Falling out of the Sky

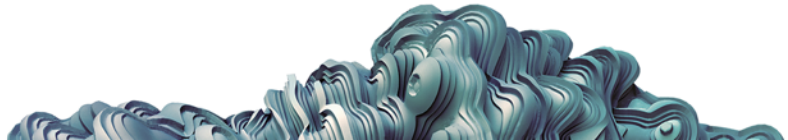
Security Risks that can Bring Your Cloud Down to the Ground

David Fiser

Security Fest 2019, 22th – 24th May

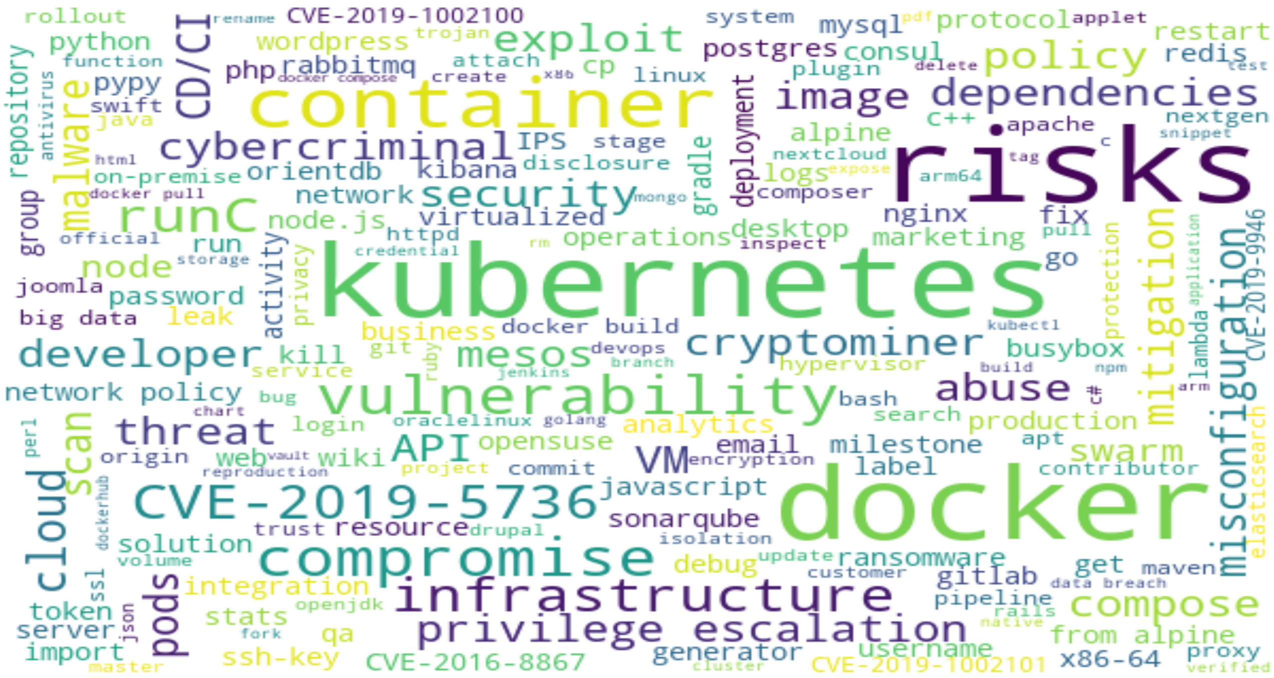
whoami

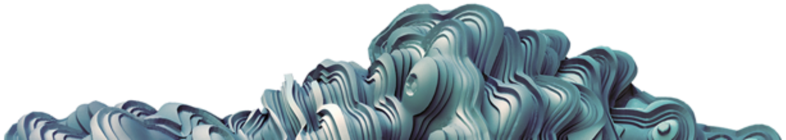
- security researcher
  - vulnerability research since 2017
- malware analyst
  - 2010 – 2017 (avast)
- contact
  - @anu4is, david_fiser (at) trendmicro.com



whoami

Agenda

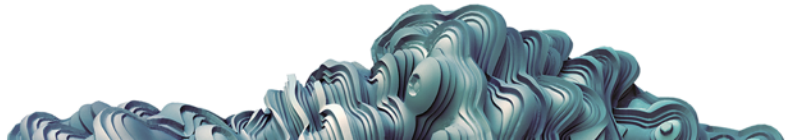




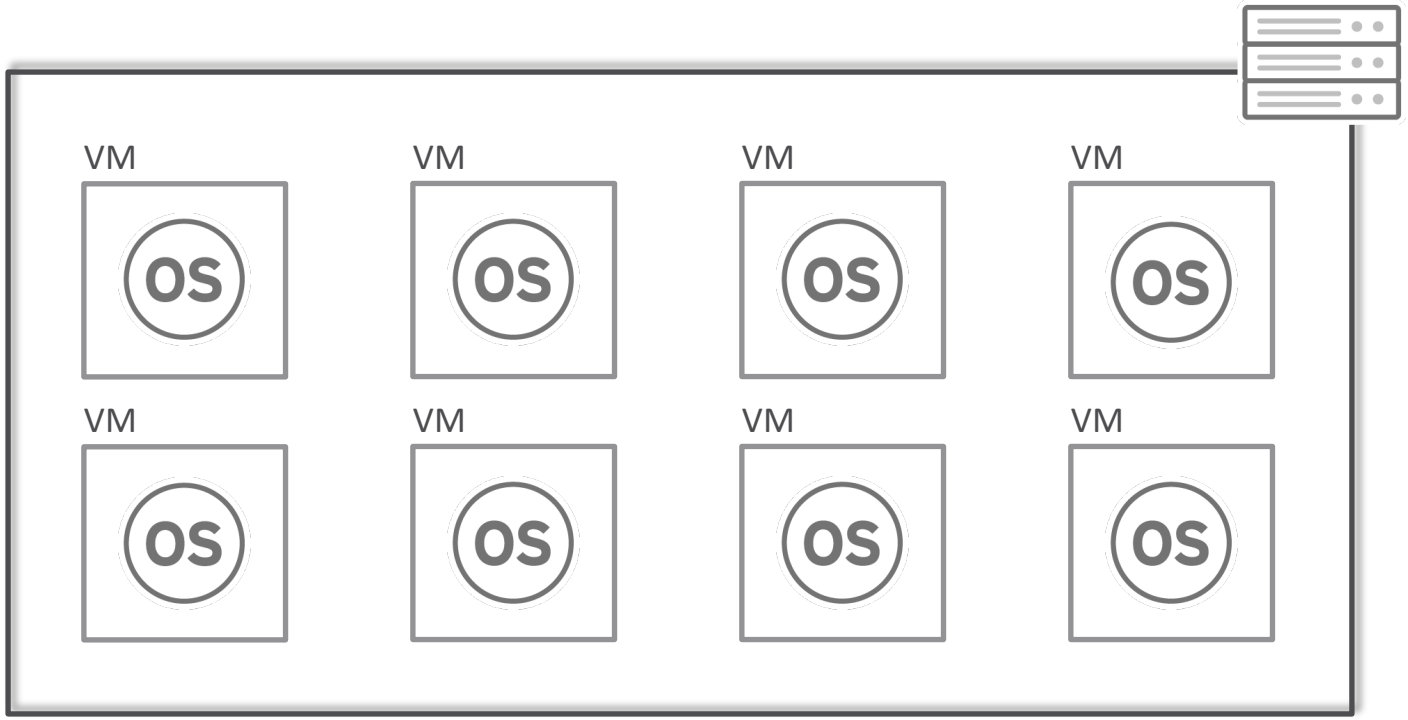
TREND MICRO

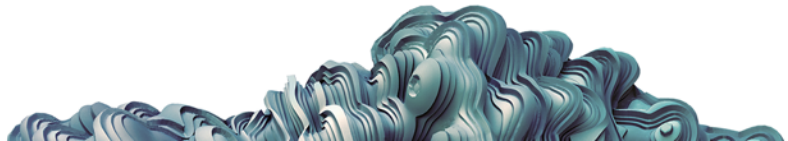
History

Physical HW → Virtual Machines → Containers



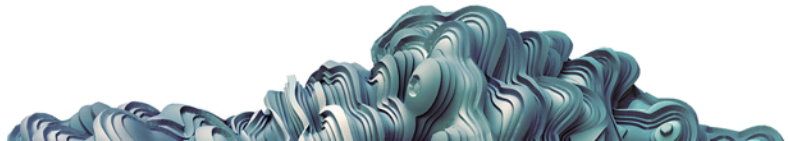
Virtual Machines





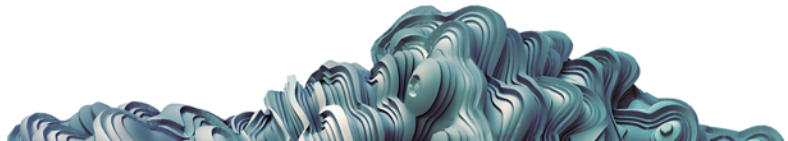
Virtual Machines

- abstraction of HW
  - CPU, memory, storage, etc.
- install OS, configure, manage
- …
- deploy application

