Build and Deploy IoT Honeypots with Python and Docker

pyistanbul - Birkan Kolcu - 07.07.2020

About Me

- Graduated from University of Arizona (UoA) with Ms. Sc. Degree in ECE.
    - Worked as grad. research assistant in UoA for about 2 years. Reserch in Cybersecurity, IoT and data visualization.
- Graduated from Ozyegin University with Computer Science Bachelor Degree.
    - Personal projects/internships/research experience in embedded Linux systems, cloud, IoT, robotics, High Performance Computing.
    - Co-founded dusuncembu.com for helping businesses collect customer feedback in physical places.
- Github: https://github.com/ResearcherOne

Overview

1. Building a Simple Honeypot with Python
2. Dockerizing the Honeypot written in Python
3. Deployment of Dockerized Honeypot on a Raspberry Pi
4. Honeypot in action!

1- Building a Simple Honeypot with Python

1. What is honeypot?
2. Where did honeypot came from?
3. Real world honeypot examples.
4. Building honeypot with Python.

1- Building a Simple Honeypot with Python

1. What is honeypot?
    a. Imitation of real world vulnerable system.
2. Where did honeypot came from?
3. Real world honeypot examples.
4. Building honeypot with Python.

1- Building a Simple Honeypot with Python

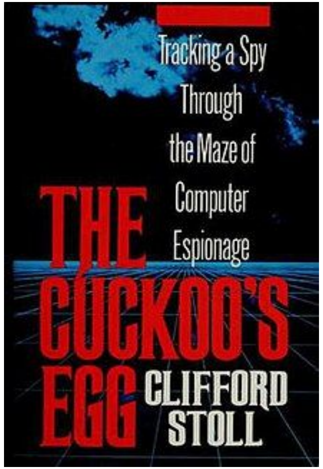
1. What is honeypot?
    a. Imitation of real world vulnerable system.
    b. Production vs research.
2. Where did honeypot came from?
3. Real world honeypot examples.
4. Building honeypot with Python.

1- Building a Simple Honeypot with Python

1. What is honeypot?
    a. Imitation of real world vulnerable system.
    b. Production vs research.
2. Where did honeypot came from?
    a. 1989, Clifford Stoll, The Cuckoo's Egg
3. Real world honeypot examples.
4. Building honeypot with Python.

1- Building a Simple Honeypot with Python

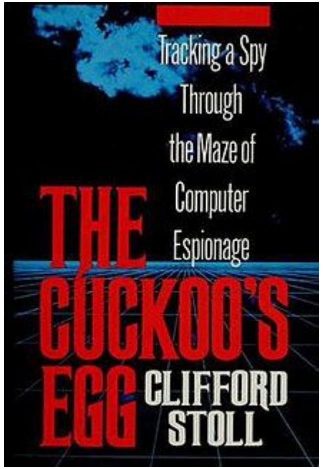
1. What is honeypot?
    a. Imitation of real world vulnerable system.
    b. Production vs research.
2. Where did honeypot came from?
    a. 1989, Clifford Stoll, The Cuckoo's Egg
3. Real world honeypot examples.
4. Building honeypot with Python.



The Cuckoo’s Egg

1- Building a Simple Honeypot with Python

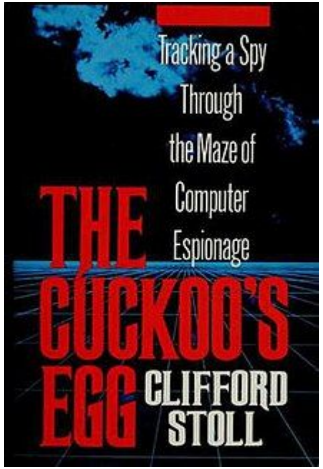
1. What is honeypot?
   a. Imitation of real world vulnerable system.
   b. Production vs research.
2. Where did honeypot came from?
   a. 1989, Clifford Stoll, The Cuckoo's Egg
3. Real world honeypot examples.
   a. sshesame, Mert Sarıca “Tuzak Sistem ile Hacker Avı”
4. Building honeypot with Python.



