دانشگاه صنعتی اصفهان

دانشکده برق و کامپیوتر

آزمایشگاه سیستم عامل

دستور کار جلسه اول

آشنایی و شروع کار با سیستم عامل لینوکس

علی فانیان

زینب زالی

تابستان ۱۴۰۱

به نام خدا

ورود شما را به آزمایشگاه سیستم عامل خوش‌آمد می‌گوییم:) در این جلسه با موارد زیر آشنا می‌شوید:

۱- سیستم‌عامل‌های مبتنی بر unix

۲- کرنل لینوکس و آشنایی مختصر با ساختار سورس کرنل

۳- آشنایی مختصر با فایل سیستم لینوکس

۴- آشنایی و آشتی با CLI در لینوکس و دستورات پرکاربرد خط فرمان لینوکس

۵-آشنایی با انواع و دسترسی‌های فایل‌ها و معرفی ویرایشگر vim

نکته: سعی کنید مطالب مهم را از موارد بیان‌شده در پیش‌گزارش دستور کار یاد بگیرید نیازی به حفظ مطالب و مخصوصاً دستورها نیست. به مرور با استفاده زیاد، هر یک از دستورهای shell را که پرکاربرد هستند فرامی‌گیرید. به شکل‌ها دقت کنید و موارد بیان‌شده را در سیستم لینوکس خود پیگیری کنید مثلاً ساختار دایرکتوری ریشه یا ساختار کرنل. همچنین دستورات جدول‌ها را آن‌طور که خواسته شده امتحان کنید.

۱- سیستم عامل Unix

اولین نسخه این سیستم عامل در سال ۱۹۶۹ توسط تیمی از مهندسین آزمایشگاه Bell به سرپرستی Kenneth Thompson و Dennis Ritchie نوشته شد. در همین زمان زبان برنامه نویسی C ایجاد شد و Dennis Ritchie اولین کامپایلر C را نیز به وجود آورد. این زبان به عنوان ابزاری برای نگهداری ساختمان داده‌ها و ایجاد تغییرات در Unix به کار گرفته شد. با گذشت زمان بر قابلیت‌های یونیکس اضافه شد و شرکت‌های بزرگ نسخه‌های متفاوتی از این سیستم عامل را برای خود ایجاد کردند و به فروش رساندند. از جمله تیمی از دانشگاه برکلی سعی در ارتقاء یونیکس کردند و حاصل تلاش آنها سیستمی با نام Berkeley Software Distribution (BSD) شد.

در سال ۱۹۹۱ سیستم عاملی انقلابی با نام لینوکس، مبتنی و شبیه به یونیکس توسط Linus Torvalds نوشته شد. آقای Torvalds وقتی دید پروژه نوشتن این سیستم‌عامل بیش از اندازه بزرگ شده است متوجه شد برای ادامه‌ی کار نیاز به یک مدیریت نسخه ضروری است. از این رو

Git توسط ایشان و تیم‌شان توسعه داده شد. پس از آن تعداد افراد بسیاری کار را از راه دور زیر نظر آقای Torvalds ادامه دادند. کرنل لینوکس، یک پروژه متن‌باز بسیار بزرگ است که توسعه‌دهندگان زیادی از سرتاسر دنیا برای آن کد می‌نویسند و در نسخه‌های جدید کرنل، خطهای زیادی نسبت به نسخه قبلی تغییر می‌کند.

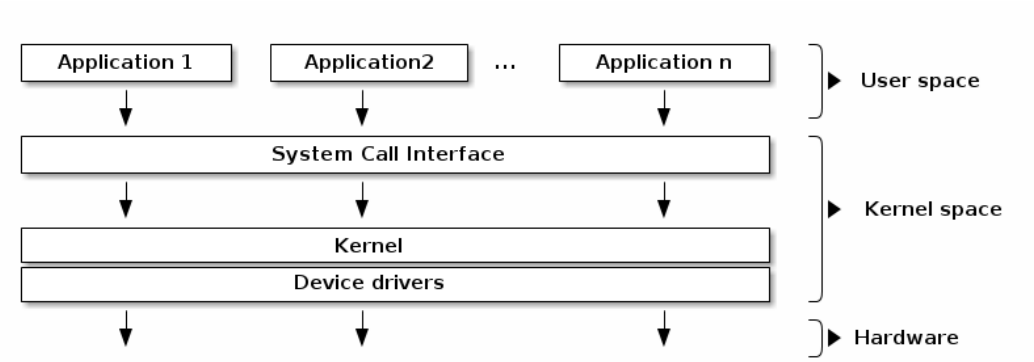
بعد از ایجاد کرنل لینوکس شرکت‌ها و گروه‌های مختلفی این کرنل را به همراه نرم‌افزارهای سیستمی و کتابخانه‌های جانبی در قالب توزیع‌های لینوکس (linux distribution) ارائه دادند. توزیع‌های لینوکس بسیار گسترده هستند و خانواده‌های متفاوتی دارند. معروفترین خانواده‌های این توزیع‌ها Debain، Fedora(RHEL)، Arch و OpenSUSE هستند. در بین این خانواده‌ها غالبا خانواده‌ی Debian به دلیل سادگی در بین کاربران عادی بیشتر استفاده می‌شود. معروفترین توزیع‌های این خانواده Ubuntu و mint است.

در این آزمایشگاه، شما با یکی از توزیع‌های لینوکس کار خواهید کرد. لینوکس یک سیستم عامل متن باز است که از ویژگی‌های اصلی آن ماژولار بودن آن است. به دلیل اینکه تمام توزیع‌های لینوکس کرنل مشابه استفاده می‌کنند و این کرنل مبتنی بر یونیکس است بنابراین یادگیری کار با یک توزیع، تجربه‌ی مشابه‌ی را در دیگر نسخه‌ها برای شما رقم خواهد زد.

در این جلسه با مفاهیم و بخش‌های اصلی این سیستم عامل آشنا می‌شویم.