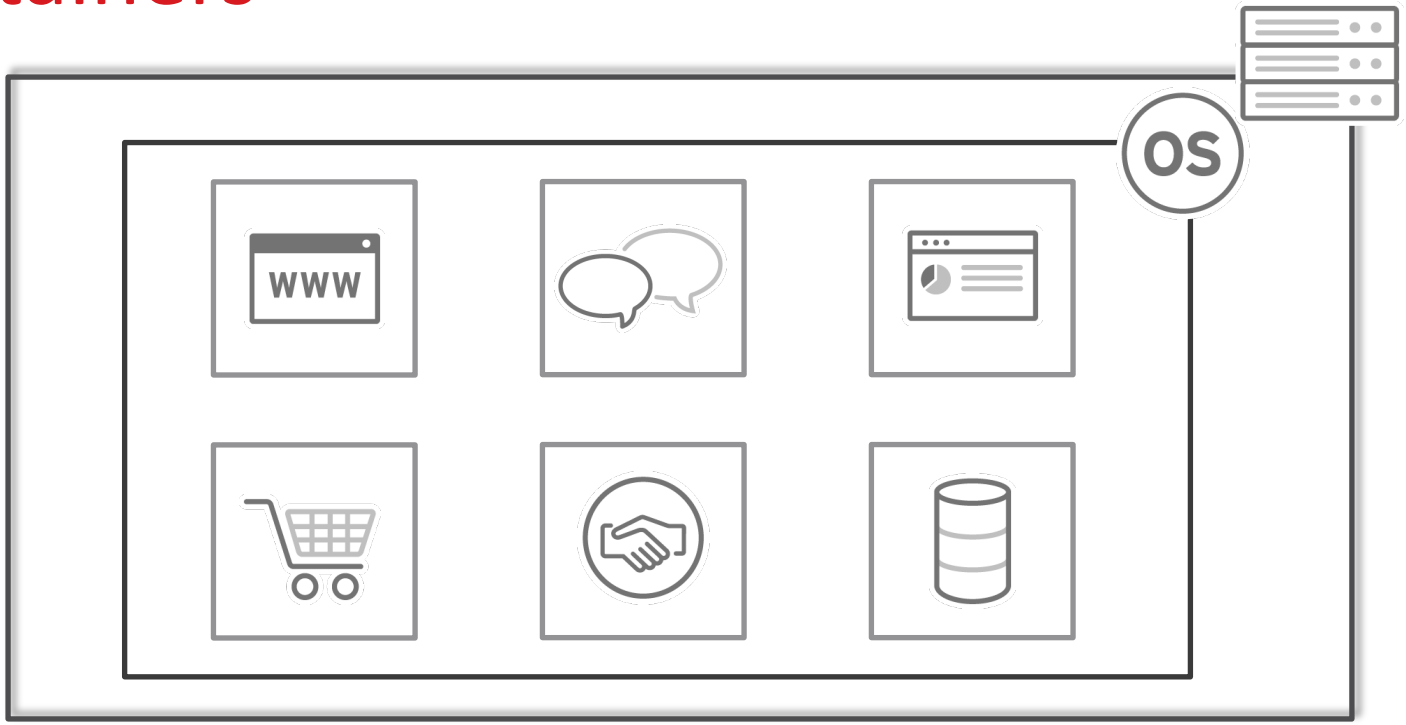


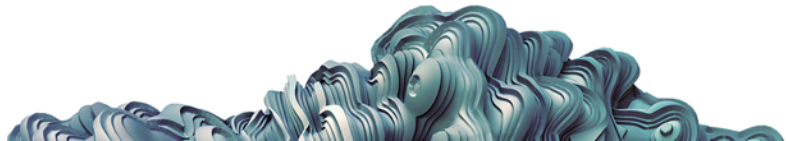
Containers

- no HW emulation
- running on **host** OS
- shared **kernel** with the **host**
- **container engine**

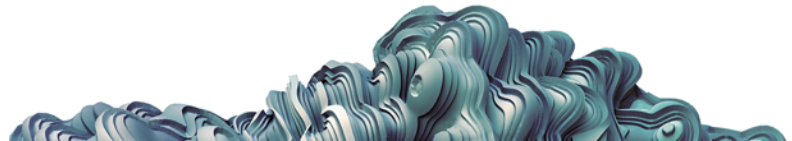


Containers





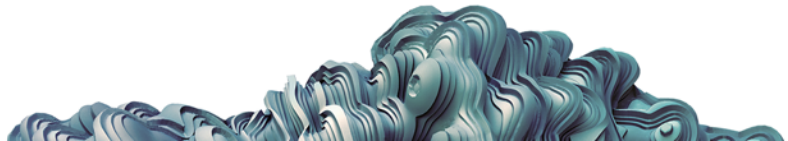
Containers



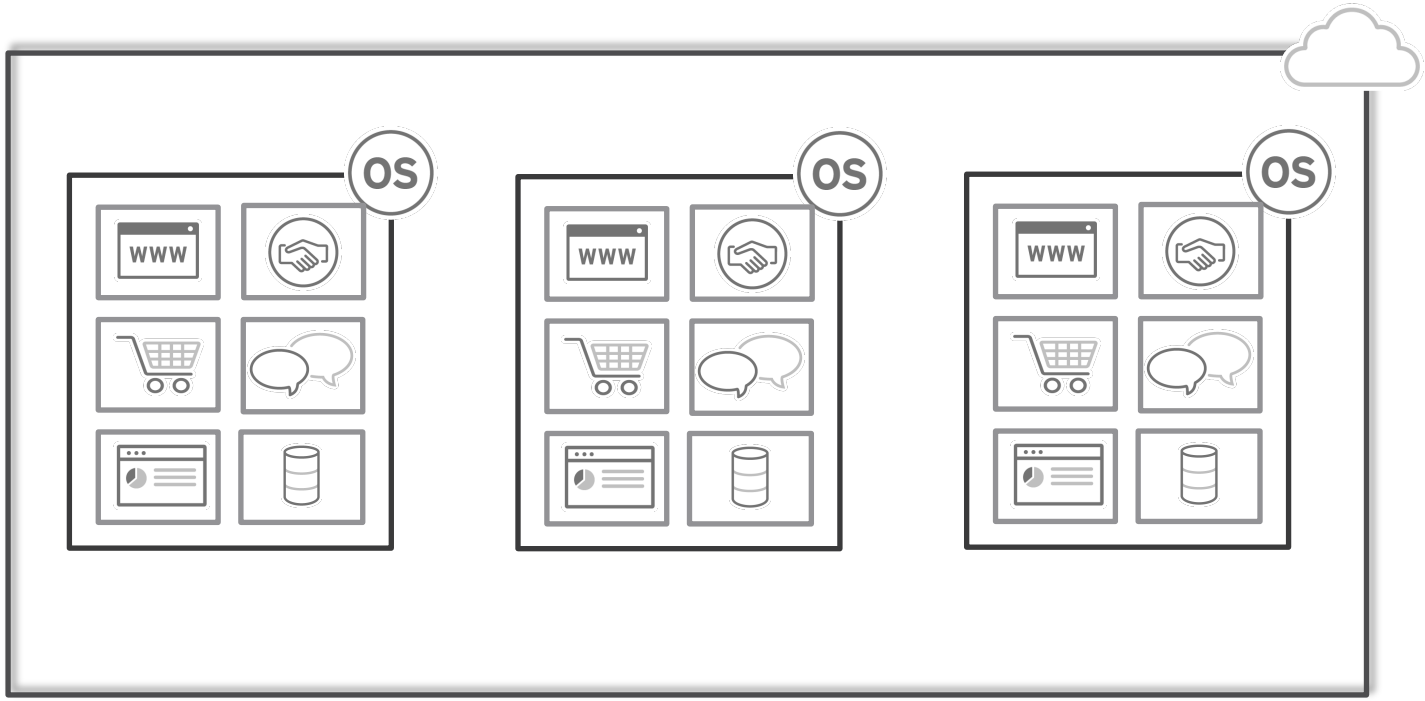
VM x Container attacker perspective

- compromise VM
- compromise network

- compromise container
- compromise host
- compromise cluster



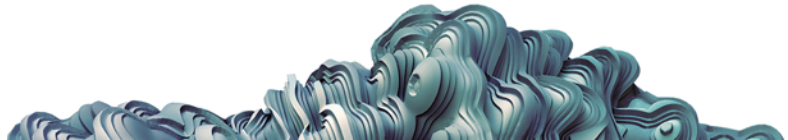
Containers

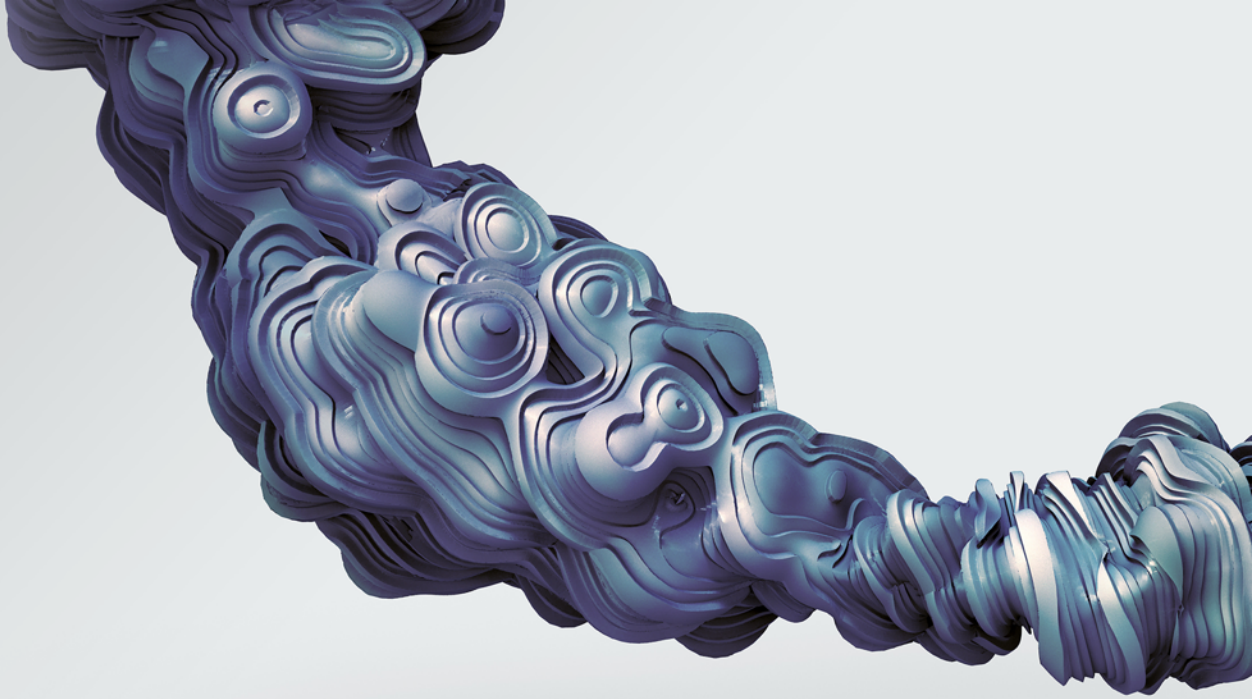


Docker & Kubernetes

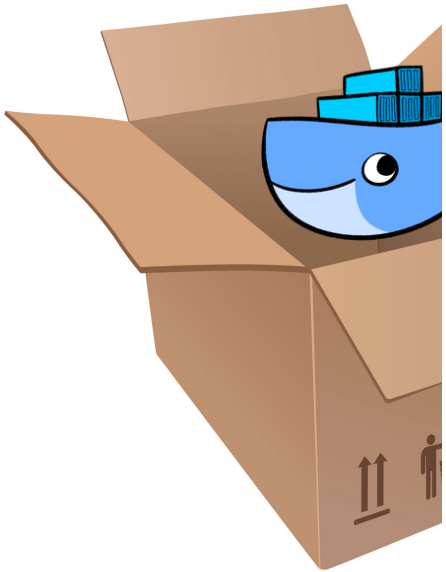
- Docker
  - container engine

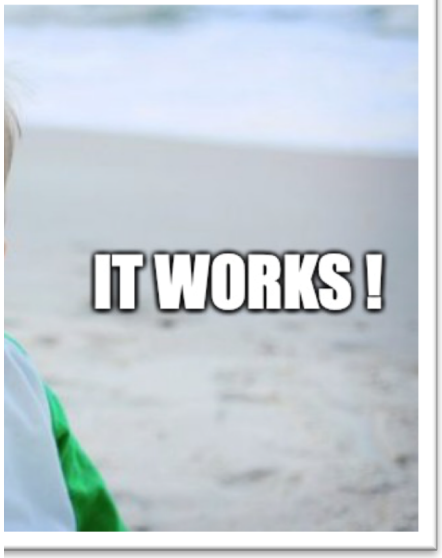
- Kubernetes
  - orchestrator (deployment manager)

