9001 ways
to break out of a container

josephine pfeiffer, 08/2024

itinerary

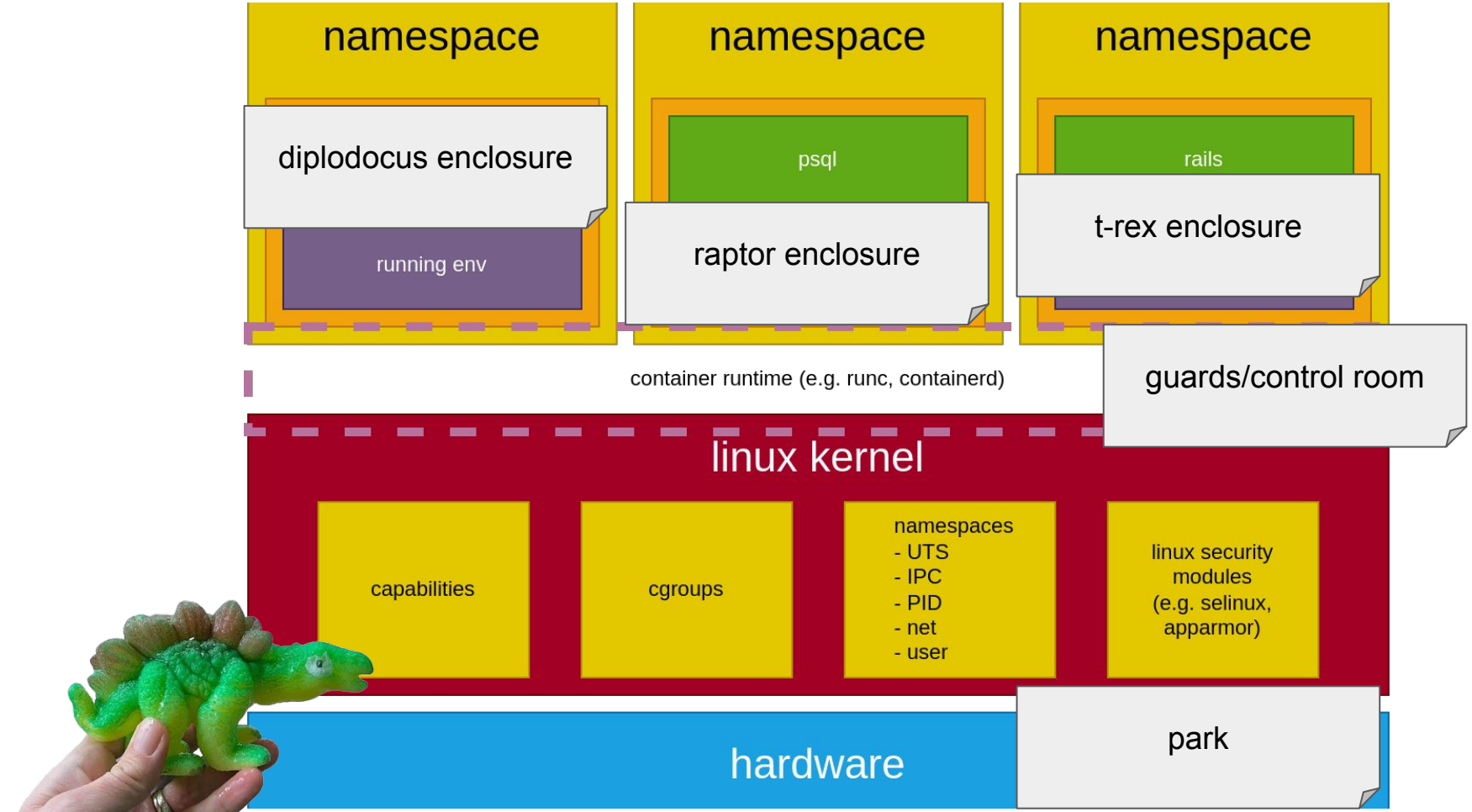
- refresher on container concepts
- low-level linux stuff for host separation
  - cgroups, namespaces, container runtimes, processes…
- types of container escapes
  - filesystem, memory, kernel, runtime…
- real world examples
  - eBPF linux kernel vulnerability (demo!)
  - supply chain attack
- what can you do to prevent this
  - selinux, apparmor, seccomp, tap…
- q&a :)