۲- کرنل لینوکس

کرنل هر سیستم عامل، دسترسی و استفاده از سخت‌افزار سیستم را به صورت امن و عادلانه برای برنامه‌های کاربردی فراهم می‌کند. *شکل ۱*، نحوه قرارگیری لایه‌های مختلف سیستم‌عامل را نسبت به هم نشان می‌دهد. در سیستم‌عامل‌ها سه لایه‌ی اصلی سطح کاربر، سطح کرنل و سطح سخت‌افزارها در نظر گرفته می‌شود. هر برنامه‌ای که شما به صورت عادی می‌نویسید در سطح کاربر خواهد بود. برنامه‌های سطح کاربر برای اجرا نیازمند سخت‌افزارهای مختلفی هستند که برای مدیریت این دسترسی و اجرای همزمان برنامه‌های کاربری مختلف کرنل به جود آماده است. برای اینکه برنامه‌های کاربری نیازمندی خود را به کرنل اعلام کنند باید از مجموعه‌ای از توابع که کرنل در اختیارشان قرار داده است استفاده کنند. نام این مجموعه توابع که به نوعی حلقه‌ای اتصال سطح کاربر به سطح کرنل محسوب می‌شود را system call نهاده‌اند. این توابع به عنوان APIهای سطح کرنل به حساب می‌آیند که در سطح کاربر قابل صدا زدن هستند. این توابع با APIهای کتابخانه‌های متداول، متفاوت است. زیرا فراخوانی توابع این API منجر به تغییر حالت سیستم از کاربر به کرنل می‌شود. در حالت کرنل دسترسی به حافظه و سخت‌افزارها متفاوت از حالت سطح کاربر است.



شکل ۱ – لایه‌های مختلف سیستم

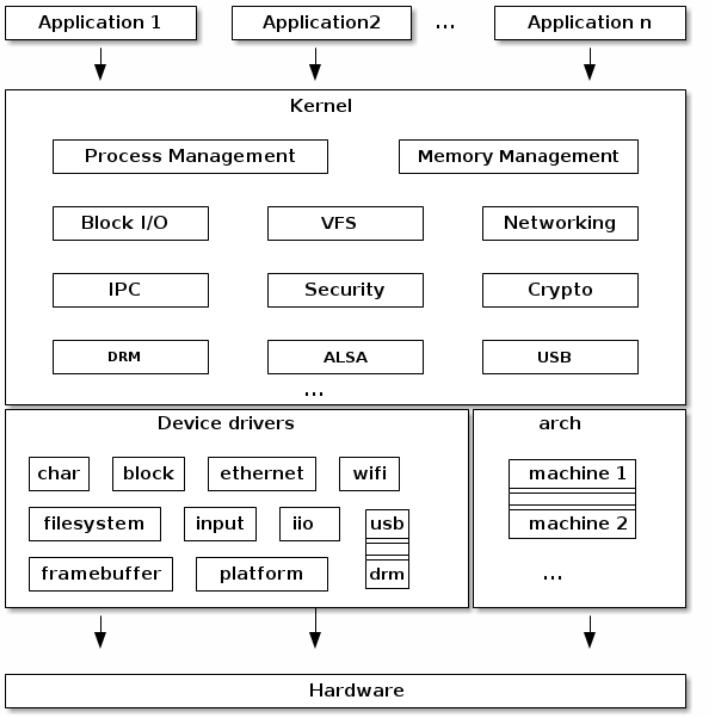
هدف از آزمایشگاه سیستم‌عامل این است که دانشجویان به سطح کرنل، روش استفاده از systemcallها و دستکاری‌های کرنل را آموزش ببینند. کدهای سطح کرنل را هم می‌توان به دو بخش کدهای هسته اصلی کرنل و کدهای درایورها یا ماژول‌های کرنل تقسیم کرد. *شکل ۲* لایه‌های کرنل را به خوبی نمایش می‌دهد.

- هسته اصلی کرنل

شامل عملیات بخش‌های مختلف سیستم مانند دسترسی به فایل، مدیریت پروسس‌ها، مدیریت حافظه و پیاده‌سازی پروتکل‌های لایه‌ی پایین شبکه است.

- ماژول‌ها و درایورها

این بخش از کدهای کرنل برای هدف خاص یا تجهیز خاص تعبیه شده اند. بنابراین این بخش می‌تواند در سیستم‌عامل‌های مختلف بسته به تجهیزات نصب شده یا اهدافی که استفاده می‌شوند متفاوت باشد. به طور مثال استفاده از تجهیزات صدا، پرینتر، مانیتور، کیبورد و ماوس در این قسمت قرار دارد.



شکل ۲- معماری کرنل لینوکس

سورس کرنل لینوکس (با دستور apt install linux-source) قابل دانلود است و در شاخه *usr*/src/ قرار می‌گیرد. با این دستورات در ادامه آشنا خواهید شد.

بعد از دریافت کد کرنل لینوکس با فایل‌ها و دایرکتوری‌های مختلفی روبرو خواهید شد. در ادامه به صورت مختصر برخی از مهمترین دایرکتوری‌های کد کرنل معرفی خواهند شد:

fs: کد فایل سیستم و درایورهای مختلف فایل سیستم

kernel: کدهای مربوط به مدیریت پروسس‌ها و threadها

mm: کد مدیریت حافظه

ipc: پیاده‌سازی system callهای مختلف مربوط به ارتباط بین پروسس‌ها (inter process communication)

arch: حاوی کدهای مربوط به سخت‌افزارهای مختلف مانند arm یا x86 است.

block: شامل کدهای مربوط به خواندن و نوشتن از دیوایس‌های بلاک است.

include: فایل‌های سرآیند

init: کد initialization که هنگام بوت سیستم اجرا می‌شود.

lib: توابع عمومی مختلف مانند جستجو، فشرده‌سازی، checksum و غیره.

net: پیاده‌سازی استک پروتکل‌های شبکه

بخش اصلی کرنل لینوکس به صورت یکپارچه نوشته شده است (monolitic). اما جهت انعطاف‌پذیری، امکان نوشتن ماژول‌های دلخواه و اضافه‌کردن آنها به کرنل وجود دارد. بدین ترتیب هر ماژولی قابلیت اضافه یا حذف‌شدن از کرنل را در زمانی که کرنل در حال اجرا است دارد و نیاز نیست با نوشتن یک ماژول جدید، کرنل را از ابتدا کامپایل و اجرا کنیم. به همین دلیل لینوکس را لایه‌ای یا ماژولار معرفی‌میکنند.

۳- فایل سیستم لینوکس

شما احتمالا بارها از فایل سیستم ویندوز استفاده کرده اید. فایل سیستم به معنی روشی است که سیستم‌عامل به شما فایل‌ها و دایرکتوری‌ها را ارائه می‌کند. به طور مثال در ویندوز استفاده از درایوهای مختلف Cو D یا محل ذخیره سازی عکس‌ها و فیلم‌ها که به طور مثال در پوشه‌ی Pictures در آدرس C:/Users/<user> قرار دارد آشنا هستید.