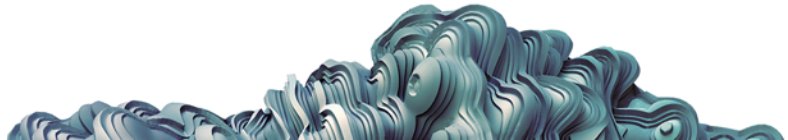


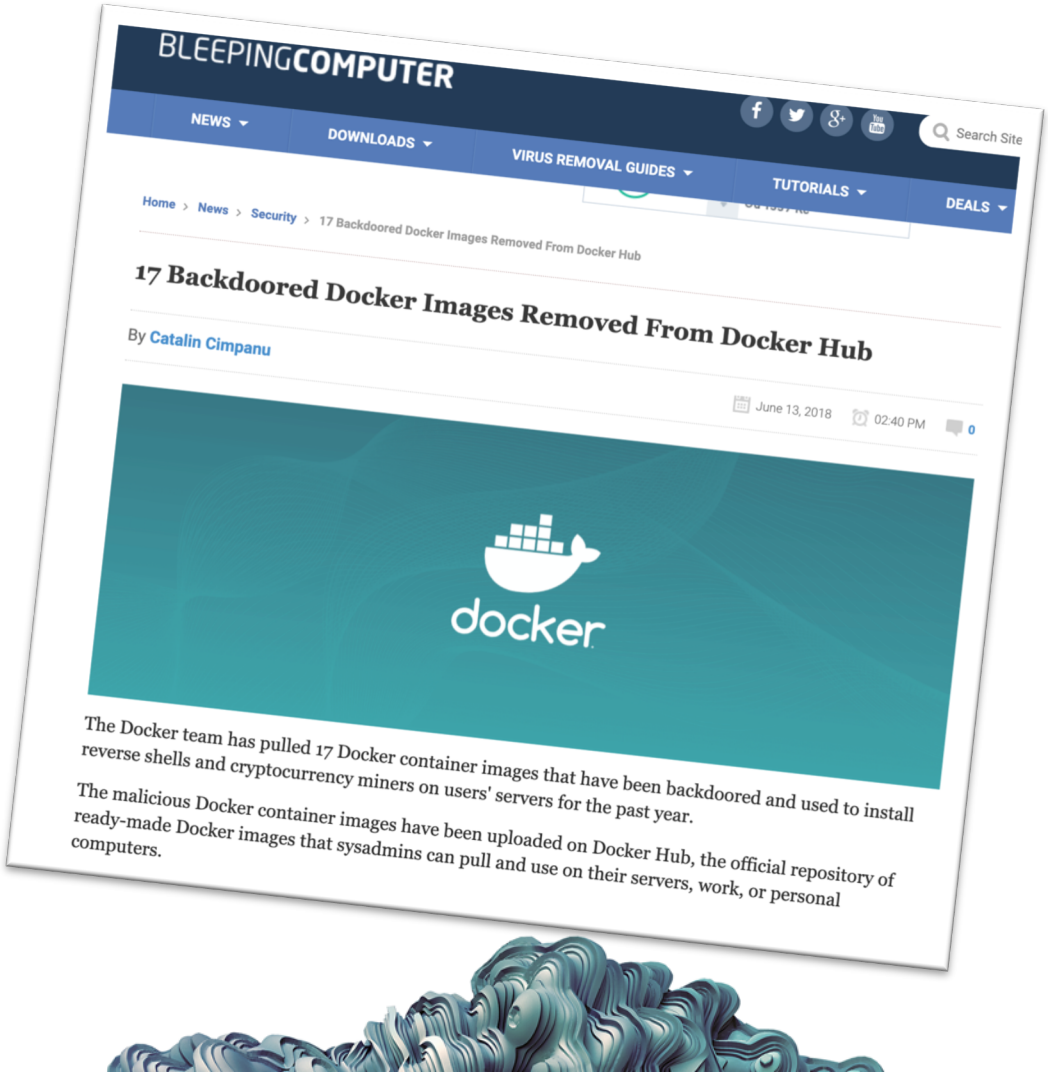


Security Risks



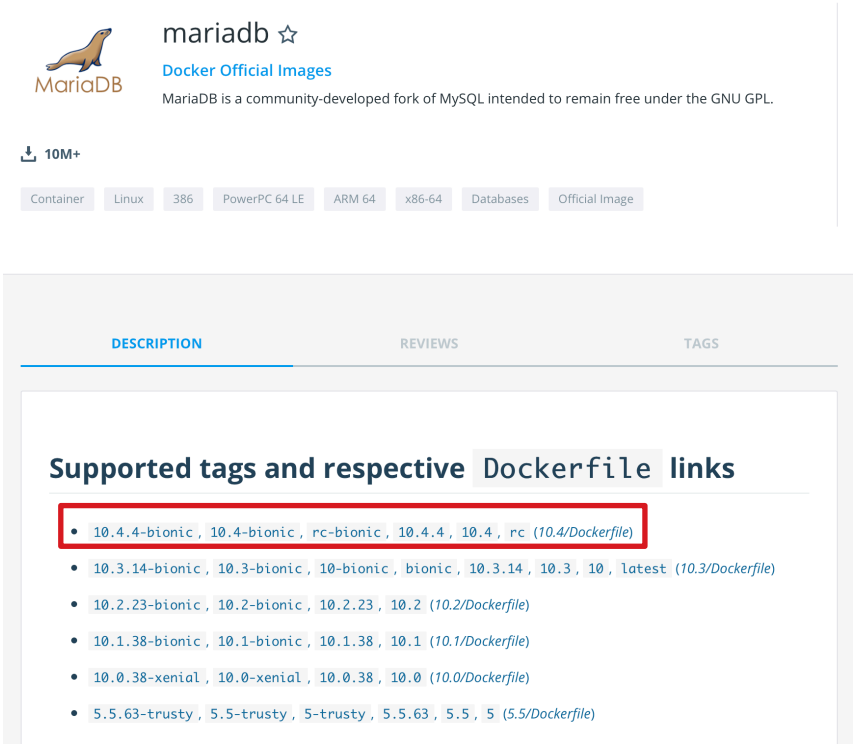


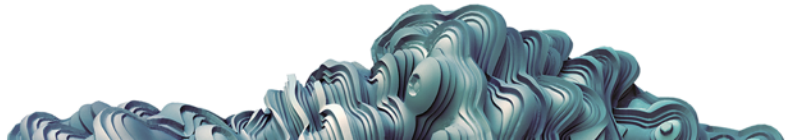




Docker image

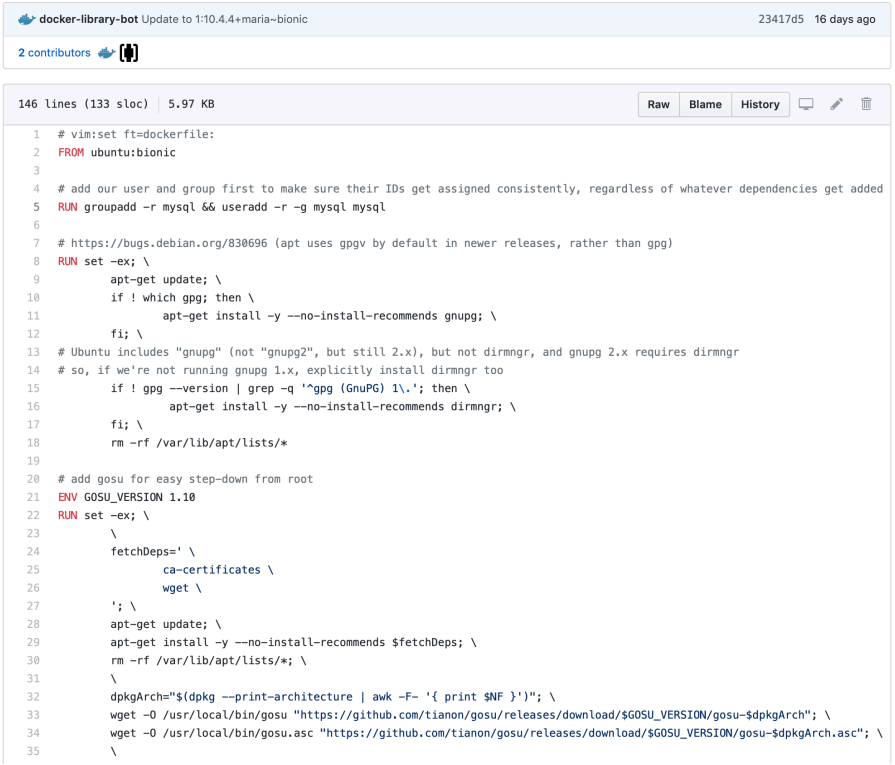
- docker pull mariadb
- docker run …





Docker image

- configuration info
  - public
  - auditable
    - for user
    - for attacker

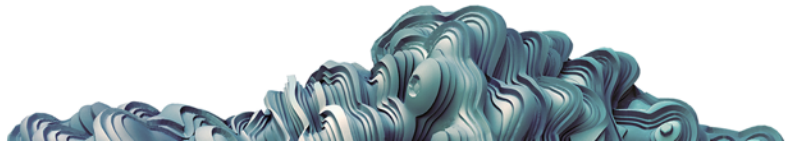


docker-library-bot Update to 1:10.4.4+maria~bionic 23417d5 16 days ago

2 contributors

146 lines (133 sloc) | 5.97 KB Raw Blame History

```
# vim:set ft=dockerfile:
FROM ubuntu:bionic

# add our user and group first to make sure their IDs get assigned consistently, regardless of whatever dependencies get added
RUN groupadd -r mysql && useradd -r -g mysql mysql

# https://bugs.debian.org/830696 (apt uses gpgv by default in newer releases, rather than gpg)
RUN set -ex; \
        apt-get update; \
        if ! which gpg; then \
                apt-get install -y --no-install-recommends gnupg; \
        fi; \
# Ubuntu includes "gnupg" (not "gnupg2", but still 2.x), but not dirmngr, and gnupg 2.x requires dirmngr
# so, if we're not running gnupg 1.x, explicitly install dirmngr too
        if ! gpg --version | grep -q '^gpg (GnuPG) 1\.'; then \
                apt-get install -y --no-install-recommends dirmngr; \
        fi; \
        rm -rf /var/lib/apt/lists/*

# add gosu for easy step-down from root
ENV GOSU_VERSION 1.10
RUN set -ex; \
        \
        fetchDeps=' \
                ca-certificates \
                wget \
        '; \
        apt-get update; \
        apt-get install -y --no-install-recommends $fetchDeps; \
        rm -rf /var/lib/apt/lists/*; \
        \
        dpkgArch="$(dpkg --print-architecture | awk -F- '{ print $NF }')"; \
        wget -O /usr/local/bin/gosu "https://github.com/tianon/gosu/releases/download/$GOSU_VERSION/gosu-$dpkgArch"; \
        wget -O /usr/local/bin/gosu.asc "https://github.com/tianon/gosu/releases/download/$GOSU_VERSION/gosu-$dpkgArch.asc"; \
