک در مورد استفاده از دستور.
- **V, --version:** نمایش نسخه دستور.
- **H, --no-header:** حذف خط سرصفحه از خروجی.
- **Q, --no-queues:** حذف ستون‌های صف ارسال و دریافت از خروجی.
- **O, --oneline:** نمایش اطلاعات هر سوکت در یک خط.
- **n, --numeric:** نمایش آدرس‌ها و پورت‌ها به صورت عددی بدون ترجمه به نام میزبان و سرویس.
- **r, --resolve:** تلاش برای ترجمه آدرس‌های عددی به نام میزبان و سرویس.
- **a, --all:** نمایش تمام سوکت‌ها (گوش‌دادن و برقرار شده).
- **l, --listening:** نمایش فقط سوکت‌های در حال گوش‌دادن.
- **B, --bound-inactive:** نمایش فقط سوکت‌های متصل شده اما غیرفعال (بدون گوش دادن یا اتصال).
- **o, --options:** نمایش اطلاعات زمان‌سنج برای سوکت‌هایTCP .
- **e, --extended:** نمایش اطلاعات دقیق سوکت(UID ، inode، کوکی).
- **m, --memory:** نمایش اطلاعات حافظه استفاده شده توسط سوکت‌ها.

مثال‌ها:

مثال‌های کاربردی برای دستور ss

دستور ss ابزاری قدرتمند برای نمایش اطلاعات دقیق در مورد سوکت‌های شبکه است. در زیر، چندین مثال کاربردی برای گزینه‌های مختلف این دستور آورده شده است:

نمایش اطلاعات کلی

- **نمایش تمام سوکت‌ها:**

```
ss
```

- **نمایش نسخه دستور:**

```
ss -V
```

- **نمایش کمک:**

```
ss -h
```

فیلتر کردن بر اساس نوع سوکت

- **نمایش فقط سوکت‌های TCP:**

```
ss -t
```

- **نمایش فقط سوکت‌های UDP:**

```
ss -u
```

- **نمایش فقط سوکت‌های RAW:**

```
ss -w
```

نمایش وضعیت سوکت

- **نمایش سوکت‌های در حال گوش دادن:**

```
ss -l
```

- **نمایش تمام سوکت‌ها (گوش دادن و برقرار شده):**

```
ss -a
```

- **نمایش سوکت‌های متصل شده اما غیرفعال:**

```
ss -B
```

نمایش اطلاعات دقیق

- **نمایش اطلاعات دقیق سوکت‌ها (UID، inode، کوکی):**

```
ss -e
```

- **نمایش اطلاعات زمان‌سنج برای سوکت‌های TCP:**

```
ss -o
```

- **نمایش اطلاعات حافظه استفاده شده توسط سوکت‌ها:**

```
ss -m
```

قالب‌بندی خروجی

- **نمایش اطلاعات هر سوکت در یک خط:**

```
ss -O
```

- **حذف خط سرصفحه از خروجی:**

```
ss -H
```

- **حذف ستون‌های صف ارسال و دریافت:**

```
ss -Q
```

نمایش آدرس‌ها و پورت‌ها

- **نمایش آدرس‌ها و پورت‌ها به صورت عددی:**

```
ss -n
```

- **تلاش برای ترجمه آدرس‌های عددی به نام میزبان و سرویس:**

```
ss -r
```

مثال‌های ترکیبی

- **نمایش تمام سوکت‌های TCP و UDP به صورت عددی، همراه با اطلاعات دقیق و بدون سرصفحه:**

```
ss -tunpeH
```

- **نمایش سوکت‌های در حال گوش دادن روی پورت ۸۰، همراه با اطلاعات زمان‌سنج:**

```
ss -lt' sport = :80 '-o
```

نکات مهم:

- با ترکیب گزینه‌های مختلف، می‌توانید خروجی دستور `ss` را به طور دقیق تنظیم کنید.
- برای فیلتر کردن خروجی، می‌توانید از عبارات منظم استفاده کنید.
- دستور `ss` بسیار قدرتمند است و امکانات زیادی را برای تحلیل ترافیک شبکه فراهم می‌کند.
- برای اطلاعات بیشتر در مورد گزینه‌ها و خروجی دستور `ss`، می‌توانید از دستور `man ss` استفاده کنید.

مثال‌های پیشرفته:

- **یافتن اتصالات به یک سرور خاص:**

```
ss -tn' dst addr 192.168.1.100'
```

- **یافتن فرآیندهایی که از یک پورت خاص استفاده می‌کنند:**

```
ss -tulwnp' sport = :80'
```

- **بررسی اتصالات به یک سرویس خاص (مثلاً SSH):**

```
ss -tn' dport = :22'
```

با استفاده از این مثال‌ها و گزینه‌های مختلف دستور ss، می‌توانید اطلاعات جامعی در مورد وضعیت شبکه سیستم خود به دست آورید و مشکلات احتمالی را شناسایی و برطرف کنید.

روز سی ام

۵۳.دستور iw

دستور iw در لینوکس برای مدیریت رابط‌های شبکه بی‌سیم (Wi-Fi) استفاده می‌شود. این دستور به کاربر امکان می‌دهد تا اطلاعات مربوط به اتصالات Wi-Fi را مشاهده کرده، تنظیمات رابط‌های بی‌سیم را تغییر دهد و اطلاعات مربوط به شبکه‌های Wi-Fi موجود در دسترس را ببیند.

گزینه های دستور:

```
iw [ OPTIONS ] { help [ command ] | OBJECT COMMAND }

OBJECT := { dev | phy | reg }

OPTIONS := { --version | --debug }
```

مثال های دستور:

گزینه dev:

این گزینه برای مشخص کردن رابط شبکه بی‌سیم مورد نظر استفاده می‌شود.

```
iw dev wlan0 link
```

گزینه scan:

این گزینه برای اسکن شبکه‌های بی‌سیم در دسترس استفاده می‌شود.

```
iw dev wlan0 scan
```

گزینه connect:

این گزینه برای اتصال به یک شبکه بی‌سیم استفاده می‌شود.

```
iw dev wlan0 connect MyWiFiSSID
```

گزینه disconnect:

این گزینه برای قطع اتصال از یک شبکه بی‌سیم استفاده می‌شود.

```
iw dev wlan0 disconnect
```

گزینه info:

این گزینه اطلاعات کاملی از رابط بی‌سیم فعلی نمایش می‌دهد.

```
iw dev wlan0 info
```

گزینه event:

این گزینه برای نمایش رویدادهای مربوط به رابط بی‌سیم استفاده می‌شود.

```
iw dev wlan0 event
```

گزینه monitor:

این گزینه برای فعال کردن حالت مانیتور (monitor mode)برای رابط بی‌سیم استفاده می‌شود.

```
iw dev wlan0 set monitor control
```

گزینه frequency:

این گزینه برای تنظیم فرکانس بی‌سیم استفاده می‌شود.

```
iw dev wlan0 set freq 2462
```

گزینه txpower:

این گزینه برای تنظیم توان انتقال بی‌سیم استفاده می‌شود.

```
iw dev wlan0 set txpower fixed 20dBm
```

۵۴.دستور kill

فرض کنید که یک برنامه داشته باشید که پاسخی نداشته باشد چگونه از آن خلاص شوید؟ در حالی که دستور kill برای فرآیندها مورد استفاده قرار می گیرد، هدف واقعی آن ارسال *سیگنال ها* به فرآیندها است. برنامه ها (اگر به درستی نوشته شده باشند) برای سیگنال ها به سیستم عامل گوش فرا می دهند و به آنها پاسخ می دهند، اغلب اجازه می دهد تا برخی از روش های منحصر به فرد، قطع شود. دستور killمی تواند سیگنال های مختلفی را برای فرآیندها ارسال کند. تایپ کنید:

```
kill -l
```

به یک لیست از سیگنال های آن را پشتیبانی می کند. اکثر آنها نسبتا مبهم هستند، اما چندین مورد مفید هستند:

سیگنال	نام	شرح
1	**Sighup**	قطع کردن سیگنال برنامه ها می توانند برای این سیگنال گوش دهند و بر آن عمل کنند. هنگامی که ترمینال را ببندید، این سیگنال به فرآیندهای در حال اجرا در یک ترمینال ارسال می شود.
2	**Sigint**	سیگنال وقفه این سیگنال به فرآیندها داده می شود تا آنها را قطع کنند. برنامه ها می توانند این سیگنال را پردازش کنند و بر آن عمل کنند. همچنین می توانید این سیگنال را به طور مستقیم با تایپ کردن Ctrl-c در پنجره ترمینال که در آن برنامه اجرا می شود را صادر کنید.
15	**Sigterm**	سیگنال خاتمه دادن این سیگنال به فرآیندها داده می شود تا آنها را خاتمه دهد. دوباره برنامه ها می توانند این سیگنال را پردازش کنند و بر آن عمل کنند. اگر سیگنال مشخص نشده باشد، این سیگنال به طور پیش فرض توسط دستور killارسال می شود.
9	**Sigkill**	سیگنال را بکشید این سیگنال منجر به توقف روند توسط هسته لینوکس می شود. برنامه ها نمی توانند برای این سیگنال گوش کنند.

ساختار دستور:

```
kill [-signal|-s signal|-p] [-q value] [-a] [--timeout
milliseconds signal] [--] pid|name...
```

```
kill -l [number] | -L
```

مثال های دستور:

گزینه s- یاsignal-- :

برای ارسال یک سیگنال خاص به پردازه‌ها.

```
kill -s SIGTERM <PID>
```

گزینهp- :

برای ارسال سیگنال به گروه پردازه‌ها.

```
kill -p <PGID>
```

گزینهq- :

برای تنظیم مقدار اولویت برای ارسال سیگنال.

```
kill -q 15 <PID>
```

گزینهa- :

برای ارسال سیگنال به همه پردازه‌های فرزند.

```
kill -a <PID>
```

گزینهtimeout-- :

برای تنظیم زمان انتظار قبل از ارسال سیگنال.

```
kill --timeout 5000 SIGKILL <PID>
```

گزینهl- :

برای نمایش لیست سیگنال‌های موجود.

```
kill -l
```

گزینهL- :

برای نمایش لیست نام‌های سیگنال‌ها.

```
kill -L
```

روز سی و یکم

۵۵.دستور ps

شرح دستور

دستور psیکی از پرکاربردترین دستورات در سیستم‌عامل‌های یونیکس‌مانند و لینوکس است که برای مشاهده فرآیندهای در حال اجرا در سیستم استفاده می‌شود. با استفاده از این دستور می‌توانید اطلاعات مفیدی در مورد هر فرآیند مانند PID (شناسه فرآیند)، نام فرآیند، کاربر اجراکننده، استفاده از حافظه و CPU و ... به دست آورید.

ساختار دستور

ps [options] [argument]

- **options:**گزینه‌هایی هستند که رفتار دستور را تغییر می‌دهند و نوع اطلاعات نمایش داده شده را کنترل می‌کنند.
- **argument:**آرگومانی است که برای انتخاب فرآیندهای خاص استفاده می‌شود) مثلاً PID فرآیند

کاربردهای دستور

- **مشاهده لیستی از تمام فرآیندهای در حال اجرا:** با استفاده از گزینه‌های مختلف، می‌توانید لیستی از تمام فرآیندها یا فرآیندهای خاصی را مشاهده کنید.
- **یافتن PID یک فرآیند خاص:** با استفاده از نام فرآیند یا کاربر اجراکننده، می‌توانید PID فرآیند مورد نظر را پیدا کنید.
- **بررسی وضعیت و منابع مصرفی یک فرآیند:** با استفاده از گزینه‌های مختلف، می‌توانید اطلاعات دقیقی در مورد وضعیت و منابع مصرفی هر فرآیند به دست آورید.
- **پیدا کردن فرآیندهای مشکل‌ساز:** با استفاده از دستور psمی‌توانید فرآیندهایی که باعث کندی سیستم یا مشکلات دیگر می‌شوند را شناسایی کنید.

گزینه‌های مهم دستور

- **a** :نمایش تمام فرآیندها، حتی فرآیندهای متعلق به کاربران دیگر.
- **c** :نمایش نام کامل دستور(command name) .
- **e** :نمایش اطلاعات گسترده در مورد فرآیندها.
- **g** :نمایش فرآیندهای گروهی.
- **n** :نمایش نام کاربری عددی به جای نام کاربری واقعی.
- **w** :نمایش خطوط طولانی‌تر.
- **x** :نمایش فرآیندهایی که ترمینال کنترلی ندارند (مانند دیمون‌ها).
- **l** :نمایش اطلاعات طولانی در مورد فرآیندها.
- **s** :نمایش اطلاعات خلاصه در مورد فرآیندها.
- **u** :نمایش اطلاعات مربوط به کاربر.
- **v** :نمایش اطلاعات مجازی در مورد فرآیندها.
- **t tty:**نمایش فرآیندهای مرتبط با ترمینال مشخص شده.
- **X** :نمایش اطلاعات اضافی.

- **ProcessNumber** :نمایش اطلاعات مربوط به فرآیند با شناسه مشخص شده.

مثال‌های کاربردی دستور ps با گزینه‌های مختلف

دستور psابزاری قدرتمند برای مشاهده فرآیندهای در حال اجرا در سیستم‌عامل‌های یونیکس‌مانند و لینوکس است. با استفاده از گزینه‌های مختلف، می‌توانید اطلاعات دقیق و مفیدی درباره فرآیندها به دست آورید. در ادامه، چندین مثال کاربردی برای گزینه‌های مختلف این دستور آورده شده است:

نمایش کلی فرآیندها

- **نمایش تمام فرآیندها، شامل فرآیندهای کاربران دیگر:**

```
ps aux
```

- **نمایش نام کامل دستور برای همه فرآیندها:**

```
ps auxc
```

- **نمایش اطلاعات گسترده برای همه فرآیندها:**

```
ps auxe
```

فیلتر کردن فرآیندها

- **نمایش فرآیندهای مربوط به کاربر علی:**

```
ps -u ali
```

- **نمایش فرآیندهای مرتبط با ترمینال pts/0 :**

```
ps -t pts/0
```

- **نمایش فرآیندهای بدون ترمینال کنترلی (دیمون‌ها):**

```
ps aux | grep -v TTY
```

نمایش اطلاعات خاص

- **نمایش اطلاعات طولانی در مورد فرآیند با PID 1234 :**

```
ps -l 1234
```

- **نمایش اطلاعات خلاصه در مورد همه فرآیندها:**

```
ps as
```

- **نمایش اطلاعات مربوط به کاربر برای همه فرآیندها:**

```
ps aux
```

نمایش اطلاعات پیشرفته

- **نمایش اطلاعات مجازی در مورد فرآیندها:**

```
ps auxv
```

- **نمایش اطلاعات اضافی برای همه فرآیندها:**

```
ps auxX
```

ترکیب گزینه‌ها

- **نمایش نام کامل دستور، اطلاعات گسترده و فرآیندهای گروهی:**

```
ps auxeg
```

- **نمایش نام کاربری عددی و فرآیندهای مرتبط با ترمینال pts/1 :**

```
ps -nt pts/1
```

روز سی و دوم

۵۶.دستور ulimit

شرح دستور

دستور `ulimit`یک ابزار خط فرمان در لینوکس است که برای تنظیم و نمایش محدودیت‌های منابع سیستم برای یک فرآیند یا کاربر استفاده می‌شود. این محدودیت‌ها می‌توانند شامل موارد زیر باشند:

- **حافظه:** حداکثر مقدار حافظه‌ای که یک فرآیند می‌تواند استفاده کند.
- **تعداد فایل‌های باز:** حداکثر تعداد فایل‌هایی که یک فرآیند می‌تواند به طور همزمان باز کند.
- **اندازه فایل هسته:** حداکثر اندازه فایلی که در صورت وقوع خطا ایجاد می‌شود.
- **تعداد پردازش‌ها:** حداکثر تعداد پردازش‌هایی که یک کاربر می‌تواند به طور همزمان اجرا کند.

با استفاده از این دستور، می‌توان از منابع سیستم به طور مؤثرتر استفاده کرد و از بروز مشکلاتی مانند اشغال بیش از حد حافظه یا ایجاد تعداد زیادی فرآیند جلوگیری کرد.

ساختار دستور:

```
ulimit [options] [limit]
```

- **options:**گزینه‌هایی برای تنظیم نوع محدودیت و مقدار آن.
- **limit:**مقدار عددی برای محدودیت.

کاربردهای دستور:

- **نمایش محدودیت‌های فعلی:** بدون هیچ گزینه‌ای، `ulimit`محدودیت‌های فعلی را نمایش می‌دهد.
- **تنظیم محدودیت‌ها:** با استفاده از گزینه‌های مختلف، می‌توان محدودیت‌های خاصی را تنظیم کرد.
- **تنظیم محدودیت‌های نرم و سخت:** محدودیت‌های نرم (soft limits) محدودیت‌هایی هستند که سیستم به طور پیش‌فرض اعمال می‌کند و می‌توان آن‌ها را به طور موقت افزایش داد. محدودیت‌های سخت (hard limits) محدودیت‌های بالاتری هستند که حتی با افزایش موقت نیز قابل تجاوز نیستند.

گزینه‌های مهم دستور:

- **a–:** تمام محدودیت‌ها را نمایش می‌دهد.
- **c–:** محدودیت اندازه فایل هسته (core file size) را تنظیم می‌کند.
- **d–:** محدودیت اندازه بخش داده (data segment size) را تنظیم می‌کند.
- **f–:** محدودیت اندازه فایل (file size) را تنظیم می‌کند.
- **n–:** حداکثر تعداد فایل‌های باز (open files) را تنظیم می‌کند.
- **m–:** حداکثر اندازه حافظه (memory size) را تنظیم می‌کند.

- **p–**: اندازه بافر لوله (pipe size) را تنظیم می‌کند.
- **s–**: اندازه پشته (stack size) را تنظیم می‌کند.
- **t–**: حداکثر زمان پردازنده (CPU time) را تنظیم می‌کند.
- **u–**: حداکثر تعداد پردازش‌ها (user processes) را تنظیم می‌کند.
- **v–**: حداکثر اندازه حافظه مجازی (virtual memory) را تنظیم می‌کند.
- **S–**: محدودیت نرم را تنظیم می‌کند.
- **H–**: محدودیت سخت را تنظیم می‌کند.

مثال‌های کاربردی

نمایش تمام محدودیت‌ها

```
ulimit -a
```

این دستور تمام محدودیت‌های فعلی سیستم را به صورت جدول‌وار نمایش می‌دهد.

تنظیم محدودیت اندازه فایل هسته

```
ulimit -c 1024
```

این دستور حداکثر اندازه فایلی که در صورت وقوع خطا ایجاد می‌شود را به ۱۰۲۴ بلوک تنظیم می‌کند.

تنظیم محدودیت اندازه بخش داده

```
ulimit -d unlimited
```

این دستور محدودیت اندازه بخش داده را نامحدود می‌کند.

تنظیم محدودیت اندازه فایل

```
ulimit -f 2048
```

این دستور حداکثر اندازه فایلی که یک فرآیند می‌تواند ایجاد کند را به ۲۰۴۸ بلوک تنظیم می‌کند.

تنظیم حداکثر تعداد فایل‌های باز

```
ulimit -n 4096
```

این دستور حداکثر تعداد فایل‌هایی که یک فرآیند می‌تواند به طور همزمان باز کند را به ۴۰۹۶ فایل تنظیم می‌کند.

تنظیم حداکثر اندازه حافظه

```
ulimit -m 2G
```

این دستور حداکثر اندازه حافظه‌ای که یک فرآیند می‌تواند استفاده کند را به ۲ گیگابایت تنظیم می‌کند.

تنظیم اندازه بافر لوله

```
ulimit -p 16384
```

این دستور اندازه بافر لوله را به ۱۶۳۸۴ بایت تنظیم می‌کند.

تنظیم اندازه پشته

```
ulimit -s 8192
```

این دستور اندازه پشته را به ۸۱۹۲ کیلوبایت تنظیم می‌کند.

تنظیم حداکثر زمان پردازنده

```
ulimit -t 60
```

این دستور حداکثر زمان پردازنده‌ای که یک فرآیند می‌تواند استفاده کند را به ۶۰ ثانیه تنظیم می‌کند.

تنظیم حداکثر تعداد پردازش‌ها

```
ulimit -u 256
```

این دستور حداکثر تعداد پردازش‌هایی که یک کاربر می‌تواند به طور همزمان اجرا کند را به ۲۵۶ پردازش تنظیم می‌کند.

تنظیم حداکثر اندازه حافظه مجازی

```
ulimit -v 4G
```

این دستور حداکثر اندازه حافظه مجازی که یک فرآیند می‌تواند استفاده کند را به ۴ گیگابایت تنظیم می‌کند.

تنظیم محدودیت نرم و سخت

تنظیم محدودیت نرم تعداد فایل‌های باز به ۱۰۲۴ و محدودیت سخت به ۲۰۴۸

```
ulimit -S -n 1024 -H -n 2048
```

این دستور محدودیت نرم تعداد فایل‌های باز را به ۱۰۲۴ فایل و محدودیت سخت را به ۲۰۴۸ فایل تنظیم می‌کند. این بدان معناست که به طور پیش‌فرض، یک فرآیند می‌تواند حداکثر ۱۰۲۴ فایل را باز کند، اما با استفاده از برخی مکانیزم‌ها می‌توان این محدودیت را تا ۲۰۴۸ فایل افزایش داد.

توجه:

- مقادیر عددی در مثال‌ها به عنوان نمونه آورده شده‌اند و شما می‌توانید آن‌ها را با توجه به نیاز خود تغییر دهید.
- برای تنظیم محدودیت‌ها به صورت دائمی، باید فایل‌های پیکربندی مربوطه (مانند `/etc/security/limits.conf`) را ویرایش کنید.
- تنظیم محدودیت‌های بسیار پایین می‌تواند باعث ایجاد مشکلاتی در اجرای برنامه‌ها شود.

مثال عملی: فرض کنید می‌خواهید حداکثر تعداد فایل‌هایی که یک کاربر می‌تواند به طور همزمان باز کند را به ۲۰۴۸ فایل محدود کنید و همچنین می‌خواهید مطمئن شوید که هیچ فرآیندی نتواند بیش از ۴ گیگابایت حافظه استفاده کند. برای این کار می‌توانید از دستورات زیر استفاده کنید:

```
ulimit -n 2048
ulimit -m 4G
```

با اجرای این دستورات، محدودیت‌های مورد نظر برای پوسته فعلی تنظیم می‌شوند. برای اعمال این محدودیت‌ها برای تمام کاربران، باید فایل‌های پیکربندی مربوطه را ویرایش کنید.

روز سی و سه ام

۵۷.مدیریت سرویس ها و اجرای سطوح در لینوکس

systemdیک سیستم init و مدیر سیستم است که به طور گسترده به استاندارد جدیدی برای توزیع های لینوکس تبدیل شده است. به دلیل پذیرش زیاد آن، آشنایی با systemdآن به دردسر می ارزد، زیرا مدیریت سرورها را به میزان قابل توجهی آسان تر می کند. یادگیری و استفاده از ابزارها و شیاطین موجود systemdبه شما کمک می‌کند تا قدرت، انعطاف‌پذیری و قابلیت‌هایی را که ارائه می‌دهد بهتر درک کنید یا حداقل به شما کمک می‌کند تا کار خود را با حداقل دردسر انجام دهید. ابزار مدیریت مرکزی برای کنترل سیستم init است. لطفاً توجه داشته باشید که اگرچه systemdبه سیستم init پیش‌فرض برای بسیاری از توزیع‌های لینوکس تبدیل شده است، اما به طور جهانی در همه توزیع‌ها اجرا نمی‌شود. همانطور که این آموزش را دنبال می کنید، اگر ترمینال شما خطا bash: systemctl is not installedرا صادر کند، احتمالاً دستگاه شما یک سیستم init متفاوت نصب کرده است.