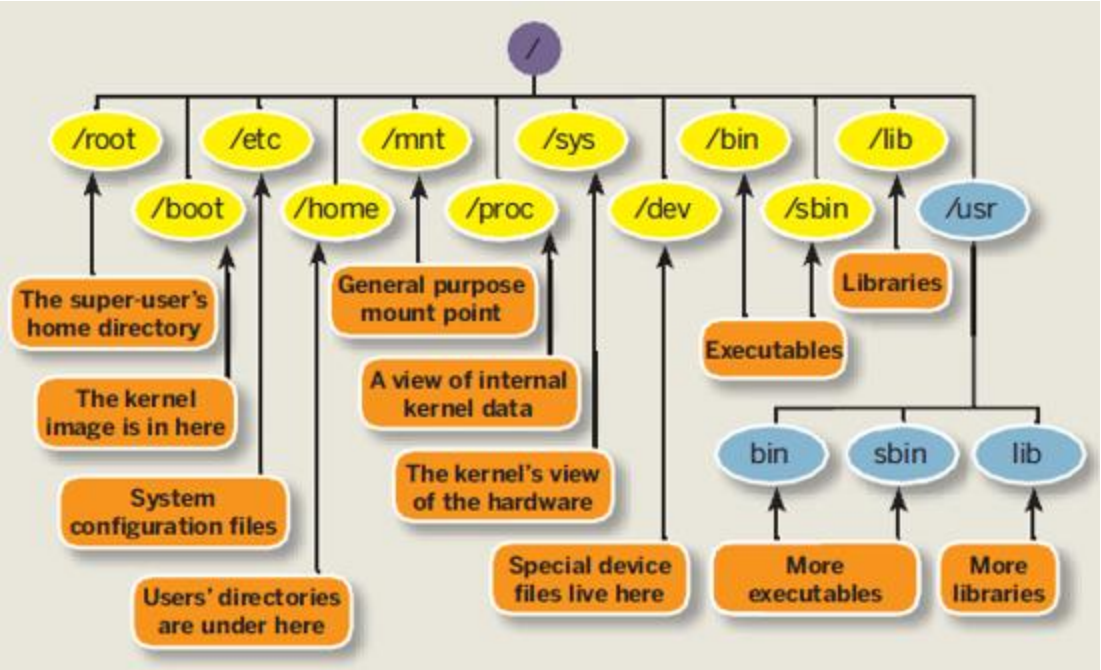
فکر می‌کنید در لینوکس کدام یک از موارد زیر در فایل سیستم است؟ پروسس‌ها؟ دیوایس‌ها؟ ساختمان داده‌های کرنل و پارامترهای تنظیمات کرنل؟ کانال‌های ارتباطی بین پروسس‌ها؟

اگر سیستم، مبتنی بر یونیکس باشد، همه موارد ذکرشده و موارد بسیار دیگری در فایل سیستم قرار می‌گیرد و به صورت فایل قابل دسترسی است. هدف اصلی فایل سیستم مدیریت و نمایاندن فضای ذخیره (storage) سیستم است. اما برنامه‌نویسان به منظور سادگی برای مدیریت آبجکت‌های دیگر هم از فایل سیستم استفاده می‌کنند. به همین دلیل هر آبجکتی به فضای نام فایل سیستم map می‌شود. برای مثال، فایل‌های دیوایس، راهی برای ارتباط برنامه‌های کاربردی با درایور درون کرنل است. آنها واقعا فایل‌های حاوی داده نیستند بلکه از طریق فایل سیستم کنترل می‌شوند و ویژگی‌های آنها روی دیسک ذخیره می‌شود (در آینده با نحوه برنامه‌نویسی ماژول کرنل و اضافه‌کردن این‌گونه فایل‌های دیوایس آشنا خواهید شد). فرض کنید شما روش نوشتن و خواندن از فایل را در سیستم‌عامل لینوکس یادبگیرید. به محض اینکه این آموزش تمام شود شما قادر خواهید بود در سوکت‌های شبکه، Pipeهای موجود در سیستم‌عامل (با این موارد جلسات آینده آشنا خواهید شد)، دیوایس‌ها مانند کیبورد و ماوس و خیلی از موارد دیگر اطلاعات خود را بنویسید و بخوانید چرا که تمامی این موارد به صورت فایل در اختیار شماست.

در لینوکس فایل سیستم به صورت یک ساختار سلسله‌مراتبی (درختی) یکتا با شروع از شاخه ریشه (/) شروع می‌شود. ریشه را می‌توان مانند My Computer در ویندوز در نظر گرفت. بدین ترتیب تمام مسیرهای فایل‌ها با / شروع می‌شوند و مانند ویندوز از پارتیشن‌های مختلف بهره نمی‌برد.

برای مشاهده ساختار سلسله مراتبی شاخه اصلی فایل سیستم در یونیکس می‌توانید دستور man hier را در ترمینال اجرا کنید. قسمتی از این سلسله مراتب و توضیح محتویات هر شاخه در شکل ۳ مشاهده می‌شود. همچنین جدول ۱ شرح مختصری از محتویات هر شاخه اصلی را نشان می‌دهد. آشنایی با این موارد برای کار با لینوکس بسیار مهم است. به طور مثال شما باید بدانید که بر خلاف ویندوز هر برنامه که نصب می‌شود اطلاعات تنظیمات آن، فایل گزارش‌های(Log) آن و فایل‌های اجرایی آن در پوشه‌های مختلفی قرار می‌گیرد.

برای هر کاربر یک دایرکتوری خانه ساخته می‌شود که در مسیر /home/username/ قرار دارد (username نام کاربر موردنظر است). همچنین ~ نیز نمادی از شاخه خانه است (دایرکتوری خانه یا */home/* یا ~).