The Cuckoo’s Egg

1- Building a Simple Honeypot with Python

1. What is honeypot?
    a. Imitation of real world vulnerable system.
    b. Production vs research.
2. Where did honeypot came from?
    a. 1989, Clifford Stoll, The Cuckoo's Egg
3. Real world honeypot examples.
    a. sshesame, Mert Sarıca “Tuzak Sistem ile Hacker Avı”
4. **Building honeypot with Python.**



The Cuckoo’s Egg

1.4- Building honeypot with Python

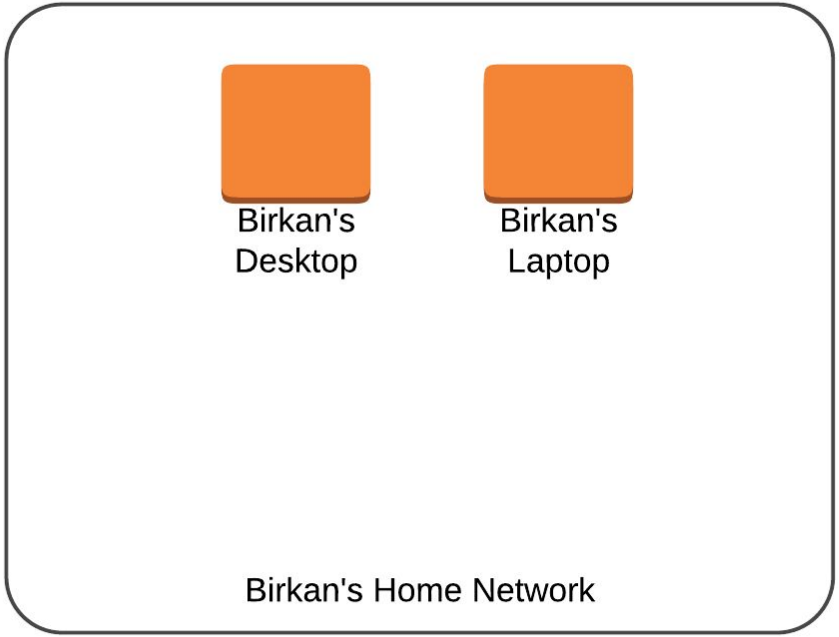
1. Our Goal
   a. Imitate ftp (file transfer protocol).