مدیریت سرویس ها

هدف اساسی یک سیستمinit ، مقداردهی اولیه اجزایی است که باید پس از بوت شدن هسته لینوکس راه اندازی شوند) که به طور سنتی به عنوان اجزای "Userland" شناخته می شود. (سیستم init همچنین برای مدیریت سرویس ها و دیمون ها برای سرور در هر نقطه ای که سیستم در حال اجرا است استفاده می شود. با در نظر گرفتن این موضوع، ما با برخی از عملیات مدیریت خدمات اولیه شروع خواهیم کرد. در systemd هدف اکثر اقدامات "واحدها" هستند که منابعی هستند که systemdمی دانند چگونه مدیریت کنند. واحدها بر اساس نوع منبعی که نشان می دهند دسته بندی می شوند و با فایل هایی که به عنوان فایل های واحد شناخته می شوند تعریف می شوند. نوع هر واحد را می توان از پسوند انتهای فایل استنباط کرد.برای وظایف مدیریت خدمات، واحد هدف واحدهای خدماتی خواهد بود که دارای فایل های واحد با پسوند .service. با این حال، برای اکثر دستورات مدیریت سرویس، می‌توانید .serviceپسوند را کنار بگذارید ، زیرا systemdبه اندازه کافی هوشمندانه است که بدانید احتمالاً می‌خواهید هنگام استفاده از دستورات مدیریت سرویس، روی یک سرویس کار کنید.

شروع و توقف سرویس

برای شروع یک systemdسرویس، اجرای دستورالعمل ها در فایل واحد سرویس، از startدستور استفاده کنید . اگر به عنوان یک کاربر غیر ریشه در حال اجرا هستید، باید از sudoآن استفاده کنید زیرا این روی وضعیت سیستم عامل تأثیر می گذارد:

```
sudo systemctl start application.service
```

همانطور که در بالا ذکر کردیم، systemdمی‌داند که به دنبال .service*فایل‌ها برای دستورات مدیریت سرویس می‌گردد، بنابراین این دستور به راحتی می‌تواند به صورت زیر تایپ شود:

```
sudo systemctl start application
```

اگرچه ممکن است از قالب فوق برای مدیریت عمومی استفاده کنید، اما برای وضوح، ما از .serviceپسوند برای بقیه دستورات استفاده می کنیم تا در مورد هدفی که روی آن کار می کنیم صریح باشیم.

برای متوقف کردن یک سرویس در حال اجرا، می توانید به جای آن از stopدستور استفاده کنید:

```
sudo systemctl stop application.service
```

راه اندازی مجدد و بارگذاری مجدد سرویس

برای راه اندازی مجدد یک سرویس در حال اجرا، می توانید از `restart`دستور زیر استفاده کنید:

```
sudo systemctl restart application.service
```

اگر برنامه مورد نظر بتواند فایل های پیکربندی خود را مجدداً بارگیری کند (بدون راه اندازی مجدد)، می توانید `reload`دستور شروع آن فرآیند را صادر کنید:

```
sudo systemctl reload application.service
```

اگر مطمئن نیستید که سرویس قابلیت بارگذاری مجدد پیکربندی خود را دارد، می توانید:

`reload-or-restart`دستور را صادر کنید. در صورت موجود بودن، این پیکربندی را دوباره بارگیری می‌کند. در غیر این صورت، سرویس را دوباره راه اندازی می کند تا پیکربندی جدید انتخاب شود:

```
sudo systemctl reload-or-restart application.service
```

فعال و غیرفعال کردن سرویس

دستورات بالا برای شروع یا توقف خدمات در جلسه جاری مفید هستند. برای اطلاع `systemd`از شروع خودکار سرویس ها در هنگام بوت، باید آنها را فعال کنید.

برای راه اندازی یک سرویس در هنگام بوت، از `enable`دستور زیر استفاده کنید:

```
sudo systemctl enable application.service
```

این یک پیوند نمادین از کپی سیستم از فایل سرویس (معمولاً در:

`/lib/systemd/system/` یا `/etc/systemd/system`) به مکانی روی دیسک که در آن `systemd`فایل‌های شروع خودکار را جستجو می‌کند ایجاد می‌کند (معمولاً . ما بعداً در این راهنما به این خواهیم پرداخت که هدف چیست

`/etc/systemd/system/some_target.target.wants`

برای غیرفعال کردن شروع خودکار سرویس، می توانید تایپ کنید:

```
sudo systemctl disable application.service
```

با این کار پیوند نمادینی که نشان می دهد سرویس باید به طور خودکار شروع شود حذف می کند. به خاطر داشته باشید که فعال کردن یک سرویس آن را در جلسه فعلی شروع نمی کند. اگر می خواهید سرویس را راه اندازی کنید و همچنین آن را در هنگام بوت فعال کنید، باید هر دو دستور `start`و `enable`را صادر کنید.

بررسی وضعیت سرویس

برای بررسی وضعیت یک سرویس در سیستم خود، می توانید از `status`دستور زیر استفاده کنید :

```
systemctl status application.service
```

با این کار وضعیت سرویس، سلسله مراتب cgroup و چند خط ورود اولیه به شما ارائه می شود. به عنوان مثال ، هنگام بررسی وضعیت یک سرورNginx ، ممکن است خروجی مانند زیر را مشاهده کنید:

Output

```
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2015-01-27 19:41:23 EST; 22h ago
 Main PID: 495 (nginx)
   CGroup: /system.slice/nginx.service
```

```
           ├─495 nginx: master process /usr/bin/nginx -g pid /run/nginx.pid; error_log
stderr;
           └─496 nginx: worker process
Jan 27 19:41:23 desktop systemd[1]: Starting A high performance web server and a reverse
proxy server...
Jan 27 19:41:23 desktop systemd[1]: Started A high performance web server and a reverse
proxy server.
```

این به شما یک نمای کلی از وضعیت فعلی برنامه می دهد و شما را از هر گونه مشکل و هر اقدامی که ممکن است لازم باشد مطلع می کند. همچنین روش هایی برای بررسی وضعیت های خاص وجود دارد. به عنوان مثال ، برای بررسی اینکه آیا یک واحد در حال حاضر فعال است (در حال اجرا)، می توانید از `is-active` دستور استفاده کنید :

```
systemctl is-active application.service
```

این حالت واحد فعلی را که معمولاً `active` یا `inactive`. کد خروج در صورت فعال بودن "۰" خواهد بود و تجزیه نتیجه را در اسکریپت های پوسته ساده تر می کند. برای اینکه ببینید واحد فعال است، می توانید از `is-enabled` دستور زیر استفاده کنید :

```
systemctl is-enabled application.service
```

این خروجی را نشان می دهد که آیا سرویس است `enabled` یا `disabled` بسته به پاسخ به سؤال دستور، کد خروج را روی "۰" یا "۱" تنظیم می کند. بررسی سوم این است که آیا دستگاه در وضعیت خراب است یا خیر. این نشان می دهد که مشکلی در راه اندازی واحد مورد نظر وجود داشته است:

```
systemctl is-failed application.service
```

این بازگشت `active` آن است که اگر به درستی در حال اجرا و یا `failed` اگر یک خطا رخ داده است. اگر واحد عمدا متوقف شده باشد، ممکن است برگردد `unknown` یا `inactive`. وضعیت خروج "۰" نشان می دهد که یک خرابی رخ داده است و وضعیت خروج "۱" نشان دهنده هر وضعیت دیگری است.

روز سی و چهارم

دستور systemctl

systemctl با اضافه کردن پرچم‌های اضافی، می‌توانیم بگوییم که اطلاعات مختلف را خروجی کنیم. به عنوان مثال ، برای مشاهده تمام واحدهایی که systemdبارگیری شده اند (یا تلاش کرده اند)، صرف نظر از اینکه در حال حاضر فعال هستند یا خیر، می توانید از --allپرچم مانند زیر استفاده کنید:

```
systemctl list-units --all
```

این نشان می دهد هر واحدی که systemdبارگذاری شده یا تلاش کرده است، صرف نظر از وضعیت فعلی آن در سیستم. برخی از واحدها پس از اجرا غیرفعال می شوند و برخی واحدهایی که systemdسعی در بارگیری داشتند ممکن است روی دیسک پیدا نشده باشند. می توانید از پرچم های دیگر برای فیلتر کردن این نتایج استفاده کنید. برای مثال ، می‌توانیم از --stateپرچم برای نشان دادن حالت‌هایLOAD ، ACTIVE یا SUB که می‌خواهیم ببینیم استفاده کنیم. شما باید --allپرچم را نگه دارید تاsystemctl واحدهای غیرفعال نمایش داده شوند:

```
systemctl list-units --all --state=inactive
```

یکی دیگر از فیلترهای رایج فیلتر است --type=. ما می‌توانیم بگوییمsystemctlکه فقط واحدهایی از نوع مورد علاقه ما نمایش داده می‌شوند. به عنوان مثال ، برای مشاهده فقط واحدهای خدمات فعال، می‌توانیم از موارد زیر استفاده کنیم:

```
systemctl list-units --type=service
```

فهرست کردن همه فایل های واحد

این list-unitsدستور فقط واحدهایی را نشان می دهد که systemdسعی در تجزیه و بارگذاری در حافظه داشته اند. از آنجایی که systemdفقط واحدهایی را می خواند که فکر می کند به آن نیاز دارد، لزوماً همه واحدهای موجود در سیستم را شامل نمی شود. برای مشاهده هر فایل واحد موجود در systemdمسیرها، از جمله مواردی که systemdتلاشی برای بارگیری نکرده‌اند، می‌توانید به list-unit-filesجای آن از دستور استفاده کنید:

```
systemctl list-unit-files
```

واحدها بازنمایی منابعی هستند که systemdدر مورد آن می دانند. از آنجایی systemdکه لزوماً همه تعاریف واحد را در این نما نخوانده است، فقط اطلاعات مربوط به خود فایل ها را ارائه می دهد. خروجی دارای دو ستون است: فایل واحد و وضعیت.

Output

```
UNIT FILE                              STATE
proc-sys-fs-binfmt_misc.automount      static
dev-hugepages.mount                    static
dev-mqueue.mount                       static
proc-fs-nfsd.mount                     static
proc-sys-fs-binfmt_misc.mount          static
sys-fs-fuse-connections.mount          static
sys-kernel-config.mount                static
sys-kernel-debug.mount                 static
tmp.mount                              static
var-lib-nfs-rpc_pipefs.mount           static
org.cups.cupsd.path                    enabled
```

. . .

حالت معمولاً ,static ,disabled ,enabled یا masked خواهد بود . در این زمینه، static به این معنی است که فایل واحد شامل یک install نیست که برای فعال کردن یک واحد استفاده می شود. به این ترتیب، این واحدها را نمی توان فعال کرد. معمولاً این بدان معنی است که واحد یک عمل یکباره انجام می دهد یا فقط به عنوان وابستگی واحد دیگری استفاده می شود و نباید به تنهایی اجرا شود.

مدیریت واحد

تا کنون با سرویس ها کار کرده ایم و اطلاعات مربوط به واحد و فایل های واحدی را که اطلاع دارد را نمایش systemd می دهیم. با این حال، ما می توانیم با استفاده از برخی دستورات اضافی، اطلاعات دقیق تری در مورد واحدها پیدا کنیم.

نمایش فایل واحد

برای نمایش فایل واحدی که systemd در سیستم آن بارگذاری شده است، می توانید از cat استفاده کنید)این در systemd نسخه ۲۰۹ اضافه شده است .(به عنوان مثال ، برای دیدن فایل واحد atd شبح زمان‌بندی، می‌توانیم تایپ کنیم:

```
systemctl cat atd.service
Output
 [Unit]
Description=ATD daemon
[Service]
Type=forking
ExecStart=/usr/bin/atd
[Install]
WantedBy=multi-user.target
```

خروجی فایل واحدی است که در systemd فرآیند در حال اجرا شناخته شده است. اگر اخیراً فایل‌های واحد را تغییر داده‌اید یا اگر گزینه‌های خاصی را در یک قطعه فایل واحد لغو کرده‌اید، می‌تواند مهم باشد.

نمایش وابستگی ها

برای دیدن درخت وابستگی یک واحد، می توانید از list-dependencies دستور زیر استفاده کنید :

```
systemctl list-dependencies sshd.service
```

این یک نقشه سلسله مراتبی را نشان می دهد که وابستگی هایی که باید برای راه اندازی واحد مورد نظر با آنها برخورد کرد. وابستگی‌ها، در این زمینه، شامل آن دسته از واحدهایی می‌شوند که توسط واحدهای بالای آن مورد نیاز یا مورد نیاز هستند.

```
    Output
sshd.service
├─system.slice
└─basic.target
  ├─microcode.service
  ├─rhel-autorelabel-mark.service
  ├─rhel-autorelabel.service
  ├─rhel-configure.service
  ├─rhel-dmesg.service
  ├─rhel-loadmodules.service
  ├─paths.target
  ├─slices.target
```

برای نشان دادن وابستگی های معکوس (واحدهایی که به واحد مشخص شده بستگی دارند)،

روز سی و پنجم

دستورات متوقف ، خاموش و راه اندازی مجدد در Linux

در زیر موارد معمول استفاده از توقف ، خاموش و راه اندازی مجدد وجود دارد.

halt: به سخت افزار دستور می دهد عملکردهای CPU را متوقف کند.

poweroff به سیستم دستور می دهد تا خاموش شود.

reboot سیستم را مجدداً راه اندازی یا راه اندازی مجدد می کند.

دستورات فوق فقط توسط کاربر فوق العاده قابل اجرا هستند زیرا این اقدامات شامل توقف سخت افزار سیستم است. اگر کاربر در عنوان عضو وارد نشده اید کاربر فوق العاده پس از آن دستور *sudo* دستور را می توان مورد استفاده برای اجرای این دستورات. این به سخت افزار دستور می دهد تا تمام عملکردهای پردازنده را متوقف کند.

```
// syntax of halt command halt [OPTION]...
```

از توقف به راحتی می توان برای متوقف کردن، خاموش کردن سیستم و راه اندازی مجدد سیستم استفاده کرد:

برای متوقف کردن عملکردهای پردازنده به سخت افزار دستور می دهد.

```
halt
```

خاموش شدن سیستم

```
halt -p
```

ریستارت شدن سیستم

```
halt --reboot
```

با این کار کاربر می تواند سیستم را از خط فرمان خاموش کند. با استفاده از توقف تمام عملکردهای پردازنده متوقف می شود و سیستم به حالتی وارد می شود که کاربر بتواند سطح پایین نگهداری را انجام دهد.

poweroff سیگنال ACPI ارسال می کند که به سیستم دستور می دهد تا خاموش شود. در اینجا دستور دستور poweroff آمده است:

ساختار دستور:

```
poweroff [OPTION] ...
```

از Poweroff به راحتی می توان برای **متوقف کردن** ، **خاموش کردن** و **راه اندازی مجدد** سیستم استفاده کرد:

قدرت خاموش سیستم

```
poweroff
```

روز سی و ششم

مدیریت زمان و تاریخ

۵۸.دستور cal

دستور cal یک تقویم فرمت شده را در ترمینال نمایش می‌دهد. اگر هیچ گزینه ای مشخص نشده باشد، cal ماه جاری را با برجسته شدن روز جاری نمایش می‌دهد.

ساختار دستور

```
$ cal [general options] [-jy] [[month] year]
```

گزینه های دستور

- h– تاریخ امروز را نماش نمیدهد.
- m month– یک ماه برای نمایش مشخص کنید. تعیین کننده ماه یک نام کامل ماه (مثلاً فوریه)، مخفف ماه حداقل سه حرف (مثلاً فوریه)، یا یک عدد (مثلاً ۲) است. اگر شما. یک عدد را مشخص کنید و به دنبال آن حرف "f" یا "p"، ماه سال بعد یا سال قبل را به ترتیب نمایش دهید. به عنوان **مثال**، m 2f– فوریه سال آینده را نمایش می‌دهد. یک سال برای نمایش مشخص کنید. به عنوان **مثال**، y 2023– نمایش می‌دهد.
- A num– ماه و چهار ماه بعد از این یکی

مثال های دستور

۱.تقویم این ماه را با برجسته شدن امروز نمایش دهید.

```
$ cal
```

۲. مانند دستور قبلی، اما امروز را نشان نمیدهد.

```
$ cal -h
```

۳. ماه گذشته، این ماه و ماه آینده نمایش داده شود.

```
$ cal -3
```

۴. نمایش تقویم کل این سال

```
$ cal -y
```

۵. نمایش کل تقویم سال ۲۰۲۲

```
$ cal -y  2023
```

۶. همانند دستور قبلی

```
$ cal  2023
```

۷. نمایش تقویم برای دسامبر سال جاری.

```
$ cal -m [December, Dec, or 12]
```

۱۰. تقویم دسامبر ۲۰۲۰۰ را نمایش دهید.

```
$ cal 12 2022
```

روز سی و هفتم

۵۹.دستور fc-cache

در لینوکس اسکن فهرست فونت برای برنامه های کاربردی که از **fontconfig** برای دسترسی به انواع فونت های نصب شد، هستند، استفاده میشود.. هنگامی که دستوراجرا میشود، یک حافظه پنهان ایجاد می شود که شامل خصوصیات هر فونت و نام پرونده ایجاد شده و این حافظه پنهان بیشتر برای سرعت بخشیدن به برنامه هنگام استفاده از کتابخانه ***fontconfig*** هنگام بوت استفاده می شود.

ساختار نحوی دستور:

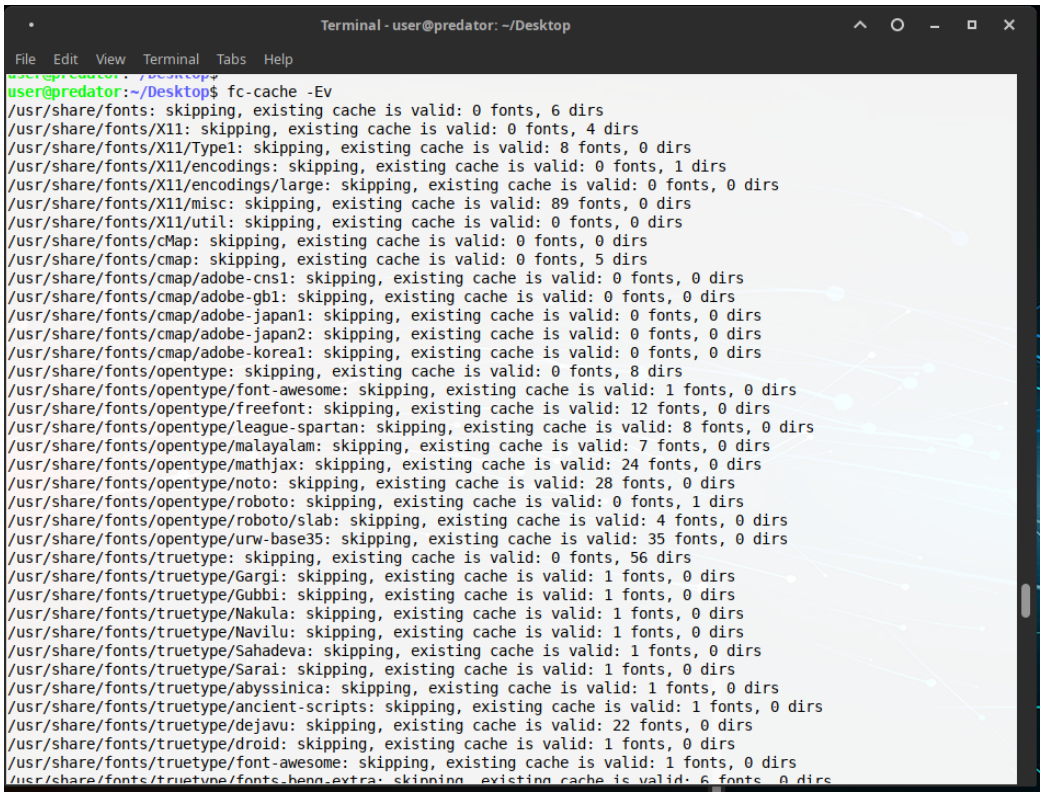
fc-cache [-EfrsvVh] [–error-on-no-fonts] [–force] [–really-force] [[-y dir] [–sysroot dir]] [–system-only] [–verbose] [–version] [–help] [dir…]

گزینه های دستور:

- **(–error-on-no-fonts)** E هنگامی که از این گزینه استفاده می شود اگر هیچ فونت موجود در dir یا فهرست ها در پیکربندی نباشد خطایی ایجاد می‌شود.
- **(–force)** fهنگامی که از این دستور استفاده می شود ، پرونده های حافظه پنهان به ظاهر به روز به زودی ایجاد می شود، این امر مهمترین زمان بررسی Timestamp است.
- **(–really-force)** rاین گزینه تمام پرونده های حافظه پنهان موجود را پاک کرده و مجدداً اسکن می کند.
- s این دستور فقط فهرست های کل سیستم را اسکن می کند و مکان های واقع در/usr/home را حذف می کند.
- **(–verbose)** v در حین کار خروجی را نمایش میدهد.
- **(–version)** V نسخه را نمایش می دهد و از آن خارج می شود.

مثال های دستور:

fc-cache -Ev



توجه : اگر cache با موفقیت تولید شود و غیر صفر باشد ، **fc-cache** صفر را برمی‌گرداند

۶۰.دستور fc-list

دستور **fc-list** بخشی از سیستم *fontconfig* است. برای لیست کردن فونت ها و سبک های فونت موجود استفاده می شود. با استفاده از گزینه format می توان لیست تمام فونت ها را فیلتر و مرتب کرد.

ساختار دستور:

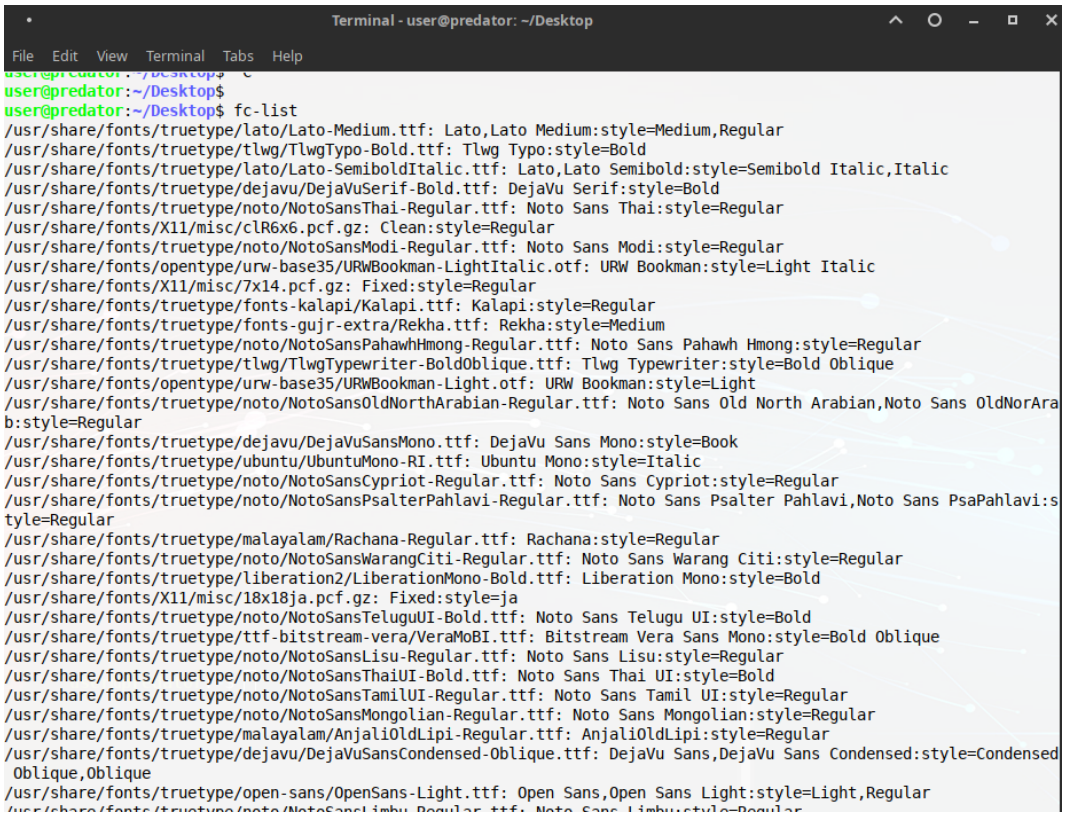
fc-list [-vqVh] [-f FORMAT] [–verbose] [–format=FORMAT] [–quiet] [–version] [–help] [pattern] {element …} List fonts matching [pattern]

مثال : همه مکان های پرونده های فونت موجود در سیستم را با نام فونت ، فاصله و نوع سبک آنها چاپ می کند.

