Intro to Falco

Intrusion Detection for Containers

Shane Lawrence
Shopify

Intro to "Intro to Intro to Falco"

shanelawrence.info

Present

- Falco end user
- Infrastructure Security @Shopify

Past

- Intrusion Detection Systems (IDS)
- Security Information and Event Management (SIEM)

I. Intro

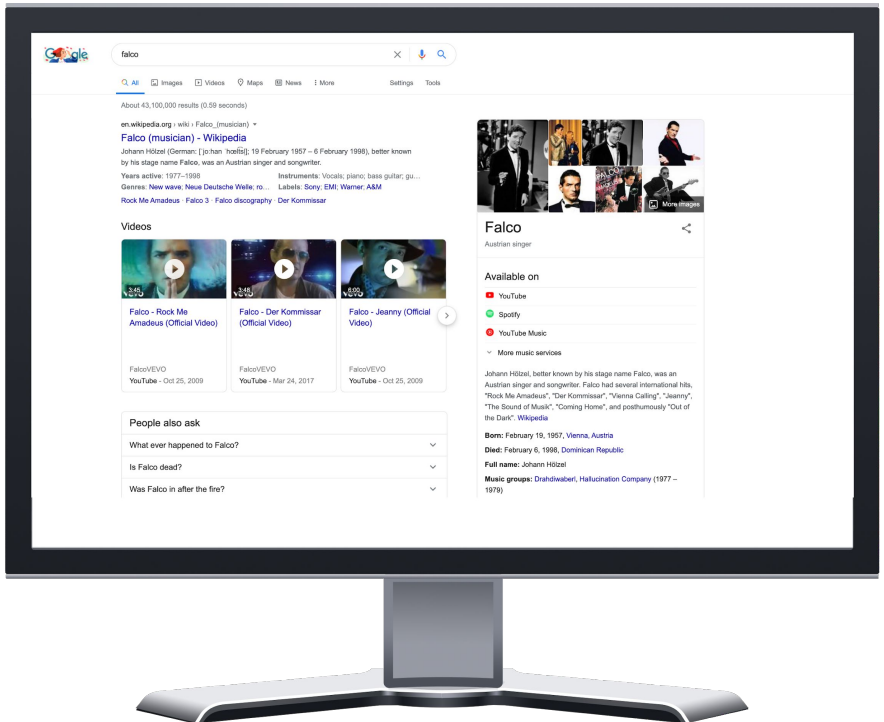
Intro to Intro to Falco

- Intro
- Deployment
- Technical Challenges
- Use Cases
- Next Steps

see also: "Open Source Intrusion Detection for Containers"

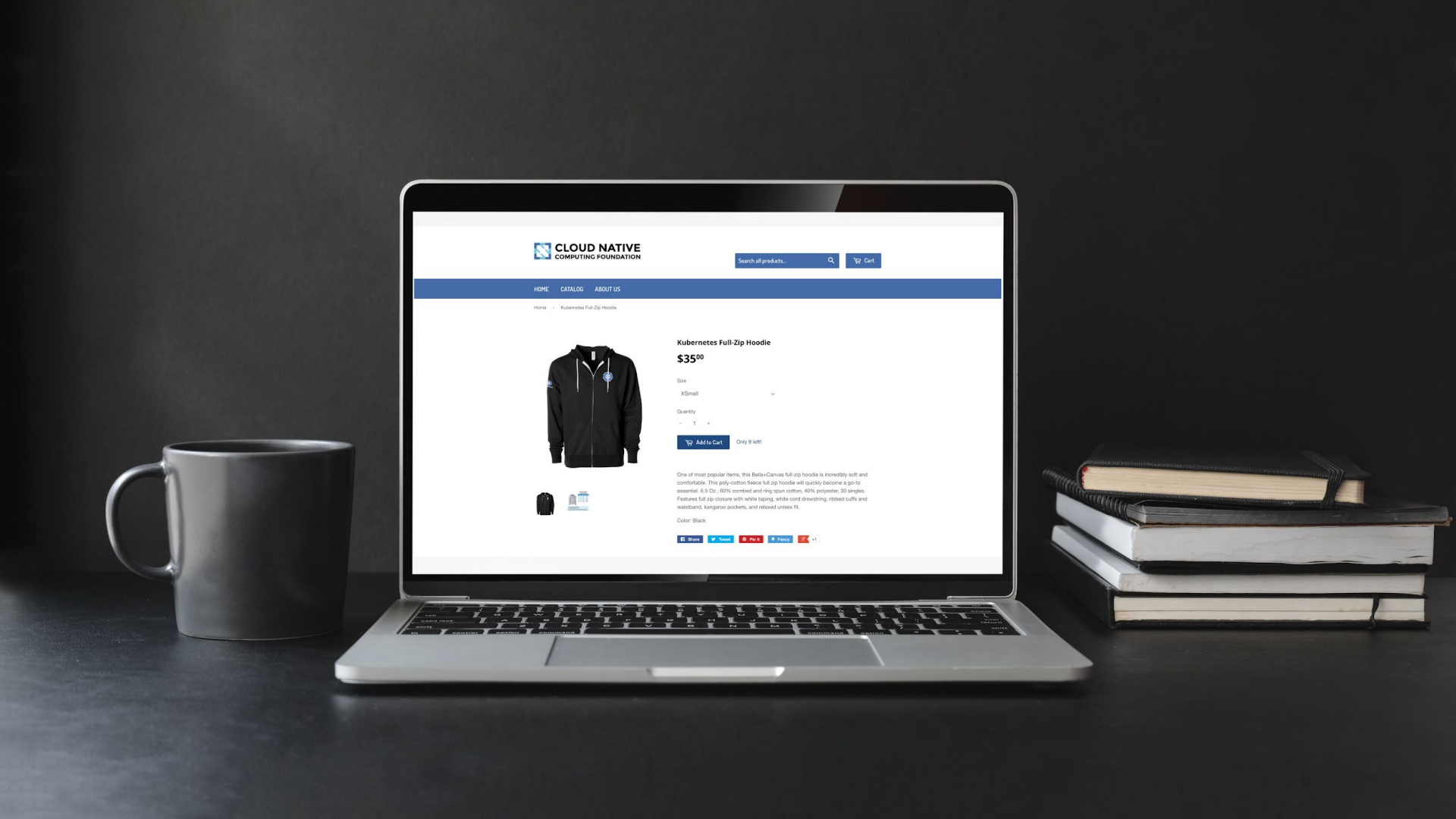
Not Intro to Falco:

