Linux Host Security

Tuning the Linux Kernel with Sysctl

Eghbal aminejad

LINUX AUTHOR AND TRAINER

github: @eqba1

Tuning the linux kernel with sysctl

- Course Objectives

- **The procfs in Linux**
- **Using the command sysctl**
- **Understanding ASLR**
- **Tuning ICMP settings**

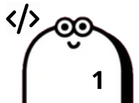
Weight

```
$ grep '^proc' /proc/self/mounts

$ find -L /etc /proc -maxdepth 1 -samefile /proc/self/mounts
```

The procfs

procfs یک فایل سیستم مجازی است که در مسیر /proc مستقر می شود و حاوی وضعیت جاری سیستم لینوکسی است. بیشتر محتوای آن فقط خواندنی است اما بعضی را نیز می توان تنظیم کرد. نصب شدن فایل سیستم که بخشی از وضعیت اجرایی سیستم را تشکیل میدهد در مسیر /proc/self/mounts لیست شده است. این فایل لینک شده به دو فایل دیگر در مسیرهای /etc/mtab و /proc/mounts است .



```
$ man 5 proc
```

Finding Help on /proc

برای راهنمایی بیشتر و کامل تر در مورد این دایرکتوری با دستور man می‌توانید دایکومنت کامل را مشاهده کنید.

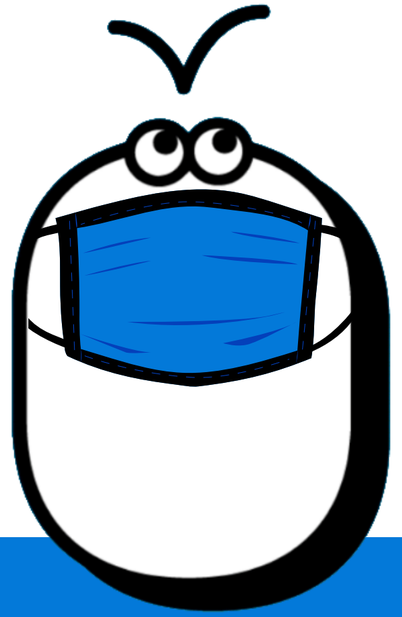
```
$ sudo apt install tree

$ tree -L 1 /proc/sys
```

Configuration Below /proc/sys

لیست پارامترهای قابل کنترل توسط ما در مسیر /proc/sys قرار دارد. تعداد زیادی فایل برای تنظیمات در این قسمت وجود دارد اما میتوان لیست دایرکتوری ها را به صورت یک درخت مرتب و هر اندازه دلخواه با استفاده از دستور tree مشاهده کرد تا یک ایده کلی از ساختار پارامترها داشته باشیم.

Let's drop out to the command line to investigate the procfs



Demo

```
$ cat /proc/sys/kernel/domainname

$ sysctl -a

$ sysctl -ar domainname

$ sysctl -anr domainname
```

Reading Values with sysctl

پیکربندی تنظیمات در proc/sys/ را میتوان از خود فایلها خواند اما این فایلها و همچنین فایل etc/sysctl.conf/ طراحی شده اند تا با استفاده از دستور sysctl کنترل شوند.
نکته: عبارت domainname که در این مثال ها استفاده میکنیم برای سرویس NIS Domain Name است. هرچند در لینوکس ها مدرن از آن استفاده نمی‌شود.

```
$ echo "example" | sudo tee /proc/sys/kernel/domainname

$ sudo sysctl -w kernel.domainname='example'
```

Writing Values with sysctl

تمام کلیدواژه که قابلیت نوشتن دارند را میتوان به صورت مستقیم از طریق فایل مقداردهی کرد اما مرسوم ترین روش کنترل کردن آنها با استفاده از دستور sysctl است. مقدار دهی کردن این عبارت ها نیاز به دسترسی ادمین دارد.

```
$ echo 'kernel.domainname=example' | sudo tee /etc/sysctl.d/60-nis-domain.conf
$ sudo sysctl -p /etc/sysctl.d/60-nis-domain.conf
```

sudo sysctl -p reads and configures /etc/sysctl.conf

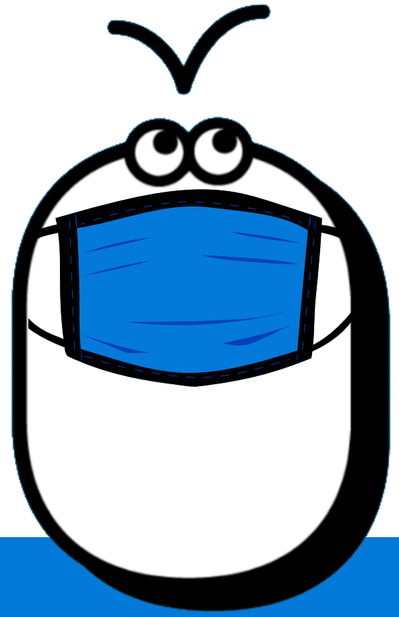
sudo sysctl -p <filename> just the named file

sudo sysctl --system emulates system startup reading all

Persisting Values with /etc/sysctl.conf

هر تنظیماتی که بر روی فایل ها چه به صورت مستقیم چه با استفاده از دستور sysctl -w انجام دهیم تاثیر موقتی دارند و با راه اندازی مجدد سیستم، از بین خواهد رفت. در صورتی که میخواهیم تنظیمات اعمال شده مداومت داشته باشه باید در فایل /etc/sysctl.conf یا به دایرکتوری /etc/sysctl.d اضافه گردد.