گزینه های دستور:

- **v- verbose:** –برای نشان دادن کل الگوی فونت به صورت کلامی استفاده می شود.
- **f- format = FORMAT:** –برای استفاده از قالب خروجی داده شده.
- **q- quiet:** –همه خروجی های را متوقف می کند .
- **V - version:** –نسخه پیکربندی فونت را نشان میدهد و خارج میشود.
- **h- help:** –برای نشان دادن پیام راهنمای دستور است.

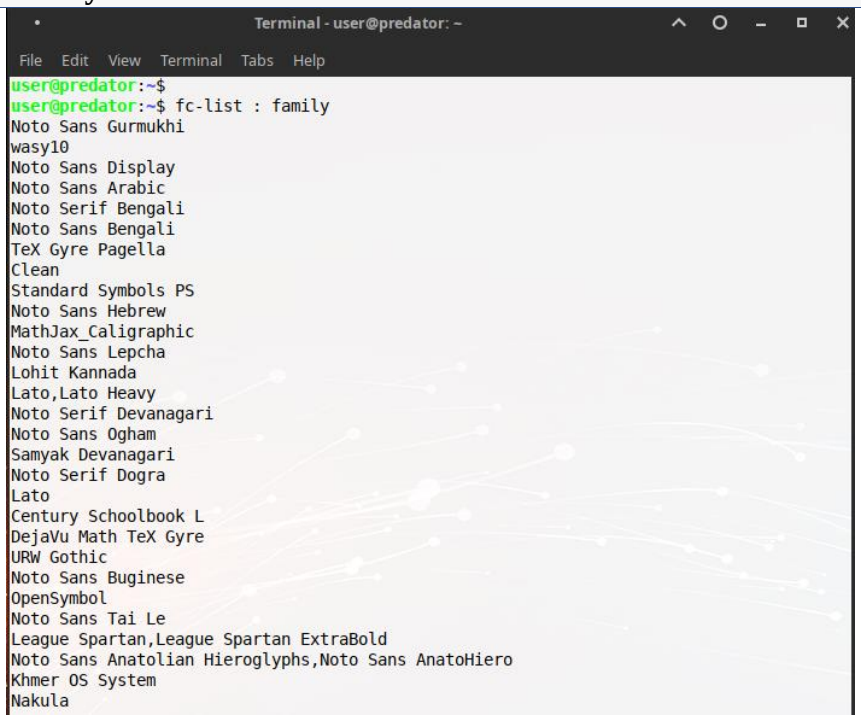
fc-list



مثال های دیگر:

در دستور زیر فقط نام خانواده های فونت را چاپ می کند ، بدون اینکه سایر جزئیات فوق را نشان دهد.

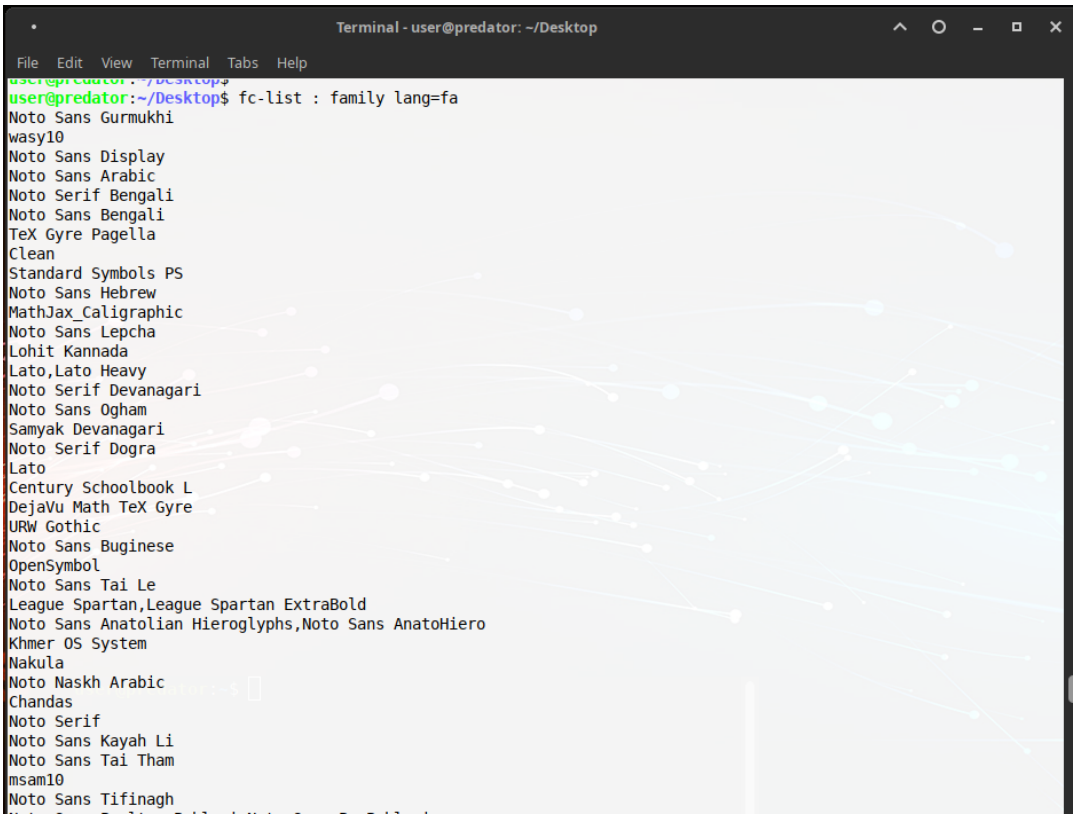
fc-list : family



- **fc-list with font family + Language Selector:** این فقط نام خانواده های فونت را که از کد زبان انتخاب شده پشتیبانی می کنند ، بدون نمایش سایر جزئیات فوق چاپ می کند.

توجه: کد زبان مورد استفاده در اینجا ***"fa" است*** که مخفف زبان **farsi است** .

fc-list : family lang=fa

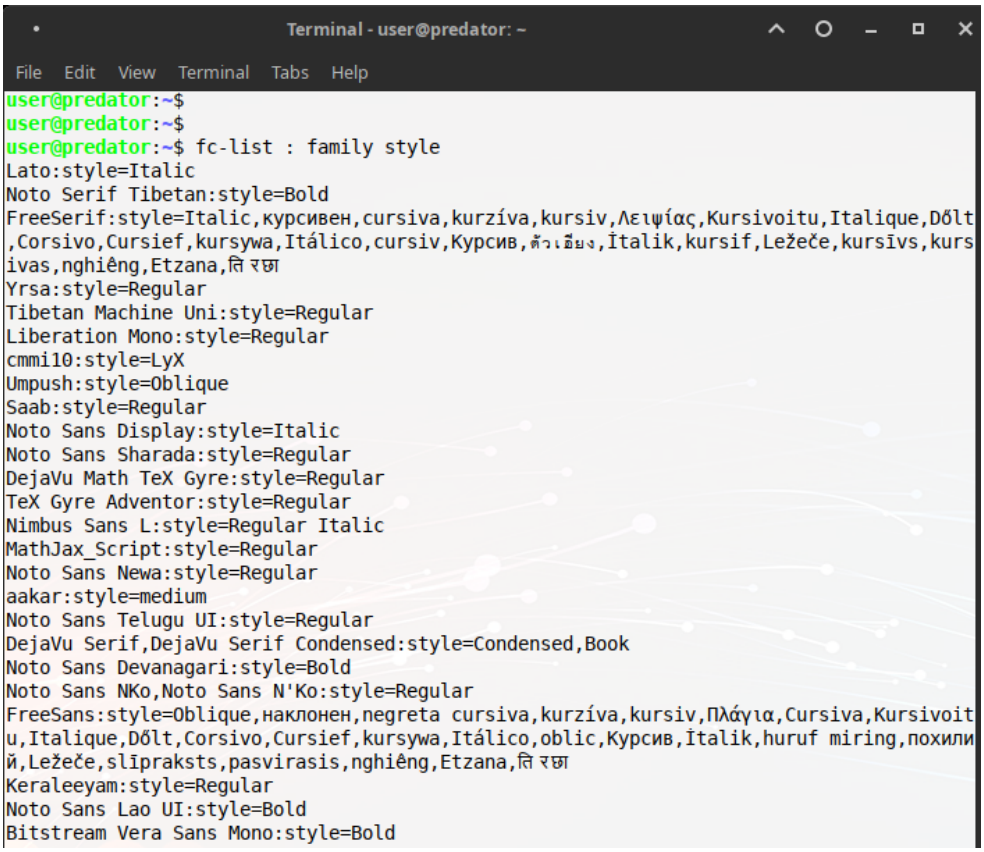


توجه: کد زبان مورد استفاده در اینجا"hi" است که مخفف زبان **هندی** است .

```
fc-list : family lang=hi
```

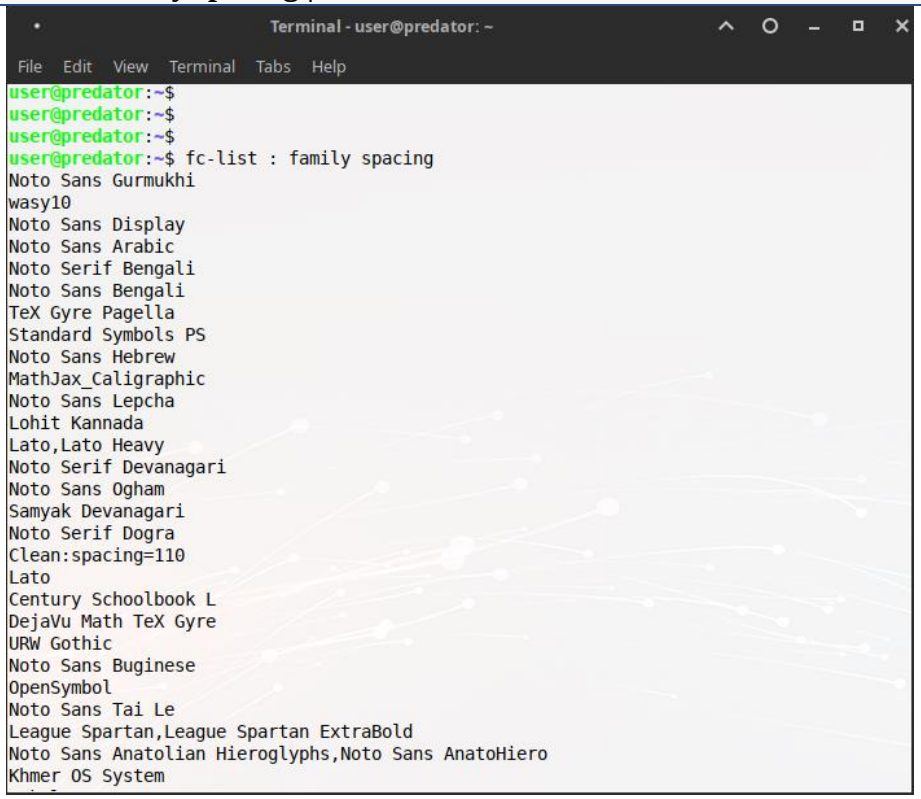
- **fc-list with other selectors:** مشابه انتخاب خانواده، ما همچنین می توانیم محل فایل ، فاصله یا / و سبک فونت های مورد نیاز را برای نمایش در صفحه انتخاب کنیم.

```
fc-list : family style
```



توجه: مرتب سازی و منحصر به فرد نیز می تواند همراه با این دستور ترکیب شود

```
fc-list : family spacing |
```



- **گزینه :fc-list with format از** این گزینه برای قالب بندی متن خروجی به الگوی مورد نیاز کاربر استفاده می شود. در این مثال ، از گزینه format برای بدست آوردن نام خانوادگی همه فونت ها ، مرتب و منحصر به فرد استفاده شده است.

```
fc-list --format="%{family[0]}\n"
```

روز سی و هشتم

۶۱.دستور chmod

دستورchmodبرای تغییر مجوز یک فایل یا فهرست استفاده می شود. برای استفاده از آن، تنظیمات مجاز مورد نظر و فایل یا فایل هایی را که می خواهید تغییر دهید را تعیین می کنید. دو راه برای مشخص کردن مجوز وجود دارد. از تنظیمات مجوز به عنوان یک سری از بیت ها (که کامپیوترها در مورد آنها فکر می کنند) آسان است.

ساختار دستور:

```
chmod [OPTION]... MODE[,MODE]... FILE...
chmod [OPTION]... OCTAL-MODE FILE...
chmod [OPTION]... --reference=RFILE FILE...
```

. در اینجا مشخص است که چگونه کار می کند:

```
rwx  rwx  rwx = 111 111 111
 rw- rw- rw- = 110 110 110
 rwx   ---   --- = 111 000 000

یا

 rwx = 111  ۷ = در دودویی
 rw- = 110  = در دودویی6
 rx = 101  = در دودویی5
 r   -- = 100  = در دودویی4
```

معادل عدد ۷ در مبنای دودویی ۱۱۱ است و همین طور به ترتیب در خط بالا توضیح داده شد.

در اینجا یک جدول اعداد است که تمام تنظیمات رایج را پوشش می دهد.

مقدار	توضیح
777	(rwxrwxrwx)هیچ محدودیتی در مجوزها وجود ندارد. هر کسی می تواند کاری انجام دهد. به طور کلی یک تنظیم مطلوب نیست
755	(rwxr-xr-x)صاحب پرونده ممکن است فایل را بخواند، نوشت و اجرا کند. همه دیگران ممکن است فایل را بخوانند و اجرا کنند. این تنظیم رایج برای برنامه هایی است که توسط همه کاربران مورد استفاده قرار می گیرد.
700	(rwx -- -- --)صاحب پرونده ممکن است فایل را بخواند، نوشتن و اجرا کند. هیچ کس هیچ حقوق ندارد این تنظیم برای برنامه هایی مفید است که تنها مالک می تواند از آن استفاده کند و باید از دیگران خصوصی باشد.
666	(rw-rw-rw-)همه کاربران ممکن است فایل را بخوانند و بنویسند.
644	(rw-r-r --)صاحب ممکن است فایل را بخواند و نوشت، در حالی که دیگران فقط فایل را می خوانند. یک تنظیم رایج برای فایل های داده ای که همه می توانند بخوانند، اما تنها مالک ممکن است تغییر کند.
600	(rw -- -- ---)مالک ممکن است فایل را بخواند و نوشت. همه دیگران حقوق ندارند. یک تنظیم رایج برای فایلهای داده ای که مالک می خواهد خصوصی نگه دارد.

گزینه های دستور

- c, changes مانند verbose، اما فقط در صورتی که تغییری ایجاد شود، گزارش داده می‌شود.
- nopreserveroot: '/'را به‌صورت خاص نگیرید (پیش‌فرض).
- Preserveroot عملیات بازگشتی روی '/' انجام ندهید.
- f, silent, quiet بیشتر پیام‌های خطا را سرکوب کنید.
- v, verbose برای هر فایل پردازش شده، تشخیصی را خروجی دهید.
- reference=RFILE از حالت RFILE به جای مقادیر MODE استفاده کنید.
- R, recursive فایل‌ها و دایرکتوری‌ها را به صورت بازگشتی تغییر دهید.
- Help این راهنما را نمایش دهید و خارج شوید.
- Version اطلاعات نسخه را چاپ کنید و خارج شوید

مثال های دستور

- c, changes:

```
chmod -c 644 file.txt
```

- nopreserveroot:

```
chmod --nopreserveroot 600 folder
```

- Preserveroot:

```
chmod --Preserveroot 755 file.txt
```

- f, silent, quiet:

```
chmod -f 777 file.txt
```

- v, verbose:

```
chmod -v 600 file.txt
```

- reference=RFILE:

```
chmod --reference=reference_file file.txt
```

- R, recursive:

```
chmod -R 755 folder
```

- Help:

```
chmod --help
```

- Version:

```
chmod --version
```

۶۲. دستور chage

از دستور **chage** برای مشاهده و تغییر اطلاعات منقضی شده رمز عبور کاربر استفاده می شود. این دستور هنگامی استفاده می شود که ورود به سیستم برای مدت زمان محدودی برای کاربر ارائه شود یا اینکه لازم است گذرواژه ورود به سیستم را به زمان دیگری تغییر دهید. با کمک این دستور می توانیم اطلاعات مربوط به پیری یک حساب ، تاریخ تغییر رمز عبور، تنظیم زمان تغییر رمز ورود ، قفل کردن حساب پس از مدت زمان مشخص و غیره را مشاهده کنیم.

دستور chage در زیر آورده شده است:

ساختار دستور:

```
chage [options] LOGIN
```

برای مشاهده لیست گزینه های قابل استفاده با دستور chage از گزینه help استفاده کنید

ورودی:

```
chage -h
```

خروجی دستور:

```
Terminal - user@predator: ~
File  Edit  View  Terminal  Tabs  Help
user@predator:~$ chage -h
Usage: chage [options] LOGIN

Options:
  -d, --lastday LAST_DAY        set date of last password change to LAST_DAY
  -E, --expiredate EXPIRE_DATE  set account expiration date to EXPIRE_DATE
  -h, --help                    display this help message and exit
  -i, --iso8601                 use YYYY-MM-DD when printing dates
  -I, --inactive INACTIVE       set password inactive after expiration
                                to INACTIVE
  -l, --list                    show account aging information
  -m, --mindays MIN_DAYS        set minimum number of days before password
                                change to MIN_DAYS
  -M, --maxdays MAX_DAYS        set maximum number of days before password
                                change to MAX_DAYS
  -R, --root CHROOT_DIR         directory to chroot into
  -W, --warndays WARN_DAYS      set expiration warning days to WARN_DAYS

user@predator:~$
```

مثال های دستور:

گزینه l- برای مشاهده اطلاعات انقضا حساب از این گزینه استفاده کنید. برای مشاهده اطلاعات ریشه، من از *sudo* استفاده می کنم :

```
sudo chage -l root
```

خروجی دستور:

```
Terminal - user@predator: ~
File  Edit  View  Terminal  Tabs  Help
user@predator:~$ sudo chage -l root
Last password change                                    : May 24, 2021
Password expires                                        : never
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 0
Maximum number of days between password change          : 99999
Number of days of warning before password expires       : 7
user@predator:~$
```

گزینه -d: از این گزینه برای تنظیم آخرین تاریخ تغییر رمز عبور به تاریخ مشخص شده در دستور استفاده کنید. برای تغییر اطلاعات ریشه ، من از کلمه کلیدی "sudo" استفاده می کنم. بعلاوه من از گزینه -l برای مشاهده تاریخ تغییر یافته استفاده می کنم.

```
sudo chage -d 2018-12-01 root
```

گزینه -E: برای تعیین تاریخ انقضا حساب باید از این گزینه استفاده کنید. به منظور تغییر اطلاعات ریشه ، من از کلمه کلیدی sudoاستفاده می کنم. بعلاوه من از گزینه -l برای مشاهده تاریخ تغییر یافته استفاده می کنم.

ورودی:

```
sudo chage -E root
```

خروجی دستور:

```
pi@raspberrypi:~ $ sudo chage -E 2018-12-31 root
pi@raspberrypi:~ $ sudo chage -l root
Last password change                                    : Dec 01, 2018
Password expires                                        : never
Password inactive                                       : never
Account expires                                         : Dec 31, 2018
Minimum number of days between password change          : 0
Maximum number of days between password change          : 99999
Number of days of warning before password expires       : 7
pi@raspberrypi:~ $
```

گزینه -m: یا M از این گزینه برای تعیین حداکثر و حداقل تعداد روز بین تغییر رمز عبور استفاده کنید. برای تغییر اطلاعات ریشه ، من از کلمه کلیدی sudo استفاده می کنم. بعلاوه من از گزینه -l برای مشاهده دوره تغییر یافته استفاده می کنم.

```
sudo chage -M 5 root
```

خروجی دستور:

```
pi@raspberrypi:~ $ sudo chage -M 5 root
pi@raspberrypi:~ $ sudo chage -l root
Last password change                                    : Dec 01, 2018
Password expires                                        : Dec 06, 2018
Password inactive                                       : never
Account expires                                         : Dec 31, 2018
Minimum number of days between password change          : 0
Maximum number of days between password change          : 5
Number of days of warning before password expires       : 0
pi@raspberrypi:~ $
```

گزینه -I: از این گزینه برای تعیین تعداد روزهایی که حساب باید پس از انقضا غیرفعال باشد استفاده کنید. لازم است که کاربر پس از انقضا رمز ورود خود را تغییر دهد ، این دستور زمانی مفید است که کاربر پس از انقضا وارد سیستم نمی شود. حتی پس از این دوره عدم فعالیت، در صورت عدم تغییر رمز ورود، حساب قفل شده و کاربر باید برای باز کردن قفل آن به مدیر مراجعه کند. برای تغییر اطلاعات انقضا ریشه ، من از کلمه کلیدی sudo استفاده می کنم. بعلاوه من از گزینه -l برای مشاهده دوره عدم فعالیت استفاده کردم.

```
sudo chage -I 5 root
```

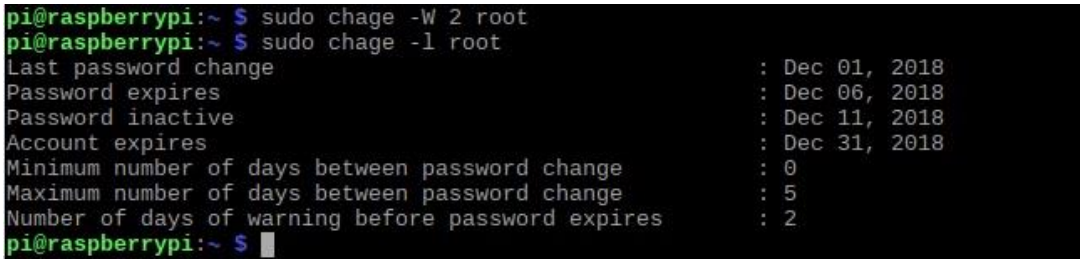
خروجی دستور:

```
pi@raspberrypi:~ $ sudo chage -I 5 root
pi@raspberrypi:~ $ sudo chage -l root
Last password change                                    : Dec 01, 2018
Password expires                                        : Dec 06, 2018
Password inactive                                       : Dec 11, 2018
Account expires                                         : Dec 31, 2018
Minimum number of days between password change          : 0
Maximum number of days between password change          : 5
Number of days of warning before password expires       : 0
pi@raspberrypi:~ $
```

گزینه: W– از این گزینه برای دادن اخطار قبلی قبل از انقضا رمز عبور استفاده کنید. ورودی داده شده در دستورتعداد روزهای قبل از تاریخ انقضا است که باید هشدار داده شود. برای تغییر اطلاعات منقضی شدن root من از کلمه کلیدی *sudo* استفاده میکنم. بعلاوه من از گزینه -l برای مشاهده دوره هشدار استفاده میکنم.

```
sudo chage -W 2 root
```

خروجی دستور:



روز سی و نهم

مدیریت چندرسانه ای

۶۳.دستور amixer

میکسر خط دستوربرای درایور کارت صدا یعنی ALSA (Advanced Linux Sound Architecture) است. می تواند از چندین کارت صدا پشتیبانی کند. بدون هیچ آرگومان تنظیمات فعلی میکسر را برای کارت صدا پیش فرض و همچنین دستگاه نمایش می دهد. این روش خوبی برای دیدن لیستی از کنترل های ساده میکسر است که می توانید استفاده کنید.