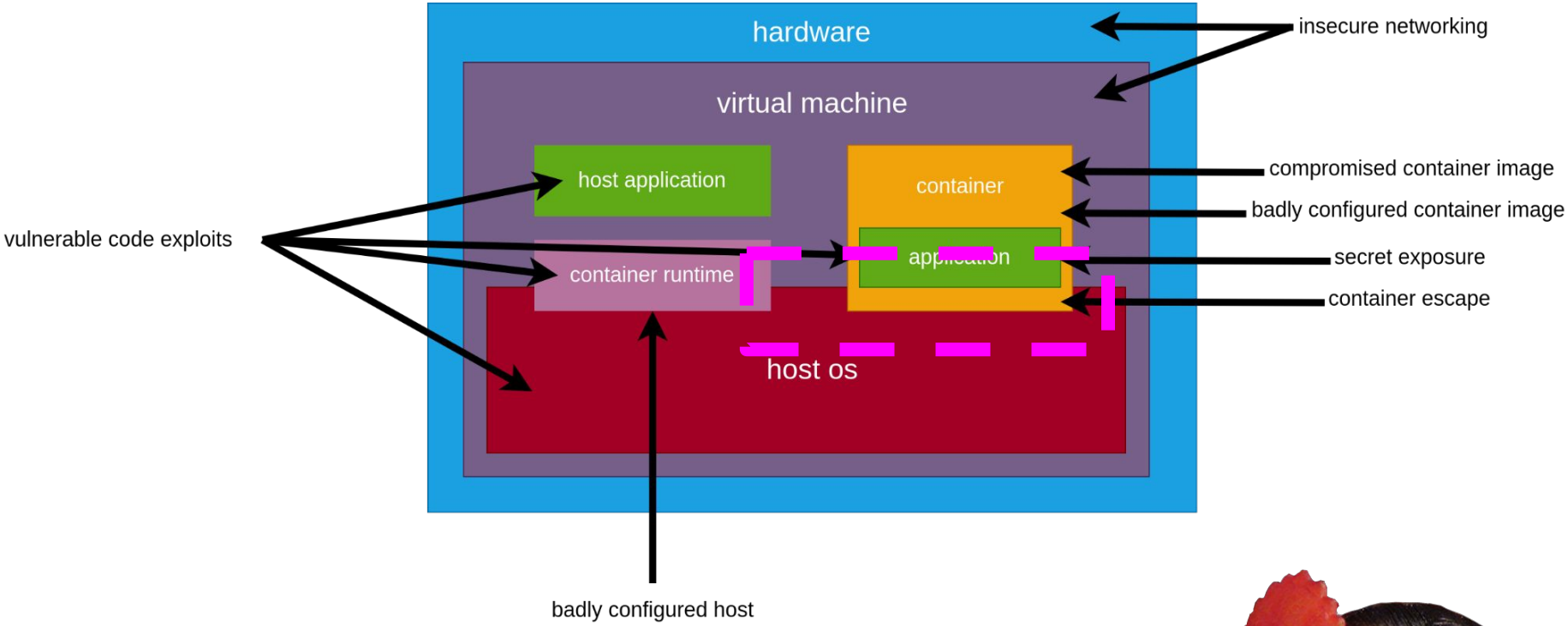
refresher on container basics

encapsulating applications and their dependencies into a single package that runs in isolated environments on a shared operating system kernel

- portable
- scalable
- fast
- **isolated?**

there is **no** container –
it's just another process
running on your machine


namespace
namespace
namespace
diplodocus enclosure
running env
psql
raptor enclosure
rails
t-rex enclosure
guards/control room
container runtime (e.g. runc, containerd)
linux kernel
capabilities
cgroups
namespaces
- UTS
- IPC
- PID
- net
- user
linux security modules (e.g. selinux, apparmor)
hardware
park


hardware
virtual machine
host application
container
container runtime
application
host os
insecure networking
compromised container image
badly configured container image
secret exposure
container escape
vulnerable code exploits
badly configured host

