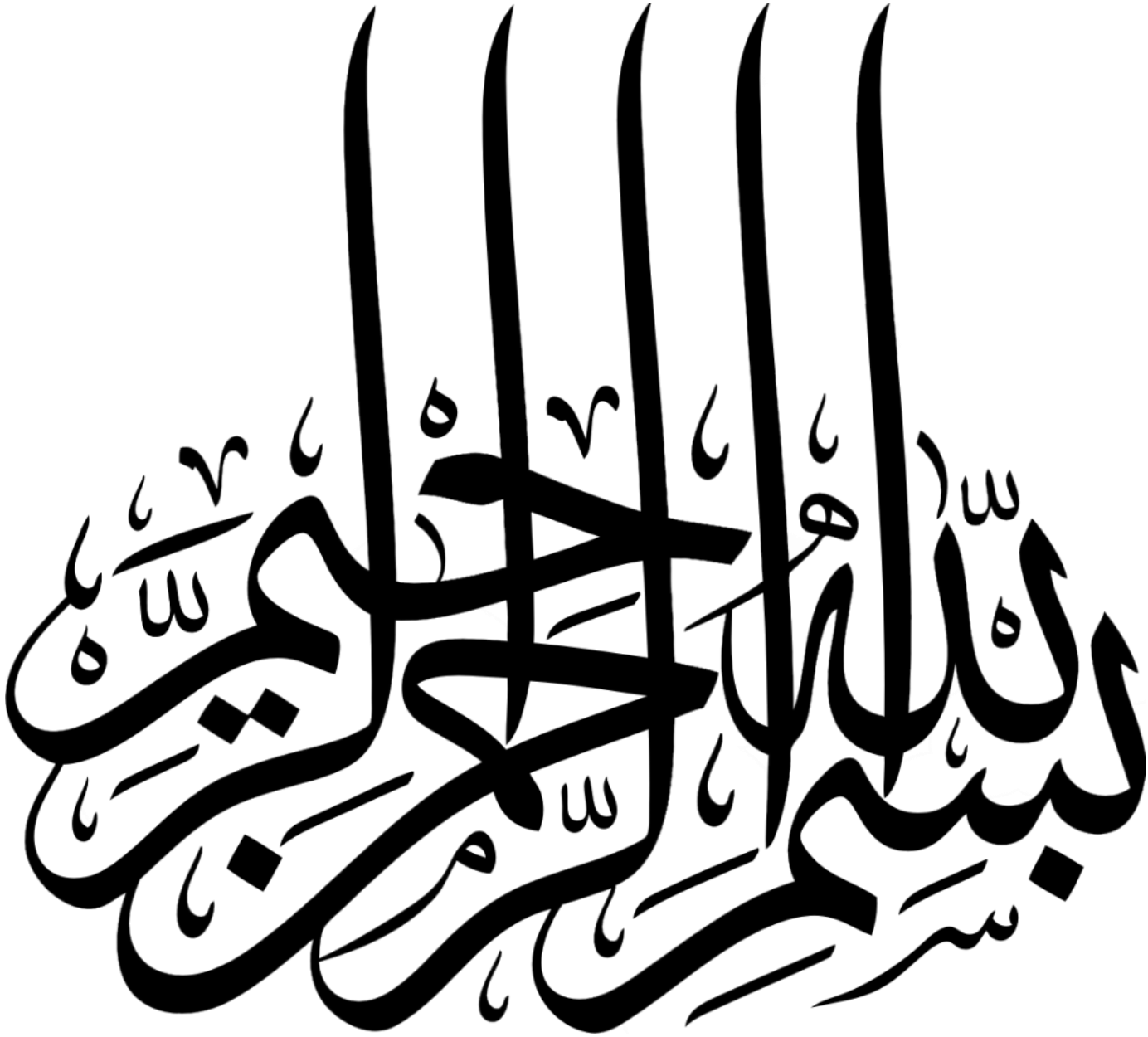

بسم الله الرحمن الرحيم

کتاب الکترونیکی کوتاه و کاربردی از لینوکس برای تست نفوذگران

تهیه شده توسط

احسان نیک آور

فهرست مطالب

مقدمه

با عرض سلام و ادب خدمت کلیه دوستان عزیز، این کتاب به عنوان یک راهنمای کوچک ولی کاربردی برای آشنایی هر چه بهتر و استفاده مناسب از سیستم‌عامل لینوکس برای شما تهیه شده است.

اغلب بخش‌های این کتاب برگرفته از کتاب Linux Basics for Hackers می‌باشد. لازم به ذکر است که کتاب جاری، ترجمه کامل کتاب Linux Basics for Hackers نبوده و برخی از بخش‌های آن حذف گردیده است.

این کتاب در شانزده فصل تهیه شده است که در هر بخش، مطالب کاربردی جهت استفاده هر چه بهتر از سیستم‌عامل لینوکس در آن مطرح شده است.

پس پیشنهاد می‌کنیم تا انتهای این کتاب با ما همراه باشید.

هر کتاب و مطلبی قطعا کامل نیست و دارای نواقصی می‌باشد. به همین منظور هر گونه انتقاد، پیشنهاد و نظرات شما دوستان گرامی که این مطلب را مطالعه می‌نمایند، راه گشای ما برای تهیه کتاب ها و مطالب بعدی خواهدبود.

لذا از کلیه شما دوستان گرامی خواهشمند هستیم تا نظرات خود را به ایمیل info@securityworld.ir با موضوع Linux Basic ارسال نمایید.

بسیار ممنون می‌شویم که اگر این کتاب را مفید دانستید، آن را با دوستان خود به اشتراک بگذارید تا از این کتاب استفاده نمایند.

همچنین فصل‌های این کتاب به صورت جداگانه در وب سایت securityworld.ir قرار گرفته است. لازم به ذکر است که وب سایت securityworld.ir دارای محتواهای بسیار زیادی در خصوص تست نفوذ و امنیت اطلاعات می‌باشد که پیشنهاد می‌کنیم از این وب سایت نیز بازدید به عمل آوردید.

با سپاس فراوان

احسان نیک آور

فصل اول

آشنایی با اصطلاحات و دایرکتوری‌های لینوکس

در ابتدا ما قصد داریم تا شما را برخی از اصطلاحات رایج در لینوکس و همچنین دایرکتوری‌های مهم در آن آشنا نماییم.

Binaries: این اصطلاح مربوط به فایل‌هایی است که مشابه سیستم‌عامل ویندوز قابلیت اجرا دارند. این فایل‌ها به صورت کلی در مسیر /usr/bin یا /usr/sbin قرار دارند و شامل برنامه‌های کاربردی مانندps ، cat، lsو موارد مشابه هستند. همچنین برنامه‌های کاربردی دیگری مانند ابزارهای تست نفوذ وایرلس و یا سیستم تشخیص نفوذ Snort و موارد دیگر نیز دارای فایل‌های باینری هستند که در این بخش قرار دارند.

Case sensitivity: بر خلاف سیستم‌عامل ویندوز، لینوکس به حروف کوچک و بزرگ حساس است و این بدان معنی است که Desktop با desktop در سیستم‌عامل لینوکس متفاوت هستند.

Directory: این عبارت مشابه مفهوم فولدر در ویندوز است. یک دایرکتوری روشی برای ساماندهی فایل‌ها بوده و به صورت سلسله مراتبی می‌باشد.

Home: هر کاربر در لینوکس دارای دایرکتوری home مخصوص به خود می‌باشد و جایی است که فایل‌های شما به صورت پیش فرض در آن ذخیره می‌گردد.

Kali: کالی لینوکس یک توزیع از لینوکس است که به صورت خاص برای انجام فرآیندهای تست نفوذ طراحی شده است. این سیستم‌عامل دارای صدها ابزار مختلف در حوزه تست نفوذ است که نیاز تست نفوذگران را برای نصب ابزارهای مختلف مرتفع می‌سازد. همچنین شما می‌توانید این سیستم‌عامل و ابزارهای داخل آن را در بازه‌های زمانی مختلف بروزرسانی نمایید.

Root: همانند هر سیستم‌عاملی، لینوکس نیز دارای یک حساب کاربری با دسترسی مدیر می‌باشد که برای استفاده توسط افراد مطمئن در نظر گرفته شده است تا مدیریت این سیستم‌عامل را انجام دهد. قابلیت تنظیمات سیستمی، اضافه نمودن کاربران، تغییر کلمات عبور و موارد مشابه از وظایف کاربری با نام root می‌باشد. همچنین اکثر ابزارهای تست نفوذ، نیازمند استفاده از کاربر root می‌باشند.

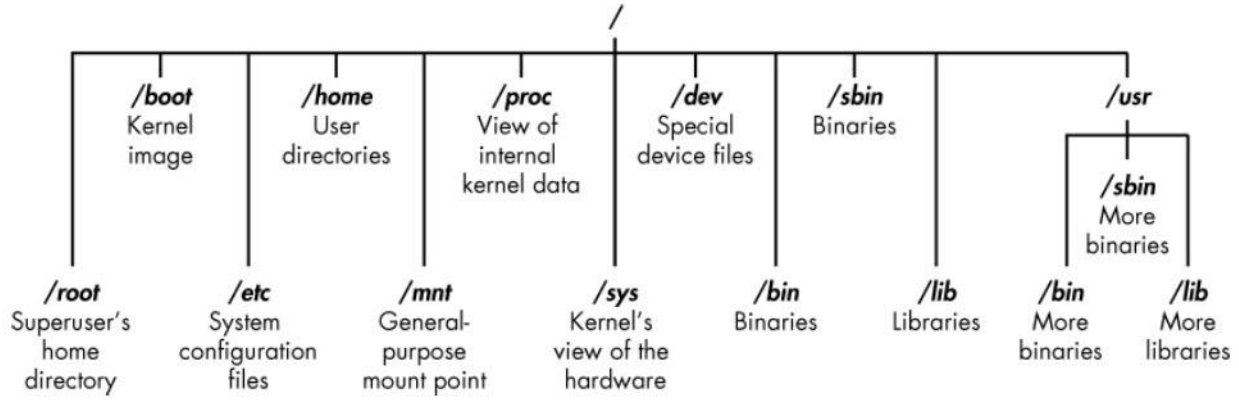
Script: اسکریپت در واقع مجموعه‌ای از دستورات است که در یک محیط مفسری (interpretive) اجرا شده که هر خط را به یک کد منبع تبدیل می‌کند. بسیاری از ابزارهای تست نفوذ در واقع اسکریپت‌های ساده هستند. این اسکریپت ها می‌توانند با زبان‌های مختلف مانند پایتون، پرل، روبی یا حتی Bash سیستم‌عامل لینوکس نوشته شده باشند.

Shell: شل یک محیط و مفسر برای اجرای دستورات در لینوکس است. یک شل که به صورت گسترده‌ای در لینوکس مورد استفاده قرار می‌گیرد Bash بوده که مخفف Bourne-Again Shell می‌باشد. البته شل‌های دیگری مانند C Shell، Z Shellو موارد دیگر نیز وجود دارد ولی ما در این کتاب از bash استفاده می‌نماییم.

Terminal: ترمینال یک واسط کاربری دستوری یا Command Line Interface است. شما با کلیک بر روی آیکون موجود در کالی لینوکس به ترمینال آن دسترسی خواهید داشت.

فایل سیستم در لینوکس

ساختار سیستم فایل در لینوکس با ویندوز متفاوت است. لینوکس دارای درایو فیزیکی مانند درایو C یا درایوهای مشابه که در ویندوز استفاده می‌شود نبوده و به جای آن از سیستم فایل منطقی یا logical filesystem استفاده می‌کند. بالاترین بخش از ساختار سیستم فایل در لینوکس "/" می‌باشد و اغلب به سیستم فایل root اشاره دارد. دایرکتوری‌های دیگری نیز در زیرمجموعه "/" وجود دارد که تصویر زیر این دایرکتوری ها را نمایش می‌دهد:



در ادامه به برخی از دایرکتوری‌های مهم در لینوکس می‌پردازیم:

/root: دایرکتوری Home مربوط به کاربر root است.

/etc: این دایرکتوری شامل فایل‌های تنظیمات در لینوکس است. این فایل‌ها برای پیکربندی سرویس‌ها و برنامه‌ها مورد استفاده قرار می‌گیرند.

/home: دایرکتوری Home مربوط به کاربران است.

/mnt: در این بخش سیستم فایل‌های دیگری که متصل می‌شوند و یا Mount می‌شوند قرار می‌گیرند.

/media: در این بخش دستگاه‌های USB و CD که به سیستم فایل متصل یا Mount می‌شوند، قرار می‌گیرند.

/bin: در این بخش برنامه‌های باینری که قابل اجرا هستند قرار دارند که در بخش پیشین به آن اشاره گردید.

/lib: در این بخش فایل‌های کتابخانه‌ای قرار دارد. این فایل‌ها مشابه فایل‌های DLL در ویندوز می‌باشد.

فصل دوم

دستورات کاربردی لینوکس

یکی از مواردی که باید با آن آشنا باشید، دستورات کاربردی لینوکس است. به همین دلیل در ادامه به معرفی این دستورات می‌پردازیم.

Pwd: این دستور دایرکتوری که در حال حاضر در آن قرار دارید را به شما نمایش می‌دهد. پس از ورود به لینوکس با توجه به اینکه شما با چه کاربری وارد شده باشید، به صورت پیش فرض در دایرکتوری آن کاربر خواهید بود.

Whoami: این دستور به شما نشان می‌دهد که با چه کاربری وارد سیستم شده اید.

Cd: این دستور برای تغییر دایرکتوری یا Change Directory مورد استفاده قرار می‌گیرد و پس از آن باید نام دایرکتوری که قصد وارد شدن به آن دارید را قرار دهید.

همچنین با استفاده از دو نقطه می‌توانید وارد دایرکتوری قبل از دایرکتوری موجود شوید. با توجه به این موضوع اگر قصد رفتن به دو دایرکتوری بالاتر را داشته باشید عبارت دو نقطه را دو بار نوشته و اگر قصد رفتن به سه دایرکتوری بالاتر را دارید سه بار عبارت دو نقطه را تایپ می‌کنید.

Ls: از این دستور برای لیست نمودن فایل‌ها و دایرکتوری‌های موجود در دایرکتوری حاضر و یا دایرکتوری دیگر استفاده می‌شود. برای مشاهده جزئیات بیشتر در مورد محتویات دایرکتوری و همچنین دایرکتوری‌ها و فایل‌های مخفی شده می‌توانید از دستور ls -la استفاده کنید.

راهنمای دستورات : برای مشاهده راهنمای دستورات یا ابزارها در لینوکس می‌توانید از دو عبارت h- و یا help– در انتهای دستور استفاده نمایید. توجه داشته باشید که در لینوکس اگر از یک کلمه به عنوان سوییچ یک دستور استفاده می‌کنید نیاز به دو - بوده و در صورتی که از یک حرف استفاده نمایید نیاز به استفاده از یک - می‌باشد.

مشاهده Manual Page دستورات :علاوه بر سوییچ help برخی از دستورات و برنامه‌ها یک صفحه راهنمای یا Manualدارند که اطلاعات بیشتری را در اختیار شما قرار می‌دهند. برای مشاهده این صفحات کافی است تا درابتدای دستور یا ابزار مورد نظر خود عبارت man را تایپ نمایید. پس از مشاهده این صفحه با فشردن کلید q می‌توانید از بخش Manualخارج شوید.

Locate: یکی از دستورات مربوط به جست و جو در لینوکس است که در ادامه آن عبارتی که قصد جست و جوی آن را دارید قرار می‌گیرد. این دستور کل سیستم فایل را برای پیدا نمودن عبارت مورد نظر، جست و جو نموده و نتیجه آن را به شما نمایش می‌دهد.

نکته : این دستور کامل نبوده و در برخی موارد نتایج جست و جوی آن قابل اطمینان نمی‌باشد. دستور locate از یک دیتابیس استفاده می‌کند که معمولا به صورت روزانه بروزرسانی می‌شود و اگر شما یک فایل را طی چند دقیقه یا چند ساعت گذشته ایجاد کرده باشید، ممکن است توسط این دستور شناسایی نشود.

Whereis: اگر شما به دنبال یک فایل باینتری می‌گردید، می‌توانید از دستور whreris برای این منظور استفاده نمایید. این دستور نه تنها مسیر باینری آن را به شما نمایش می‌دهد بلکه منبع اصلی و صفحه Manual آن را در صورت وجود برای شما نمایش خواهد داد.

در این مورد دستور whereis به جای بازگرداندن هر مسیری که عبارت مورد نظر در آن قرار دارد، فقط مسیر باینری و صفحه Manual آن را باز می‌گرداند.

Which: این دستور تنها مسیر باینری‌های موجود در متغیر PATH در لینوکس را نمایش می‌دهد. البته در بخش‌های بعدی به متغیر PATH اشاره خواهیم کرد. اما در حال حاضر همین مقدار بدانید کافی است کهPATH ، دایرکتوری‌های مربوط به دستورات وارد شده توسط شما در خط فرمان را نگه می‌دارد تا بتواند آن‌ها را اجرا نماید. به عنوان مثال هنگامی که شما دستور aircrack-ng را در خط فرمان وارد می‌کنید، سیستم‌عامل به متغیر PATH نگاه کرده تا دایرکتوری مربوط به aircrack-ng را پیدا نماید.

Find: از این دستور هم برای جست و جو در سیستم‌عامل لینوکس مورد استفاده قرار می‌گیرد. با استفاده از این دستور شما می‌توانید در هر دایرکتوری جست و جو نموده و یا جست و جو را بر اساس پارامترهای مختلف شامل نام فایل، تاریخ ایجاد و تغییر، نام مالک، گروه، مجوزها و یا سایز، انجام دهید. ساختار استفاده از دستور find به صورت زیر است:

```
find directory options expression
```

بنابراین اگر بخواهیم یک فایل را که نام آن apache2 باشد جست و جو نموده و این جست و جو را از دایرکتوری root آغاز کند، دستور ما به صورت زیر خواهد بود:

```
find / -type f -name apache2
```

نکته : دستور find تنها مواردی که دقیقاً مشابه با نام جست و جو شده، باشد را نمایش می‌دهد به عنوان مثال اگر شما عبارت apache2 را جست و جو کرده باشید، اگر این فایل دارای یک پسوند باشد مانندapache2.conf ، دستورfind آن را پیدا نخواهد کرد. البته ما می‌توانیم این محدودیت را با استفاده از Wildcard ها حل کنیم. در مثال زیر فایل‌هایی که با apache2 شروع شده است را با هر پسوندی جست و جو خواهد نمود:

```
find /etc -type f –name apache2.*
```

کوتاه در مورد wildcard ها

برای مثال ما یک جست و جو را در دایرکتوری که فایل‌هایcat ، hat، whatو bat در آن قرار دارند انجام می‌دهیم . Wildcardها می‌توانند شامل [] and ? , . * باشند.

عبارت ? برای یک کاراکتر استفاده می‌شود. بنابراین نتایج جست و جو با عبارت at? فایل های hat ، catو bat بوده و شامل what نمی‌شود زیرا فایل what دارای دو کاراکتر پیش از at است و شامل نتایج این جست و جو نمی‌شود.

عبارت [] برای جست و جوی حروف یا کلماتی است که در داخل آن استفاده می‌شوند.

به عنوان مثال اگر جست و جو برای عبارت at[c,b] انجام شود، نتایج فایل‌های cat و bat بوده و فایل‌های hat و what نمایش داده نمی‌شوند.

Wildcard دیگر * است که کاربرد بیشتری دارد و تمام کاراکترهایی با هر اندازه‌ای را شامل می‌شود بدین صورت که اگر شما در جست و جو عبارت at* را وارد نمایید، نتایج آن cat ، bat، hatو what خواهد بود.

کاربرد wildcard در عبور از فایروال ها

یکی از مواردی که می‌توان از wildcard استفاده نموده، عبور از امضاهای فایروال‌ها و سیستم‌های تشخیص نفوذ است. یکی از دستوراتی که در حملات Command Injection کاربرد دارد، دستور cat /etc/passwd می‌باشد. برخی از فایروال‌ها این عبارت را به عنوان دستور مخرب شناسایی نموده و اجازه اجرای آن را نمی‌دهند. با توجه به مواردی که در خصوص wildcard خدمت شما مطرح نمودیم، شما می‌توانید به جای استفاده از عبارت اصلی، از دستور cat /etc/pass??استفاده نمایید که این دستور می‌تواند منجر به عبور از امضای فایروال یا سیستم تشخیص نفوذ شود.

```
root@kali:~# cat /e?c/pass?? | head -5
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
```

Grep: اکثر اوقات شما از دستورات مختلف استفاده می‌کنید ولی شما تنها به دنبال کلمه با عبارت خاصی هستید. برای این منظور استفاده از grep به شما پیشنهاد می‌گردد. به عنوان مثال دستور ps با سوییچ aux اطلاعات مربوط به کلیه پروسس‌های در حال اجرا را به شما نمایش می‌دهد. برای اینکه بخواهید تنها اطلاعات مربوط به یک پروسس خاص را مشاهده نمایید می‌توانید از grep به صورت زیر برای این منظور استفاده نمایید.

ps aux | grep apache2

در واقع این دستور به لینوکس می‌گوید که کلیه پروسس‌ها را نمایش داده و سپس خروجی آن را به grep ارسال کند و در ادامه تنها بخش‌هایی که عبارت apache2 در آن وجود دارد را نمایش بدهد.

ایجاد و تعییرات در فایل و دایرکتوری

در ادامه معرفی دستورات کاربردی لینوکس به نحوه ایجاد و تغییرات در فایل و دایرکتوری می‌پردازیم. برای ایجاد فایل راه‌های مختلفی وجود دارد که در این بخش به برخی از آن‌ها اشاره می‌کنیم.

Cat: از این دستور معمولا برای مشاهده محتویات فایل‌ها استفاده می‌شود ولی شما می‌توانید به وسیله آن یک فایل نیز ایجاد کنید. البته پیشنهاد می‌شود از این دستور برای مشاهده فایل‌های کوچک استفاده کنید و برای مشاهده فایل‌های حجیم از ابزارهایی مانندvim ، leafpad، geditو ابزارهای مشابه استفاده نمایید.
برای ایجاد فایل با استفاده از دستور cat به صورت زیر عمل می‌کنیم:

```
cat > securityworld
```

پس از تایپ دستور بالا و فشردن کلیدEnter ، لینوکس وارد مد interactive شده و منتظر می‌ماند تا شما محتوای دلخواه خود را برای فایل securityworld تایپ نمایید. پس از تایپ محتوا، با فشردن کلید CTRL و D شما از این محیط خارج خواهید شد. اگر بخواهید که محتوایی را به انتهای فایل بالا اضافه نمایید به جای استفاده از >باید از >> استفاده نمایید.

Touch: از این دستور نیز می‌توان برای ایجاد فایل استفاده نمود.
Mkdir: از این دستور برای ایجاد یک دایرکتوری استفاده می‌شود.
Cp: از این دستور برای کپی فایل‌ها استفاده می‌شود که نحوه استفاده از آن به صورت زیر است:

```
cp securityworld /root/test
```


mv: متاسفانه لینوکس دستور مجزایی برای تغییر نام فایل‌ها نداشته و از دستور mv برای این منظور استفاده می‌کند. از دستور mv به صورت پیش فرض برای move نمودن فایل‌ها استفاده می‌شود.
Rm: از این دستور برای حذف یک فایل استفاده می‌شود.
Rmdir: از این دستور برای حذف یک دایرکتوری استفاده می‌شود. توجه داشته باشید این دستور، دایرکتوری که خالی نباشد را حذف نمی‌کند و برای حذف دایرکتوری در هر شرایطی باید از سوییچ r- استفاده نمایید.

فصل سوم

ویرایش متن در لینوکس

در لینوکس تقریبا همه مواردی که مستقیماً با آن‌ها سر و کار دارید، یک فایل بوده و اغلب به صورت متنی می‌باشند. به عنوان مثال کلیه فایل‌های پیکربندی موجود در لینوکس، فایل‌های متنی هستند. بنابراین برای پیکربندی برنامه‌ها و سرویس‌های لینوکس شما باید این فایل‌ها را باز نموده و متن داخل آن را تغییر دهید. همچنین پس از تغییر در فایل و ذخیره آن باید سرویس مورد نظر را مجدد راه اندازی نمایید. بنابراین نحوه کار کردن با فایل‌های متنی در لینوکس بسیار حائز اهمیت می‌باشد. به همین دلیل در این بخش شما با نحوه تغییر و ویرایش فایل‌های متنی آشنا می‌شوید.

در ادامه ما برای توضیح ویرایش فایل‌ها در لینوکس از فایل‌های مربوط به سیستم تشخیص نفوذ Snort استفاده می‌کنیم. اگر شما این ابزار را بر روی کالی لینوکس خود ندارید، می‌توانید با استفاده از دستور apt-get install snort این ابزار را بر روی سیستم خود نصب نمایید.

همانطور که در بخش پیشین هم به آن اشاره کردیم، یکی از ابزارهای کاربردی در لینوکس، cat می‌باشد که با آن می‌توانید محتویات فایل را مشاهده و یا ویرایش نمایید. علاوه بر این ابزار، دستورات دیگری نیز در لینوکس وجود دارد که در ادامه به آن‌ها اشاره خواهیم کرد.

دستور head

از این دستور برای مشاهده بخش‌های ابتدایی فایل‌ها استفاده می‌شود. به صورت پیش فرض این دستور، ۱۰ خط ابتدایی فایل را به شما نمایش می‌دهد. در ادامه از این دستور برای نمایش بخش ابتدایی فایل پیکربندی snort استفاده می‌کنیم:

head /etc/snort/snort.conf


```
root@kali:~# head /etc/snort/snort.conf
#--------------------------------------------------
#   VRT Rule Packages Snort.conf
#
#   For more information visit us at:
#     http://www.snort.org                   Snort Website
#     http://vrt-blog.snort.org/    Sourcefire VRT Blog
#
#     Mailing list Contact:      snort-sigs@lists.sourceforge.net
#     False Positive reports:    fp@sourcefire.com
#     Snort bugs:                bugs@snort.org
```

همچنین شما می‌توانید با استفاده از عبارت - در دستورhead ، تعداد خطوط ابتدایی از فایل که نمایش داده می‌شود را تعیین نمایید:

head -20 /etc/snort/snort.conf

دستور tail

این دستور شبیه به دستور head بوده ولی برای نمایش خطوط انتهایی فایل از آن استفاده می‌شود.

tail /etc/snort/snort.conf

```
root@kali:~# tail /etc/snort/snort.conf
# include $SO_RULE_PATH/smtp.rules
# include $SO_RULE_PATH/snmp.rules
# include $SO_RULE_PATH/specific-threats.rules
# include $SO_RULE_PATH/web-activex.rules
# include $SO_RULE_PATH/web-client.rules
# include $SO_RULE_PATH/web-iis.rules
# include $SO_RULE_PATH/web-misc.rules

# Event thresholding or suppression commands. See threshold.conf
include threshold.conf
```

همانند دستور head شما می‌توانید با استفاده از عبارت - در دستورtail ، تعداد خطوط انتهایی از فایل که نمایش داده می‌شود را تعیین نمایید:

tail -20 /etc/snort/snort.conf

دستور nl

در اغلب موارد شما می‌بایست فایل‌هایی با خطوط زیاد را ویرایش نمایید. در برخی موارد شما نیاز دارید تا شماره خطوط فایل‌ها را مشاهده نمایید. برای این منظور از دستور nl استفاده می‌شود.

nl /etc/snort .conf | grep output

```
root@kali:~# nl /etc/snort/snort.conf | grep output
    33  #  6) Configure output plugins
   445  # Step #6: Configure output plugins
   450  # output unified2: filename merged.log, limit 128, nostamp, mpls_event_types, vlan_event_types
   451  output unified2: filename snort.log, limit 128, nostamp, mpls_event_types, vlan_event_types
   453  # output alert_unified2: filename snort.alert, limit 128, nostamp
   454  # output log_unified2: filename snort.log, limit 128, nostamp
   456  # output alert_syslog: LOG_AUTH LOG_ALERT
   458  # output log_tcpdump: tcpdump.log
```

با توجه به خروجی دستور، ما به دنبال عبارت Step #6 هستیم و البته پنج خط پیش از آن را نیز نیاز داریم. بدین منظور با توجه به شماره خط عبارت مورد نظر از دستور زیر برای مشاهده پنج خط پیشین عبارت Step #6 استفاده می‌نماییم:

tail -n+440/etc/snort/snort.conf | head -n 6

نکته‌ای که در این جا حائز اهمیت است، وجود خطوط خالی است که با توجه به اینکه به خطوط خالی شماره تعلق نمی‌گیرد، دستور بالا خروجی صحیحی را به ما نمایش نمی‌دهد. بدین منظور دستور nl را با سوییچ b- و مقدار a فراخوانی می‌کنیم تا به خطوط خالی نیز شماره تعلق بگیرد.

```
root@kali:~# nl -b a /etc/snort/snort.conf | grep output
    34  #  6) Configure output plugins
   529  # Step #6: Configure output plugins
   535  # output unified2: filename merged.log, limit 128, nostamp, mpls_event_types, vlan_event_types
   536  output unified2: filename snort.log, limit 128, nostamp, mpls_event_types, vlan_event_types
   539  # output alert_unified2: filename snort.alert, limit 128, nostamp
   540  # output log_unified2: filename snort.log, limit 128, nostamp
   543  # output alert_syslog: LOG_AUTH LOG_ALERT
   546  # output log_tcpdump: tcpdump.log
```

همانطور که در تصویر بالا نیز قابل مشاهده می‌باشد، شماره خط عبارت مورد نظر ما به ۵۲۹ تغییر پیدا کرده است. به همین منظور به جای عدد ۴۴۰ در مثال پیشین باید از عدد ۵۲۴ استفاده کنیم تا پنج خط پیشین عبارت مورد نظر ما را نمایش دهد.

