Welcome to

9. Breaking Out

KEA Kompetence Computer Systems Security 2023

Henrik Kramselund he/him han/ham xhek@kea.dk @kramse

Slides are available as PDF, kramse@Github

9-breaking-out.tex in the repo security-courses

Goals for today

Todays goals:

- Discuss Kernel Hardening and Process Isolation
- Methods for keeping systems secure – Lynis
- Patch Management
- Rowhammer – depending on your interest

Photo by Thomas Galler on Unsplash

Subjects

- MLSH chapter 11: Discuss Kernel Hardening and Process Isolation
- Methods for keeping systems secure – Lynis
- DSH chapter 16: Patch Management
- Rowhammer – if we got time, but related to sandboxing, capabilities and VM escape

Exercises

- Lynis
- DDoS testing

MLSH 11: Kernel Hardening and Process Isolation

DSH chapter 16: Vulnerability Management

Browse: Using Memory Errors to Attack a Virtual Machine paper, An Experimental Study of DRAM Disturbance Errors, Exploiting the DRAM rowhammer bug to gain kernel privileges https://en.wikipedia.org/wiki/Row_hammer

Kernel Hardening and Process Isolation

Although the Linux kernel is already fairly secure by design, there are still a few ways to lock it down even more. It's simple to do, once you know what to look for. Tweaking the kernel can help prevent certain network attacks and certain types of information leaks. (But fear not – you don't have to recompile a whole new kernel to take advantage of this.)

Source: *Mastering Linux Security and Hardening* (MLSH), third edition

- Secure by design might not be my wording, well understood and mature might be better
- A lot of more secure designs exist

MLSH Chapter 11: Kernel Hardening and Process Isolation

With process isolation, our aim is to prevent malicious users from performing either a **vertical** or a **horizontal privilege escalation**. By isolating processes from each other, we can help prevent someone from taking control of either a **root user process** or a process that belongs to some other user. Either of these types of privilege escalation could help an attacker either **take control of a system** or **access sensitive information**.

Source: *Mastering Linux Security and Hardening* (MLSH), third edition

- Remember the CIA model, but most likely if there are cracks an attacker can exploit it

MLSH Chapter 11: Topics covered in this chapter

In this chapter, we'll take a quick tour of the /proc filesystem ...

- Understanding the /proc filesystem
- Setting kernel parameters with sysctl / Configuring the sysctl.conf file
- An overview of process isolation
- Control groups / Namespace isolation / Kernel capabilities
- SECCOMP and system calls
- Using process isolation with Docker containers
- Sandboxing with Firejail
- Sandboxing with SnappyKernel Hardening and Process Isolation
- Sandboxing with Flatpak

Source: *Mastering Linux Security and Hardening* (MLSH), third edition

We will not go into all the details from this chapter

MLSH Chapter 11: Linux /proc

Understanding the /proc filesystem

If you cd into the /proc/ directory of any Linux distro and take a look around, you'll be excused for thinking that there's nothing special about it. You'll see files and directories, so it looks like it could just be another directory. In reality, though, it's very special. It's one of several different **pseudo-filesystems** on the Linux system. (The definition of the word pseudo is fake, so you can also think of it as a *fake* filesystem.)

Source: *Mastering Linux Security and Hardening* (MLSH), third edition

Can we spot the:

- In Unix *everything* is a file
- Init was the old *System V init* with *run-levels*
- Systemd is used in most Linux distributions today
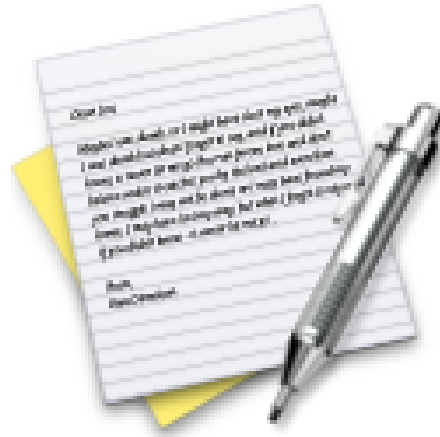- Other Unix systems, like BSD still use an init process

MLSH Chapter 11: Configuring the sysctl.conf file

If you look in the /etc/sysctl.d/10-network-security.conf file, you'll see it enabled there. So, there's no need to uncomment these two lines. Next, we see this:

```
# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1
```

Source: *Mastering Linux Security and Hardening* (MLSH), third edition

- The main file used to be `/etc/sysctl.conf`, check on your various distributions
- Today I mostly use Ansible to configure this
- You can read more about SYN cookies at: https://en.wikipedia.org/wiki/SYN_cookies
- In older times we also tuned our TCP and UDP memory space, but mostly use default values today

Now lets do the exercise

⚠ Lynis Auditing, System hardening, and Compliance testing

which is number **37** in the exercise PDF.

Understanding Control Groups (cgroups)

Control Groups, more commonly called **cgroups**, were introduced back in 2010 in Red Hat Enterprise Linux 6. Originally, they were just an add-on feature, and a user had to jump through some hoops to manually create them. Nowadays, with the advent of the systemd init system, cgroups are an integral part of the operating system, and each process runs in its own cgroup by default.

