

به نام خداوند بخشاینده بخشایشگر

«علمی که در کار جلوه کند، بهترین نوع دانش است.» حضرت علی (ع)

آزمایشگاه سیستم عامل

پیشگفتار

ضمن آرزوی موفقیت برای شما دانشجویان گرامی، هدف از این آزمایشگاه آشنایی با سیستم عامل Unix و نگارش Linux از آن و همچنین برخی مفاهیم اساسی در سیستم عامل ویندوز می باشد. در این آزمایشگاه شما با محیط گرافیکی و متنی و مبانی این سیستم عامل آشنا شده و سپس در قسمت دوم آزمایشگاه با برنامه نویسی در محیط Linux آشنا خواهید شد.

آزمایش‌ها در گروه‌های دو نفری که جلسه اول مشخص شده و تا انتهای جلسات ادامه می‌یابد انجام می‌گیرند. هر آزمایش شامل مراحل است که باید انجام شوند و سپس نتایج در قالب یک گزارش کار دقیق و کامل ارائه گردند. گزارش هر آزمایش می‌بایست حداکثر دو هفته بعد از انجام آن آزمایش تحویل داده شود. بهتر است گزارش‌های خود را تایپ کرده و نسخه soft copy آن را تحویل دهید تا به این ترتیب در مصرف کاغذ صرفه جویی شده و به زنده نگه داشتن درخت‌ها کمک کنیم!

ارزشیابی آزمایشگاه شامل موارد زیر خواهد بود:

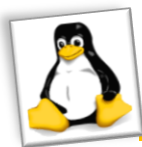
۱- کارهای عملی انجام شده

۲- گزارش‌ها

۳- امتحان پایان‌ترم

موفق باشید

پوردستی



آشنایی با سیستم عامل Linux

مقدمه

لینوکس، سیستم عامل رایگانی است که هسته اصلی آن توسط Linus Torvalds نوشته شده و اولین نسخه آن در سال ۱۹۹۱ انتشار یافت. سپس گروهی برنامه نویسی با استفاده از خصوصیات کد باز (open source) آن را توسعه دادند. لینوکس بر روی هر نوع کامپیوتر شخصی با حداقل سرعت ۳۸۶ مگاهرتز و ۴ مگابایت حافظه RAM قابل نصب است. لینوکس دارای نسخه‌های فراوانی بوده که معروف ترین آنها Red Hat، SuSe، Mandrake، Ubuntu و Fedora است.

لینوکس یک سیستم عامل قدرتمند و بسیار پایدار، دارای محیطی با امکانات چند کاربره (multi user) پیشرفته و چند وظیفه‌ای (multi-tasking)، بدون نیاز به سخت افزارهای گران قیمت و در بسیاری از موارد بدون نیاز به خرید نرم‌افزارهای پرهزینه می‌باشد.

صرف نظر از نوع لینوکس مورد استفاده، این بسته نرم‌افزاری همواره به همراه هزاران نرم‌افزار کاربردی قدرتمند عرضه می‌شود که هریک جایگزینی مطمئن و مناسب برای برنامه‌های ویندوزی می‌باشند که کاربران اغلب از آنها استفاده نموده و به آن‌ها عادت کرده‌اند.

لینوکس یک سیستم عامل کاملاً استاندارد و سازگار با استانداردهای POSIX یا شباهت بسیار به یونیکس می‌باشد و در اغلب موارد کلیه ابزارهای استاندارد یونیکس در لینوکس وجود دارند و قابل استفاده هستند. برای نرم‌افزارهای تحت لینوکس میتوان قابلیت‌هایی همچون امکان انتقال و اجرا در سایر سیستم‌عامل‌ها (Portability)، پایداری (stability) و انعطاف‌پذیری (flexibility) را ذکر کرد.

نقاط قوت لینوکس عبارتند از (ساق پا !!!)

- ۱- امنیت بسیار بالا (به دلیل وجود دیوار آتش یا firewall در هسته سیستم عامل و همچنین وجود لایه‌های امنیتی مختلف در آن، تعداد انگشت شمار ویروس‌های کامپیوتری شناخته شده برای آن)
- ۲- پایداری بسیار بالا (به دلیل طراحی درست هسته (به نحوی که اشکال در یک نرم‌افزار باعث ناپایداری کل سیستم نمی‌گردد).
- ۳- سرعت بالای سیستم عامل (به دلیل در دسترس بودن کد متن برنامه و در نتیجه امکان کامپایل برنامه برای یک سخت‌افزار خاص)
- ۴- قیمت بسیار پایین آن (خود سیستم عامل و اکثر نرم‌افزارهای آن به صورت رایگان در دسترس هستند)

لینوکس یک هسته کاملاً چند وظیفه‌ای است یعنی توانایی انجام چند کار را به صورت همزمان دارد برخلاف DOS که یک سیستم عامل تک وظیفه‌ای بود. به علاوه لینوکس سیستمی چند کاربری است به این مفهوم که چند کاربر می‌توانند به طور همزمان و بدون دخالت در وظایف یکدیگر از یک سیستم استفاده کنند.

محیط‌های کاری در لینوکس:

سیستم عامل لینوکس در ابتدا در حالت متنی به وجود آمد، سپس دو رابط گرافیکی GNOME و KDE ساخته شدند. امروزه نیز رابط‌های گرافیکی بیشتری ساخته شده که از مهم‌ترین آن‌ها FVWM (Free Virtual Windows Manager) را می‌توان نام برد.

نمونه سوال تستی) کدام یک از موارد زیر جزء رابط‌های گرافیکی سیستم عامل لینوکس نمی‌باشد؟

الف) GNOME ب) KDE ج) FVWM د) Terminal

ورود به محیط متنی:

۱- از قسمت منوی اصلی به System Tools رفته و برنامه Terminal را اجرا می‌کنیم.

۲- در پنجره run program، xterm را تایپ کرده و سپس enter می‌زنیم.

۳- از کلیدهای ترکیبی Ctrl+Alt+(F1-F7) استفاده می‌کنیم.

محیط کاری GNOME:

یکی از محیط‌های گرافیکی که در لینوکس وجود دارد، GNOME است. این محیط در لینوکس به عنوان پیش‌فرض انتخاب شده است. به محیط‌های گرافیکی GUI می‌گویند. کلمه GUI مخفف Graphical User Interface به معنی واسطه گرافیکی کاربر است. آیتم‌هایی که در این محیط گرافیکی می‌بینید عبارتند از :

۱- محیط کاری (desktop): فضای اصلی کار است که بیشترین فضای صفحه را اشغال کرده است. کلیه پنجره‌ها روی این محیط قرار می‌گیرند. مدارک، پوشه‌ها، میان‌برها و دیگر داده‌ها نیز روی این محیط قرار می‌گیرند.