علاوه بر روش مذکور شما می‌توانید از دستور grep نیز برای حذف خطوط خالی استفاده نمایید. بدین منظور ابتدا با دستور grep و با استفاده از عبارت نقطه، خطوط خالی را حذف نموده و سپس از شماره خطوط برای چاپ ۵ خط پیشین عبارت مورد نظر استفاده می‌کنیم:

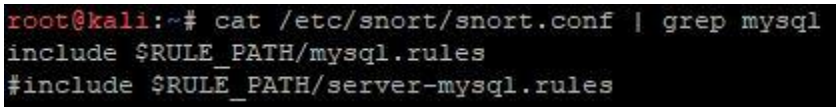
```
root@kali:~# grep . /etc/snort/snort2.conf | tail -n+440 | head -n 6
#    priority whitelist, \
#    nested_ip inner, \
#    whitelist $WHITE_LIST_PATH/white_list.rules, \
#    blacklist $BLACK_LIST_PATH/black_list.rules
##################################################
# Step #6: Configure output plugins
```


دستور sed

دستور sed این امکان را به شما می‌دهد تا بوسیله آن عبارت مورد نظر خود را در یک فایل پیدا نموده و اقدام مورد نظر را بر روی آن انجام دهید. نام این دستور کوتاه شده عبارت Stream Editor می‌باشد. در ابتدایی ترین حالت، دستور sed همانند Find and Replace در ویندوز عمل می‌کند.

در ابتدا ما قصد جستجوی عبارت mysql در فایل تنظیمات snort را داریم که بدین منظور از دستور زیر استفاده می‌کنیم:

cat /etc/snort/snort.conf | grep mysql



در ادامه ما قصد داریم تا عبارت mysql را با عبارت MySQL در فایل تنظیمات snort جابجا نماییم. بدین منظور از دستور sed به شکل زیر استفاده می‌کنیم و خروجی آن را در یک فایل جدید با نام snort1.conf ذخیره می‌نماییم.

sed s/mysql/MySQL/g /etc/snort/snort.conf > snort1.conf

در دستور بالا عبارت s برای جستجو استفاده شده و پس از اسلش عبارت مورد جستجو و بعد از آن و عبارت اسلش بعد عبارتی که باید جایگزین عبارت اول شود قرار داده می‌شود. عبارت g نیز موجب می‌شود تا کلیه عبارت مورد نظر جایگزین شوند و در صورت عدم استفاده از g تنها اولین عبارت جایگزین می‌شود.

به مثال‌های زیر و خروجی مربوط به آن‌ها توجه نمایید:

content of file:
foo bar foo bar foo bar foo bar

sed ‘s/foo/FOO/’
FOO bar foo bar foo bar foo bar

sed ‘s/foo/FOO/3’
foo bar foo bar FOO bar foo bar

sed ‘s/foo/FOO/g’
FOO bar FOO bar FOO bar FOO bar

sed ‘s/foo/FOO/3g’
foo bar foo bar FOO bar FOO bar

مشاهده فایل با more و less

دستوراتی مانند cat بسیار کاربردی هستند ولی زمانی که فایل مورد نظر ما دارای خطوط زیادی باشد، دستور cat کلیه محتویات فایل را در صفحه به صورت یک جا چاپ می‌کند. برای مشاهده خطوط بالایی فایل نیاز به Scroll صفحه خواهیم داشت. این موضوع در فایل‌های بزرگ جالب به نظر نمی‌رسد و کار ما را برای مشاهده محتویات فایل کمی با سختی روبرو می‌کند. برای حل این مشکل می‌توان از دستور more یا less استفاده نمود.

دستور more خروجی را به صورت صفحه به صفحه نمایش می‌دهد و شما با فشردن کلید Space می‌توانید به صفحه بعدی رفته و در انتها نیز با فشردن کلید q مرور فایل را پایان دهید.

علاوه بر دستور more شما می‌توانید از دستور less نیز استفاده کنید. دستور less علاوه بر اینکه امکان مشاهده صفحه به صفحه فایل را فراهم می‌کند، شما می‌توانید عبارت مورد نظر خود را نیز در آن فیلتر نموده و به صورت پرنگ تر مشاهده نمایید. برای فیلتر نمودن یک عبارت خاص ابتدا دستور less و سپس آدرس فایل مورد نظر را تایپ نموده و پس از فشردن کلیدEnter ، عبارت / را تایپ نموده و سپس عبارت مورد نظر را تایپ کنید. در ادامه با فشردن کلید Space اگر عبارت تایپ شده در هر جایی از فایل باشد، آن عبارت با بکگراند مشخص نمایش داده خواهد شد.

```
#################################################
# Step #6: Configure output plugins
# For more information, see Snort Manual, Configuring Snort - Output Modules
#################################################

# unified2
# Recommended for most installs
# output unified2: filename merged.log, limit 128, nostamp, mpls_event_types, vlan_event_types
output unified2: filename snort.log, limit 128, nostamp, mpls_event_types, vlan_event_types

# Additional configuration for specific types of installs
# output alert_unified2: filename snort.alert, limit 128, nostamp
# output log_unified2: filename snort.log, limit 128, nostamp

# syslog
# output alert_syslog: LOG_AUTH LOG_ALERT

# pcap
# output log_tcpdump: tcpdump.log

# metadata reference data.  do not modify these lines
include classification.config
include reference.config
```

منابع این بخش:

https://serverfault.com/

https://www.howtoforge.com/

https://www.linuxquestions.org/

فصل چهارم

آنالیز و مدیریت شبکه در لینوکس

یکی از مواردی که برای یک تست نفوذگر حائز اهمیت می‌باشد، درک از شبکه و فرآیندهای موجود در آن است. هنگامی که تست نفوذگر به یک سرور مبتنی بر لینوکس دسترسی پیدا می‌کند، جهت بهره برداری هر چه بهتر از این سرور باید با دستورات مرتبط به شبکه در این سیستم‌عامل آشنایی لازم را داشته باشد. در این فصل ما به آنالیز و مدیریت شبکه در لینوکس می‌پردازیم.

دستور ifconfig

یکی از دستورات پایه و کاربردی در مدیریت شبکه سیستم‌عامل لینوکس، دستور ifconfig می‌باشد. با استفاده از این دستور شما می‌توانید اطلاعات مربوط به کارت شبکه مانند نام کارت شبکه(eth0) ، آدرس مک(00:0c:29:ba:82:0f) و آدرس IP (10.0.0.1) را مشاهده نمایید.

نکته: ممکن است سیستم‌عامل مورد نظر شما بیشتر از یک کارت شبکه داشته باشد. بنابراین نام کارت‌های شبکه شما می‌تواند eth1 ، eth2و... باشد. همچنین منظور از eth همان ethernet می‌باشد که بیانگر کارت شبکه کابلی است.

نکته: احتمالاً نام دیگری را نیز در لیست کارت‌های شبکه با وارد کردن دستور ifconfig خواهید دید و آن عبارت lo می‌باشد. این عبارت بیانگر کارت شبکه loopback بوده و در برخی موارد Localhost هم به آن گفته می‌شود. از این آدرس به منظور ارتباط با سیستم خودتان استفاده می‌شود و زمانی که قصد ارتباط با سیستم خود را داشته باشید از آدرس localhost برای این منظور استفاده خواهید نمود.

همچنین سرویس‌ها و نرم افزارهایی که بر روی سیستم شما قرار دارند از این آدرس استفاده می‌نمایند. همچنین به صورت عمومی از آدرس ۱۲۷.۰.۰.۱ به عنوان آدرسی برای loopback مورد استفاده قرار می‌گیرد.

دستور iwconfig

علاوه بر کارت‌های شبکه مذکور، در صورت وجود کارت شبکه وایرلس، این کارت شبکه و مشخصات آن با نام wlan0 نمایش داده خواهند شد. لازم به ذکر است برای مشاهده اطلاعات شبکه وایرلس در لینوکس می‌توانید از دستور iwconfig استفاده نمایید.

در خروجی این دستور علاوه بر مشخصات کارت شبکه، نوع شبکه وایرلس که با عنوان ۸۰۲.۱۱ شناخته می‌شود نیز قابل مشاهده است که نوع آن می‌تواند از سری a ، b، nو موارد دیگر باشد.

مورد دیگری که در خروجی دستور iwconfig قابل مشاهده می‌باشد، Mode است. این بخش وضعیت کارت شبکه را مشخص می‌کند که در حالت مدیریت یا Managed بوده و یا حالت بی قاعده یا Promiscuous می‌باشد. لازم به ذکر است که حالت Managed در مقابل حالت Promiscuous می‌باشد. از حالت Promiscuous برای انجام حملات وایرلس و دریافت کلیه بسته های شبکه وایرلس استفاده می‌گردد.

در ادامه خروجی دستور iwconfig مشاهده می‌شود که کارت شبکه وایرلس به هیچ اکسس پوینتی متصل نمی‌باشد (Not Associated)و همچنین قدرت سیگنال دهی آن نیز مشخص می‌شود.

تغییر اطلاعات شبکه

اولین تغییر در اطلاعات کارت شبکه، تغییر آدرس IP می‌باشد. برای تغییر در آدرس IP کارت شبکه از دستور ifconfig به همراه نام کارت شبکه و آدرس IP مورد نظر استفاده می‌شود:

```
ifconfig eth0 10.0.0.1
```

همچنین شما می‌توانید آدرس زیر شبکه یا همان subnet mask و Broadcast را نیز برای این آدرس تعیین کنید که در غیر این صورت آدرس subnet mask به صورت پیش فرض در نظر گرفته می‌شود:

```
ifconfig eth0 10.0.0.1 netmask 255.255.0.0 broadcast 10.0.0.255
```


تغییر یا جعل آدرس مک

یکی از روش‌های که برای مخفی نگه داشتن هویت نفوذگر از آن استفاده می‌شود، تغییر آدرس مک می‌باشد که یکی از روش‌های کاربردی برای عبور از مکانیزم‌های کنترل دسترسی مبتنی بر مک آدرس می‌باشد. برای تغییر آدرس مک باید ابتدا کارت شبکه را غیرفعال نموده و سپس اقدام به تغییر در آدرس مک نمود. پس از آن نیز کارت شبکه باید مجدداً فعال گردد:

```
ifconfig eth0 down
ifconfig eth0 hw ether 00:11:22:33:44:55
ifconfig eth0 up
```

دریافت آدرس IP از سرور DHCP

برای اختصاص آدرس IP به صورت خودکار از سرویس DHCP استفاده می‌شود که این سرویس می‌تواند بر روی یک سرور ویندوز، یک سخت افزار مانند روتر و یا یک سرور لینوکسی پیاده سازی شود. این سرویس قادر است تا آدرسIP ، Subnet Mask، Default Gateway، آدرس DNS و موارد مشابه را به کلاینت اختصاص دهد. برای درخواست IP از سرور DHCP از دستور dhclient به همراه نام کارت شبکه استفاده می‌شود:

```
dhclient eth0
```

پس از وارد نمودن دستور بالا در ترمینال لینوکس، جهت اطمینان از اختصاص آدرس می‌توانید از دستور ifconfig استفاده نمایید.

تغییر اطلاعات DNS

یکی از سرویس‌های مهم دیگر در شبکه، سرویس DNS می‌باشد. وظیفه این سرویس تبدیل آدرس IP به نام و بالعکس است. یکی از دستورات کاربردی در این بخش، دستور dig بوده که یکی از راه‌های جمع آوری اطلاعات از طریق DNS می‌باشد. در صورتی که دستور dig را همراه با سوییچ ns به کار بگیرید، آدرس سرورهای نام مربوط به دامنه مورد نظر را به شما نمایش خواهد داد.

```
root@kali:~# dig sans.org ns

; <<>> DiG 9.11.5-P4-5.1+b1-Debian <<>> sans.org ns
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5287
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;sans.org.                      IN      NS

;; ANSWER SECTION:
sans.org.               4780    IN      NS      dns31a.sans.org.
sans.org.               4780    IN      NS      dns31b.sans.org.
sans.org.               4780    IN      NS      dns21a.sans.org.
sans.org.               4780    IN      NS      dns21b.sans.org.

;; Query time: 33 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Wed Nov 06 07:26:09 EST 2019
;; MSG SIZE  rcvd: 121
```

همچنین برای به دست آوردن اطلاعات مربوط به mail server شما می‌توانید از سوییچ mx استفاده نمایید.

برای تغییر در تنظیمات سرور DNS کاربر، شما می‌توانید وارد مسیر /etc/resolv.conf شوید. این فایل را با یک ویرایشگر متنی باز نموده و آدرس مربوط به سرور DNS مورد نظر خود را وارد نمایید.

```
root@kali:~# cat /etc/resolv.conf
# Generated by NetworkManager
nameserver 8.8.8.8
```

همچنین شما می‌توانید از دستور زیر برای اضافه کردن آدرس سرور DNS خود به فایل DNS استفاده نمایید:

echo “nameserver 8.8.8.8” > /etc/resolv.conf

Map نمودن آدرس IP به نام

یکی از فایل‌هایی که در سیستم‌عامل لینوکس وجود دارد، فایل hosts می‌باشد. در این فایل آدرس‌های IP به نام‌های مورد نظر اصطلاحاً Map شده‌اند. همچنین اولویت در تبدیل نام در سیستم با این فایل بوده و سپس تنظیمات DNS که در بخش پیشین به آن اشاره شد، اجرا می‌گردد.

مسیر این فایل /etc/hosts می‌باشد.

```
root@kali:~# cat /etc/hosts
127.0.0.1       localhost
127.0.1.1       kali

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

این فایل را نیز می‌توان با ویرایشگر متنی باز نموده و تغییرات لازم را در آن ایجاد نمود.

فصل پنجم

نصب و حذف نرم افزار در لینوکس

یکی از مواردی که در لینوکس باید به آن توجه داشته باشید نحوه نصب و حذف برنامه‌ها در این سیستم‌عامل می‌باشد. برخی از نرم افزارها برای اجرا، نیازمند نصب برنامه‌های دیگری هستند و اغلب شما آنچه را که نیاز دارید در قالب یک بسته نرم افزاری پیدا خواهید کرد که گروهی از فایل‌های کتابخانه ای و وابستگی‌های دیگری است که برای اجرای موفق یک نرم افزار به آن نیاز خواهید داشت.

در این فصل با سه روش اضافه نمودن نرم افزارها در لینوکس که شامل مدیریت بستهapt ، مدیریت نصب مبتنی بر GUI و git آشنا خواهید شد.

استفاده از apt برای مدیریت نرم افزار

در سیستم‌عامل‌های لینوکس مبتنی بر Debian که کالی لینوکس و ubuntu هم شامل آن‌ها می‌شوند، مدیریت نرم‌افزار به صورت پیش فرض بر عهده apt یا Advanced Packaging Tool می‌باشد. در ساده ترین حالت شما می‌توانید از دستور apt-get برای دانلود و نصب بسته‌های نرم افزاری جدید استفاده نمایید، اما شما همچنین می‌توانید update و upgradeنرم افزارها را نیز به وسیله این دستور انجام دهید.

پیش از نصب یک بسته شما باید بررسی کنید که آیا بسته مورد نظر شما در مخزن یا همان Repository سیستم شما موجود می‌باشد یا خیر. شما می‌توانید برای این منظور از دستور زیر استفاده نمایید:

```
apt-cache search keyword
```

به عنوان مثال ما به دنبال بررسی سیستم تشخیص نفوذ Snort هستیم و خروجی دستور مربوط به بررسی آن به صورت زیر خواهد بود:

```
kali >apt-cache search snort
fwsnort - Snort-to-iptables rule translator
ippl - IP protocols logger
--snip--
snort - flexible Network Intrusion Detection System
snort-common - flexible Network Intrusion Detection System - common files
--snip--
```

همانطور که در تصویر قابل مشاهده می‌باشد، تعدادی فایل که کلید واژه snort در آن‌ها وجود دارند لیست می‌شوند که نرم‌افزار مورد نظر ما snort – flexible Network Intrusion Detection System می‌باشد.

حالا که ما متوجه شدیم، بسته snort در Repository سیستم وجود دارد، با استفاده از دستور apt-get می‌توانیم اقدام به نصب این نرم افزار نماییم. برای نصب این ابزار از دستور زیر استفاده می‌کنیم:

```
apt-get install snort
```

حذف یک نرم افزار

به منظور حذف یک نرم افزار شما می‌توانید از دستور زیر استفاده نمایید:

```
apt-get remove snort
```

توجه داشته باشید که دستور remove فایل‌های پیکربندی را حذف نمی‌کند و این بدین معنی است که شما می‌توانید در آینده بدون تنظیم مجدد، همان بسته را مجدد نصب نمایید. اگر شما قصد حذف فایل‌های پیکربندی را دارید می‌توانید از دستور زیر برای این منظور استفاده نمایید:


```
apt-get purge snort
```

ممکن است شما در هنگام حذف برنامه با پیام The following packages were automatically installed and are no longer require مواجه شوید. توجه داشته باشید که برای کوچکتر شدن موارد، بسیاری از بسته‌های لینوکس به واحدهای نرم افزاری مختلف تقسیم می‌شوند که بسیاری از برنامه‌های مختلف ممکن است از آن‌ها استفاده کنند. هنگامی که Snort را نصب کردید، چندین وابستگی یا کتابخانه را با آن نصب نمودید که Snort برای اجرای

درست به آن‌ها نیاز داشته است. هم اکنون که Snort را حذف نموده اید، آن کتاب خانه‌ها یا وابستگی‌های دیگر را لازم نخواهید داشت بنابراین آن‌ها نیز حذف می‌گردند.

پیام مذکور به این مورد اشاره دارد.

بروزرسانی Packages

مخازن نرم افزارها به صورت دوره‌ای با نسخه‌های جدیدی از نرم افزارها، بروزرسانی می‌شوند. این بروزرسانی‌ها به طور خودکار به شما نخواهد رسید، بنابراین شما باید برروزرسانی‌ها را درخواست نمایید تا با دریافت آن‌ها از مخازن موجود، بروزرسانی انجام پذیرد.

توجه داشته باشید که بروزرسانی یا Update با ارتقا یا Upgrade متفاوت می‌باشد.

Update لیست بسته‌های موجود برای دانلود از مخازن را به روز می‌کند در حالی که Upgrade بسته را به آخرین نسخه موجود در مخزن ارتقا می‌دهد.

به منظور بروزرسانی شما می‌توانید از دستور زیر استفاده نمایید:

```
apt-get update
```

با این دستور لیست نرم افزارهای موجود در Repository سیستم شما بروزرسانی خواهد شد. در صورتی که بروزرسانی موفقیت آمیز باشد، شما در ترمینال عبارت Reading package lists... Done را مشاهده خواهید کرد. با توجه به تعداد مخازن موجود، و سایز آن‌ها، میزان زمان بروزرسانی متفاوت خواهد بود.

ارتقا یا Upgrade Packages

به منظور ارتقای بسته‌های موجود بر روی سیستم شما می‌توانید از دستور زیر استفاده نمایید:

```
apt-get upgrade
```

به دلیل اینکه ارتقای بسته‌های ممکن است نرم افزار شما را دچار تغییر نماید، شما باید برای اجرای این دستور با کاربر root وارد شده باشید.

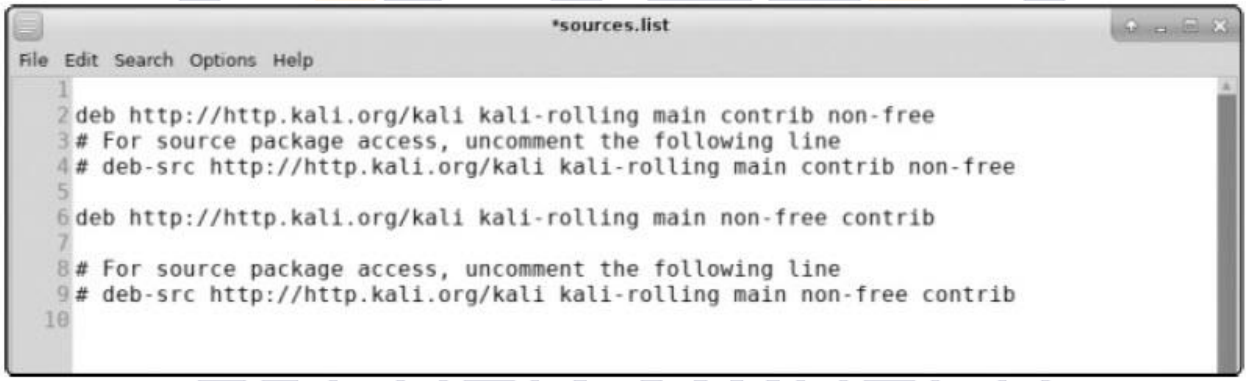
اضافه نمودن مخازن به فایل sources.list

سرورهایی که نرم افزارهای مختلف را برای توزیع‌های خاص لینوکس در اختیار دارند به عنوان مخازن یا همان Repositoryشناخته می‌شوند. تقریباً هر توزیع از لینوکس، مخازن مربوط به خود را دارد که برای آن پیکربندی شده است و ممکن است برای توزیع‌های دیگر به خوبی یا اصلاً کار نکند. مخازن موجود بر روی سیستم شما در یک فایل با نام sources.listذخیره شده‌اند و شما می‌توانید این فایل را تغییر داده و مخازن مورد نظر خود را برای دانلود نرم افزارهای مختلف به آن اضافه نمایید. اضافه نمودن مخازن Ubuntu پس از مخازن کالی لینوکس می‌تواند گزینه مناسبی باشد. در این حالت ابتدا نرم‌افزار در مخازن کالی لینوکس جست و جو شده و در صورت عدم وجود نرم افزار در آن، مخازنUbuntu برای این منظور جست و جو خواهد شد.

مسیر ذخیره سازی فایل مخازن به صورت زیر است که شما می‌توانید با ابزارهای مختلفی مانند leafpad یا nano اقدام به مشاهده و ویرایش آن نمایید:

```
leafpad /etc/apt/sources.list
```

پس از اجرای دستور بالا شما پنجره‌ای مشابه پنجره زیر مشاهده خواهید کرد:



اغلب توزیع‌های لینوکس، مخازن خود را به دسته مختلفی تقسیم می‌کنند. به عنوان نمونه، Ubuntu دارای دسته بندی‌های زیر برای مخازن خود می‌باشد:

Main: شامل نرم افزارهای متن باز پشتیبانی شده یاSuported

universe: شامل نرم افزارهای متن بازCommunity-Maintained

multiverse: شامل نرم افزارهایی که توسط قانون کپی رایت یا موارد قانونی دیگر محدود شده اند.

restricted: شامل درایورهای دستگاه های اختصاصی

backports: شامل بسته‌هایی از نسخه های بعدی

توصیه می‌شود که از مخازن آزمایشی یا نا پایدار در فایل sources.list استفاده نکنید چرا که می‌تواند موجب بروز مشکل در سیستم شما شده و به آن صدمه وارد کند.

جهت اضافه نمودن یک مخزن جدید به لیست مخازن شما تنها کافی است که آدرس آن را در انتهای فایل sources.list اضافه نمایید. به عنوان مثال برای نصب Oracle Java 8 که به صورت پیش فرض در سیستم کالی لینوکس امکان نصب آن وجود ندارد، باید ابتدا آدرس مخازن زیر را به فایل sources.list اضافه نموده و سپس دستور apt-get install oracle-java8-installerرا اجرا نمایید:

deb http://ppa.launchpad.net/webupd8team/java/ubuntu trusty main

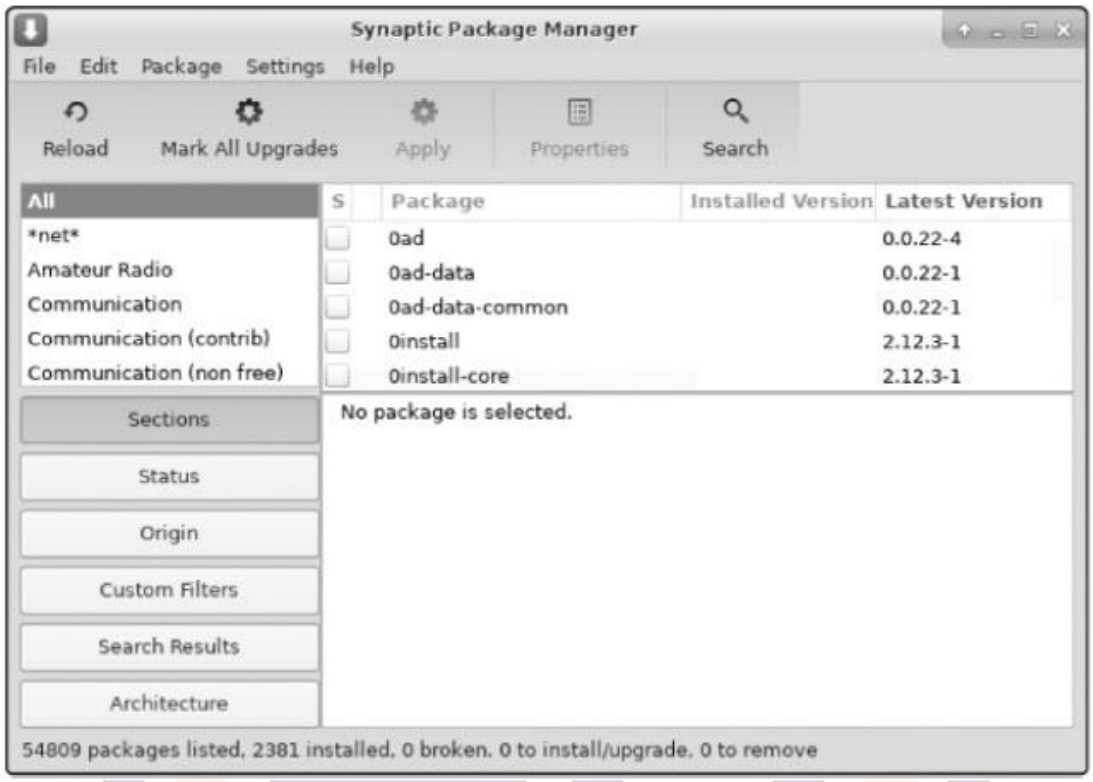
debsrc http://ppa.launchpad.net/webupd8team/java/ubuntu precise main

استفاده از یک نصب کننده گرافیکی

نسخه‌های جدیدتر کالی لینوکس دیگر شامل یک ابزار نصب گرافیکی نیست، اما شما همواره می‌توانید یکی از این ابزارها را با استفاده از دستور apt-get بر روی سیستم نصب کنید. دو ابزار رایج برای این منظور Synaptic و Gdebi می‌باشند. به عنوان نمونه برای نصب Synaptic باید دستور زیر را اجرا نمایید:

apt-get install synaptic

پس از نصب این برنامه شما می‌توانید از منوی Setting و با نام Synaptic Package Manager آن را فراخوانی نمایید:



هم اکنون شما می‌توانید بسته مورد نظر را در این برنامه جست و جو نموده و نصب نمایید.

نصب نرم افزار با git

در برخی از موارد، ابزارهای مورد نیاز شما در مخازن موجود نیستند اما ممکن است بر روی گیت هاب بتوانید آن‌ها را پیدا کنید. گیت هاب، وب سایتی است که توسعه دهندگان از آن برای به اشتراک گذاری نرم افزارهایشان برای دانلود، استفاده و دریافت بازخورد بهره می‌گیرند. به عنوان نمونه اگر شما به دنبال ابزار bludiving هستید که یک ابزار تست نفوذ بلوتوث می‌باشد، می‌توانید با جست و جوی آن درون سایت گیت هاب، لینک مربوط به این ابزار را پیدا نموده و با استفاده از دستور git clone آن را بر روی سیستم خود نصب نمایید:


```
git clone https://www.github.com/balle/bluediving.git
```

دستور git clone کلیه فایل‌ها و داده‌های برنامه را بر روی سیستم شما کپی می‌کند و یک دایرکتوری جدید با نام نرم افزار برای شما ایجاد خواهد کرد.

فصل ششم

سطح دسترسی فایل و فولدر در لینوکس

لینوکس هم مانند هر سیستم‌عامل دیگر، روش‌هایی را برای دسترسی به فایل‌ها و فولدرها فراهم می‌کند. در لینوکس به کاربر Root و مالک فایل یا فولدر، اجازه داده می‌شود تا با اعطای مجوزهای انتخاب شده به کاربران، امکان خواندن، نوشتن یا اجرا شدن را توسط دسترسی‌های غیرمجاز کنترل نموده و آن‌ها را از دستکاری ناخواسته و غیرمجاز محافظت نماید. این ویژگی در سیستم‌عامل‌هایی که دارای چندین کاربر مختلف می‌باشند یک ضرورت محسوب می‌شود.

در ادامه به شما نشان داده خواهد شد که چگونه می‌توانید مجوزهای مربوط به فایل‌ها و فولدرها را در لینوکس تنظیم نمایید.

انواع کاربران مختلف

همانطور که شما می‌دانید، در لینوکس کاربر Root دارای قدرت بسیار زیادی بوده و اساساً قادر به انجام هر کاری بر روی سیستم می‌باشد و همچنین کاربران دیگر دارای قابلیت‌های محدودتری نسبت به این کاربر می‌باشند. این کاربران معمولا در گروه‌هایی جمع می‌شوند که اغلب دارای عملکرد مشابهی هستند.

به عنوان مثال واحدهای مالی، مهندسی، فروش و ... هر کدام می‌توانند به عنوان یک گروه تعریف شده و کاربران مخصوص خود را داشته باشند.

در یک غالب کلی افراد دارای نیاز مشابه در یک گروه قرار می‌گیرند و مجوزهای لازم نیز به آن‌ها اعطا می‌گردد و اعضای این گروه، سطح دسترسی اعطا شده به گروه را به ارث خواهد برد.

انواع سطح دسترسی

هر فایل و فولدر در لینوکس دارای سطح دسترسی مربوط به خود می‌باشد که این سطح دسترسی به سه بخش زیر تقسیم می‌گردد:

R مخفف Read: این نوع از دسترسی تنها برای باز کردن و مشاهده فایل استفاده می‌شود.
W مخفف Write: این نوع از دسترسی برای نوشتن است و به کاربر اجازه مشاهده و ویرایش یک فایل را خواهد داد.
X مخفف Execute: این نوع از دسترسی برای اجرا بوده و به کاربر اجازه اجرای یک فایل را می‌دهد.
به این ترتیب کاربر Root می‌تواند سطوح دسترسی مختلف را مطابق با نیاز هر یک از کاربران به آن‌ها اعطا نماید.

هنگامی که یک فایل ایجاد می‌شود، معمولاً کاربری که آن را ایجاده نموده است، به عنوان مالک آن شناخته می‌شود. صاحب فایل می‌تواند سطوح دسترسی مختلف را به آن اعطا نماید.

اعطای مالکیت به یک کاربر خاص

به منظور تغییر مالک یک فایل به یک کاربر دیگر، که این کاربر قادر به کنترل سطح دسترسی آن باشد، از دستور chown استفاده می‌شود. در این دستور باید نام کاربر و مسیر فایلی که قصد تغییر مالک آن را داریم، وارد نماییم:

```
chown bob /tmp/bobsfile
```

اعطای مالکیت به یک گروه خاص

جهت تغییر مالکیت یک فایل از یک گروه به گروهی دیگر از دستور chgrp استفاده می‌شود.