موارد استفاده:

- ✓ تنظیم سطح صدا
- ✓ تنظیم تعادل صدا بین کانال های مختلف
- ✓ فعال یا غیرفعال کردن میوت
- ✓ تنظیم تن صدا و بیس
- ✓ انتخاب منبع ورودی صدا
- ✓ انتخاب خروجی صدا
- ✓ مشاهده اطلاعات مربوط به کارت صدا

نحوه استفاده:

- ✓ بدون هیچ گونه آرگومانی، amixer اطلاعات مربوط به تمام کنترل های موجود را نمایش می‌دهد.
- ✓ برای مشاهده اطلاعات مربوط به یک کنترل خاص، نام آن را به عنوان آرگومان به دستور بدهید.
- ✓ برای تنظیم مقدار یک کنترل، از گزینه s– و نام کنترل و مقدار جدید استفاده کنید.
- ✓ برای افزایش یا کاهش مقدار یک کنترل، از گزینه های i– و d– استفاده کنید.

ساختار دستور:

```
amixer [-option] [دستور]
amixer [-option] [command]
```

گزینه های دستور:

- c – : انتخاب کارت صدا
- D – : نمایش اطلاعات به صورت دیباگ
- h – : نمایش راهنمای دستور
- i – : افزایش مقدار یک کنترل
- m – : میوت کردن یک کنترل
- s – : تنظیم مقدار یک کنترل
- u unmute – : کردن یک کنترل

info– : اطلاعات را نمایش می دهد و سپس خارج می شود.

```
nishanth@nishanth-VirtualBox:~$ amixer info
Card default 'I82801AAICH'/'Intel 82801AA-ICH with AD1980 at irq 21'
  Mixer name    : 'Analog Devices AD1980'
  Components    : 'AC97a:41445370'
  Controls      : 46
  Simple ctrls  : 32
nishanth@nishanth-VirtualBox:~$
```

scontols– : لیست کاملی از کنترل های ساده را نشان می دهد و سپس خارج می شود.

```
nishanth@nishanth-VirtualBox:~$ amixer scontrols
Simple mixer control 'Master',0
Simple mixer control 'Master Mono',0
Simple mixer control 'Headphone Jack Sense',0
Simple mixer control '3D Control - Switch',0
Simple mixer control 'PCM',0
Simple mixer control 'PCM Out Path & Mute',0
Simple mixer control 'Surround',0
Simple mixer control 'Surround Jack Mode',0
Simple mixer control 'Center',0
Simple mixer control 'LFE',0
Simple mixer control 'Line',0
Simple mixer control 'Line Jack Sense',0
Simple mixer control 'CD',0
Simple mixer control 'Mic',0
Simple mixer control 'Mic Boost (+20dB)',0
Simple mixer control 'Mic Select',0
Simple mixer control 'Video',0
Simple mixer control 'Phone',0
Simple mixer control 'Beep',0
Simple mixer control 'Aux',0
Simple mixer control 'Mono Output Select',0
Simple mixer control 'Capture',0
```

scontents– : لیست کاملی از کنترل های ساده میکسر و همچنین محتوای آنها را نشان می دهد.

```
nishanth@nishanth-VirtualBox:~$ amixer scontents
Simple mixer control 'Master',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 31
  Mono:
  Front Left: Playback 17 [55%] [-21.00dB] [on]
  Front Right: Playback 17 [55%] [-21.00dB] [on]
Simple mixer control 'Master Mono',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 63
  Mono:
  Front Left: Playback 0 [0%] [-94.50dB] [off]
  Front Right: Playback 0 [0%] [-94.50dB] [off]
Simple mixer control 'Headphone Jack Sense',0
  Capabilities: pswitch pswitch-joined
  Playback channels: Mono
  Mono: Playback [off]
Simple mixer control '3D Control - Switch',0
  Capabilities: pswitch pswitch-joined
  Playback channels: Mono
  Mono: Playback [off]
Simple mixer control 'PCM',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
```

set or sset : محتویات مخلوط کن ساده را تنظیم می کند.

```
nishanth@nishanth-VirtualBox:~$ amixer set Master 10%
Simple mixer control 'Master',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 31
  Mono:
  Front Left: Playback 4 [13%] [-40.50dB] [on]
  Front Right: Playback 4 [13%] [-40.50dB] [on]
nishanth@nishanth-VirtualBox:~$
```

get or sget : محتوای کنترلی ساده را نشان می دهد.

```
nishanth@nishanth-VirtualBox:~$ amixer get Master
Simple mixer control 'Master',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 31
  Mono:
  Front Left: Playback 4 [13%] [-40.50dB] [on]
  Front Right: Playback 4 [13%] [-40.50dB] [on]
```

controls : لیست کاملی از کنترل های کارت را نمایش می دهد.

```
nishanth@nishanth-VirtualBox:~$ amixer controls
numid=3,iface=MIXER,name='Master Mono Playback Switch'
numid=4,iface=MIXER,name='Master Mono Playback Volume'
numid=1,iface=MIXER,name='Master Playback Switch'
numid=2,iface=MIXER,name='Master Playback Volume'
numid=42,iface=MIXER,name='Headphone Jack Sense'
numid=32,iface=MIXER,name='3D Control - Switch'
numid=31,iface=MIXER,name='PCM Out Path & Mute'
numid=20,iface=MIXER,name='PCM Playback Switch'
numid=21,iface=MIXER,name='PCM Playback Volume'
numid=40,iface=MIXER,name='Surround Jack Mode'
numid=22,iface=MIXER,name='Surround Playback Switch'
numid=23,iface=MIXER,name='Surround Playback Volume'
```

content: لیست کاملی از کنترل های کارت را با محتوای آنها نیز نمایش می دهد.

```
nishanth@nishanth-VirtualBox:~$ amixer contents
numid=3,iface=MIXER,name='Master Mono Playback Switch'
  ; type=BOOLEAN,access=rw------,values=2
  : values=off,off
numid=4,iface=MIXER,name='Master Mono Playback Volume'
  ; type=INTEGER,access=rw---R--,values=2,min=0,max=63,step=0
  : values=0,0
  | dBscale-min=-94.50dB,step=1.50dB,mute=0
numid=1,iface=MIXER,name='Master Playback Switch'
  ; type=BOOLEAN,access=rw------,values=2
  : values=on,on
numid=2,iface=MIXER,name='Master Playback Volume'
  ; type=INTEGER,access=rw---R--,values=2,min=0,max=31,step=0
  : values=4,4
  | dBscale-min=-46.50dB,step=1.50dB,mute=0
numid=42,iface=MIXER,name='Headphone Jack Sense'
  ; type=BOOLEAN,access=rw------,values=1
  : values=off
numid=32,iface=MIXER,name='3D Control - Switch'
  ; type=BOOLEAN,access=rw------,values=1
```

cset : برای تنظیم کنترلهای کارت استفاده می شود.

```
nishanth@nishanth-VirtualBox:~$ amixer cset name='Line Playback Volume' 66,77
numid=13,iface=MIXER,name='Line Playback Volume'
  ; type=INTEGER,access=rw---R--,values=2,min=0,max=63,step=0
  : values=63,63
  | dBscale-min=-34.50dB,step=1.50dB,mute=0
```

cget : این محتویات کنترل کارت را نشان می دهد. شناسه دارای همان **نحو** دستور cset است.

```
nishanth@nishanth-VirtualBox:~$ amixer cget name='Line Playback Volume'
numid=13,iface=MIXER,name='Line Playback Volume'
  ; type=INTEGER,access=rw---R--,values=2,min=0,max=63,step=0
  : values=63,63
  | dBscale-min=-34.50dB,step=1.50dB,mute=0
```

-c : شماره کارتی را که باید کنترل شود انتخاب می کند.

```
nishanth@nishanth-VirtualBox:~$ amixer -c I82801AAICH
Simple mixer control 'Master',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 31
  Mono:
  Front Left: Playback 4 [13%] [-40.50dB] [on]
  Front Right: Playback 4 [13%] [-40.50dB] [on]
Simple mixer control 'Master Mono',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 63
  Mono:
  Front Left: Playback 0 [0%] [-94.50dB] [off]
  Front Right: Playback 0 [0%] [-94.50dB] [off]
Simple mixer control 'Headphone Jack Sense',0
  Capabilities: pswitch pswitch-joined
  Playback channels: Mono
  Mono: Playback [off]
Simple mixer control '3D Control - Switch',0
```

-D : نام دستگاهی را که باید کنترل شود انتخاب می کند.

```
nishanth@nishanth-VirtualBox:~$ amixer -D default
Simple mixer control 'Master',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 31
  Mono:
  Front Left: Playback 4 [13%] [-40.50dB] [on]
  Front Right: Playback 4 [13%] [-40.50dB] [on]
Simple mixer control 'Master Mono',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 63
  Mono:
  Front Left: Playback 0 [0%] [-94.50dB] [off]
  Front Right: Playback 0 [0%] [-94.50dB] [off]
Simple mixer control 'Headphone Jack Sense',0
  Capabilities: pswitch pswitch-joined
  Playback channels: Mono
  Mono: Playback [off]
Simple mixer control '3D Control - Switch',0
  Capabilities: pswitch pswitch-joined
  Playback channels: Mono
```

- یا-stdin : از stdin بخوانید و هر دستور را به ترتیب اجرا کنید.

```
nishanth@nishanth-VirtualBox:~$ amixer -s
^Z
[4]+  Stopped                 amixer -s
```

-q : نتیجه تغییرات را نشان نمی دهد.

```
nishanth@nishanth-VirtualBox:~$ amixer -q
Simple mixer control 'Master',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 31
  Mono:
  Front Left: Playback 4 [13%] [-40.50dB] [on]
  Front Right: Playback 4 [13%] [-40.50dB] [on]
Simple mixer control 'Master Mono',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 63
  Mono:
  Front Left: Playback 0 [0%] [-94.50dB] [off]
  Front Right: Playback 0 [0%] [-94.50dB] [off]
Simple mixer control 'Headphone Jack Sense',0
  Capabilities: pswitch pswitch-joined
  Playback channels: Mono
  Mono: Playback [off]
```

R- : از مقدار خام برای ارزیابی نمایندگی درصد که حالت پیش فرض نیز است استفاده می کند.

```
nishanth@nishanth-VirtualBox:~$ amixer -R
Simple mixer control 'Master',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 31
  Mono:
  Front Left: Playback 4 [13%] [-40.50dB] [on]
  Front Right: Playback 4 [13%] [-40.50dB] [on]
Simple mixer control 'Master Mono',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 63
  Mono:
  Front Left: Playback 0 [0%] [-94.50dB] [off]
  Front Right: Playback 0 [0%] [-94.50dB] [off]
Simple mixer control 'Headphone Jack Sense',0
  Capabilities: pswitch pswitch-joined
  Playback channels: Mono
  Mono: Playback [off]
Simple mixer control '3D Control - Switch',0
  Capabilities: pswitch pswitch-joined
  Playback channels: Mono
```

M- : از حجم نگاشت شده برای ارزیابی درصد نمایش مانند **alsamixer** استفاده می کند تا طبیعی تر برای گوش انسان باشد.

```
nishanth@nishanth-VirtualBox:~$ amixer -M
Simple mixer control 'Master',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 31
  Mono:
  Front Left: Playback 4 [5%] [-40.50dB] [on]
  Front Right: Playback 4 [5%] [-40.50dB] [on]
Simple mixer control 'Master Mono',0
  Capabilities: pvolume pswitch
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 63
  Mono:
  Front Left: Playback 0 [0%] [-94.50dB] [off]
  Front Right: Playback 0 [0%] [-94.50dB] [off]
Simple mixer control 'Headphone Jack Sense',0
  Capabilities: pswitch pswitch-joined
```

v- : نسخه را نمایش می دهد.

```
nishanth@nishanth-VirtualBox:~$ amixer -v
amixer version 1.1.3
```

۶۴.دستور brasero

برنامه ساده و آسان برای رایت CD/DVD برای دسکتاپ Gnome. البته در سایر میزهای دیگه هم قابل نصب و اجرا است. همچنین Braseroیک ابزار قدرتمند برای رایت دیسک های نوری **(CD/DVD)** در سیستم عامل لینوکس است. این برنامه به شما امکان می دهد تا انواع مختلفی از داده ها را بر روی دیسک های نوری رایت کنید، مانند فایل های صوتی، تصویری، ویدیویی، و داده های کامپیوتری.

ساختار دستور :

```
$ brasero [option] [uri] [URI]
```

rasero یک برنامه رایت دیسک است. طراحی آن ساده و استفاده از آن آسان است. این امکان ایجاد CD/DVD داده، CD صوتی، کپی CD/DVD و رایت تصاویر را فراهم می‌کند. دارای برخی از ویژگی های خوب، مانند پیش نمایش آهنگ، تصویر و ویدیو. همچنین می تواند فایل ها را جستجو کند و لیست پخش و محتویات آن را نمایش دهد.

گزینه های دستور:

- h نمایش راهنمای دستور
- v نمایش نسخه برنامه
- d مشخص کردن دستگاه رایت دیسک (پیش فرض /dev/sr0)
- s سرعت رایت (پیش فرض حداکثر سرعت)
- f نام فایل یا پوشه ای که می خواهید رایت کنید
- l لیست کردن محتویات دیسک نوری
- e پاک کردن دیسک نوری قابل بازنویسی
- b رایت بوت لودر
- t نوع دیسک نوری (پیش فرض CDROM)
- c ایجاد یک دیسک bootable
- i رایت یک image ISO
- ن‌ام فایل image ISO برای خروجی
- n تعداد کپی های مورد نظر برای رایت
- j رایت مستقیم از یک پوشه
- q فعال کردن حالت سکوت
- p پیش نمایش عملیات رایت قبل از اجرا

رایت یک تصویر ISO یا داده‌ها روی یک دیسک:

```
$brasero --burn /path/to/image.iso
```

کپی کردن یک دیسک:

```
$brasero --copy /dev/cdrom
```

رایت یک دیسک صوتی:

```
$brasero --audio /path/to/audio/files
```

پاک کردن محتوای یک دیسک قابل رایت:

```
$brasero --erase /dev/cdrom
```

جدا کردن دیسک از درایو:

```
$brasero --eject /dev/cdrom
```

رایت یک فایل صوتی به نام song.mp3 بر روی CD:

```
brasero -f song.mp3
```

رایت یک پوشه به نام data بر روی DVD:

```
brasero -f data -t DVD
```

لیست کردن محتویات یک دیسک CD:

```
brasero -l
```

پاک کردن یک دیسک DVD قابل بازنویسی:

```
brasero -e
```

رایت بوت لودر Grub بر روی دیسک CD:

```
brasero -b grub
```

ایجاد یک دیسک bootable از سیستم عامل Ubuntu:

```
brasero -i ubuntu.iso -c
```

رایت ۵ کپی از یک فایل image ISO به نام image.iso بر روی CD:

```
brasero -i image.iso -n 5
```

رایت مستقیم محتویات یک پوشه به نام my_project بر روی DVD:

```
brasero -j my_project -t DVD
```

رایت یک فایل صوتی به نام music.wav بر روی CD با سرعت x۴:

```
brasero -f music.wav -s 4x
```

رایت یک دیسک CD بدون نمایش هیچ گونه پیامی:

```
brasero -f data -q
```

لطفاً توجه داشته باشید که در هر مثال، `/path/to/image.iso` و `/path/to/audio/files` و `/dev/cdrom` به ترتیب باید با مسیر و فایل‌های واقعی که می‌خواهید از آن‌ها استفاده کنید جایگزین شوند. همچنین، برای اطلاعات دقیق‌تر و گزینه‌های بیشتر، می‌توانید از دستور `brasero --help` برای نمایش راهنمای استفاده از برنامه Brasero استفاده کنید.

۶۵.دستور eject

دستورeject برای از خارج کردن رسانه هایی مانند DVD / CD ROM یا فلاپی دیسک از system استفاده می شود. اجازه می دهد تا یک رسانه قابل جابجایی (معمولاً CD-ROM ، فلاپی دیسک ، نوار ، یا دیسک JAZ یا ZIP) را با استفاده از نرم افزار خارج کنید. همچنین می توانید برخی از تعویض های CD-ROM چند دیسک ، ویژگی خودکار خروج را کنترل کنید (توسط برخی دستگاه ها پشتیبانی می شود) و همچنین سینی دیسک برخی از درایوهای CD-ROM را ببندید. روش های بیرون انداختن به نوع دستگاه بستگی دارد ، چه CD-ROM باشد ، چه دستگاه SCSI ، فلاپی قابل جدا شدن یا نوار. عملکرد پیش فرض eject این است که هر چهار روش را به ترتیب امتحان می کند تا زمانی که به موفقیت برسد. اگر mount باشد ، قبل از بیرون انداختن دستگاه از آن خارج نمی شود.

```
$ eject /dev/cdrom
$ eject /mnt/cdrom /
$ eject /dev/sda
```

ساختار دستور:

```
eject [... OPTIONS]
```

گزینه های دستور:

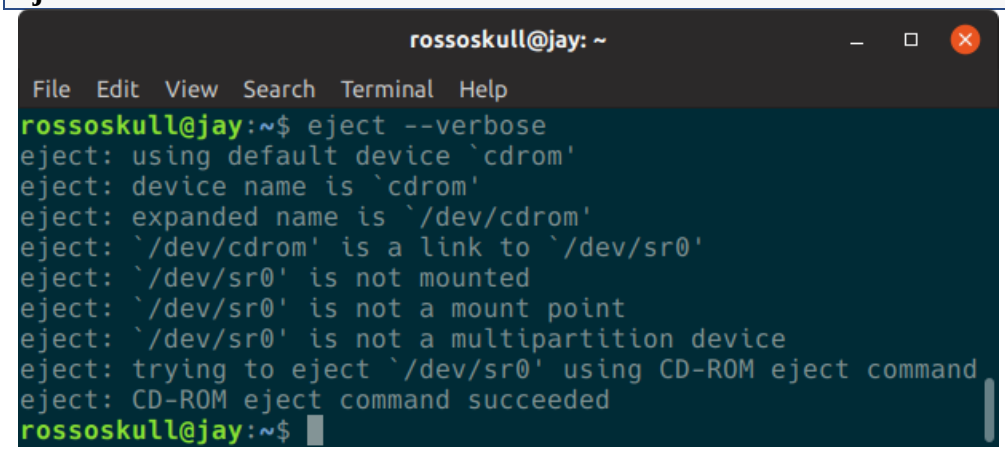
- f به طور پیش فرض، eject قبل از خارج کردن دیسک، از کاربر تأیید می‌خواهد. با استفاده از این گزینه، دیسک بدون تأیید کاربر خارج می‌شود.
- l لیست درایوهای نوری موجود را نمایش می‌دهد.

- d نام درایو نوری specific را برای خارج کردن دیسک از آن مشخص می‌کند.
- t نوع دیسک را برای خارج کردن مشخص می‌کند.
- m پیامی را قبل از خارج کردن دیسک نمایش می‌دهد.
- h راهنمای دستور را نمایش می‌دهد.

خروجی دستور:

- **–verbose: یا v** – اطلاعات بیشتر در مورد اجرای دستور را از سیستم خارج کنید.

```
eject -v
```

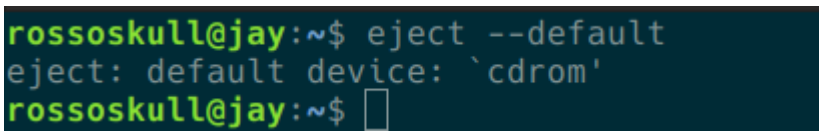


–**d or --default** نام دستگاه پیش فرض را لیست کنید.

ساختار دستور:

```
eject -d
```

خروجی دستور:



با این دستور حالت بیرون انداختن خودکار را که توسط برخی دستگاه ها پشتیبانی می شود کنترل کنید. هنگام بسته شدن دستگاه ، این کار را برای بیرون انداختن خودکار فعال کنید.

ساختار دستور:

```
eject -a on
```

- –**c or –changerslot** : یک اسلات CD را از یک تعویض CD-ROM ATAPI / IDE انتخاب کنید. برای استفاده از این ویژگی به Linux 2.0+ نیاز دارید. شما باید مطمئن شوید که اسلات CD از قبل استفاده نشده است. اولین شکاف موجود با عنوان صفر شناخته می شود.

ساختار دستور:

```
eject -c 0
```

–**t or –trayclose** : با استفاده از این دستور درایو را به یک دستور close می دهید.

توجه: پشتیبانی سخت افزاری از این ویژگی محدود است.

ساختار دستور:

```
eject -t
```

- **-T یا –traytoggle** : در صورت باز بودن شکاف ، دستور بستن سینی را می دهد یا اگر شکاف بسته است ، یک دستور eject می دهد.

توجه: پشتیبانی سخت افزاری برای این دستور محدود است، به عنوان آن استفاده می کند از بالا *-t* فرمان.

ساختار دستور:

```
eject -T
```

- **-x یا –cdspeed** : با این گزینه به CD-ROM دستور speed را به درایو بدهید. آرگومان یک عدد است که نشان دهنده سرعت مورد نظر است (به عنوان مثال ۳ برای سرعت 3X یا ۰ برای حداکثر سرعت داده. پشتیبانی دستگاه از این دستور محدود است و شما فقط می توانید سرعتی را که درایو توانایی آن را دارد مشخص کنید. با تغییر رسانه ، این گزینه پاک می شود. این گزینه با دستور c- یا t- نیز قابل استفاده است.

ساختار دستور:

```
eject -x 0
```

- **-X or –listspeed** : لیستی از سرعتهای موجود را از درایو CD-ROM با استفاده از این گزینه نمایش دهید. می توانید از سرعت های ذکر شده به عنوان آرگومان در گزینه x- استفاده کنید. برای کار با این گزینه به Linux 2.6.13+نیاز دارید. در نسخه های قدیمی ، فقط حداکثر سرعت را گزارش می دهد. همچنین ، برخی از درایوها ممکن است سرعت را به درستی گزارش نکنند ، بنابراین این گزینه با آنها کار نمی کند.

ساختار دستور:

```
eject -X
```

- **-n or –noop** : دستگاه انتخاب شده را نمایش دهید ، اما با استفاده از این گزینه هیچ عملی انجام **ندهید** .

ساختار دستور:

```
eject -n
```

خروجی دستور:

```
rossoskull@jay:~$ eject -n
eject: device is `/dev/sr0'
rossoskull@jay:~$
```

- **-r or –cdrom** : مشخص کنید که درایو باید با استفاده از دستور CDROM eject با استفاده از این گزینه خارج شود.

ساختار دستور:

```
eject -r
```

- **-s or –scsi** : مشخص کنید که درایو باید با استفاده از دستورات SCSI با استفاده از این گزینه خارج شود.