شکل۳- فایل سیستم سلسله مراتبی لینوکس

جدول ۱: دایرکتوری‌های موجود در دایرکتوری root در لینوکس

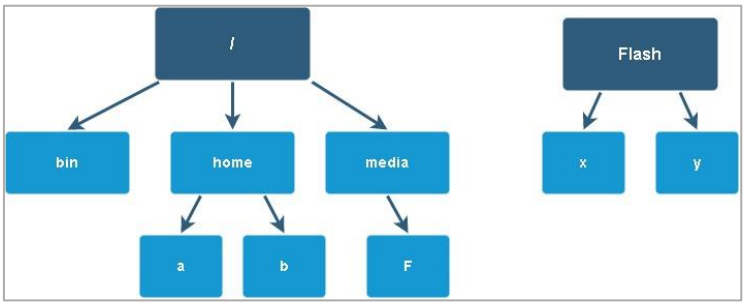
bin	دستورات اصلی سیستم و فایل باینری اجرایی برنامه‌های نصب‌شده. بعضی دستورات غیراصلی سیستم در /usr/bin/ نصب می‌شود.
boot	فایل‌های لازم جهت بوت سیستم
dev	فایل ارتباطی دیوایس‌های سیستم برای درایورها
etc	فایل‌های تنظیمات مربوط به سیستم و بوت سیستم(config files)
lib	کتابخانه‌های اصلی shared و ماژول‌های کرنل (شامل کتابخانه‌هایی که برای بوت سیستم و اجرای دستورات و برنامه‌های موجود در /bin/ و /sbin/ نیاز است
media	نقطه انتصاب (mount point) برای فضاهای ذخیره‌سازی جدا از سیستم (removable media) مانند حافظه فلش
mnt	نقطه انتصاب برایmount کردن موقت فایل سیستم توسط کاربر سیستم
opt	پکیج‌های افزودنی (add-on) نرم‌افزارهای سیستم
proc	محل قرارگرفتن اطلاعات مربوط به پروسس‌ها
run	دیتاهای مربوط به پروسس‌های سیستم از زمان بوت سیستم (برای مثال فایل حاوی pid پروسس‌ها)
sbin	فایل‌های باینری ضروری سیستم که فقط توسط root قابل اجرا هستند در این شاخه و در /usr/sbin/ و /usr/local/sbin قرار می‌گیرند. این فایل‌ها جهت بوت سیستم و ریکاوری آن نیاز است.
srv	دیتاهای مربوط به سرویس‌های اجرایی سیستم
sys	اطلاعات دیوایس‌ها، درایورها و بعضی ویژگیهای کرنل در این دایرکتوری ذخیره می‌شود.
tmp	فایل‌های موقت این فایل‌ها در رم قرار دارند.
usr	بخش اصلی دوم فایل سیستم (دایرکتوری فایل‌های سرآیند سیستم(include) ، فایل‌های آبجکت و کتابخانه‌ها، نرم‌افزارهایی که به صورت محلی توسط root نصب می‌شوند، در این شاخه قرار دارند)
var	فایل‌های دیتای متغیر مانند فایل‌هایlog ، فایل‌های cache و فایل‌های dump سیستم
root	شاخه مربوط به داده‌های کاربر ریشه (ادمین لینوکس)

گفتنی است مسیر یک فایل را می‌توان به دو روش absolute یا relative بیان کرد. در روش اول مسیر از ریشه(/) شروع می‌شود. اما در روش دوم مسیر بنا بر جایی که هم اکنون هستید اعلام می‌شود.در روش دوم محل فعلی شما یعنی شاخه جاری در نظر گرفته شده می‌شود و سپس با توجه به آن مسیر مشخص می‌شود. برای مثال اگر هم اکنون در شاخه /*home*/oslab باشیم و در این شاخه فایلی با نام os1 باشد مسیر

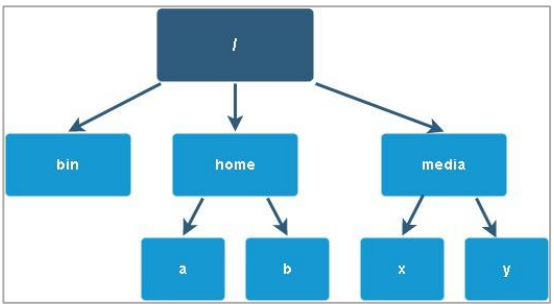
absolute این فایل */home/oslab/os1* است در حالی که مسیر relative به صورت os1 در نظر گرفته می‌شود. باید گفت در هر دایرکتوری دو مسیر .(نقطه) و ..(دو نقطه) به عنوان دایرکتوری جاری و دایرکتوری پدر وجود دارند که در مسیردهی به روش نسبی بسیار پرکاربرد هستند.

۳-۱-انتصاب (mount)

فرض کنید شما به سیستم خود یک فلش مموری یا هارد خارجی متصل کردید. برای دسترسی به این تجهیز لازم است جایی در درخت فایل سیستم لینوکس برای این مهم در نظر گرفته شود. اضافه‌کردن موقت یک آدرس خارجی به دایرکتوری یونیکس را mount می‌نامند. هر دایرکتوری خارج از فایل سیستم (همانند یک دایرکتوری از یک حافظه قابل حمل مانند فلش یا هارد دیسک) باید نسبت به / آدرس‌دهی شود در صورتی که شما یک هارد دیسک خارجی را به سیستم خود متصل کرده باشید خود هارد دیسک دارای یک درخت فایل سیستم است مثلا ممکن است چندین پوشه تودرتو و فایل‌های متعدد داشته باشد. این مسیرها باید در درخت فایل سیستم اصلی شما اضافه شود و از ریشه آدرس دهی انجام گیرد. عمل mount فایل سیستمی که روی یک دیوایس است را به درخت فایل سیستم لینوکس اضافه می‌کند. نقطه انتصاب یا mount point به جایی از فایل سیستم گفته می‌شود مقصد اتصال فایل سیستم خارجی به درخت اصلی است. برای مثال در *شکل۴* یک usb flash که با دو دایرکتوری x و y وجود دارد در صورت اتصال به سیستم باید به عنوان بخشی از فایل سیستم اصلی شناخته شوند. در *شکل ۵* دایرکتوری media به عنوان نقطه انتصاب قرار گرفته و دایرکتوری‌های x و y از مسیر / media/ قابل دسترس‌اند. به طور مثال مسیر پوشه x به صورت /media/x قابل دسترسی است. البته در نسخه‌های جدید توزیع‌های لینوکس، به محض اتصال دیوایس جدید، عمل mount به صورت اتوماتیک در شاخه /media انجام می‌شود. ولی گاهی که نوع فایل سیستم، خاص یا متفاوت است نیاز به mount کردن دستی است یا در یک basic unix این عمل خودکار انجام نمی‌شود. همچنین گاهی شاید ترجیح دهید دیوایسی را خارج یا اصطلاحا umount کنید و در موقع نیاز دوباره mountکنید.



شکل ۴- فایل سیستم قبل از mount کردن



شکل ۵- بعد از mount کردن

۴- آشنایی با پوسته‌های لینوکس و دستورات خط فرمان

در هر سیستم عامل برای کاربر پوسته‌ای تعبیه شده است تا بتواند با آن کار کند. این پوسته جایی که شما به فایل سیستم‌ها و ورودی و خروجی برنامه‌ها دسترسی دارید. حتما در محیط ویندوز با پوسته گرافیکی کار کرده‌اید. در لینوکس نیز برای کاربران پوسته‌های گرافیکی مختلفی ایجاد شده است تا کار با آن ساده باشد. پوسته‌های گرافیکی نیز مانند توزیع‌های مختلف لینوکس نسخه‌های مختلفی دارد و هر کدام طرفداران خاص خود را پیدا کرده است. یکی از مهمترین پوسته‌هایی که در لینوکس Ubuntu استفاده می‌شود GNOME نام دارد. پوسته‌های دیگری مانند KDE، Xfce یا Mate نیز معروف شده‌اند.

