

Terminating in the name of eBPF

Shutting down Podman containers
offending against Seccomp

Tuebix 2023

Lightning Talk by Cedric Casper

What's the goal?

Taking Seccomp security a bit further

- Podman containers can be started with Seccomp profiles
 - Nice way to reduce attack surface
- When a process inside the container offends against the Seccomp profile...
 - ...not only the process should be terminated...
 - *...but also the whole container!*

Why?

Taking Seccomp security a bit further

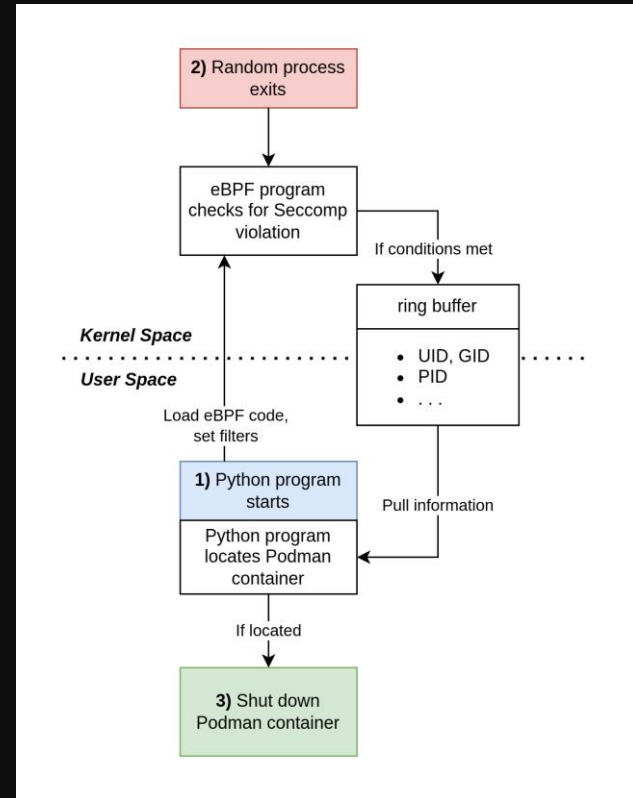
- Currently only processes inside the container are blocked by Seccomp
 - Parent process is still running...
 - ...and so, the container is still running.
- Malware or attacker inside the container could still try out other attack methods, allowed System Calls, overseen System Calls, ...
- We wanted the container and therefore potential threats to be shut down for good.

One way of doing it

Using eBPF to spot Seccomp offences

- Writing a program, using Python and BCC [1]
- Tracing exited processes and filtering/checking:
 - How did it exit? (Checking for Signal 31 and abnormal exit)
 - Did it use Seccomp? (`exited_process->seccomp.mode == 2` or 3)
- Getting info on the container:
 - Container owner UID = owner of the Seccomp violating process' PID namespace
 - PID namespace ID as container identifier
- Shutting down the container:
 - Sudo-ing into the user, listing Podman containers filtered by previously determined PID namespace ID
 - -> `podman kill <container ID>`

[1] <https://github.com/lovisor/bcc> (Great source of inspiration for eBPF programs)



Program flow

Result

- Bonus filters and infos:
 - Only shutdown container(s) of certain users or groups
 - Only shutdown certain container(s)
 - Get command that violated the Seccomp profile -> adjusting/debugging the Seccomp profile
 - And probably many more...
- After the Seccomp offence inside a container, that container is shut down
- -> Threat is reduced, the incident can be investigated

```
ubuntu@hfu-demo:~$ sudo ./seccompSurv.py -v -u 1003
# Preparing eBPF program...
# Done preparing eBPF prog.
# Start tracing seccomp violations...
# Mode: pidns

# Only monitoring containers of user: 1003

# Potential Seccomp violation spotted.
# Issued command: rm
# Real UID and GID: 1003 1003
# Mapped UID and GID: 1003 1003
# Looking for user with UID: 1003
# User with UID 1003 has username john_titor
# Found container(s):
# ['e4039a806ec5']
# Stopping container(s) with ID(s):
# ['e4039a806ec5']
# Looking for user with UID: 1003
# User with UID 1003 has username john_titor
# Killed podman container(s) with ID(s) e4039a806ec5

john_titor@hfu-demo:~$ podman run --security-opt seccomp=seccomp_profiles/bash.json -ti fedora:30 bash
[root@e4039a806ec5 /]# ls
bin  etc  lib64  mnt  root  srv  usr
boot  home  lost+found  opt  run  sys  var
dev  lib  media  proc  sbin  tmp
[root@e4039a806ec5 /]# cat /etc/fedora-release
Fedora release 30 (Thirty)
[root@e4039a806ec5 /]# rm /etc/fedora-release
rm: remove symbolic link '/etc/fedora-release'? y
Bad system call (core dumped)
[root@e4039a806ec5 /]# john_titor@hfu-demo:~$
```

Example of a spotted Seccomp offence, followed by container shutdown

Code available on Github:

<https://github.com/hashkeks/seccompSurv>

Thank you!

For more information feel free to write me:

info@cedriccasper.com