۲- پنل (Panel): نوار بلندی است که سرتاسر پایین یا بالای محیط کاری را گرفته است و تعدادی میانبر و آپلت (Applet) روی آن قرار دارد. پنل قابل تغییر و پیکربندی است و می‌توان ابزارها یا آپلت‌هایی را به پنل اضافه کرد.

۳- دکمه منوی اصلی: معمولاً اولین دکمه از سمت چپ (بالا یا پایین) روی پنل است که به شکل یک کلاه قرمز (آرم GNOME) است. این منو شامل زیرمنوها، آپلت‌ها و ابزارهای کاربردی است. بیشتر برنامه‌های لینوکس را می‌توان از آنجا یافت.

۴- میانبرها (Launcher): آیکون‌هایی هستند که روی محیط‌های کاری می‌آیند و شما با کلیک روی آن‌ها می‌توانید به برنامه یا محیط مربوطه دسترسی پیدا کنید. مثلاً اگر روی آیکن مرورگر وب کلیک کنید به آن مرورگر وب دسترسی خواهید داشت.

۵- پوشه Home: این پوشه دسترسی ما را به فهرست‌های مخصوص کاربران که در `/home/your user` قرار دارند امکان‌پذیر می‌کند.

اضافه کردن اپلت‌ها: روی پنل کلیک راست کنید، Applet مورد نظر را انتخاب کنید. در پنجره جدید اگر روی ابزار مورد نظر خود کلیک کنید، آن ابزار به پنل اضافه خواهد شد.

استفاده از راهنما (help): در روی پنل دکمه‌ای وجود دارد به اسم help یا به شکل علامت سوال که با کلیک روی آن می‌توانید به راهنمای GNOME دسترسی پیدا کنید. این راهنما یکی از مفیدترین ابزارهای است که میتواند به شما کمک بسیاری کند.

پایان کار در GNOME: در منوی اصلی GNOME دکمه‌ای به نام Log out وجود دارد که شما با استفاده از این دکمه می‌توانید عمل shutdown یا reboot کردن را انجام دهید.

تمرینها:

ابتدا برنامه VMware را در ویندوز اجرا کنید. سپس Red hat Linux را انتخاب کرده و در پنجره ظاهر شده `power on this virtual machine` را کلیک کنید تا لینوکس اجرا شود. سپس با نام کاربری و رمز عبور خود به محیط کاری لینوکس وارد شوید.

۱- با توجه به توضیحاتی که در مورد محیط گرافیکی GNOME در این گزارش داده شد این محیط و قسمت‌های مختلف آن را بررسی کنید.

۲- با استفاده از روش‌هایی که در گزارش کار ذکر شد از محیط گرافیکی وارد محیط متنی لینوکس شوید. کدام یک از ترکیب کلیدهای گفته شده برای رفتن به محیط گرافیکی استفاده می‌شود؟

۳- برای برنامه Terminal روی دسکتاپ یک میانبر ایجاد کنید.

۴- در محیط کاری که قرار دارید background را تغییر دهید.

۵- درباره محیط‌های گرافیکی KDE و FVWM از اینترنت جستجو کرده و اجزای اصلی آنها را معرفی نمایید. شباهت‌ها و تفاوت‌های این محیط‌های گرافیکی با یکدیگر و با GNOME در چیست؟

۶- درباره استانداردهای POSIX تحقیق کنید.

گزارش بندهای ۲ تا ۷ را در گزارش کار خودتحويل دهید.

آشنایی با سیستم فایل لینوکس

در یک سیستم مبتنی بر یونیکس هرچیزی یک فایل است یا یک فایل به عنوان نماینده خود دارد. در سیستم عامل لینوکس درست مثل یونیکس هیچ تفاوتی بین فایل و دایرکتوری وجود ندارد، به این معنا که یک دایرکتوری هم فایلی است که حاوی اسامی تعدادی فایل یا دایرکتوری دیگر است. برنامه‌ها، سرویس‌ها، متن‌ها و تصاویر و سایر موارد مشابه و همچنین دستگاه‌های ورودی و خروجی (input and output devices) و عموماً همه ابزارهای سیستمی همگی با یک فایل به سیستم معرفی می‌شوند. اگر بخواهیم همگی این فایل‌های متنوع را در یک ساختار منطقی مرتب کرده و نگهداری کنیم باید به آن‌ها به صورت یک ساختار درختی نگاه کنیم که روی هارد دیسک استقرار یافته است. این ساختار درختی از یک ریشه اصلی یا root تشکیل شده و شاخه‌های بزرگتر منشعب شده از آن دارای زیرشاخه و شاخه‌های انتهایی دارای برگ‌هایی هستند که همان فایل‌های ما هستند.

فایل‌ها در یک ساختار درختی از دایرکتوری‌ها ذخیره می‌شوند. هر دایرکتوری می‌تواند حاوی فایل‌ها و دایرکتوری‌های دیگری باشد. سیستم فایل لینوکس مثل یک درخت وارونه است که در بالاترین نقطه، دایرکتوری ریشه قرار دارد که به وسیله یک اسلش (/) نشان داده می‌شود. در زیر آن دایرکتوری‌های عمومی و سیستمی لینوکس قرار می‌گیرند مانند bin, dev, home و tmp. هر کدام از این دایرکتوری‌ها محتوی دایرکتوری‌های دیگر هم هستند.

فایل سیستم مربوط به ویندوز:

اولین فایل سیستم ویندوز FAT16 (مخفف File Allocation Table) بود. بعد از آن FAT32 عرضه شد و همراه با ویندوز XP، فایل سیستم NTFS (مخفف New Technology File System) عرضه شد.

فرق FAT32 و NTFS:

- ۱- از لحاظ نهایت حجم فایل: نهایت حجم فایل بر روی FAT32 چهار گیگا بایت می‌باشد اما نهایت حجم فایل روی NTFS، به اندازه کل درایو می‌تواند باشد.
- ۲- از لحاظ امنیت FAT32: رمزگذاری توسط FAT32 پشتیبانی نمی‌شود اما NTFS پشتیبانی میکند.
- ۳- از لحاظ نهایت طول نام فایل: در هر دو سیستم فایل، مجموع مسیرفایل + نام فایل + پسوند فایل نباید بیشتر از 259 کاراکتر شود.