اما یک نوع پوسته‌ی دیگر در لینوکس بسیار کاربردی است. پوسته‌های مبتنی بر خط فرمان به دلیل آن که با دادن دستور از طریق کیبورد کار می‌کنند حرفه‌ای‌تر، ساده‌تر و قابل تنظیم تر هستند. تنها کافی است چند دستوری را حفظ کرده باشید.

خط فرمان سیستم‌های مبتنی بر یونیکس از جمله لینوکس، مهمترین اینترفیس سیستم محسوب می‌شود. در واقع مجموعه دستورهایی به صورت برنامه‌های مختلف در سیستم‌های لینوکس وجود دارد که از طریق یک CLI (Command Line Interface) امکان استفاده از آنها وجود دارد. در سیستم‌های لینوکس کلیدهای ctrl+Alt+Fn که Fn یکی از کلیدهای F است، یک CLI مجزا از محیط گرافیکی برای ما باز میکند (امتحانش کنید). برای بازگشت به محیط گرافیکی کافیست ctrl+Alt+F2 را بزنید (در بعضی توزیع‌های لینوکس کلید F دیگری ازجمله F7 ما را به محیط گرافیکی برمی‌گرداند).

از طرف دیگر در توزیع‌های لینوکس برنامه‌های CLI مختلفی نیز تعبیه شده است که اصطلاحا به آنها shell گفته می‌شود. این shell ها در محیط گرافیکی باز می‌شوند. برای مثال bash یک نمونه CLI در لینوکس است (کلیدهای ctrl+alt+T را بزنید).

شاید فکر کنید تایپ دستورات در یک صفحه تکستی سیاه یا سفید، کار خسته‌کننده و حوصله‌سربری باشد یا چرا وقتی یک محیط گرافیکی راحت و خوش‌دست داریم از ترمینال CLI استفاده کنیم؟

در مدیریت سرورها و سیستم‌ها، معمولا از طریق remote به سرور موردنظر متصل می‌شوند و تغییرات و تنظیمات لازم را روی آن اعمال می‌کنند یا اجرای سرویس‌ها را کنترل می‌کنند. در این حالت معمولا یک CLI به صورت remote در دسترس است و همه کارها باید از طریق آن انجام شوند. همچنین در صورتی که بخواهید با سیستم‌های embedded کار کنید معمولا یک سیستم عامل سبک روی چنین سیستم‌هایی نصب می‌کنند که گرافیک ندارد و نمی‌تواند یک پوسته‌ی گرافیکی را اجرا نماید. کنترل و اجرای برنامه و سرویس‌ها روی آن از طریق ارتباط با ابزارهایی مانند putty از طریق پورت‌های سیستم یا از طریق ارتباط remote از طریق پورت شبکه صورت می‌گیرد. در این حالت هم یک

CLI بیشتر در اختیار ندارید. به عنوان دلیلی دیگر در سیستم‌های سرور معمولا پوسته‌ی گرافیکی نصب نمی‌شود تا از رم و سخت‌افزار استفاده نکندو سبک‌تر باشند. همچنین حتی در سیستم‌هایی که محیط گرافیکی هم فراهم است حرفه‌ای‌کاران می‌دانند که کار با CLI سریعتر است و امکانات بیشتر و جذاب‌تری برای کنترل سیستم در اختیار آنها قرار می‌دهد. پس سعی کنید جذابیت‌های کار با shell لینوکس را کشف کرده و با آن دوست شوید:)

در ادامه مهمترین و جذاب‌ترین دستورها و برنامه‌های خط فرمان معرفی می‌شود. دقت داشته باشید که لینوکس در همه قسمت‌ها case sensitive است، بنابراین در ورود دستورات و اسامی فایل‌ها بزرگ و کوچک‌بودن حروف، تفاوت ایجاد می‌کند. در این گزارش، سعی شده دستورات پرکاربرد بیان شود، اما دستورات بسیاربسیارپرکاربرد هایلایت شده‌اند.

۴-۱- اجرای برنامه‌های اجرایی در سیستم فایل لینوکس

اگر بخواهید یک برنامه اجرایی را در shell لینوکس اجرا کنید، کافی است مسیر absolute فایل آن را در خط فرمان بنویسید و enter بزنید. فراموش نکنید که ./ مسیر absolute شاخه جاری را به ما می‌دهد، پس روش متداول اجرای برنامه‌ای با نام prg1 از شاخه حاوی این برنامه با اجرای prg1./ در خط فرمان صورت می‌گیرد.

وقتی یک برنامه را در خط فرمان اجرا می‌کنید، برنامه شروع به اجرا می‌کند و تا اتمام اجرا شما دیگر خط فرمان را نمی‌بینید و نمی‌توانید دستور دیگری اجرا کنید. این حالت اجرا برای وقتی که برنامه interactive است به کار می‌آید. اما گاهی نیاز دارید که برنامه‌ای را اجرا کنید و سپس به اجرای دستورات و برنامه‌های دیگر بپردازید. در این صورت اصطلاحاً می‌گوییم برنامه را باید در background یا پس‌زمینه اجرا کنیم. بدین‌منظور کافیست یک & در انتهای دستور اضافه کنید. بدین ترتیب برنامه مورد نظر اجرا شده و شما دوباره به خط فرمان برمی‌گردید تا دستورهای دیگری را اجرا کنید، هرچند برنامه قبلی تمام نشده و در سیستم در حال اجراست.

man

از این پس اطلاعات کامل چیستی و نحوه کار هر دستوری را می‌توانید با کمک دستور man مشاهده کنید. برای مثال بنویسید: man man با کلیدهای arrow یا page down و page up و یا اسکرول موس می‌توانید روی صفحه توضیحات جابه‌جا شوید. برای خروج از توضیحات man کلید q را فشار دهید. هر دستور دارای یک سری پارامتر ورودی و تعدادی آپشن است (معمولا با یک – شروع می‌شوند). توضیحات کامل تمام این موارد در manual دستور مربوطه شرح داده شده است. یکی از موارد مهم در دستور man فصل‌های این دستور

است. به طور مثال فصل ۱ در مورد دستورات مختلف shell است و فصل دوم در مورد systemcall هایی که برای زبان c ارائه شده است.