```
chgrp security test.sh
```

دستور بالا گروه مالک فایل test.sh را به گروه security منتقل می‌کند.

بررسی سطوح دسترسی

برای بررسی اعمال Permission ها به فایل‌ها و پوشه‌های مد نظر، از دستور ls با سوییچ l- استفاده می‌شود. در تصویر زیر دستور مذکور برای فایل hashcat اجرا شده است:

```
kali >ls -l /usr/share/hashcat
total 32952
❶    ❷ ❸ ❹    ❺      ❻     ❼
drwxr-xr-x 5 root root  4096   Dec 5 10:47 charsets
-rw-r--r-- 1 root root  33685504 June 28 2018 hashcat.hcstat
-rw-r--r-- 1 root root  33685504 June 28 2018 hashcat.hctune
drwxr -xr-x 2 root root  4096   Dec 5 10:47 masks
drwxr -xr-x 2 root root  4096   Dec 5 10:47 OpenCL
drwxr -xr-x 3 root root  4096   Dec 5 10:47 rules
```

خروجی دستور ls -la حاوی اطلاعاتی در خصوص فایل مورد نظر می‌باشند که در ادامه به توضیح این موارد می‌پردازیم:

بخش یک : نوع فایل را مشخص می‌کند که در تصویر بالا عبارت d بیانگر ماهیت دایرکتوری و عبارت دش یا – بیانگر ماهیت فایل می‌باشد.

بخش دو : سطح دسترسی یا Permission مربوط به مالک (سه کاراکتر اول)، گروه(سه کاراکتر دوم) و کلیه کاربران(سه کاراکتر سوم) کاربران را نشان می‌دهد.

بخش سوم : تعداد لینک های هر یک از موارد را نمایش می‌دهد.

بخش چهارم : مالک هر بخش را مشخص می‌کند.

بخش پنجم : سایز هر بخش را به بایت نشان می‌دهد.

بخش ششم : زمان ایجاد یا آخرین تغییر را نمایش می‌دهد.

بخش هفتم : نام فایل را نشان می‌دهد.

در ادامه به توضیحاتی در مورد بخش دوم می‌پردازیم.

هر یک از این سطوح دسترسی با سه کاراکتر مشخص می‌شوند که می‌توانند شامل rwx باشند. اگر شما یک r در ابتدای هر یک از سطوح دسترسی مشاهده نمودید، این بدین معناست امکان باز نمودن و خواندن فایل یا دایرکتوری وجود دارد.

در صورت وجود w در ادامه آن، امکان تغییر یا نوشتن بر روی فایل یا دایرکتوری وجود دارد و در صورت مشاهده x در ادامه کاراکترهای پیشین، دسترسی اجرا به فایل یا دایرکتوری اعطا شده است. همچنین در صورتی که به جای هر یک از موارد rwx یک عبارت - قرار داده شده بود، به معنی عدم دسترسی فایل یا دایرکتوری به آن مورد خاص بوده که می‌تواند دسترسی خواندن، نوشتن یا اجرا کردن باشد.

به عنوان مثال تصویر زیر را در نظر بگیرید:

```
-rw-r--r-- 1  root  root   33685504 June 28 2018 hashcat.hcstat
```


عبارت – در ابتدای بخش دسترسی، نمایانگر ماهیت فایل است.

دسترسی‌های فایل مذکور دارای سه بخش زیر است:

-rw-: این بخش به ما می‌گوید که دسترسی مالک فایل تنها خواندن و نوشتن بر روی آن است و امکان اجرای این فایل برای مالک وجود ندارد.

-r–: این بخش به ما می‌گوید که دسترسی گروه به این فایل تنها خواندن بوده و امکان تغییر و اجرای این فایل برای گروه وجود ندارد.

r–: این بخش به ما می‌گوید که کابران دیگر نیز تنها قادر به خواندن این فایل بوده و امکان تغییر و اجرای آن را ندارند.
لازم به ذکر است که این مجوزها غیرقابل تغییر نبوده و کاربر root و مالک فایل یا دایرکتوری می‌تواند سطح دسترسی آن‌ها را تغییر دهد که در ادامه به چگونگی تغییر سطح دسترسی یا Permission ها می‌پردازیم.

تغییر سطوح دسترسی

برای تغییر سطح دسترسی یک فایل یا دایرکتوری از دستور chmod استفاده می‌شود که بکارگیری این دستور تنها برای کاربر root و مالک فایل یا دایرکتوری امکان پذیر می‌باشد. تغییر سطح دسترسی به دو روش عددی و کاراکتری انجام می‌شود که در ادامه توضیحات و نحوه پیاده سازی هر یک را بررسی خواهیم کرد.

تغییر سطوح دسترسی با استفاده از نماد عددی

در این روش به ازای هر یک از عباراتrwx ، مقادیر ارزش مکانی آن‌ها بر اساس باینری را به دست آورده و سپس از مقدار دسیمال آن برای تغییر سطح دسترسی استفاده می‌کنیم.

در صورت وجود هر یک از مقادیر rwx از عدد یک و در صورت عدم وجود آن‌ها از عدد صفر برای نمایش استفاده می‌نماییم. بدین معنی که اگر سطح دسترسی مالک به یک فایل rwx باشد مقدار باینری آن ۱۱۱ می‌باشد که در صورت تبدیل آن به دسیمال مقدار ۷ بازگردانده خواهد شد. برای درک بیشتر این موضوع به لیست زیر دقت نمایید:

۰۰۰ ۰ —
۰۰۱ ۱ –x
۰۱۰ ۲ -w-
۰۱۱ ۳ -wx
۱۰۰ ۴ r–
۱۰۱ ۵ r-x
۱۱۰ ۶ rw-
۱۱۱ ۷ rwx

با توجه به لیست بالا، در صورتی که قصد تنظیم سطح دسترسی برای هر یک از بخش های مالک، گروه و کاربران را داشته باشیم می‌توانیم از معادل عددی هر بخش استفاده کنیم.

به عنوان مثال اگر قصد اعمال دسترسی خواند و نوشتن به مالک را داشته باشیم دسترسی به شکل rw- شده که مطابق با لیست بالا معادل عدد ۶ بوده و اگر دسترسی برای گروه و کاربران تنها خواندن باشد، برای هر کدام از آن‌ها عبارت r– در نظر گرفته خواهد شد که مقدار عددی هر دو بخش مطابق جدول بالا برابر ۴ می‌باشد و سطح دسترسی کلی به شکل rw-r–r– می‌باشد.

در این صورت عدد تنظیم سطح دسترسی برابر با ۶۴۴ می‌شود که از این عدد برای تغییر سطح دسترسی به شکل زیر استفاده می‌شود:

chmod 644 hashcat

تغییر سطح دسترسی با UGO

اگرچه تعیین سطح دسترسی به روش عددی بیشتر مورد استفاده قرار می‌گیرد ولی روش سمبلیک یا UGO نیز طرفداران خود را دارد. در روش سمبلیک، UGOیک مخفف بوده و حرف U برای User یا همان Owner بوده حرف G برای Groupو حرف O برگرفته از Other می‌باشد.

استفاده از روش UGO بسیار راحت است. ابتدا همانند روش عددی از دستور chmod استفاده کرده و پس از آن برای تغییر سطح دسترسی هر بخش از حرف مورد نظر آن که پیش تر به آن اشاره شد استفاده می‌نماییم. همچنین بوسیله علامات، نوع تغییر در سطح دسترسی را مشخص می‌کنیم:

علامت (-) منجر به حذف دسترسی می‌گردد.

علامت (+) منجر به اضافه نمودن دسترسی می‌گردد.

علامت (=) برای تنظیم یک دسترسی استفاده می‌شود.

برای استفاده از روش UGO ابتدا یکی از حروف آن یعنی U یا G یا O را وارد نموده و سپس بسته به نوعی تغییر سطح دسترسی که مد نظرمان می‌باشد از علامات بالا استفاده می‌نماییم و پس از آن حرف مربوط به سطح دسترسی (rwx) را قرار می‌دهیم و در ادامه نام فایل قرار می‌گیرد:

chmod u-w hachcat

دستور بالا دسترسی Write را از فایل hashcat برای کاربر u یا همان Owner حذف می‌کند. همچنین شما می‌تواند چند نوع سطح دسترسی را در یک دستور وارد نمایید:

```
chmod u+x, o+x hashcat
```

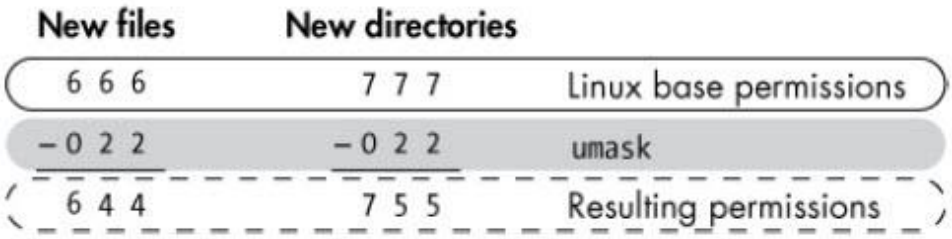
نکته

در برخی موارد شما نیاز به دانلود یک ابزار جدید دارید ولی لینوکس به صورت خودکار سطح دسترسی اجرا را به فایل دانلود شده اختصاص نداده و شما در هنگام اجرای این ابزار جدید با پیام خطا مواجه خواهید شد. برای رفع این مشکل، باید دسترسی Execute را به برنامه اختصاص داده که شما می‌توانید با هر دو روش مذکور این دسترسی را به فایل مورد نظر اعمال نمایید.

تنظیم سطوح دسترسی پیش فرض با Mask

لینوکس به صورت پیش فرض سطح دسترسی را به صورت خودکار به فایل ها و دایرکتوری ها اعمال می‌کند. این سطح دسترسی معمولا برای فایل ها ۶۶۶ و برای فولدرها ۷۷۷ می‌باشد. برای تغییر در این سطوح دسترسی پیش فرض از روش umaskاستفاده می‌شود.

روش umask نشان دهنده مجوزهایی است که می‌خواهید از مجوز پیش فرض بر روی فایل‌ها یا فولدرها حذف کنید. در این روش از سه عدد استفاده می‌شود که هر کدام برای تغییر در سطح دسترسی‌های مربوط به سه بخش rwx است. به تصویر زیر توجه نمایید:



در مثال بالا برای umask عدد ۰۲۲ تنظیم شده است. این بدین معنی است که مقدار ۰۲۲ از مقدار پیش فرض که ۶۶۶ برای فایل و ۷۷۷ برای فولدرها می‌باشد کم شده و از این پس سطح دسترسی پیش فرض برای فایل‌ها در لینوکس ۶۴۴ و دسترسی پیش فرض برای فولدرها در لینوکس ۷۵۵ می‌باشد.

اعطای موقت مجوز root بوسیله SUID

همانطور که می‌دانید، یک کاربر تنها زمانی می‌تواند یک فایل را اجرا کند که دسترسی Execute را بر روی آن داشته باشد. اگر کاربر تنها دسترسی خواندن و نوشتن بر روی یک فایل را داشته باشد، نمی‌تواند آن را اجرا نماید. این مورد ممکن است ساده به نظر برسد ولی استثنائاتی هم در این مورد وجود دارد.

شما ممکن است با موردی مواجه شوید که در آن یک فایل برای اجرا نیاز به دسترسی root داشته باشد؛ حتی برای کاربران معمولی و غیر Root

به عنوان مثال یک فایل که به کاربران اجازه می‌دهد تا کلمه عبور خود را تغییر دهند، نیاز به دسترسی به فایل /etc/shadow دارد که به منظور اجرای این فایل نیاز به دسترسی root می‌باشد و کاربرانی که دسترسی root را ندارند، امکان دسترسی به این فایل و اجرای آن را نخواهند داشت. در این مورد شما می‌توانید با تنظیم بیت SUID بر روی برنامه، دسترسی اجرا را به فایل اختصاص دهید.

در اصل، بیت SUID می‌گوید که هر کاربر می‌تواند فایل را با مجوزهای Owner آن اجرا کند اما این مجوزها فراتر از استفاده از آن فایل نیست. برای تنظیم بیت SUID کافی است تا عدد ۴ را پیش از اعداد مربوط به سطح دسترسی معمولی قرار دهید به عنوان مثال اگر عدد سطح دسترسی برابر ۶۴۴ باشد با اضافه نمودن مقدارSUID ، مقدار نهایی برابر با ۴۶۶۴ می‌شود. برای انجام آن نیز از دستور chmod استفاده می‌شود:

```
chmod 4644 filename
```

اعطای موقت مجوز root بوسیله SGID

SGID هم مجوزهای موقت را اعطا می کند، اما مجوزهای اعطا شده توسط آن مربوط به گروه صاحب فایل می‌باشد. برای تنظیم بیت SGID کافی است عدد ۲ را پیش از عدد سطح دسترسی معمول فایل قرار داد. بنابراین اگر عدد دسترسی برای یک فایل عدد ۶۴۴ باشد، برای تنظیم بیتSGID ، عدد نهایی برابر با ۲۶۴۴ می‌باشد.

برای تنظیم SGID هم از دستور chmod استفاده می‌گردد:

```
chmod 2644 filename
```

بیت Sticky

بیت Sticky یک بیت مجوزی است که می‌توانید آن را برای یک دایرکتوری تنظیم نمایید تا به کاربر امکان حذف یا تغییر نام فایل‌های داخل آن دایرکتوری را بدهد. بیت Sticky یک میراث از سیستم‌های قدیمی یونیکس بوده و سیستم‌های مدرن مانند لینوکس آن را نادیده می‌گیرند. به همین دلیل در این بخش ما بیشتر از این به آن ردازیم ولی به دلیل اینکه شما باید حداقل آن را شنیده باشید در این بخش به آن اشاره شده است.

مجوزهای خاص و بالا بردن دسترسی

برای یک نفوذگر، مجوزهای خاص می‌تواند برای سوء استفاده از سیستم‌های لینوکس از طریق بالا بردن دسترسی مورد استفاده قرار گیرد و به موجب آن، یک کاربر معمولی، امتیازات کاربر root و مجوزهای مرتبط با آن را به دست آورد.

یکی از راه‌های انجام این کار، بهره برداری از بیت SUID است. مدیران سیستم یا توسعه دهندگان نرم افزار ممکن است بیت SUID را بر روی یک برنامه تنظیم کنند تا به آن برنامه اجازه دسترسی به فایل های با امتیازات root را بدهند.

به عنوان مثال، اسکریپت‌هایی که نیاز به تغییر کلمه عبور را دارند، اغلب بیت SUID برای آن‌ها تنظیم شده است. نفوذگر می‌تواند از این مجوزها برای به دست آوردن امتیازات موقت root وانجام فعالیت‌های مخرب مانند دسترسی به کلمات عبور در فایل /etc/shadow استفاده نماید.

برای این منظور ما باید با استفاده از دستور find به دنبال فایل‌هایی که بیت SUID برای آن‌ها تنظیم شده است باشیم. یکی از ویژگی‌های دستور find نسبت به دستورات دیگر مانند locate و which امکان جست و جوی فایل‌هایی با بیت SUID می‌باشد.

دستور زیر برای جست وجوی فایل‌هایی با دسترسی ۴۰۰۰ که مالک آن‌ها root می‌باشد مورد استفاده قرار می‌گیرد:

```
find / -user root -perm -4000
```

تصویر زیر خروجی دستور بالا در سیستم‌عامل کالی لینوکس را نمایش می‌دهد:

```
root@kali:~# find / -user root -perm -4000
find: '/run/user/130/gvfs': Permission denied
find: '/proc/21733': No such file or directory
find: '/proc/21734/task/21734/fd/6': No such file or directory
find: '/proc/21734/task/21734/fdinfo/6': No such file or directory
find: '/proc/21734/fd/5': No such file or directory
find: '/proc/21734/fdinfo/5': No such file or directory
/usr/lib/eject/dmcrypt-get-device
/usr/lib/chromium/chrome-sandbox
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/openssh/ssh-keysign
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/xorg/Xorg.wrap
/usr/bin/gpasswd
/usr/bin/mount
/usr/bin/kismet_cap_linux_bluetooth
/usr/bin/umount
/usr/bin/ntfs-3g
/usr/bin/chsh
/usr/bin/pkexec
/usr/bin/kismet_cap_linux_wifi
/usr/bin/bwrap
/usr/bin/su
/usr/bin/sudo
/usr/bin/fusermount
/usr/bin/newgrp
/usr/bin/passwd
```

در ادامه وارد مسیر /usr/bin شده خروجی آن را به فایل‌هایی که با su شروع می‌شوند محدود می‌نماییم:

```
root@kali:/usr/bin# ls -la su*
-rwsr-xr-x 1 root root  63568 Jul 28 11:14 su
-rwxr-xr-x 1 root root     76 Sep  9 10:06 sublist3r
-rwxr-xr-x 1 root root  30792 Aug  7 04:30 sucrack
-rwsr-xr-x 1 root root 161512 Oct 28 21:27 sudo
lrwxrwxrwx 1 root root      4 Oct 28 21:27 sudoedit -> sudo
-rwxr-xr-x 1 root root  64352 Oct 28 21:27 sudoreplay
-rwxr-xr-x 1 root root  43752 Aug  6 14:45 sum
-rwxr-xr-x 1 root root  60360 Nov 25  2015 sunrpcfuzz
-rwxr-xr-x 1 root root   1342 Oct 10 06:59 sushi
-rwxr-xr-x 1 root root   3120 Jun 13  2014 su-to-root
```

همانطور که در تصویر بالا قابل مشاهده می‌باشد، دسترسی مشخص شده برای فایل‌های su و sudo با فایل‌های دیگر کمی متفاوت بوده و در بخش اول از سطح دسترسی به جای حرف x از حرف s استفاده شده است. این مورد نشانگر تنظیم بودن SUID برای فایل‌های مذکور است.

این بدان معناست که هر فردی که فایل sudo را اجرا کند، از امتیازات کاربر root برخوردار است و این موضوع می‌تواند نگرانی‌های امنیتی را برای مدیران سیستم به همراه داشته و یک Attack Vector بالقوه باشد.

به عنوان مثال، برخی از برنامه‌ها برای انجام موفقیت آمیز وظایف خود نیاز به دسترسی به فایل /etc/shadow را دارند. اگر نفوذگر بتواند کنترل این برنامه را به دست بگیرد، با توجه به سطح دسترسی که این برنامه به فایل /etc/shadow دارد، می‌تواند از دسترسی آن برنامه برای دسترسی به کلمات عبور در سیستم لینوکس استفاده نماید.

در انتها توجه داشته باشید که لینوکس دارای یک سیستم امنیتی توسعه یافته است که از فایل‌ها و فولدرها در برابر دسترسی‌های غیرمجاز محافظت می‌کند. به همین منظور نفوذگران و البته متخصصان امنیت باید درک اساسی از این سیستم داشته باشند. در برخی موارد، نفوذگران می‌توانند از مجوزهای SUID و SGID برای افزایش امتیازات از یک کاربر معمولی به یک کاربر root استفاده نمایند.

فصل هفتم

مدیریت Process ها در لینوکس

یکی از مفاهیم مهم و کاربردی دیگر در سیستم‌های لینوکسی، پروسس‌ها می‌باشند. در هر زمان پروسس‌های مختلفی بر روی سیستم در حال اجرا هستند و هر سرویسی که بر روی سیستم در حال فعالیت است دارای یک پروسس می‌باشد. برای مدیریت این پروسس‌ها باید با آن‌ها آشنا شده و دستورات مربوط به پروسس‌ها را در سیستم‌عامل لینوکس به خاطر داشته باشیم.

به عنوان یک تست نفوذگر نیز برای غیرفعال نمودن برخی از ویژگی‌های امنیتی مانند فایروال، آنتی ویروس و برنامه‌های کاربردی خاص، باید با نحوه کار کردن با پروسس‌ها آشنایی لازم را داشته باشیم.

مشاهده Process ها

اولین مرحله برای مدیریت پروسس‌ها در لینوکس، چگونگی مشاهده پروسس‌های در حال اجرا بر روی سیستم است. اصلی‌ترین ابزار برای مشاهده پروسس‌ها دستور ps می‌باشد.

```
root@kali:~# ps
  PID TTY          TIME CMD
28788 pts/1    00:00:00 bash
28871 pts/1    00:00:00 ps
```

هسته یا کرنل لینوکس که تقریباً همه موارد را در لینوکس کنترل می‌کند، به هر پروسس که ایجاد می‌شود یک شناسه یا ID یکتا اختصاص می‌دهد و شما برای کار کردن با پروسس‌ها در لینوکس باید از این شناسه استفاده نمایید که به آن PIDگفته می‌شود.

در صورتی که دستور ps را به تنهایی اجرا نمایید، تنها به پروسس‌هایی که توسط کاربر جاری اجرا شده است دسترسی خواهید داشت و برای مشاهده تمامی پروسس‌های اجرا شده بر روی سیستم توسط کلیه کاربران می‌توانید از سوییچ aux استفاده نمایید.

```
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root      1329  0.0  0.0 235108   336 tty2     Sl+  Jun03   0:00 /usr/lib/gnome-disk-utility/gsd-disk-utility-notify
root      1331  0.0  0.1 995908  2896 tty2     SNl+ Jun03   0:03 /usr/lib/tracker/tracker-miner-fs
root      1409  0.0  0.1 1091068 3696 ?        Ssl  Jun03   0:07 /usr/lib/evolution/evolution-calendar-factory
root      1430  0.0  0.0 752168  1356 ?        Ssl  Jun03   0:05 /usr/lib/evolution/evolution-addressbook-factory
root      1431  0.0  0.0 164788  1476 ?        Ssl  Jun03   0:00 /usr/lib/gvfs/gvfsd-metadata
root      1523  0.0  0.0 391320    92 ?        Sl   Jun03   0:00 /usr/lib/gvfs/gvfsd-trash --spawner :1.8 /org/gtk/gvfs/exec_spaw/0
root      1546  0.0  0.0 500504     0 ?        Ssl  Jun03   0:01 /usr/lib/gnome-terminal/gnome-terminal-server
root      1553  0.0  0.0   7852   260 pts/0    Ss+  Jun03   0:00 bash
root      2462  0.0  0.0 465452   320 ?        Sl   Jun03   0:00 /usr/lib/gvfs/gvfsd-network --spawner :1.8 /org/gtk/gvfs/exec_spaw/
root      2581  0.0  0.0 595944   208 ?        Sl   Jun03   0:00 /usr/lib/gvfs/gvfsd-dnssd --spawner :1.8 /org/gtk/gvfs/exec_spaw/2
postgres  3815  0.0  0.1 208412  2692 ?        S    Nov01   3:22 /usr/lib/postgresql/11/bin/postgres -D /var/lib/postgresql/11/main
postgres  3817  0.0  0.0 208560   768 ?        Ss   Nov01   0:03 postgres: 11/main: checkpointer
postgres  3818  0.0  0.0 208412   780 ?        Ss   Nov01   1:18 postgres: 11/main: background writer
postgres  3819  0.0  0.0 208412   120 ?        Ss   Nov01   1:17 postgres: 11/main: walwriter
postgres  3820  0.0  0.1 208948  2144 ?        Ss   Nov01   2:34 postgres: 11/main: autovacuum launcher
postgres  3821  0.0  0.0  63600  1120 ?        Ss   Nov01   2:43 postgres: 11/main: stats collector
postgres  3822  0.0  0.0 208816   960 ?        Ss   Nov01   0:05 postgres: 11/main: logical replication launcher
root      6960  0.0  0.0 200024    32 ?        Ss   Oct06   5:00 /usr/sbin/apache2 -k start
www-data  6961  0.0  0.0 199952   328 ?        S    Oct06   0:00 /usr/sbin/apache2 -k start
www-data  6962  0.0  0.0 200096   156 ?        S    Oct06   0:00 /usr/sbin/apache2 -k start
www-data  6963  0.0  0.0 200096    76 ?        S    Oct06   0:00 /usr/sbin/apache2 -k start
www-data  6964  0.0  0.0 200096   152 ?        S    Oct06   0:00 /usr/sbin/apache2 -k start
www-data  6965  0.0  0.0 200088     8 ?        S    Oct06   0:00 /usr/sbin/apache2 -k start
www-data  6966  0.0  0.0 200096     8 ?        S    Oct06   0:00 /usr/sbin/apache2 -k start
root      8113  0.0  0.0 397384   636 ?        Ssl  Sep08   0:11 /usr/lib/udisks2/udisksd
root      8822  0.0  0.0 314952  1260 ?        Ssl  Sep08   0:01 /usr/sbin/ModemManager --filter-policy=strict
root      9311  0.0  0.0 221772   868 ?        Ssl  Dec01   2:42 /usr/sbin/rsyslogd -n -iNONE
uuidd     9734  0.0  0.0   7664     0 ?        Ss   Sep08   0:00 /usr/sbin/uuidd --socket-activation
root     11118  0.0  0.0      0     0 ?        I<   Sep08   0:00 [xfsalloc]
root     11119  0.0  0.0      0     0 ?        I<   Sep08   0:00 [xfs_mru_cache]
root     11122  0.0  0.0      0     0 ?        S    Sep08   0:00 [jfsIO]
root     11123  0.0  0.0      0     0 ?        S    Sep08   0:00 [jfsCommit]
root     11124  0.0  0.0      0     0 ?        S    Sep08   0:00 [jfsCommit]
root     11125  0.0  0.0      0     0 ?        S    Sep08   0:00 [jfsSync]
root     11146  0.0  0.0   8096    84 ?        Ss   Sep08  11:22 /usr/sbin/haveged --Foreground --verbose=1 -w 1024
rtkit    11256  0.0  0.0 152644     0 ?        SNsl Sep08   1:50 /usr/libexec/rtkit-daemon
```

خروجی دستور ps aux در سیستم‌های مختلف با توجه به برنامه‌هایی که بر روی سیستم اجرا شده اند، متفاوت خواهدبود ولی ستون‌های مهمی که در خروجی نمایش داده شده و برای ما اهمیت دارند شامل موارد زیر هستند:

User: نشان دهنده کاربری است که پروسس را ایجاد نموده است.

PID: نشان دهنده شناسه پروسس می‌باشد.

%CPU: نمایانگر درصدی از CPU بوده که توسط پروسس در حال استفاده است.

%MEM: نمایانگر درصدی از حافظه بوده که توسط پروسس در حال استفاده می‌باشد.

COMMAND: نشان دهنده نام دستوری است که پروسس را آغاز کرده است.

فیلتر نمودن با نام پروسس

هنگامی که به دنبال یک پروسس خاص هستید، نمایش همه پروسس‌ها برای ما کارایی نداشته و در این حالت نیاز به فیلتر نمودن خروجی پروسس‌های سیستم می‌باشد. برای این منظور از دستور grep استفاده می‌شود. برای نمایش این بخش ابتدا دستور msfconsole را اجرا می‌نماییم تا کنسول نرم افزار متاسپلویت اجرا گردد. سپس دستور ps را مطابق تصویر زیر اجرا می‌کنیم:

```
msf5 > ps aux | grep msfconsole
[*] exec: ps aux | grep msfconsole

root     31452 17.9 10.6 678780 217168 pts/1    Sl+  03:37   0:14 ruby /usr/bin/msfconsole
root     31995  0.0  0.0   2384    700 pts/1    S+   03:38   0:00 sh -c ps aux | grep msfconsole
root     31997  0.0  0.0   6144    956 pts/1    S+   03:38   0:00 grep msfconsole
```

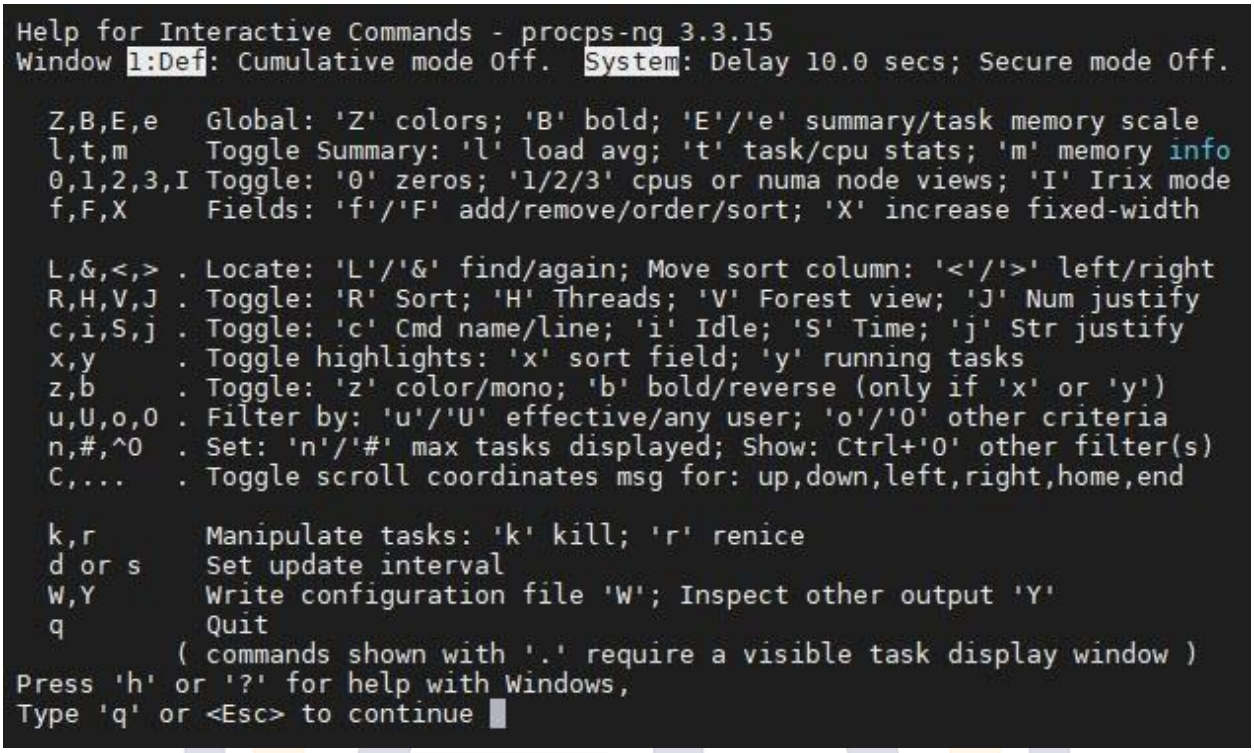
همانطور که در تصویر بالا مشخص می‌باشد، تنها پروسس‌هایی که با msfconsole مرتبط می‌باشند نمایش داده می‌شود.

یافتن پروسس های پرمصرف با دستور top

هنگامی که شما دستور ps را اجرا می‌کنید، پروسس‌ها به ترتیب ایجاد شدن در سیستم به شما نمایش داده می‌شود و در واقع بر اساس شناسه پروسس مرتب شده‌اند. اغلب ما به دنبال پروسس‌هایی هستیم که بیشترین منابع را به خود اختصاص داده‌اند. این جا زمانی است که باید از دستور top استفاده نماییم. دستور top پروسس‌ها را بر اساس میزان مصرف منابع مرتب می‌نماید و بر خلاف دستور ps که یک snapshot از پروسس‌ها را به نمایش می‌گذارد، ابزار top به صورت live بوده و لیست نمایش داده شده توسط این ابزار refresh می‌گردد.