ساختار دستور:

```
eject -s
```

- **f or –floppy–** : مشخص کنید که درایو باید با استفاده از یک دستور فلاپی دیسک قابل جابجایی با استفاده از این گزینه خارج شود.

ساختار دستور:

```
eject -f
```

- **q or –tape–** : مشخص کنید که درایو باید با استفاده از یک دستور آفلاین درایو tape با استفاده از این گزینه خارج شود.

ساختار دستور:

```
eject -q
```

- **p or –proc-** : با این دستور از / proc / mounts به جای / etc / mtab استفاده کنید. همچنین از گزینه-n برای umount عبور می کند.

ساختار دستور:

```
eject -p
```

- **–no-unmountیا m–** : اجازه کار کردن با درایورهای دستگاه را که به طور خودکار رسانه های قابل جابجایی را نصب می کنند کار می کند و بنابراین باید همیشه نصب شود. این گزینه به eject می گوید که سعی نکنید دستگاه مورد نظر را پیاده کنید ، حتی اگر دستگاه آن مطابق با:

```
/ proc / mounts
```

یا

```
/ etc / mtab
```

ساختار دستور:

```
eject -m
```

- **V or –version–** : شماره نسخه را برای کنسول وارد کرده و از آن خارج شوید.

ساختار دستور:

```
eject -V
```

۶۶.دستور find

نحوه استفاده از دستور "find" برای جستجو برای نام فایل های چندگانه (پسوند) در linux

چندین بار، ما در موقعیتی قرار داریم که در آن ما باید چندین فایل با پسوند های مختلف را جستجو کنیم، این احتمالا به چندین کاربر linux خصوصا در داخل ترمینال رخ داده است. چندین ابزار linux وجود دارد که ما می توانیم از آنها برای یافتن یا find فایل ها در سیستم فایل استفاده کنیم، اما find نام فایل های متعدد و یا فایل های با پسوند های مختلف گاهی اوقات می تواند پیچیده و نیاز به دستورات خاص. یکی از بسیاری از امکانات برای یافتن فایل ها در یک سیستم فایل linux ابزار findاست و در این راهنمای نحوه استفاده از چند نمونه از استفاده از findبرای کمک به ما در find چندین نام فایل در یک زمان وارد می کنیم.

گزینه های دستور:

- path مسیر دایرکتوری را برای جستجو مشخص می کند.
- name نام فایل را برای جستجو مشخص می کند.
- type نوع فایل را برای جستجو مشخص می کند.
- mtime فایل هایی را پیدا می کند که در مدت زمان مشخص شده تغییر کرده

- atime فایل هایی را پیدا می کند که در مدت زمان مشخص شده به آنها دسترسی
- ctime فایل هایی را پیدا می کند که در مدت زمان مشخص شده تغییر
- size فایل هایی را با اندازه مشخص شده پیدا می کند.
- perm فایل هایی را با مجوز مشخص شده پیدا می کند.
- exec دستور مشخص شده را برای هر فایل یافت شده اجرا می
- print مسیر هر فایل یافت شده را به صورت

مثال های دستور find

قسمت اول – دستورالعمل اساسی برای یافتن فایل با نام

تمام فایل هایی که نام آن hosein .txt را در یک فهرست کاری فعلی پیدا کنید پیدا کنید.

```
# find .-name hosein .txt
```

همه فایل ها را در زیر فهرست /home با نام hosein .txtپیدا میکند.

```
# find / home-name hosein .txt
/home/ hosein .txt
```

تمام فایل هایی که اسم hosein .txtرا پیدا کرده که حاوی حروف بزرگ و کوچک در فهرست /home است.

```
# find / home-name hosein .txt
./ hosein .txt
./ hosein .txt
```

یافتن راهنماها با استفاده از نام

یافتن همه فهرست هایی که نام آن hosein و در فهرست ریشه است.

```
# find / -type d -name hosein
/ hosein
```

همه فایل های PHP با استفاده از نام

پیداکردن تمام فایل های phpکه نام آن hosein .phpدر یک فهرست کاری فعلی است.

```
# find .-type f -name hosein .php
./ hosein .php
```

تمام فایل های PHP در فهرست

همه فایل های phpرا در یک فهرست پیدا می کند.

```
# find .-type f -name “* .php”
./ hosein .php
./login.php
./index.php
```

یافتن فایل ها براساس مجوزهای آنها

جستجو فایل ها با مجوز ۷۷۷

تمام فایل هایی که مجوزها 777هستند پیدا می کند.

```
# find  . –type  f  -perm 0777  -print
```

فایل های SGID با مجوز ۶۴۴

تمام فایلهای bit SGID که مجوزها آنها 644است تنظیم شده را پیدا می کند.

```
 # find / -perm  2644
```

یافتن فایل های bit با ۵۵۱ مجوز

تمام فایل های مجموعه Sticky Bit را که اجازه دارند 551پیدا کنید پیدا می کند.

```
# find / -perm 1551
```

همه فایل های SUID را پیدا می کند.

```
# find / -perm / u = s
```

فایل هایSGID

تمام فایل های مجموعه SGID

```
# find / -perm / g = s
```

فقط فایل های قابل خواندن[۱] را پیدا می کند.

```
# find / -perm / u = r
```

یافتن فایل های اجرایی

همه فایلهای اجرایی را پیدا میکند.

```
# find / -perm / a = x
```

فایل ها را با ۷۷۷ مجوز و Chmod به ۶۴۴ پیدا میکند.

```
 # find / -type f -perm 0777 -print -exec chmod 644 {} \;
```

یافتن راهنماها با مجوز ۷۷۷ و تغییر مجوز به ۷۵۵

```
 # find / -type d -perm 777 -print  -exec chmod 755 {} \;
```

جستجو یک فایل به نام .txt hoseinو حذف آن.

```
# find .-type f -name “ hosein .txt” -exec rm -f {} \;
```

فایل چندگانه را پیدا کنید و حذف کنید.

برای find و حذف چندین فایل مانند .mp3یا .txt، سپس از آن استفاده کنید.

```
# find .-type f -name “* .txt” -exec rm -f {} \;
```

یا

```
# find .-type f -name “* .mp3” -exec rm -f {} \;
```

تمام فایل های empty را پیدا کنید.

برای جستجو همه فایلهای خالی[۲] در مسیر خاص

```
# find / tmp -type f -type
```

همه راهنمایی های خالی را پیدا کنید.

---

[۱] read only

[۲] empty file

```
# find / tmp -type d-empty
```

برای جستجو همه فایل های مخفی[۱]، از دستور زیر استفاده کنید.

```
# find / tmp -type f -name “. *”
```

فایل های جستجو بر اساس مالکیت و گروه ها

برای جستجو همه یا یک فایل تک به نام .hosein txtزیر /root با مالکیت root پیدا میکند.

```
# find / -user root -name  hosein .txt
```

جستجو همه فایل هایی که متعلق به hosein کاربر زیر فهرست / home هستند.

```
# find / home -user  hosein
```

برای جستجو همه فایل هایی که متعلق به گروه توسعه در زیر فهرست / home هستند.

```
# find / home -group developer
```

برای جستجو همه فایل های .txtاز کاربر hosein زیر فهرست /home .

```
# find / home -user  hosein  -name “* .txt”
```

جستجو فایل ها و راهنماها بر اساس تاریخ و زمان

جستجو تمام فایل هایی که به مدت 50روز به آنها دسترسی پیدا شده است.

```
# find / -atme 50
```

جستجو تمام فایل هایی که بیش از 50روز قبل و کمتر از 100روز تغییر کرده اند.

```
# find / -mtime +50 -mtime -100
```

جستجو همه فایل هایی که در 1ساعت گذشته تغییر کرده اند.

```
# find / -cmin -60
```

جستجو همه فایل هایی که در 1ساعت گذشته اصلاح شده اند.

```
# find / -mmin -60
```

جستجو همه فایل هایی که در 1ساعت گذشته دیده می شوند.

```
# find / -amin -60
```

فایل ها و راهنماها بر اساس اندازه

جستجو تمام فایل های 50مگابایتی

```
# find / -size 50M
```

جستجو همه فایل هایی که بیش از 50مگابایت و کمتر از 100MBهستند.

```
# find / -size + 50M-size -100M
```

جستجو همه فایل های 100MBو حذف آنها با استفاده از یک دستور تنها.

```
# find / -size + 100M -exec rm -rf {} \;
```

جستجو همه فایل های .mp3را با بیش از 10مگابایت پیدا میکند و آنها را با استفاده از دستور rm[۲]حذف میکند.

---

[۱] hidden file

[۲] remove

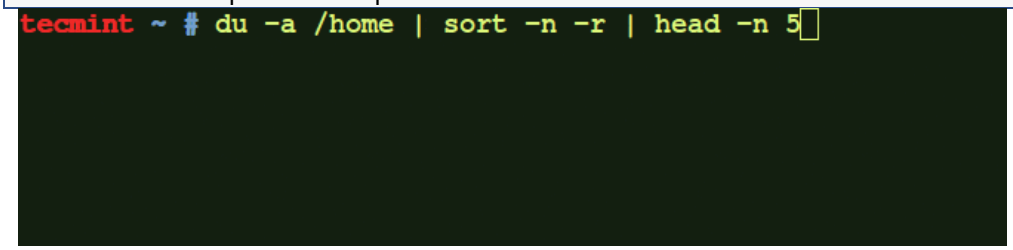
```
# find / -type f -name * .mp3 -size + 10M -exec rm {} \;
```

به عنوان یک مدیر linux، باید به صورت دوره ای بررسی کنید که چه فایل ها و فهرست ها فضای دیسک بیشتری را مصرف می کنند.بسیار مهم است که junks های غیر ضروری را پیدا کنید و آنها را از هارد دیسک خود آزاد کنید.

نحوه یافتن بزرگترین فایل ها و راهنماها در linux

دستور زیر را برای find بزرگترین بزرگترین فهرست زیر /homeپارتیشن / home.

```
# du -a / home | sort -n -r | head -n 5
```



مدیریت شبکه

۶۷.دستور hostname

از این دستور استفاده می شود تا به راحتی دستگاه را در یک شبکه در فرمت قابل خواندن انسان تشخیص دهد.جای تعجب نیست، اما در linux system، نام host را می توان به راحتی با استفاده از دستور ساده به عنوان " نام میزبان "تغییر داد

گزینه های دستور:

- A تمام نام های میزبان (aliasرا نمایش می‌دهد.
- d نام دامنه (domain nameرا نمایش می‌دهد.
- f نام فایل حاوی نام های میزبان را مشخص می‌کند.
- i آدرس IP سیستم را نمایش می‌دهد.
- s نام میزبان را تنظیم می‌کند.
- v اطلاعات نسخه و جزئیات مربوط به دستور را نمایش می‌دهد.
- h راهنمای دستور را نمایش می‌دهد.

مثال های دستور:

نام hostخود را به بدون هیچ پارامتری، نام hostsystem عامل linux خود را به این صورت نشان می دهد:

```
$ hostname
hosein
```

اگر میخواهید نام linux hostsystem خود را تغییر دهید یا تنظیم کنید، به سادگی اجرا کنید:

```
$ hostname NEW_HOSTNAME
```

البته، باید "NEW_HOSTNAME"را با نام hostواقعی که می خواهید تنظیم کنید، جایگزین کنید.این نام hostsystem را بلافاصله تغییر می دهد، اما یک مشکل وجود دارد – نام اصلی اصلی بعد از restart دوباره بازیابی خواهد شد. راه دیگری برای تغییر نام hostsystem وجود دارد که به طور دائمی است. ممکن است قبلا آن را کشف کرده باشید که در برخی از فایل های تنظیمات نیاز به تغییر دارد.

نمایش نام میزبان:

```
hostname
```

نمایش تمام نام های میزبان:

```
hostname -A
```

نمایش نام دامنه:

```
hostname -d
```

تنظیم نام میزبان به “my-hostname”:

```
hostname -s my-hostname
```

نمایش آدرس IP:

```
hostname -i
```

نمایش اطلاعات نسخه:

```
hostname -v
```

نمایش راهنمای دستور:

```
hostname -h
```

۶۸.دستور ip

دستور ip در net-tools وجود دارد که برای انجام چندین کار مدیریت شبکه استفاده می شود. IP مخفف Internet Protocol است. این دستور برای نمایش یا دستکاری مسیریابی، دستگاه ها و تونل ها استفاده می شود. می‌تواند کارهایی مانند پیکربندی و اصلاح مسیریابی پیش‌فرض و ثابت، راه اندازی تونل روی IP، فهرست کردن آدرس‌های IP و اطلاعات دارایی، تغییر وضعیت رابط، تخصیص، حذف و تنظیم آدرس‌ها و مسیرهای IP را انجام دهد. همچنین دستور ip ابزاری قدرتمند در سیستم عامل لینوکس برای پیکربندی و مدیریت رابط های شبکه، آدرس های IP و روتینگ است. این دستور جایگزینی برای دستورات قدیمی تر مانند ifconfig و route است و امکانات و انعطاف پذیری بیشتری را ارائه می دهد.

ساختار دستور:

```
$ ip [ OPTIONS ] OBJECT { COMMAND | help }
```

گزینه های دستور:

تنظیمات رابط

- a نمایش اطلاعات مربوط به همه رابط های شبکه.
- l نمایش اطلاعات مربوط به رابط های شبکه به صورت لیست.
- s نمایش آمار مربوط به رابط های شبکه.
- u رابط شبکه را فعال می کند.
- d رابط شبکه را غیرفعال می کند.
- up رابط شبکه را بالا می آورد.
- down رابط شبکه را پایین می آورد.
- link set وضعیت لینک رابط شبکه را تنظیم می کند.
- address add آدرس IP را به رابط شبکه اضافه می کند.
- address del آدرس IP را از رابط شبکه حذف می کند.

تنظیمات روتینگ

- route add مسیر را به جدول روتینگ اضافه می کند.
- route del مسیر را از جدول روتینگ حذف می کند.
- rule add قانون را به جدول روتینگ اضافه می کند.
- rule del قانون را از جدول روتینگ حذف می کند.

سایر گزینه ها

- h نمایش راهنمای دستور.
- V نمایش نسخه دستور.

مثال های دستور ip

نمایش نسخه ip گزینه – V:

```
$ ip -V
```

نمایش راهنمای دستور ip گزینه – h:

```
$ ip -h
```

نمایش آمار شبکه گزینه –s x:

```
$ ip -s
```

نمایش جزئیات بیشتر گزینه – d:

```
$ ip -d
```

رزولوشن آدرس‌ها به نام گزینه – r:

```
$ ip -r
```

استفاده از واحدهای IEC برای نمایش گزینه – iec:

```
$ ip -iec
```

استفاده از خانواده آدرس IPv4 گزینه – f inet x:

```
$ ip -f inet
```

استفاده از فرمت JSON برای خروجی گزینه – j:

```
$ ip -j
```

نمایش اطلاعات در یک خط گزینه – o:

```
$ ip -o
```

نمایش اطلاعات به صورت خلاصه گزینه – br:

```
$ ip -br
```

گزینه – 4: نمایش فقط آدرس‌های IPv4

```
$ ip -4 address show
```

گزینه – 6: نمایش فقط آدرس‌های IPv6

```
$ ip -6 address show
```

گزینه – B: نمایش جداول پهنای باند

```
$ ip -B link show
```

گزینه – : 0 نمایش جداول پهنای باند در حالت صفر که نشان دهنده برداشتن پهنای باند است

```
$ ip -0 link show
```

گزینه – : l maximum-addr-flush-attempts تعیین تعداد بیشینه تلاش‌ها برای حذف آدرس‌ها در حالت oops

```
$ ip -l 3 addrflush
```

گزینه – : o نمایش خروجی در یک خط

```
$ ip -o link show
```

گزینه – : rcvbuf [size] تنظیم اندازه بافر ورودی جایگزین برای سوکت‌های rcvbuf

```
$ ip -rcvbuf 8192 link show
```

گزینه – : t نمایش زمان در خروجی

```
$ ip -t link show
```

گزینه – : ts نمایش زمان به صورت مختصر در خروجی

```
$ ip -ts link show
```

گزینه – : n name انتقال دستور به فضای نام مشخص شده

```
$ ip -n namespace link show
```

گزینه – : N نمایش آدرس‌ها به صورت عددی

```
$ ip -N address show
```

گزینه – : a نمایش همه موارد

```
$ ip -a link show
```

گزینه – : c استفاده از رنگ در خروجی

```
$ ip -c link show
```

گزینه – : br نمایش خروجی به صورت خلاصه

```
$ ip -br link show
```

گزینه – : j نمایش خروجی به صورت JSON

```
$ ip -j link show
```

گزینه – : p نمایش خروجی به صورت زیبا

```
$ ip -p link show
```

گزینه : link نمایش و مدیریت اطلاعات مربوط به لینک‌های شبکه

```
$ ip link show
$ ip link set dev eth0 up
```

گزینه : address نمایش و مدیریت اطلاعات آدرس‌های IP

```
$ ip address show
$ ip address add 192.168.1.10/24 dev eth0
```

گزینه : addrlabel نمایش و مدیریت برچسب‌های آدرس

```
$ ip addrlabel show
$ ip addrlabel add prefix 192.168.1.0/24 label 10
```

گزینه route : نمایش و مدیریت جداول مسیریابی

```
$ ip route show
$ ip route add default via 192.168.1.1 dev eth0
```

گزینه rule : نمایش و مدیریت قوانین مسیریابی

```
$ ip rule show
$ ip rule add from 192.168.1.0/24 table 10
```

گزینه neigh : نمایش و مدیریت جداول همسایگی(ARP/NDP)

```
$ ip neigh show
$ ip neigh add 192.168.1.1 lladdr 00:11:22:33:44:55 nud permanent dev eth0
```

گزینه ntable : نمایش و مدیریت جداول جاری شبکه

```
$ ip ntable show
$ ip ntable add table 10 type fib num 100
```

گزینه tunnel : نمایش و مدیریت تونل‌های شبکه

```
$ ip tunnel show
$ ip tunnel add tun0 mode gre remote 192.168.1.1 local 192.168.1.2
```

گزینه tuntap : نمایش و مدیریت دستگاه‌های TUN/TAP

```
$ ip tuntap show
$ ip tuntap add name tun0 mode tun
```

گزینه maddress : نمایش و مدیریت آدرس‌های چندگانه

```
$ ip maddress show
$ ip maddress add 239.0.0.1 dev eth0
```

گزینه mroute : نمایش و مدیریت مسیرهای چندگانه

```
$ ip mroute show
$ ip mroute add 239.0.0.1 dev eth0
```

گزینه mrule : نمایش و مدیریت قوانین چندگانه

```
$ ip mrule show
$ ip mrule add from 192.168.1.0/24 table 10
```

گزینه monitor : نمایش وضعیت مانیتورینگ

```
$ ip monitor
```

گزینه xfrm : نمایش و مدیریت امنیت و تونل‌ها

```
$ ip xfrm state show
$ ip xfrm policy add src 192.168.1.0/24 dst 10.0.0.0/24 dir out
```

گزینه netns : نمایش و مدیریت فضاهای نام شبکه

```
$ ip netns list
$ ip netns add mynamespace
```

گزینه l2tp : نمایش و مدیریت تونل‌های L2TP

```
$ ip l2tp show tunnel
$ ip l2tp add tunnel local 192.168.1.1 remote 192.168.2.1
```

گزینه tcp_metrics : نمایش و مدیریت معیارهای TCP

```
$ ip tcp_metrics show
$ ip tcp_metrics add 192.168.1.0/24 rtt 100ms
```

گزینه : token نمایش و مدیریت توکن‌ها

```
$ ip token show
$ ip token add dev eth0
```

گزینه : macsec نمایش و مدیریت امنیت MACsec

```
$ ip macsec show
$ ip macsec add dev eth0
```

گزینه : vrf نمایش و مدیریت - منابع مجازی شبکه

```
$ip vrf show
$ip vrf add myvrf
```

گزینه : mptcp نمایش و مدیریت - چندگانه پروتکل کنترل انتقال

```
$ip mptcp show
$ip mptcp add subflow dev eth0
```

گزینه : ioam نمایش و مدیریت –مانیتورینگ آگاه از مسیر

```
$ip ioam show
$ip ioam add hoplimit 64
```

گزینه : stats نمایش آمارهای شبکه

```
$ip -s link show
$ip -s route show
```

۶۹.دستور ufw

دستور ufw در سیستم‌عامل‌های Ubuntu و دیگر توزیع‌های Linux برای مدیریت فایروال ((firewallاستفاده می‌شود. این دستور به کاربر امکان می‌دهد تا قوانین و تنظیمات مربوط به فایروال را تعریف و مدیریت کند تا امنیت سیستم خود را بهبود دهد. همچنین دستور ufw (مخفف Uncomplicated Firewall) برای پیکربندی فایروال در سیستم عامل های لینوکس مبتنی بر iptables به صورت آسان و کاربرپسند استفاده می‌شود. این ابزار به شما امکان می‌دهد تا قوانین فایروال را به صورت ساده و بدون نیاز به دانش عمیق iptablesایجاد و مدیریت کنید.

گزینه های دستور:

- h نمایش راهنمای دستور
- v نمایش نسخه ufw
- f فعال کردن ufw
- d غیرفعال کردن ufw
- s نمایش وضعیت فعلی ufw
- a اضافه کردن یک قانون جدید
- i اضافه کردن یک قانون جدید به ابتدای لیست قوانین
- d حذف یک قانون
- r جایگزینی یک قانون
- l نمایش لیست قوانین
- n نمایش شماره خط قوانین
- c پاک کردن تمام قوانین
- p پروتکل مورد نظر برای قانون (tcp, udp, icmp, etc.))

- m ماژول iptables مورد نظر برای قانون ((state, limit, etc.
- s آدرس IP مبدا
- d آدرس IP مقصد
- sport پورت مبدا
- dport پورت مقصد
- t نام جدول iptables برای قانون
- j عمل مورد نظر برای قانون ((ACCEPT, DROP, REJECT, etc.
- نام رابط شبکه برای قانون
- v تنظیم سطح verbosity

مثال های دستور:

فعال کردن ufw:

```
ufw enable
```

غیرفعال کردن ufw:

```
ufw disable
```

نمایش وضعیت فعلی ufw:

```
ufw status
```

اضافه کردن قانون جدید برای اجازه ورود به پورت ۲۲ SSH:

```
ufw allow 22
```

اضافه کردن قانون جدید به ابتدای لیست قوانین برای اجازه ورود به پورت ۸۰ HTTP:

```
ufw insert 1 allow 80
```

حذف قانون اجازه ورود به پورت ۲۵ SMTP:

```
ufw delete allow 25
```

جایگزینی قانون اجازه ورود به پورت ۸۰ HTTP با قانون جدیدی که فقط از آدرس IP 192.168.1.0/24 اجازه ورود می‌دهد:

```
ufw replace allow 80 from 192.168.1.0/24
```

نمایش لیست قوانین:

```
ufw list
```

نمایش لیست قوانین با شماره خط:

```
ufw list -n
```

پاک کردن تمام قوانین:

```
ufw reset
```

اجازه ورود به تمام ترافیک از رابط شبکه eth0:

```
ufw allow out on eth0
```

مسدود کردن تمام ترافیک به پورت ۲۵ SMTP:

```
ufw reject 25
```

نمایش راهنمای کامل:

```
ufw --help
```

۷۰.دستور tcpdump

دستور tcpdump در لینوکس برای بررسی و ضبط ترافیک شبکه استفاده می‌شود. این دستور امکان نمایش و ذخیره بسته‌های شبکه را فراهم می‌کند تا بتوانید بررسی‌های مختلفی روی ترافیک شبکه انجام دهید.