دستور man mkdir و man 2 mkdir را بررسی کنید.

دستورات فایل سیستم

این دسته از دستورات، جزو پراستفاده‌ترین دستورات هستند. جدول ۲، بعضی از این دستورات را نشان می‌دهد. سعی کنید همه این دستورات را با ورودی‌ها و آپشن‌های مختلف با کمک man دستورات امتحان کنید. دقت کنید که همه حالت‌ها و آپشن‌ها توضیح داده نشده و لازم است هر یک را با کمک man اجرا کنید تا با نحوه کار آن آشنا شوید.

جدول ۲: دستورات فایل سیستم

ls	مشاهده لیست همه محتویات یک مسیر یا شاخه جاری(ls -l و ls /home را امتحان کنید)
cd	تغییر شاخه به یک مسیر یا دایرکتوری جدید (می‌توانید به هر یک از دایرکتوریهای شاخه جاری که با ls مشاهده کرده‌اید وارد شوید)
cp	کپی یک فایل یا دایرکتوری در مسیر جدید (جهت کپی دایرکتوری از آپشن -R استفاده کنید)
touch	ایجاد یک فایل جدید و یا آپدیت زمان دسترسی به فایلی که قبلاً وجود داشته است (یک فایل جدید ایجاد کنید، ls -l بگیرید، زمان فایل را ببینید. دوباره دستور touch را برای آن فایل اجرا کنید و ls -l گرفته زمان را با زمان قبلی مقایسه کنید)
rm	حذف یک فایل یا دایرکتوری (برای حذف یک دایرکتوری با همه محتویات از آپشن -r استفاده کنید)
mkdir	ایجاد یک دایرکتوری جدید
mv	انتقال یک فایل یا دایرکتوری به محل جدید یا تغییر نام
pwd	مشاهده مسیر کامل شاخه فعلی
ln	ایجاد shortcut از یک فایل یا دایرکتوری در مسیر جدید
.	شاخه یا دایرکتوری فعلی
..	شاخه قبلی (دستور cd .. را امتحان کنید)
~	شاخه home کاربر فعلی (دستور cd ~ را امتحان کنید)

ابزار apt

جهت دانلود و نصب برنامه‌ها و ابزارهای لینوکس از روی اینترنت، ابزار بسیار پرکاربرد و خوبی با نام apt وجود دارد که در جدول ۳ بعضی از آپشن‌های استفاده از آن مشاهده می‌شود (ابزارهای دیگری غیر از apt نیز پدید آمده که آن‌ها نیز بسیار کارآمد هستند مانند snap و یا ابزار pip برای نصب بسته‌ها و کتابخانه‌های python و غیره). apt بسته‌ها را از سرورهای مختلفی دانلود می‌کند که با عنوان repository شناخته می‌شوند و آدرس آن‌ها در تنظیمات apt قرار می‌گیرد یا از طریق apt اضافه می‌شوند.

جدول ۳: ابزار apt

apt-get install pack_name	نصب بسته‌ای با نام pack_name
apt-get remove pack_name	Uninstallکردن بسته‌ای با نام pack_name
apt-cache search name	جستجوی نام دقیق یک بسته (گاهی نام دقیق بسته‌ای که می‌خواهید نصب کنید را نمی‌دانید ابتدا از این دستور استفاده کرده و پس از پیداکردن بسته مورد نظر با استفاده از apt-get آن را نصب یاuninstall کنید)
apt update	آپدیت لیست بسته‌های موجود در repository های تنظیم‌شده

مجموعه دستورات پرکاربردی در جداول بعدی آمده است.

جدول ۴: دستورات جستجو در فایل سیستم

find	جستجوی یک فایل. (فرم کلی find path -name pattern_or_name به جای path مسیر موردجستجو و به جای pattern_or_name نام فایل موردنظر که می‌تواند به صورت Regular experessionهم داده شود قرار می‌گیرد)
whereis	محل فایل باینری یک برنامه نصب‌شده در سیستم را نشان می‌دهد. (دستور man whereis را امتحان کنید)
which	محل فایل باینری برنامه نصب‌شده در سیستم را که در محیط فعلی اجرا می‌شود نشان می‌دهد. (خروجی دستور man which را بررسی ببینید)
locate	یک نام فایل را در کل سیستم جستجو می‌کند و همه مطابقت‌ها را در خط‌های جداگانه نشان می‌دهد . (man locate را امتحان کنید)
wc	شمارش تعداد حروف یا خطوط فایل(دستور man wc را ببینید همچنین آپشن l- را بررسی کنید)
grep	یک عبارت یا RE را در یک متن یا فایل حاوی متن جستجو می‌کند. grep "man" ~/.bash_history را ببینید.

	دستور فوق را با آپشن c- دوباره امتحان کنید

جدول ۵: دستورات سیستم

shutdown	خاموش کردن سیستم
halt	خاموش کردن سیستم
reboot	ریبوت سیستم
sudo	اجرای دستورات با کاربر root ، پس از اجرای این دستور پسورد root سؤال می‌شود (کاربر root ادمین سیستم لینوکس است و در حالت معمول، کاربر جاری سیستم، rootنیست. جهت protection اجرای بعضی دستورات لینوکس فقط توسط root امکان‌پذیر است.)
su	تغییر کاربر سیستم (استفاده این دستور بدون دادن نام کاربر، کاربر را به root تغییر می‌دهد)
addusr	اضافه‌کردن کاربر جدید به سیستم (این دستور فقط از طریق root قابل اجراست)
passwd	تغییر پسورد یک کاربر
whoami	نشان‌دادن نام کاربری کاربر جاری

جدول ۶: دستورات shell

exit	خروج از shell جاری
clear	پاک‌کردن همه نوشته‌های ترمینال جاری (کلید ctrl+l را هم امتحان کنید)
|	یک دنباله از دستورات که توسط علامت | از یکدیگر جدا شده‌اند به صورت موازی قابل اجرا هستند که این حالت را به این صورت است که خروجی دستور سمت چپ | به pipeline کردن دستورات گویند. عمل‌کرد pipeline کردن بیش از یک دستور نیز وجود دارد که در pipeline عنوان ورودی دستور سمت راست | استفاده می‌شود. امکان این صورت اجرا از سمت چپ به صورت موازی شروع می‌شود (بسیار پرکاربرد)
>	با این علامت می‌توان ورودی یک برنامه را از محلی غیر از ورودی استاندارد گرفت برای مثال از یک فایل
<	با این علامت خروجی یک برنامه را می‌توان در محلی غیر از خروجی استاندارد ذخیره کرد