        \
```

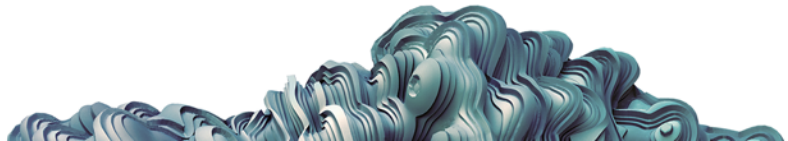
TREND MICRO

Docker image

Alpine Linux Docker images ship a root account with no password

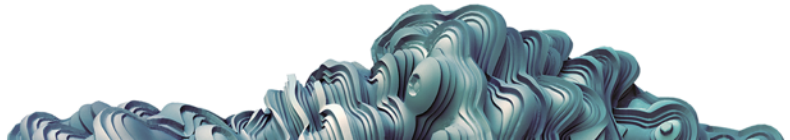
Attackers can authenticate on vulnerable systems using the root user and no password.

By Catalin Cimpanu for Zero Day | May 8, 2019 -- 21:10 GMT (22:10 BST) | Topic: Security



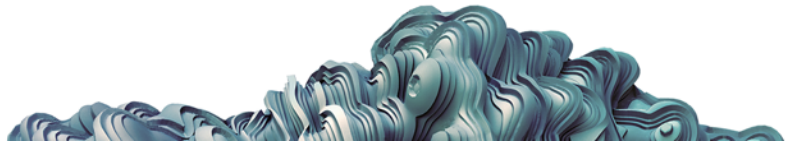
Misconfiguration

- running privileged containers
  - running as **root**
    - **interacts with kernel as root**
- network security policies
  - reaching containers/host/internal network
- volume mounting
  - fs mounting & size limits



Misconfiguration

- exposing API
  - not default
  - giving HW resources for free
  - daemon runs under root
  - targeted scans for TCP 2375/2376



Exposing API - Docker

Shodan | Developers | Book | View All...

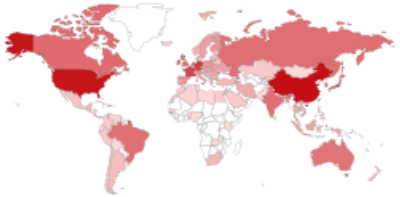
SHODAN docker Explore Pricing Enterprise Access

Exploits Maps

TOTAL RESULTS

22,319

TOP COUNTRIES



China	5,547
United States	4,513
Germany	2,065
Hong Kong	1,613
France	1,604

ERROR: The requested URL could not be retrieved

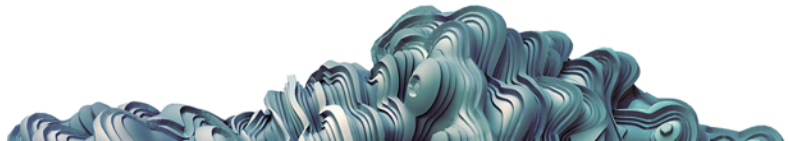
106.75.175.70
China Telecom Guangdong
Added on 2019-04-10 13:33:52 GMT
China

```
HTTP/1.1 400 Bad Request
Server: squid/3.5.27
Mime-Version: 1.0
Date: Wed, 10 Apr 2019 13:34:43 GMT
Content-Type: text/html;charset=utf-8
Content-Length: 1716
X-Squid-Error: ERR_INVALID_URL 0
Vary: Accept-Language
Content-Language: en
X-Cache: MISS from docker-e02ccp
X-Cache-Lookup: NON...
```

13.208.169.22

ec2-13-208-169-22.ap-northeast-3.compute.amazonaws.com
linux

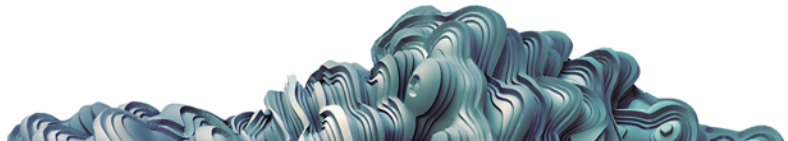
```
HTTP/1.1 404 Not Found
Content-Length: 29
```



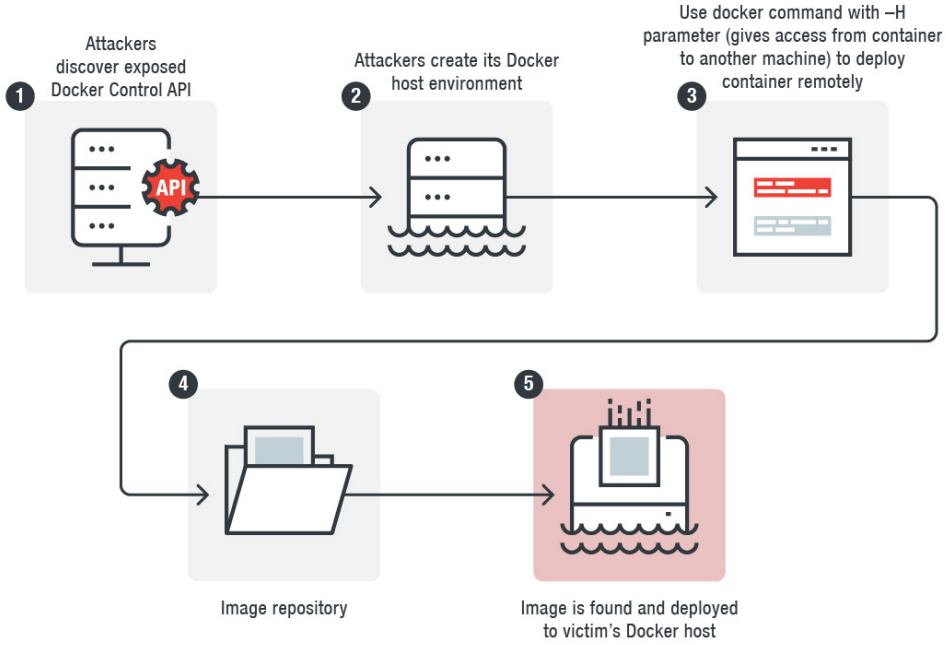
TREND MICRO

Exposing API - Docker

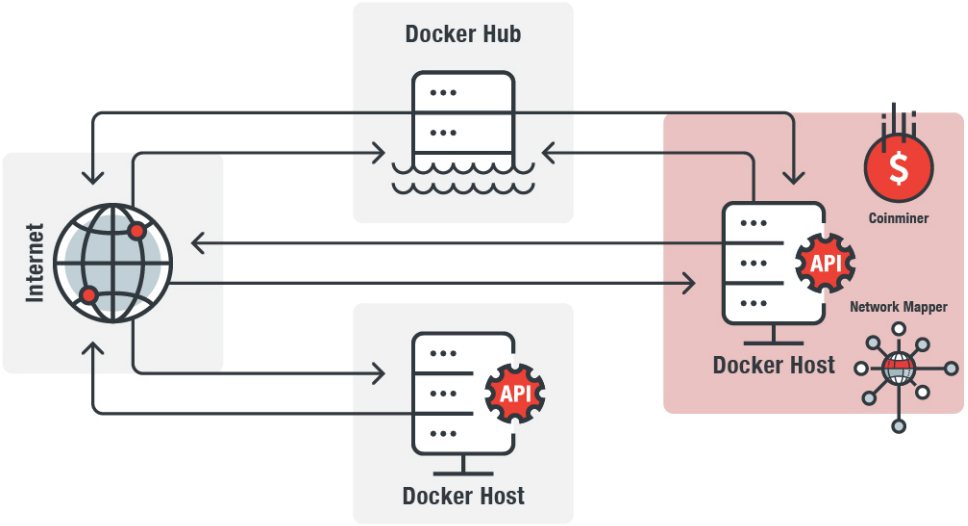
- docker –H x.x.x.x images
- docker –H x.x.x.x container ls
- docker –H x.x.x.x build –t mybot .
- docker –H x.x.x.x run –d mybot