گزینه‌های دستور:

- h نمایش راهنمای دستور
- v نمایش نسخه tcpdump
- i نام رابط شبکه مورد نظر برای ضبط ترافیک
- p فیلتر کردن بر اساس پروتکل (tcp, udp, icmp, etc.)
- s تعداد بایت های دریافتی برای هر بسته
- w ذخیره ترافیک ضبط شده در فایل
- r خواندن ترافیک ضبط شده از فایل
- n عدم نمایش نام های دامنه و آدرس های IP به صورت عددی
- A نمایش اطلاعات مربوط به تمام پروتکل ها
- t عدم نمایش timestamp در خروجی
- c تعداد بسته های دریافتی قبل از توقف
- F استفاده از فیلتر BPF
- X نمایش اطلاعات مربوط به لایه های مختلف پروتکل
- l عدم ارسال بسته ها به لایه های بالاتر
- q عدم نمایش اطلاعات اضافی
- d رمزگشایی پروتکل های مختلف
- e رمزگشایی فیلدهای خاص در پروتکل ها
- vvv نمایش اطلاعات مربوط به جزئیات

مثال‌های دستور:

ضبط تمام ترافیک شبکه:

```
tcpdump
```

ضبط ترافیک شبکه فقط از رابط eth0:

```
tcpdump i eth0
```

ضبط ترافیک شبکه فقط از پروتکل TCP:

```
tcpdump p tcp
```

ضبط ترافیک شبکه فقط از پروتکل UDP:

```
tcpdump p udp
```

ضبط ترافیک شبکه فقط از پروتکل ICMP:

```
tcpdump p icmp
```

ضبط ۱۰۰ بایت اول هر بسته:

```
tcpdump s 100
```

ذخیره ترافیک ضبط شده در فایل tcpdump.pcap:

```
tcpdump w tcpdump.pcap
```

خواندن ترافیک ضبط شده از فایل tcpdump.pcap:

```
tcpdump r tcpdump.pcap
```

نمایش نام های دامنه و آدرس های IP به صورت عددی:

```
tcpdump n
```

نمایش اطلاعات مربوط به تمام پروتکل ها:

```
tcpdump A
```

عدم نمایش timestamp در خروجی:

```
tcpdump t
```

دریافت ۱۰۰ بسته و سپس توقف:

```
tcpdump c 100
```

ستفاده از فیلتر BPF برای فقط نمایش بسته های TCP با پورت ۸۰:

```
tcpdump F “tcp port 80”
```

نمایش اطلاعات مربوط به لایه های مختلف پروتکل:

```
tcpdump X
```

دم ارسال بسته ها به لایه های بالاتر:

```
tcpdump l
```

عدم نمایش اطلاعات اضافی:

```
tcpdump q
```

رمزگشایی پروتکل HTTP:

```
tcpdump d “tcp port 80”
```

رمزگشایی فیلد source IP در پروتکل IP:

```
tcpdump e “ip.src”
```

.نمایش اطلاعات مربوط به جزئیات

ضبط ترافیک شبکه فقط از رابط eth1 و فقط از پروتکل TCP با پورت ۸۰:

```
tcpdump i eth1 p tcp port 80
```

ضبط ترافیک شبکه فقط از رابط eth0 و فقط از پروتکل UDP با پورت ۵۳:

```
tcpdump i eth0 p udp port 53
```

ضبط ترافیک شبکه فقط از رابط eth1 و فقط از پروتکل ICMP:

```
tcpdump i eth1 p icmp
```

۷۱.دستور Wget

قصد داریم ابزار wgetرا بررسی کنیم که فایل ها را از وب سایت جهانی (WWW) دانلود می کند. با استفاده از پروتکل های گسترده مانند HTTP، HTTPSو FTP.ابزار Wgetبسته آزادانه موجود است و مجوز تحت مجوز GNU GPL است.این ابزار می تواند هر system عامل یونیکس مانندWindowsو MAC OSرا install کند.این یک ابزار خط دستور غیر تعاملی است.ویژگی اصلی Wget آن استحکام است.این به گونه ای طراحی شده است که در اتصالات شبکه آهسته یا ناپایدار شبکه کار کند.

Wgetبارگیری کننده شبکه غیرتعاملی است که برای بارگیری پرونده ها از سرور استفاده می شود حتی اگر کاربر به سیستم وارد نشده باشد و بدون اینکه مانع روند فعلی شود می تواند در پس زمینه کار کند. به صورت خودکار شروع به download میکند و در صورت مشکل در شبکه، آن را لغو میکند .

- GNU wgetیک ابزار آزاد برای بارگیری غیر تعاملی پرونده ها از وب است. از پروتکل های HTTP ، HTTPSو FTPو همچنین بازیابی از طریق پروکسی های HTTP پشتیبانی می کند.
- wgetغیر تعاملی است ، به این معنی که می تواند در پس زمینه کار کند ، در حالی که کاربر به سیستم وارد نشده است. با این کار می توانید بازیابی را شروع کرده و از سیستم جدا شوید و به wget اجازه دهید کار را تمام کند. در مقابل ، بیشتر مرورگرهای وب به حضور مداوم کاربر نیاز دارند ، که می تواند مانع بزرگی در هنگام انتقال اطلاعات زیادی باشد.
- wgetمی تواند پیوندها را در صفحات HTML و XHTML دنبال کند و نسخه های محلی وب سایت های از راه دور را ایجاد کند و ساختار فهرست سایت اصلی را به طور کامل بازسازی کند. گاهی اوقات از این به عنوان بارگیری بازگشتی یاد می شود. در حالی که این کار را انجام می دهید ، wgetبه استاندارد حذف Robot (/robots.txt) احترام می گذارد. به wgetمی توان دستور داد که پیوندهای موجود در پرونده های HTML بارگیری شده را به پرونده های محلی برای مشاهده آفلاین تبدیل کند.
- wgetبرای مقاومت در برابر اتصالات شبکه کند یا ناپایدار طراحی شده است. اگر بارگیری به دلیل مشکلی در شبکه انجام نشود ، تا زمانی که کل پرونده بازیابی نشود ، دوباره در تلاش است. اگر سرور از سرگیری پشتیبانی کند ، به سرور دستور می دهد تا بارگیری را از جایی که متوقف شده است ادامه دهد.

ساختار دستور:

```
wget [گزینه] [URL]
```

مثال های دستور:

برای بارگیری ساده یک صفحه وب:

```
wget http://example.com/sample.php
```

برای بارگیری پرونده در پس زمینه

```
wget -b http://www.example.com/samplepage.php
```

برای بازنویسی log wile دستورwget

```
wget http://www.example.com/filename.txt -o /path/filename.txt
```

برای از سرگیری پرونده ای که بارگیری شده است

```
wget -c http://example.com/samplefile.tar.gz
```

تعداد دفعات مشخصی را امتحان کنید

```
wget --tries = 10 http://example.com/samplefile.tar.gz
```

گزینه های دستور:

-v / –version : این برای نمایش نسخه wget موجود در سیستم شما استفاده می شود.

```
$ wget -v
```

-h / –help : این برای چاپ یک پیام **راهنما** استفاده می شود که تمام گزینه های ممکن از دستور line را که با گزینه های خط فرمان wget در دسترس است، نشان می دهد.

```
$ wget -h [URL]
```

-o logfile : از این گزینه برای هدایت کلیه پیام های تولید شده توسط سیستم به سمت پرونده ای که توسط گزینه مشخص شده است هدایت می شود و هنگامی که روند کار به پایان می رسد ، تمام پیام های تولید شده در پرونده ورود به سیستم موجود است. اگر پرونده ورود به سیستم مشخص نشده باشد، پیام های خروجی به پرونده ورود به سیستم پیش فرض یعنی wget -log هدایت می شوند

```
$ wget -o logfile [URL]
```

-b / –background : این گزینه برای ارسال فرآیند به پس زمینه به محض شروع فرآیند استفاده می شود تا فرایندهای دیگر انجام شوند. اگر هیچ پرونده خروجی از طریق گزینه o- مشخص نشده باشد ، خروجی به طور پیش فرض به wget-log هدایت می شود.

```
$ wget -b [URL]
```

–a : از این گزینه برای ضمیمه کردن پیام های خروجی به پرونده ورودی خروجی فعلی بدون رونویسی پرونده به عنوان در گزینه o- ، پرونده ورودی خروجی رونویسی می شود اما با استفاده از این گزینه ، ثبت دستور قبلی ذخیره می شود و ثبت فعلی بعد از قبلی ثبت می شود.

۷۲.دستور ping

ping یا Packet Internet Groper برای بررسی اتصال شبکه بین میزبان و سرور/میزبان استفاده می شود. این دستور آدرس IP یا URL را به عنوان ورودی می گیرد و یک بسته داده را با پیام PING به آدرس مشخص شده ارسال می‌کند و پاسخی از سرور/میزبان دریافت می‌کند که این بار ثبت می شود که به آن تاخیر می گویند. پینگ از ICMP (پروتکل پیام کنترل اینترنت) برای ارسال پیام اکو ICMP به میزبان مشخص شده در صورت موجود بودن آن میزبان استفاده می‌کند، سپس پیام پاسخ ICMP را ارسال می‌کند. پینگ معمولاً در میلی ثانیه اندازه گیری می شود که هر سیستم عامل مدرنی این پینگ را از قبل نصب کرده باشد.

ساختار دستور:

دستور پینگ، به دنبال نام میزبان، نام یک وب سایت یا آدرس IP دقیق است.

```
$   ping [option] [hostname] or [IP address]
```

گزینه های دستور:

- h نمایش راهنمای دستور
- v نمایش نسخه ping
- n تعداد بسته های ارسالی
- c تعداد دفعات ارسال بسته ها
- i فاصله زمانی بین ارسال بسته ها (بر حسب ثانیه)
- t ارسال مداوم بسته ها تا زمانی که کاربر با Ctrl+C متوقف کند
- s اندازه بسته های ارسالی (بر حسب بایت)
- f عدم تکه تکه شدن بسته ها
- p مشخص کردن پروتکل مورد نظر برای ارسال بسته ها (tcp, udp, icmp, etc.)
- m مشخص کردن TTL بسته های ارسالی
- Q فعال کردن timestamp در بسته های ارسالی
- l فعال کردن timestamp در پاسخ های دریافتی
- A فعال کردن timestamp در هر دو بسته های ارسالی و دریافتی

- T فعال کردن timestamp دقیق در پاسخ های دریافتی
- ۴ استفاده از پروتکل IPv4
- ۶ استفاده از پروتکل IPv6
- r فعال کردن مسیریابی معکوس
- D فعال کردن Do Not Fragment (DF) bit در بسته های ارسالی
- M فعال کردن Record Route (RR) option در بسته های ارسالی
- S فعال کردن Source Specific Routing (SSR) option در بسته های ارسالی
- فعال کردن Timestamp option در بسته های ارسالی
- t فعال کردن Traceroute option در بسته های ارسالی
- a فعال کردن IP Address Resolution (ARP)
- b فعال کردن Broadcast ping
- g فعال کردن Loose Source Routing (LSR) option در بسته های ارسالی
- I مشخص کردن رابط شبکه مورد نظر برای ارسال بسته ها
- w حداکثر زمان انتظار برای پاسخ (بر حسب ثانیه)
- Q فعال کردن Quiet mode
- v فعال کردن Verbose mode

مثال های دستور:

۱. نمایش راهنمای دستور:

```
ping -h
```

۲. نمایش نسخه ping:

```
ping -v
```

۳. ارسال ۵ بسته ping به آدرس IP 192.168.1.100:

```
ping 192.168.1.100 -n 5
```

۴. ارسال ۱۰ بسته ping به آدرس IP 192.168.1.100 با فاصله زمانی ۲ ثانیه بین هر بسته:

```
ping 192.168.1.100 -c 10 -i 2
```

۵. ارسال مداوم بسته های ping به آدرس IP 192.168.1.100 تا زمانی که کاربر با Ctrl+C متوقف کند:

```
ping 192.168.1.100 -t
```

۶. ارسال بسته های ping به آدرس IP 192.168.1.100 با اندازه ۱۰۲۴ بایت:

```
ping 192.168.1.100 -s 1024
```

۷. عدم تکه تکه شدن بسته های ارسالی:

```
ping 192.168.1.100 -f
```

۸. ارسال بسته های ping به آدرس IP 192.168.1.100 با استفاده از پروتکل UDP:

```
ping 192.168.1.100 -p udp
```

۹. ارسال بسته های ping به آدرس IP 192.168.1.100 با TTL 5:

```
ping 192.168.1.100 -m 5
```

۱۰. فعال کردن timestamp در بسته های ارسالی:

```
ping 192.168.1.100 -Q
```

۱۱. فعال کردن timestamp در پاسخ های دریافتی:

```
ping 192.168.1.100 -l
```

۱۲. فعال کردن timestamp در هر دو بسته های ارسالی و دریافتی:

```
ping 192.168.1.100 -A
```

۱۳. فعال کردن timestamp دقیق در پاسخ های دریافتی:

```
ping 192.168.1.100 -T
```

۱۴. استفاده از پروتکل IPv4 برای ارسال بسته های ping:

```
ping 192.168.1.100 -4
```

۱۵. استفاده از پروتکل IPv6 برای ارسال بسته های ping:

```
ping 192.168.1.100 -6
```

۱۶. فعال کردن مسیریابی معکوس:

```
ping 192.168.1.100 -r
```

۱۷. فعال کردن Do Not Fragment (DF) bit در بسته های ارسالی:

```
ping 192.168.1.100 -D
```

۱۸. فعال کردن Record Route (RR) option در بسته های ارسالی:

```
ping 192.168.1.100 -M
```

۱۹. فعال کردن Source Specific Routing (SSR) option در بسته های ارسالی:

```
ping 192.168.1.100 -S
```

۲۰. فعال کردن Timestamp option در بسته های ارسالی:

```
ping 192.168.1.100 -O
```

۲۱. فعال کردن Traceroute option در بسته های ارسالی:

```
ping 192.168.1.100 -t
```

۲۲. فعال کردن IP Address Resolution (ARP):

```
ping 192.168.1.100 -a
```

۲۳. فعال کردن Broadcast ping:

```
ping 192.168.1.255 -b
```

۲۴. فعال کردن Loose Source Routing (LSR) option در بسته های ارسالی:

```
ping 192.168.1.100
```

۷۳.دستور nmap

دستور nmap یک ابزار اسکن شبکه قدرتمند است که در سیستم‌عامل‌های مختلف مانند لینوکس و ویندوز قابل استفاده است. این ابزار به شما کمک می‌کند تا اطلاعات مربوط به دستگاه‌ها، پورت‌ها، سرویس‌ها و تنظیمات امنیتی شبکه را بررسی کنید.

گزینه های دستور

گزینه های اسکن

- a نمایش میانگین معیارها در هر ثانیه
- dstat a n 10 t 1 cpu,mem,net
- این دستور ۱۰ بار معیارهای مربوط به CPU، حافظه و شبکه را با فاصله زمانی ۱ ثانیه جمع آوری و میانگین آنها را در هر ثانیه نمایش می دهد.
- c نمایش معیارها به صورت نمودار
- dstat c t 2 cpu,mem
- این دستور معیارهای مربوط به CPU و حافظه را به صورت نمودار با فاصله زمانی ۲ ثانیه نمایش می دهد.
- g نمایش معیارها به صورت گرافیکی
- dstat g t 3 cpu,disk
- این دستور معیارهای مربوط به CPU و دیسگزینه های اسکن
- sT اسکن) TCP SYNنیمه باز)
- sS اسکن TCP SYN مخفی
- sU اسکن UDP
- sF اسکن TCP FIN
- sX اسکن TCP Xmas
- sN اسکن TCP Null
- A اسکن جامع (شامل تمام اسکن های (TCP

گزینه های هدف

- h آدرس IP یا نام هاست
- n اسکن بدون نامگذاری (فقط آدرس (IP
- p پورت یا محدوده پورت برای اسکن
- exclude آدرس IP یا نام هاست برای حذف از اسکن
- range محدوده آدرس IP برای اسکن

گزینه های خروجی

- v افزایش جزئیات خروجی
- vv افزایش بیشتر جزئیات خروجی
- oX ذخیره خروجی در فرمت XML
- oG ذخیره خروجی در فرمت Greppable
- oA ذخیره خروجی در تمام فرمت های پیش فرض

گزینه های پیشرفته

- T زمان انتظار برای پاسخ (بر حسب ثانیه)
- r تعداد دفعات ارسال درخواست
- version نمایش نسخه Nmap
- h نمایش راهنمای دستور

مثال های دستور:

گزینه های اسکن

sT اسکن TCP SYNنیمه باز)

```
nmap -sT 192.168.1.100
```

این دستور هاست ۱۹۲٫۱۶۸٫۱٫۱۰۰ را با اسکن) TCP SYNنیمه باز) اسکن می کند.

sS اسکن TCP SYN مخفی

```
nmap -sS example.com
```

این دستور هاست example.com را با اسکن TCP SYN مخفی اسکن می کند.

sU اسکن UDP

```
nmap -sU -p 53 192.168.1.0/24
```

این دستور تمام هاست های موجود در شبکه ۱۹۲٫۱۶۸٫۱٫۰/۲۴ را برای پورت UDP 53 (DNS) اسکن می کند.

sF اسکن TCP FIN

```
nmap -sF -p 80,443 10.0.0.1
```

این دستور پورت های ۸۰ و ۴۴۳ را در هاست ۱۰٫۰٫۰٫۱ با اسکن TCP FIN اسکن می کند.

sX اسکن TCP Xmas

```
nmap -sX 192.168.1.1-10
```

این دستور هاست های ۱۹۲٫۱۶۸٫۱٫۱ تا ۱۹۲٫۱۶۸٫۱٫۱۰ را با اسکن TCP Xmas اسکن می کند.

sN اسکن TCP Null

```
nmap -sN example.com -p 22
```

این دستور پورت ۲۲ را در هاست example.com با اسکن TCP Null اسکن می کند.

A اسکن جامع (شامل تمام اسکن های TCP)

```
nmap -A 192.168.1.100
```

این دستور هاست ۱۹۲٫۱۶۸٫۱٫۱۰۰ را با اسکن جامع (شامل تمام اسکن های TCP)اسکن می کند.

گزینه های هدف

h آدرس IP یا نام هاست

```
nmap -h 192.168.1.100
```

این دستور اطلاعات مربوط به هاست ۱۹۲٫۱۶۸٫۱٫۱۰۰ را نمایش می دهد.

n اسکن بدون نامگذاری (فقط آدرس IP)

```
nmap -n 10.0.0.1-254
```

این دستور تمام هاست های موجود در محدوده ۱۰٫۰٫۰٫۱ تا ۱۰٫۰٫۰٫۲۵۴ را بدون نامگذاری اسکن می کند.

p پورت یا محدوده پورت برای اسکن

```
nmap -p 80-85 example.com
```

این دستور پورت های ۸۰ تا ۸۵ را در هاست example.com اسکن می کند.

exclude آدرس IP یا نام هاست برای حذف از اسکن

```
nmap -A 192.168.1.0/24 --exclude 192.168.1.100
```

این دستور تمام هاست های موجود در شبکه ۱۹۲,۱۶۸,۱,۰/۲۴ را به جز هاست ۱۹۲,۱۶۸,۱,۱۰۰ با اسکن جامع اسکن می کند.

range محدوده آدرس IP برای اسکن

```
nmap -sS --range 192.168.1.1-10
```

این دستور هاست های ۱۹۲,۱۶۸,۱,۱ تا ۱۹۲,۱۶۸,۱,۱۰ را با اسکن TCP SYN مخفی اسکن می کند.

گزینه های خروجی

v افزایش جزئیات خروجی

```
nmap -v example
```

۷۴.دستور ssh

دستور ssh در لینوکس مخفف "Secure Shell" است. این یک پروتکل است که برای اتصال ایمن به سرور/سیستم راه دور استفاده می شود. ssh امن تر است به این معنا که داده ها را به صورت رمزگذاری شده بین میزبان و مشتری منتقل می‌کند. ssh در پورت TCP/IP 22 اجرا می شود.

ساختار دستور:

```
$ ssh user_name@host(IP/Domain_Name)
$ ssh -i private.key user_name@host
```

اجزای دستور

- options گزینه های مختلف SSH
- user نام کاربری برای ورود به سیستم
- hostname آدرس IP یا نام هاست کامپیوتر مقصد
- command دستوری که می خواهید در کامپیوتر مقصد اجرا کنید (اختیاری)

گزینه های دستور:

- p پورت SSHطپیش فرض
- i مسیر فایل کلید خصوصی SSH
- v افزایش سطح گزارش دهی
- q کاهش سطح گزارش دهی
- C فعال کردن فشرده سازی داده ها
- X ارسال X11 forwarding
- A فعال کردن agent forwarding
- t اتصال به صورت tty
- l نام کاربری برای ورود به سیستم (پیش فرض نام کاربری فعلی)
- g گروه های کاربری برای ورود به سیستم
- تنظیمات سفارشی SSH

مثال های دستور ssh

۱.از یک شماره پورت متفاوت برای اتصال SSH استفاده کنید:

```
$ ssh test.server.com -p 3322
```

۲. –i ssh به سرور راه دور با استفاده از کلید خصوصی

```
$ ssh -i private.key user_name@host
```

۳. –l ssh نام کاربری دیگری را مشخص می‌کند

```
$ ssh -l alternative-username sample.ssh.com
```

مثال های بیشتر ssh

گزینه –4 استفاده از IPv4 برای اتصال.

```
$ssh -4 remote_host
```

گزینه –6 :استفاده از IPv6 برای اتصال.

```
$ssh -6 remote_host
```

گزینه –A :فعال کردن فراهم کردن هویت(Agent Forwarding) .

```
$ssh -A remote_host
```

گزینه –a :غیرفعال کردن فراهم کردن هویت(Agent Forwarding) .

```
$ssh -a remote_host
```

گزینه –C :فشرده‌سازی ترافیک.

```
$ssh -C remote_host
```

گزینه –f :اجرای ssh به صورت پس‌زمینه(background) .

```
$ssh -f remote_host
```

گزینه –G :چاپ تنظیمات پیکربندی متصل شده بدون برقراری اتصال.

```
$ssh -G remote_host
```

گزینه –g :فعال کردن فروادینگ ترافیک(Traffic forwarding) .

```
$ssh -g remote_host
```

گزینه –K :فعال کردن مدیریت کلید (GSSAPI Key Exchange) با تمام پشتیبانی ها.

```
$ssh -K remote_host
```

گزینه –k :غیرفعال کردن مدیریت کلید(GSSAPI Key Exchange) .

```
$ssh -k remote_host
```

گزینه –M :فعال کردن حالت تقاطعی چند جلسه(Multiplex Master Mode) .

```
$ssh -M remote_host
```

گزینه –N :عدم اجرای دستور پس از برقراری اتصال.

```
$ssh -N remote_host
```

گزینه –n :عدم خواندن ورودی ازSTDIN .

```
$ssh -n remote_host
```

گزینه –q :حالت سکوت(Quiet mode) .

```
$ssh -q remote_host
```

گزینه –s :فعال کردن ترمینال محلی برای اتصال.

```
$ssh -s remote_host
```

گزینه –T :غیرفعال کردن ترمینال پس از برقراری اتصال.

```
$ssh -T remote_host
```

گزینه –V :چاپ نسخهssh .

```
$ssh -V
```

گزینه –v :حالت) verbose تفصیلی.

```
$ssh -v remote_host
```

۷۵.دستور lpinfo

دستور lpinfo در سیستم‌های Unix/Linux برای نمایش اطلاعات مربوط به سیستم چاپ‌گرها و چاپ‌سرورها استفاده می‌شود. این دستور به کاربران این امکان را می‌دهد تا اطلاعاتی مانند لیست چاپ‌گرهای موجود، ویژگی‌های آن‌ها، و تنظیمات مربوط به چاپ را مشاهده کنند.