جدول ۷: دستورات کار با فایل‌ها

cat	نشان‌دادن محتوای کامل یک فایل در خط فرمان و برگشت به خط فرمان

less	مشاهده محتوای یک فایل به صورت صفحه به صفحه (cat امکان اسکرول کردن ندارد و به صورت یک‌دفعه‌ای تا انتهای فایل را نشان می‌دهد)
more	مشابه less ولی فقط امکان اسکرول به سمت پایین را دارد (more ~./bash_history و ls -a | more را امتحان کنید)
tail	نمایش محتوای انتهای یک فایل (tail -10 ~/.bash_history را امتحان کنید همچنین دستور فوق را با آپشن f- بررسی کنید)
head	نمایش محتوای ابتدای یک فایل (head -10 ~/.bash_history را امتحان کنید)
tar	بازکردن یک فایل آرشیو
zip	بازکردن یک فایل فشرده

جدول۸: دستورات پروسس‌ها

ps	نمایش لیست پروسس‌های در حال اجرا (این دستور را با آپشن‌های مختلف از جمله بدون آپشن و با a- امتحان کرده تفاوت آن‌ها را پیدا کنید)
top	نمایش آنلاین لیست پروسس‌های در حال اجرا در سیستم همراه با اطلاعات نحوه مصرف منابع سیستم (حتماً امتحانش کنید)
kill	ارسال یک سیگنال به پروسسی در حال اجرا در سیستم (اجرای این دستور معمولاً جهت بستن یک پروسس به کار می‌رود زیرا پیش‌فرض این دستوریعنی استفاده از kill بدون ذکر شماره سیگنال، سیگنال ۹ که مربوط به از بین‌بردن یک پروسس است را ارسال می‌کند)
killall	این دستور جهت بستن همه پروسس‌های با یک نام بسیار مفید است. برای مثال وقتی chrom با تبهای زیاد باز است killall chrome همه پروسسهای chrome یعنی همه تبها را میبندد.

جدول ۹: دستورات شبکه

ifconfig	نمایش اطلاعات و آدرس‌های کارت شبکه‌های سیستم (آن را اجرا کنید، معادل آن در ویندوز چیست؟)
ping	Pingکردن یک آدرس در شبکه (معمولاً جهت کشف مشکلات مربوط به عدم دسترسی به یک آدرس مفید است)
traceroute	نمایش تمامhop های مسیر تا رسیدن به یک آدرس مشخص (دستور traceroute iut.ac.ir را امتحان کنید)
wget	دانلود محتوا از یک آدرس وب

iptables	ابزاری برای کنترل ورود و خروج بسته ها (فیلترینگ)
ssh	اتصال امن به یک کامپیوتر دیگر در شبکه
scp	کپی یک فایل یا دایرکتوری به کامپیوتر ریموتی در شبکه یا از روی کامپیوتری در شبکه

جدول ۱۰: دستورات دیسک

fdisk	نمایش و مدیریت و تغییر فضاهای حافظه ثانویه سیستم و اطلاعات آن‌ها(fdisk -l را امتحان کنید)
lsblk	نمایش دیوایس‌های بلاکی سیستم
mount	انتصاب فایل سیستم خارجی به فایل سیستمroot
dd	این دستور جهت کپی کامل یک فایل image یا دیسک مفید است (همچنین ساخت فلشbootable)
df	اطلاعاتی راجع به میزان پر یا خالی بودن فایل سیستم روی دیوایس‌های متصل به سیستم را نشان می‌دهد.

۵- مدیریت و کار با فایل‌ها در لینوکس

همانطور که قبلا گفته شد در لینوکس هر موجودیتی تحت عنوان یک فایل شناخته می‌شود. از طرف دیگر از هر سیستم تعدادی کاربر استفاده می‌کنند که هر یک از آنها متعلق به یک یا چند گروه تعریف‌شده در سیستم هستند. هر فایل در سیستم متعلق به یک کاربر و یک گروه است. مالک و گروه هر فایل در هنگام ایجاد آن تعیین می‌شود. به طور پیش‌فرض مالک هر فایل ایجادکننده آن و گروه هر فایل همان گروهی است که مالک فایل در لحظه ایجاد فایل به آن تعلق دارد. می‌توان پس از ایجاد فایل، مالک و گروه آن را تغییر داد. برای هر فایل در یونیکس برای سه گروه، سطح دسترسی تعریف شده است: مالک فایل (owner)، گروه فایل (group) و سایر افراد (others). برای هر یک از سه حالت فوق سه سطح دسترسی در نظر گرفته شده است: خواندن (read)، نوشتن (write) و اجراکردن (execute). دقت داشته باشید که برای دایرکتوری‌ها همین موارد وجود دارد و خواندن به معنای مشاهده لیست فایل‌های داخل آن است ولی برای دسترسی به درون دایرکتوری باید گزینه اجرا نیز فعال باشد. با اجرای دستور ls -l می‌توان سطح دسترسی هر فایل یا دایرکتوری را مشاهده کرد که در یک رشته ۱۰ کاراکتری قرار دارد: rwx rwx rwx - . کاراکتر اول نوع فایل را مشخص می‌کند که در جدول ۱۱ انوع آن آمده است.

کاربری با نام root در همه سیستم‌های لینوکس تعریف شده است که دسترسی کامل به سیستم دارد و درواقع ادمین سیستم محسوب می‌شود. بسیاری دستورهای سیستمی فقط به root اجازه اجرا یا نوشتن را می‌دهد. معمولاً توزیع‌های لینوکس به صورت پیش‌فرض با کاربر root لاگین نمی‌شوند. همانطور که قبلاً بیان شد برای اجرای هردستور با دسترسی root کافیست در خط فرمان آن دستور را با sudo اجرا کنیم (اضافه

کردن sudo در ابتدای دستور). همچنین اگر بخواهیم خط فرمان به طور کلی در اختیار کاربر root قرار گیرد، در بعضی توزیع‌ها دستور su بدون وارد کردن نام کاربر، خط فرمان را در دسترس root قرار می‌دهد، در بعضی توزیع‌ها نیز sudo -i این کار را می‌کند. در همه این حالت‌ها پسورد root سؤال می‌شود.

جدول ۱۱: انواع فایل

Regular	-
Directory	d
Socket	s
named pipe	p
symbolic link	l
block device	b
char device	c

از آن پس هر دسته ۳تایی کاراکترها به ترتیب سطح دسترسی مالک، گروه و سایر افراد را مشخص می‌کند. برای هر یک از این سطح دسترسی‌ها یک مقدار octal در نظر گرفته شده‌است: execute=1, write=2, read=4.