۴- از لحاظ ذخیره سازی تغییرات: درایوی که FAT32 باشد، تغییرات بر روی ساختار فایل ها در لحظه خاموش شدن ویندوز ثبت نهایی می شود اما درایو NTFS در همان لحظه ثبت می کند. بنابراین اگر تغییراتی بر روی یک درایو FAT32 داده باشیم و سیستم به طور ناگهانی خاموش شود درایو FAT32 نیاز دارد که قبل از لود شدن ویندوز برای بررسی صحت داده ها Scan Disk شود اما درایو NTFS نیاز به این کار ندارد.

فایل سیستم مربوط به لینوکس:

- ❖ در ابتدا فایل سیستم لینوکس Minix FS بود سپس بر اساس آن فایل سیستم هایی مانند ext02, ext01, ext03 و در نهایت ext04 ارائه شد.
- ❖ فایل سیستم لینوکس از امنیت بالاتری نسبت به ویندوز برخوردار است.
- ❖ در نام فایل های لینوکس همه نوع کاراکتر به جز / میتوان استفاده کرد در حالی که در فایل سیستم های ویندوزی، کاراکترهای “ |? <> : * \ / غیرمجاز است.
- ❖ در لینوکس همه چیز یک فایل به حساب می آید حتی یک پوشه. بنابراین یک فایل و پوشه هم نام در کنار هم نمی توانند قرار گیرند. معمولاً کاربران لینوکس به پوشه ها Directory می گویند که دقیقاً همان مفهوم Folder در ویندوز را دارد. البته کلمه Directory بیشتر در محیط Terminal پیشنهاد می شود. در محیط Desktop بهتر است از همان کلمه Folder استفاده شود.
- ❖ کاربران عادی در لینوکس اجازه ذخیره کردن فایل ها را در خارج از پوشه home مربوط به خود را ندارند.
- ❖ لینوکس از ساختار FHS در مدیریت پوشه استفاده میکند.
- ❖ پسوند در لینوکس: در حالت کلی لینوکس تمایل به استفاده از پسوند برای فایل ها ندارد و مفهوم پسوند بیشتر در مورد فایل های مورد استفاده کاربر (سندها، عکسها و غیره) استفاده میشود. در لینوکس از محتوای فایل در مورد آن تصمیم گیری میشود و نه پسوند آن.

انواع فایل ها در لینوکس:

۱- فایل های معمولی (regular files):

- بیشتر فایل ها فقط معمولی هستند که حاوی اطلاعات معمولی مثل متن، کدهای اجرایی برنامه ها، ورودی و خروجی های برنامه های کاربردی دیگر می باشند.

۲- فایل های اختصاصی:

- دایرکتوری: فایلی حاوی اسامی فایل های دیگر
- فایل های ویژه: حاوی مکانیزم های چگونگی ورود و خروج اطلاعات به کامپیوتر (همه فایل های موجود در مسیر /dev از این دسته هستند).

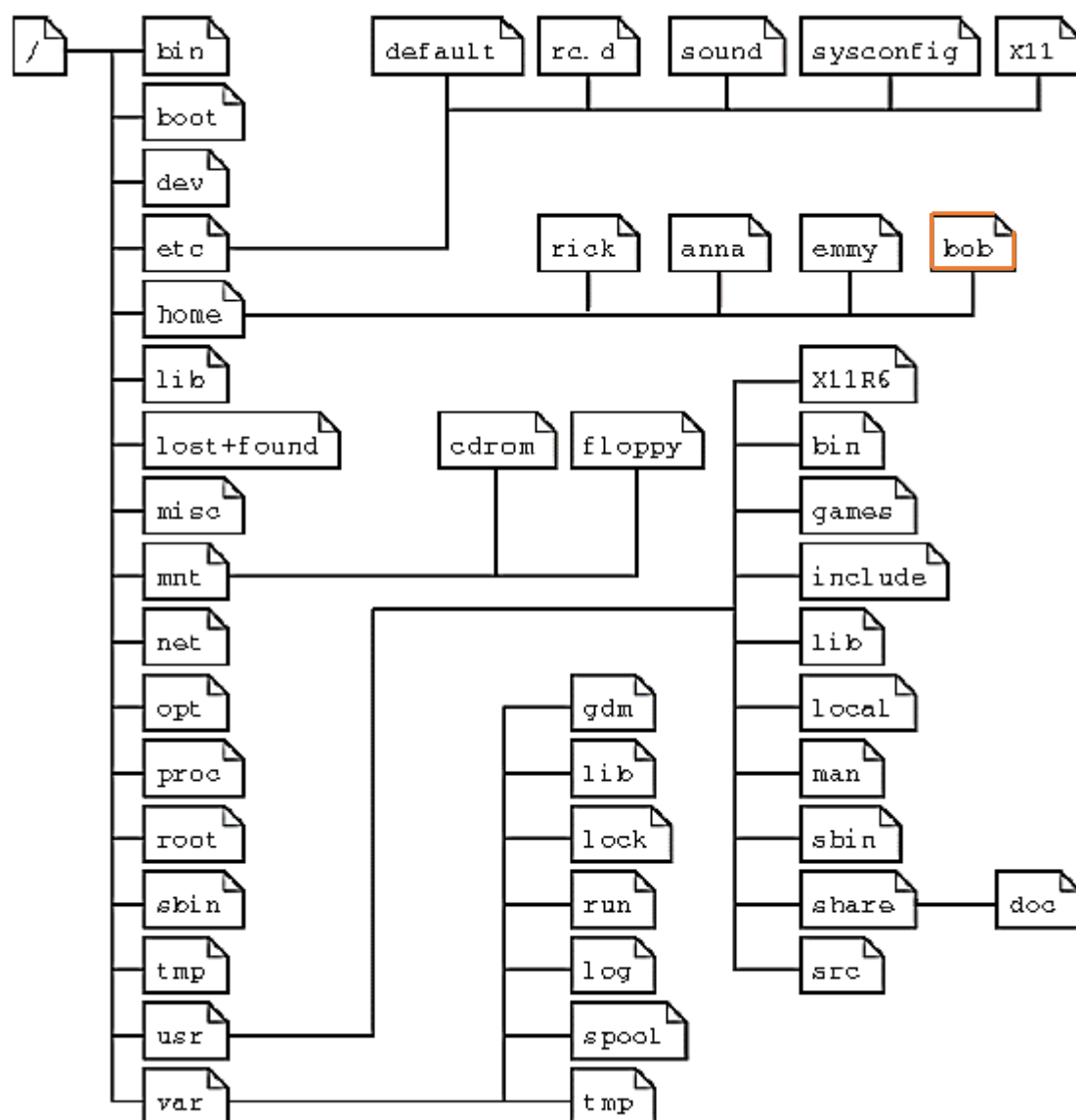
- **Links:** سیستمی را فراهم می‌کند که یک فایل یا دایرکتوری در نقاط مختلف ساختار درختی بدون نیاز به تکرار محتوی قابل مشاهده و در دسترس باشند.
- **Socket یا Domain:** نوع به خصوصی از فایل‌ها که همانند سوکت‌های TCP/IP روندهایی را برای کنترل دسترسی به فایل سیستم در شبکه به صورت کنترل شده فراهم می‌آورند.
- **Named Piped:** کمابیش عملی همانند سوکت‌ها انجام می‌دهند ولی متفاوت با روالی که سوکت‌ها در پیش می‌گیرند.

