MesosCon
EUROPE

WTF, my container just spawned a shell

Jorge Salamero Sanz, Sysdig

THE LINUX FOUNDATION

% whoami

Jorge Salamero Sanz

<jorge.salamero@sysdig.com>

@bencerillo

- Working on OSS last 12+ years
- Working on monitoring for 3+ years
- Containers gamer @ Sysdig

Sysdig: monitoring & troubleshooting

- 100% open-source
- 1M+ downloads
- Container troubleshooting
- sysdig.org

- SaaS & *on-prem*
- Kubernetes, Swarm, **DC/OS Mesos**, etc.
- Dashboards, alerts, events, teams
- sysdig.com

Sysdig: security & forensics

- 100% open-source
- DIY: Yahoo, Paypal, etc
- security monitoring
- sysdig.org/falco

- SaaS & *on-prem*
- Kubernetes, Swarm, **DC/OS Mesos**, etc.
- **run-time security + forensics**
- sysdig.com

Scanning

What are my containers doing?

- Static scanning
- Dynamic scanning

Why static scanning?

Yay, this was soo easy to deploy! I ❤ Docker Hub!
(your developers too, actually they were already using it :P)

uhm… wait, is someone maintaining this image?

```
RUN apt-get install -y wget build-essential python python-dev python-pip
python-virtualenv
RUN wget http://nodejs.org/dist/node-latest.tar.gz
RUN tar xvzf node-latest.tar.gz
RUN cd node-v* && ./configure && CXX="g++ -Wno-unused-local-typedefs" make && CXX="g++
-Wno-unused-local-typedefs" make install
```

How does it work?

- Scan contents of images looking for software versions with known defects

- Container image layering can make this efficient (exploits immutable nature of images)

Why run-time scanning?

OK, no known vulnerabilities, still secure?

Containers are black boxes exposing a behaviour, is something misbehaving?