```
root@kali:~# top
top - 03:57:14 up 206 days, 13:32,  2 users,  load average: 0.01, 0.02, 0.00
Tasks: 245 total,   1 running, 244 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.5 us,  0.7 sy,  0.0 ni, 98.8 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   1994.3 total,    302.0 free,   1486.6 used,    205.7 buff/cache
MiB Swap:   2044.0 total,    892.4 free,   1151.6 used.    326.2 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
 3820 postgres  20   0  208948   2052   1688 S   0.3   0.1   2:34.59 postgres
 6477 root      20   0    9264   3800   3072 R   0.3   0.2   0:00.04 top
28778 root      20   0   13080   6844   6232 S   0.3   0.3   0:13.58 sshd
28789 root      20   0    6584   2592   2444 S   0.3   0.1   0:17.57 bash
    1 root      20   0  169768   4780   2024 S   0.0   0.2  16:28.84 systemd
    2 root      20   0       0      0      0 S   0.0   0.0   0:04.93 kthreadd
    3 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 rcu_gp
    4 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 rcu_par_gp
    6 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/0:0H
    8 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 mm_percpu_wq
    9 root      20   0       0      0      0 S   0.0   0.0  23:14.66 ksoftirqd/0
   10 root      20   0       0      0      0 I   0.0   0.0  63:35.97 rcu_sched
   11 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_bh
   12 root      rt   0       0      0      0 S   0.0   0.0   0:47.76 migration/0
   14 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/0
   15 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/1
```

هنگامی که دستور top را اجرا نموده و پروسس‌ها نمایش داده می‌شوند، شما می‌توانید برای مشاهده سوییچ‌های کاربردی دستور top کلید ؟ را فشرده تا راهنمای این دستور برای شما نمایش داده شود:



به عنوان مثال برای مشاهده میزان حافظه مصرفی می‌توانید در هنگام اجرای دستور top کلید m را فشار دهید یا برای زمان refresh شدن صفحه از کلید s استفاده نموده و میزان زمان refresh شدن صفحه را بر اساس ثانیه وارد نمایید.

مدیریت پروسس ها

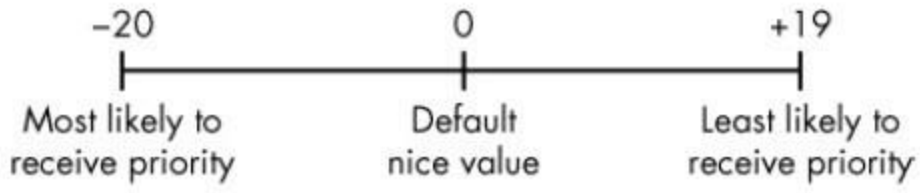
نفوذگران معمولا نیاز به چندین پروسه داشته و سیستم‌عامل کالی برای این منظور بسیار مناسب می‌باشد. یک نفوذگر ممکن است یک اسکنر پورت را اجرا نموده و همچنین قصد اجرای یک اسکنر آسیب پذیری را نیز به صورت همزمان داشته باشد.

این امر مستلزم آن است که نفوذگر این پروسس‌ها را به نحو احسن مدیریت نماید تا بتواند به بهترین نحو ممکن از منابع سیستم استفاده نماید.

تغییر اولویت پروسس ها باnice

شما ممکن است تا کنون نام ابزار nice را نشنیده باشید. ولی در حوزه امنیت و تست نفوذ معمولا از این دستور برای اولویت بندی یک پروسس به هسته استفاده می‌شود. توجه داشته باشید که هسته حرف آخر را در مورد اولیت بندی یک فرآیند می‌زند، اما شما می‌توانید با استفاده از ابزار nice برای اولویت بندی پروسس استفاده نمایید.

مقداری که برای دستور nice در نظر گرفته شده است بین منفی ۲۰ تا مثبت ۱۹ می‌باشد و مقدار آن به صورت پیش فرض صفر می‌باشد. مقدار بالای nice اشاره به اولویت پایین و مقدار پایین nice اشاره به اولویت بالا دارد. هنگامی که یک پروسس آغاز به کار می‌کند، مقدار nice آن از پروسس بالاتر آن یا Parent Process به ارث می‌رسد. صاحب یک پروسس می‌تواند اولویت پروسس را کاهش دهد ولی قادر به افزایش اولویت آن نخواهد بود. البته کاربر root می‌تواند مقدار دلخواه مورد نظر خود را برای nice تعیین نماید.



تغییر اولویت هنگام آغاز یک پروسس

هنگامی که شما یک پروسس را آغاز می‌نمایید، می‌توانید سطح اولویت آن را با دستور nice تعیین نمایید و پس از آن با استفاده از دستور renice می‌توانید اولویت یک پروسس در حال اجرا را تغییر دهید.

به عنوان مثال فرض کنید که یک پروسس با نام slowprocess داریم که در مسیر /bin/slowprocess قرار دارد. اگر قصد افزایش سرعت تکمیل این فرآیند را داشته باشیم می‌توانید از دستور nice به شکل زیر استفاده کنیم:

```
nice -n -10 /bin/slowproces
```

این دستور مقدار nice را توسط ۱۰- افزایش داده و منجر به افزایش اولویت آن و همچنین افزایش منابع بیشتر برای این پروسس می‌گردد. از طرفی دیگر اگر بخواهیم اولویت این پروسس را کاهش دهیم که برای کاربران دیگر خوشایند باشد از دستور زیر استفاده می‌کنیم:


```
nice -n 10 /bin/slowprocess
```

تغییر اولویت یک پروسس در حال اجرا باrenice

دستور renice مقادیر مطلق بین منفی ۲۰ تا مثبت ۱۹ را گرفته و اولویت را برای سطح خاص تنظیم می‌کند. علاوه بر این دستور renice به PID پروسس نیز نیاز دارد.

بنابراین اگر پروسس slowprocess از مقدار ناچیزی از منابع بر روی سیستم شما استفاده می‌کند و شما می خواهید اولویت کمتری به آن اختصاص دهید، می‌توانید از دستور زیر استفاده نمایید. در این دستور PID پروسس مذکور ۶۹۹۶ می‌باشد:

renice 20 6996

البته با این کار شما به سایر پروسس‌ها اولویت بالاتر و منابع بیشتری اختصاص می‌دهید.

همانند دستورnice ، تنها کاربر root می‌تواند مقدار یک پروسس را به عدد منفی تبدیل کند تا اولویت بالاتری داشته باشد. اما هر کاربر می‌تواند اولویت پروسس را کاهش دهد.

همچنین شما می‌توانید از ابزار top هم برای تغییر مقدار nice استفاده کنید. برای این منظور پس از اجرای ابزارtop ، می‌بایست کلید R را فشرده و سپس مقدار PID و nice مورد نظر را وارد نمایید.

Kill نمودن پروسس ها

در بعضی مواقع، یک پروسس بیش از حد منابع سیستم را مصرف می‌کند و یا رفتار غیرمعمولی را از خود نشان می‌دهد. برای متوقف نمودن یک پروسس شما می‌توانید از دستور kill استفاده نمایید. دستور kill دارای ۶۴ نوع مختلف از اصطلاحا kill signal می‌باشد و نحوه استفاده از آن به صورت زیر می‌باشد:

kill -signal pid

در صورت عدم استفاده از signal در دستور kill به صورت پیش فرض SIGTERM در نظر گرفته می‌شود که برابر - ۱۵ می‌باشد.

Signal name	Number for option	Description
SIGHUP	1	This is known as the *Hangup (HUP)* signal. It stops the designated process and restarts it with the same PID.
SIGINT	2	This is the *Interrupt (INT)* signal. It is a weak kill signal that isn't guaranteed to work, but it works in most cases.
SIGQUIT	3	This is known as the *core dump*. It terminates the process and saves the process information in memory, and then it saves this information in the current working directory to a file named *core*. (The reasons for doing this are beyond the scope of this book.)
SIGTERM	15	This is the *Termination (TERM)* signal. It is the `kill` command's default kill signal.
SIGKILL	9	This is the absolute kill signal. It forces the process to stop by sending the process's resources to a special device, */dev/null*.

با استفاده از دستور top شما می‌توانید پروسس‌هایی که منابع زیادی را به خود اختصاص داده اند شناسایی نمایید. اکثر پروسس‌هایی که در خروجی ابزار top مشاهده می‌شوند، پروسس‌های قانونی و مجاز می‌باشند ولی برخی از آن‌ها پروسس های غیرمجاز هستند که باید متوقف شوند.

اگر شما قصد ریست نمودن یک پروسس را با سیگنال HUG داشته باشیم، از سوییچ -۱ برای دستور kill استفاده می‌نماییم:

```
kill -1 6996
```

هنگامی که قصد متوقف نمودن یک پروسس به صورت مطلق را داشته باشیم از سوییچ ۹- استفاده می‌کنیم:

```
kill -9 6996
```

در صورتی که PID مربوط به یک پروسس را ندانید، شما می‌توانید از دستور killall استفاده نموده و نام پروسس را در ادامه دستور قرار دهید:

```
killall -9 zombieprocess
```

همچنین شما می‌توانید با استفاده از ابزار top هم یک پروسس را متوقف نمایید. برای این کار کافی است تا پس از اجرای دستورtop ، کلید K را فشرده و PID مربوط به پروسس مورد نظر خود را وارد نمایید.

زمانبندی برای پروسس ها

هم مدیران سیستم و هم نفوذگران، در برخی موارد نیاز به برنامه ریزی برای اجرای یک پروسس خاص در زمان مشخص را دارند.

به عنوان مثال مدیر سیستم ممکن است قصد اجرای برنامه پشتیبان گیری از سیستم را در هر شنبه ساعت ۲ صبح داشته باشد و یا ممکن است یک نفوذگر قصد اجرای یک اسکریپت را به طور منظم برای شناسایی پورت باز و یا آسیب پذیری خاص داشته باشد. در لینوکس این موارد با دستورات at و crond قابل پیاده سازی می‌باشد.

معمولا از دستور at برای برنامه ریزی یک job برای یک بار اجرا نمودن در یک نقطه از زمان آینده مورد استفاده قرار می‌گیرد و دستور crond برای برنامه ریزی کارهای روزمره، هفتگی، یا ماهانه مناسب تر می‌باشد که جزئیات مربوط به آن در بخش های بعدی از دوره آموزشی مقدمات لینوکس پرداخته خواهد شد.

جدول زیر نمونه هایی از استفاده دستور at می‌باشد:

Time format	Meaning
`at 7:20pm`	Scheduled to run at 7:20 PM on the current day
`at 7:20pm June 25`	Scheduled to run at 7:20 PM on June 25
`at noon`	Scheduled to run at noon on the current day
`at noon June 25`	Scheduled to run at noon on June 25
`at tomorrow`	Scheduled to run tomorrow

فصل هشتم

مدیریت متغیرهای محیطی کاربر

یکی از مواردی که در لینوکس حائز اهمیت می‌باشد، متغیرهای محیطی می‌باشد که درک مناسب از آن‌ها در مدیریت عملکرد بهینه و حتی مخفی سازی موثر خواهد بود. از منظر فنی دو نوع متغیر وجود دارد که شامل متغیرهای شل و محیطی می‌باشد.

متغیرهای محیطی، متغیرهای گسترده‌ای هستند که درون سیستم قرار داده شده‌اند ولی متغیرهای شل به طور معمول تنها در یک شل که در آن قرار دارید معتبر می‌باشد.

متغیرها در واقع رشته های ساده‌ای هستند که به صورت کلید و مقدار (KEY=value) تعریف شده و در مواردی که چندین مقدار وجود داشته باشد به صورت KEY=value1:value2 تعریف خواهد شد.

در محیط کالی لینوکس محیط یا Environment شما bash shell می‌باشد. هر کاربر از جمله کاربر root یک مجموعه پیش فرض از متغیرهای محیطی را دارد و شما می‌توانید مقادیر این متغیرها را تغییر دهید تا سیستم کارآمدتر شده و محیط کار خود را متناسب با نیازهای شخصی خود تنظیم نمایید.

شما می‌توانید تمامی متغیرهای محیطی را با وارد نمودن دستور env در ترمینال لینوکس مشاهده نمایید:

```
kali >env
XDG_VTNR=7
SSHAGENT_PID=922
XDG_SESSION_ID=2
XDG_GREETER_DATA_DIR=/var/lib/lightdm/data/root
GLADE_PIXMAP_PATH=:echo
TERM=xterm
SHELL=/bin/bash
--snip--
USER=root
--snip--
PATH=/usr/local/sbin :usr/local/bin:/usr/sbin:/sbin/bin
--snip--
HOME=/root
--snip--
```

نمایش تمامی متغیرهای محیطی

برای مشاهده کلیه متغیرهای محیطی، شامل متغیرهای شل، متغیرهای محلی و توابع شل، می توان از دستور set استفاده نمود. برای مشاهده خط به خط متغیرهای شما می‌توانید از دستور زیر استفاده نمایید:

```
set | more
```

فیلتر متغیرهای خاص

به منظور فیلتر نمودن یک متغیر خاص در خروجی ابزار set می‌توان از دستور grep استفاده نمود. به عنوان مثال ما به دنبال متغیر HISTSIZE هستیم. این متغیر شامل بیشترین تعداد دستوراتی است که دستور history قادر به ذخیره آن می‌باشد. این متغیر تنها تعداد دستورات را در خود ذخیره می‌نماید. برای فیلتر نمودن این متغیر از دستور زیر استفاده می‌کنیم:

```
set | grep HISTSIZE
```

تغییر مقدار متغیر برای یکSession

در این بخش به تغییر مقدار متغیر در یک Session می‌پردازیم. در این بخش نیز از متغییر HISTSIZE استفاده می‌کنیم. در برخی موارد ممکن است شما قصد ذخیره شدن دستورات وارد شده را نداشته باشید. (شاید شما می‌خواهید که هیچ گونه شواهدی بر روی سیستم باقی نگذارید)

در این مورد شما می‌توانید مقدار HISTSIZE را به صفر تغییر دهید تا سیستم دستورات وارد شده را ذخیره ننماید. بدین منظور شما کافی است تا در محیط ترمینال عبارت HISTSIZE=0 را وارد نمایید.

ایجاد متغیر با مقدار ثابت

هنگامی که شما مقدار یک متغیر محیطی را تغییر می‌دهید، این تغییر تنها در یک محیط خاص انجام می‌شود(Bash Shell Session) بدین صورت هنگامی که ترمینال بسته شود، تمامی تغییرات صورت گرفته از بین رفته و مقادیر تنظیم شده به مقادیر پیش فرض تغییر خواهد یافت.

اگر شما قصد پایدار نمودن این تغییرات را داشته باشید باید از دستور export استفاده نمایید. این دستور مقدار جدیدی از محیطی که شما در آن هستید (bash shell) به بقیه سیستم صادر نموده و تا زمان تغییر شما مجدداً آن را تغییر ندهید در تمامی محیط ها در دسترس خواهد بود.

با توجه به اینکه متغیرها در واقع از رشته‌های مختلفی تشکیل شده‌اند، پیشنهاد می‌شود جهت احتیاط، پیش از اصلاح متغیرها، محتوای آن‌ها را در یک فایل متنی ذخیره نمایید تا در صورت بروز مشکل به متغیرهای پیشین دسترسی داشته باشید.
به منظور ذخیره سازی یکی از متغیرها به صورت زیر عمل نموده و نام متغیر را بعد از دستور echo وارد می‌نماییم:


```
echo $HISTSIZE> ~/valeueofHISTSIZE.txt
```

همچنین به منظور ذخیره سازی کلیه اطلاعات مربوط به متغیرها، شما می‌توانید از دستور زیر استفاده نموده و آن‌ها را در یک فایل متنی ذخیره نمایید:

```
set> ~/valueofALL08012020.txt
```

سپس شما می‌توانید از دستور export استفاده نموده و پس از مقداردهیHISTSIZE=0 ، آن را اصطلاحاً Permanentنمایید:

export HISTSIZE

تغییر PATH جاری

یکی از مهمترین متغیرهای محیطی، متغیر PATH می‌باشد که دستورات مربوط به Shell استفاده شده توسط شما مانند cd، ls، echoو موارد دیگر را کنترل می‌کند. اغلب دستورات در زیرمجموعه bin یا sbin /usr/local/sbin یا /usr/local/binقرار گرفته‌اند. اگر Bash Shell دستور وارد شده توسط شما را در یکی از دایرکتوری‌های موجود در متغیر PATH پیدا نکند، پیام Command not found به شما نمایش داده می‌شود. حتی اگر دستور در دایرکتوری وجود داشته باشد ولی در PATH وجود نداشته باشد، باز هم پیام Command not found نمایش داده خواهد شد.

برای مشاهده دایرکتوری‌های موجود در متغیر PATH از دستور زیر استفاده می‌کنیم:

```
kali >echo $PATH
/usr/local/sbin:usr/local/bin:/usr/sbin:/sbin/bin
```

این‌ها دایرکتوری‌هایی هستند که ترمینال شما برای دستیابی به هر دستوری، آن را جست و جو می‌نماید. به عنوان مثال، هنگامی که شما دستور ls را وارد می‌کنید، سیستم در دایرکتوری‌های بالا به دنبال دستور ls می‌گردد و در صورت پیدا کردن آن، دستور مورد نظر اجرا می‌شود.

توجه داشته باشید که دایرکتوری‌ها با عبارت : از یکدیگر جدا می‌شوند.

اضافه کردن مقدار به متغیرPATH

در صورتی که شما یک ابزار را از اینترنت دانلود نموده و آن را نصب نمایید، دستورات مربوط به این ابزار، تنها زمانی که شما در دایرکتوری آن قرار دارید، قابل اجرا هستند. بدین ترتیب در صورتی که شما در دایرکتوری دیگری باشید، امکان اجرای این دستور را نخواهید داشت.

فرض کنید که ابزار در دایرکتوری /root/newhackingtool قرار دارد. با توجه به اینکه این مسیر در متغیر PATH تعریف نشده است، در صورتی که خارج از دایرکتوری، دستورات ابزار مورد نظر را وارد نمایید با پیام Command not found مواجه خواهید شد.

به منظور استفاده از دستورات این ابزار در کل سیستم، باید مسیر آن را به متغیر PATH اضافه نمایید. برای این منظور باید از دستور زیر استفاده کنید:

```
PATH=$PATH:/root/newhackingtool
```

پس از اجرای دستور بالا در صورتی که محتویات متغیر PATH را بررسی نمایید، دایرکتوری مورد نظر به دایرکتوری‌های موجود در این متغیر اضافه گردیده است و از این پس شما می‌توانید دستورات مربوط به این ابزار را در بخش‌های مختلف ترمینال وارد نمایید.

```
kali >echo $PATH
/usr/local/sbin:usr/local/bin:/usr/sbin:/sbin/bin:/root/newhackingtool
```

نکته:

توجه داشته باشید که گرچه اضافه کردن دایرکتوری‌های مربوط به ابزارهای مختلف در متغیر PATH امکان دسترسی به آن‌ها را برای شما آسان تر می‌کند ولی در صورتی که مقادیر مربوط به این متغیر زیاد باشد، هنگام وارد نمودن دستورات در محیط ترمینال، این مسیرها برای پیدا کردن دستور وارد شده جست و جو می‌شوند و در صورت زیاد بودن این مسیرها، سرعت اجرای دستورات در ترمینال شما کاهش خواهد یافت.

یک اشتباه در اضافه کردن متغیرPATH

یکی از اشتباهاتی که برای کاربران در ابتدای کار با متغیر PATHممکن است رخ دهد، استفاده از دستور زیر برای اضافه نمودن دایرکتوری /root/newhackingtool می‌باشد:

```
PATH=/root/newhackingtool
```

در صورت استفاده از دستور بالا، مقادیر تنظیم شده پیشین متغیر PATH حذف شده و تنها مقدار دایرکتوری بالا درون این متغیر قرار می‌گیرند. بدین صورت تنها دستوارت موجود در این دایرکتوری اجرا شده و دستورات دیگر، اجرا نخواهند

شد. به عنوان مثال زمانی که شما دستور ls را در محیط ترمینال خود وارد می‌کنید با پیام Command not found مواجه خواهید شد.

فصل نهم

نکاتی در خصوص نوشتن اسکریپت

یکی از تکنیک‌هایی که تست نفوذگران باید به آن آگاهی داشته و در پروژه های خود از آن استفاده نمایند، نوشتن اسکریپت‌های کاربردی می‌باشد. آن‌ها در واقع با این کار، برخی از فرآیندهای تست نفوذ را به صورت خودکار انجام می‌دهند و در وقت خود صرفه جویی می نمایند.

در لینوکس قابلیت اسکریپت نویسی با استفاده از شل، امکان پذیر می‌باشد که در ادامه به نحوه انجام آن و برخی از دستورات کاربردی شل اسکریپت نویسی در لینوکس می‌پردازیم.

Shell یک واسط بین کاربر و سیستم‌عامل است که شما را قادر می‌سازد تا با آن تعامل نموده و فعالیت‌های مختلفی مانند اجرای دستورات، ابزارها، برنامه و موارد مشابه را انجام دهند. در لینوکس Shell های مختلفی در دسترس هستند که Z shell، Korn shell، C shellنمونه ای از این Shell ها هستند.

لازم به ذکر است که Bash به عنوان شناخته ترین نوع Shell ها در لینوکس است که به صورت گسترده‌ای مورد استفاده قرار می‌گیرد.

شما در بخش‌های پیشین با دستورات مختلفی مانندpwd ، set، cdو دستوارت دیگر آشنا شدید. در این بخش ما برای نوشتن اسکریپت‌های خود به دو دستور کاربردی نیاز داریم.

دستور اول echo می‌باشد که از آن برای نمایش پیام‌ها در خروجی استفاده می‌نماییم.

دستور دوم read می‌باشد که از آن برای دریافت اطلاعات از ورودی استفاده می‌نماییم.

برای نوشتن اسکریپت‌های خود شما نیاز به یک ویرایشگر متنی دارید. برای این منظور شما می‌توانید از ابزارهایvi ، vim، gedit، kate، nanoو leafpad استفاده نمایید.

نوشتن اولین اسکریپت در لینوکس

برای شل اسکریپت نویسی در لینوکس باید ساختار دستوری آن را رعایت نمود که در ادامه به بررسی این موارد می‌پردازیم.

در اولین اسکریپت ما قصد داریم تا تنها عبارت "Hello you are a Pentester!" را در خروجی چاپ نماییم. برای شروع شما باید به سیستم‌عامل بگویید که interpreter شما می‌خواهد برای اسکریپت استفاده شود. برای این منظور شما باید یک اصطلاحاً Shebang را وارد کنید که یک شارپ به همراه یک علامت تعجب است(!#) .

پس از آن شما باید نام شل مورد نظر خود را وارد کنید که در این جا ما از Bash استفاده می‌کنیم که باید مسیر آن را به صورت /bin/bash در ادامه Shebang وارد کنیم. با این کار ما به سیستم‌عامل می‌گوییم که قصد استفاده از Bash Shell Interpreter را داریم.

در خط بعدی از اسکریپت ما از دستور echo برای چاپ یک پیام در خروجی استفاده می‌کنیم.

```
#! /bin/bash

# This is my first bash script.

echo "Hello you are a Pentester!"
```

همانطور که در تصویر بالا قابل مشاهده می‌باشد، شما می‌توانید توضیحاتی را نیز در اسکریپت خود اضافه نمایید. برای این منظور شما می‌توانید با اضافه نمودن عبارت شارپ در ابتدای خط، هر توضیحی را به اسکریپت خود اضافه نمایید.

در ادامه باید اسکریپت را با نام دلخواه خود ذخیره نموده و سطح دسترسی اجرا یا Execute را به آن اعمال نماییم تا امکان اجرای اسکریپت وجود داشته باشد. برای این منظور از دستور chmod به صورت زیر استفاده می‌نماییم:

```
chmod 755 firstscript
```

اجرای اسکریپت

برای اجرای اسکریپت که در دایرکتوری جاری وجود دارد، کافی است تا شما نقطه و اسلش را به ابتدای نام اسکریپت اضافه نموده و آن را اجرا نمایید:

```
root@kali:~# ./FirstScript
Hello you are a Pentester!
```

اضافه نمودن برخی قابلیت ها به اسکریپت

تا به حال شما یک اسکریپت ساده که تنها مقداری را در خروجی چاپ می‌کند نوشته‌اید. در ادامه با استفاده از عبارت read و همچنین به کارگیری متغیرها، اسکریپت را کمی تعاملی تر می‌کنیم.

برای نوشتن اسکریپت، یک فایل جدید ایجاد نموده و آن را با یک ویرایشگر متنی باز می‌کنیم. مطابق تصویر زیر، کدها را درون آن وارد می‌نماییم:

```
#! /bin/bash

# This is Second bash script

echo "What is your name?"

read name

echo "What is your course?"

read course

echo "Welcome "$name" to "$course" Course in SecurityWorld.ir"
```

عبارتread ، مقداری را از ورودی دریافت و در متغیری که با یک نام در جلوی آن مشخص می‌شود، قرار می‌دهد. سپس در انتها نیز این متغیرها را در متن خوش آمدگویی انتهای اسکریپت قرار داده و به کاربر نمایش می‌دهد.

پس از ذخیره سازی اسکریپت و تغییر سطح دسترسی آن مشابه اسکریپت پیشین برای اجرای اسکریپت، شما می‌توانید آن را اجرا نمایید. پس از اجرا عبارت What is your name چاپ شده و از شما یک ورودی درخواست می‌کند و پس از وارد کردن نام و فشردن کلید Enter عبارت What is your course چاپ شده و از شما نام دوره را درخواست می‌کند.

پس از ورود نام دوره و فشردن کلیدEnter ، عبارت خوش آمد گویی برای شما چاپ شده و نام و دوره را به شما نمایش خواهد داد:

```
root@kali:~# ./SecondScript
What is your name?
Ehsan
What is your course?
Linux
Welcome Ehsan to Linux Course in SecurityWorld.ir
```

یک اسکریپت برای اسکن پورت

پیش از اینکه اقدام به نوشتن این اسکریپت نماییم، ابتدا باید با یک ابزار در زمینه تست نفوذ با نام Nmap آشنا شویم. از ابزار Nmap برای اسکن شبکه و پورت استفاده می‌شود Nmap .به صورت پیش فرض در سیستم‌عامل کالی لینوکس نصب شده است.

در ابزار Nmap برای اسکن از ساختار زیر استفاده می‌شود:

```
nmap type-of-scan target-ip target-port
```

در این مثال از دستورsT- برای اسکن از نوع TCP استفاده می‌کنیم. در این مثال پورت ۳۳۰۶ که مربوط به سرویس mysql می‌باشد استفاده می‌نماییم:

```
nmap -sT ip -p 3306
```

برای ساختن اسکریپت یک فایل با نام ScannerMysql ساخته و کدهای زیر را در آن وارد می‌کنیم:

```
#! /bin/bash

# This script is designed to find hosts with MySQL Installed

nmap -sT 50.87.99.0/24 -p 3306 >/dev/null -oG mysqlscan

cat mysqlscan | grep open > mysqlscan2

cat mysqlscan2
```

پس از توضیحات از دستور nmap استفاده می‌کنیم و پورت ۳۳۰۶ را در رنج مورد نظر اسکن می‌کنیم و از /dev/null استفاده می‌کنیم تا خروجی Nmap به ما نمایش داده نشود. همچنین خروجی را با استفاده از سوییچ oG- که برای ذخیره به صورت Grapable است در یک فایل دیگر ذخیره می‌نماییم. خروجی Grapable به ما این امکان را می‌دهد تا از دستور grep برای جست و جو در آن استفاده نماییم.

سپس در خط بعدی فایل ایجاد شده را با دستور cat خوانده و به دنبال بخشی که عبارت open نوشته شده است می‌گردیم و با استفاده از دستور grep آن را به یک فایل دیگر منتقل می‌کنیم.

در انتها نیز فایل نهایی را با دستور cat چاپ می‌نماییم:

```
root@kali:/dev# ./ScannerMysql
Host: 50.87.99.12 (50-87-99-12.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.22 (50-87-99-22.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.30 (50-87-99-30.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.37 (50-87-99-37.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.38 (50-87-99-38.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.42 (50-87-99-42.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.48 (50-87-99-48.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.51 (50-87-99-51.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.52 (50-87-99-52.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.58 (50-87-99-58.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.74 (50-87-99-74.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.76 (50-87-99-76.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.78 (50-87-99-78.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.79 (50-87-99-79.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.80 (50-87-99-80.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.91 (50-87-99-91.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.96 (50-87-99-96.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.101 (50-87-99-101.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.106 (50-87-99-106.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.108 (50-87-99-108.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.111 (50-87-99-111.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.127 (50-87-99-127.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.130 (50-87-99-130.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.135 (50-87-99-135.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.138 (50-87-99-138.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.140 (50-87-99-140.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.142 (50-87-99-142.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.143 (50-87-99-143.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.149 (50-87-99-149.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
```

بهینه سازی اسکریپتMySQL

برای بهبود بخشیدن کارایی اسکریپتی که ایجاد نموده‌ایم، می‌توانیم پارامترهای این اسکریپت را از کاربر دریافت نماییم.

برای این منظور اسکریپت را مطابق با تصویر زیر تغییر می‌دهیم:

```
#! /bin/bash

# This script is designed to find hosts with MySQL Installed

echo "Enter the starting IP address : "

read FirstIP

echo "Enter the last octet of last IP address : "

read LastOctetIP

echo "Enter the port number you want to scan for : "

read port

nmap -sT $FirstIP-$LastOctetIP -p $port >/dev/null -oG mysqlscan

cat mysqlscan | grep open > mysqlscan2

cat mysqlscan2
```

با اجرای این اسکریپت، آدرس IP ابتدایی یا شروع آدرس را سوال می‌نماید. سپس بخش انتهایی آدرس را سوال می‌کند و در ادامه نیز شماره پورت مورد نظر را دریافت می‌کند:

```
root@kali:~# ./ScannerMysql
Enter the starting IP address :
50.87.99.0
Enter the last octet of last IP address :
255
Enter the port number you want to scan for :
3306
Host: 50.87.99.12 (50-87-99-12.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.22 (50-87-99-22.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.30 (50-87-99-30.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.37 (50-87-99-37.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.38 (50-87-99-38.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.42 (50-87-99-42.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.48 (50-87-99-48.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.51 (50-87-99-51.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.52 (50-87-99-52.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.58 (50-87-99-58.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.74 (50-87-99-74.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.76 (50-87-99-76.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.78 (50-87-99-78.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.79 (50-87-99-79.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.80 (50-87-99-80.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.91 (50-87-99-91.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.96 (50-87-99-96.unifiedlayer.com)        Ports: 3306/open/tcp//mysql///
Host: 50.87.99.101 (50-87-99-101.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.106 (50-87-99-106.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.108 (50-87-99-108.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.111 (50-87-99-111.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
Host: 50.87.99.127 (50-87-99-127.unifiedlayer.com)      Ports: 3306/open/tcp//mysql///
```

همچنین شما می‌توانید با کمی تغییر در اسکریپت، آدرس دهی آن را بر اساس /۲۴ یا موارد مشابه دریافت نمایید.

توضیح کوتاهی در خصوص/dev

لینوکس دستگاه‌های جانبی و غیر جانبی را به صورت فایل در نظر می‌گیرد و نمایش می‌دهد(یعنی به ازای هر دستگاه، یک فایل معادل آن در نظر می‌گیرد) و دایرکتوری dev حاوی تعدادی فایل‌های ویژه یا special file یا فایل دستگاهی می‌باشد که این فایل‌ها در حقیقت نماینده دستگاه‌ها هستند یعنی لینوکس هر دستگاه را با فایلی در نظر می‌گیرد و اطلاعات مربوط به آن دستگاه را در فایل مربوط به آن ذخیره می‌کند.

در حقیقت فایل‌های دستگاهی فایل به معنایی که ما با آن آشناییم نیستند اما این فایل‌ها به صورت فایل‌های معمولی نمایش داده می‌شوند به عنوان مثال /dev/sda مربوط به اولین درایو SATA در سیستم است. اگر بخواهیم این درایو را پارتیشن بندی کنیم، می‌توانیم برای یک برنامه پارتیشن بندی فایل /dev/sda را مشخص کنیم تا پارتیشن بندی را آغاز کند.

این دایرکتوری علاوه بر فایل دستگاهی حاوی شبه دستگاه‌ها نیز می‌باشد. شبه دستگاه‌ها، دستگاه‌هایی مجازی هستند که به ازای آن‌ها سخت‌افزاری وجود ندارد. به عنوان مثال dev/random /اعداد تصادفی تولید می‌کند.

/dev/null یک دستگاه مجازی است که خروجی‌ای ندارد و ورودی‌هایی که به آن داده می‌شود را در نظر نمی‌گیرد و دور می‌اندازد (یکی از استفاده‌های آن این است که زمانی که می‌خواهیم یک خروجی نمایش داده نشود یعنی می‌خواهیم خروجی دور ریخته شود) مثلا پیام error یا خطایی که رخ می‌دهدمی‌توانیم آن را به این دایرکتوری پایپ کنیم.

```
echo “we are here” | /dev/null
```

منبع بخش انتهایی :**ویکی بوک**

فصل دهم

فشرده سازی در لینوکس

تست نفوذگران اغلب نیاز به دانلود و نصب نرم افزارهای جدید دارند که این ابزارها یا اسکریپت ها معمولاً به صورت فایل‌های فشرده در دسترسی هستند. در این بخش، شما با دستوراتی که به شما در فشرده سازی و از حالت فشرده خارج نمودن فایل‌ها کمک می‌کند، آشنا خواهید شد.

البته موضوع فشرده سازی، دارای بخش‌های مختلفی است که موارد مطرح شده برای آن می‌تواند در قالب یک کتاب عرض شود. در این بخش ما تنها به آشنایی با دستورات مربوط به آن می‌پردازیم.

معمولا اولین کاری که شما در هنگام فشرده سازی پرونده‌ها انجام می‌دهید، ترکیب آن‌ها به یک آرشیو می‌باشد. یکی از دستوراتی که برای این منظور استفاده می‌شود، دستور tar می‌باشد.

با استفاده از دستور tar شما می‌توانید چندین فایل را در قالب یک فایل فشرده نمایید:

```
tar -cvf test.tar file1 file2 file3 ...
```

در دستور بالا، سوییچ c به معنای Create یا ایجاد بوده، سوییچ v به معنای verbose بوده و فایل‌هایی که به tar تبدیل می‌شوند را نمایش می‌دهد و سوییچ f نیز به معنای ذخیره داخل یک فایل می‌باشد.

بدین صورت ما یک فایل نهایی با نام test.tar خواهیم داشت.

ما می‌توانیم برای مشاهده فایل‌های درون test.tar بدون Extract آن، از سوییچ t- به صورت زیر استفاده نماییم:


```
tar -tvf test.tar
```

برای Extract نمودن این فایل نیز می‌توان از دستور زیر استفاده نمود:

```
tar -xvf test.tar
```

لازم به ذکر است برای عدم مشاهده جزئیات در خروجی شما می‌تواند سوییچ v را حذف نمایید.

فشرده سازی فایل

لینوکس برای فشرده سازی فایل‌ها دارای دستورات مختلفی می‌باشد که نمونه ای از آن‌ها عبارتند از:

gzip که از پسوندهای tar.gz یا gz استفاده می‌کند.
bzip2 که از پسوند tar.bz2 استفاده می‌کند.
compress که از پسوند tar.z استفاده می‌کند.

تمامی دستورات بالا به منظور فشرده سازی فایل‌های شما مورد استفاده قرار می‌گیرند ولی هر کدام از آن‌ها دارای الگوریتم‌های مخصوص به خود برای فشرده سازی فایل‌ها هستند.

به صورت کلی، دستور compress سریعتر عمل می‌کند ولی نتیجه آن یک فایل بزرگتر است؛ دستور bzip2 آهسته تر عمل می‌کند ولی نتیجه آن یک فایل با حجم کمتر است؛ دستور gzip هم بین این دو عمل می‌کند.

فشرده سازی باgzip

دستور gzip برگرفته از GNU zip می‌باشد. این دستور یکی از عمومی ترین دستورات برای فشرده سازی در لینوکس می‌باشد. شما می‌توانید برای فشرده سازی فایل ها باgzip ، از دستور زیر استفاده نمایید:

```
gzip test.*
```

توجه داشته باشید که ما در این دستور از عبارت wildcard ستاره برای فشرده سازی تمامی فایل‌هایی که با نام test شروع شده و مختلفی دارند، استفاده نموده‌ایم.

جهت خارج نمودن فایل مورد نظر از حالت فشرده، از دستور gunzip که در ادامه آن نام فایل قرار گرفته است، استفاده می‌نماییم.

فشرده سازی باbzip۲

دستور دیگری که برای فشرده سازی از آن استفاده می‌شود، دستور bzip2 می‌باشد. این دستور عملکردی مشابه gzip دارد ولی عملیات فشرده سازی را بهتر انجام می‌دهد. بدین معنی که فایل نهایی دارای سایز کوچکتری می‌باشد.

bzip2 test.*

برای خارج نمودن فایل از حالت فشرده نیز از دستور bunzip2 استفاده می‌شود.

فشرده سازی باcompress

علاوه بر دستورات gzip وbzip2 ، دستور compress نیز برای فشرده سازی فایل‌ها در لینوکس مورد استفاده قرار می‌گیرد. برای فشرده سازی فایل‌ها از دستور compress و برای خارج نمودن فایل‌ها از حالت فشرده از دستور uncompress استفاده می‌شود.

ایجاد یک کپی فیزیکی یا بیت به بیت

یکی از دستوراتی که در امنیت اطلاعات و حتی جرم شناسی دارای کاربردهای فراوانی است، دستور dd می‌باشد. دستور ddیک کپی بیت به بیت از یک فایل، فایل سیستم و یا محتویات فایل تهیه می‌کند. این بدین معنی است که حتی فایل‌های حذف شده نیز کپی می‌شوند (بله این مهم است که بدانید فایل‌های حذف شده شما قابلیت بازگردانی دارند). توجه داشته باشید که فایل‌های حذف شده با ابزارهایی مانند cp و غیره کپی نمی‌شوند.

گرچه دستور dd بسیار کاربردی بوده ولی باید توجه داشته باشید که شما نباید از آن به صورت یک ابزار برای تهیه کپی یا پشتیبان به صورت روزانه استفاده نمایید. زیرا این ابزار با توجه به قابلیتی که دارد، بسیار آهسته عمل می‌کند و دارای سرعت پایین تری از ابزارهای دیگر می‌باشد.

ساختار کلی دستور dd به صورت زیر می‌باشد:

```
dd if=inputefile of=outputfile
```


دستور زیر نمونه‌ای از استفاده ابزار dd می‌باشد:

```
dd if=/dev/sdb of=/root/flashcopy
```

دستور dd دارای سوییچ‌های مختلفی می‌باشد که دو مورد از آن‌ها، سوییچ noerror و سوییچ bs می‌باشند. همانطور که از نام سوییچ noerror مشخص می‌باشد، از آن به منظور تهیه کپی حتی در شرایطی که خطایی رخ دهد مورد استفاده قرار می‌گیرد.

سوییچ bs برای تعیین بلاک های داده استفاده می‌شود که به صورت پیش فرض ۵۱۲ بایت برای آن در نظر گرفته شده است که شما می‌توانید با این سوییچ، مقدار پیش فرض تعیین شده را تغییر دهید.

دستور زیر نمونه‌ای از استفاده دستور dd با سوییچ‌های مذکور می‌باشد:

```
dd if=/dev/media of=/root/flashcopy bs=4096 conv:noerror
```

فصل یازدهم

مدیریت فایل سیستم و دستگاه ذخیره ساز

اتصال دستگاه ها و ساختار فایل سیستم در ویندوز و لینوکس با یکدیگر متفاوت می‌باشد. یکی از مباحثی مهم در مدیریت فایل سیستم در لینوکس که در زمینه فایل سیستم و دستگاه‌های ذخیره ساز مورد توجه قرار می‌گیرد، مبحث Mount نمودن در لینوکس می‌باشد. در یک توضیح ساده، Mounting به معنی متصل شدن درایوها یا دیسک ها به فایل سیستم جهت استفاده در سیستم‌عامل می‌باشد.

به عنوان یک تست نفوذگر درک چگونگی مدیریت فایل سیستم در لینوکس و دستگاه‌ها ذخیره ساز برای بهره برداری از سیستم هدف یا حتی سیستم خود (کالی لینوکس) حائز اهمیت می‌باشد.

تست نفوذگران اغلب از یک دستگاه خارجی مانند هارد اکسترنال یا فلش مموری برای بارگذاری داده‌ها، ابزارهای تست و یا حتی سیستم‌عامل های دیگر استفاده می‌نمایند.

دایرکتوری dev

در سیستم‌عامل لینوکس یک دایرکتوری خاص وجود دارد که شامل فایل‌هایی است برای نمایش هر دستگاهی که به این سیستم‌عامل متصل شده است. نام این دایرکتوری dev می‌باشد. تصویر زیر برخی از محتویات این دایرکتوری را نمایش می‌دهد:

```
root@kali:/dev# ls -l
total 0
crw-------  1 root       root     10, 175 Feb  4 03:49 agpgart
crw-r--r--  1 root       root     10, 235 Feb  4 03:49 autofs
drwxr-xr-x  2 root       root         360 Feb  4 03:46 block
drwxr-xr-x  2 root       root          80 Jun  3  2019 bsg
crw-rw----  1 root       disk     10, 234 Feb  4 03:49 btrfs-control
lrwxrwxrwx  1 root       root           3 Feb  4 03:49 cdrom -> sr0
lrwxrwxrwx  1 root       root           3 Feb  4 03:49 cdrw -> sr0
drwxr-xr-x  2 root       root        2780 Feb  4 03:46 char
crw-------  1 root       root      5,   1 Feb  4 03:49 console
lrwxrwxrwx  1 root       root          11 Jun  3  2019 core -> /proc/kcore
drwxr-xr-x  4 root       root          80 Sep  8 03:13 cpu
crw-------  1 root       root     10,  62 Feb  4 03:49 cpu_dma_latency
crw-------  1 root       root     10, 203 Jun  3  2019 cuse
drwxr-xr-x  6 root       root         120 Jun  3  2019 disk
brw-rw----  1 root       disk    254,   0 Feb  4 03:49 dm-0
brw-rw----  1 root       disk    254,   1 Feb  4 03:49 dm-1
drwxr-xr-x  3 root       root         100 Jun  3  2019 dri
lrwxrwxrwx  1 root       root           3 Feb  4 03:49 dvd -> sr0
crw-rw----  1 root       video    29,   0 Feb  4 03:49 fb0
lrwxrwxrwx  1 root       root          13 Jun  3  2019 fd -> /proc/self/fd
brw-rw----  1 root       disk      2,   0 Feb  4 03:49 fd0
crw-rw-rw-  1 root       root      1,   7 Feb  4 03:49 full
crw-rw-rw-  1 root       root     10, 229 Feb  4 03:49 fuse
crw-------  1 root       root     10, 228 Feb  4 03:49 hpet
drwxr-xr-x  2 root       root           0 Jun  3  2019 hugepages
lrwxrwxrwx  1 root       root          12 Jun  3  2019 initctl -> /run/initctl
drwxr-xr-x  3 root       root         220 Jun  3  2019 input
drwxr-xr-x  2 root       root          80 Jun  3  2019 kali-vg
crw-r--r--  1 root       root      1,  11 Feb  4 03:49 kmsg
lrwxrwxrwx  1 root       root          28 Jun  3  2019 log -> /run/systemd/journal/dev-log
brw-rw----  1 root       disk      7,   0 Feb 18 19:09 loop0
brw-rw----  1 root       disk      7,   1 Feb  4 03:51 loop1
brw-rw----  1 root       disk      7,   2 Feb 10 16:53 loop2
```

به صورت پیش فرض، دستگاه‌ها در این دایرکتوری به ترتیب حروف الفبا نمایش داده می‌شوند. شما ممکن است برخی از این دستگاه‌ها مانندcpu ، cdrom یا موارد مشابه را بشناسید ولی برخی دیگر دارای نام‌های متفاوت و نسبتاً رمزنگاری‌شده‌ای هستند.

هر دستگاه بر روی سیستم‌عامل بوسیله یک فایل در دایرکتوری dev نمایش داده می‌شوند، از جمله دستگاه‌هایی که شاید قبلاً از آن‌ها استفاده نکرده باشید یا حتی متوجه آن‌ها نشده‌اید.

اگر محتویات دایرکتوری dev را بیشتر مورد بررسی قرار دهید، شما دستگاه‌هایی مانندsda1 ، sda2، sda3 ، sdbو sdb1 را مشاهده می‌کنید که این‌ها هارد درایوها و پارتیشن‌های مربوط به آن‌ها بوده و یا مربوط به USB فلش‌ها و پارتیشن‌های مربوط به آن‌ها می‌باشند.

```
brw-rw----  1 root       disk      8,   0 Feb  4 03:49 sda
brw-rw----  1 root       disk      8,   1 Feb  4 03:49 sda1
brw-rw----  1 root       disk      8,   2 Feb  4 03:49 sda2
brw-rw----  1 root       disk      8,   5 Feb  4 03:49 sda5
```

نمایش دستگاه‌های ذخیره ساز در لینوکس

لینوکس از برچسب‌های منطقی برای درایوها استفاده می‌کند که بر روی فایل سیستم Mount شده‌اند. این برچسب‌های منطقی بسته به مکانی که Mount شده‌اند، متفاوت خواهند بود، بدین معنی که ممکن است یک هارد دیسک برچسب‌های مختلف را در زمان‌های مختلف داشته باشد، این اختصاص نام، بسته به زمان و مکان Mount شدن آن می‌باشد.

پیش از این و در ابتدا لینوکس درایوهای مربوط به فلاپی که را با fd0 و هارد دیسک‌ها را با hda نمایش می‌داد. البته شما هنوز هم گاهی اوقات می‌توان این نوع از نمایش را در برخی توزیع‌های لینوکس مشاهده نمود.(البته اگر فلاپی درایو هنوز وجود داشته باشد).

با این وجود، هارد دیسک‌هایی که از رابط‌های IDE یا EIDE استفاده می‌کنند، هنوز هم در قالب‌های hda نمایش داده می‌شوند. درایوهای با رابط SATA و SCSI به عنوان sda نمایش داده می‌شوند.

بعضی اوقات درایوها به بخش‌هایی به عنوان پارتیشن تقسیم می‌شوند که با اعداد در انتهای اسامی آن‌ها مشخص می‌شوند. هنگامی که سیستم‌ها بیش از یک هارد دیسک داشته باشند، لینوکس با افزودن یک حرف به انتهای نام درایو که به ترتیب حروف الفبا می‌باشد، آن را به صورت سریالی نامگذاری می‌کند.

بنابراین درایو اول با نام sda بوده و درایو دوم با نام sdb و درایو سوم با نام sdc نمایش داده می‌شوند.

Device file	Description
sda	First SATA hard drive
sdb	Second SATA hard drive
sdc	Third SATA hard drive
sdd	Fourth SATA hard drive

آشنایی با Drive Partitions

به منظور مدیریت و جدا نمودن اطلاعات، می‌توان درایوها را به بخش‌های دیگر تقسیم بندی نمود. همانطور که در بخش پیشین نیز اشاره گردید، لینوکس هر پارتیشن را با اعداد که در انتهای نام درایو قرار می گیرد، برچسب گذاری می‌کند.

بدین ترتیب، اولین پارتیشن از اولین درایوSATA ، برابر با sda1 خواهد بود.

Partition	Description
sda1	The first partition (1) on the first (a) SATA drive
sda2	The second (2) partition on the first (a) drive
sda3	The third (3) partition on the first (a) drive
sda4	The fourth (4) partition on the first (a) drive

در صورتی که بخواهید پارتیشن‌های موجود در سیستم لینوکس را مشاهده کنید و ظرفیت آن را ببینید، می‌توانید از دستور fdisk استفاده نمایید. با استفاده از سوییچ l- در ابزارfdisk ، شما می‌توانید تمام پارتیشن‌های مربوط به درایوها را مشاهده نمایید.

```
root@kali:~# fdisk -l
Disk /dev/sda: 50 GiB, 53687091200 bytes, 104857600 sectors
Disk model: Virtual disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xec2a3acb

Device     Boot  Start       End   Sectors  Size Id Type
/dev/sda1  *      2048    499711    497664  243M 83 Linux
/dev/sda2       501758 104855551 104353794 49.8G  5 Extended
/dev/sda5       501760 104855551 104353792 49.8G 8e Linux LVM
```

آشنایی با Character and Block Devices

نکته دیگر در مورد نامگذاری فایل‌ها در دایرکتوریdev ، حروف b و c در ابتدای بخش مربوط به سطح دسترسی آن، می‌باشد. این حروف بیانگر دو راهی است که دستگاه‌ها، داده‌ها را به داخل و خارج انتقال می‌دهند.

حرف c مخفف Character بوده و دستگاه‌هایی که به عنوان Character Device شناخته می‌شوند را نمایش می‌دهد.

به تعریف ساده، دستگاه‌هایی مثل Keyboard ، پورت‌های Serial ، پرینترها و ... که کارکردشان و نحوه انتقال داده در آن‌ها بصورت کاراکتر به کاراکتر است Character Device گفته می‌شود. Character Device ها بصورت کلی به دستگاه هایی گفته می‌شود که می‌توانند Stream هایی از Character را بخوانند و یا بنویسند.

حرف b مخفف Block بوده و دستگاه‌هایی که به عنوان Block Device شناخته می‌شوند را نمایش می‌دهد.

به دستگاه‌هایی که در حوزه ذخیره سازی اطلاعات کار می‌کنند Block Device گفته می‌شود. اگر بخواهیم مثالی از Block Deviceها بزنیم می‌توانیم به هارد دیسک‌ها، فلش درایوها، نوارهای مغناطیسی و از این قبیل دستگاه‌ها اشاره کنیم. Block Device ها بصورت کلی به دستگاه‌هایی گفته می‌شود که بوسیله استفاده از Buffer های بلوک بلوک شده File system از Buffer cache که توسط کرنل سیستم‌عامل تامین می‌شود می توانند عملیات I/O انجام دهند.

این چنین دستگاه‌ها نیاز به توان داده با سرعت بالا دارند و بنابراین داده‌ها را در بلاک‌ها ارسال و دریافت می‌کنند.(تعداد زیادی کاراکتر یا بایت به طور همزمان)

لیست نمودن Block Devices and Information with lsblk

دستور lsdlk که کوتاه شده list block بوده و برای لیست نمودن اطلاعات مربوط به Block Device ها در دایرکتوری devمی‌باشد.

نتیجه اجرای این دستور مشابه با خروجی دستور fdisk -l است ولی با این تفاوت که در خروجی دستورlsdlk ، پارتیشن‌های مربوط به هر هارد نیز نمایش داده می‌شوند. لازم به ذکر است که این دستور برای اجرا، نیاز به دسترسی rootندارد.

```
root@kali:~# lsblk
NAME                  MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
fd0                     2:0    1     4K  0 disk
loop1                   7:1    0 254.7M  1 loop /snap/rocketchat-server/1420
loop2                   7:2    0  91.3M  1 loop /snap/core/8592
loop3                   7:3    0 254.7M  1 loop /snap/rocketchat-server/1422
loop4                   7:4    0  91.4M  1 loop /snap/core/8689
sda                     8:0    0    50G  0 disk
├─sda1                  8:1    0   243M  0 part /boot
├─sda2                  8:2    0     1K  0 part
└─sda5                  8:5    0  49.8G  0 part
  ├─kali--vg-root     254:0    0  47.7G  0 lvm  /
  └─kali--vg-swap_1   254:1    0     2G  0 lvm  [SWAP]
sr0                    11:0    1  1024M  0 rom
```

همانطور که در تصویر بالا نیز قابل مشاهده می‌باشد، خروجی دستور lsblk فلاپی درایو را با نام fd0 و DVD درایو را با نام sr0 نیز نمایش می‌دهد.

Mount نمودن و Unmounting

اغلب سیستم‌عامل‌های جدید، اقدام به Mount نمودن خودکار Storage Device ها، هنگام اتصال به سیستم‌عامل می‌نمایند. با این قابلیت، شما هنگامی که یک فلش درایو یا هارد دیسک را به سیستم متصل می‌نمایید، این دستگاه‌ها به صورت خودکار، به فایل سیستم متصل شده و امکان دسترسی به آن میسر خواهد بود.

بخشی از دایرکتوری که دستگاه به آن متصل می‌شود به عنوان Mount Point شناخته می‌شود. دو Mount Point اصلی در لینوکس /mnt و /media می‌باشند.

به عنوان یک قاعده کلی، هارد دیسک‌های داخلی در بخش /mnt اصطلاحاً Mount شده و دستگاه‌های خارجی مانند فلش درایوها و هارد دیسک‌های خارجی، در بخش/media ، Mount می‌شوند.

Mount نمودن دستگاه های ذخیره ساز خودتان

در برخی از نسخه‌های لینوکس شما نیاز خواهد داشت تا یک درایو را برای دسترسی به اطلاعات آن، به صورت دستی Mountنمایید. به منظور Mount نمودن یک درایو به فایل سیستم، از دستور mount استفاده می‌گردد. جهت تعیین Mount Point برای دستگاه، باید یک دایرکتوری خالی را انتخاب نمایید.

برای Mount نمودن یک هارد دیسک جدید (sdb1) در دایرکتوری /mnt شما می‌توانید از دستور زیر استفاده نمایید:

```
mount /dev/sdb1 /mnt
```


اگر شما قصد Mount نمودن یک فلش درایو (sdc1) را در دایرکتوری /media دارید می‌توانید از دستور زیر استفاده نمایید:

```
mount /dev/sdc1 /media
```

فایل سیستم‌های Mount شده در یک سیستم در فایل /etc/fstab نگهداری می‌شوند که توسط سیستم در هربار راه اندازی مجدد خوانده می‌شود .fstab مخفف FileSystem Table می‌باشد.

Umount نمودن با دستور unmount

اگر شما از سیستم‌عامل‌های ویندوز و یا Mac استفاده کرده باشید، شما در حال انجام Unmount نمودن درایوها هستید بدون اینکه از آن آگاه باشید.

شما پیش از اینکه یک فلش درایو را از سیستم جدا کنید، آن را eject می‌نمایید تا از بروز مشکل برای اطلاعات ذخیره شده در آن و همچنین آسیب سخت افزاری جلوگیری به عمل آورید. در واقع Eject نام دیگری برای Unmount می‌باشد.

همانند دستورmount ، شما می‌توانید از دستور unmount برای Unmount نمودن یک درایو از سیستم‌عامل استفاده نمایید.

دستور زیر نمونه‌ای از Unmount نمودن یک درایو می‌باشد:

```
unmount /dev/sdb1
```

توجه داشته باشید، دستگاهی که در حال فعالیت می‌باشد را نمی‌توان Unmount نمود. بنابراین اگر سیستم در حال خواندن و نوشتن بر روی دستگاه باشد، شما با اجرای دستور بالا، با پیام خطا مواجه خواهید شد.

ضمن تشکر از جناب آقای نصیری و وب سایت توسینسو، برخی از مطالب بخش مدیریت فایل سیستم در لینوکس از **وب سایت توسینسو** و نوشته جناب آقای امیرحسین کریم پور گرفته شده است.

فصل دوازدهم

سیستم ثبت لاگ در لینوکس

یکی از موارد ضروری برای هر کاربر لینوکس که باید از آن اطلاع داشته باشد، نحوه کار با فایل‌های لاگ است.

فایل‌های لاگ، اطلاعاتی مربوط به وقایعی که در سیستم‌عامل و اپلیکیشن ها، هنگام اجرا اتفاق افتاده است در خود ذخیره می‌کنند که البته این اطلاعات می‌تواند شامل هر پیام خطا یا هشدارهای امنیتی نیز باشد.

فایل‌های لاگ می‌توانند حاوی ردپایی از فعالیت‌های نفوذگر باشند. بنابراین دانستن اطلاعات لازم در مورد ثبت لاگ در لینوکس بسیار حائز اهمیت خواهد بود.

سرویس rsyslog

از یک سرویسی که با نام syslogd شناخته می‌شود، برای ثبت لاگ در لینوکس استفاده می‌شود. چندین syslog مختلف مانند rsyslog و syslog-ng در لینوکس وجود دارد که در توزیع‌های مختلف لینوکس متفاوت می‌باشند.

اگرچه این سرویس‌ها اغلب مشابه یکدیگر عمل می‌کنند ولی در برخی موارد جزئی با هم تفاوت دارند.
با توجه به اینکه **کالی لینوکس** مبتنی بر Debian بوده و سیستم‌عامل Debian نیز به صورت پیش فرض با rsyslog کار می‌کند، ما در این بخش از دوره مقدمات لینوکس نیز به این syslog اشاره می‌کنیم.
در ابتدا کلیه فایل‌هایی که مرتبط با rsyslog هستند را در سیستم جست و جو می‌کنیم. برای این کار از دستور زیر استفاده می‌کنیم:

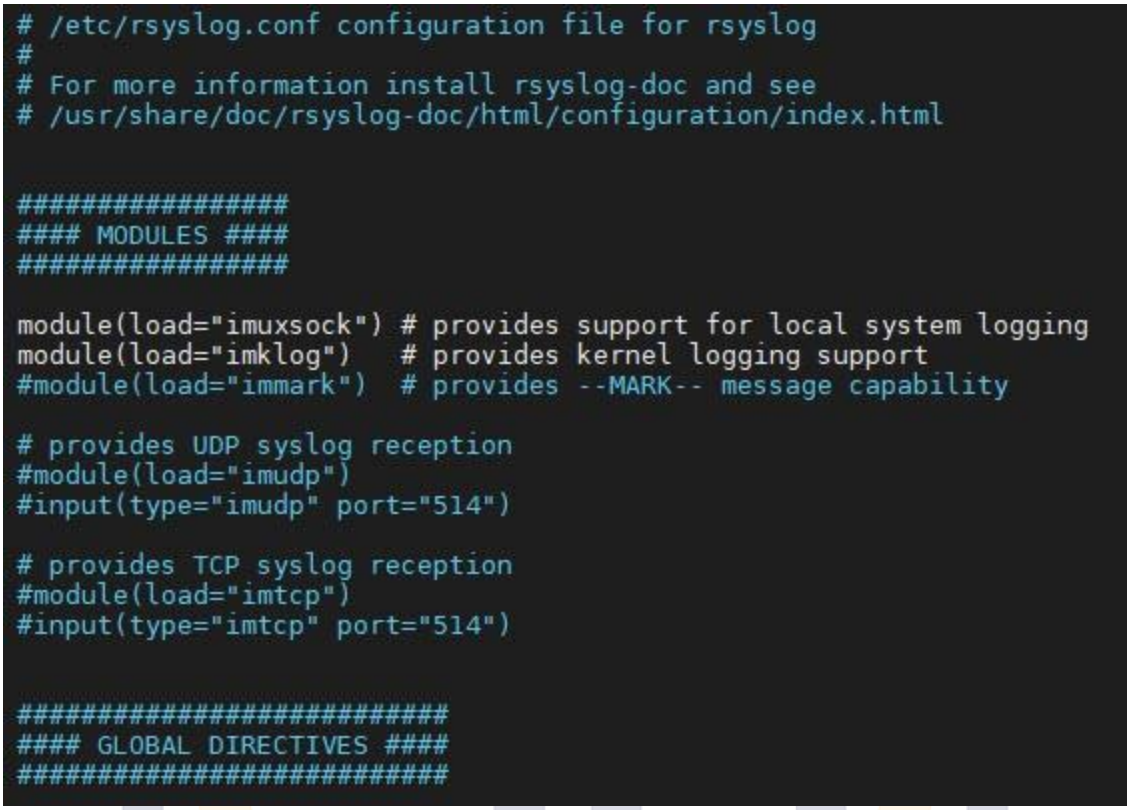
```
locate rsyslog
```