در لینوکس دستور ls به همراه آرگومان -l نوع فایل هر فایل را مشخص می‌کند. که در ادامه با این دستور بیشتر آشنا خواهیم شد.

به طور خلاصه FS یا File System، لایه‌ای است در هسته هر سیستم عامل که وظیفه آن کار با فایل‌ها و مدیریت آن است. کارهایی مثل ایجاد فایل، تعیین سطح دسترسی فایل، تعیین امنیت فایل، رمزگذاری فایل‌ها و کاراکترهای مجاز در نام فایل.

بخش‌های درخت سیستم فایل در لینوکس:

در شکل زیر با انواع شاخه‌ها و بخش‌های مختلف درخت سیستم فایل در لینوکس آشنا می‌شویم.



شاخه /bin: در این شاخه، دستورات سیستم عامل که برای تمام کاربران قابل دسترسی هستند و برخی دستورات مدیریتی سیستم قرار می گیرند. فرامین عمومی سیستم عامل لینوکس در این دایرکتوری قرار دارند مثل `ls`، `sort` و `chmod`.

شاخه /boot: فایل های مربوط به نرم افزار مدیریت منوی `boot` که در ابتدای راه اندازی سیستم مشاهده می شود در این پوشه قرار می گیرند. این شاخه می تواند در یک پارتیشن جداگانه بوده و به پوشه `boot` متصل شده باشد، حاوی برنامه راه انداز بوت سیستم عامل است مانند **Grub** و **Lilo**.

شاخه /home: در این شاخه دایرکتوری های خانگی کاربران سیستم و اطلاعات آن ها قرار می گیرد. فایل های شخصی مربوط به کاربران مختلف سیستم عامل مثل عکس ها، اسناد، ویدئو و ... در این پوشه قرار می گیرند. هر کاربر یک پوشه هم نام با نام کاربری اش در این پوشه خواهد داشت. مثلاً در شکل صفحه قبل کاربری به

نام bob در پوشه home یک پوشه برای خود دارد که داده‌های خود را فقط می‌تواند داخل آن ایجاد یا ویرایش کند. بهتر است این شاخه در یک پارتیشن جداگانه قرار گرفته و متصل شود. این کار حفاظت از داده‌های کاربران را بالاتر خواهد برد. ضمناً در صورت نیاز به فرمت و نصب مجدد سیستم، اطلاعات کاربران دست نخورده باقی مانده و پس از نصب مجدد، کلیه تنظیمات شخصی و اطلاعات آن‌ها بدون تغییر باقی خواهند ماند.

شاخه /root: فایل‌های مشخصه کاربر ریشه‌دار این پوشه قرار می‌گیرد.

شاخه /tmp: فایل‌های پوشه‌های موقتی کاربر یا سیستم عامل در این پوشه قرار می‌گیرد، محل قرار گیری برخی فایل‌های موقتی برنامه‌های کاربردی هم هست.

شاخه /usr: برنامه‌ای که کاربر یا سیستم عامل نصب می‌کند در این پوشه قرار می‌گیرد. مثلاً **XWindow** در این شاخه قرار دارد. همچنین برخی دستورات و دستورات مدیریتی نیز در مسیرهای /usr/bin و /usr/sbin قرار می‌گیرند.

شاخه /sbin: دستورات و برنامه‌های مدیریتی سیستم در این شاخه قرار می‌گیرند که مخصوص کاربر root است.

شاخه /dev: محل قرار گیری نقطه دسترسی‌های ابزارهای سخت‌افزاری است. مثلاً تمام پورت‌ها، پارتیشن‌ها و غیره در اینجا دارای یک فایل هستند. برای مثال فایل /dev/fd0 نشان دهنده فلاپی درایو سیستم است و به همین ترتیب سایر فایل‌های موجود.

شاخه /etc: در این شاخه فایل‌های پیکربندی برنامه‌های سرویس‌دهنده و برخی فایل‌های سیستمی دیگر قرار می‌گیرند. مثلاً **فایل پیکربندی سرویس دهنده وب آپاچی**، فایل مربوط به اطلاعات کاربران و گروه‌های کاری (passwd, group).

شاخه /lib: این شاخه همان‌طور که از نامش پیداست محل قرارگیری فایل‌های کتابخانه‌ای برنامه‌ها است. این کتابخانه‌ها توسط برنامه‌های کاربردی و ابزارهای برنامه‌نویسی به کار گرفته می‌شوند.

شاخه /mnt: این شاخه معمولاً دارای زیرشاخه‌هایی مانند floppy و CD-ROM بوده و محل اتصال درایوهای floppy و CD-ROM می‌باشد. همچنین پارتیشن‌های دیگر دیسک سخت را نیز می‌توانید در این شاخه متصل نمایید. اتصال و برداشتن پارتیشن‌ها و درایوها با استفاده از دستور mount صورت می‌گیرد.

شاخه /proc: این شاخه یک سیستم فایل مجازی است که برخی اطلاعات مربوط به سیستم و هسته از آن قابل دستیابی هستند.