- Dynamic Scanning:
  - Enforcement
  - Auditing

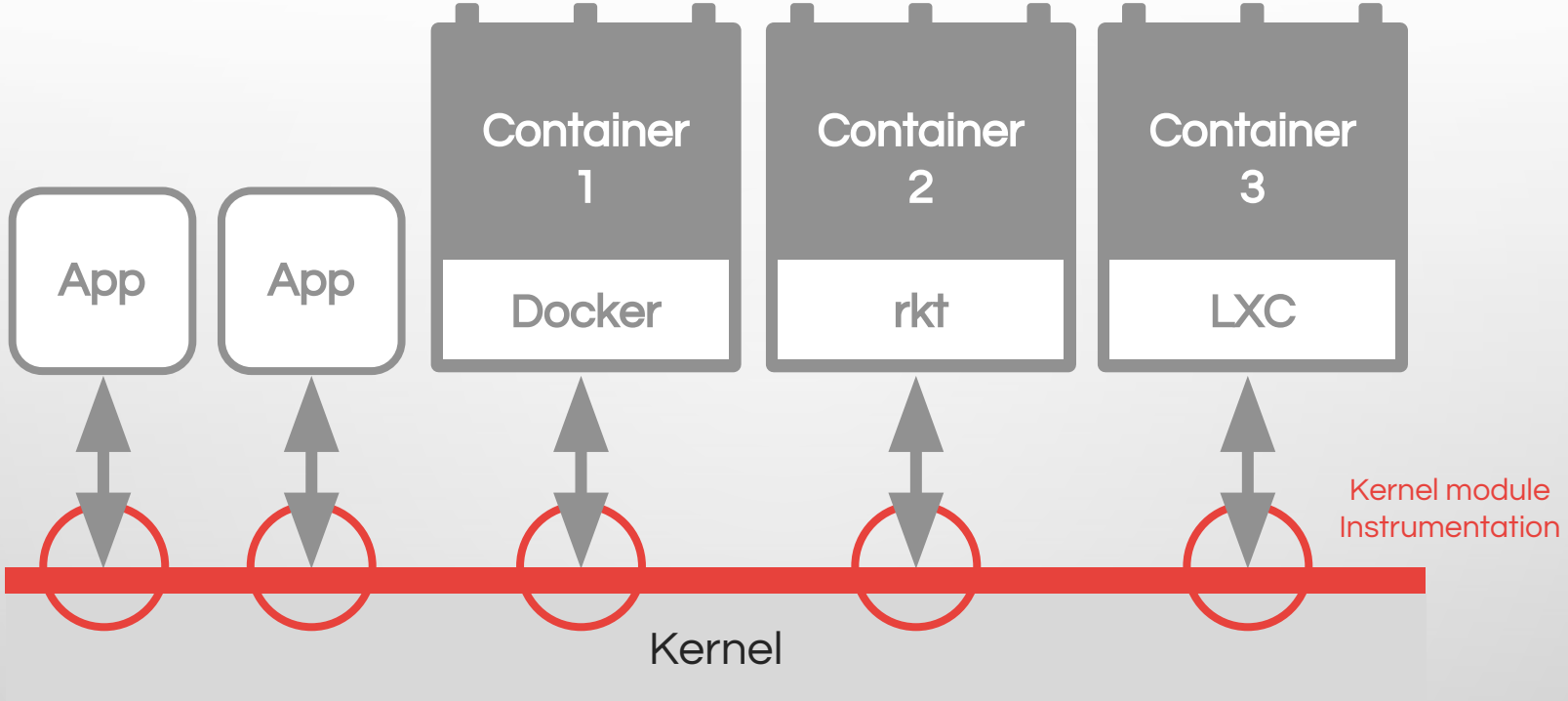
Tools

- Basic sandboxing: seccomp
- Sandboxing with policies: seccomp-bpf
- Mandatory access control systems: SELinux, AppArmor
- System auditing: Auditd
- Behavioral monitoring: Falco
- Run-time protection and forensics: Sysdig Secure

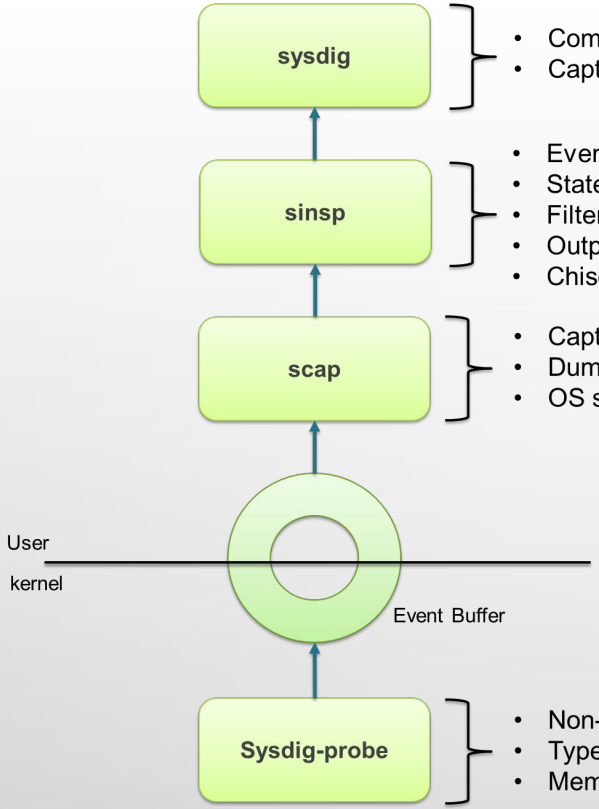
How sysdig works?