در هر حالت اگر دسترسی وجود داشته باشد عدد آن را لحاظ می‌کنیم و اگر دسترسی وجود نداشته باشد، مقدار معادل آن را ۰ در نظر می‌گیریم.

برای محاسبه عدد نهایی سطح دسترسی این ۳ مقدار با یکدیگر جمع زده می‌شوند.

جدول ۱۲: سطح دسترسی‌های فایل‌ها

سطح دسترسی خواندن و نوشتن و اجرا	4+2+1 = 7
سطح دسترسی خواندن و نوشتن	4+2+0 = 6
سطح دسترسی خواندن و اجرا	4+0+1 = 5

در جدول۱۲ دستورات تغییر سطح دسترسی ها را مشاهده میکنید.

جدول ۱۳: دستورات سطح دسترسی فایل‌ها

chmod	تغییر سطح دسترسی فایل (یک فایل با دستور touch ایجاد کرده و سپس سطح دسترسی آن را با ls -l ببینید و سپس با دستور chmod 755 new.txt سطح دسترسی آن را به خواندن و نوشتن و اجرا برای مالک و خواندن و نوشتن برای بقیه تبدیل کنید) روش دیگر غیر از روش عددی مد فایل، استفاده از chmod با استفاده از معادل الفبایی سطح دسترسی است. برای مثال جهت اضافه کردن امکان اجرایی به فایل new.sh از این دستور استفاده می‌شود chmod +x new.sh
chown	تغییر مالک فایل
chgrp	تغییر گروه فایل

۵-۱- ویرایشگرهای لینوکس

ویرایشگرهای مختلف گرافیکی (مانند gedit، atom) و غیرگرافیکی (مانند vi، vim) برای کار با فایل‌ها در لینوکس ارائه شده‌است. در این جلسه، با ابزار vim که از معروف‌ترین ویرایشگرهای مورداستفاده است آشنا می‌شوید. فکر نکنید این ویرایشگر قدیمی شده است و در قرن ۲۱ به بعد نیازی به آن نیست:)) به همان دلایلی که در بخش توضیحات CLI مطرح شد هنوز هم این ویرایشگرها استفاده جدی دارند. پس بد نیست چند جلسه‌ای از این نوع ویرایشگرها استفاده کنید ;)

طی سالیان متمادی vi به عنوان ویرایشگر پیش‌فرض همراه با همه سیستم عامل‌های مبتنی بر یونیکس ارائه شده‌است. این ویرایشگر در عین سادگی، قابلیت پیکربندی و انعطاف آن به قدری بالاست که از محبوبترین ویرایشگرهای جهان به شمار می‌آید. نسخه‌های مختلفی از این ویرایشگر از جمله vim وجود دارد که در این آزمایشگاه از آن استفاده می‌کنید.

معمولا vi به صورت پیش‌فرض روی توزیع‌های لینوکس نصب شده‌است. برای نصب vim یا vi improved کافیست از apt-get استفاده کنید: apt-get install vim

فایلی با نام vimrc. وجود دارد که معمولا در home یا در شاخه etc قرار دارد. از طریق این فایل می‌توان vim را با گزینه‌های مختلفی پیکربندی کرد. گاهی این فایل به صورت پیش‌فرض با نصب vim ساخته نمی‌شود و کاربر می‌تواند خودش آن را ایجاد کند (در حالت معمول کاری با این فایل ندارید)

برای کار با vim یا نیاز دارید فایلی که از قبل وجود دارد را باز کرده ویرایش کنید یا فایل جدیدی ایجاد کرده و کار کنید. اگر vim را با نام یک فایل (درواقع مسیر آن فایل) اجرا کنید، در صورت وجود باز می‌شود و در غیر این صورت ابتدا ساخته شده و سپس باز می‌شود.

:	در این حالت vim منتظر دستوری برای ایجاد تغییر می‌شود
help:	نمایش راهنما
:w	ذخیره‌سازی تغییرات اعمال‌شده
:q	خروج از vim در صورتی که هیچ تغییری وارد نشده باشد

:q!	خروج از vim بدون ذخیره‌سازی تغییرات اعمال‌شده
:wq	ذخیره تغییرات و خروج ازvim
/	جستجوی یک کلمه یا عبارت در فایل
s%	جایگزین کردن یک کلمه با کلمه جدید (با این دستورold_word ها را با new_word ها جایگزین کنید : :s/old_word/new_word%)
d	پاک‌کردن یک خط
shift+v	انتخاب یک خط کامل
v	رفتن به وضعیتvisual mode ، در این حالت کلمات در فاصله‌ای که اشاره‌گر اکنون قرار دارد تا هرکجا که قرار بگیرد انتخاب می‌شود.
u	مشابه عمل undo در ویرایشگرهای دیگر
5u	خنثی کردن آخرین ۵ عمل
ctrl+r	مشابهredo
d	انتقال کلمات انتخاب‌شده به حافظه و پاک‌کردن آنها
y	کپی کلمات انتخاب‌شده به حافظه
8y	کپی کلمات از جایی که اشاره گر قرار دارد تا انتهای خط جاری و همچنین ۸ خط بعدی
p	کلمات منتقل‌شده به حافظه را در محل اشاره گر درج می‌کند
3p	کلمات منتقل‌شده به حافظه را سه بار در محل اشاره گر درج می‌کند
gg	انتقال اشاره‌گر به خط اول فایل
G	انتقال اشاره‌گر به خط آخر فایل
:11	انتقال به خط ۱۱

پس از این‌که فایلی را باز کردید، محتویات آن را در همان صفحه CLI مشاهده می‌کنید. وقتی فایلی باز است ممکن است در دو وضعیت قرار داشته باشید: command mode یا insert mode. برای قرارگرفتن در حالت insert باید کلید insert را فشار دهید و برای خروج از این حالت و ورود به command می‌توانید از کلید esc استفاده کنید. همچنین وقتی در وضعیت command هستید می‌توانید در فایل جابه جا شوید، مقداری را جستجو کنید، تغییرات فایل را ذخیره کنید و سایر موارد دستوری را اعمال کنید. در حالت insert می‌توانید مقادیر نوشته‌شده در فایل را تغییر دهید. در جدول۱۴ بعضی موارد قابل استفاده در وضعیت command را مشاهده می‌کنید.

همچنین دستوراتی مانند :set nu و :set nonu و :syntax on را نیز امتحان کنید.