1.4- Building honeypot with Python

1. Our Goal
    a. Imitate ftp (file transfer protocol).
    b. Trigger alarm on login.

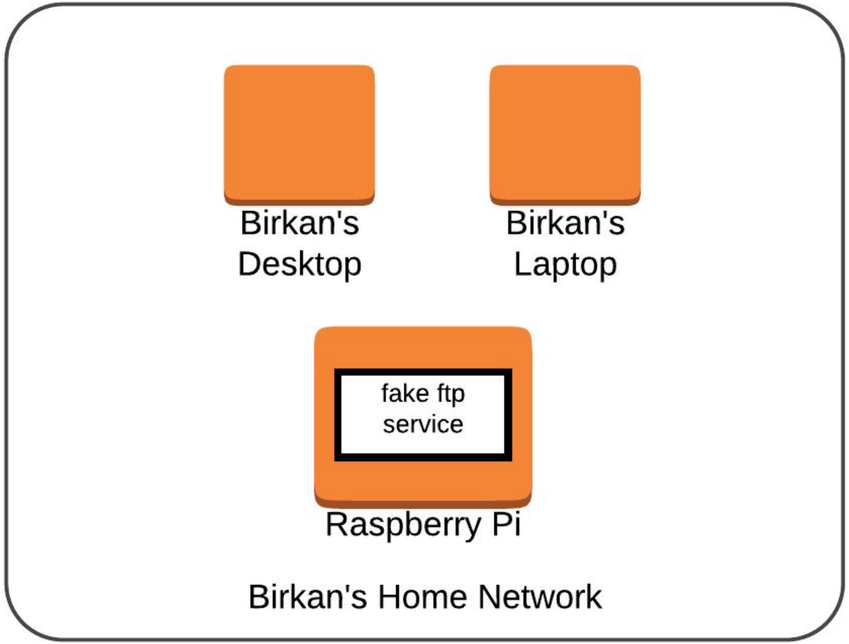
1.4- Building honeypot with Python

1. Our Goal
   a. Imitate ftp (file transfer protocol).
   b. Trigger alarm on login.



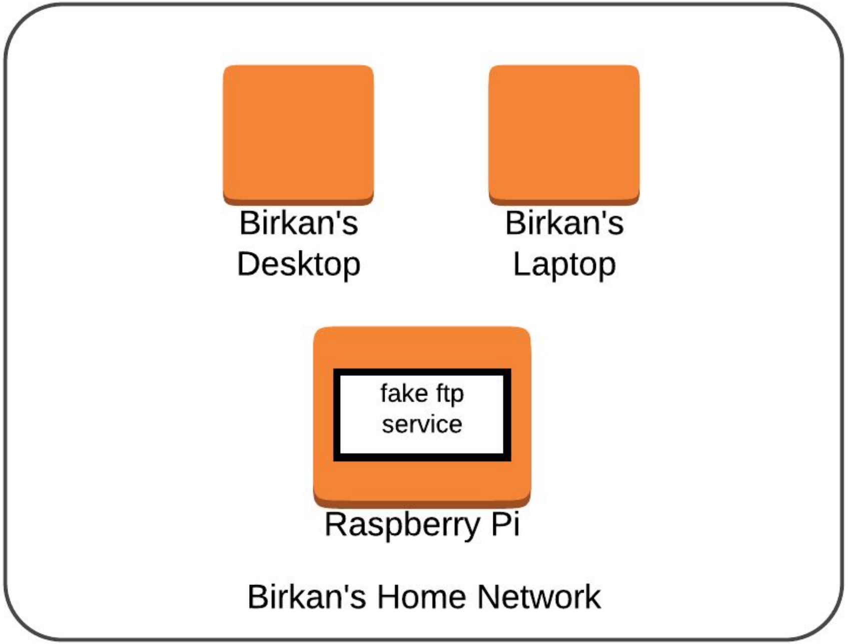
1.4- Building honeypot with Python

1. Our Goal
    a. Imitate ftp (file transfer protocol).
    b. Trigger alarm on login.



1.4- Building honeypot with Python

1. Our Goal
   a. Imitate ftp (file transfer protocol).
   b. Trigger alarm on login.
2. Let's check out source code
   a. https://github.com/ResearcherOne/iot-honeypot/blob/master/src/ftp-honeypot.py



1.4- Building honeypot with Python

https://github.com/ResearcherOne/iot-honeypot/blob/master/src/ftp-honeypot.py

THIS IS AN EXPERIMENTAL HONEYPOT. DO NOT USE IN PRODUCTION ENVIRONMENTS.

2- Dockerizing the Honeypot written in Python

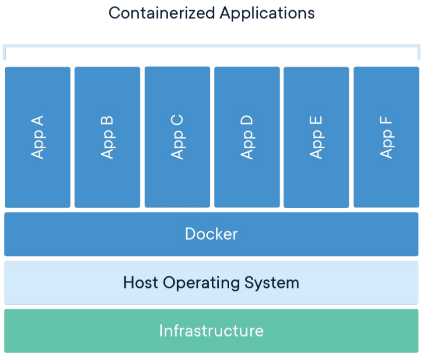
1. What is Docker and why are we using it?
2. What does “Dockerization” mean?
3. Dockerizing Python honeypot (or almost any other Python application)

2- Dockerizing the Honeypot written in Python

1. What is Docker and why are we using it?
    a. Docker is an engine for running Docker containers.
2. What does “Dockerization” mean?
3. Dockerizing Python honeypot (or almost any other Python application)

2- Dockerizing the Honeypot written in Python

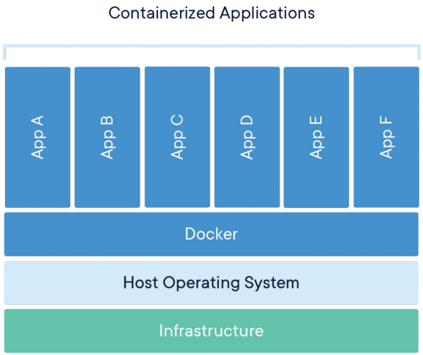
1. What is Docker and why are we using it?
   a. Docker is an engine for running Docker containers.
2. What does “Dockerization” mean?
3. Dockerizing Python honeypot (or almost any other Python application)



Docker Overview

2- Dockerizing the Honeypot written in Python

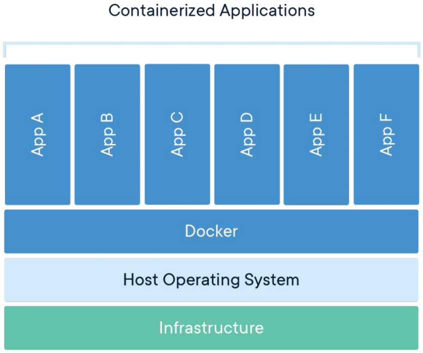
1. What is Docker and why are we using it?
   a. Docker is an engine for running Docker containers.
   b. Docker makes it super easy to deploy applications.
2. What does “Dockerization” mean?
3. Dockerizing Python honeypot (or almost any other Python application)