گزینه های دستور:

استفاده از lpinfo

برای استفاده از lpinfo می توانید از دستور زیر استفاده کنید

```
lpinfo [options]
```

- h نمایش راهنمای دستور
- v نمایش اطلاعات
- l نمایش لیست چاپگرها
- p نمایش اطلاعات مربوط به یک چاپگر خاص
- c نمایش اطلاعات مربوط به یک کار چاپ خاص
- u نمایش اطلاعات مربوط به کارهای چاپ یک کاربر خاص

مثال های دستور:

نمایش لیست تمام چاپ‌گرهای موجود: این دستور برای نمایش لیست تمام چاپ‌گرهای موجود در سیستم استفاده می‌شود.

```
lpinfo -v
```

نمایش جزئیات یک چاپ‌گر خاص: این دستور برای نمایش جزئیات و ویژگی‌های یک چاپ‌گر خاص بر اساس شناسه آن استفاده می‌شود.

```
lpinfo -v -l -m
```

نمایش تنظیمات مربوط به چاپ: این دستور برای نمایش تنظیمات مربوط به چاپ و چاپ‌گرها استفاده می‌شود.

```
lpinfo -u
```

پاک کردن داده ها در linux

۷۶.دستور shred

دستور shred برای حذف امن فایل‌ها در سیستم‌عامل‌های یونیکس و شبه یونیکس مانند لینوکس و macOS استفاده می‌شود. این دستور با داده‌های فایل با الگوهای تصادفی، بازیابی اطلاعات را دشوار یا غیرممکن می‌کند.

ساختار دستور:

```
shred [options] file...
```

گزینه‌های دستور:

- f اجبار به حذف فایل، حتی اگر فقط خواندنی باشد.
- n تعداد دفعات überschreiben داده‌ها را مشخص می‌کند. پیش‌فرض ۳ بار است.
- u پس از حذف فایل، فضای دیسک را آزاد می‌کند.
- v اطلاعات مربوط به فرآیند حذف را نمایش می‌دهد.
- z قبل از überschreiben، فایل را با صفر پر می‌کند.
- الگوی überschreiben را مشخص می‌کند. پیش‌فرض الگوی DoD 5220.22M است.

مثال‌های دستور:

حذف امن یک فایل:

```
shred file.txt
```

حذف امن یک فایل و آزاد کردن فضای دیسک:

```
shred -u file.txt
```

حذف امن یک فایل با ۷ بار überschreiben:

```
shred -n 7 file.txt
```

حذف امن یک فایل با الگوی Gutmann:

```
shred -o gutmann file.txt
```

حذف امن تمام فایل‌های یک پوشه:

```
shred -v /path/to/folder/*
```

نکته:

استفاده از دستور shred می‌تواند زمان‌بر باشد، به خصوص برای فایل‌های بزرگ.

قبل از استفاده از دستور shred، از اینکه واقعاً می‌خواهید فایل را حذف کنید، مطمئن شوید.

مدیریت فایلهای آرشیو و پشتیبانی

۷۷.مدیریت فایلهای آرشیو دستور tar

برای آرشیو است که توسط بسیاری از مدیران linux system / یونیکس برای تهیه نسخه پشتیبان استفاده می شود.دستور tar به منظور جمع آوری فایلها و فهرستها در فایل آرشیو فشرده با نام تجاری tarballیا tar، gzipو bzip در linux استفاده می شود. tar .به طور گسترده ای مورد استفاده قرار می گیرد برای ایجاد فایل های آرشیو فشرده و می تواند به راحتی از یک دیسک به دیسک دیگر یا ماشین به ماشین منتقل می شود. در این کتاب ما قصد بررسی و بحث در مورد نمونه های مختلف دستور tar را از جمله **نحوه** ایجاد فایل های بایگانی با استفاده ازtar ، tar.gzو tar.bz2فشرده سازی، چگونگی استخراج فایل بایگانی، استخراج یک فایل، مشاهده محتوای فایل ، فایل را تأیید کنید .

ایجاد آرشیو فایلtar

دستور زیر مثال زیر یک فایل آرشیو tarرا برای hosein -14-09-12.tar برای یک فهرست /home / hosein در فهرست کاری فعلی ایجاد می کند.

ساختار دستور:

```
tar [options] [archive] [files or directories]
```

اجزای دستور:

- options گزینه‌هایی برای تعیین نحوه عملکرد دستور
- archive نام فایل آرشیو
- files or directories لیست فایل‌ها یا دایرکتوری‌هایی که می‌خواهید آرشیو یا استخراج شوند

گزینه‌های دستور:

- c ایجاد یک آرشیو جدید
- x استخراج محتویات آرشیو
- t نمایش لیست محتویات آرشیو
- v نمایش جزئیات عملیات
- f تعیین نام فایل آرشیو
- z فشرده‌سازی آرشیو با gzip
- j فشرده‌سازی آرشیو با bzip2
- l نمایش پیوندهای نمادین به جای دنبال کردن آنها
- p حفظ مجوزها و مالکیت فایل‌ها
- m حفظ زمان‌های اصلاح فایل‌ها

مثال های دستور :

ایجاد فایل آرشیو tar.gz

برای ایجاد فایل آرشیو فشرده gzipما از این گزینه به عنوان z استفاده می کنیم.به عنوان مثال دستور زیر یک فایل MyImages-14-09-12.tar.gzفشرده برای فهرست / MyImages / home ایجاد می کند .

```
# tar cvzf  MyImages-14-09-12.tar.gz / home / MyImages
```

یا

```
# tar cvzf  MyImages-14-09-12.tgz / home / MyImages
```

```
/ home / MyImages /
/home/MyImages/Sara-Khan-and-model-Priyanka-Shah.jpg
/home/MyImages/RobertKristenviolent101201.jpg
/home/MyImages/Justintimerlake101125.jpg
/home/MyImages/Mileyphoto101203.jpg
/home/MyImages/JenniferRobert101130.jpg
/home/MyImages/katrinabarbiedoll231110.jpg
/home/MyImages/the-japanese-wife-press-conference.jpg
/home/MyImages/ReesewitherspoonCIA101202.jpg
/home/MyImages/yanaguptabaresf231110.jpg
```

ویژگی bz2فشرده شده و فایل بایگانی کمتر از اندازه gzipرا ایجاد می کند.فشرده سازی bz2زمان بیشتری را برای فشرده سازی و فشرده سازی فایلها در مقایسه با gzipمی گیرد که زمان کمتری را می گیرد.برای ایجاد یک فایل tar کاملا فشرده

ما از گزینه jاستفاده می کنیم.دستور زیر مثال زیر فایل phpfiles-org.tar.bz2را برای یک فهرست / home / php ایجاد می کند.

```
# tar cvfj Phpfiles-org.tar.bz2 / home / php
```

یا

```
# tar cvfj Phpfiles-org.tar.tbz / home / php
```

یا

```
# tar cvfj Phpfiles-org.tar.tb2 / home / php
```

/ home / php /
/home/php/iframe_ew.php
/home/php/videos_all.php
/home/php/rss.php
/home/php/index.php
/home/php/vendor.php
/home/php/video_title.php
/home/php/report.php
/home/php/object.html
/home/php/video.php

ایجاد فایل آرشیو Untar وtar

برای جدا کردن یا استخراج یک فایلtar ، فقط دستور زیر را با استفاده از گزینه x[1] صادر کنید.به عنوان مثال دستور زیر فایل public_html-14-09-12.tarرا در فهرست فعلی باز خواهد کرد.

```
# tar -xvf thumbnails-14-09-12.tar.gz
```

/ home / public_html / videos / thumbnails /
/home/public_html/videos/thumbnails/katdeepika231110.jpg
/home/public_html/videos/thumbnails/katrinabarbiedoll231110.jpg
home/public_html/videos/thumbnails/onceuponatime101125.jpg
/home/public_html/videos/thumbnails/playbutton.png
/home/public_html/videos/thumbnails/ReesewitherspoonCIA101202.jpg
/home/public_html/videos/thumbnails/snagItNarration.jpg
/home/public_html/videos/thumbnails/Minissha-Lamba.jpg
/home/public_html/videos/thumbnails/Lindsaydance101201.jpg
/home/public_html/videos/thumbnails/Mileyphoto101203.jpg

برای غیرفعال کردن فایل tar.bz2بسیار فشرده، فقط از دستور زیر استفاده کنید.دستور زیر مثال زیر تمام فایل های .flvرا از فایل آرشیو جدا می کند.

```
# tar -xvf videos-14-09-12.tar.bz2
```

[1] extract

/home/public_html/videos/flv/katrinabarbiedoll231110.flv
/home/public_html/videos/flv/BrookmuellerCIA101125.flv
/home/public_html/videos/flv/dollybackinbb4101125.flv
/home/public_html/videos/flv/JenniferRobert101130.flv
/home/public_html/videos/flv/JustinAwardmovie101125.flv
/home/public_html/videos/flv/Lakme-Fashion-Week.flv
/home/public_html/videos/flv/Mileyphoto101203.flv
/home/public_html/videos/flv/Minissha-Lamba.flv

فهرست محتوای فایل tar

برای فهرست محتویات فایل آرشیوtar ، فقط دستور زیر را با گزینه t لیست محتوا اجرا کنید.دستور زیر محتوی فایل uploadprogress.tarرا فهرست می کند.

```
# tar -tvf uploadprogress.tar
```

-rw-r - r -- chregu / staff 2276 2011-08-15 18:51:10 package2.xml
-rw-r - r -- chregu / ۱۸:۵۱:۱۰ ۱۵-۰۸-۲۰۱۱ ۷۸۷۷ uploadprogress / examples / index.php
-rw-r - r -- chregu / staff 1685 2011-08-15 18:51:10 uploadprogress / examples / server.php
-rw-r - r -- chregu / staff 1697 2011-08-15 18:51:10 uploadprogress / examples / info.php
-rw-r - r -- chregu / ۱۸:۵۱:۱۰ ۱۵-۰۸-۲۰۱۱ ۳۶۷ uploadprogress / config.m4
-rw-r - r -- chregu / ۱۸:۵۱:۱۰ ۱۵-۰۸-۲۰۱۱ ۳۰۳ uploadprogress / config.w32
-rw-r - r -- chregu / ۱۸:۵۱:۱۰ ۱۵-۰۸-۲۰۱۱ ۳۵۶۳ uploadprogress / php_uploadprogress.h
-rw-r - r -- chregu / staff 15433 2011-08-15 18:51:10 uploadprogress / uploadprogress.c
-rw-r - r -- chregu / staff 1433 2011-08-15 18:51:10 package.xml

فهرست محتوای فایل tar.gz

دستورالعمل زیر را برای لیست محتویات فایل tar.gzاستفاده کنید.

```
# tar -tvf staging. hosein .com.tar.gz
```

-rw-r - r -- root / root 0 2012-08-30 04:03:57 staging. hosein .com-access_log
-rw-r - r -- root / root 587 2012-08-29 18:35:12 staging. hosein .com-access_log.1
-rw-r - r -- root / root 156 2012-01-21 07:17:56 staging. hosein .com-access_log.2
-rw-r - r -- root / root 156 2011-12-21 30:56 staging. hosein .com-access_log.3
-rw-r - r -- root / root 156 2011-11-20 17:28:24 staging. hosein .com-access_log.4
-rw-r - r -- root / root 0 2012-08-30 04:03:57 staging. hosein .com-error_log
-rw-r - r -- root / root 3981 2012-08-29 18:35:12 staging. hosein .com-error_log.1
-rw-r - r -- root / root 211 2012-01-21 07:17:56 staging. hosein .com-error_log.2
-rw-r - r -- root / root 211 2011-12-21 30:56 staging. hosein .com-error_log.3
-rw-r - r -- root / root 211 2011-11-20 17:28:24 staging. hosein .com-error_log.4

فهرست محتوای فایل tar.bz2

برای فهرست محتویات فایل tar.bz2، دستور زیر را صادر کنید.

```
# tar -tvf   Phpfiles-org.tar.bz2
```

drwxr-xr-x root / root 0 2012-09-15 03:06:08 / home / php /
-rw-r - r -- root / root 1751 2012-09-15 03:06:08 /home/php/iframe_ew.php
-rw-r - r -- root / root 11220 2012-09-15 03:06:08 /home/php/videos_all.php
-rw-r - r -- root / root 2152 2012-09-15 03:06:08 /home/php/rss.php
-rw-r - r -- root / root 3021 2012-09-15 03:06:08 /home/php/index.php
-rw-r - r -- root / root 2554 2012-09-15 03:06:08 /home/php/vendor.php
-rw-r - r -- root / root 406 2012-09-15 03:06:08 /home/php/video_title.php

برای استخراج یک فایل به نام cleanfiles.sh ازیک فایل cleanfiles.sh.tar از دستور زیر استفاده کنید.

```
# tar -xvf cleanfiles.sh.tar cleanfiles.sh
```

یا

```
# tar --extract --file = cleanfiles.sh.tar cleanfiles.sh
```

برای استخراج یک فایل hosein backup.xml از فایل آرشیو hosein backup.tar.gz، از دستور زیر استفاده کنید.

```
# tar -zxvf hosein backup.tar.gz hosein backup.xml
```

یا

```
# tar --extract --file = hosein backup.tar.gz hosein backup.xml
```

برای استخراج یک فایل به نام index.php از فایل Phpfiles-org.tar.bz2 از گزینه زیر استفاده کنید.

```
# tar -jxvf Phpfiles-org.tar.bz2 home / php / index.php
```

یا

```
# tar --extract --file = Phpfiles-org.tar.bz2 /home/php/index.php
```

/home/php/index.php

برای استخراج یا جدا کردن چندین فایل از فایل آرشیو tar، tar.gz و tar.bz2 به عنوان مثال :

دستور زیر "file1 " " file 2 "را از فایل های بایگانی استخراج[1] می کند.

```
# tar -xvf hosein -14-09-12.tar " file 1 " " file 2 "
```

یا

```
# tar -zxvf MyImages-14-09-12.tar.gz " file 1" " file2 "
```

یا

```
# tar -jxvf Phpfiles-org.tar.bz2 " file1 " " file2"
```

برای استخراج گروهی از فایلهای ما از استخراج بر اساس کلمات استفاده می‌کنیم.برای مثال ، برای استخراج گروهی از تمام فایل هایی که الگوی آنها با phpآغاز می شود از یک فایل آرشیو tar، tar.gz و tar.bz2 .

```
# tar -xvf Phpfiles-org.tar - wildcards '* .php'
```

```
# tar -zxvf Phpfiles-org.tar.gz - wildcards '* .php'
```

```
# tar -jxvf Phpfiles-org.tar.bz2 - wildcards '* .php'
```

/home/php/iframe_ew.php
/home/php/videos_all.php
/home/php/rss.php
/home/php/index.php
/home/php/vendor.php
/home/php/video_title.php
/home/php/report.php

[1] extract

برای اضافه کردن فایل ها یا فهرست ها به فایل های آرشیو شده موجود، از گزینه (append) rاستفاده می کنیم.برای مثال : فایل xyz.txtو فهرست phpرا به فایل آرشیو hosein -14-09-12.tarاضافه میکنیم.

```
# tar -rvf  hosein - 14-09-12.tar xyz.txt
```

```
# tar -rvf  hosein -14-09-12.tar php
```

بررسی اندازه فایل های آرشیو

برای بررسی حجم فایل tar، tar.gzو tar.bz2از دستور زیر استفاده کنید.به عنوان مثال دستور زیر حجم فایل بایگانی را در کیلوبایت (KB) نمایش می دهد.

```
# tar -czf -  hosein -14-09-12.tar |WC-C
```

12820480

```
# tar -czf - MyImages-14-09-12.tar.gz |WC-C
```

112640

```
# tar -czf - Phpfiles-org.tar.bz2 |WC-C
```

20480

۷۸.دستور python

دستور python در سیستم‌هایی که Python نصب شده است برای اجرای اسکریپت‌ها و کدهای نوشته شده به زبان Python استفاده می‌شود. این دستور به شما این امکان را می‌دهد که اسکریپت‌های Python خود را اجرا کرده و پارامترهای مورد نیاز آن‌ها را تنظیم کنید.

ساختار دستور

```
python [options] [script.py] [arguments]
```

گزینه‌های دستور

- V- نمایش نسخه پایتون
- h- نمایش راهنمای دستور
- i- ورود به حالت تعاملی مفسر
- c- اجرای کد پایتون به صورت خط فرمان
- m- اجرای یک ماژول پایتون
- O- بهینه‌سازی کد پایتون برای افزایش سرعت
- W- فعال کردن هشدارهای مربوط به خطاهای احتمالی
- q- غیرفعال کردن چاپ پیغام‌های خوش‌آمد و خداحافظی
- B- فعال کردن حالت batch
- u- فعال کردن Unicode
- E- فعال کردن shebang line

مثال های دستور

اجرای یک اسکریپت Python:این دستور برای اجرای یک فایل Python با پسوند .py استفاده می‌شود.

```
python myscript.py
```

اجرای کد Python در حالت تعاملی (Interactive Mode): این دستور برای ورود به محیط تعاملی Python به صورت تعاملی و اجرای دستورات Python به صورت تک تک استفاده می‌شود.

```
python
```

نصب پکیج‌های Python با استفاده از pip: این دستور برای نصب یک پکیج Python از Python Package Index (PyPI) با استفاده از pip استفاده می‌شود.

```
python -m pip install requests
```

اجرای کد Python به صورت Optimized: این دستور برای اجرای یک اسکریپت Python به صورت Optimized (با بهینه‌سازی) استفاده می‌شود.

```
python -O myscript.py
```

نمایش راهنمای استفاده از دستور Python: این دستور برای نمایش راهنمای استفاده از دستور python و گزینه‌های موجود آن استفاده می‌شود.

```
python --help
```

با استفاده از دستور python می‌توانید اسکریپت‌ها و کدهای Python خود را اجرا کرده و پیکربندی‌های مختلفی را برای اجرای آن‌ها تنظیم کنید.

مثال بیشتر:

نمایش نسخه پایتون:

```
python -V
```

ورود به حالت تعاملی مفسر:

```
python -i
```

اجرای کد پایتون به صورت خط فرمان:

```
python -c “print(‘Hello, world!’)”
```

اجرای یک ماژول پایتون:

```
python -m math
```

بهینه‌سازی کد پایتون برای افزایش سرعت:

```
python -O script.py
```

فعال کردن هشدارهای مربوط به خطاهای احتمالی:

```
python -W all script.py
```

غیرفعال کردن چاپ پیغام‌های خوش‌آمد و خداحافظی:

```
python -q script.py
```

فعال کردن حالت batch:

```
python -B script.py
```

فعال کردن Unicode:

```
python -u script.py
```

فعال کردن shebang line:

```
python -E script.py
```

۷۹.دستور pip

pipابزاری قدرتمند برای مدیریت بسته‌های پایتون در سیستم‌های لینوکس است. این ابزار به شما امکان می‌دهد تا کتابخانه‌ها و چارچوب‌های پایتون را نصب، حذف، به‌روزرسانی و مدیریت کنید.

ساختار دستور:

```
pip [options] <command> [arguments]
```

گزینه های دستور:

- -h, --help

راهنمای برنامه را نشان میدهد.

- --debug

اجازه میدهد استثناء های بدون دستکاری خارج از زیر برنامه اصلی منتشر شوند، به جای ثبت آنها در stderr.

- --isolated

pip را در یک حالت جدا اجرا میکند، متغیرهای محیطی و پیکربندی کاربر را نادیده بگیرید.

- --require-virtualenv

اجازه میدهد pip فقط در یک محیط مجازی اجرا شود؛ در غیر این صورت با خطا خارج شوید.

- --python <python>

pip را با تفسیرگر Python مشخص شده اجرا میکند.

- -v, --verbose

بیشتر خروجی میدهد. گزینه افزودنی است و می تواند تا ۳ بار استفاده شود.

- -V, --version

نسخه را نشان میدهد و خارج میشود.

- -q, --quiet

کمتر خروجی نشان میدهد.

- --log <path>

توضیحات بیشتر.

- --no-input

غیرفعال کردن درخواست برای ورودی.

- --keyring-provider <keyring_provider>

اگر ورودی کاربر مجاز است، جستجوی اعتبارنامه را از طریق کتابخانه keyring فعال میکند.

- --proxy <proxy>

یک پروکسی را با فرم scheme://[user:passwd@]proxy.server:port مشخص میکند.

- --retries <retries>

حداکثر تعداد تلاش هایی که هر اتصال باید انجام دهد (پیش فرض ۵ بار).

- --timeout <sec>

تنظیم زمان انقضای سوکت (پیش فرض ۱۵ ثانیه).

- --exists-action <action>

عمل پیش فرض وقتی یک مسیر از قبل وجود دارد: (s)witch, (i)gnore, (w)ipe, (b)ackup, (a)bort.

- --trusted-host <hostname>

این میزبان یا جفت میزبان:port را به عنوان مورد اعتماد علامت گذاری میکند، حتی اگر HTTPS معتبر یا هیچ HTTPS نداشته باشد.

- --cert <path>

مسیر به بسته گواهینامه CA با کدگذاری PEM. اگر ارائه شود، پیش فرض را نادیده می گیرد. برای اطلاعات بیشتر ‘ SSL Certificate Verification’ را در مستندات pip ببینید.

- --client-cert <path>

مسیر به گواهینامه مشتری SSL، یک فایل تکی شامل کلید خصوصی و گواهینامه در فرمت PEM.

- --cache-dir <dir>

داده های کش را در dir ذخیره میکند.

- --no-cache-dir

کش را غیرفعال میکند.

- --disable-pip-version-check

به صورت دوره ای PyPI را برای تعیین اینکه آیا نسخه جدیدی از pip برای دانلود موجود است، چک نمیکند.

- --no-color

خروجی رنگی را غیرفعال میکند.

- --no-python-version-warning

عدم هشدار برای از رده خارج شدن نسخه های Python های در بروزرسانی های بعدی.

- --use-feature <feature>

قابلیت جدید را فعال میکند، که ممکن است با نسخه های قبلی ناسازگار باشد.