host separation

- namespaces
- cgroups
- container runtime

types of container escapes

- filesystem
- memory
- container runtime
- kernel

eBPF

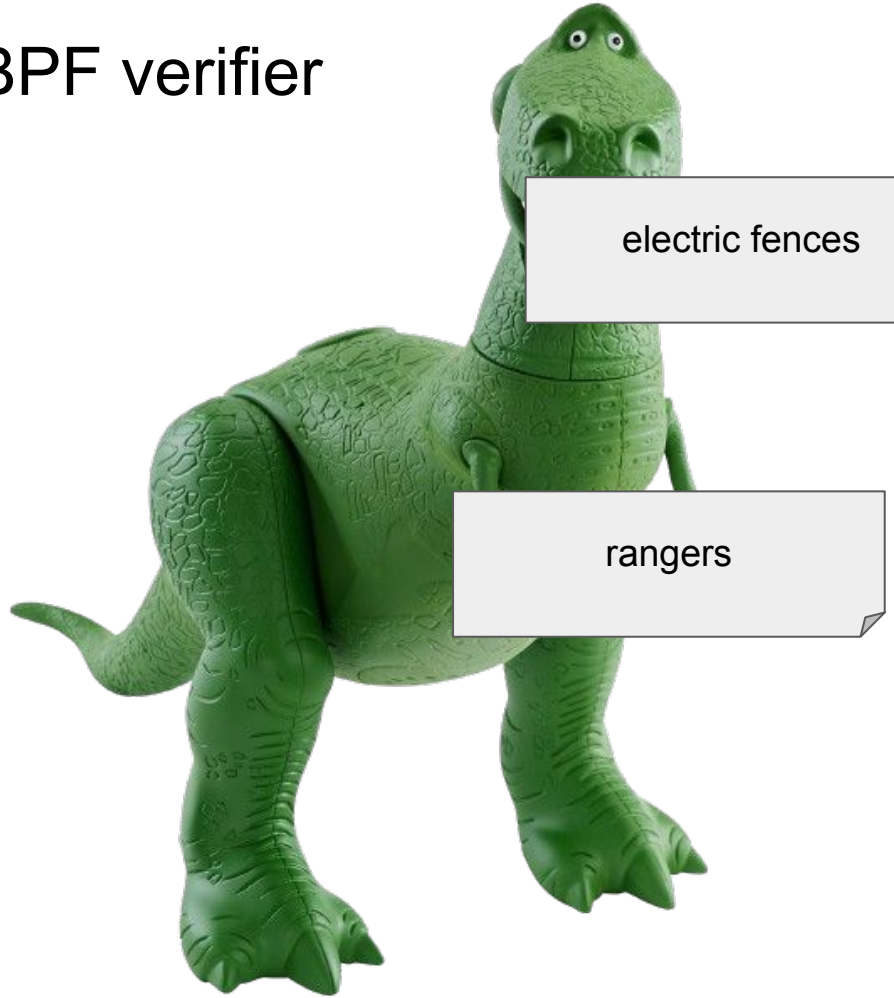
originally designed for packet filtering but later extended to allow running user-space code directly in the kernel

does to the linux kernel what js does to html (in a sense)

eBPF attack vectors

- eBPF verifier
  - trick kernel into loading unsafe bytecode
  - arithmetic/logic errors
- JIT compiler
  - errors in compilation of bytecode
- memory corruption to overwrite eBPF programs

eBPF verifier



directed acyclic graph (DAG) check to prevent loops and validate the control flow graph (CFG), ensuring there are no unreachable instructions

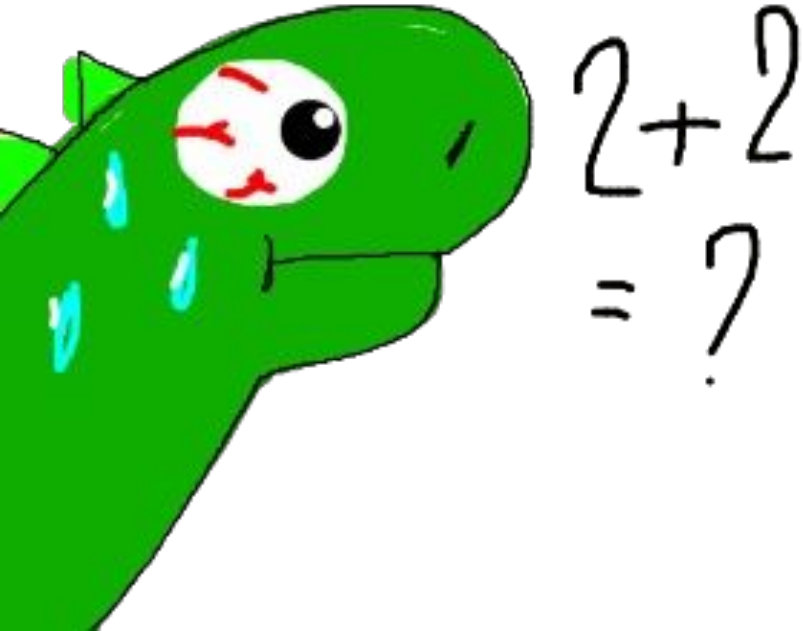
2. simulates execution of every instruction, examining all possible execution paths to observe how each affects the state of registers and the stack