Docker Overview

2- Dockerizing the Honeypot written in Python

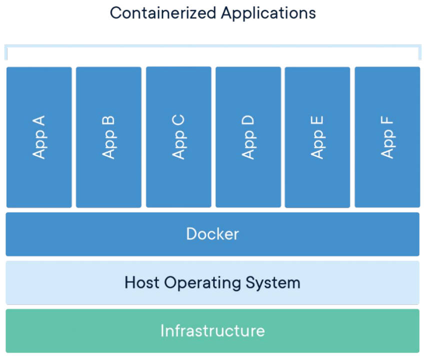
1. What is Docker and why are we using it?
    a. Docker is an engine for running Docker containers.
    b. Docker makes it super easy to deploy applications.
    c. Other use cases: reproducibility, isolation, security, etc.
2. What does “Dockerization” mean?
3. Dockerizing Python honeypot (or almost any other Python application)



Docker Overview

2- Dockerizing the Honeypot written in Python

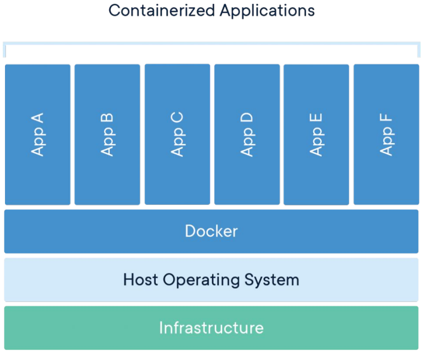
1. What is Docker and why are we using it?
   a. Docker is an engine for running Docker containers.
   b. Docker makes it super easy to deploy applications.
   c. Other use cases: reproducibility, isolation, security, etc.
2. What does “Dockerization” mean?
   a. Packing an application to run in Docker container.
3. Dockerizing Python honeypot (or almost any other Python application)



Docker Overview

2- Dockerizing the Honeypot written in Python

1. What is Docker and why are we using it?
   a. Docker is an engine for running Docker containers.
   b. Docker makes it super easy to deploy applications.
   c. Other use cases: reproducibility, isolation, security, etc.
2. What does “Dockerization” mean?
   a. Packing an application to run in Docker container.
3. **Dockerizing Python honeypot (or almost any other Python application)**



Docker Overview

2.3- Dockerizing Python honeypot

The “Normal” Way to Run App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src

2.3- Dockerizing Python honeypot

The “Normal” Way to Run App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- sudo apt install python3-pip -y
- sudo pip3 install virtualenv
- virtualenv -p python3 venv
- source venv/bin/activate

2.3- Dockerizing Python honeypot

The “Normal” Way to Run App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- sudo apt install python3-pip -y
- sudo pip3 install virtualenv
- virtualenv -p python3 venv
- source venv/bin/activate
- pip3 install sendgrid

2.3- Dockerizing Python honeypot

The “Normal” Way to Run App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- sudo apt install python3-pip -y
- sudo pip3 install virtualenv
- virtualenv -p python3 venv
- source venv/bin/activate
- pip3 install sendgrid
- export SENDGRID_API_KEY="your-top-secret-sendgrid-api-key" && ...

2.3- Dockerizing Python honeypot

The “Normal” Way to Run App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- sudo apt install python3-pip -y
- sudo pip3 install virtualenv
- virtualenv -p python3 venv
- source venv/bin/activate
- pip3 install sendgrid
- export SENDGRID_API_KEY="your-top-secret-sendgrid-api-key" && ...
- sudo -E python3 ftp-honeypot.py

2.3- Dockerizing Python honeypot

The “Normal” Way to Run App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- sudo apt install python3-pip -y
- sudo pip3 install virtualenv
- virtualenv -p python3 venv
- source venv/bin/activate
- pip3 install sendgrid
- export SENDGRID_API_KEY="your-top-secret-sendgrid-api-key" && ...
- sudo -E python3 ftp-honeypot.py

Dockerizing This App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src

2.3- Dockerizing Python honeypot

The “Normal” Way to Run App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- sudo apt install python3-pip -y
- sudo pip3 install virtualenv
- virtualenv -p python3 venv
- source venv/bin/activate
- pip3 install sendgrid
- export SENDGRID_API_KEY="your-top-secret-sendgrid-api-key" && ...
- sudo -E python3 ftp-honeypot.py

Dockerizing This App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- touch Dockerfile && nano Dockerfile

2.3- Dockerizing Python honeypot

The “Normal” Way to Run App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- sudo apt install python3-pip -y
- sudo pip3 install virtualenv
- virtualenv -p python3 venv
- source venv/bin/activate
- pip3 install sendgrid
- export SENDGRID_API_KEY="your-top-secret-sendgrid-api-key" && ...
- sudo -E python3 ftp-honeypot.py

Dockerizing This App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- touch Dockerfile && nano Dockerfile

  a. `FROM python:3`

  b. `ADD ftp-honeypot.py /`

  c. `RUN pip install sendgrid`

  d. `CMD ["python", "./ftp-honeypot.py"]`

2.3- Dockerizing Python honeypot

The “Normal” Way to Run App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- sudo apt install python3-pip -y
- sudo pip3 install virtualenv
- virtualenv -p python3 venv
- source venv/bin/activate
- pip3 install sendgrid
- export SENDGRID_API_KEY="your-top-secret-sendgrid-api-key" && ...
- sudo -E python3 ftp-honeypot.py

Dockerizing This App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- touch Dockerfile && nano Dockerfile
  a. `FROM python:3`
  b. `ADD ftp-honeypot.py /`
  c. `RUN pip install sendgrid`
  d. `CMD ["python", "./ftp-honeypot.py"]`
- sudo docker build -t ftp-honeypot .