شاخهٔ /var: این شاخه که معمولاً در کامپیوترهای سرویس‌دهنده در یک پارتیشن جداگانه قرار می‌گیرد، مخصوص برنامه‌های سرویس‌دهنده‌ای مانند وب و FTP و بانک‌های اطلاعاتی است. مثلاً فایل‌های مربوط به یک وب سایت می‌توانند در `/var/www` قرار بگیرند.

سؤال) پوشه **Program Files** در لینوکس کجاست؟

در لینوکس پوشه **Program Files** وجود ندارد. چون ساختار سیستم فایل لینوکس کاملاً با ویندوز متفاوت است در لینوکس هر فایل اجرایی برنامه بر اساس تقسیم‌بندی‌های خاصی در یکی از پوشه‌های `/usr/bin` یا `/usr/sbin` قرار می‌گیرد. نرم‌افزارهای محلی نصب شده نیز در پوشه‌های `/opt` یا `/usr/local` قرار می‌گیرند. برخی از فایل‌های اجرایی نیز در `/sbin` یا در `/usr/sbin` قرار می‌گیرند.

مسیرهای نسبی و مطلق:

مسیری که با علامت `/` شروع شود و بیان‌کنندهٔ مسیر از ریشهٔ سیستم باشد مسیر مطلق نام دارد. برای مثال `/usr/bin/wc` یک مسیر مطلق است. مسیری که با `/` شروع نمی‌شود یک مسیر نسبی است. برای مثال `bin/wc` یک مسیر نسبی است.

کار با فایل‌ها و دایرکتوری‌ها در لینوکس:

در سیستم عامل لینوکس، جهت انجام عملیات گوناگون بر روی فایل‌ها و دایرکتوری‌ها فرامین مختلفی در نظر گرفته شده است. این فرامین در محیط گرافیکی و محیط متنی قابل استفاده می‌باشند. انجام کلیهٔ عملیات در محیط گرافیکی لینوکس مثل سیستم عامل ویندوز و از طریق منوها و با استفاده از ماوس قابل انجام است. در محیط متنی با تایپ دستور، عملیات مورد نظر بر روی فایل یا دایرکتوری انجام می‌شود. در زیر لیستی از دستوراتی که در کار با فایل‌ها و دایرکتوری‌ها استفادهٔ فراوان دارند ذکر شده است:

دستور `ls`: مشاهدهٔ محتویات یک دایرکتوری

دستور `cat`: مشاهدهٔ محتویات یک فایل، ایجاد فایل و تغییر محتوای یک فایل

دستور `pwd`: مشاهدهٔ مسیری که هم‌اکنون کاربر در آن قرار دارد.

دستور `cd`: تغییر مسیر به دایرکتوری مورد نظر

دستور `mv`: انتقال و تغییر نام فایل

دستور `cp`: کپی کردن فایل

دستور `ln`: ساختن لینک برای یک فایل

دستور `rm`: حذف یک فایل

دستور `mkdir`: ساختن دایرکتوری

دستور `rmdir`: حذف دایرکتوری

امنیت فایل‌ها در لینوکس:

سیستم‌های مبتنی بر یونیکس به معنای واقعی سیستم‌های چندکاربره هستند و برای هر فایل به تنهایی یا یک دایرکتوری می‌توان سطوح دسترسی کاربران و گروه‌های کاربری تعریف کرد و هر کاربر به صورت پیش-فرض دارای یک محدوده امن اطلاعاتی از فایل‌های شخصی خانگی است.

هر کاربر در سیستم لینوکس یک دایرکتوری شخصی به نام `home` خواهد داشت که کلیه اختیارات دسترسی فایل در آن برای او مجاز بوده و می‌تواند در آن به ایجاد و حذف فایل یا عملیات دیگر بپردازد و هیچ یک از کاربران یا گروه‌های دیگر به جز کاربر ریشه `root` در حالت عادی و بدون اجازه او نخواهند توانست به اطلاعات او دسترسی یابند.

در سیستم‌های مبتنی بر یونیکس مثل لینوکس هر کاربری که مالک یا ایجادکننده یک فایل یا دایرکتوری باشد خواهد توانست با تعیین سطوح دسترسی فایل یا دایرکتوری مربوطه برای دیگر کاربران یا گروه‌های کاربری آنان را از امکان خواندن و ایجاد تغییر و یا اجرای فایل اجرایی محروم یا بهره‌مند سازد. بدیهی است که مجوزهای تعریف شده برای یک گروه کاربری برای تمام اعضای آن گروه اعمال می‌شود و هر کاربری که عضو آن گروه باشد از تمام مزایا یا محدودیت‌های وصف شده بهره‌مند خواهد شد. به طور کلی سه سطح دسترسی برای هر فایل یا دایرکتوری تعریف می‌شود که عبارتند از:

۱- مجوز خواندن یا `read permission (r)`: به یک کاربر اجازه خواندن محتوای یک فایل را می‌دهد و برای تمام دایرکتوری‌ها امکان لیست کردن محتویات داخل دایرکتوری را برای کاربر فراهم می‌سازد.

۲- مجوز ایجاد تغییرات یا `write permission (w)`: امکان ایجاد تغییرات در محتوای فایل‌ها را به کاربر می‌دهد. برای دایرکتوری‌ها این گزینه امکان ایجاد فایل جدید در دایرکتوری یا حذف فایل‌های موجود در آن را برای کاربر مهیا می‌سازد حتی اگر ایم کاربر یا گروه کاربری مالک یا ایجاد کننده اصلی فایل یا دایرکتوری نباشد.

۳- مجوز اجرا یا `execute permission (x)`: این مجوز در مورد فایل‌های اجرایی امکان اجرای فایل را به یک کاربر می‌دهد و در مورد یک دایرکتوری این گزینه امکان ورود به داخل دایرکتوری را به کاربر یا گروه خاص می‌دهد. توجه داشته باشید که این مجوز را با مجوز `read` اشتباه نگیرید، چون صرفاً با داشتن مجوز `execute` و بدون داشتن مجوز `read` کاربر یا گروه می‌تواند وارد دایرکتوری خاصی شوند ولی امکان خواندن محتویات داخلی آن دایرکتوری به هیچ عنوان برایشان فراهم نخواهد بود.

در لینوکس برای تعیین سطح دسترسی کاربران به یک فایل یا دایرکتوری می‌توان از دستور `chmod` استفاده کرد. با استفاده از این دستور می‌توان مشخص نمود هریک از کاربران اجازه انجام چه عملیاتی را بر روی فایل یا دایرکتوری مورد نظر دارند.