in the wild…

vulnerabilities in the eBPF verifier, which can be exploited for local privilege escalation:

- CVE-2021-3490
- CVE-2023-2163
- CVE-2024-41003

maths!



the exploit relies on a flaw in the eBPF verifier's handling of certain arithmetic operations, allowing attackers to manipulate kernel memory and execute arbitrary code

blind spot in the fence: raptors escape if they hit the fence just right

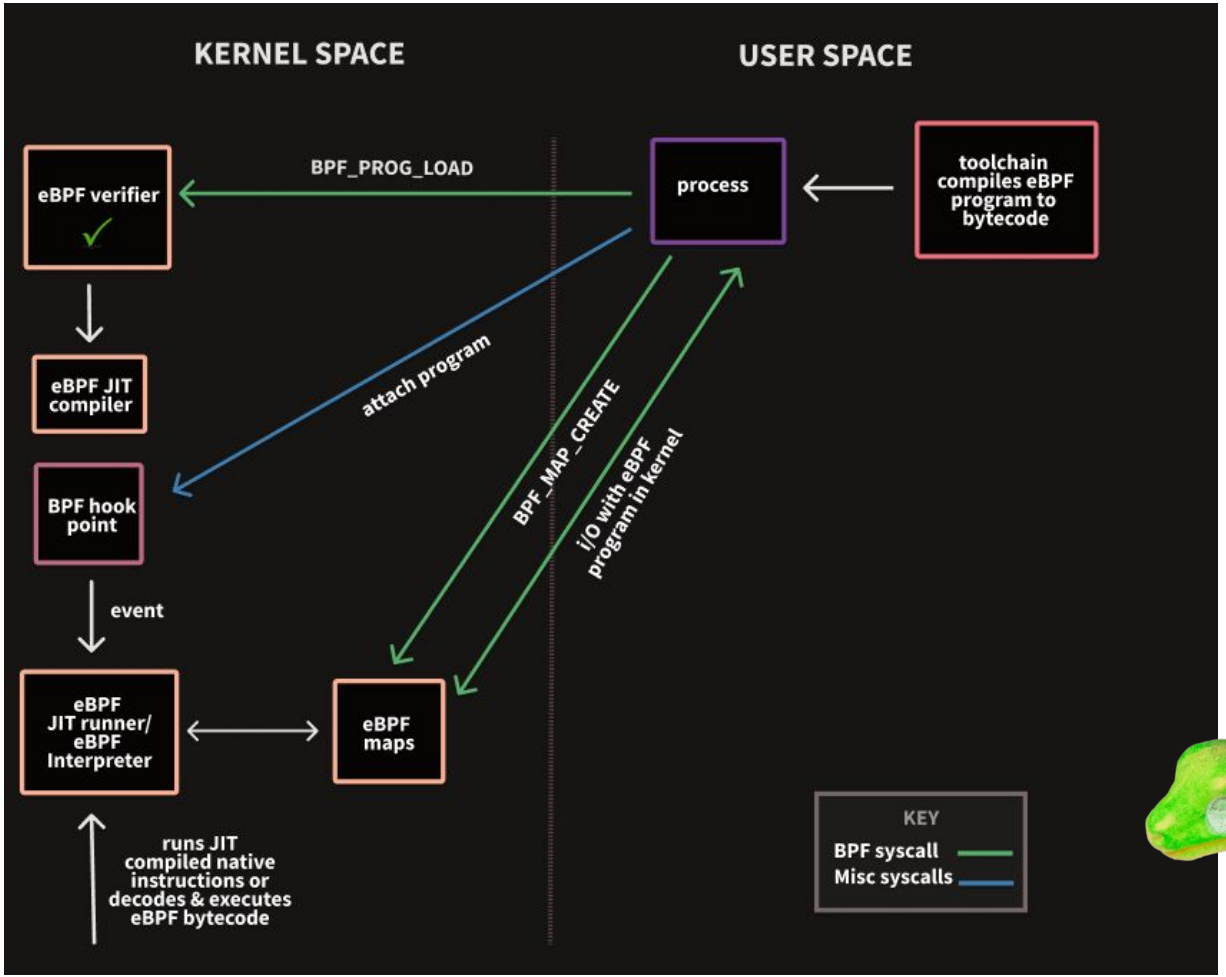


diagram:
@chompie1337

demo :D

what now?

- explore the network
- data exfiltration, steal credentials/files