Source: *Mastering Linux Security and Hardening* (MLSH), third edition

- Docker and other containers use cgroups along with namespaces to provide isolation and resource constraints

Understanding namespace isolation

Namespaces are a kernel security feature that was introduced in Linux kernel version 2.4.19, all the way back in 2002. A namespace allows a process to have its own set of computer resources that other processes can't see. They're especially handy for times when you might have multiple customers sharing resources on the same server. The processes for each user will have their own namespaces.

Source: *Mastering Linux Security and Hardening* (MLSH), third edition

- Kubernetes also use and provide namespaces as a concept to isolate applications

Understanding kernel capabilities

However, having services run with full root privileges can be a bit of a security problem. For- tunately, there are some ways to mitigate that.

For example, any web server service, such as Apache or NGINX, needs to **start with root privileges** in order to **bind to ports 80 and 443** , which are **privileged ports**. However, both Apache and Nginx mitigate this problem by either dropping root privileges once the service has started or by **spawning child processes** that belong to a **non-privileged user**.

...

Capabilities allow the Linux kernel to divide what the root user can do into distinct units.

Source: *Mastering Linux Security and Hardening* (MLSH), third edition

- Book uses example capabilities with Python 2: `python2 -m SimpleHTTPServer 80`
- I recommend using only Python 3 now, and instead would use: `python3 -m http.server 8000`
- Example with capabilities for ping can be compared to running setuid in the old days

Availability Policies

An availability policy ensures that a resource or service can be accessed in some way in a timely fashion

Often expressed as *quality of service*

Denial of service occurs when this resource or service becomes unavailable

Fairness and starvation

Fairness policy prevents starvation, often rephrased as - process will make progress

If one process gets all resources, memory, cpu, network the others will starve - not have enough resources to progress

Compare to old operating systems Windows 3 / Mac OS 9
Cooperative multitasking vs pre-emptive multitasking

DSH Chapter 16: Vulnerability Management

Contrary to what some vendors' marketing material would have us believe, **a huge quantity of successful breaches do not occur because of complex 0-day vulnerabilities**, lovingly handcrafted by artisanal exploit writers. Although this does happen, a **lack of patching**, failure to **follow good practices** for configuration, or **neglecting** to change **default passwords** are to blame for a far **larger number of successful attacks** against corporate environments. Even those capable of deploying tailor-made exploits against your infrastructure will prefer to make use of these types of vulnerabilities.

Vulnerability management is the terminology used to describe the overall program of activities that oversees vulnerability scanning and detection through to remediation. This is a program that ultimately raises the security of your network by removing potential flaws.

Source: *Defensive Security Handbook* (DSH), Lee Brotherston, Amanda Berlin ISBN: 978-1-491-96038-7
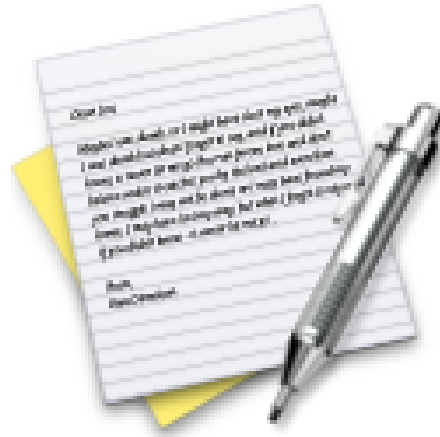
- Lynis is a nice example of a tool that can look into configuration also
- Patch management is a task that should be prioritized

Availability and Network flooding attacks

- SYN flood is the most basic and very common on the internet towards 80/tcp and 443/tcp
- ICMP and UDP flooding are the next targets
- Supporting litterature is TCP Synfloods - an old yet current problem, and improving pf's response to it, Henning Brauer, BSDCan 2017
- All of them try to use up some resources
- Memory space in specific sections of the kernel, TCP state, firewalls state, number of concurrent sessions/connections
- interrupt processing of packets - packets per second
- CPU processing in firewalls, pps
- CPU processing in server software
- Bandwidth - megabits per second mbps

See also DDoS protection with low level technical measures to implement at

https://github.com/kramse/security-courses/tree/master/presentations/network/ddos-test-troopers22

kea

Now lets do the exercise

🛈 SYN flooding 101 - 60min

which is number **38** in the exercise PDF.

Rowhammer

Sorry, I referenced too much with papers and web sites

- It is enough to browse the wikipedia page! https://en.wikipedia.org/wiki/Row_hammer
- *Using Memory Errors to Attack a Virtual Machine paper* by Sudhakar Govindavajhala and Andrew Appel `memerr.pdf`
- *Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors* Yoongu Kim, Ross Daly, Jeremie Kim, Chris Fallin, Ji Hye Lee, Donghyuk Lee, Chris Wilkerson, Konrad Lai, Onur Mutlu `kim-isca14-flipping-bits.pdf`
- *Exploiting the DRAM rowhammer bug to gain kernel privileges*
  `Project Zero_ Exploiting the DRAM rowhammer bug to gain kernel privileges.pdf`

You should recognize the name rowhammer, see the complexity and remember that such complex attacks do exist too!

For Next Time
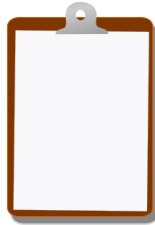
Think about the subjects from this time, write down questions

Check the plan for chapters to read in the books

Visit web sites and download papers if needed

Retry the exercises to get more confident using the tools