- --use-deprecated <feature>

قابلیت منسوخ شده را فعال میکنید،

مثال های دستور:

فرض کنید می خواهید راهنمای کامل دستور pip install را مشاهده کنید. برای این کار:

```
pip install --help
```

این دستور خروجی مفصل تمام گزینه‌ها و آرگومان‌های قابل استفاده با pip install را نمایش می‌دهد.

debug--

این گزینه معمولا برای توسعه دهندگان pip و اشکال زدایی آن کاربرد دارد. در استفاده روزمره نیازی به این گزینه نخواهید داشت.

isolated--

فرض کنید در حال کار بر روی یک پروژه خاص هستید و نمی‌خواهید تنظیمات و محیط کلی سیستم شما بر نصب بسته‌ها تاثیر بگذارد. با استفاده از این گزینه، pip به صورت ایزوله اجرا شده و تنظیمات محیطی را نادیده می‌گیرد.

```
pip install --isolated some-package
```

require-virtualenv--

این گزینه pip را مجبور می‌کند تا تنها در یک محیط مجازی (virtual environment) اجرا شود. در صورتی که خارج از محیط مجازی باشید، با خطا مواجه خواهید شد. این گزینه برای اطمینان از اینکه بسته‌ها در یک محیط ایزوله نصب می‌شوند، مفید است.

```
pip install --require-virtualenv some-package
```

python <python interpreter path>--

فرض کنید چندین نسخه از پایتون در سیستم خود دارید و می‌خواهید pip از یک نسخه خاص استفاده کند. با این گزینه می‌توانید مسیر آن نسخه را مشخص کنید.

```
pip install --python /usr/bin/python3.9 some-package
```

v- (یا verbose)--

این گزینه میزان جزئیات خروجی pip را افزایش می‌دهد. با استفاده چندباره از این گزینه (تا سقف ۳ بار)، خروجی با جزئیات بیشتری همراه خواهد بود.

```
pip install -v some-package  # خروجی با جزئیات کم
pip install -vv some-package # خروجی با جزئیات متوسط
pip install -vvv some-package # خروجی با جزئیات زیاد
```

keyring-provider

این گزینه به شما امکان می‌دهد تا از طریق کتابخانه keyring، اطلاعات احراز هویت را بازیابی کنید. این موضوع در شرایطی که نیاز به رمز عبور برای دسترسی به یک منبع دارید، مفید خواهد بود.

```
pip install --keyring-provider=subprocess some-package
```

proxy

با استفاده از این گزینه می‌توانید یک پروکسی برای اتصال به اینترنت در pip مشخص کنید.

```
pip install --proxy=http://user:pass@proxy.example.com:8080 some-package
```

retries

این گزینه حداکثر تعداد تلاش‌های pip برای اتصال به یک منبع را تعیین می‌کند.

```
pip install --retries=10 some-package
```

timeout

با این گزینه می‌توانید زمان انتظار pip برای اتصال به یک منبع را مشخص کنید.

```
pip install --timeout=30 some-package
```

exists-action

این گزینه نحوه برخورد pip با وجود یک مسیر از قبل موجود را تعیین می‌کند.

```
pip install --exists-action=i some-package  # Ignore existing path
pip install --exists-action=w some-package  # Wipe existing path
```

trusted-host

این گزینه به شما امکان می‌دهد تا یک میزبان (یا میزبانپورت) را به عنوان قابل اعتماد علامت گذاری کنید، حتی اگر گواهینامه HTTPS معتبر نداشته باشد.

```
pip install --trusted-host=example.com some-package
```

cert

با این گزینه می‌توانید مسیر یک گواهی CA (Certificate Authority) را برای بررسی گواهینامه HTTPS مشخص کنید.

```
pip install --cert=/path/to/ca.pem some-package
```

client-cert

این گزینه مسیر یک گواهی SSL (Secure Sockets Layer) را برای احراز هویت کلاینت در pip مشخص می‌کند.

```
pip install --client-cert=/path/to/client.pem some-package
```

cache-dir

با این گزینه می‌توانید مسیر ذخیره‌سازی حافظه پنهان pip را تغییر دهید.

```
pip install --cache-dir=/path/to/cache some-package
```

مدیریت کاربران و گروه ها

مشاهده کاربران در سیستم

۸۰. دستور users

دستور users در لینوکس برای نمایش لیست کاربران متصل به سیستم استفاده می‌شود. این دستور اطلاعات مختصری از جمله نام کاربری، نام میزبان، زمان ورود و وضعیت هر کاربر را نمایش می‌دهد.

ساختار دستور:

```
users [options]
```

گزینه‌های دستور:

- a– یا --all: نمایش تمام کاربران، حتی کاربرانی که در حال حاضر غیرفعال هستند.
- h– یا --help: نمایش راهنمای دستور
- l– یا --long: نمایش اطلاعات بیشتر از هر کاربر، مانند شناسه کاربر(UID) ، شناسه گروه (GID) و زمان بیکاری
- i– یا --idle: نمایش لیست کاربران بیکار
- n– یا --numeric: نمایش شناسه کاربری (UID) و شناسه گروه (GID) به جای نام کاربری
- p– یا --processes: نمایش تعداد پردازش‌هایی که هر کاربر اجرا می‌کند
- s– یا --sort: مرتب‌سازی لیست کاربران بر اساس معیارهای مختلف، مانند نام کاربری، زمان ورود یا زمان بیکاری
- t– یا --tty: نمایش ترمینالی که هر کاربر از آن استفاده می‌کند

مثال‌های دستور:

نمایش لیست مختصری از کاربران:

```
users
```

نمایش اطلاعات کامل از تمام کاربران:

```
users -a -l
```

نمایش لیست کاربران بیکار:

```
users -i
```

۸۱.دستور whomai

دستور whoami در لینوکس برای نمایش اطلاعات مربوط به کاربر فعلی سیستم استفاده می‌شود. این اطلاعات شامل نام کاربری، نام میزبان و شناسه کاربری ((UIDاست.

ساختار دستور:

```
whoami [options]
```

گزینه‌های دستور:

- h– یا --help: نمایش راهنمای دستور
- u– یا --uid: نمایش شناسه کاربری ((UID
- l– یا --long: نمایش اطلاعات بیشتر شامل نام کامل و گروه کاربری

مثال های دستور:

نمایش نام کاربری:

```
whoami
```

نمایش شناسه کاربری:

```
whoami -u
```

نمایش اطلاعات کامل:

```
whoami -l
```

۸۲.دستور who

دستور **$who** برای یافتن اطلاعات زیر استفاده می شود:

۱. زمان آخرین بوت شدن سیستم

۲. سطح اجرای فعلی سیستم

۳. لیست کاربران وارد شده در سیستم و موارد دیگر.

ساختار دستور:

```
$who [options] [نام پرونده]
```

مثال های دستور:

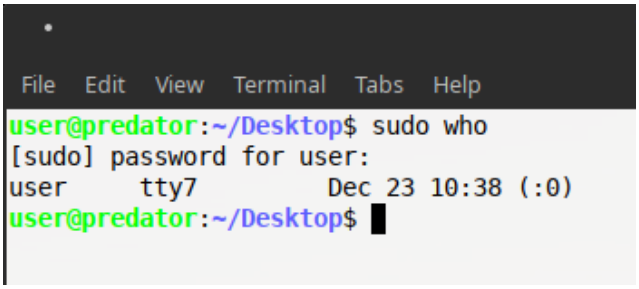
1.دستور $who در صورت عدم ارائه گزینه ، اطلاعات زیر را برای هر کاربر که در حال حاضر وارد سیستم شده است نمایش می دهد:

۱. نام ورود کاربران

۲. شماره خط ترمینال

۳. زمان ورود کاربران به سیستم

۴. نام میزبان از راه دور کاربر



۸۳.دستور w

دستورw در سیستم عامل‌های Unix و مشتقات آن مانند Linux برای نمایش اطلاعات جامع در مورد کاربرانی که وارد سیستم شده‌اند، وضعیت سیستم و زمان سیستم استفاده می‌شود. این دستور اطلاعاتی از جمله نام کاربری، ترمینال، زمان ورود، زمان فعلی، مدت زمانی که سیستم روشن بوده و بار سیستم را نمایش می‌دهد.

ساختار دستور:

```
w [options]
```

گزینه‌های دستور:

- h- یا help-- :نمایش راهنمای دستور
- u- یا users-- :فقط لیست کاربران را نمایش می‌دهد
- s- یا short-- :فقط اطلاعات خلاصه را نمایش می‌دهد
- l- یا long-- :اطلاعات کامل را نمایش می‌دهد
- o- یا output-- :فرمت خروجی را مشخص می‌کند
- f- یا full-path-- :مسیر کامل فایل‌ها را نمایش می‌دهد
- t- یا tty-- :نام ترمینال کاربر را نمایش می‌دهد
- i- یا idle-- :زمان بیکاری کاربر را نمایش می‌دهد
- p- یا pid-- :شناسه فرآیند (PID) کاربر را نمایش می‌دهد
- c- یا cpu-- :درصد استفاده از CPU توسط کاربر را نمایش می‌دهد
- m- یا memory-- :درصد استفاده از حافظه توسط کاربر را نمایش می‌دهد
- q- یا quiet-- :فقط تعداد کاربران را نمایش می‌دهد

مثال های دستور:

نمایش لیست کاربران:

```
w
```

نمایش اطلاعات خلاصه:

```
w -s
```

نمایش اطلاعات کامل:

```
w -l
```

۸۴.دستور last

آخرین دستور لیستی از آخرین کاربران وارد شده را نمایش می دهد.

```
$ last
```

دستور **last** در لینوکس استفاده می شود برای نمایش لیستی از تمام کاربران در داخل و خارج به سیستم وارد از آنجا که فایل **/ var / log / wtmp** /ساخته شده است. می توان یک یا چند نام کاربری را به عنوان آرگومان برای نمایش زمان ورود (و یا خارج شدن) و نام میزبان آنها ارائه داد.

۸۵.دستور id

دستورid نشان می دهد که اطلاعات کاربر و گروه برای کاربر فعلی یا نام کاربری مشخص شده است که در زیر نشان داده شده است.

```
$ id hosein
```

از دستور id در Linux برای یافتن نام کاربر و گروه و id عددی (UID) یا id گروه) کاربر فعلی یا هر کاربر دیگر در سرور استفاده می شود. این دستور برای یافتن اطلاعات زیر به شرح زیر مفید است:

- نام کاربری و id کاربری واقعی
- UIDخاص کاربران.
- UIDو همه گروه های مرتبط با کاربر

- لیست تمام گروه های عضو کاربر.
- زمینه امنیتی کاربر فعلی

ساختار دستور:

```
id [OPTION]... [کاربر]
```

گزینه های دستور:

- **g** فقط id موثر گروه را چاپ کنید.
- **G** همه id های گروه را چاپ کنید.
- **n** نام را به جای شماره چاپ می کند.
- **r** id واقعی را به جای اعداد چاپ می کند.
- **u** فقط id کاربری موثر را چاپ می کند.

help– نمایش پیام های راهنما و خروج از آن.

version– اطلاعات نسخه را نمایش داده و از آن خارج شوید.

توجه: بدون هیچ OPTION ، آن مجموعه ای از اطلاعات شناسایی شده یعنی id های عددی را چاپ می کند.

مثال های دستور:

برای چاپ id شخصی خود بدون هیچ گزینه:

```
id
```

```
root@master-VirtualBox:~# id
uid=0(root) gid=0(root) groups=0(root)
root@master-VirtualBox:~#
```

۸۶.دستور useradd

دستور useradd در سیستم عامل‌های Unix و مشتقات آن مانند Linux برای ایجاد یک کاربر جدید در سیستم استفاده می‌شود. با استفاده از این دستور، می‌توانید یک حساب کاربری جدید ایجاد کرده و تنظیمات مربوط به آن کاربر را مشخص کنید.

گزینه های دستور:

- c COMMENT توضیحات مربوط به کاربر را مشخص می‌کند.
- d HOME_DIR مسیر پوشه خانگی کاربر را مشخص می‌کند.
- e EXPIRATION_DATE تاریخ انقضای حساب کاربری را مشخص می‌کند.
- f INACTIVE_TIME تعداد روزهایی که حساب کاربری قبل از غیرفعال شدن می‌تواند بدون استفاده بماند را مشخص می‌کند.
- g GROUP_NAME گروه پیش‌فرض کاربر را مشخص می‌کند.
- G GROUP_NAMES لیستی از گروه‌های ثانویه که کاربر عضو آنها خواهد بود را مشخص می‌کند.
- m به جای ایجاد پوشه خانگی، فقط اطلاعات مربوط به کاربر را در فایل‌های سیستم ایجاد می‌کند.
- M از ایجاد پوشه خانگی و کپی کردن فایل‌های پیش‌فرض به آن جلوگیری می‌کند.
- n LOGIN_NAME نام ورود به سیستم کاربر را مشخص می‌کند.
- r یک حساب کاربری سیستمی ایجاد می‌کند.

- s SHELL پوسته پیش‌فرض کاربر را مشخص می‌کند.
- u UID شناسه کاربری ((UIDرا مشخص می‌کند.

مثال های دستور:

ایجاد یک حساب کاربری ساده:

```
useradd ali
```

ایجاد یک حساب کاربری با توضیحات:

```
useradd -c “ ”کاربر جدیدali
```

ایجاد یک حساب کاربری با پوشه خانگی در مسیر دلخواه:

```
useradd -d /home/custom/ali ali
```

ایجاد یک حساب کاربری با تاریخ انقضا:

```
useradd -e “2024-12-31” ali
```

۸۷.دستور passwd

دستور **passwd** در سیستم عامل‌های **Unix** و مشتقات آن مانند **Linux** برای تغییر رمز عبور یک کاربر استفاده می‌شود. با استفاده از این دستور، کاربر می‌تواند رمز عبور خود را تغییر دهد.

گزینه های دستور:

- a رمز عبور را قبل از تغییر آن با استفاده از الگوریتم‌های رمزگذاری مختلف بررسی می‌کند.
- d حساب کاربری را غیرفعال می‌کند.
- e EXPIRATION_DATE تاریخ انقضای رمز عبور را مشخص می‌کند.
- f INACTIVE_TIME تعداد روزهایی که حساب کاربری قبل از غیرفعال شدن می‌تواند بدون استفاده بماند را مشخص می‌کند.
- l حساب کاربری را قفل می‌کند.
- n حداقل طول رمز عبور را مشخص می‌کند.
- r حساب کاربری را فعال می‌کند.
- s وضعیت رمز عبور را نمایش می‌دهد.
- u رمز عبور را به طور تصادفی تولید می‌کند.
- x حداکثر تعداد روزهایی که می‌توان از رمز عبور استفاده کرد را مشخص می‌کند.

مثال های دستور:

تغییر رمز عبور خود:

```
passwd
```

تغییر رمز عبور کاربر دیگر:

```
sudo passwd ali
```

غیرفعال کردن حساب کاربری:

```
sudo passwd -d ali
```

فعال کردن حساب کاربری:

```
sudo passwd -r ali
```

قفل کردن حساب کاربری:

```
sudo passwd -l ali
```

نمایش وضعیت رمز عبور:

```
passwd -s ali
```

تعیین تاریخ انقضای رمز عبور:

```
sudo passwd -e “2024-12-31” ali
```

تعیین حداقل طول رمز عبور:

```
sudo passwd -n 8 ali
```

تولید رمز عبور تصادفی:

```
sudo passwd -u ali
```

۸۸.دستور usermod

پس از ایجاد حساب های کاربری، در برخی از شرایط که در آن ما نیاز به تغییر ویژگی های یک کاربر موجود مانند تغییر فهرست خانگی کاربر، نام کاربری، پوسته ورود، تاریخ انقضا رمز عبور، و غیره، در جایی که در این صورت دستور ’usermod‘استفاده می شود. هنگامی که ما دستور ’usermod‘ را در ترمینال اجرا می کنیم، فایل های زیر مورد استفاده و تحت تاثیر قرار می گیرند.

۱. / - etc / passwd اطلاعات حساب کاربری
۲. / - etc / shadow اطلاعات حساب امن
۳. / - etc / group اطلاعات حساب گروهی.
۴. / - etc / gshadow اطلاعات حساب امن گروه.
۵. / - etc/login.defsپیکربندی مجموعه ای از رمز عبور سایه

ساختار دستور:

```
usermod [گزینه ها] نام کاربری
```

گزینه های دستور:

- c = – cمیتوانیم فیلد comment را برای useraccount اضافه کنیم.
- d = – dتغییر فهرست برای هر حساب کاربر موجود.
- e = – eبا استفاده از این گزینه می توانیم مدت زمان مشخص حساب کاربری را لغو کنیم.
- g = – gگروه اصلی را برای یک کاربر تغییر دهید.
- G = – Gاضافه کردن یک گروه مکمل
- a = – aبرای اضافه کردن هر گروه از گروه به یک گروه ثانویه.
- l = – lبرای تغییر نام ورود از hosein به hosein _admin .
- L = – Lبرای قفل کردن حساب کاربری ، بنابراین نمی توانیم از حساب استفاده کنیم.
- m = – mانتقال محتویات فهرست خانگی از home موجود به فهرست جدید.

- p = – برای استفاده از رمز عبور رمزگذاری نشده برای رمز عبور جدید
- s = – یک پوسته مشخص برای حساب های جدید ایجاد کنید.
- u = – به UID اختصاص داده شده برای حساب کاربری بین ۰ تا ۹۹۹.
- U = – برای باز کردن حساب های کاربری.این قفل رمز عبور را حذف می کند و به ما اجازه می دهد از حساب کاربری استفاده کنیم.

مثال های دستور:

تغییر توضیحات مربوط به کاربر:

```
usermod -c “ ”کاربر با سابقهali
```

تغییر مسیر پوشه خانگی کاربر:

```
usermod -d /home/custom/ali ali
```

تغییر تاریخ انقضای حساب کاربری:

```
usermod -e “2024-12-31” ali
```

تغییر گروه پیش‌فرض کاربر:

```
usermod -g developers ali
```

تغییر عضویت کاربر در گروه‌ها:

```
usermod -G developers,admins ali
```

تغییر نام ورود به سیستم کاربر:

```
usermod -l alireza ali
```

قفل کردن حساب کاربر:

```
usermod -L ali
```

باز کردن حساب کاربر:

```
usermod -U ali
```

روز چهلم

ماژول‌های کرنل

ماژول‌های کرنل قطعه‌هایی از کد هستند که می‌توانند در صورت تقاضا در هسته بارگیری و تخلیه شوند. آنها عملکرد هسته را بدون نیاز به راه اندازی مجدد سیستم گسترش می دهند. برای ایجاد یک ماژول هسته، می توانید راهنمای برنامه نویسی ماژول کرنل لینوکس را بخوانید. یک ماژول را می توان به صورت داخلی یا قابل بارگذاری پیکربندی کرد. برای بارگذاری یا حذف پویا یک ماژول، باید به عنوان یک ماژول قابل بارگذاری در پیکربندی هسته پیکربندی شود (بنابراین خط مربوط به ماژول حرف M را نشان می دهد). ماژول های لینوکس توده هایی از کد هستند که می توانند در هر نقطه پس از بوت شدن سیستم به صورت پویا به هسته پیوند داده شوند. آنها را می توان از هسته جدا کرد و زمانی که دیگر مورد نیاز نیست حذف شد.

۸۹.دستور lsmod

از دستور lsmod برای نمایش وضعیت ماژول ها در هسته لینوکس استفاده می شود. این منجر به لیستی از ماژول های بارگذاری شده می شود. lsmod یک برنامه پیش پا افتاده است که به خوبی محتویات قالب بندی شده را/proc/modules نشان می دهد که ماژول های هسته در حال حاضر بارگیری می شوند.

ساختار دستور:

```
lsmod
```

گزینه های دستور:

- a– نمایش تمام ماژول‌ها، حتی ماژول‌های هسته داخلی.
- b– نمایش نام کامل درایور برای هر ماژول.
- c– نمایش تعداد دفعات استفاده از هر ماژول.
- d– نمایش وابستگی‌های هر ماژول.
- e– نمایش اطلاعات مربوط به ماژول‌های هسته خارجی.
- f– نمایش اطلاعات مربوط به ماژول‌های هسته بارگذاری شده در یک فایل.
- m– فقط ماژول‌هایی را که با نام یا شناسه مطابقت دارند نمایش می‌دهد.
- n– نمایش نام نمادین برای هر ماژول.
- p– نمایش شناسه فرآیندی که هر ماژول توسط آن استفاده می‌شود.
- q– فقط نام ماژول‌ها را نمایش می‌دهد.
- s– نمایش اندازه هر ماژول.
- t– نمایش نوع هر ماژول.
- v– نمایش اطلاعات نسخه برای هر ماژول.
- x– نمایش اطلاعات مربوط به ماژول‌های هسته بارگذاری شده در یک سیستم دیگر.

به عنوان مثال : اجرای **lsmod** در خط دستور به لیست تمام ماژول های کرنل فعال است.

```
lsmod
```

خروجی دستور:

```
narru@narru-HP-Pavilion-g4-Notebook-PC:~$ lsmod
Module                  Size  Used by
ccm                    20480  6
rfcomm                 77824  0
bnep                   20480  2
kvm_amd                86016  0
kvm                   598016  1 kvm_amd
irqbypass              16384  1 kvm
```

فرمت خروجی دستور: سه ستون در خروجی وجود دارد.

- ابتدا ستون برای " **نام** " پرونده ماژول استفاده می شود. یا می توانید بگویید
- نام = نام پرونده ماژول – پسوند پرونده ماژول (به عنوان مثال o. یا ko).
- دوم ، ستون برای " **اندازه** " ماژول در بایت استفاده می شود.
- ستون سوم در خروجی برای " **استفاده شده** "است. در اصل برای نمایش تعداد نمونه های ماژول استفاده می شود.

۹۰.دستور modinfo

از دستور **modinfo** در سیستم لینوکس برای نمایش اطلاعات مربوط به ماژول هسته لینوکس استفاده می شود. این دستور اطلاعات را از ماژول های هسته لینوکس استخراج شده در خط فرمان استخراج می کند. اگر نام ماژول نام پرونده نباشد ، به طور پیش فرض فهرست **kernel-version / modules / lib /** جستجو می شود. **modinfo** می تواند از هر یک از معماری هسته لینوکس ماژول ها را درک کند.

ساختار دستور:

modinfo [-0] [-F field] [-k kernel] [modulename|filename...]

مثال های دستور:

modinfo bluetooth

```
algoscale@algoscale-Lenovo-ideapad-330-15IKB:~$ modinfo bluetooth
filename:       /lib/modules/4.15.0-45-generic/kernel/net/bluetooth/bluetooth.ko
alias:          net-pf-31
license:        GPL
version:        2.22
description:    Bluetooth Core ver 2.22
author:         Marcel Holtmann <marcel@holtmann.org>
srcversion:     CF600F16CF3D8912A42EC69
depends:        ecdh_generic
retpoline:      Y
intree:         Y
name:           bluetooth
vermagic:       4.15.0-45-generic SMP mod_unload
parm:           disable_esco:Disable eSCO connection creation (bool)
parm:           disable_ertm:Disable enhanced retransmission mode (bool)
```

گزینه های دستور:

دستور modinfo با گزینه help: این دستورالعمل عمومی **modinfo** را به همراه گزینه های مختلف چاپ می کند و در مورد هر گزینه توضیح مختصری می دهد.

```
algoscale@algoscale-Lenovo-ideapad-330-15IKB:~$ modinfo --help
Usage:
        modinfo [options] filename [args]
Options:
        -a, --author                Print only 'author'
        -d, --description           Print only 'description'
        -l, --license               Print only 'license'
        -p, --parameters            Print only 'parm'
        -n, --filename              Print only 'filename'
        -0, --null                  Use \0 instead of \n
        -F, --field=FIELD           Print only provided FIELD
        -k, --set-version=VERSION   Use VERSION instead of `uname -r`
        -b, --basedir=DIR           Use DIR as filesystem root for /lib/modules
        -V, --version               Show version
        -h, --help                  Show this help
```

modinfo -V : این گزینه اطلاعات نسخه دستور **modinfo** را می دهد.

modinfo -V

```
algoscale@algoscale-Lenovo-ideapad-330-15IKB:~$ modinfo -V
kmod version 22
-XZ -ZLIB -EXPERIMENTAL
```

سخن پایانی

این کتاب هدیه ای بود برای همه بچهای خوب سرزمینم.

این کتاب برگرفته شده از یک پنجم کتاب ۱۰۰۱ دستور لینوکسی که یک مرجع فارسی کامل از دستورات لینوکسی است.

برای دریافت کتاب کامل دستورات لینوکسی به آدرس آکادمی زیر مراجعه نمایید.

https://learninghive.ir/

قطع کتاب A4 و حاشیه سمت راست ۲,۵ سانتی متر است و شرایط چاپ را دارد.

LEARNING HIVE

contact me:

seilany.ir
learninghive.ir
emperor-os.ir
Hubuntu.ir
predator-os.ir
telegram:@seilany

HosseinSeilani

Designer,Developer and Linux sysadmin

Founder and Developer of Emperor-OS, Hubuntu and Predator-OS. I bring significant experience as a Linux/Windows sysadmin and graphical web design,UX/UI to the Open Source Community.

LITTLE-PSYCHO
DESTRUCTIVE LINUX

Emperor-OS

Predator-OS