2.3- Dockerizing Python honeypot

The “Normal” Way to Run App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- sudo apt install python3-pip -y
- sudo pip3 install virtualenv
- virtualenv -p python3 venv
- source venv/bin/activate
- pip3 install sendgrid
- export SENDGRID_API_KEY="your-top-secret-sendgrid-api-key" && ...
- sudo -E python3 ftp-honeypot.py

Dockerizing This App

- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- touch Dockerfile && nano Dockerfile
    a. `FROM python:3`
    b. `ADD ftp-honeypot.py /`
    c. `RUN pip install sendgrid`
    d. `CMD ["python", "./ftp-honeypot.py"]`
- sudo docker build -t ftp-honeypot .
- sudo docker run -p 21:21 -e "SENDGRID_API_KEY=your-top-secret-sendgrid-api-key" -e "more_env_variables_here" ftp-honeypot

3- Deployment of Dockerized Honeypot on a Raspberry Pi

1. What is Raspberry Pi and Docker Pirates?
2. How to install Docker Pirates on a Raspberry Pi?
3. How to deploy Dockerized Python Honeypot on Raspberry Pi?

3- Deployment of Dockerized Honeypot on a Raspberry Pi

1. What is Raspberry Pi and Docker Pirates?
   a. Raspberry Pi is 35$ credit-card sized computer.
2. How to install Docker Pirates on a Raspberry Pi?
3. How to deploy Dockerized Python Honeypot on Raspberry Pi?

3- Deployment of Dockerized Honeypot on a Raspberry Pi

1. What is Raspberry Pi and Docker Pirates?
    a. Raspberry Pi is 35$ credit-card sized computer.
    b. Docker Pirates is Docker-ready OS image for Pi.
2. How to install Docker Pirates on a Raspberry Pi?
3. How to deploy Dockerized Python Honeypot on Raspberry Pi?

3- Deployment of Dockerized Honeypot on a Raspberry Pi

1. What is Raspberry Pi and Docker Pirates?
   a. Raspberry Pi is 35$ credit-card sized computer.
   b. Docker Pirates is Docker-ready OS image for Pi.
2. **How to install Docker Pirates on a Raspberry Pi?**
3. How to deploy Dockerized Python Honeypot on Raspberry Pi?

3.2- How to install Docker Pirates on a Raspberry Pi?

- Source: https://blog.hypriot.com/post/releasing-HypriotOS-1-11/

3.2- How to install Docker Pirates on a Raspberry Pi?

- Source: https://blog.hypriot.com/post/releasing-HypriotOS-1-11/
- curl -O https://raw.githubusercontent.com/hypriot/flash/2.3.0/flash
- chmod +x flash
- sudo mv flash /usr/local/bin/flash

3.2- How to install Docker Pirates on a Raspberry Pi?