در لینوکس هر فایل یک مالک دارد. با استفاده از دستور `chown` می‌توان مالکیت یک فایل را به کاربر یا گروه کاربری دیگری تغییر داد.

روش‌های مختلف کمک گرفتن در مورد یک دستور:

- ۱- تایپ `help`-- بعد از نام دستور. مثال:
`cp -help`
- ۲- تایپ `man` قبل از نام دستور. مثال:
`man ls`
- ۳- تایپ `whatis` قبل از نام دستور. مثال:
`whatis pwd`
- ۴- تایپ `info` قبل از نام دستور. مثال:
`info cut`
- ۵- یافتن دستور مورد نظر از بین دستورات لینوکس: تایپ `apropos` و سپس یک کلمه کلیدی بعد از آن.

مثال: `apropos remove`

نکته: اگر راهنمای یک دستور، چندین صفحه و طولانی بود، مرور آن در محیط ترمینال ممکن است سخت به نظر برسد. برای اینکه راهنما را صفحه به صفحه مرور کنید، در انتهای این دستورات، عبارت `|less` را تایپ کنید.

مثال: `man ls |less`

برای خروج از این محیط، کلید `q` را بفشارید.

رفع مشکل Space در نام یک فایل یا پوشه:

اگر در نام پوشه‌ها یا فایل‌ها، کاراکتر فاصله تایپ شده باشد، هنگام نوشتن آدرس آن فایل یا پوشه در خط فرمان، باید دقت بیشتری داشته باشید. چون فاصله در خط فرمان علامت جداکننده آرگمان‌هاست و خط فرمان به اشتباه می‌افتد. یعنی مثلاً وقتی برای ساخت یک پوشه، تایپ می‌کنید:

`mkdir ~/Desktop/new folder`

خط فرمان فکر می‌کند `folder` یک آرگمان برای `mkdir` است (چون با فاصله از قسمت قبلی جدا شده) بنابراین خطا دریافت خواهید کرد و یا پوشه‌ای که می‌سازد به نام `new` خواهد بود. برای اینکه به خط فرمان بفهمانید که در اینجا فاصله جزئی از نام پوشه است، به یکی از روش‌های زیر عمل کنید:

۱- استفاده از دابل کوتیشن "در اطراف نام فایل یا پوشه (دور مسیر) مثال:

`mkdir ~/Desktop/"new folder"`

۲- استفاده از یک بک اسلش \ قبل از فاصله. مثال: `mkdir ~/Desktop/new\ folder`

آشنایی با چند دستور پرکاربرد:

- ۱- دستور `Cut` کردن یا همان انتقال فایل: مثال:
`mv ~/Desktop/x.txt ~/Documents`
این دستور، فایل به نام `x.txt` را از روی دسکتاپ کاربر جاری به پوشه `Documents` او منتقل میکند.
- ۲- دستور تغییر نام پوشه: اگر در دستور `mv` آدرس مبدأ و مقصد یکی باشد و فقط نام فایل عوض شده باشد، این به معنی تغییر نام فایل است. مثال:
`mv ~/Desktop/x.txt ~/Desktop/y.txt`
این دستور، نام فایل `x.txt` را به `y.txt` تغییر میدهد.

rm ~/Desktop/x.txt

۳- دستور حذف فایل:مثال:

rm -r ~/Desktop/new\ folder

۴- دستور حذف پوشه: مثال:

cat ~/Desktop/x.txt

۵- دستور نمایش محتوای فایل:مثال:

تمرینها:

با نام کاربری و رمز عبور خود به سیستم login کنید و برنامه Terminal را ایجاد کنید.

- ۱- با استفاده از دستور man df فهرست کاملی از option های مختلف دستور df را به دست آورید. با استفاده از دستور df فضای دیسک سخت در لینوکس را بررسی نمایید. تعیین کنید هر پارتیشن به چه مسیری در سیستم فایل متصل شده است. میزان فضای استفاده شده و فضای آزاد روی هر پارتیشن را بررسی کنید. خروجی دستور را به همراه توضیحات کامل در گزارش کار خود بیاورید.
- ۲- با استفاده از دستور echo \$Home مسیر دایرکتوری خانگی خود را مشخص نمایید. در صورتی که بخواهید با استفاده از مسیر نسبی از دایرکتوری خانگی خود به دایرکتوری خانگی کاربر دیگری وارد شوید، از چه مسیر نسبی ای استفاده می کنید؟ دستور لازم را اجرا کنید. نتیجه این دستور چه خواهد بود؟

- ۳- به شاخه tmp در /var تغییر مسیر بدهید. حال تنها با استفاده از یک دستور به شاخه share در /usr و سپس به شاخه doc بروید. حال با دستور مناسب مشخص کنید در کدام مسیر قرار دارید.
- ۴- به دایرکتوری /proc بروید. با استفاده از دستور ls محتویات این شاخه را بررسی کنید. در این شاخه فایل هایی وجود دارند که اطلاعاتی در مورد سیستم و سخت افزار در اختیار ما قرار می دهد. با مطالعه فایل های موجود در این شاخه اطلاعاتی درباره cpu ای که سیستم از آن استفاده می کند، میزان حافظه RAM و حافظه SWAP به دست آورید و نتایج را در گزارش کار خود بیاورید.
- ۵- به دایرکتوری /etc تغییر مسیر بدهید. با استفاده از دستور cat فایل مربوط به کاربران و گروه های کاری را بررسی کنید. چه تعداد کاربر و گروه کاری روی سیستم تعریف شده اند؟ (option مربوط به شمارش سطر را به همراه دستور به کار ببرید).
- ۶- در دایرکتوری خانگی خود یک شاخه جدید ایجاد کنید. آیا می توانید این پوشه را به دایرکتوری همسطح دایرکتوری خانگی خود در سیستم فایل انتقال دهید؟ چرا؟
- ۷- کلیه فایل های XPM در مسیر /usr/share/pixmaps را به یک پوشه جدید در دایرکتوری خانگی خود کپی کنید. حال لیست فایل های کپی شده را بر اساس ترتیب الفبایی معکوس نمایش دهید.
- ۸- در دایرکتوری خانگی خود پوشه جدیدی ایجاد کنید. سپس کلیه فایل های موجود در دایرکتوری /etc را در این پوشه کپی کنید. (option دستور cp را به گونه ای به کار ببرید که کلیه فایل ها و زیردایرکتوری های مسیر /etc و فایل های موجود در آنها نیز کپی شوند).
- ۹- به پوشه ایجاد شده در تمرین قبل وارد شوید. سپس دو دایرکتوری در آن ایجاد کنید، یکی برای فایل هایی که نام آنها با حروف بزرگ آغاز می شود و دیگری برای فایل هایی که نام آنها با حروف کوچک آغاز می شود. سپس کلیه فایل ها را به شاخه مناسب انتقال دهید.