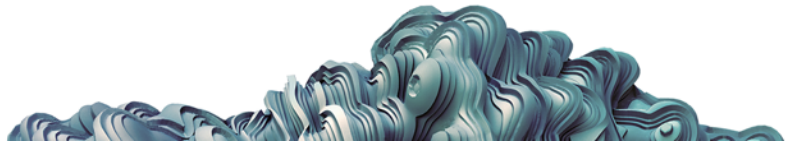


Cryptocurrency Mining case



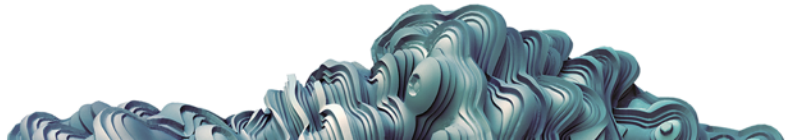
Cryptocurrency Mining case

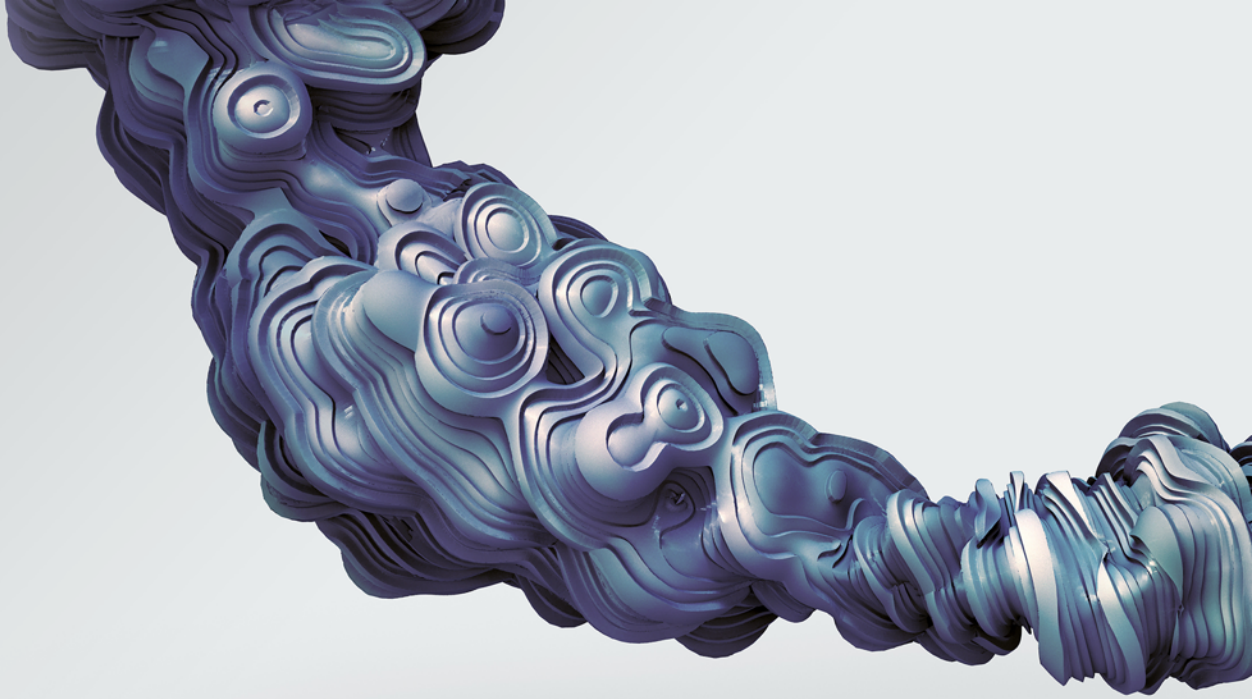




Container infiltration

- infiltrate running container
  - exploiting vulnerability
  - shell access to the container
  - explore possibilities

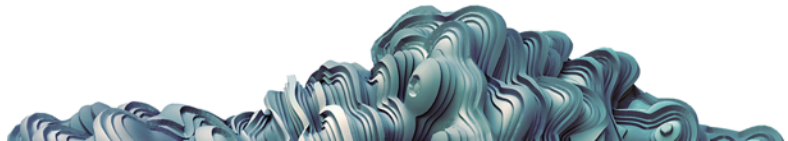




Vulnerabilities

Vulnerabilities

- Is there a vulnerability?
  - image version
    - image rebuilding
- Can it be exploited?



Vulnerabilities - demo case

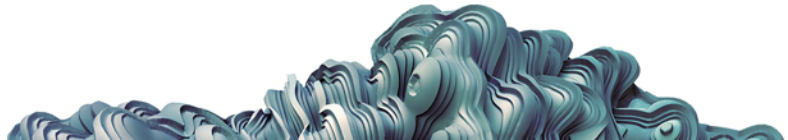
- WordPress

wp_site — Just another WordPress site

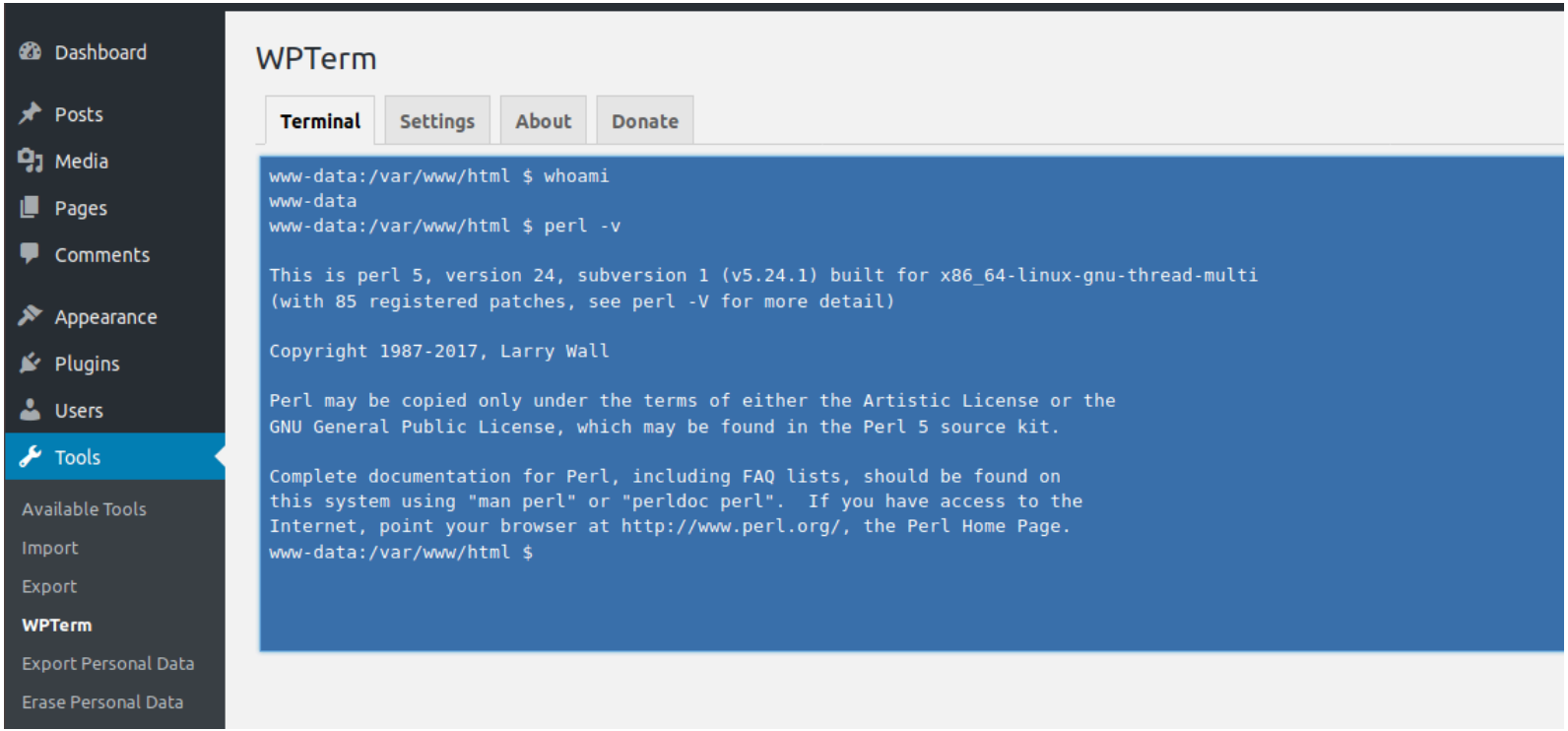
Hello world!

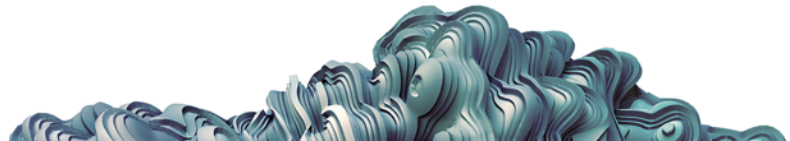
Welcome to WordPress. This is your first post. Edit or delete it, then start writing!

admin April 26, 2019 Uncategorized 1 Comment Edit



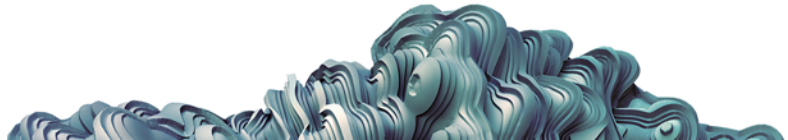
Vulnerabilities - demo case





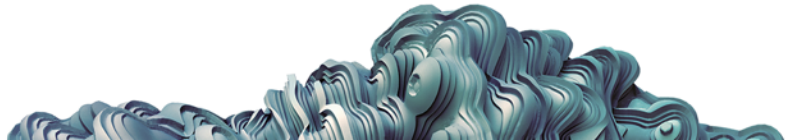
Vulnerabilities - demo case

```
www-data:/var/www/html/tmp/sbin $ ./ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.18.0.3  netmask 255.255.0.0  broadcast 172.18.255.255
        ether 02:42:ac:12:00:03  txqueuelen 0  (Ethernet)
        RX packets 7249  bytes 5152195 (5.1 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 7527  bytes 3043621 (3.0 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 176  bytes 11569 (11.5 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 176  bytes 11569 (11.5 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```



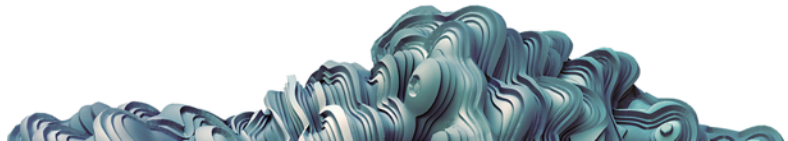
Vulnerabilities - demo case

- What can I expect?
  - other (vulnerable) containers
  - host
  - internal hidden services
  - volume mounts



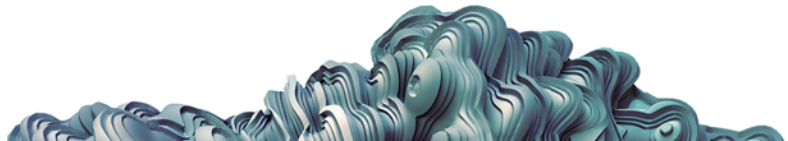
Container escape

- ultimate goal
  - escape container as root
- implications
  - owning machine



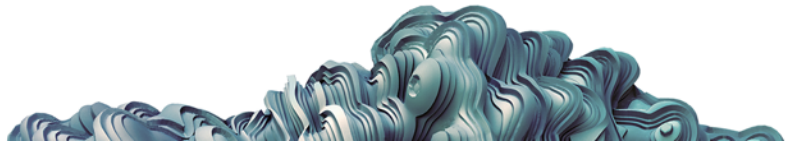
CVE-2019-5736 runC case

- container breakout
- overwriting **host** runC binary from a container



CVE-2019-5736 runC case

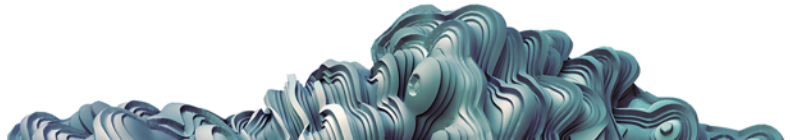
- prerequisites
  - malicious image deployment **OR** container access
  - ability to spawn docker exec
    - indirectly



CVE-2019-5736 runC case

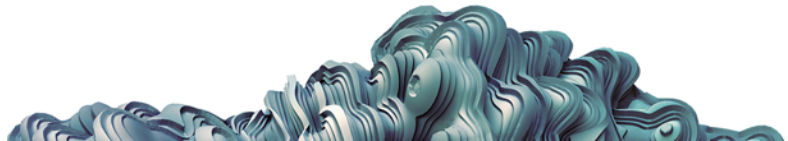
11 libcontainer/nsenter/nsexec.c