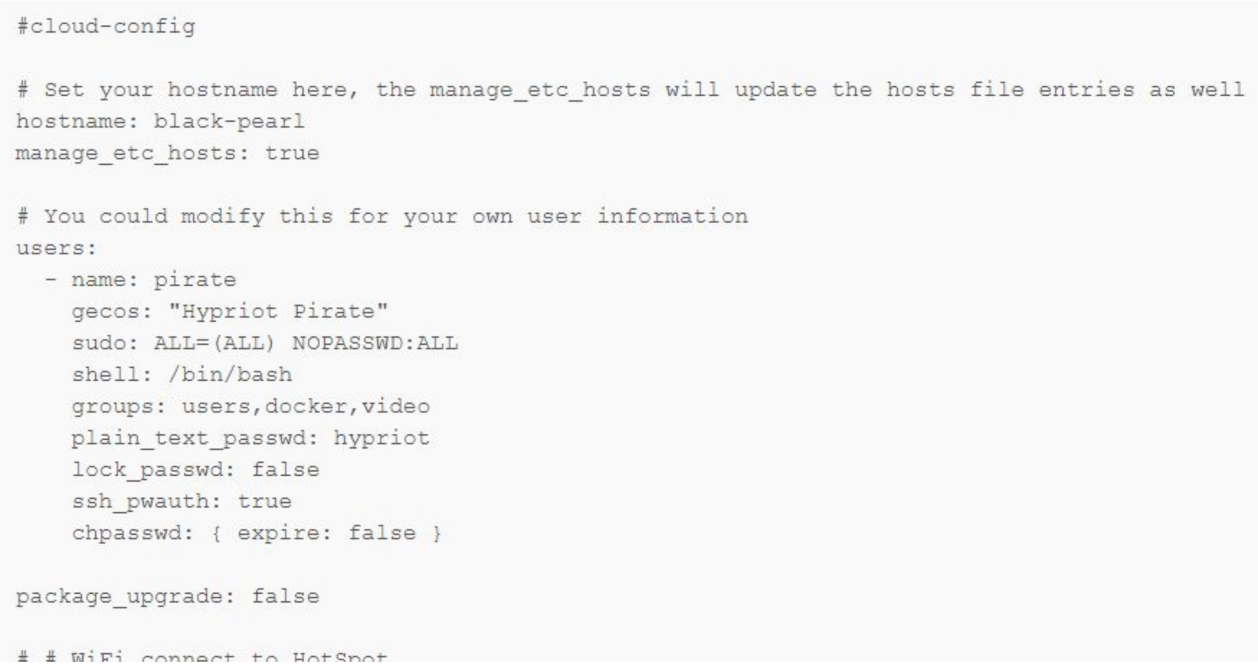
- Source: https://blog.hypriot.com/post/releasing-HypriotOS-1-11/
- curl -O https://raw.githubusercontent.com/hypriot/flash/2.3.0/flash
- chmod +x flash
- sudo mv flash /usr/local/bin/flash
- wget https://github.com/hypriot/image-builder-rpi/releases/download/v1.11.0/hypriotos-rpi-v1.11.0.img.zip

3.2- How to install Docker Pirates on a Raspberry Pi?

- Source: https://blog.hypriot.com/post/releasing-HypriotOS-1-11/
- curl -O https://raw.githubusercontent.com/hypriot/flash/2.3.0/flash
- chmod +x flash
- sudo mv flash /usr/local/bin/flash
- wget https://github.com/hypriot/image-builder-rpi/releases/download/v1.11.0/hypriotos-rpi-v1.11.0.img.zip
- flash -u wifi.yml ./hypriotos-rpi-v1.11.0.img.zip

3.2- How to install Docker Pirates on a Raspberry Pi?

wifi.yml



```
#cloud-config

# Set your hostname here, the manage_etc_hosts will update the hosts file entries as well
hostname: black-pearl
manage_etc_hosts: true

# You could modify this for your own user information
users:
  - name: pirate
    gecos: "Hypriot Pirate"
    sudo: ALL=(ALL) NOPASSWD:ALL
    shell: /bin/bash
    groups: users,docker,video
    plain_text_passwd: hypriot
    lock_passwd: false
    ssh_pwauth: true
    chpasswd: { expire: false }

package_upgrade: false

# # WiFi connect to HotSpot
```

3.2- How to install Docker Pirates on a Raspberry Pi?

- ssh pirate@black-pearl.local
- date -s '2014-12-25 12:34:56' && echo “The date should be current time”

3.2- How to install Docker Pirates on a Raspberry Pi?

- ssh pirate@black-pearl.local
- date -s '2014-12-25 12:34:56' && echo “The date should be current time”
- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src

3.2- How to install Docker Pirates on a Raspberry Pi?

- ssh pirate@black-pearl.local
- date -s '2014-12-25 12:34:56' && echo “The date should be current time”
- git clone https://github.com/ResearcherOne/iot-honeypot.git
- cd src
- sudo docker build -t ftp-honeypot .
- sudo docker run -p 21:21 -e "SENDGRID_API_KEY=your-top-secret-sendgrid-api-key" -e "more_env_variables_here" ftp-honeypot

4- Honeypot in Action!

1. The scenario:
    a. Attacker compromised wireless home network.

4- Honeypot in Action!

1. The scenario:
   a. Attacker compromised wireless home network.
   b. Performing network scanning to figure out devices and services.

4- Honeypot in Action!

1. The scenario:
    a. Attacker compromised wireless home network.
    b. Performing network scanning to figure out devices and services.
    c. Realize that an ftp service is running on a host.

4- Honeypot in Action!