- lateral movement, e.g. spawn more (privileged) containers
- establish persistence, install a backdoor
- ransomware
- phishing (mail server/dns)

testing the fences for weak spots

downloading sensitive info (like dino feeding schedules or electric fence codes)

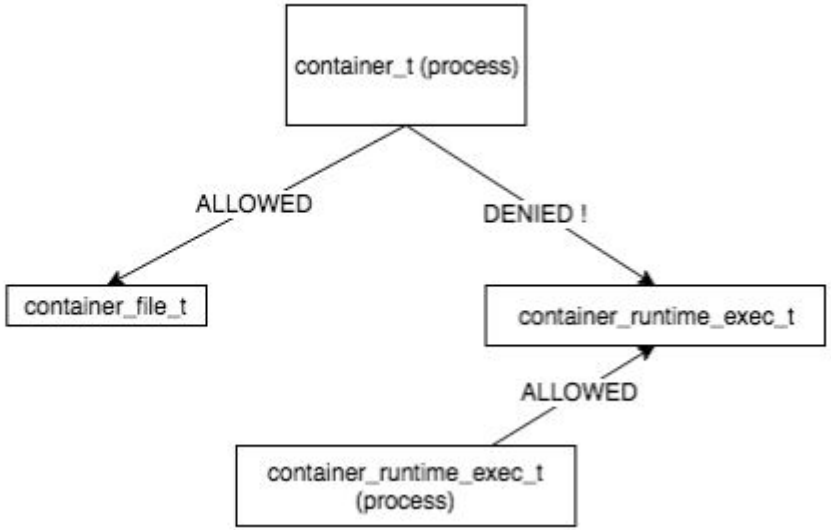
raptors once they're loose—moving from one paddock to another

hidden nest in the visitor center

container runtime

selinux

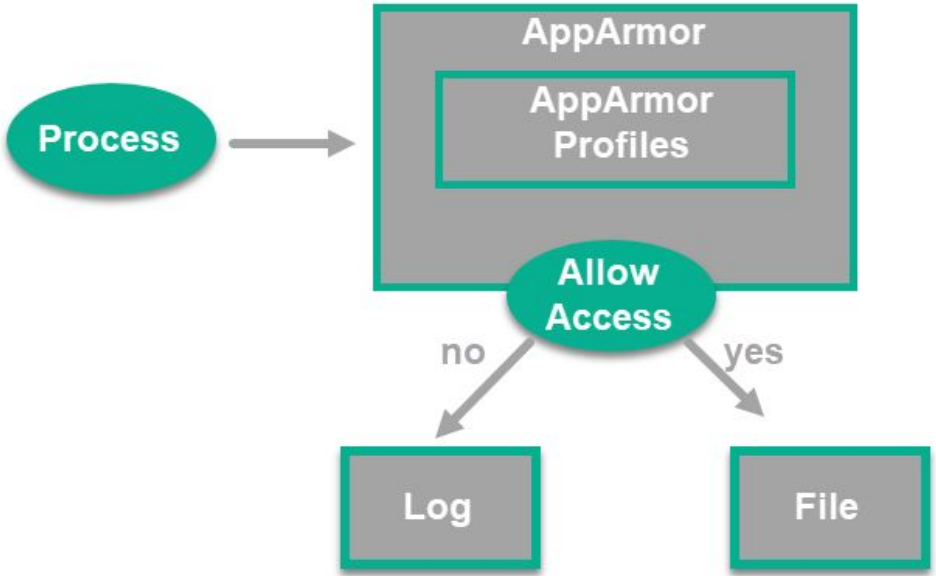
mandatory access control prevents runc from being overwritten by malicious containers, even if containers are run as root