- new wave



Not Intro to Falco:

- new wave
- all of the other things you should be doing to protect your clusters


CLOUD NATIVE
COMPUTING FOUNDATION
Search all products...
Cart
HOME
CATALOG
ABOUT US
Home › Kubernetes Full-Zip Hoodie
Kubernetes Full-Zip Hoodie
$35.00
Size
XSmall
Quantity
1
Add to Cart
Only 8 left!
One of most popular items, this Bella+Canvas full-zip hoodie is incredibly soft and comfortable. This poly-cotton fleece full zip hoodie will quickly become a go-to essential. 6.5 Oz., 60% combed and ring spun cotton, 40% polyester, 30 singles. Features full zip closure with white taping, white cord drawstring, ribbed cuffs and waistband, kangaroo pockets, and relaxed unisex fit.
Color: Black
Share
Tweet
Pin it
Fancy
+1

shopify

$61
billion
volume

300
million
shoppers

$1.5
million
sales per minute
(peak)

shopify

50+

clusters

10+

thousand

services

170+

thousand

requests per second (peak)

The case for monitoring

Misconfiguration

- insecure API endpoints
- overprivileged roles
- weak security context
- assumptions about safety/identity

Software supply chain

- deliberate backdoors
- bugs in dependencies
- typo, tag hijacking

Unmitigated Vulnerabilities

- **Heartbleed**
  CVE-2014-0160
- **Spectre v1**
  CVE-2017-5753
- **Spectre v2**
  CVE-2017-5715
- **Meltdown**
  CVE-2017-5754
- **CVE-2019-5736**
  runc Container Escape
- **CVE-2018-15664**
  Symlink directory traversal
- **CVE-2018-1002105**
  Unauthenticated Remote Privilege Escalation
- **CVE-2020-8558**
  Localhost Boundary Bypass

The case for Falco

- low-level
- container-aware
- Kubernetes-aware
- ebpf
- open source
- CNCF project

II. Deployment

Deploying Falco

The easiest way:

1. Get Falco's chart configuration.
2. Configure Falco to use the correct driver.
3. Add the repo.
4. Install falco.

Deploying Falco

A better way:

- validate rules before deploying
- pre-build the driver
- build your own image
- fine-tune the rules

III. Technical Challenges

Modifying rules

Challenge:

- update rules quickly without disruption

Potential solution:

- use Helm (and rules2helm)

Concerns:

- slow
- missed or duplicate events

Modifying rules

Challenge:

- update rules quickly without disruption

Scalable solution:

- ConfigMap -> volume -> /etc/falco
- inotifywait
- kill -HUP

Normalization

Challenge:

- Falco alerts on normal behavior.

Solution:

- Add events to an allowlist.
- Be as specific as possible.