```
root@kali:~# locate rsyslog
/etc/rsyslog.conf
/etc/rsyslog.d
/etc/init.d/rsyslog
/etc/logcheck/ignore.d.server/rsyslog
/etc/logrotate.d/rsyslog
/etc/rc0.d/K01rsyslog
/etc/rc1.d/K01rsyslog
/etc/rc2.d/S01rsyslog
/etc/rc3.d/S01rsyslog
/etc/rc4.d/S01rsyslog
/etc/rc5.d/S01rsyslog
/etc/rc6.d/K01rsyslog
/etc/rsyslog.d/20-ufw.conf
/etc/systemd/system/multi-user.target.wants/rsyslog.service
/usr/lib/rsyslog
/usr/lib/rsyslog/rsyslog-rotate
/usr/lib/systemd/system/rsyslog.service
/usr/lib/x86_64-linux-gnu/rsyslog
/usr/lib/x86_64-linux-gnu/rsyslog/fmhash.so
/usr/lib/x86_64-linux-gnu/rsyslog/imfile.so
/usr/lib/x86_64-linux-gnu/rsyslog/imjournal.so
/usr/lib/x86_64-linux-gnu/rsyslog/imklog.so
/usr/lib/x86_64-linux-gnu/rsyslog/imkmsg.so
/usr/lib/x86_64-linux-gnu/rsyslog/immark.so
/usr/lib/x86_64-linux-gnu/rsyslog/impstats.so
/usr/lib/x86_64-linux-gnu/rsyslog/imptcp.so
/usr/lib/x86_64-linux-gnu/rsyslog/imtcp.so
```

همانطور که در تصویر بالا مشاهده می‌کنید، فایل‌های بسیاری وجود دارند که حاوی عبارت syslog می‌باشند. در ادامه ما به توضیح فایل rsyslog.conf می‌پردازیم.

فایل پیکربندی rsyslog

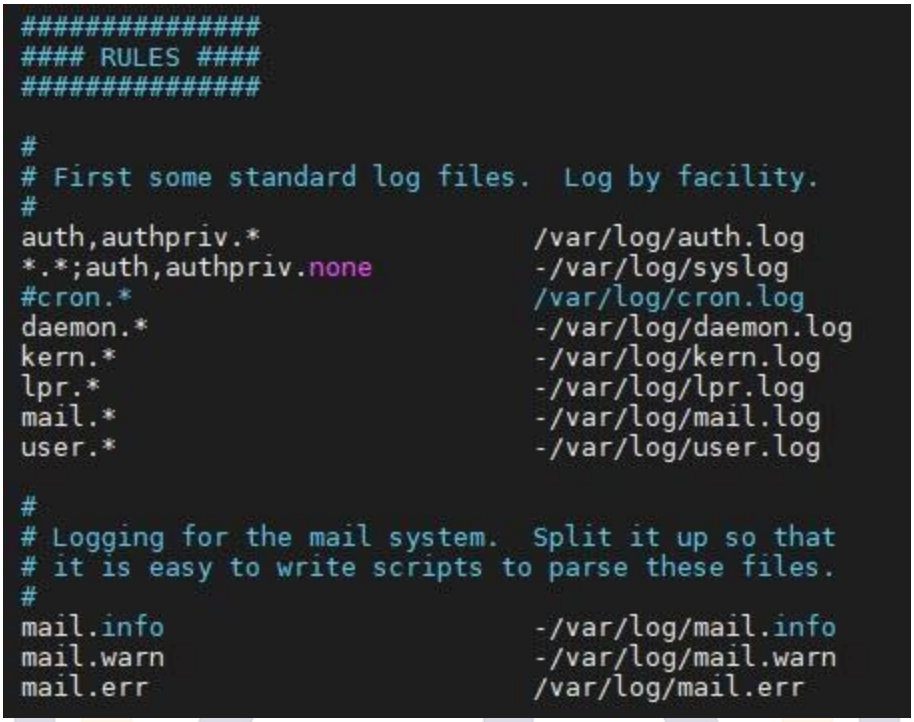
همانند اپلیکیشن‌های دیگر در لینوکس، rsyslog نیز دارای یک فایل برای تنظیم و مدیریت می‌باشد که در دایرکتوری etc قرار دارد که نام آن rsyslog.conf می‌باشد. تصویر زیر بخشی از محتویات این فایل را نمایش می‌دهد:



```
# /etc/rsyslog.conf configuration file for rsyslog
#
# For more information install rsyslog-doc and see
# /usr/share/doc/rsyslog-doc/html/configuration/index.html


#################
#### MODULES ####
#################

module(load="imuxsock") # provides support for local system logging
module(load="imklog")   # provides kernel logging support
#module(load="immark")  # provides --MARK-- message capability

# provides UDP syslog reception
#module(load="imudp")
#input(type="imudp" port="514")

# provides TCP syslog reception
#module(load="imtcp")
#input(type="imtcp" port="514")


###########################
#### GLOBAL DIRECTIVES ####
###########################
```

Rule ها در سرویس rsyslog

rules درrsyslog ، مشخص می‌کند که چه اطلاعاتی باید ثبت شود. در واقع برنامه‌ای که لاگ آن ثبت می‌شود و مسیر ذخیره سازی لاگ‌ها در این بخش مشخص می‌شود. تصویر زیر بخش rules در فایل پیکربندی rsyslog می‌باشد که حاوی اطلاعات مربوط به برنامه ها و مسیر ذخیره سازی لاگ آن‌ها می‌باشد:



```
###############
#### RULES ####
###############

#
# First some standard log files.  Log by facility.
#
auth,authpriv.*                 /var/log/auth.log
*.*;auth,authpriv.none          -/var/log/syslog
#cron.*                         /var/log/cron.log
daemon.*                        -/var/log/daemon.log
kern.*                          -/var/log/kern.log
lpr.*                           -/var/log/lpr.log
mail.*                          -/var/log/mail.log
user.*                          -/var/log/user.log

#
# Logging for the mail system.  Split it up so that
# it is easy to write scripts to parse these files.
#
mail.info                       -/var/log/mail.info
mail.warn                       -/var/log/mail.warn
mail.err                        /var/log/mail.err
```

در هر خط مشخص شده است که چه برنامه‌ای و در چه مسیری، لاگ‌های خود را ذخیره می‌نماید. فرمت اصلی این rule ها به صورت زیر می‌باشد:

facility.priority action

عبارت facility اشاره به برنامه‌ای دارد که در حال ثبت رخداد آن می‌باشیم. مواردی همچونmail ، kernelیا موارد مشابه در این بخش قرار می‌گیرند.

عبارت priority نشان دهنده نوع پیامی است که ثبت می‌شود که عبارتinfo ، warnیا err در این بخش قرار می‌گیرند که بیانگر اطلاعات، پیام هشدار و پیام خطا می‌باشند.

عبارت action نیز که در سمت راست قرار می‌گیرد، اشاره به محل ذخیره سازی لاگ را مشخص می‌کند.

لیست زیر مربوط به عباراتی است که می‌تواند در بخش facility قرار داده شود:

auth/authpriv Security/authorization messages

cron Clock daemons

daemon Other daemons

ern Kernel messages

lpr Printing system

mail Mail system

user Generic user-level messages

همچنین شما می‌توانید از ستاره نیز برای ذخیره سازی کلیه موارد استفاده نمایید.

در ادامه همچنین لیستی از priority هایی که می‌توانید از آن استفاده نمایید قرار گرفته شده است:

debug
info
notice
warning
warn
error
err
crit
alert
emerg
panic

در این بخش نیز شما می‌توانید از عبارت ستاره استفاده نمایید تا کلیه موارد ذخیره گردد.

توجه داشته باشید که کدهایwarning ، warn، error، err، emergو panic منسوخ شده‌اند و نباید از آن‌ها استفاده شود.

بخش action نیز نام فایل و مسیری است که لاگ‌ها در آن ذخیره می‌گردند. توجه داشته باشید که عموماً این لاگ‌ها در مسیر /var/log قرار داشته و نام فایل آن نیز معمولاً هم نام با همان facility می‌باشد. مثال زیر نمونه‌ای از این مورد می‌باشد:

mail.* /var/log/mail

پاک سازی خودکار لاگ‌ها با استفاده از logrotate

در صورتی که فایل‌های لاگ به شکل دوره‌ای حذف نشوند، فضای زیادی را اشغال می‌کنند. از طرفی نیز اگر شما فایل‌های لاگ را بیش از حد حذف کنید، در صورت بروز مشکل، دیگر امکان مراجعه به این فایل‌ها وجود ندارد. برای ایجاد تعادل در این مورد، شما می‌توانید از logrotate استفاده نمایید logrotate .یکی از مباحث مهم ثبت لاگ در لینوکس می‌باشد.

چرخش لاگ یاLog Rotation ، فرآیندی است که در آن مرتباً فایل‌های لاگ آرشیو شده، به محل دیگری منتقل شده و یک فایل لاگ جدید ایجاد می‌گردد. همچین فایل انتقال داده شده نیز در یک بازه زمانی خاص از سیستم حذف خواهندشد.

شما می‌توانید تنظیمات مربوط به logrotate را از طریق دسترسی به فایل تنظیمات آن در مسیر /etc/logrotate.conf انجام دهید.

```
# see "man logrotate" for details
# rotate log files weekly
weekly

# keep 4 weeks worth of backlogs
rotate 4

# create new (empty) log files after rotating old ones
create

# use date as a suffix of the rotated file
#dateext

# uncomment this if you want your log files compressed
#compress

# packages drop log rotation information into this directory
include /etc/logrotate.d

# system-specific logs may be also be configured here.
```

غیرفعال نمودن لاگ برداری

هنگامی که شما یه یک سیستم لینوکسی دسترسی پیدا می‌کنید، یکی از مواردی که باید به آن توجه نمایید، غیرفعال نمودن قابلیت ثبت لاگ و حذف شواهد است.

از بین بردن شواهد

در ابتدا شما می‌خواهید تا لاگ‌های مربوط به فعالیت‌های خود را حذف نمایید. شما می‌توانید به راحتی فایل‌های لاگ را باز نموده و هر بخشی که مدنظر دارید را حذف نمایید. توجه داشته باشید که این کار می‌تواند یک شکاف زمانی را در فایل‌های لاگ باقی بگذارد که خود مشکوک خواهد بود. همچنین فایل‌های حذف شده توسط یک بازرس قانونی ماهر که با علوم فارنزیک آشنایی کافی دارد، قابل بازیابی هستند.

یک راه حل بهتر و مطمئن‌تر، خراب کردن یا Shred نمودن فایل‌های لاگ می‌باشد.

همانطور که اشاره شد، با توجه به سیستم‌های مختلف حذف فایل، یک بازرس ماهر هنوز قادر به بازیابی فایل‌های حذف شده می‌باشند. اما فرض کنید که راهی برای پاک کردن فایل و بازنویسی چندین باره آن‌ها وجود داشته باشد که این کار منجر به بازیابی سخت تر آن‌ها خواهد شد.

خوشبختانه، لینوکس دارای یک دستور داخلی برای این منظور بوده که نام آن shred می‌باشد. این ابزار دارای Option های مختلفی می‌باشد ولی به صورت ساده می‌توان پس از این دستور، نام فایل مورد نظر را وارد نمود. بدین صورت ابزار

shred، فایل مورد را حذف نموده و چندین بار آن را رونویسی می‌کند که به صورت پیش فرض، تعداد این باز نویسی چهار بار می‌باشد.

به طور کلی، هر چه تعداد رونویسی فایل، بیشتر باشد، بازیابی آن سخت تر خواهد بود. اما به خاطر داشته باشید که هر رونویسی به زمان نیاز دارد، بنابراین برای فایل‌های با حجم زیاد، shred نمودن آن زمان بر خواهد بود.

از بین سوییچ‌های ابزارshred ، دو سوییچ f- و n- مفیدتر می‌باشند.

از سوییچ f- به منظور تغییر سطح دسترسی فایل برای رونویسی آن در صورت نیاز و از سوییچ n- برای مشخص نمودن تعداد دفعات رونویسی فایل استفاده می‌شود. دستور زیر نمونه ای از استفاده ابزار shred با سوییچ‌های مذکور می‌باشد:

```
shred -f -n 10 /var/log/auth.log.*
```

هنگامی که یک فایل را shred می‌نمایید، محتویات آن غیرقابل توصیف خواهد شد. فایل زیر نمونه‌ای از یک فایل shred شده می‌باشد:

حال اگر مهندس امنیتی یا متخصص فارنزیک، فایلی که shred شده باشد را بررسی کند، این فایل، هیچ کاربردی نداشته و همچنین قابل بازیابی نیز نخواهد بود.

غیرفعال کردن ثبت لاگ

روش دیگری که برای پاک کردن ردپا از آن استفاده می‌شود، غیرفعال نمودن ثبت رخداد است.

هنگامی که نفوذگر، کنترل یک سیستم در اختیار خود قرار می‌دهد، می‌تواند با غیرفعال نمودن قابلیت ثبت رخداد، از ثبت فعالیت‌های خود جلوگیری به عمل آورد. برای این کار شما نیاز به دسترسی root خواهید داشت.

به منظور غیرفعال نمودن ثبت رخداد، می‌بایست سرویس rsyslog را غیرفعال نمایید:

```
service rsyslog stop
```

با اجرای دستور بالا، لینوکس فرآیند تولید هر گونه لاگ را متوقف می‌نماید.

بخش سیزدهم

استفاده از سرویس ها در لینوکس

در اصطلاحات لینوکس، یک سرویس، برنامه‌ای است که در پس زمینه اجرا شده و منتظر بوده تا شما از آن استفاده نمایید. لینوکس دارای سرویس‌های متعددی است که از پیش نصب شده‌اند.

از این میان، یکی از سرویس‌های کاربردی لینوکس ، سرویس آپاچی می‌باشد که برای ایجاد، مدیریت و استقرار سرورهای وب از آن استفاده می‌شود. در این بخش ما سه سرویس را انتخاب کرده‌ایم که برای تست نفوذگران از اهمیت ویژه‌ای برخوردار هستند که عبارتند از Apache ، OpenSSH، MySQL

Starting, Stopping و Restarting سرویس ها

پیش از اینکه به توضیح سرویس‌های مذکور بپردازیم، ابتدا به توضیح در مورد start ، stopو restart سرویس‌ها در لینوکس می‌پردازیم.

برخی از سرویس‌ها را می‌توانید بوسیله محیط گرافیکی در کالی لینوکس، startیا stop نمایید ولی برخی از آن‌ها نیز باید بوسیله محیط دستوری start یا stop شوند.

ساختار کلی فعال یا غیرفعال کردن سرویس‌ها به صورت زیر می‌باشد:

```
service servicename start|stop|restart
```

به عنوان مثال برای فعال نمودن سرویسApache ، کافی است دستور زیر را وارد نمایید:


```
service apache2 start
```

همچنین برای stop نمودن این سرویس نیز می‌توانید از عبارت stop بعد از نام سرویس استفاده نمایید.

معمولا زمانی که شما یک سرویس را پیکربندی می‌کنید و یا پیکربندی آن را تغییر می‌دهید، باید سرویس را restart نمایید.

وب سرور Apache

وب سرور آپاچی در واقع رایج ترین سرویس مورد استفاده در سیستم‌عامل‌های لینوکس می‌باشد. وب سرور آپاچی در بیش از ۶۰ درصد از سرورهای وب دنیا یافت می‌شود، بنابراین یک مدیر سیستم لینوکسی باید با این سرویس آشنایی داشته باشد.

به عنوان یک تست نفوذگر نیز در صورتی که مایل به تست نفوذ وب سایت‌ها باشید، باید از عملکرد داخلی سرویس آپاچی، وب سایت و پایگاه داده‌ای که وب سایت با آن کار می‌کند، آگاهی لازم را داشته باشید.

فعال سازی وب سرور Apache

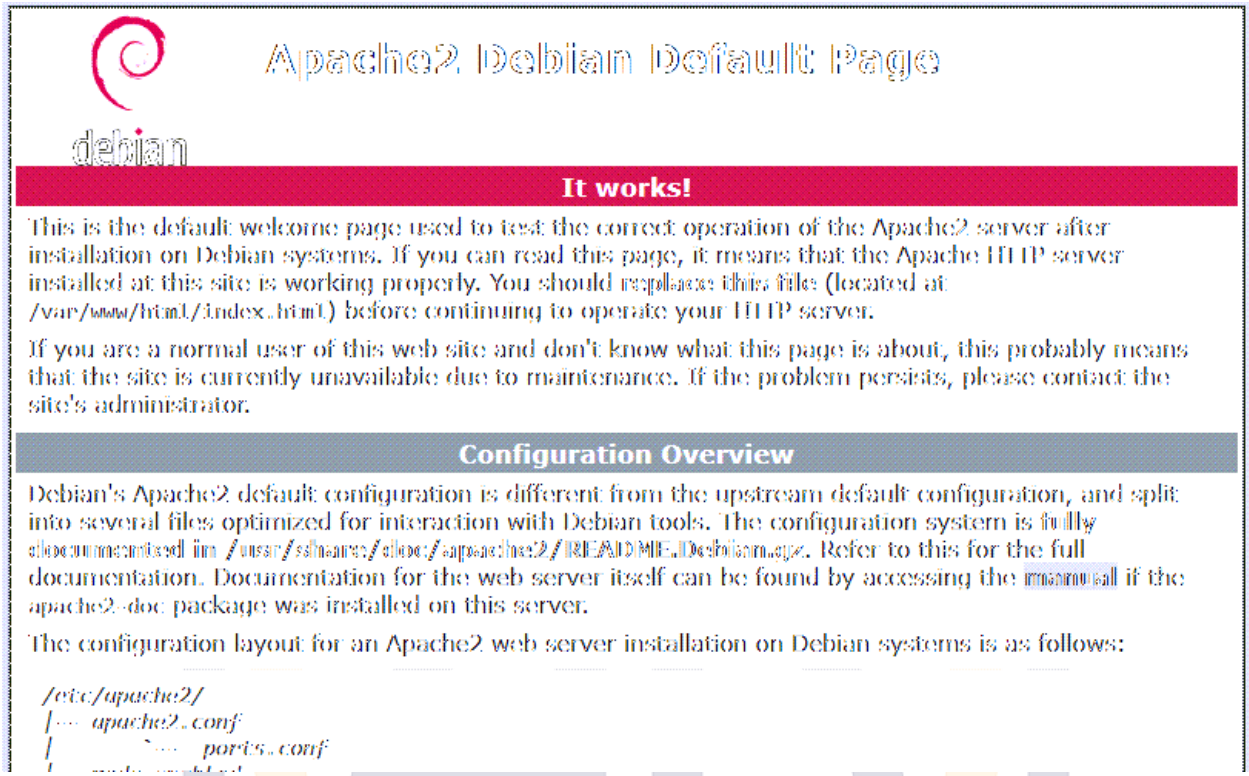
در صورتی که شما از سیستم‌عامل کالی لینوکس استفاده می‌کنید، سرویس Apache به صورت پیش فرض بر روی سیستم شما نصب می‌باشد. در صورتی که این سرویس به هر دلیلی بر روی سیستم‌عامل شما نصب نشده است می‌توانید از دستور زیر برای نصب آن استفاده نمایید:

```
apt-get install apache2
```

علاوه بر سرویس آپاچی شما برای ایجاد یک وب سایت، شما به یک پایگاه داده MySQL و همچنین برای نوشتن اسکریپت‌های وب، نیاز به یک زبان برنامه نویسی تحت وب مانند php یا perl نیز نیاز خواهید داشت.

ترکیب لینوکس، Apache، MySQLو PHP یک مجموعه قدرتمند را برای طراحی و توسعه برنامه‌های تحت وب ایجاد می‌نماید که در مجموع به عنوان LAMP نیز شناخته می‌شود. معادل همین مجموعه برای ویندوز با نام WAMP شناخته می‌شود که حرف W از این عبارت به Windows اشاره دارد.

پس از فعال کردن سرویس آپاچی، در صورتی که شما مرورگر خود را باز نموده و عبارت localhost را در آن وارد نمایید، صفحه‌ای مشابه تصویر زیر را مشاهده خواهید کرد:



شما می‌توانید با مراجعه با فایل index.html که در مسیر /var/www/html/ قرار دارد، محتوای این صفحه را ویرایش نمایید.

سرویس OpenSSH

SSHکه مخفف Secure Shell می‌باشد، شما را قادر می‌سازد که به صورت امن یک ارتباط دستوری را از راه دور ایجاد نمایید. در واقع از این سرویس می‌توان به عنوان یک جایگزین مناسب برای سرویس telnet استفاده نمود که اطلاعات را به صورت Cleartext انتقال می‌دهد.

مدیران سیستم، اغلب از SSH برای مدیریت سرورها از راه دور استفاده نموده و نفوذگران از این سرویس برای اتصال به سیستمی که آن را در اختیار خود گرفته اند، استفاده می‌نمایند.

برای فعال کردن OpenSSH که یکی دیگر از سرویس‌های کاربردی لینوکس می‌باشد، در سیستم‌عامل کالی لینوکس از دستور زیر استفاده می‌شود:

```
service ssh start
```

همچنین در صورتی که قصد اتصال به یک سیستم را بوسیله SSH داشته باشید می‌توانید از دستور زیر استفاده نمایید:

```
ssh test@192.168.1.100
```

در مثال بالا به جای test نام کاربری در سیستم مقصد را وارد نموده و پس از علامت @ آدرس IP سیستم مقصد را وارد می‌نماییم. پس از وارد نمودن دستور بالا، از شما کلمه عبور مربوط به نام کاربری وارد شده، درخواست می‌شود و پس از وارد نمودن آن، شما به سیستم مورد نظر متصل خواهید شد.

سرویس MySQL

MySQL یکی از رایج ترین برنامه‌های پایگاه داده می‌باشد که در برنامه‌های وب از آن استفاده می‌شود. پایگاه‌های داده یکی از بخش‌های مورد علاقه نفوذگران می‌باشد. زیرا آن‌ها حاوی اطلاعات مهمی برای نفوذگران می‌باشند.

همانند لینوکس، MySQL نیز متن باز بوده و اغلب بر روی سیستم‌عامل‌های لینوکس نصب شده است. تعداد زیادی از برنامه های تحت وب و حتی سیستم‌های مدیریت محتوا (CMS) مانند وروپرس، جوملا و دروپال نیز از این پایگاه داده استفاده می‌کنند.

فعال سازی سرویس MySQL

خوشبختانه MySQL به صورت پیش فرض بر روی کالی لینوکس نصب شده است و در صورتی که شما از توزیع دیگری استفاده می‌نمایید می‌توانید به آدرس زیر مراجعه نموده و آن را دانلود نمایید:

https://www.mysql.com/downloads

برای فعال سازی سرویس MySQL که یکی از سرویس‌های کاربردی لینوکس می‌باشد، از دستور زیر استفاده می‌شود:

```
service mysql start
```

برای وارد شدن به محیط MySQL از دستور زیر استفاده می‌شود:

```
mysql -u root -p
```

```
root@kali:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 49
Server version: 10.3.14-MariaDB-1 Debian buildd-unstable

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

در صورتی که شما برای اولین از این سرویس استفاده می‌نمایید، نیازی به وارد نمودن کلمه عبور نبوده و هنگام درخواست کلمه عبور، کلید Enter را فشار دهید.

تعامل با MySQL

SQL یک زبان برنامه نویسی تفسیر شده برای ارتباط با پایگاه داده می‌باشد. پایگاه داده اغلب به صورت رابطه‌ای بوده و در آن چندین جدول با ارتباطات خاص با یکدیگر ایجاد می‌شوند. برخی از دستوراتی که در این ساختار استفاده می‌گردند عبارتند از:

Select: برای بازیابی اطلاعات استفاده می‌شود.
union: برای ترکیب نتایج یک یا تعداد بیشتر دستور select از آن استفاده می‌شود.
insert: برای اضافه نمودن داده استفاده می‌شود.
update: برای تغییر در داده های موجود استفاده می‌شود.
delete: برای حذف داده ها استفاده می‌شود.
دستور زیر نمونه ای از استفاده select در پایگاه داده می‌باشد:

```
select user, password from customers where user='admin';
```

دستور بالا، ستون‌های نام کاربری و کلمه عبور از جدولی با نام customer در جایی که نام کاربری برابر با admin می‌باشد را نمایش می‌دهد.

تنظیم کلمه عبور برای MySQL

در ابتدا قصد داریم تا کلیه کاربرانی که در mysql وجود دارند را مشاهده نماییم. بدین منظور از دستور زیر استفاده می‌کنیم:

Select user, host , password from mysql.user;

```
MariaDB [(none)]> select user, host , password from mysql.user;
+------+-----------+----------+
| user | host      | password |
+------+-----------+----------+
| root | localhost |          |
+------+-----------+----------+
1 row in set (0.371 sec)
```

همانطور که در تصویر بالا مشاهده می‌کنید، کاربر کلمه عبوری برای کاربر root تنظیم نشده است. در ادامه قصد داریم تا برای این نام کاربری، یک کلمه عبور تنظیم نماییم. بدین منظور ابتدا باید پایگاه داده مورد نظر را انتخاب نماییم.

برای مشاهده لیست پایگاه داده موجود در mysql از دستور زیر استفاده می‌نماییم:

show database;

```
MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
+--------------------+
3 rows in set (0.097 sec)
```

به صورت پیش فرض سه پایگاه داده در Mysql تعریف شده است که دو پایگاه داده information_schema و performance_schemaبرای موارد مدیریت بوده و ما از آن‌ها استفاده نمی‌کنیم. در این جا برای تنظیم کلمه عبور می‌بایست از پایگاه داده mysql استفاده نماییم.

برای این منظور بوسیله دستور زیر، این پایگاه داده را فراخوانی می‌نماییم:

use mysql;

پس از فراخوانی پایگاه داده mysql و اتصال به آن بوسیله دستور بالا، برای تنظیم کلمه عبور برای آن، از دستور زیر استفاده می‌نماییم:

```
update user set password = PASSWORD("hackers-arise") where user = 'root';
```

در دستور بالا به جای عبارتhackers-arise ، شما می‌توانید کلمه عبور مورد نظر خود را وارد نمایید.

دسترسی از راه دور به دیتابیس

در برخی از موارد شما نیاز دارید که از راه دور به یک پایگاه داده متصل شوید. بدین منظور باید در ادامه دستورات اتصال به پایگاه داده، آدرس IP مقصد را نیز وارد نمایید:

```
mysql -u root -p 192.168.1.100
```

توجه داشته باشید که کاربر root در mysql به صورت پیش فرض دارای کلمه عبور نمی‌باشد و شما باید پیش از هر کاری، با دستوراتی که در بخش پیشین به آن اشاره کردیم، کلمه عبور مناسبی را برای کاربر root تنظیم نمایید.