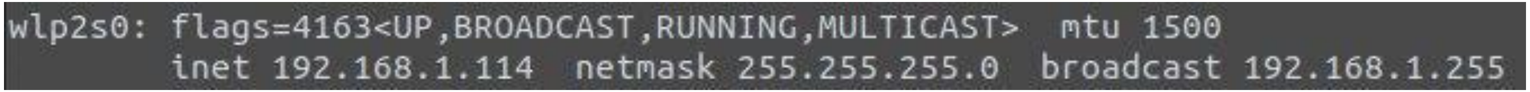
1. The scenario:
    a. Attacker compromised wireless home network.
    b. Performing network scanning to figure out devices and services.
    c. Realize that an ftp service is running on a host.
    d. Login to ftp and exploit the machine (through brute-force etc.)

4- Honeypot in Action!

1. Attacker figures out his/her ip address in home network.
    a. ifconfig

4- Honeypot in Action!

1. Attacker figures out his/her ip address in home network.
    a. ifconfig



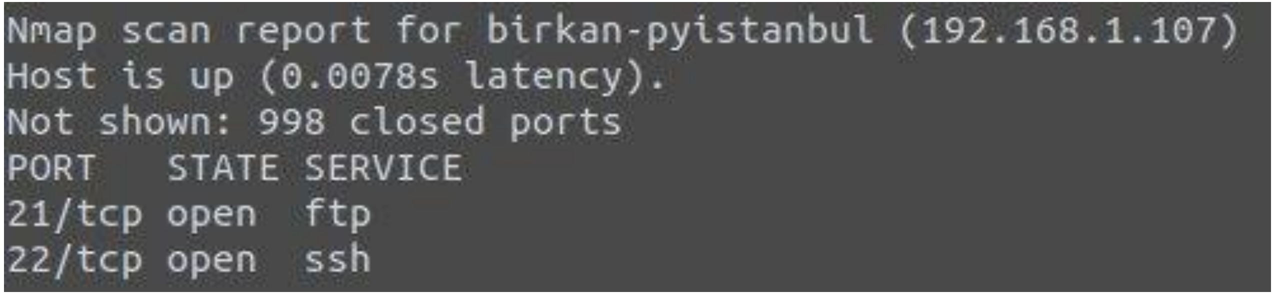
```
wlp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.114  netmask 255.255.255.0  broadcast 192.168.1.255
```

4- Honeypot in Action!

1. Performing network scanning
    a. nmap 192.168.1.0/24

4- Honeypot in Action!

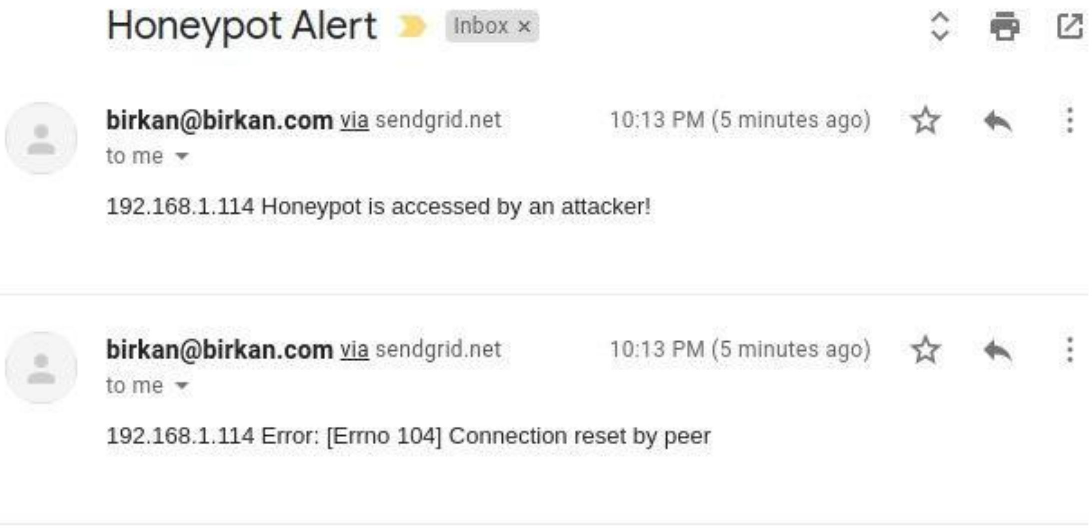
1. Performing network scanning
    a. nmap 192.168.1.0/24



```
Nmap scan report for birkan-pyistanbul (192.168.1.107)
Host is up (0.0078s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE
21/tcp open  ftp
22/tcp open  ssh
```

4- Honeypot in Action!