apparmor

similar to selinux (mac model)

path based instead of policy based

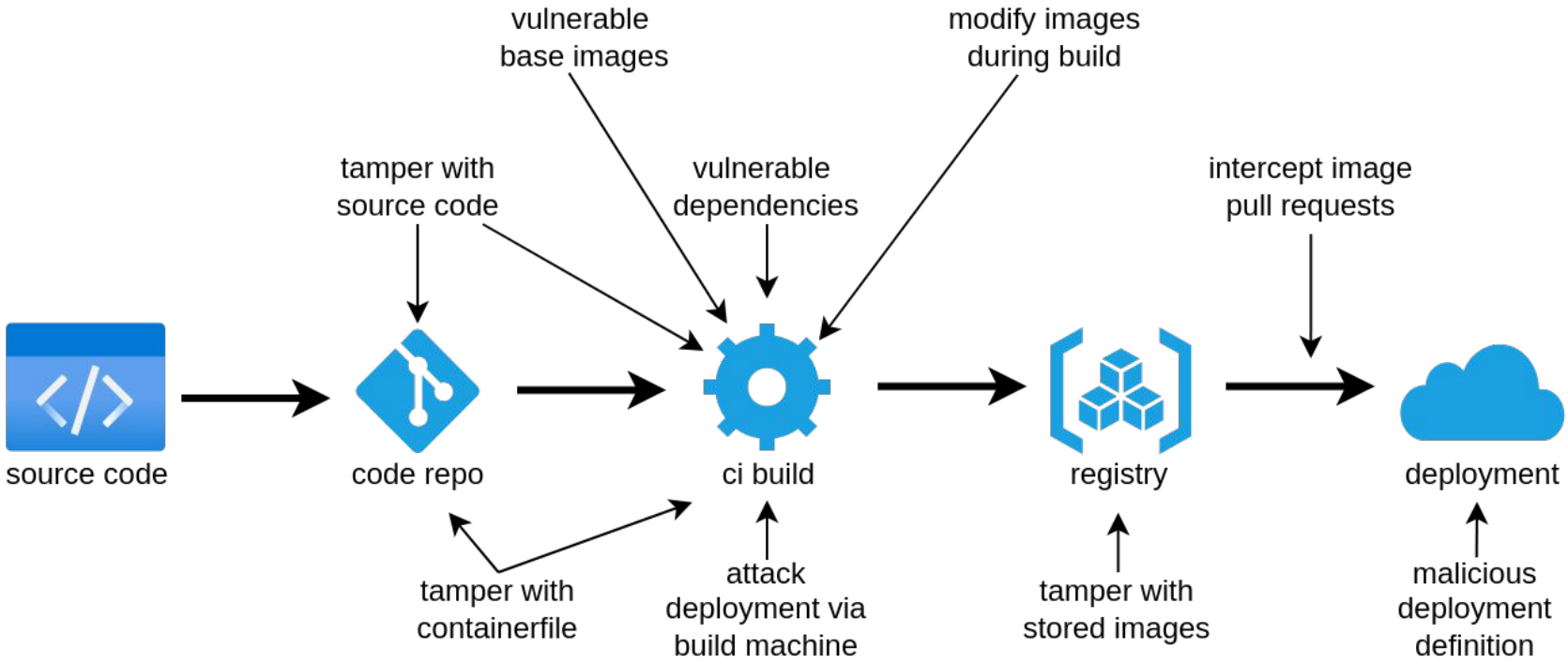


seccomp (secure computing mode)