```
@@ -534,6 +534,9 @@ void join_namespaces(char *nslist)
        free(namespaces);
 }

+/* Defined in cloned_binary.c. */
+extern int ensure_cloned_binary(void);
+
 void nsexec(void)
 {
        int pipenum;
@@ -549,6 +552,14 @@ void nsexec(void)
        if (pipenum == -1)
                return;

+       /*
+        * We need to re-exec if we are not in a cloned binary. This is necessary
+        * to ensure that containers won't be able to access the host binary
+        * through /proc/self/exe. See CVE-2019-5736.
+        */
+       if (ensure_cloned_binary() < 0)
+               bail("could not ensure we are a cloned binary");
+
        /* Parse all of the netlink configuration. */
        nl_parse(pipenum, &config);
```

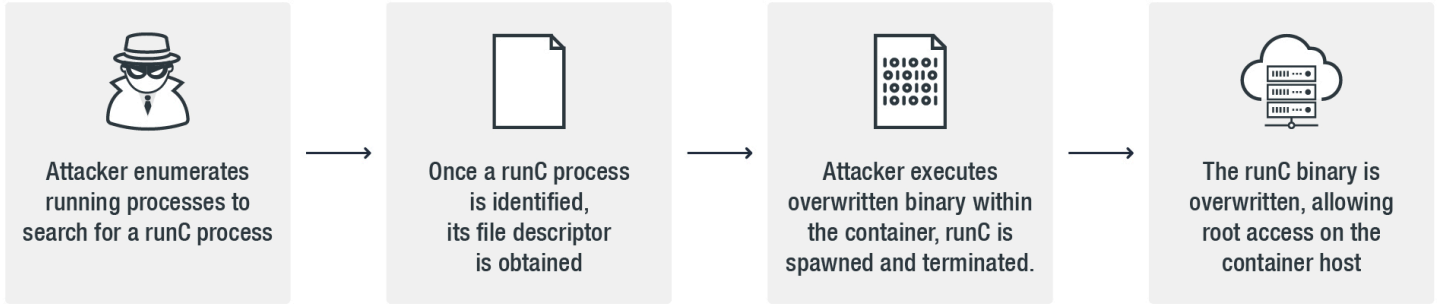


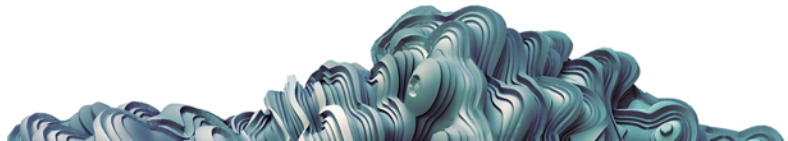
CVE-2019-5736 runC case

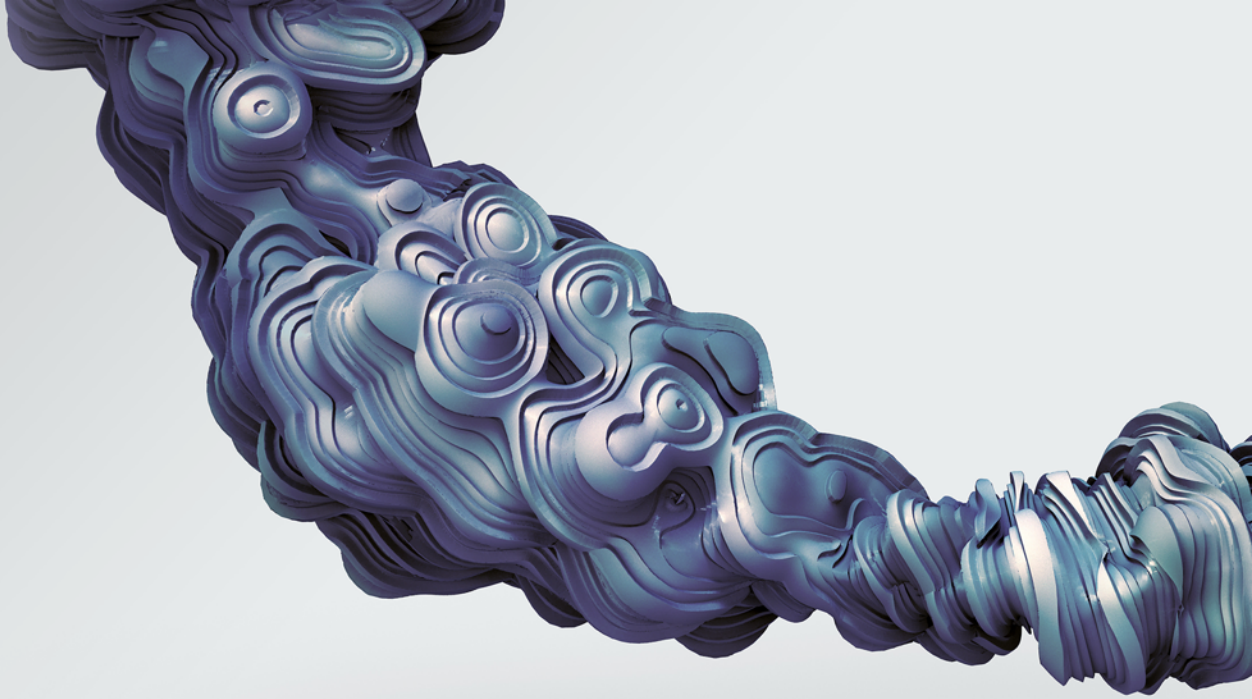
/proc/self/exe



CVE-2019-5736 runC case



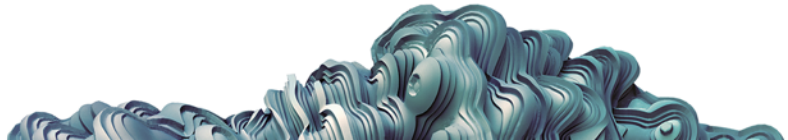




Security Risks at scale

k8

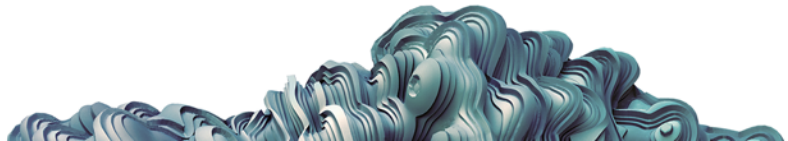
- pods
  - smallest unit
  - deployment description
- api
- **security?**



k8

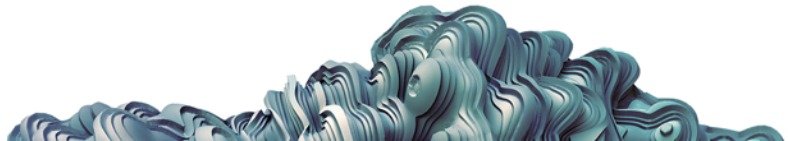
- network access
- volume access
- run as user

- **PodSecurityPolicy**



k8 – PodSecurityPolicy

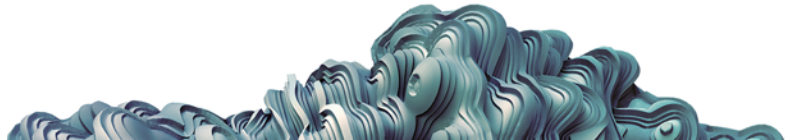
```yaml
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
  name: privileged
  annotations:
    seccomp.security.alpha.kubernetes.io/allowedProfileNames: '*'
spec:
  privileged: true
  allowPrivilegeEscalation: true
  allowedCapabilities:
  - '*'
  volumes:
  - '*'
  hostNetwork: true
  hostPorts:
  - min: 0
    max: 65535
  hostIPC: true
  hostPID: true
  runAsUser:
    rule: 'RunAsAny'
  seLinux:
    rule: 'RunAsAny'
  supplementalGroups:
    rule: 'RunAsAny'
  fsGroup:
    rule: 'RunAsAny'
```



k8 - NetworkPolicy

- ”by default, pods are non-isolated; they accept traffic from **any source**”