جداول دیتابیس

یکی از مواردی که باید به آن توجه داشته باشید، مشاهده جداول یک پایگاه داده می‌باشد. برای مشاهده جداول یک پایگاه داده ابتدا باید پایگاه داده مورد نظر را با دستور use انتخاب نموده و با دستورshow tables ، لیست جداول آن را مشاهده نمایید:

```
MariaDB [mysql]> show tables;
+---------------------------+
| Tables_in_mysql           |
+---------------------------+
| column_stats              |
| columns_priv              |
| db                        |
| event                     |
| func                      |
| general_log               |
| gtid_slave_pos            |
| help_category             |
| help_keyword              |
| help_relation             |
| help_topic                |
| host                      |
| index_stats               |
| innodb_index_stats        |
```

جهت نمایش نوع داده‌های استفاده شده در هر یک از جداول می‌توانید از دستور describe استفاده نمایید:

```
MariaDB [mysql]> describe host;
+-----------------------+---------------+------+-----+---------+-------+
| Field                 | Type          | Null | Key | Default | Extra |
+-----------------------+---------------+------+-----+---------+-------+
| Host                  | char(60)      | NO   | PRI |         |       |
| Db                    | char(64)      | NO   | PRI |         |       |
| Select_priv           | enum('N','Y') | NO   |     | N       |       |
| Insert_priv           | enum('N','Y') | NO   |     | N       |       |
| Update_priv           | enum('N','Y') | NO   |     | N       |       |
| Delete_priv           | enum('N','Y') | NO   |     | N       |       |
| Create_priv           | enum('N','Y') | NO   |     | N       |       |
| Drop_priv             | enum('N','Y') | NO   |     | N       |       |
| Grant_priv            | enum('N','Y') | NO   |     | N       |       |
| References_priv       | enum('N','Y') | NO   |     | N       |       |
| Index_priv            | enum('N','Y') | NO   |     | N       |       |
| Alter_priv            | enum('N','Y') | NO   |     | N       |       |
| Create_tmp_table_priv | enum('N','Y') | NO   |     | N       |       |
| Lock_tables_priv      | enum('N','Y') | NO   |     | N       |       |
| Create_view_priv      | enum('N','Y') | NO   |     | N       |       |
| Show_view_priv        | enum('N','Y') | NO   |     | N       |       |
| Create_routine_priv   | enum('N','Y') | NO   |     | N       |       |
| Alter_routine_priv    | enum('N','Y') | NO   |     | N       |       |
| Execute_priv          | enum('N','Y') | NO   |     | N       |       |
| Trigger_priv          | enum('N','Y') | NO   |     | N       |       |
+-----------------------+---------------+------+-----+---------+-------+
```

در دستور بالا اطلاعات مربوط به جدول host در پایگاه داده mysql نمایش داده شده است.

استخراج اطلاعات با دستور Select

به منظور استخراج اطلاعات جداول، از دستور select استفاده می‌شود. در دستور select شما نیاز به نام جدول مورد نظر و نام ستون‌های آن دارید. ساختار کلی این دستور به صورت زیر می‌باشد:

Select columns from table

همچنین شما برای مشاهده کلیه ستون‌های جدول مورد نظر می‌توانید به جای نوشتن نام کلیه ستون‌های آن، از ستاره استفاده نمایید.

فصل چهاردهم

ناشناس ماندن در لینوکس

امروزه تقریبا هر کاری که در اینترنت انجام می‌دهید، ردیابی می‌شود. هر کسی که ردیابی را انجام می‌دهد، خواه گوگل باشد که جست و جوی‌های آنلاین، بازدیدهای وب سایت و ایمیل ما را بررسی می‌کند یا آژانس امنیت ملی آمریکا(NSA) ، هر دو در حال ثبت فعالیت‌های ما به گونه‌های مختلف بوده و سپس آن‌ها را برای منافع شخصی یا سازمانی خود مورد استفاده قرار می‌دهند.

به طور خاص یک نفوذگر، باید چگونگی محدود کردن این ردیابی‌ها و همچنین نحوه ناشناس ماندن در وب را بداند. در این بخش ما قصد داریم تا شما را با نحوه ناشناس ماندن در وب آشنا نماییم.

البته باید به این نکته توجه داشت که هیچ روش کاملاً مطمئنی وجود ندارد که فعالیت‌های شما را از چشمان کنجکاو در امان نگه دارد و با توجه به زمان و منابع کافی، هر چیزی را می‌توان ردیابی نمود. با این حال، این روش‌ها احتمالا کار ردیاب‌ها را بسیار دشوارتر می‌کند.

نحوه کارکرد اینترنت

هنگامی که شما از وارد فضای اینترنت می‌شوید، آدرس IP شما شناسایی می‌گردد. داده‌های ارسالی از سیستم شما، معمولاً با آدرس IP شما برچسب گذاری می‌شود و این موضوع ردیابی فعالیت‌های شما را آسان می‌کند.

همچنین شرکت هایی مانند گوگل و سایر سرویس های ایمیل، متن ایمیل شما را می‌خوانند و به دنبال کلمات کلید برای کارآمدتر نمودن تبلیغات به شما هستند.

بیایید ابتدا به این نکته بپردازیم که آدرس‌های IP در اینترنت چگونه تشخیص داده می‌شوند.

هنگامی که شما یک بسته اطلاعاتی را از طریق اینترنت ارسال می‌کنید، این بسته شامل آدرس IP مبدا و مقصد می‌باشد. در این حالت، بسته می‌داند که به کجا می رود و همچنین پاسخ این بسته باید به کجا ارسال شود. این بسته مسیر خود را از طریق روترهایی که بین راه در اینترنت قرار دارند پیدا می‌کند. وظیفه این روترها، مسیریابی بسته های اینترنتی می‌باشد.

ممکن است بین فرستنده تا گیرنده ۲۰ تا ۳۰ روتر وجود داشته باشد که معمولاً هر بسته اینترنتی با گذشتن از حدود ۱۵ روتر به مقصد خود خواهد رسید. در اصطلاح به هر روتر یک hop گفته می‌شود.

برای مشاهده اینکه چه تعداد روتری بین شما و مقصد مورد نظر وجود دارد، از دستور traceroute استفاده می‌شود که پس از آن نام وب سایت یا آدرس IP مورد نظر شما قرار می‌گیرد.

بدین صورت شما زمانی که وارد یک وب سایت می‌شوید، این وب سایت می‌تواند آدرس شما که همان IP مبدا می‌باشد را در سیستم خود به عنوان یکLog ، ثبت نماید.

سیستم TOR

در دهه نود، US Office of Naval Research (ONR)تصمیم گرفت تا روشی را برای گشت و گذار ناشناس در اینترنت برای اهداف جاسوسی تهیه نماید. این برنامه برای ایجاد شبکه‌ای از روترهای مجزا از روترهای اینترنتی طراحی گردیده بود که می‌توانست ترافیک را رمزنگاری نموده و تنها آدرس IP مربوط به روتر پیشین خود را به صورت رمزنگاری نشده ذخیره می کرد.

این بدین معنی است که آدرس تمامی روترهای دیگر در طول مسیر رمزنگاری شده بودند. ایده اصلی این بود که هر کسی که ترافیک را مشاهده کند، نتواند مبدا یا مقصد داده‌ها را تشخیص دهد. این تحقیق در سال ۲۰۰۲ به عنوان پروژه The Onion Routerیا همان Tor معروف شد و در حال حاضر در دسترس همگان قرار دارد. امروزه از این روش می‌توان به عنوان یکی از ابزارهای ناشناس ماندن در لینوکس و در بستر وب استفاده نمود.

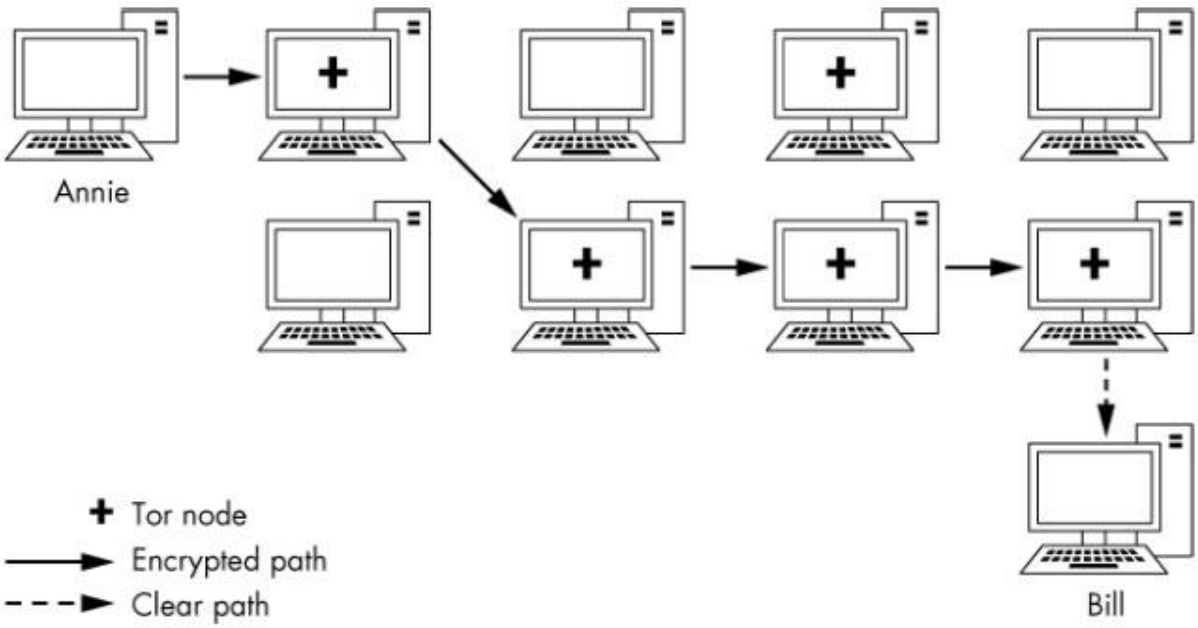
نحوه کارکرد TOR

بسته‌های ارسالی در Tor از طریق روترهای معمولی که تعداد زیادی از آن‌ها مورد نظارت قرار می‌گیرند، ارسال نمی‌شوند. بلکه به لطف داوطلبانی که اجازه می‌دهند تا سیستم آن‌ها توسط Tor مورد استفاده قرار گیرد، از طریق شبکه‌ای با بیش از هفت هزار روتر در سراسر جهان ارسال می‌شوند.

علاوه بر استفاده از یک شبکه روتر کاملاً مجزا، Torاطلاعات، آدرس مبدا و مقصد هر بسته را نیز رمزنگاری می‌کند. در هرhop ، اطلاعات در صورت دریافت، رمزنگاری شده و سپس توسط hop بعدی رمزگشایی می‌شوند. بدین ترتیب، هر بسته تنها شامل اطلاعاتی در مورد hop قبلی خود در طول مسیر بوده و آدرس IP مبدا را نمی‌داند.

اگر کسی اقدام به انجام عملیات Intercept نماید، تنها قادر به مشاهده آدرس IP مربوط به hop قبلی بوده و به قولی صاحب یک وب سایت تنها می‌تواند آدرس IP آخرین روتری که اطلاعات از آن دریافت شده است را مشاهده نماید.

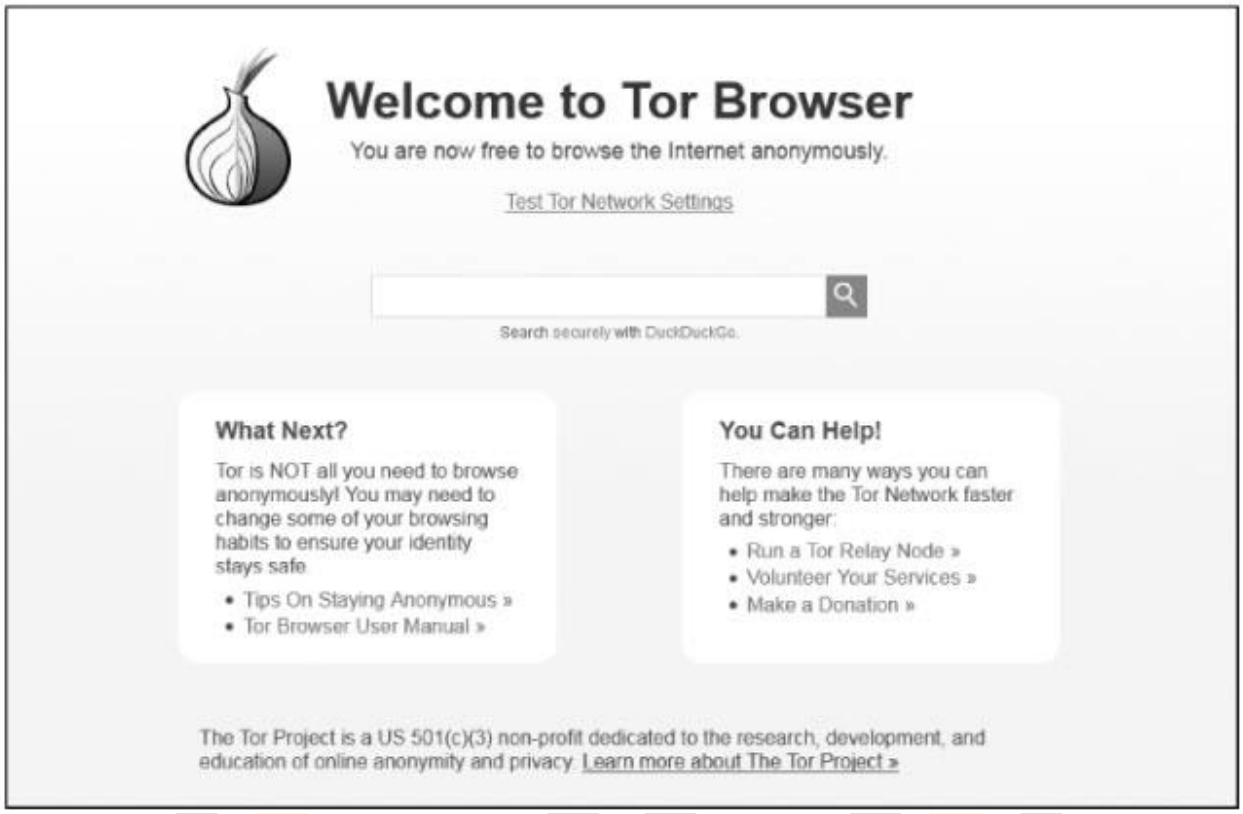
این امر، ناشناس ماندن نسبی را در سراسر اینترنت تضمین می‌کند. به تصویر زیر توجه نمایید:



به منظور فعال نمودنTor ، تنها کافی است تا مرورگر Tor را از آدرس زیر دانلود نموده و نصب نمایید:

https://www.torproject.org

پس از نصب، شما با تصویری مشابه تصویر زیر مواجه خواهید شد که مشابه یک مرروگر اینترنت قدیمی به نظر می‌رسد:



با استفاده از این مرورگر، شما می‌توانید، فعالیت‌های اینترنتی خود را از طریق روترهای مجزا انجام داده و بدون اینکه فعالیت‌های شما ردیابی شود، اقدام به مشاهده وب سایت‌های مورد نظر خود نمایید.

متاسفانه، گشت و گذار از طریق مرورگر Tor به علت عدم وجود روترهای فراوان، بسیار کندتر از گشت و گذار معمولی شما خواهد بود.

مرورگر Tor شما را قادر می‌سازد تا به Dark Web دسترسی داشته باشید. وب سایت‌هایی که Dark Web را تشکیل می‌دهند، خواهان ناشناس ماندن هستند، بنابراین تنها از طریق مرورگر Tor امکان دسترسی به آن‌ها فراهم می‌باشد و آدرس آن‌ها نیز به onion. ختم می‌شود.

Dark Webبه دلیل فعالیت های غیرقانونی که در آن صورت می گیرد، بدنام است در حالی که تعدادی سرویس مشروع نیز در آن وجود دارد.

نگرانی های امنیتی در TOR

سرویس‌های اطلاعات و جاسوسی ایالات متحده و سایر کشورها، شبکه Tor را تهدیدی برای امنیت ملی می‌دانند. آن‌ها معتقدند چنین شبکه ناشناسی، دولت‌ها و تروریست‌های خارجی را قادر می‌سازد تا بدون اینکه دیده شوند ارتباطات مورد نظر خود را برقرار کنند.

به همین دلیل، تعدادی از پروژه‌های تحقیقاتی مختلف بر روی شکستن ناشناس بودن Tor کار می‌کنند. لازم به ذکر است که ناشناس مانندن Tor توسط این مقامات شکسته شده است و احتمالاً دوباره نیز شکسته می‌شود.

نکته: NSA به عنوان نمونه، روترهای Tor خود را اجرا می‌کند و این بدین معنی است که شما هنگام استفاده از Tor ، ممکن است در مسیر روترهای NSA حرکت نموده باشید.

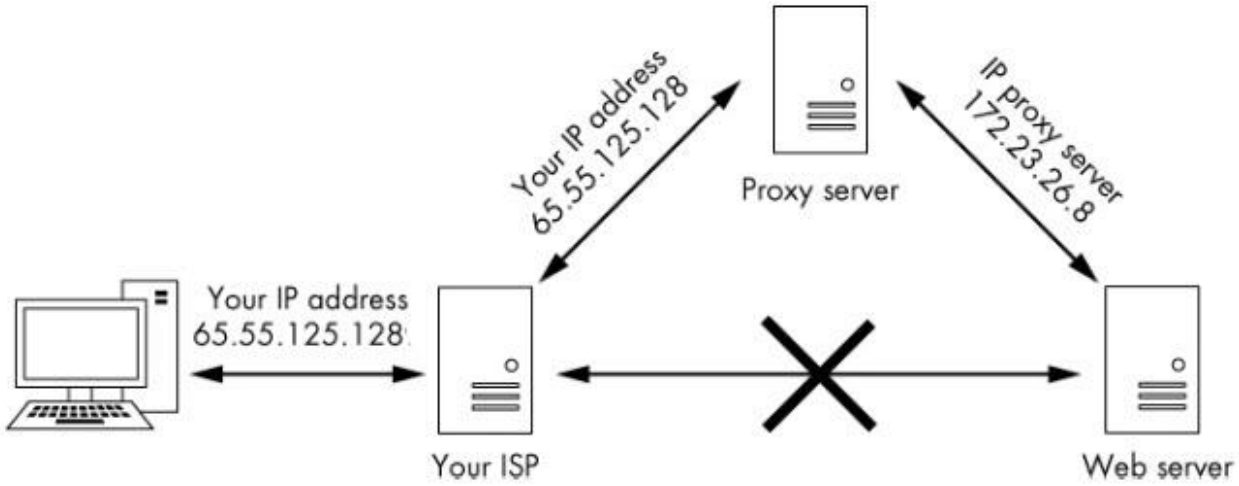
اگر ترافیک شما از روترهای NSA خارج شود، این به مراتب بدتر خواهد بود. چراکه روترهای خروجی همیشه مقصد شما را می‌شناسند.

NSA همچنین روش به نام همبستگی ترافیکی یا Traffic Correlation دارد که شامل جست و جوی الگوهایی در ترافیک ورودی و خروجی می‌باشد که این موضوع می‌تواند ناشناس بودن در Tor را از بین ببرد.

سرورهای پراکسی

استراتژی دیگری که برای ناشناس ماندن در اینترنت از آن بهره برده می‌شود، استفاده از پراکسی‌ها می‌باشد. پراکسی‌ها در واقع سیستم‌های واسطی هستند که ترافیک را از سیستم مبدا دریافت نموده و آن را به سیستم مقصد ارسال می‌کنند.

در این روش، آدرس IP که به عنوان آدرس مبدا، در سیستم نهایی ثبت می‌شود، آدرس سرور پراکسی می‌باشد و آدرس مبدا اصلی در این حالت ناشناس می‌ماند.



البته در نظر داشته باشید که سرور پراکسی ترافیک‌های شما را ثبت می‌کند، اما یک بازرس برای به دست آوردن لاگ‌های این سرور نیاز به حکم قضایی خواهد داشت. برای سخت تر نمودن ردیابی، می‌توانید از زنجیره‌ای از پراکسی استفاده نمایید.

برای همین منظور، سیستم‌عامل کالی لینوکس دارای یک ابزار پراکسی به نام Proxychains می‌باشد. Proxychain یکی دیگر از ابزارهای ناشناس ماندن در لینوکس می‌باشد و نحوه استفاده از این ابزار به صورت زیر می‌باشد:

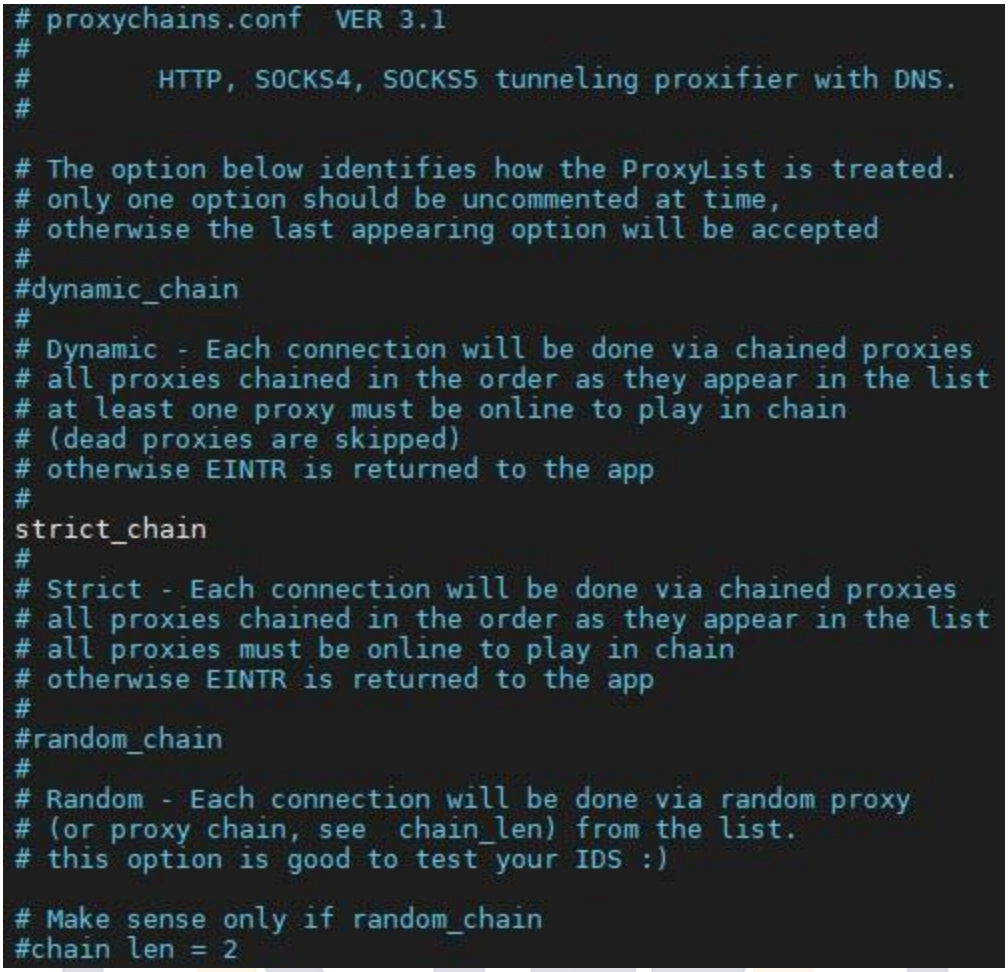
proxychains “the command you want proxied” “arguments”

دستور زیر نمونه‌ای از بکارگیری این ابزار است:

proxychains nmap -sT -Pn 20.0.0.1

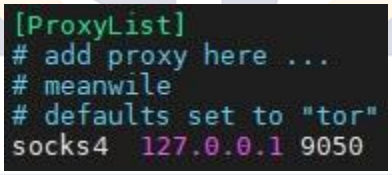
فایل تنظیمات پراکسی

در این بخش ما به نحوه تنظیم پراکسی در proxychains می‌پردازیم. برای این منظور باید با یکی از ویرایشگرهای متنی لینوکس، فایل /etc/proxychains.conf را باز نموده و اقدام به ویرایش آن نماییم:



```
# proxychains.conf  VER 3.1
#
#        HTTP, SOCKS4, SOCKS5 tunneling proxifier with DNS.
#

# The option below identifies how the ProxyList is treated.
# only one option should be uncommented at time,
# otherwise the last appearing option will be accepted
#
#dynamic_chain
#
# Dynamic - Each connection will be done via chained proxies
# all proxies chained in the order as they appear in the list
# at least one proxy must be online to play in chain
# (dead proxies are skipped)
# otherwise EINTR is returned to the app
#
strict_chain
#
# Strict - Each connection will be done via chained proxies
# all proxies chained in the order as they appear in the list
# all proxies must be online to play in chain
# otherwise EINTR is returned to the app
#
#random_chain
#
# Random - Each connection will be done via random proxy
# (or proxy chain, see  chain_len) from the list.
# this option is good to test your IDS :)

# Make sense only if random_chain
#chain_len = 2
```

برای تنظیم آدرس پراکسی شما باید به بخش ProxyList در این فایل مراجعه نمایید:



```
[ProxyList]
# add proxy here ...
# meanwile
# defaults set to "tor"
socks4  127.0.0.1 9050
```

در این بخش ما می‌توانیم آدرس IP و شماره پورت مورد نظر خود را به همراه نوع پراکسی وارد نماییم.

توجه داشته باشید که به صورت پیش فرض در صورت عدم تنظیم پراکسی، proxychainsاز Tor استفاده می‌گردد. همانطور که در انتهای فایل تنظیمات proxychains قابل مشاهده می‌باشد، آدرس ۱۲۷.۰.۰.۱ به همراه شماره پورت ۹۰۵۰ قرار داده شده است که مربوط به تنظیمات پیش فرض Tor می‌باشد.

نگرانی های امنیتی در سرورهای پراکسی

به خاطر داشته باشید که آدرس پراکسی‌های خود را با دانش کافی انتخاب نمایید. اگر قصد ناشناس ماندن را دارید، از پراکسی‌های رایگان استفاده نکنید. در حقیقت پراکسی‌های رایگان به احتمال زیاد آدرس IP شما و تاریخچه فعالیت شما را می‌فروشند. همانطور کهBruce Schneier ، رمزنگار مشهور و کارشناس امنیت، یک بار گفت: "اگر چیزی رایگان باشد، شما مشتری نیستید؛ شما محصول هستید".

به عبارت دیگر هر محصولی که رایگان باشد، احتمالاً اطلاعات شما را جمع آوری نموده و به فروش می‌رساند. اگرچه آدرس IP شما که از پراکسی خارج می‌شود ناشناس باقی می‌ماند، ولی برای آژانس‌های امنیتی و نظارتی، راه‌های دیگری نیز وجود دارد.

به عنوان نمونه، مالک سرور پراکسی، هویت شما را می‌داند در صورت فشار کافی توسط جاسوس‌ها یا سازمان‌های قانونی با احراز صلاحیت، ممکن است هویت شما را برای حفظ شغل خود ارائه نماید.

به همین خاطر مهم است که از محدودیت‌های پراکسی به عنوان یکی از منابع ناشناس بودن، آگاهی داشته باشید.

شبکه های VPN

استفاده از یک شبکه خصوصی مجازی (VPN) می‌تواند روش موثری برای حفاظت از ترافیک شما و ناشناس ماندن نسبی شما باشد. استفاده از VPN می‌تواند امنیت و حریم خصوصی شما را ارتقا بخشد و ترافیک بین شما و سرور خود را رمزنگاری نماید ولی ضمانتی برای ناشناس ماندن شما نخواهد بود.

دستگاه اینترنتی که به آن متصل می‌شوید (مانند سرور یا روتر) باید آدرس IP شما را ثبت نماید تا بتواند داده‌ها را به درستی به شما ارسال نماید، بنابراین هر کسی قادر به دستیابی به این اطلاعات ثبت شده می‌باشد.

یکی از مزایای VPN راه اندازی و اتصال آسان آن می‌باشد. شما می‌توانید از VPN به عنوان یکی از ابزارهای ناشناس ماندن در لینوکس استفاده نمایید. بدین منظور شما می‌توانید بر روی یک سرور، قابلیت VPN را فعال نموده و به آن متصل شوید.

رمزنگاری در ایمیل

سرویس دهندگان رایگان ایمیل مانند یاهو، گوگل و outlook به یک دلیل رایگان هستند، آن‌ها وسیله‌ای برای شناسایی علاقمندی‌های شما و ارائه تبلیغات بر این اساس هستند.

همانطور که پیش از این نیز اشاره گردید، اگر سرویسی رایگان باشد شما در واقع یک محصول هستند و نه یک مشتری. علاوه بر این سرویس‌های ارائه دهند ایمیل، مانند گوگل و مشابه آن، به محتوای رمز نشده شما در ایمیل‌ها دسترسی خواهند داشت و استفاده از HTTPS خللی در این مورد ایجاد نخواهد کرد.

یکی از راه‌های پیشگیری از این موضوع، استفاده از رمزنگاری ایمیل می‌باشد. سرویسProtonMail ، ایمیل شما را به صورت End to End و Browser to Browser رمزنگاری می‌نماید.

این بدان معنی است که ایمیل شما روی سرورهای ProtonMail رمزنگاری شده و حتی مدیران این سرویس نیز نمی‌توانند ایمیل های شما را بخوانند.

سرویس ProtonMail توسط گروهی از محققان و دانشمندان در سوئیس ایجاد شده است. سوئیسی ها دارای سابقه طولانی در زمینه محافظت از اسرار هستند.

سرورهای PtotonMail در اتحادیه اروپا مستقر هستند و دارای قوانین سخت گیرانه تری نسبت به ایالات متحده در خصوص به اشتراک قرار دادن اطلاعات شخصی می‌باشند.

توجه داشته باشید که هنگام تبادل ایمیل با کاربرانی غیر از کاربرانProtonMail ، امکان دارد برخی یا همه ایمیل‌ها رمزنگاری نشوند.

با این تعاریف شما می‌توانید از ProtonMail به عنوان یکی دیگر از ابزارهای ناشناس ماندن در لینوکس استفاده نمایید.

برای کسب اطلاعات بیشتر به وب سایت ProtonMail مراجعه نمایید.

https://protonmail.com

فصل پانزدهم

مدیریت ماژول های کرنل لینوکس

کلیه سیستم‌عامل‌ها حداقل از دو بخش اصلی تشکیل شده‌اند. اولین و مهمترین آن‌ها هسته یا Kernel است. کرنل در مرکز سیستم‌عامل قرار دارد و همه کارهایی که سیستم‌عامل انجام می‌دهد را کنترل می‌کند. این کارها شامل مدیریت حافظه، کنترل CPU و حتی کنترل آنچه کاربر بر روی صفحه مشاهده می‌کند نیز می‌باشد.

بخش دوم سیستم‌عامل اغلب به عنوان محیط کاربر یا User land شناخته می‌شود و تقریباً هر چیز دیگری را در بر می‌گیرد.