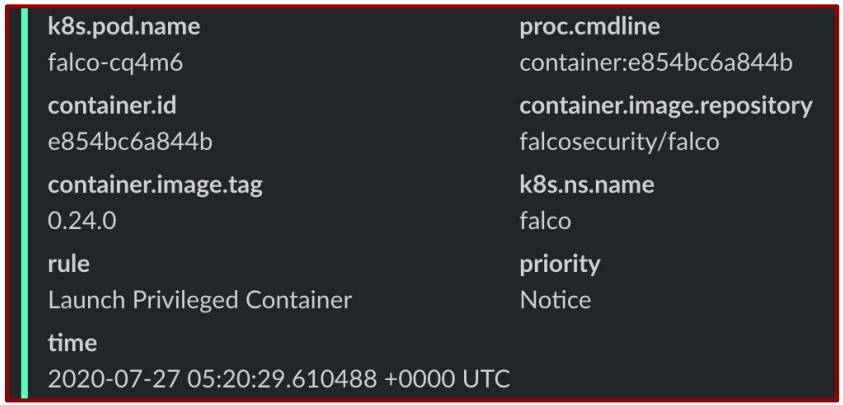
Normalization

Challenge:

- Falco alerts on normal behavior.

Example problem:

- Falco flags itself as suspicious.



Normalization

Challenge:

- Falco alerts on normal behavior.

Example solution:

- Add the missing repository to the allowlist.

```
# falco_rules.local.yaml
- list: user_privileged_images
  items: [
    gke.gcr.io/netd-amd64, gke.gcr.io/gke-metadata-server,
    gke.gcr.io/kube-proxy, falcosecurity/falco
  ]

- macro: user_privileged_containers
  condition: (container.image.repository endswith sysdig/agent or
              container.image.repository in (user_privileged_images))
```

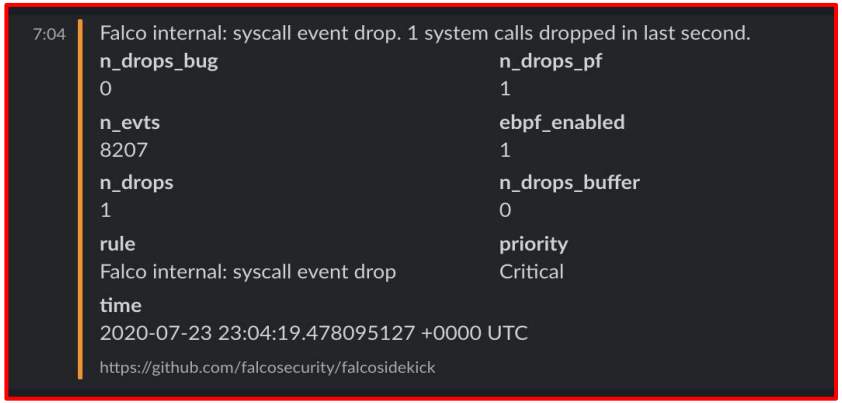
Normalization

Challenge:

- Falco generates critical alerts when it drops syscall events.

  (a small number of drops is expected)

Example Problem:



Normalization

Challenge:

- Falco generates critical alerts when it drops syscall events.

  (a small number of drops is expected)

Example Solution:

- Change the alert action to log only.
- Report the events less frequently.

```
# falco.yaml
syscall_event_drops:
  actions:
    - log
  # 1/60/60=once per hour
  rate: 0.000277777777778
max_burst: 1
```

IV. Use Cases

Suspicious shell access in container

Demo

Demo: Attacker running commands in container

keti <falco_pod> /bin/bash

apt-get install <package_name>

23:01:32.550604838: Error Package management process launched in container (user=root command=apt container_id=e37bcea88845 container_name=k8s_falco_falco-4w7d9_falco_0aae63a0-9ef9-41d9-8bfe-c7ee38298ec9_0 image=falcosecurity/falco:0.24.0) k8s.ns=falco k8s.pod=falco-4w7d9 container=e37bcea88845 k8s.ns=falco k8s.pod=falco-4w7d9 container=e37bcea88845

container.image.repository
falcosecurity/falco

container.image.tag
0.24.0

container.name
k8s_falco_falco-4w7d9_falco_0aae63a0-9ef9-41d9-8bfe-c7ee38298ec9_0