- ۱۰- فایل‌های باقی مانده در تمرین قبلی را حذف کنید. سپس دایرکتوری را به همراه تمام فایل‌ها و پوشه‌های آن با استفاده از یک تک دستور حذف کنید.
- ۱۱- در دایرکتوری خانگی خود یک لینک سمبولیک برای مسیر `/var/tmp` ایجاد کنید. کنترل کنید که لینک صحیح کار می‌کند. سپس یک لینک دیگر برای همین مسیر ایجاد کنید. حال اولین لینکی که ایجاد کرده‌اید را حذف کنید.
- ۱۲- نحوه استفاده از دستور `chmod` برای تعیین مجوزهای فایل توضیح دهید (سمبولیک و عددی). آیا می‌توانید مجوزهای دایرکتوری `/home` را تغییر دهید؟
- ۱۳- با استفاده از دستور `chown` مالکیت پوشه `/etc` را به نام کاربری خود و گروه کاربری که در آن هستید تغییر دهید.
- ۱۴- مجوز فایل `~/.bashrc` را به گونه‌ای تغییر دهید که فقط خودتان و گروه کاری که در آن قرار دارید بتوانند آن را بخوانند.
- ۱۵- تحقیق کنید پارتیشن `swap` در سیستم فایل لینوکس چه کاربردی دارد و بهتر است چه میزان فضایی برای آن در نظر گرفته شود؟
- ۱۶- درباره ساختار `FHS` برای مدیریت پوشه‌ها که در متن این گزارش به آن اشاره شد، تحقیق کنید.
- ۱۷- درباره راه‌اندازهای بوت `Lilo` (boot loader) و `Grub` که در متن این گزارش به آن‌ها اشاره شد، تحقیق کنید. تفاوت‌های عمده بین `Grub` و `Lilo` را شرح دهید.
- ۱۸- منظور از فایل پیکربندی سرویس دهنده وب آپاچی که در متن این گزارش به آن اشاره شده است، چیست؟
- ۱۹- درباره `Xwindow` که در متن این گزارش به آن اشاره شد، تحقیق کنید.
- گزارش‌های بند ۲ تا ۲۰ را در گزارش کار خود بیاورید.

آزمایشگاه سیستم عامل

آزمایش سوم

I/O Redirection و برنامه‌نویسی پوسته لینوکس (Shell Programming)

قبل از آنکه شروع به برنامه‌نویسی پوسته لینوکس کنیم، لازم است با موارد زیر آشنا شویم.

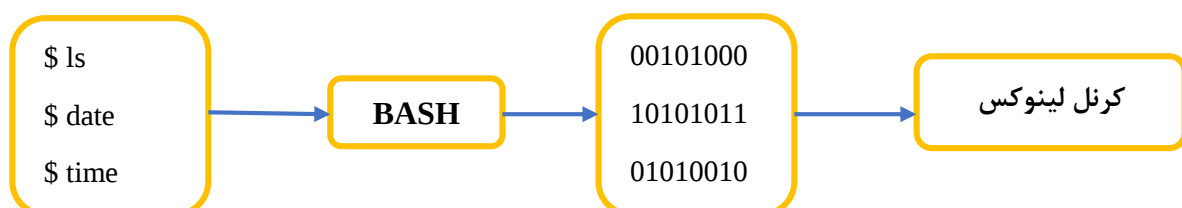
هسته (kernel): کرنل قلب سیستم عامل لینوکس است که منابع این سیستم عامل را مدیریت می‌کند. منظور از منابع امکاناتی است که در لینوکس در دسترس اند. برای مثال، امکانات ذخیره‌سازی دیتا، چاپ دیتا در پرینتر، حافظه، مدیریت فایل و غیره. کرنل تصمیم می‌گیرد چه کسی از این منابع استفاده خواهد کرد، برای چه مدت و چه موقع. برنامه‌های شما را اجرا می‌کند یا فایل‌های باینری اجرایی را ایجاد می‌کند. به طور کلی کرنل کارهای زیر را انجام می‌دهد:

- مدیریت ورودی/خروجی (I/O management)
- مدیریت فرایندها (process management)
- مدیریت وسایل (device management)
- مدیریت فایل (file management)
- مدیریت حافظه (memory management)

پوسته لینوکس (Linux Shell): همانطور که می‌دانید کامپیوتر زبان ۰ و ۱ که به زبان باینری معروف است، می‌فهمد. اوایل افراد می‌بایست دستورات را با استفاده از زبان باینری می‌نوشتند که نوشتن، خواندن و درک آن برای همه ما بسیار دشوار است. پس در سیستم عامل برنامه خاصی وجود دارد که پوسته نامیده می‌شود. پوسته دستورات یا فرمان‌های شما را که به انگلیسی هستند قبول می‌کند و آن را به زبان باینری ترجمه می‌کند. در واقع پوسته کار زیر را انجام می‌دهد:



مطابق شکل زیر وقتی شما یک فرمان را تایپ می‌کنید، پوسته آن را به زبان باینری تبدیل می‌کند تا برای کرنل قابل فهم باشد.



پوسته یک مفسر زبان خط فرمان است که دستوراتی را که از وسیله ورودی استاندارد (صفحه کلید) یا از یک فایل می‌خواند را اجرا می‌کند. لینوکس از یکی از معروف‌ترین پوسته‌های زیر استفاده می‌کند.

KSH (Korn Shell), CSH (C Shell), BASH (Bourne-Again Shell)

برای آنکه نوع پوسته سیستم عامل لینوکس خود را بدانید می‌توانید از فرمان زیر استفاده کنید:

\$ echo \$SHELL

تغییر جهت ورودی / خروجی استاندارد یا I/O Redirection :



اغلب دستورات ورودی را از صفحه کلید دریافت می‌کنند یا خروجی را به صفحه نمایش می‌فرستند، اما در لینوکس این امکان وجود دارد که خروجی را به فایل بفرستیم یا ورودی را از فایل بخوانیم. برای مثال، دستور `$ls` خروجی خود را به صفحه نمایش می‌فرستد؛ برای ارسال خروجی به یک فایل می‌توان این دستور را به صورت زیر نوشت:

نام فایل >ls

معنای این دستور این است که خروجی دستور `ls` را به فایلی که نام آن را نوشته‌ایم بفرست به جای آنکه در صفحه نمایش نشان بدهد. علامت‌هایی که برای تغییر ورودی و خروجی مورد استفاده قرار می‌گیرند عبارتند از: `>`, `>>`, `<`.