- Sysdig uses tracepoints (>= 2.6.28 - December 2008)
- Attach probes to specific functions inside the kernel perf list 'syscalls:*'
- Sysdig sees everything, unlike strace instead of attaching to a process, we filter out stuff

Sysdig


App
App
Container 1
Docker
Container 2
rkt
Container 3
LXC
Kernel module
Instrumentation
Kernel

Sysdig



- Command line parsing
- Capture management

- Event parsing
- State engine
- Filtering
- Output Formatting
- Chisel execution

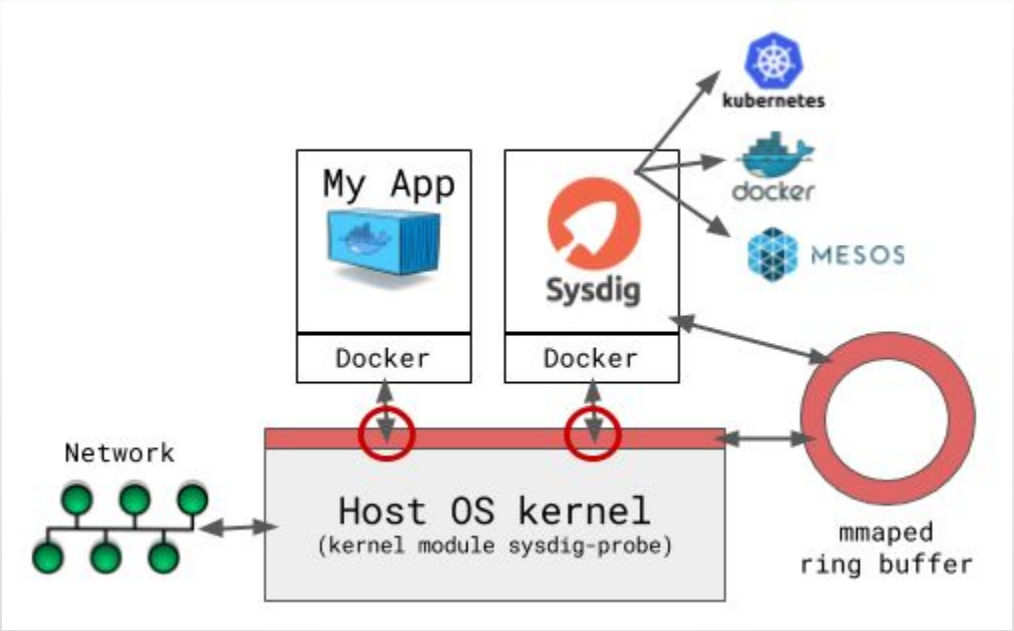
- Capture Control
- Dump files R/W
- OS state collection

- Non-blocking event collection
- Type-based event packing
- Memory mapped buffer handling