We can now visit the
command line to review how
we read and write to the procfs



Demo

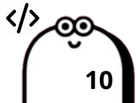
Address Space Layout Randomization

مخفف ASLR. یکی از ویژگی های کرنل لینوکس است که در سال ۲۰۰۵ اضافه شد. وجود ASLR به ما این اطمینان را میدهد که محل قرار گیری باینری های سیستم در حافظه به زمان اجرای آنها نیز بستگی داشته باشد. یعنی با هربار اجرای یک برنامه محل ذخیره سازی آن در حافظه RAM به صورتی تصادفی اتفاق بیفتد. بدون این قابلیت، باعث اختصاص خانه های حافظه یکسان برای اجرای یک برنامه شده که باعث آسیب پذیری بافر اورفلو و دیگر حملات سطح باینری RAM را به نفوذ گرها خواهد شد.

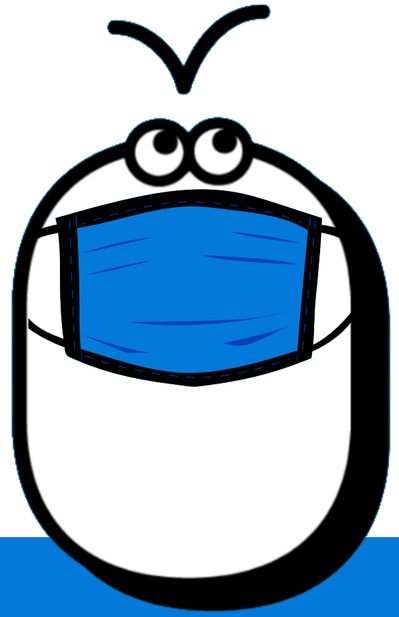
```
$ sysctl -ar randomize
$ ldd /bin/bash
$ ldd /bin/bash
$ sudo sysctl -w kernel.randomize_va_space=0
$ ldd /bin/bash
$ ldd /bin/bash
```

ASLR

به صورت پیشفرت قابلیت ASLR در بیشتر سیستم های لینوکسی فعال است. به علاوه، قابلیت PIE یا Position independent Executable به صورت پیشفرص در ماژولهای کرنل قرار دارد. با استفاده از دستور ldd میتوان ماژول های مشترک شده برای یک اپلیکیشن را دید. همچنین این دستور آدرس حافظه استفاده شده را نیز نمایش میدهد.



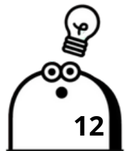
Observing the importance of ASLR



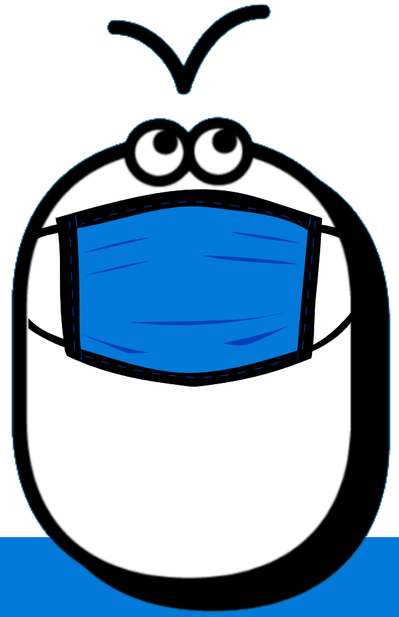
Demo

Obscuring Server from Pings

اگر یک سرور به درخواست های ping پاسخ دهد، ICMP را برگرداند، به سادگی میتواند سرور را در یک شبکه پیدا کرد. پس وقتی برای ما ساده باشد برای تخریبگران شبکه نیز بسیار ساده خواهد بود. به صورت پیشفرض broadcasts کردن پکت ICMP غیر فعال است اما unicast آن نیاز به غیر فعال کردن دارد. میتوان این عمل را از طریق دیوار آتش و sysctl انجام داد. که فعلا به مبحث دیوار آتش نمی پردازیم.



Securing ICMP requests



Demo

summary

- **Understanding the procfs**
- **Reading, writing, and persisting configuration with sysctl**
- **Address Space Layout Randomization (ASLR)**
- **Blocking ICMP requests to a system**

Weight

Next up

Monitoring Ports and Services

Eghbal aminejad

LINUX AUTHOR AND TRAINER

github: @eqba1