  - even from pwned container inside a cluster

- set up network policies !
  - https://kubernetes.io/docs/concepts/services-networking/network-policies/

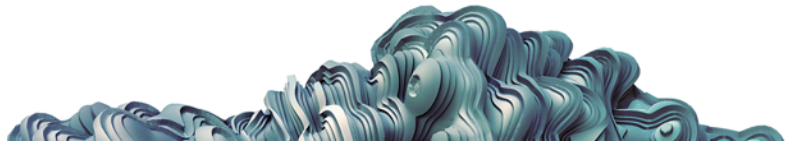


k8 – API

Securing etcd clusters

Access to etcd is equivalent to root permission in the cluster so ideally only the API server should have access to it. Considering the sensitivity of the data, it is recommended to grant permission to only those nodes that require access to etcd clusters.

https://kubernetes.io/docs/tasks/administer-cluster/configure-upgrade-etcd/



k8 – API

Shodan Developers Monitor View All...

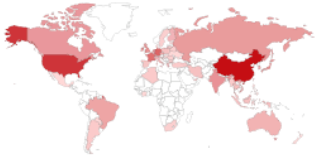
SHODAN etcd Explore Pricing Enterprise Access

Exploits Maps

TOTAL RESULTS

2,558

TOP COUNTRIES



China	1,193
United States	533
Germany	168
France	128
Singapore	66

TOP SERVICES

2379	2,429
Docker	70
8081	21
Splunk	7
HTTPS (8443)	2

TOP ORGANIZATIONS

New Service: Keep track of what you have connected to the Internet. Check out Shodan Monitor

103.107.181.17
Pega Digital Content Company Limited
Added on 2019-04-29 16:07:00 GMT
Vietnam

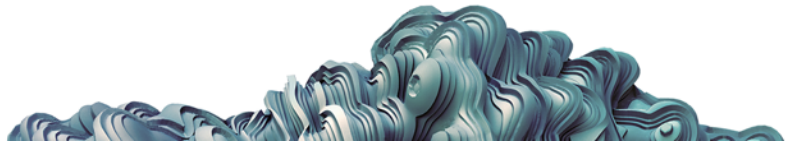
etcd
Name: default
Version: 3.3.3
Uptime: 1570h2m17.052308527s
Peers: http://172.17.0.2:2380

169.61.95.84
54.5f.3da9.ip4.static.sl-reverse.com
SoftLayer Technologies
Added on 2019-04-29 15:38:55 GMT
United States

etcd
Name: etcd0
Version: 2.3.8
Uptime: 5517h5m47.735902236s
Peers: http://10.190.60.231:2380

121.126.37.155
HAIonNet
Added on 2019-04-29 15:47:22 GMT
Korea, Republic of

etcd
Name: Enterprise-1
Version: 3.2.18
Uptime: 4m52.332596076s
Peers: http://10.10.1.154:2380



k8 – API

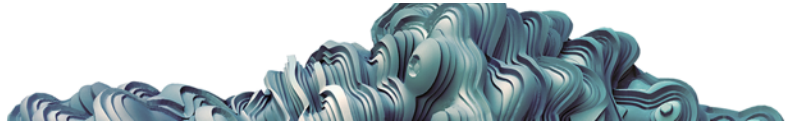
152.136.72.138
Tencent Cloud Computing (Beijing) Co.
Added on 2019-04-29 16:23:58 GMT
China

cloud

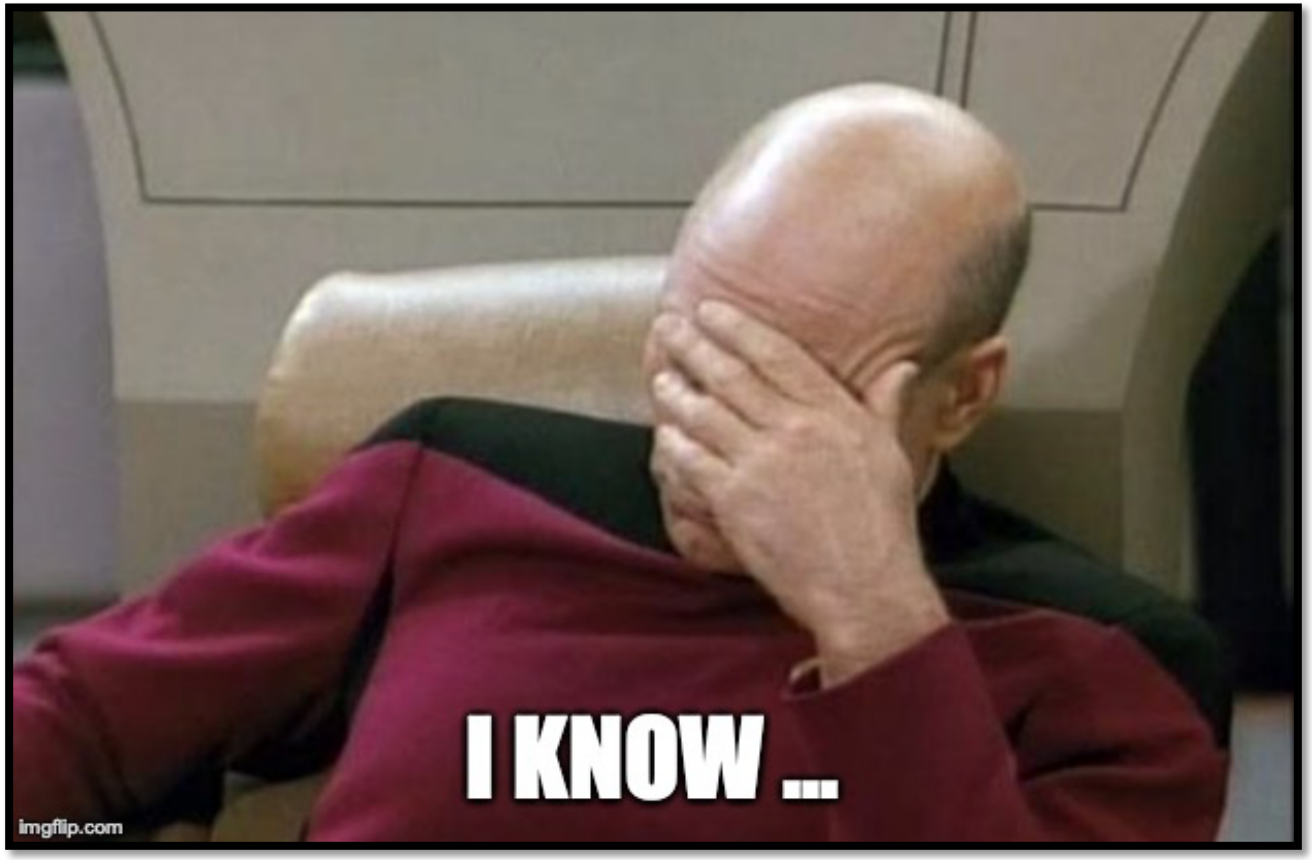
```
HTTP/1.1 200 OK
Content-Type: application/json
Date: Mon, 29 Apr 2019 16:23:57 GMT
Transfer-Encoding: chunked