تغییر مسیر خروجی دستور:

نام فایل >دستور

`Ls -l /proc > ~/Desktop/1.txt` (مثال)

دستور اجرا می‌شود و خروجی آن در فایلی که نامش را داده‌ایم نوشته می‌شود. توجه داشته باشید که اگر این فایل قبلاً وجود داشته باشد خروجی دستور در آن نوشته می‌شود یعنی محتویات قبلی آن پاک می‌شود بدون آنکه به ما هشدار در این باره بدهد. اگر بخواهیم اطلاعات قبلی موجود در این فایل حفظ شود باید از عملگر `>>` استفاده کنیم.

نام فایل >>دستور

`Ls -l /proc >> ~/Desktop/1.txt` (مثال)

با استفاده از عملگر >> نتایج دستور فعلی در انتهای فایل نوشته می‌شود یعنی فایلی که نامش را داده‌ایم باز می‌شود و اطلاعات جدید در انتهای آن نوشته می‌شوند و به این ترتیب اطلاعات قبلی موجود در این فایل از بین نمی‌رود.

تغییر مسیر ورودی دستور:

نام فایل <دستور

برای آنکه اطلاعات لازم برای یک دستور از صفحه کلید خوانده شود می‌توان از یک فایل که نامش را داده‌ایم ورودی را دریافت کند.

head< ~/Desktop/1.txt (مثال)

لوله‌بندی فرامین یا **piping commands**:

Pipe: پایپ یا لوله روشی است برای اتصال خروجی یک برنامه یا دستور به ورودی یک برنامه یا دستور دیگر بدون آنکه از هیچگونه فایل موقتی استفاده شود. در واقع پایپ یک مکان ذخیره‌سازی موقت است که خروجی یک دستور موقتاً در آن ذخیره می‌شود و سپس به عنوان ورودی به دستور بعدی فرستاده می‌شود. از پایپ‌ها برای اجرای بیش از دو دستور در یک خط فرمان استفاده می‌شود

دستور دوم | دستور اول

(مثال)

more | ls: خروجی دستور ls به عنوان ورودی به دستور more داده می‌شود و به این ترتیب خروجی به صورت تمام صفحه یکجا نمایش داده می‌شود.

sort | who: خروجی دستور who به عنوان ورودی به دستور sort داده می‌شود و به این ترتیب لیست کاربران به صورت مرتب شده نمایش داده می‌شود.

l -l | who: خروجی دستور who به عنوان ورودی به دستور wc داده می‌شود و به این ترتیب تعداد کاربرانی که به سیستم log on کرده‌اند را می‌شمارد.

l -l | wc -l: تعداد فایلی که در دایرکتوری جاری هستند نمایش داده می‌شود.

grep pnu | who: اگر کاربر خاصی به نام pnu به سیستم log on کرده باشد نمایش داده می‌شود در غیر اینصورت هیچ چیز نمایش داده نخواهد شد.

grep 'pnu' | ls -al /proc: تمام فایل‌ها و دایرکتوری‌های موجود در دایرکتوری proc را با جزئیات لیست می‌کند و به عنوان ورودی به دستور grep می‌فرستد. دستور grep از میان این دیتا تنها خطوطی را

نمایش می‌دهد که در آن‌ها کلمه pnu نوشته شده باشد یعنی در واقع فایل‌هایی را لیست می‌کند که صاحب آن‌ها کاربر به نام pnu است.

ترکیب تغییر مسیرها: با مثال زیر نشان می‌دهیم که چگونه می‌توان تغییر مسیر ورودی و خروجی دستورات را با هم ترکیب نمود.

```
grep 'ssl' < ~/Desktop/1.txt > ~/Desktop/2.txt
```

فیلتر: فیلتر نوعی پردازش را روی ورودی خود انجام داده و سپس خروجی را تولید می‌کند. برای مثال فرض کنید فایلی داریم به اسم a.txt که در آن ۱۰۰ خط دیتا وجود دارد، از این فایل می‌خواهیم خطوط بیستم تا سی‌ام را در فایل دیگری به نام b.txt ذخیره کنیم. برای این منظور دستور زیر را می‌نویسیم:

```
tail +20 < a.txt | head -n 30 > b.txt
```

در این دستور head فیلتری است که ورودی‌اش را دستور tail می‌گیرد که در اینجا دستور tail از خط شماره ۲۰ از فایل a.txt شروع به انتخاب می‌کند و این خطوط را به عنوان ورودی به head می‌فرستد که خروجی‌اش به فایل b.txt تغییر جهت داده شده است.

تمرین‌ها:

۱- دستور cat فایل‌ها را به هم متصل کرده و همه آن‌ها را در خروجی استاندارد نمایش می‌دهد. با استفاده از این دستور سه فایل در مسیر دایرکتوری خانگی خود ایجاد کنید. سپس محتویات دو فایل اول را به انتهای فایل سوم اضافه کنید.

۲- خروجی دستور date را در یک فایل ذخیره کنید. سپس لیست فایل‌های موجود در /var را به انتهای این فایل اضافه کنید.

۳- با اجرای دستور زیر چه اتفاقی می‌افتد؟

```
>time; uptime >> time; date >> time; cat < time
```

۴- از دستور cut در خروجی دستور ls -l به نحوی استفاده کنید که فقط مجوزهای فایل‌های موجود در مسیر /etc را نمایش بدهد. سپس خروجی مرحله قبل را با استفاده از دستورهای sort و uniq به گونه‌ای فیلتر کنید که خروجی را مرتب کرده و خطوط تکراری را حذف نماید. سپس با دستور wc تعداد مجوزهای متفاوت در این دایرکتوری را شمارش کنید.

۵- راهنمای دستور grep را مطالعه کنید. با استفاده از این دستور خطوطی از راهنمای دستور less را که حاوی کلمه examine و فاقد کلمه file هستند را در فایل 3.txt در دسکتاپ ذخیره کند.