کرنل به صورت محافظت شده‌ای طراحی شده است که فقط با استفاده از حساب کاربری root و یا سایر حساب‌های کاربری که دسترسی لازم را دارند، قابل دسترس می‌باشد. این موضوع به دلایل مختلفی درست است، زیرا دسترسی به کرنل می‌تواند برابر با دسترسی نا محدود به سیستم‌عامل باشد.

در نتیجه، اغلب سیستم‌عامل‌ها، تنها امکان دسترسی به بخش User land را برای کاربران فراهم می‌کنند. جایی که کاربر بدون کنترل سیستم‌عامل، تقریباً به هر چیزی که لازم دارد دسترسی خواهد داشت.

در صورت دسترسی به کرنل، به کاربر اجازه داده می‌شود تا نحوه کار، ظاهر و بخش های مختلف سیستم‌عامل را تغییر دهد. همچنین کاربر می‌تواند بخش‌هایی از سیستم‌عامل را تخریب کرده و منجر به غیرقابل اجرا شدن آن شود. با وجود چنین خطراتی، در برخی موارد، مدیر سیستم در صورت دسترسی به کرنل به دلایل عملیاتی و امنیتی، باید همواره دقت و توجه لازم را داشته باشد.

در این فصل ما به چگونگی تغییر در کارکرد کرنل و افزودن ماژول‌های جدید به آن می‌پردازیم. این نکته هم کاملاً مشخص است که اگر نفوذگر بتواند به کرنل دسترسی داشته باشد، قادر به کنترل سیستم‌عامل خواهد بود.

آشنایی با ماژول های کرنل

کرنل در واقع سیستم عصبی مرکزی سیستم‌عامل شماست که کنترل همه چیز از جمله مدیریت تعامل بین اجزای سخت‌افزاری و فراخوانی سرویس‌های لازم را بر عهده دارد.

لینوکس یک کرنل یکپارچه است که امکان افزودن ماژول های کرنل را فراهم می‌کند. به این ترتیب، ماژول‌ها را می‌توان از هسته حذف نموده و یا به آن اضافه نمود.

گاهی کرنل نیاز به بروزرسانی دارد که این امر ممکن است به دلیل نصب درایور یک دستگاه جدید مانند کارت گرافیک، بلوتوث، دستگاه هایUSB ، درایورهای فایل سیستم و حتی پسوندهای سیستم باشد. این درایورها باید در کرنل تعبیه (embedded)شوند تا کاملاً کاربردی باشند.

در برخی سیستم‌ها، برای اضافه نمودن درایورها، شما مجبور هستید تا کل هسته را مجدداً بسازید(rebuild)، کامپایل و یا راه اندازی مجدد نمایید. اما لینوکس این قابلیت را دارد که بدون طی کل این مراحل، برخی از ماژول‌ها را به هسته اضافه نماید. به این ماژول‌ها Loadable Kernel Modulesیا LKM گفته می‌شود.

LKMها بنا به ضرورت به پایین ترین سطح کرنل دسترسی دارند و همین امر آن‌ها را به یک هدف فوق العاده آسیب پذیر برای نفوذگران تبدیل نموده است. یک نوعی از بدافزارها که با نام rootkit شناخته می‌شود، خود را اغلب از طریق LKMها، درون کرنل سیستم‌عامل جاسازی می‌نماید. اگر بدافزار خود را در کرنل جاسازی نماید، نفوذگر می‌تواند کنترل کامل سیستم‌عامل را در دست بگیرد.

در صورتی که یک نفوذگر بتواند ماژول جدیدی (مخرب) را بر روی کرنل بارگذاری نماید، وی نه تنها می‌تواند کنترل سیستم هدف را در دست بگیرد، بلکه به دلیل اینکه در کرنل سیستم‌عامل قرار گرفته است، می‌توانند کنترل موارد دیگر مانند پروسس‌ها، پورت‌ها، سرویس‌ها، فضای هارد دیسک، و تقریباً هر چیزی که شما فکرش را بکنید، در اختیار بگیرد.

به عنوان نمونه، نفوذگر می‌تواند با درخواست از مدیر سیستم لینوکسی جهت اضافه نمودن ماژول یک سخت افزار جدید که در آن rootkit تعبیه شده است، پس از نصب ماژول جدید، کنترل کل سیستم هدف را در اختیار بگیرد. این روشی است که برخی از روت کیت ها از آن استفاده می‌نمایند.

توجه داشته باشید که دانش مناسب در زمینه LKM ها برای داشتن یک مدیریت موثر در لینوکس و پیشگیری از برخی مشکلات و حملات، بسیار حائز اهمیت خواهد بود.

کنترل نسخه کرنل

گام اول در درک کرنل، بررسی نسخه کرنلی است که بر روی سیستم‌عامل شما اجرا شده است. برای بررسی نسخه کرنل دو روش وجود دارد. روش اول استفاده از دستور زیر می‌باشد:

```
root@kali:~# uname -a
Linux kali 4.19.0-kali3-amd64 #1 SMP Debian 4.19.20-1kali1 (2019-02-14) x86_64 GNU/Linux
```

پاسخی که کرنل پس از اجرای این دستور به ما نشان می‌دهد، توزیع سیستم‌عامل در حال اجراست که برابر با Linux Kaliبوده و نسخه کرنل نیز ۴.۱۹ می‌باشد و همچنین معماری سیستم‌عامل نیز ۶۴ بیتی می‌باشد.

همچنین عبارت SMP در خروجی نمایش داده می‌شود که مخفف Symmetric Multi Processing است. وجود این عبارت بدین معنی است که این کرنل می‌تواند در دستگاه‌هایی با چندین هسته و پردازنده کار کند. تاریخ ساخت (build)این کرنل نیز ۱۴ فوریه ۲۰۱۹ می‌باشد. البته ممکن است خروجی دستور بالا در سیستم شما متفاوت باشد.

این اطلاعات می‌تواند هنگام نصب یا load یک درایور کرنل، مورد نیاز باشد.

روش دیگری که برای دریافت اطلاعات کرنل می‌توان از آن استفاده نمود استفاده از دستور زیر است:

```
root@kali:~# cat /proc/version
Linux version 4.19.0-kali3-amd64 (devel@kali.org) (gcc version 8.2.0 (Debian 8.2
.0-16)) #1 SMP Debian 4.19.20-1kali1 (2019-02-14)
```

تنظیم کرنل با دستور sysctl

با دستورات صحیح شما می‌توانید کرنل خود را تنظیم نمایید، بدین معنی که شما می‌توانید تخصیص حافظه را تغییر دهید، ویژگی‌های شبکه را فعال نمایید و حتی کرنل را در برابر حملات خارجی مقاوم نماید. کرنل‌های مربوط به لینوکس‌های مدرن، از دستور sysctl برای تنظیم گزینه های کرنل استفاده می‌کنند.

توجه داشته باشید تغییراتی که شما با دستور sysctl ایجاد می‌کنید فقط تا زمانی که سیستم را راه اندازی مجدد نمایید، تحت تاثیر خواهد بود. برای ایجاد تغییرات پایدار، باید فایل پیکربندی sysctl را که در مسیر /etc/sysctl.conf قرار دارد.

هشدار: هنگام استفاده از sysctl باید مراقب باشید. زیرا بدون داشتن دانش و تجربه کافی، ممکن است به راحتی سیستم خود را غیرقابل استفاده نمایید.


```
kali >sysctl -a | less
dev.cdrom.autoclose = 1
dev.cdrom.autoeject = 0
dev.cdrom.check_media = 0
dev.cdrom.debug = 0
```

دستور بالا شامل خطوط زیادی می‌باشد که مدیران سرورهای لینوکس می‌توانند با تغییر در آن‌ها، اقدام به بهینه سازی کرنل نمایند. در اینجا چند خط وجود دارد که به عنوان یک تست نفوذگر برای شما مفید خواهد بود.

به عنوان نمونه، برای تغییر درsysctl ، ما می‌خواهیم قابلیت Packet Forwarding را فعال نماییم. این قابلیت هنگامی که نفوذگر به یک سیستم نفوذ کرده و قصد انجام حمله MITM یا Man In The Middle را دارد، مورد استفاده قرار می‌گیرد.

برای این منظور دستور sysctl را برای مشاهده بخش‌های مربوط به IP های نسخه ۴ محدود می‌کنیم:

```
root@kali:~# sysctl -a | less | grep ipv4.ip
net.ipv4.ip_default_ttl = 64
net.ipv4.ip_dynaddr = 0
net.ipv4.ip_early_demux = 1
net.ipv4.ip_forward = 0
net.ipv4.ip_forward_update_priority = 1
net.ipv4.ip_forward_use_pmtu = 0
net.ipv4.ip_local_port_range = 32768    60999
net.ipv4.ip_local_reserved_ports =
net.ipv4.ip_no_pmtu_disc = 0
net.ipv4.ip_nonlocal_bind = 0
net.ipv4.ip_unprivileged_port_start = 1024
net.ipv4.ipfrag_high_thresh = 4194304
net.ipv4.ipfrag_low_thresh = 3145728
net.ipv4.ipfrag_max_dist = 64
net.ipv4.ipfrag_secret_interval = 0
net.ipv4.ipfrag_time = 30
```

بخشی که در تصویر بالا با کادر قرمز رنگ مشخص شده است، پارامتری از کرنل می‌باشد که با فعال نمودن آن، ما می‌توانیم قابل Packet Forwarding را انجام دهیم. مقدار پیش فرض برای این پارامتر برابر صفر است که به معنی غیرفعال بودن این پارامتر می‌باشد.

برای فعال نمودن این بخش باید مقدار این پارامتر به یک تغییر کند. برای این منظور از دستور زیر استفاده می‌کنیم:

```
sysctl -w net.ipv4.ip_forward = 1
```


توجه داشته باشید، همانطور که در این بخش هم به آن اشاره گردید، تغییراتی که با دستور sysctl انجام می‌دهید، پس از راه اندازی مجدد سیستم از بین خواهند رفت. برای پایدار نمودن پارامترها، شما باید فایل /etc/sysctl.conf را ویرایش نموده و علامت # موجود در ابتدای خط net.ipv4.ip_forward = 1 را حذف نمایید.

مدیریت ماژول های کرنل

در سیستم‌عامل‌های لینوکس راه‌های مختلفی برای مدیریت ماژول‌های کرنل وجود دارد. یکی از روش‌ها استفاده از گروهی از دستورات insmod که مخفف insert module است، بوده و روش دیگر استفاده از modprobe می‌باشد.

دستور lsmod یکی از دستورات مجموعه insmod می‌باشد که برای لیست نمودن ماژول های نصب شده بر روی کرنل از آن استفاده می‌شود:

```
root@kali:~# lsmod
Module                  Size  Used by
squashfs               65536  4
loop                   36864  8
veth                   24576  0
nf_conntrack_netlink    49152  0
xfrm_user              40960  1
xfrm_algo              16384  1 xfrm_user
br_netfilter           24576  0
overlay               126976  1
bluetooth             643072  0
drbg                   28672  1
ansi_cprng             16384  0
```

این دستور اطلاعات مربوط به ماژول‌های نصب شده به همراه سایز آن‌ها و ماژول‌های دیگری که ممکن است از آن‌ها استفاده نمایند، را مشخص می‌کند.

با استفاده از دستور insmod شما می‌توانید یک ماژول را بارگذاری نموده و با دستور rmmod شما می‌توانید یک ماژول را حذف نمایید.

در نظر داشته باشید که این دستورات کامل نیستند و ممکن است وابستگی‌های ماژول‌ها (module dependencies) را در نظر نگیرند. بنابراین استفاده از آن‌ها می‌تواند کرنل شما را ناپایدار یا غیرقابل استفاده کند.

توزیع‌های مدرن لینوکس اکنون دستور modprobe را اضافه نموده‌اند که به صورت خودکار، وابستگی‌ها را نیز بارگذاری می‌نماید و باعث می‌شود که منجر به بارگذاری و حذف ماژول‌های کرنل با ریسک کمتر خواهد شد.

کمی جلوتر به دستور modprobe خواهیم پرداخت.

مشاهده اطلاعات بیشتر با دستور modinfo

برای کسب اطلاعات بیشتر در خصوص ماژول‌های سطح کرنل، شما می‌توانید از دستور modinfo استفاده نمایید. برای بکارگیری این دستور، کافی است تا نام ماژول مورد نظر خود را پس از این دستور وارد نمایید:

```
root@kali:~# modinfo bluetooth
filename:       /lib/modules/4.19.0-kali3-amd64/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:     1C4A82907AADB971918D7AB
depends:        rfkill,ecdh_generic,crc16
retpoline:      Y
intree:         Y
name:           bluetooth
vermagic:       4.19.0-kali3-amd64 SMP mod_unload modversions
parm:           disable_esco:Disable eSCO connection creation (bool)
parm:           disable_ertm:Disable enhanced retransmission mode (bool)
```

همانطور که در تصویر بالا نیز قابل مشاهده می‌باشد، اطلاعات مربوط به ماژول انتخاب شده که در این جا ماژول Bluetoothمی‌باشد، به شما نمایش داده شده است. در بین این اطلاعات، بخش depends حاوی اطلاعات مربوط به وابستگی‌های ماژول انتخاب شده است که سه ماژول در این بخش به عنوان وابستگی ماژول Bluetooth در نظر گرفته شده است. وابستگی‌ها، ماژول‌هایی هستند که باید نصب شوند تا ماژول مورد نظر، عملکرد درستی داشته باشد.

حذف و اضافه ماژول با دستور modprobe

همانطور که پیشتر نیز به آن اشاره شد، در نسخه مدرن لینوکس مانند کالی لینوکس، دستور modprobe برای مدیریت ماژول‌های کرنل، اضافه گردیده است. به منظور استفاده از این دستور و اضافه نمودن یک ماژول به کرنل، شما می‌توانید از سوییچ a- استفاده نموده و برای حذف یک ماژول نیز از سوییچ r- استفاده نمایید:

```
modprobe -a module_name
modprobe -r module_name
```

فصل شانزدهم

خودکارسازی وظایف در لینوکس

مشابه هر کسی که از لینوکس استفاده می‌کند، تست نفوذگران هم اغلب، اسکریپت‌ها و Job هایی دارند که می‌خواهند به صورت دوره‌ای اجرا شوند. شما ممکن است به عنوان مثال، قصد داشته باشید که به صورت خودکار از فایل‌ها پشتیبان تهیه نمایید یا شاید شما بخواهید تا فایل‌های لاگ را در یک بازه زمانی مشخص rotate نمایید که در بخش‌های پیشین به آن اشاره گردید. همچنین ممکن است شما بخواهید تا یک اسکریپت در یک بازه زمانی مشخص اجرا نمایید. این‌ها مثال‌هایی از Job های خودکار زمانبندی شده می‌باشد.

در این فصل بهjob ها در لینوکس و چگونگی استفاده از سرویس cron و crontab برای تنظیم اسکریپت‌ها جهت اجرای خودکار آن‌ها خواهیم پرداخت.

زمانبندی یه رویداد با Job برای اجرا به صورت خودکار

سرویس cron و crontab از ابزارهای کاربردی برای زمانبندی وظایف می‌باشندcrond ، یک سرویسی (deamon) است که در پس زمینه اجرا می‌شود. سرویسcron ، جدول cron را که دستورات آن در زمان‌های معین اجرا می‌شود را بررسی می‌کند. ما می‌توانیم این جدول را تغییر دهیم تا یک وظیفه یا Job برای زمان یا تاریخ مشخصی اجرا شود.

به منظور برنامه ریزی وظایف و Job ها، شما باید آن‌ها را در فایل جدول cron که در مسیر /etc/crontab قرار دارد وارد نمایید. جدول cron دارای هفت فیلد است. از ۵ مورد اول برای برنامه ریزی زمان اجرای وظایف استفاده شده، بخش ششم کاربر را مشخص نموده و فیلد آخر برای مشخص نمودن مسیری که مربوط به دستور مورد نظر جهت اجرا می‌باشد، استفاده می‌شود.

در صورتی که بخواهید از جدول cron برای زمان بندی اجرای یک Script استفاده کنید، شما به راحتی می‌توانید این کار را با قرار دادن مسیر اسکریپت در بخش هفتم این جدول انجام دهید.

هر یک از ۵ قسمت ابتدایی جدول cron یک عنصر متفاوت از زمان را نشان می‌دهد که شامل دقیقه، ساعت، روز ماه، ماه و روز هفته است. هر بخش از زمان باید به صورت عددی نشان داده شود، بنابراین ماه مارس با عدد سه نشان داده می‌شود. روزهای هفته نیز از صفر شروع می‌شود که اولین روز هفته نیز یکشنبه می‌باشد.

Field	Time unit	Representation
1	Minute	0–59
2	Hour	0–23
3	Day of the month	1–31
4	Month	1–12
5	Day of the week	0–7

ستون‌های موجود در فایل crontab دارای برچسب‌هایی است که شما را راهنمایی می‌نماید:

```
M H DOM MON DOW USER COMMAND
30 2 * * 1-5 root /root/myscanningscript
```

در مثال بالا عدد ۳۰ بیانگر دقیقه، عدد ۲ بیانگر ساعت، عدد ۱-۵ بیانگر روزهای دوشنبه تا جمعه، عبارت root بیانگر کاربری است که تعریف می‌گردد و بخش انتهایی نیز مسیر اسکریپت مورد نظر را نمایش می‌دهد.

همچنین بخش سوم و چهارم که با ستاره مشخص شده است، بیانگر این موضوع است که ما قصد داریم تا اسکریپت مورد نظر، همه روزهای دوشنبه تا جمعه هر ماه اجرا شود.

برای ویرایش crontab شما می‌توانید دستور crontab را با سوییچ e- اجرا نمایید. علاوه بر این شما می‌توانید از دستور زیر برای ویرایش crontab استفاده نمایید:

```
nano /etc/crontab
```

```
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# Example of job definition:
# .---------------- minute (0 - 59)
# |  .------------- hour (0 - 23)
# |  |  .---------- day of month (1 - 31)
# |  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
# |  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# |  |  |  |  |
# *  *  *  *  * user-name command to be executed
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
```

حالا شما کافی است تا برای ایجاد یک وظیفه یاjob ، تنها یک خط با ویژگی‌هایی که در همین بخش به آن اشاره شد، ایجاد نموده و به انتهای فایل crontab اضافه نمایید.

زمانبندی برای تهیه Backup

بیایید در ابتدا، این ابزار را از منظر مدیر سیستم بررسی نماییم. به عنوان مدیر سیستم، شما اغلب می‌خواهید تا در یک زمان مشخص که سیستم کمترین استفاده را دارد و منابع آزاد هستند، از فایل‌های خود نسخه پشتیبان تهیه نمایید. یکی از زمان‌های ایده آل برای این منظور، نیمه شب آخر هفته می‌باشد.

توجه داشته باشید که فیلد ساعت، به جای اینکه به دو بخش AM و PM تقسیم بندی شود، دارای ۲۴ ساعت می‌باشد. بنابراین ساعت ۱ عصر یا همان PM برابر با ساعت ۱۳ خواهد بود. همچنین روزهای هفته (DOW) نیز از یکشنبه (برابر عدد صفر) شروع شده و تا شنبه (برابر عدد ۶) ادامه دارد.

برای ایجاد یک Job شما می‌بایست یک خط را به انتهای فایل crontab با قالب مشخص شده که در بخش پیشین نیز توضیح داده شد، اضافه نمایید.

فرض کنید که شما می‌خواهید با استفاده از یک حساب کاربری با نامbackup ، از کلیه فایل‌های خود نسخه پشتیبان تهیه نمایید. همچنین شما یک اسکریپت را برای این منظور آماده نموده‌اید که عملیات پشتیبان گیری را برای شما انجام دهد. نام این فایل systembsckup.sh بوده و در دایرکتوری /bin ذخیره شده است.

شما می‌خواهید که عملیات پشتیبان گیری ساعت ۲ نیمه شب روز یکشنبه (شنبه شب) در ماه انجام گیرد. بدین منظور شما باید خط زیر را در انتهای فایل crontab اضافه نمایید:

۰۰ ۲ * * ۰ backup /bin/systembackup.sh

استفاده از crontab برای اجرای اسکریپت شما

پس از درک ابتدایی مفاهیم زمانبندی یکJob ، در ادامه، اسکریپتی که در بخش‌های پیشین ایجاد نموده بودیم را به صورت یک Job تعریف می‌نماییم. اگر به خاطر داشته باشید، این اسکریپت، به منظور اسکن پورت ۳۳۰۶ که مربوط به سرویس MySQL می‌باشد، طراحی شده بود. برای ایجاد یک Job ما کافی است تا خط زیر را به انتهای فایل crontab اضافه نماییم:

۰۰ ۹ * * * user /bin/MySQLScanner.sh

اسکریپت مورد نظر ما، در ساعت ۹ هر روز از ماه، در کلیه ماه‌های سال و روزهای هفته، با استفاده از حساب کاربری user اجرا خواهد شد.

دستورات میانبر برای crontab

علاوه بر ساختار نوشتاری که می‌توانید در فایل crontab از آن استفاده نمایید، shortcutهایی نیز به صورت پیش فرض وجود دارد که از آن نیز می‌توان برای ایجاد یک job استفاده نمود. این shortcut ها عبارتند از:

@yearly

@annually

@monthly

@weekly

@daily

@midnight

@noon

@reboot

بنابراین اگر بخواهید که اسکریپت اسکن پورت شما، هر شب و در نیمه شب اجرا گردد، می‌توانید خط زیر را در انتهای فایل crontab اضافه نمایید:

@midnight user /bin/MySQLScanner.sh

اجرای job در Startup

هنگامی که شما سیستم لینوکسی خود را روشن می‌کنید، تعدادی اسکریپت برای تنظیم محیط کاری شما، اجرا می‌گردد. این اسکریپت‌ها به اسکریپت‌های rc معروف هستند.

پس از اینکه کرنل، تمامی ماژول‌های خود را بارگذاری نمود، اقدام به Start سرویسی (Daemon) می‌نماید که به init یا init.d معروف است. این Daemon سپس شروع به اجرای تعدادی از اسکریپت‌های یافت شده در مسیر /etc/inin.d/rc می‌نماید.

این اسکریپت‌ها شامل دستوراتی است که برای Start بسیاری از سرویس‌های لازم، جهت اجرای سیستم شما، از آن‌ها استفاده می‌شود.

Runlevel ها در لینوکس

لینوکس دارای چدین Runlevel می‌باشد که نشان می‌دهد، چه سرویس‌هایی باید هنگام Boot شدن سیستم، Start شوند. به عنوان مثال، Runlevel 1حالت Single User Mode بوده و سرویس‌هایی مانند سرویس Networking در این Runlevel ، Startنخواهند شد. اسکریپت‌هایrc ، بسته به اینکه شما چه Runlevel ای را انتخاب نموده اید، اجرا خواهند شد:

0 Halt the system

1 Single-user/minimal mode

2–5 Multiuser modes

6 Reboot the system

کوتاه در مورد Runlevel ها

به مفهوم ساده، Run Levelبه معنی سطح اجرا می‌باشد، سطح اجرای اسکریپت‌ها و نرم افزارهایی که برای یک کاربر در سیستم‌عامل لینوکس می‌تواند اجرا شود.

شما اگر سیستم‌عاملی دارید که فقط یک کاربر می‌تواند از آن استفاده کند در واقع Run Level ای که دارد تعریف کننده این هست که این سیستم‌عامل Single User است. در سیستم‌عامل لینوکس ما با تعریف کردن سطح اجرا یا Run Level نوع استفاده از یک سیستم‌عامل را تعریف می‌کنیم.

اینکه سیستم‌عامل ما گرافیکی بوت شود یا اینکه دسکتاپ گرافیکی داشته باشد در اسکریپت‌هایی که در Run Level های لینوکس قرار دارد تعریف می‌شود.

ساختار این Run Level ها بسیار ساده است. در واقع Run Level یک پوشه است که داخل آن یک سری اسکریپت وجود دارد. هر فردی با توجه به نیاز خود یکی از این پوشه‌ها را انتخاب می‌کند و تمامی اسکریپت‌های داخل آن اجرا می‌شود و این یعنی Run Level شما تغییر کرده است.

Run Levelهای لینوکس از شماره صفر شروع می‌شوند و به شماره ۶ ختم می‌شوند. شماره صفر به معنی Shutdown یا Poweroff در سیستم‌عامل است. اگر شما این Run Level را فراخوانی کنید سیستم شما بلافاصله خاموش می‌شود.

Run Levelشماره ۱ زمانی استفاده می‌شود که شما یک سیستم‌عامل تک کاربره یا Single User می‌خواهید. البته در برخی از سیستم‌عامل‌ها به این Run Level ریکاوری یا Recovery Mode هم می‌گویند که همانطور که از اسمش پیداست برای مصارف تعمیر کردن سیستم‌عامل استفاده می‌شود.

Run Levelشماره ۲ رابط کاربری Text Mode است اما با این تفاوت که Multi User است و همزمان چند کاربر می توانند از آن استفاده کنند.

Run Levelشماره ۳ هم Multi User است و همانند Run Level شماره ۲ است با این تفاوت که قابلیت‌های Networkingهم داخل آن تعبیه شده است.

Run Levelشماره ۴ کاملا بلااستفاده است و در واقع شما می‌توانید این Run Level را دلخواه سازی کنید و برای خودتان یک Run Level اختصاصی ایجاد کنید.

Run Levelشماره ۵ رابط گرافیکی دارد و شما وقتی دارید از دسکتاپی به نام KDE استفاده می‌کنید در واقع از این نوع Run Level استفاده می‌کنید.

Run Levelشماره ۶ هم برای Reboot کردن سیستم است و شما می‌توانید با فراخوانی اسکریپ‌های آن سیستم را Rebootکنید.

برخی از مباحث این بخش **برگرفته از مقاله جناب آقای مهندس نصیری در سایت توسینسو** می‌باشد.

https://linux.tosinso.com/fa/tips/32493

اضافه نمودن سرویس به rc.d

شما می‌توانید سرویس‌های مد نظر خود را برای اجرا درهنگام بوت شدن سیستم‌عامل، اضافه نمایید. این کار با استفاده از دستور update-rc.d انجام می‌شود. شما به راحتی می‌توانید از این دستور استفاده نموده و پس از آن نام سرویس مورد نظر را وارد نمایید و سپس ویژگی آن را در ادامه مشخص نمایید.

به عنوان مثال، فرض کنید که شما می‌خواهید سرویس PostgreSQL همواره پس از راه اندازی مجدد سیستم، Start شود. در این حالت شما با استفاده از دستورuodate-rc.d ، اقدام به افزودن یک خط به اسکریپت rc.d خود می‌نمایید تا با هر بار راه اندازی مجدد سیستم، سرویس مورد نظر نیز Start شود.

حالا با استفاده ازupdate-rc.d ، اسکریپت rc.d خودمان را بروزرسانی می‌کنیم تا سرویس PostgreSQL به صورت خودکار پس از راه اندازی مجدد سیستم، Startشود:

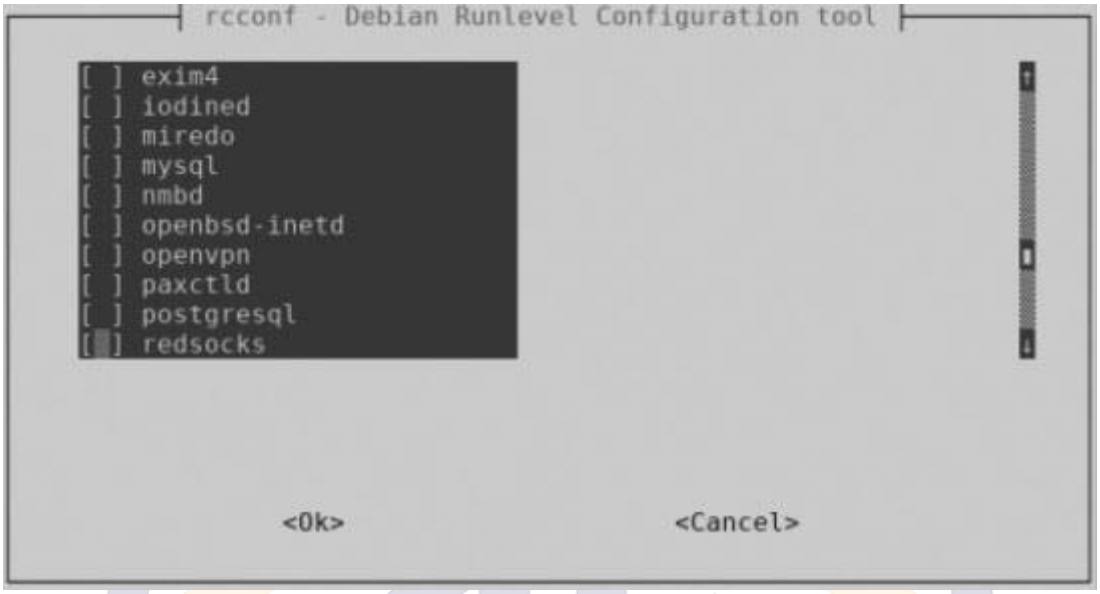
```
update-rc.d postgresql defaults
```

اضافه نمودن سرویس به صورت گرافیکی

اگر شما برای اضافه نمودن سرویس‌ها در Startup سیستم لینوکس، ابزارهای گرافیکی را ترجیح می‌دهید، می‌توانید ابزار rcconf که در مخازن کالی نیز وجود دارد را دانلود نمایید. بدین منظور می‌توانید از دستور زیر استفاده نمایید:

```
apt-get install rcconf
```

پس از تکمیل نصب rcconf شما می‌توانید از دستور rcconf برای اجرای این ابزار استفاده نمایید.



همانطور که در تصویر بالا قابل مشاهده می‌باشد، شما می‌توانید با Scroll نمودن صفحه، سرویس مورد نظر خود را پیدا نموده و با انتخاب آن و همچنین کلیک بر روی دکمهOK ، سرویس مورد نظر خود را در Startup قرار دهید.