a4d
{
  "paths": [
    "/api",
    "/api/v1",
    "/apis",
    "/apis/",
    "/apis/admissionregistration.k8s.io",
    "/apis/admissionregistration.k8s.io/v1alpha1",
    "/apis/adm...
```

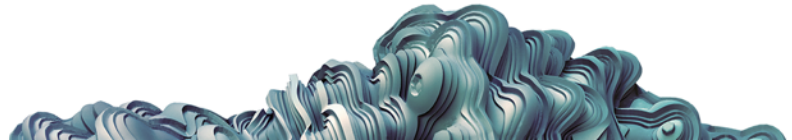
122.193.33.115
China Unicom Jiangsu
Added on 2019-04-29 15:41:42 GMT
China

```
HTTP/1.1 200 OK
Content-Type: application/json
Date: Mon, 29 Apr 2019 15:41:42 GMT
Transfer-Encoding: chunked

908
{
  "paths": [
    "/api",
    "/api/v1",
    "/apis",
    "/apis/",
    "/apis/admissionregistration.k8s.io",
    "/apis/admissionregistration.k8s.io/v1beta1",
    "/apis/apie...
```



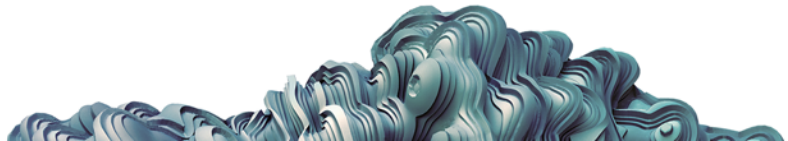
TREND MICRO





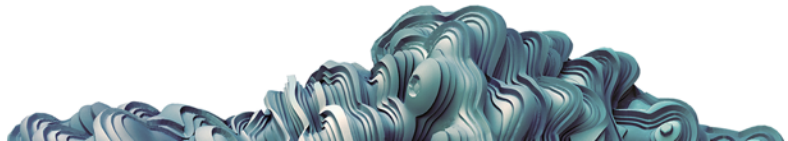
k8 – API

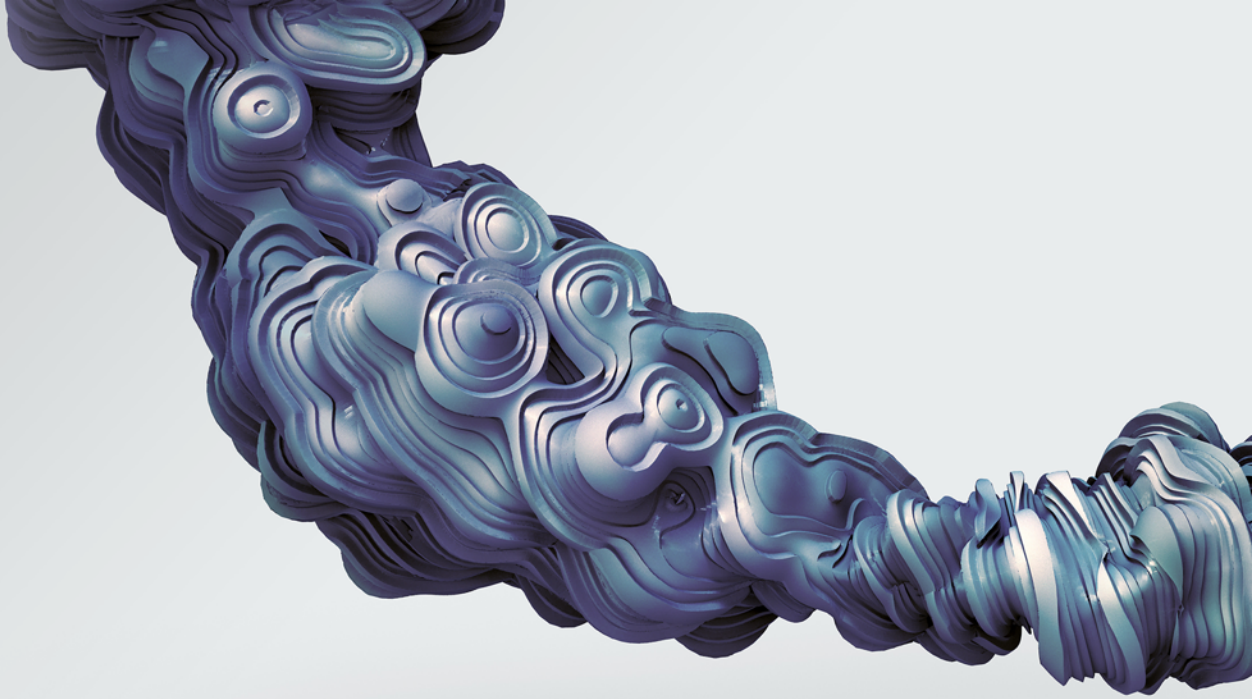
- kubectl create –f http://x.x.x.x/mypod.yaml



k8 - Dashboard

- not a default
  - “dashboard lets you create and deploy a containerized application as a Deployment and optional Service with a simple wizard. You can either manually specify application details, or upload a YAML or JSON file containing application configuration”

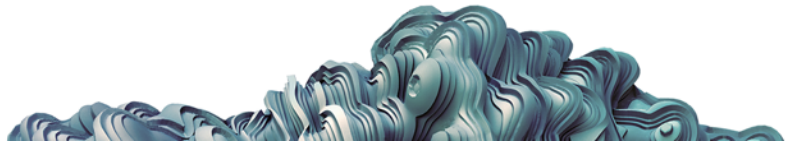




Mitigations

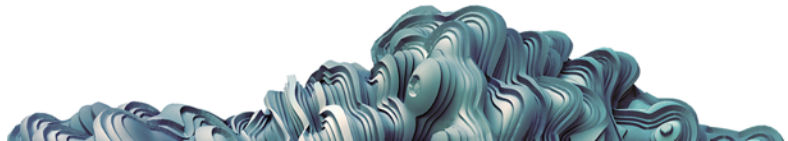
Mitigations

- images
  - using certified / verified images
  - vulnerability scanning
  - continuous image rebuilding



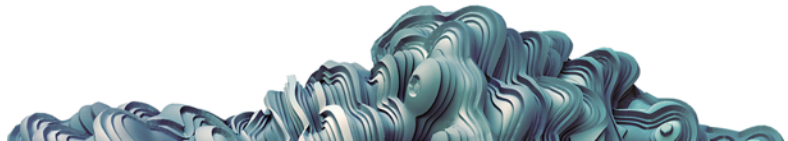
Mitigations

- containers
  - integrity scanning
  - log inspection
  - lifespan limit



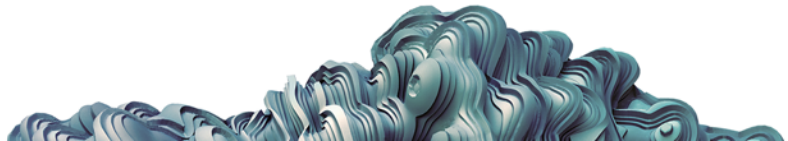
Mitigations

- follow best practices
  - principle of least privilege
  - audit tools
  - security tools
  - limit resources
- network isolation



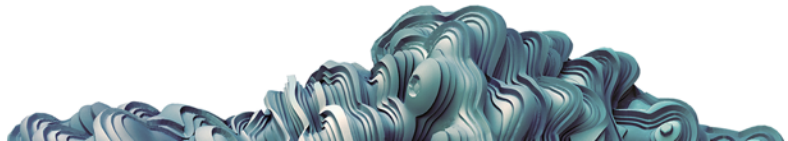
Mitigations

- App armor
- SELinux



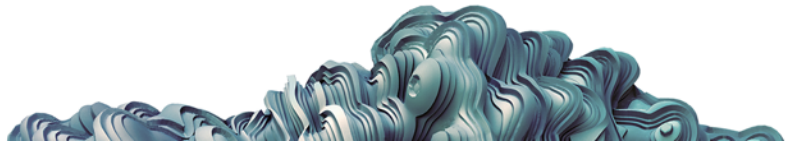
Acknowledgments

- Alfredo Oliveira
- Brandon Niemczyk
- Mohamad Mokbel

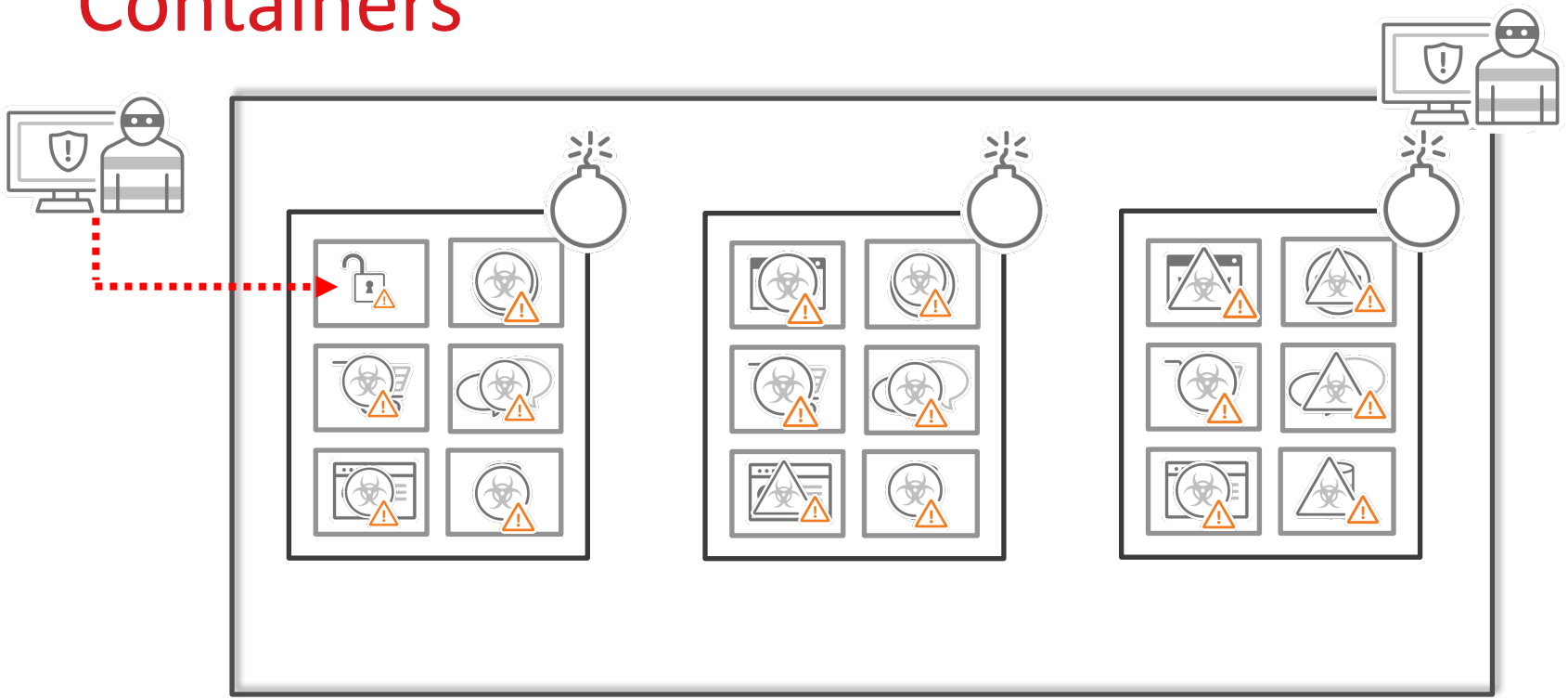


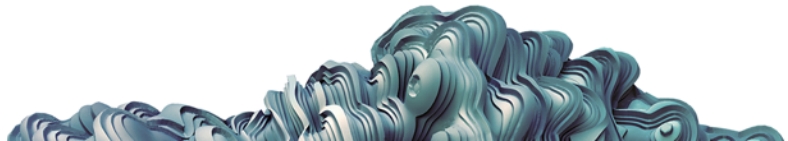
Conclusions

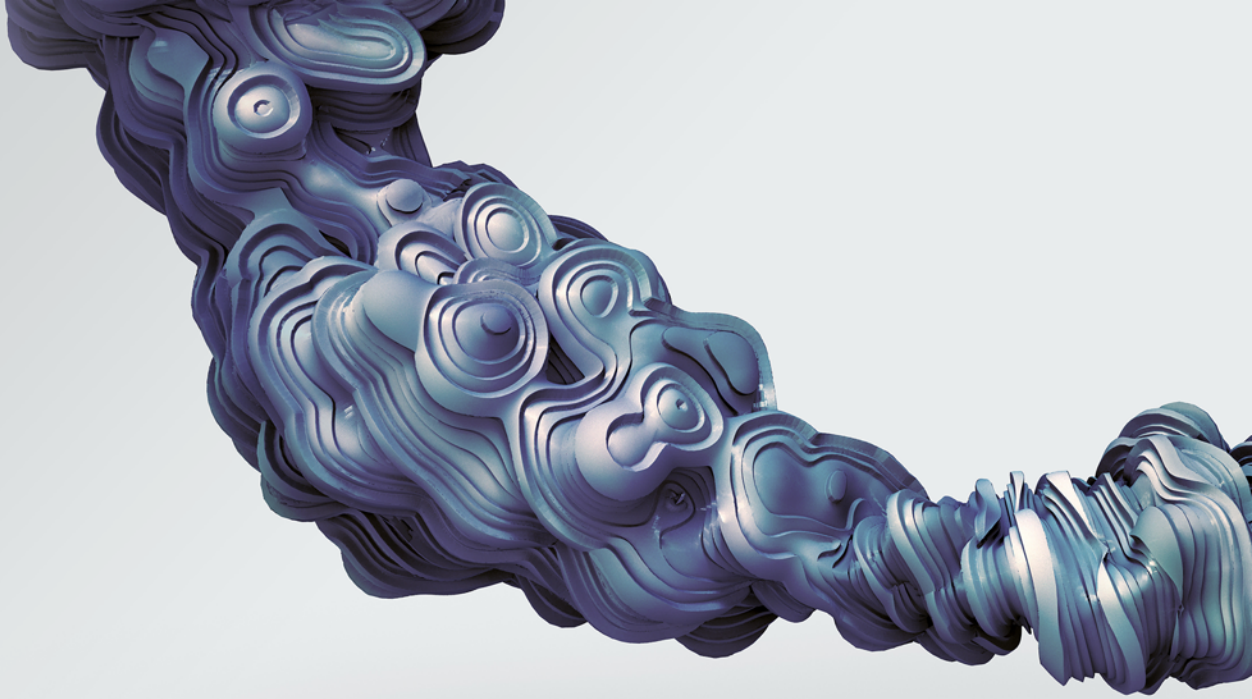
- containers are not VMs
- use best practices
- keep security in mind
- a **little flaw** can lead into a **disaster**



Containers







Questions?