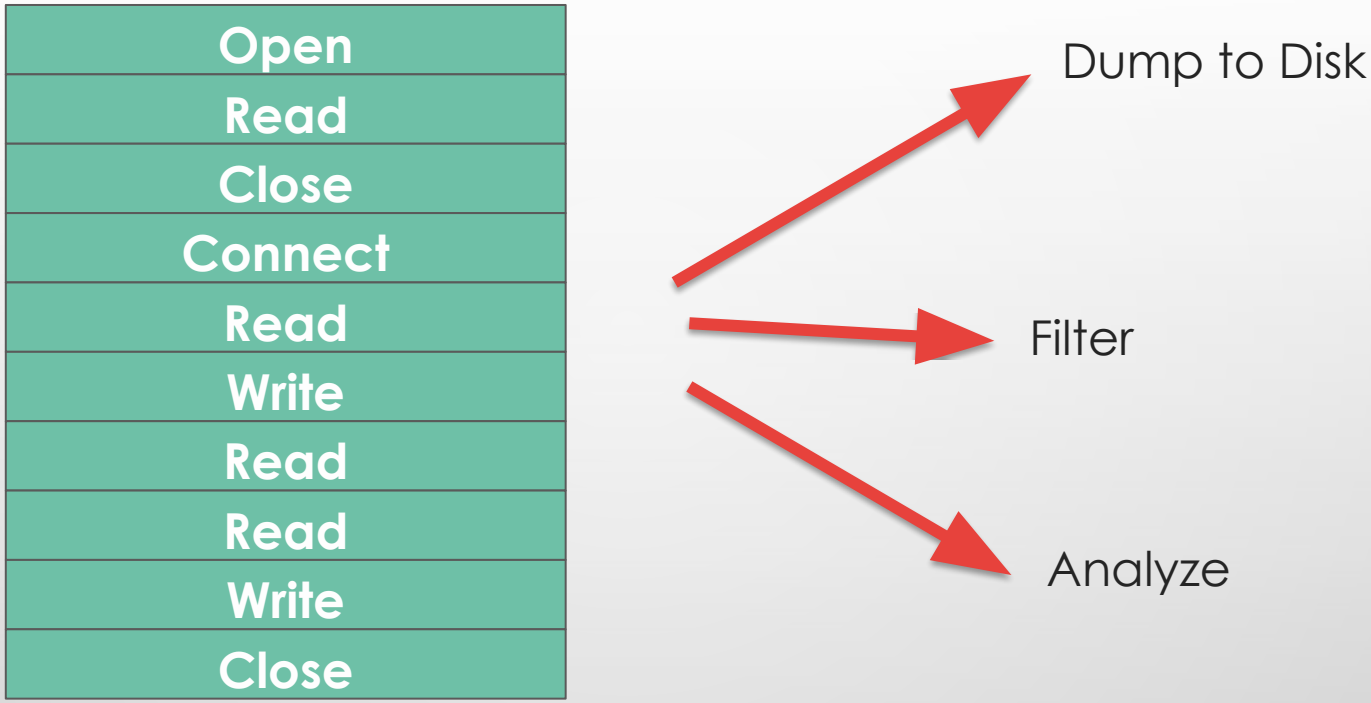
enter/exit glibc

probe: syscalls

Sysdig


kubernetes
docker
MESOS
My App
Sysdig
Docker
Docker
Network
Host OS kernel
(kernel module sysdig-probe)
mmaped
ring buffer

Event stream



How to use syscalls?

- clone() and execve() give you insight into process creation and command execution.
- open(), close(), and the FD read and write functions offer visibility on disk I/O.
- socket(), connect(), and accept() give insight into network activity.

Not just syscalls!

Syscalls are turned into events with context:

- the process name performing the system call
- the process's parents, grandparents, etc
- the remote IP address to which the process is communicating
- the directory of the file being read/written
- the current memory usage of the process

Sysdig Falco

- Detects suspicious activity defined by a set of easy rules
- Uses Sysdig's flexible and powerful filtering expressions (in userspace -single point of failure-)
- Container support (Docker, Kubernetes, Mesos, etc)
- Flexible notification methods
- Open Source

Falco rules

A shell is run in a container	`container.id != host and proc.name = bash`
Overwrite system binaries	`fd.directory in (/bin, /sbin, /usr/bin, /usr/sbin) and write`
Container namespace change	`evt.type = setns and not proc.name in (docker, sysdig)`
Non-device files written in /dev	`(evt.type = creat or evt.arg.flags contains O_CREAT) and proc.name != blkid and fd.directory = /dev and fd.name != /dev/null`
Process tries to access camera	`evt.type = open and fd.name = /dev/video0 and not proc.name in (skype, webex)`

Falco outputs

- Events that match filter expression (rule) result in alerts

output field used to format event into alert message

  - syslog
  - file
  - stdout
  - shell (e.g. mail -s "Falco Notification" alerts@example.com)

Demo time!

- Sysdig Falco
  Falco Demo
- Sysdig Secure
  on DC/OS MEsos

Thank You!

@bencerillo
@sysdig
sysdig.com | sysdig.org


MesosCon
EUROPE
THE LINUX FOUNDATION