1. This action triggered alert on the honeypot:



Honeypot Alert

birkan@birkan.com via sendgrid.net 10:13 PM (5 minutes ago)
to me

192.168.1.114 Honeypot is accessed by an attacker!

birkan@birkan.com via sendgrid.net 10:13 PM (5 minutes ago)
to me

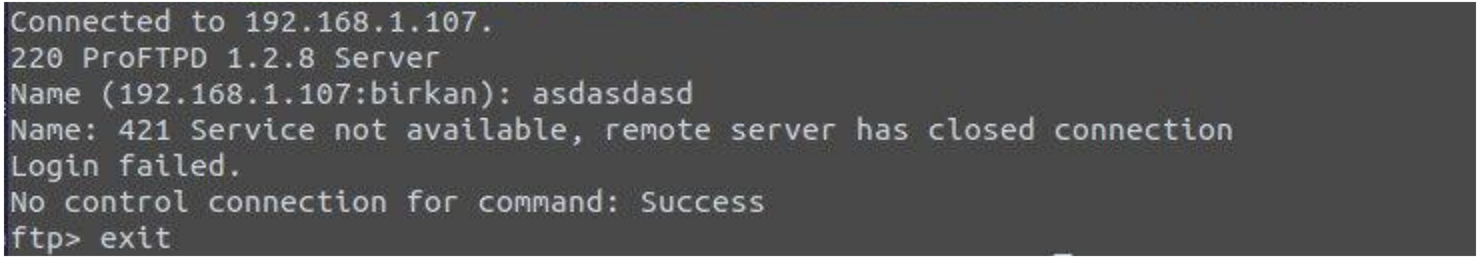
192.168.1.114 Error: [Errno 104] Connection reset by peer

4- Honeypot in Action!

1. Attacker connects to ftp service on target machine.
   a. ftp 192.168.1.107

4- Honeypot in Action!

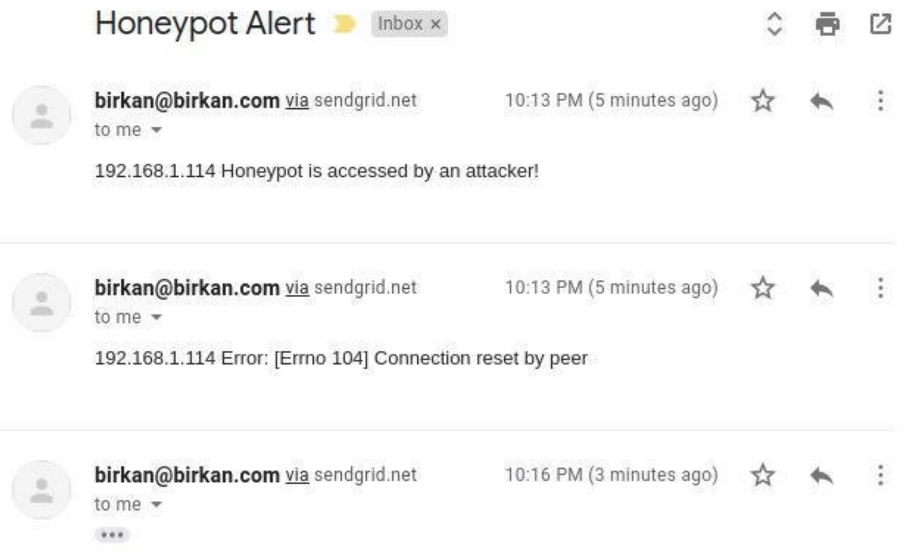
1. Attacker connects to ftp service on target machine.
   a. ftp 192.168.1.107



```
Connected to 192.168.1.107.
220 ProFTPD 1.2.8 Server
Name (192.168.1.107:birkan): asdasdasd
Name: 421 Service not available, remote server has closed connection
Login failed.
No control connection for command: Success
ftp> exit
```

4- Honeypot in Action!

1. This action also triggered alert on the honeypot:



Honeypot Alert

birkan@birkan.com via sendgrid.net 10:13 PM (5 minutes ago)
to me

192.168.1.114 Honeypot is accessed by an attacker!

birkan@birkan.com via sendgrid.net 10:13 PM (5 minutes ago)
to me

192.168.1.114 Error: [Errno 104] Connection reset by peer

birkan@birkan.com via sendgrid.net 10:16 PM (3 minutes ago)
to me

References

- https://searchsecurity.techtarget.com/definition/honey-pot
- https://en.wikipedia.org/wiki/The_Cuckoo%27s_Egg
- sshesame - https://github.com/jaksi/sshesame
- Mert Sarıca “Tuzak Sistem ile Hacker Avı” -

  https://www.mertsarica.com/tuzak-sistem-ile-hacker-avi/
- https://www.docker.com/resources/what-container
- https://www.linode.com/docs/applications/containers/when-and-why-to-use-docker/