k8s.ns.name
falco

k8s.pod.name
falco-4w7d9

proc.cmdline
apt

container.id
e37bcea88845

user.name
root

rule
Launch Package Management Process in Container

priority
Error

time
2020-07-23 23:01:32.550604838 +0000 UTC

https://github.com/falcosecurity/falcosidekick

Falco APP 7:00 PM

23:00:36.625246794: Notice A shell was spawned in a container with an attached terminal (user=root k8s.ns=falco k8s.pod=falco-4w7d9 container=e37bcea88845 shell=bash parent=runc cmdline=bash terminal=34816 container_id=e37bcea88845 image=falcosecurity/falco) k8s.ns=falco k8s.pod=falco-4w7d9 container=e37bcea88845

container.image.repository
falcosecurity/falco

proc.name
bash

container.id
e37bcea88845

k8s.ns.name
falco

k8s.pod.name
falco-4w7d9

proc.cmdline
bash

proc.pname
runc

user.name
root

rule
Terminal shell in container

priority
Notice

time
2020-07-23 23:00:36.625246794 +0000 UTC

https://github.com/falcosecurity/falcosidekick

Instance metadata service

Demo

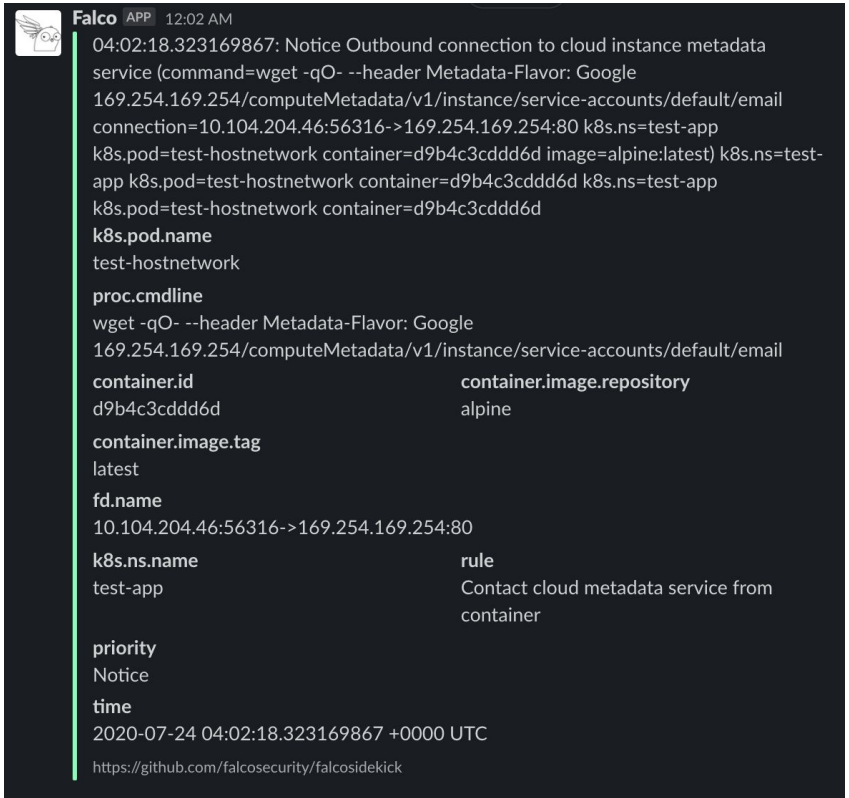
Instance metadata service

```
/ $  wget -qO- --header 'Metadata-Flavor: Google' 169.254.169.254/computeMetadata/v1/instance/attributes/
cluster-location
cluster-name
cluster-uid
# can only get things you should be allowed to access

/ $  wget -qO- --header 'Metadata-Flavor: Google' 169.254.169.254/computeMetadata/v1/instance/service-accounts/default/
aliases
email
identity
scopes
token

/ $ wget -qO- --header 'Metadata-Flavor: Google' 169.254.169.254/computeMetadata/v1/instance/service-accounts/default/email
falco-demo-test-app@shopify-codelab-and-demos.iam.gserviceaccount.com/
# can only get a token for this app's service account:
```

Instance metadata service (hostNetwork)

Demo

Use case: instance metadata service (hostNetwork)



Use case: instance metadata service (privileged)

- provider needs to redirect to proxy
- generally this is done with the network fabric

  (e.g. iptables rule points metadata.google.internal to the proxy)
- privileged container can just change the rules

Use case: CVE-2020-8557

Demo

V. Next Steps

Managing alerts

Output

- visible
- searchable
- aggregated
- annotated

Normalization

- prompt
- simple

Volume

- manageable

Reaching out

- #falco on Kubernetes Slack
- weekly community call
- GitHub Issue
- Pull Request 🏆

Thank you!

shanelawrence.info

shopify.com/careers

engineering.shopify.com

github.com/falcosecurity

falco.org

slack.k8s.io #falco