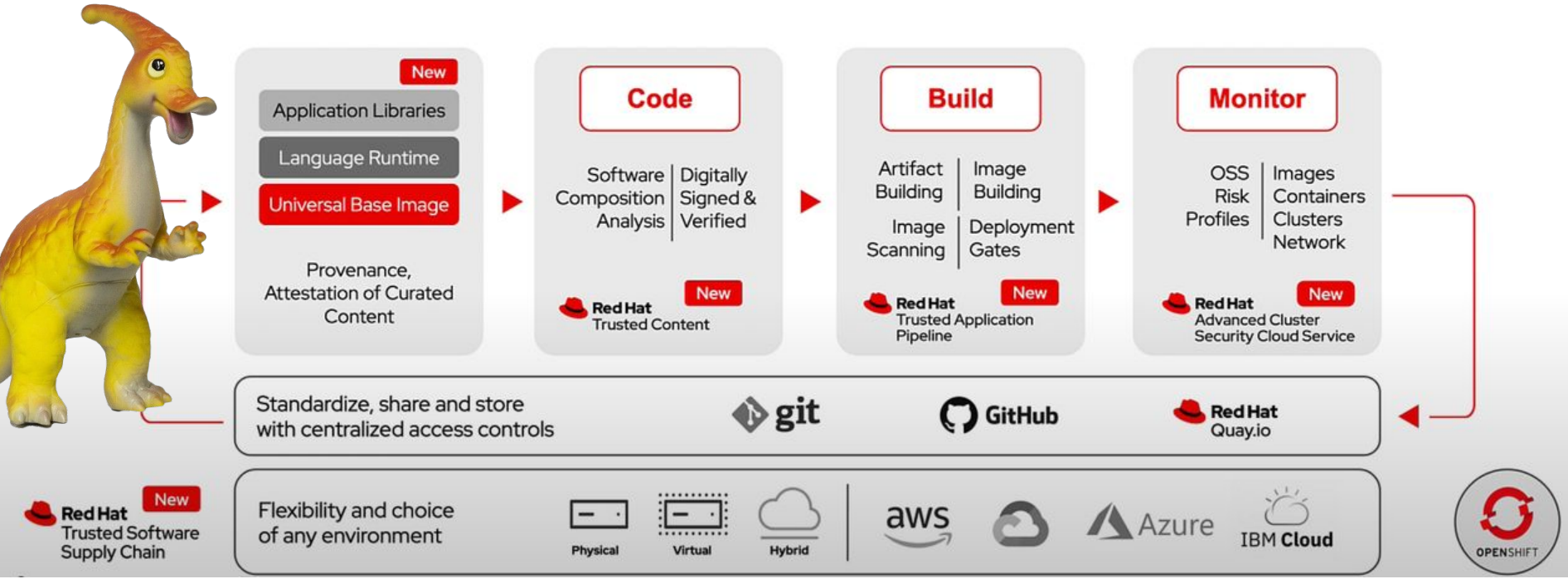
kernel feature that restricts the system calls a process can make

ensures that even if a malicious program gets inside a container, it can't easily make a jailbreak by exploiting system calls to interact directly with the host kernel or gain unauthorized access to resources.

supply chain attacks

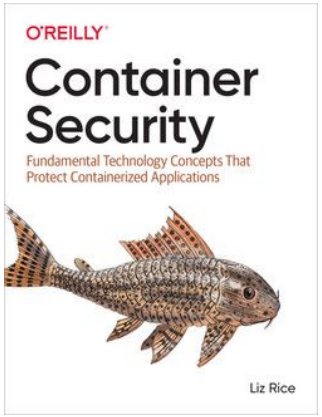


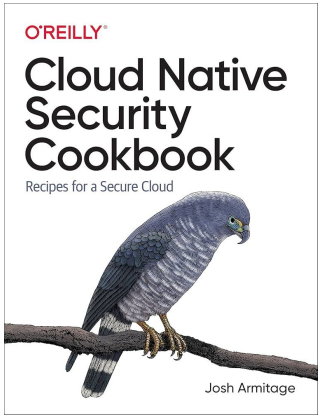
trusted software supply chain

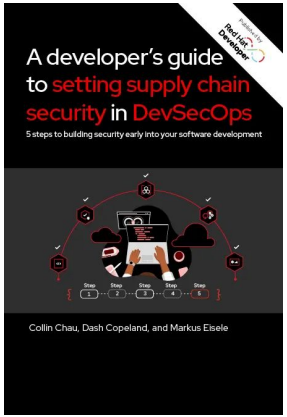


Further Reading :D

- aquasec.com/cloud-native-academy/container-security/container-security
- redhat.com/en/blog/hardening-docker-containers-images-and-host-security-toolkit
- chomp.ie/Blog+Posts/Kernel+Pwning+with+eBPF+-+a+Love+Story
- bughunters.google.com/blog/6303226026131456/a-deep-dive-into-cve-2023-2163-how-we-found-and-fixed-an-ebpf-linux-kernel-vulnerability
- blog.doyensec.com/2022/10/11/ebpf-bypass-security-monitoring.html







josie@redhat.com

q&a